Off-Peak Energy-Wise Link Reconfiguration for Virtualized Network Environment

Ebrahim Ghazisaeedi, Changcheng Huang, James Yan
Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada
{eghazisaeedi; hunag; jimyan}@sce.carleton.ca

***Abstract*—Energy consumption in Information and Communication Technology (ICT) is 10% of the total energy consumed in industrial countries. Recently, Virtualized Network Environment (VNE) has been emerged in this technology. Therefore, it is essential to develop novel techniques that reduce VNE's energy consumption. In this paper, we formulate a Binary Integer Linear Program (BILP) that reconfigures already allocated virtual networks to minimize VNE's link power consumption, during off-peak periods. Because the formulated BILP is $\mathcal{NP}$-hard, a novel heuristic algorithm is also suggested. The simulation results confirm the proposed solutions save notable amount of energy in VNE's substrate links, during off-peak hours.**

I. INTRODUCTION

Power consumption in Information and Communication Technology (ICT) is rapidly increasing by 6% rate a year, while it is already 2% of total power consumption in the world, and 10% of the energy consumed in industrial countries [1]. Recently, virtualization has been proposed to share resources in network environment. Virtualized Network Environment (VNE) supports the coexistence of multiple virtual networks over a single physical network [2]. Thus, it has been regarded as a promising technology for flexibly utilizing shared network resources. Consequently, it is necessary to develop novel techniques that reduce VNE's power consumption.

Nevertheless, there are few very recent works that concerned about energy consumption in VNE. Some of the proposed energy-aware optimization programs for VNE in [3]–[6] modify VNE embedding process to reduce substrate network's power consumption. However, they are not scalable, and also decrease the network's admittance level for new virtual network requests, due to the additional constraints they impose to the mapping process. Besides, the off-line heuristic algorithm proposed in our previous work [7], tries to maximize the number of sleep mode physical links during off-peak period of VNE. It finds alternative paths over the allocated virtual networks. However, it does not benefit from the possibility of modifying the virtual network allocation in order to save more energy. Moreover, the algorithm in [8] reconfigures the peak allocation of VNE, right after the embedding phase, in order to save power. But, it is an initial unformulated heuristic that does not consider the traffic load on the network. Nonetheless, it is not efficient to have the same power saving strategy for any traffic load level. Additionally, the objective in [8] is not efficient, because the heuristic re-allocates virtual links while it does not consider physical links as the actual power consumers.

Due to the simpler technical implementation and high potential of energy saving in network links, this paper is restricted to power saving solutions for links in VNE. In this paper, we formulate a Binary Integer Linear Program (BILP) that reconfigures some of the already allocated virtual links' capacities, in order to minimize VNE's link power consumption, during off-peak time. However, the formulated BILP is $\mathcal{NP}$-hard and therefore it is not scalable for large network sizes. Hence, a novel heuristic algorithm is also proposed for the same problem. The proposed heuristic is scalable to large network sizes, and because it reconfigures the already mapped virtual networks, it does not decrease network's admittance level for new virtual network requests. Besides, it saves the energy according to the off-peak traffic load.

This paper is organized as follows: A power model for physical links is defined in Section II. The BILP is formulated in Section III and the suggested heuristic is described at Section IV. The proposed energy saving solutions are evaluated on random VNEs and the results are analyzed in Section V. The paper will conclude in Section VI.

II. PHYSICAL LINK POWER MODEL

The link power consumption in VNE, comes from physical links' energy consumption. In order to find out the parameters affecting the physical link's power consumption, it is needed to study the physical link power model. We study a physical link power model that is widely used in today's communication networks [9].

This link power model considers constant amount of power consumption P^m, that is the maximum link power consumption. According to this model, the traffic load on the link does not affect the link's power consumption. Therefore, a physical link consumes a fix amount of power (P^m), if it is active, regardless of its traffic load. We call this model *Fixed* link power model. In this regard, power consumption $\tilde{p}(l_s^{i,j})$ of a physical link $l_s^{i,j}$ is equal to $\alpha(l_s^{i,j})P^m$. Where, $l_s^{i,j}$ is a substrate link connects physical node i to physical node j. $\alpha(l_s^{i,j})$ refers to $l_s^{i,j}$ state. $\alpha(l_s^{i,j})$ is 1 when the link is active, otherwise it is 0. In addition, P^m when bandwidth capacity of the physical link is in the range of 0-100Mbps, 100-600Mbps, or 600-1000Mbps is 0.48Watt, 1.00Watt, or 2.00Watt, respectively [9].

The power model determines the power saving methodology. In this paper, we assume all the physical links in the substrate network, are in the same range of physical link bandwidth capacity as defined above. According to *Fixed* link power model, an active physical link consumes a constant amount of energy. In consequence, the most effective strategy to minimize VNE's link power consumption, is setting maximum number of physical links into sleep mode.

III. BINARY INTEGER LINEAR PROGRAM

During off-peak hours, the traffic demand of virtual networks over the virtual links are decreased. Decreased traffic demand during the off-peak hours brings the opportunity of saving energy in VNE. Accordingly, given substrate network topology, mapped virtual networks' topologies, and bandwidth capacity of every substrate and virtual link, as well as off-peak traffic demands of each virtual network, it is required to find a set that contain maximum number of capable physical links to be set into sleep mode during off-peak period. Nonetheless, network still needs to support off-peak traffic demands of allocated virtual networks.

The substrate network is modeled as a directed graph $G_s = (V_s, E_s)$ where V_s is the set of substrate vertices, and E_s is the set of substrate edges. Vertices represent nodes and edges denote links in network environment. Since the graph is directed, we have higher level of flexibility in terms of rerouting the traffic flows. Similarly, the n_{th} virtual network, from the set of all involved virtual networks Φ, is also modeled as a directed graph $G_n = (V_n, E_n).V_n$ and E_n stand for n_{th} virtual network vertices and edges, respectively. $L_n = |E_n|$ denotes total number of virtual links in n_{th} virtual network. In this regard, a virtual link $l_n^{a,b}$, belonging to n_{th} virtual network, connects virtual node allocated on physical node a to virtual node allocated on physical node b. The allocated virtual links of n_{th} VN are given as a set of ordered allocated virtual node pairs $l_n^{a_m,b_m}$, $m = 1, 2, \ldots, L_n$. $l_n^{a_m,b_m}$ represents m_{th} virtual link, belonging to n_{th} VN. In addition, $\acute{r}_n^{i,j}(m)$ denotes the off-peak traffic flows through allocated virtual link capacity for $l_n^{a_m,b_m}$ on $l_s^{i,j}$. $\acute{r}_n^{i,j}(m)$ for every virtual link, on each physical link is given. During the off-peak period, the reserved capacity for $l_n^{a_m,b_m}$ on $l_s^{i,j}$ is equal to its off-peak traffic demand $\acute{r}_n^{i,j}(m)$, and the rest of available physical capacity could be shared.

We approach this problem by reconfiguring some of the already allocated virtual links' capacities. According to the discussion in Section II, the reconfiguration objective is setting maximum number of physical links into sleep mode during off-peak period. However, we reconfigure only allocated virtual links' capacities on less stressed physical links, in order to decrease possible service disruption due to reconfiguration.

The stress rate $\tilde{s}(l_s^{i,j})$ of a physical link $l_s^{i,j}$ is equal to $\frac{\eta(l_s^{i,j})}{|\Phi|}\left(\frac{\sum_{\{n|G_n\in\Phi\}}\sum_{m=1}^{L_n}\acute{r}_n^{i,j}(m)}{C_b(l_s^{i,j})}\right)$. Where, $\eta(l_s^{i,j})$ is the number of virtual networks involved in physical link $l_s^{i,j}$, and $C_b(l_s^{i,j})$ is the bandwidth capacity of substrate link $l_s^{i,j}$. $\tilde{s}(l_s^{i,j})$ denotes the intensity of involved VNs and the total off-peak traffic on the physical link. In this regard, we do not reroute the off-peak traffic of physical links with $\tilde{s}(l_s^{i,j}) \geq 0.6$. Note that the stress rate threshold (we consider 0.6 here) is adjustable. Setting a smaller $\tilde{s}$ threshold, decreases amount of power the solutions could save because smaller number of physical links are considered for power saving. But, it also reduces the service interruptions due to reconfiguration.

The defined link reconfiguration problem could be formulated as a BILP in category of multi-commodity flow problems. We assume traffic is non-splittable, in order to avoid out-of-order packet delivery. Therefore, it is not possible to reroute the off-peak traffic through multiple paths. Consequently, aggregating all the traffic in each physical link and then rerouting the bundled traffic is not suitable for this case, as there are less alternative paths and smaller search zone that could support the bundled traffic. Thus, in the context of this problem each off-peak traffic $\acute{r}_n^{i,j}(m)$ that flows through the allocated virtual link capacity on a physical link with $\tilde{s}(l_s^{i,j}) < 0.6$, is a commodity. The program reroutes the off-peak traffic $\acute{r}_n^{i,j}(m)$. Assuming $l_s^{i,j}$ and $l_s^{x,y}$ as two single substrate links, $z^{x,y}\left(\acute{r}_n^{i,j}(m)\right)$ is a binary variable that helps to find a single replaced path for each commodity $\acute{r}_n^{i,j}(m)$. $z^{x,y}\left(\acute{r}_n^{i,j}(m)\right)$ is 1, if commodity $\acute{r}_n^{i,j}(m)$ is rerouted through $l_s^{x,y}$. Otherwise, $z^{x,y}\left(\acute{r}_n^{i,j}(m)\right)$ is 0. Besides, $\beta(\acute{r}_n^{i,j}(m))$ is a binary variable. It shows $\acute{r}_n^{i,j}(m)$ status, after reconfiguration. It is 0 in the case the previously reserved capacity for $\acute{r}_n^{i,j}(m)$ is removed, after reconfiguration. Otherwise, $\beta(\acute{r}_n^{i,j}(m))$ is 1.

Consequently, the defined off-peak energy-wise link reconfiguration problem, for non-splittable traffic, could be formulated as a binary integer linear program (BILP) as follows:

Objective Function:

$$Minimize \sum_{(i,j)\in E_s} \alpha(l_s^{i,j}) \tag{1}$$

Constraints:

$$\forall x \in V_s, \forall n \in \{n|G_n \in \Phi\}, m = 1, 2, \ldots, L_n, \forall (i,j) \in E_s :$$
$$\sum_{\{y|(x,y)\in E_s\}} z^{x,y}\left(\acute{r}_n^{i,j}(m)\right) - \sum_{\{y|(y,x)\in E_s\}} z^{y,x}\left(\acute{r}_n^{i,j}(m)\right) =$$
$$= \begin{cases} 1-\beta(\acute{r}_n^{i,j}(m)) & if x = i \\ \beta(\acute{r}_n^{i,j}(m)) - 1 & if x = j \\ 0 & otherwise \end{cases} \tag{2}$$

$$\forall x \in V_s, \forall n \in \{n|G_n \in \Phi\}, m = 1, 2, \ldots, L_n, \forall (i,j) \in E_s :$$
$$\sum_{\{y|(x,y)\in E_s\}} z^{x,y}\left(\acute{r}_n^{i,j}(m)\right) + \sum_{\{y|(y,x)\in E_s\}} z^{y,x}\left(\acute{r}_n^{i,j}(m)\right) \leq 2 \tag{3}$$

$$\forall (i,j) \in E_s : \; r(l_s^{i,j}) \leq C_b(l_s^{i,j}) \tag{4}$$

where:

$$r(l_s^{i,j}) = \sum_{\{n|G_n\in\Phi\}} \sum_{m=1}^{L_n} \left\{\beta(\acute{r}_n^{i,j}(m))\acute{r}_n^{i,j}(m)\right\}$$
$$+ \sum_{(x,y)\in E_s} \sum_{\{n|G_n\in\Phi\}} \sum_{m=1}^{L_n} \left\{z^{i,j}\left(\acute{r}_n^{x,y}(m)\right)\acute{r}_n^{x,y}(m)\right\} \tag{5}$$

$$\forall (i,j) \in E_s : \; r(l_s^{i,j}) \leq B\alpha(l_s^{i,j}) \tag{6}$$

$$\forall (i,j) \in \{(i,j)|(i,j) \in E_s, \tilde{s}(l_s^{i,j}) \geq 0.6\}, \forall n \in \{n|G_n \in \Phi\},$$
$$m = 1, 2, \ldots, L_n : \; \beta(\acute{r}_n^{i,j}(m)) = 1 \tag{7}$$

Variable bounds:

$$\forall (x,y) \in E_s, \forall n \in \{n|G_n \in \Phi\}, m = 1, 2, \ldots, L_n,$$
$$\forall (i,j) \in E_s : \; \alpha(l_s^{i,j}) \in \{0,1\}, z^{x,y}\left(\acute{r}_n^{i,j}(m)\right) \in \{0,1\},$$
$$\beta(\acute{r}_n^{i,j}(m)) \in \{0,1\} \tag{8}$$

Equation 1 minimizes the number of active physical links. If the program sets a physical link into sleep mode, it needs to find a capable alternative path for each virtual link capacity that is already allocated on the link. The alternative path is identified by two constraints in Equation 2, and Equation 3. If the program decides to remove $\acute{r}_n^{i,j}(m)$, $\beta(\acute{r}_n^{i,j}(m))$ will be equal to 0. Therefore, Equation 2 needs to route a single unit of data from node i to node j. As variable $z^{x,y}\left(\acute{r}_n^{i,j}(m)\right)$

is binary, the unit of data could not be splitted. Besides, Equation 3 limits the program routing, so maximum number of incoming and outgoing flows for any node is two flows. Thus, the driven route will be a single path from node i to node j, which will be used as a replaced path for $\acute{r}_n^{i,j}(m)$. Moreover, Equation 4 ensures the total traffic on every substrate link is less than its physical bandwidth capacity. The total traffic $r(l_s^{i,j})$ on physical link $l_s^{i,j}$ is the summation of total un-reconfigured traffic ($\beta(\acute{r}_n^{i,j}(m)) = 1$) on the link as well as the diverted off-peak traffic of other links (like $l_s^{x,y}$), to $l_s^{i,j}$ through reconfiguration process. $r(l_s^{i,j})$ is calculated in Equation 5. In addition, Equation 6 makes the program linear, where B is a large integer number. B must be large enough to be greater than the largest amount of $r(l_s^{i,j})$. Furthermore, Equation 7 prevents the program to reconfigure any off-peak traffic on the physical links with $\tilde{s}(l_s^{i,j}) \geq 0.6$.

Note that the formulated BILP could be reduced to the problem discussed in [10] that is a simple two-commodity integer flow problem. It is proven at [10] that this simple two-commodity integer flow problem is $\mathcal{NP}$-hard. Hence, our formulated off-peak energy-wise link reconfiguration program is $\mathcal{NP}$-hard.

IV. Heuristic Algorithm

The discussed BILP for off-peak energy-wise link reconfiguration problem, adjusts the already allocated virtual links' capacities on less stressed physical links, during off-peak hours, when the traffic is non-splittable. Nevertheless, the formulated BILP is $\mathcal{NP}$-hard, and therefore the optimization solution is not scalable in the case of large network sizes, due to its long executing time. Hence, in this section, we propose a heuristic algorithm for the same problem. It rearranges the allocated virtual links' capacities on less stressed physical links, during off-peak period while assumes traffic is non-splittable. This heuristic is expected to set maximum number of physical links into sleep mode during off-peak time. However, the network still needs to accommodate the customers' off-peak traffic demands.

Pseudo code of the proposed heuristic algorithm is shown in Algorithm 1. The physical links with higher stress rate are more essential for connectivity and traffic demands. In this regard, first, all the physical links with $\tilde{s}(l_s^{i,j})$ less than 0.6, are sorted in ascending order based on $\tilde{s}$. The list is represented by S_L. Moreover, G_s^T is an auxiliary off-peak substrate topology. At the first phase, G_s^T is the same as substrate network topology. In the next step, the algorithm checks the possibility of removing each physical link $l_s^{i,j}$ in S_L. In this regard, there must be a single alternative path for each removed virtual link capacity that supports its off-peak traffic $\acute{r}_n^{i,j}(m)$. The algorithm uses Dijkstra algorithm, as the preferred routing algorithm, to find the shortest alternative path, while the cost of every active physical link is assumed as 1. Nonetheless, every physical link $l_s^{x,y}$ on the founded path must support the off-peak traffic $\acute{r}_n^{i,j}(m)$. The unused off-peak bandwidth capacity $\breve{C}_b(l_s^{x,y})$ on each physical link $l_s^{x,y}$ on the founded path must be equal or greater than $\acute{r}_n^{i,j}(m)$. $\breve{C}_b(l_s^{x,y})$ is equal to $C_b(l_s^{x,y}) - \sum_{\{n|G_n\in\Phi\}} \sum_{m=1}^{L_n} \acute{r}_n^{x,y}(m)$. If the algorithm does not find an alternative path, or one or multiple physical links on the founded path do not support the off-peak traffic, it does not set the physical link into sleep mode.

Algorithm 1 Off-Peak Energy-Wise Link Reconfiguration Heuristic

```
for all (i, j) such that (i, j) ∈ E_s do
    if s̃(l_s^{i,j}) < 0.6 then
        place the link in S_L in ascending order based on s̃
    end if
end for
for all (i, j) such that l_s^{i,j} is the top unchecked link in S_L do
    remove the physical link l_s^{i,j} from G_s^T
    for all n such that G_n ∈ Φ do
        for all m such that m = 1, 2, ..., L_n and ŕ_n^{i,j}(m) ≠ 0 do
            if there is an alternative path from node i to node j (by
                    Dijkstra) in G_s^T then
                for all (x, y) such that l_s^{x,y} is on the alternative path
                        do
                    C̆_b(l_s^{x,y}) = C̆_b(l_s^{x,y}) − ŕ_n^{i,j}(m)
                    if C̆_b(l_s^{x,y}) < 0 then
                        C̆_b(l_s^{x,y}) = C̆_b(l_s^{x,y}) + ŕ_n^{i,j}(m)
                        place l_s^{i,j} back to G_s^T
                        undo all the previous modifications respective to l_s^{i,j}
                        break and go for next substrate link in S_L
                    else
                        ŕ_n^{x,y}(m) = ŕ_n^{x,y}(m) + ŕ_n^{i,j}(m)
                    end if
                end for
            else
                place l_s^{i,j} back to G_s^T
                undo all the previous modifications respective to l_s^{i,j}
                break and go for next substrate link in S_L
            end if
        end for
    end for
end for
return G_s^T
```

After checking all of the physical links in S_L, G_s^T is returned as the energy-efficient off-peak substrate topology.

It could be proven that complexity of the proposed heuristic is $O\left(|E_s|^3|\Phi|^2|E_v^m|^2(|E_s| + |V_s|log|V_s|)\right)$. Where E_v^m is the set of edges of the involved virtual network with the largest number of virtual links. Consequently, the suggested heuristic algorithm is much simpler and it could be solved in a polynomial time.

V. Evaluation

The proposed energy saving solutions are supposed to reduce total link power consumption in VNE, during off-peak hours. However, they need to guarantee the off-peak traffic demands. In order to evaluate their effectiveness, several random VNE setups have been evaluated.

Substrate and virtual networks' topologies are generated by Waxman algorithm [11]. In this paper, we choose the Waxman parameters, for both substrate and virtual networks' topologies, as $\lambda = \mu = 0.5$ in the area size of 100×100. The substrate links' capacity and virtual links' peak demand are generated randomly with the uniform distribution. The bandwidth capacity of each physical link is a random amount between 100Mbps and 200Mbps, but each virtual link's peak bandwidth demand is generated randomly between 40Mbps and 80Mbps. Note that both randomly generated substrate and virtual networks are symmetric. In the next step, the created virtual nodes are mapped to the substrate nodes randomly with the uniform distribution. Afterwards, every generated virtual link's peak bandwidth demand is allocated on a substrate path through a state-of-art heuristic algorithm.

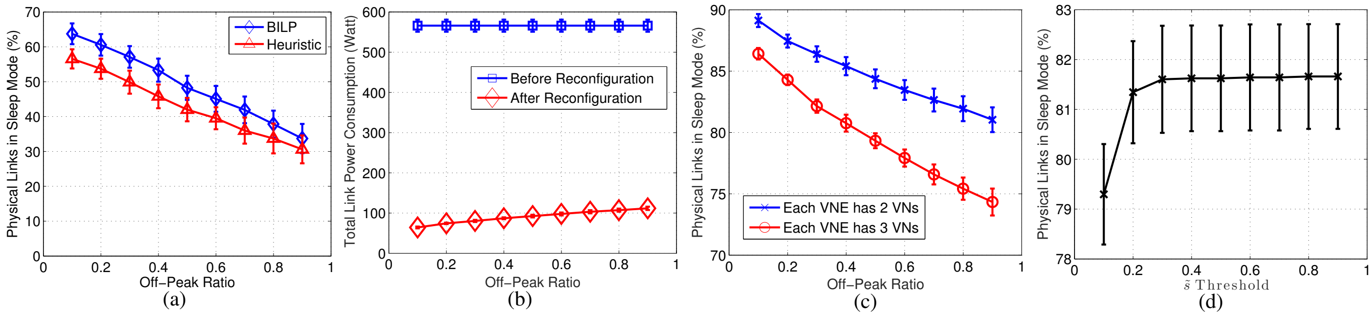

Fig. 1. (a) Physical links in sleep mode based on off-peak ratio, for the BILP and the heuristic. (b) Network's total link power consumption based on off-peak ratio, before and after applying the heuristic. (c) Physical links in sleep mode based on off-peak ratio, for heuristic when each VNE has 2 or 3 allocated VNs. (d) Physical links in sleep mode based on $\tilde{s}$ for the heuristic.

We consider two types of random simulation setups based on the network size. In this regard, every small random simulation setup contains 10 randomly generated VNEs. Each VNE has 2 random virtual networks that are allocated on a single random substrate network, while every substrate and virtual network has 10 nodes. The average number of physical links in the small random simulation setups, is 30. Furthermore, every large random simulation setup includes 10 randomly generated VNEs. Each VNE has at least 2 random virtual networks that are mapped on a single random substrate network, while the substrate network has 50 physical nodes and each virtual network has 20 virtual nodes. The average number of physical links in the large random simulation setups, is 590. Note that the average results including confidence intervals with the confidence level of 90%, are calculated for each setup.

First, the average number of physical links that are set into sleep mode by the BILP and the heuristic, are measured for different off-peak ratios. Off-peak ratio is the fraction of network off-peak traffic rate by its peak traffic rate. This is tested on a small random simulation setup, and the results are shown in Figure 1a. The results prove both of the solutions set notable number of physical links into sleep mode, during off-peak period. Besides, the heuristic works closely to the optimum points set by the BILP. Note that incrementing off-peak ratio, increases the amount of traffic the solutions need to reroute, and therefore they will be more limited in terms of finding alternative paths. Consequently, they could put fewer number of links into sleep mode, by increasing off-peak ratio.

Second, we measured network's total link power consumption for different off-peak ratios, before and after applying the proposed heuristic. This is tested on a large random simulation setup, and the results are presented in Figure 1b. The results confirm the heuristic reduces VNE's total link power consumption, effectively.

Third, the effect of different numbers of mapped VNs over the substrate network, on the heuristic's outcome is tested on a large random simulation setup. The average number of physical links the heuristic puts into sleep mode, for different off-peak ratios, when each VNE has 2 or 3 VNs are shown in Figure 1c. The heuristic must find an alternative path for each allocated virtual link capacity on the physical links, which it sets into sleep mode. Incrementing the number of mapped VNs, increases the number of allocated virtual links. Therefore, the heuristic is more limited, and it puts fewer number of physical links into sleep mode.

Fourth, the BILP and the heuristic do not reconfigure virtual links' capacities on substrate links with $\tilde{s}(l_s^{i,j}) \geq 0.6$. Figure 1d studies the effect of changing $\tilde{s}$ threshold on heuristic's outcome, over a large random simulation setup, when off-peak ratio is 0.5. By decreasing $\tilde{s}$ threshold, fewer physical links are considered for energy saving. Consequently, the heuristic puts fewer number of physical links into sleep mode.

VI. Conclusion

ICT's energy consumption is growing fast. Recently, VNE has been emerged in this technology. Therefore, it is essential to develop novel techniques that decrease VNE's power consumption. In this paper, we formulated a BILP that minimizes link power consumption in VNE, during off-peak period. Nonetheless, since the defined BILP is $\mathcal{NP}$-hard, a novel heuristic algorithm for the same problem is also suggested. Simulation results confirm the proposed energy saving solutions are noticeably effective, and the heuristic works closely to the optimum point.

References

[1] G. eSustainability Initiative, *SMART 2020: Enabling the low carbon economy in the information age*. Climate Group, 2008.

[2] N. Chowdhury and R. Boutaba, "A survey of network virtualization," *Computer Networks, Elsevier*, vol. 54, no. 5, pp. 862–876, 2010.

[3] A. Fischer, M. T. Beck, and H. D. Meer, "An approach to energy-efficient virtual network embeddings," in *International Symposium on Integrated Network Management (IM), IFIP/IEEE*, 2013, pp. 1142–1147.

[4] S. Su, Z. Zhang, X. Cheng, Y. Wang, Y. Luo, and J. Wang, "Energy-aware virtual network embedding through consolidation," in *Computer Communications Workshops (INFOCOM WKSHPS), IEEE*, 2012, pp. 127–132.

[5] B. Wang, X. Chang, J. Liu, and J. K. Muppala, "Reducing power consumption in embedding virtual infrastructures," in *Globecom Workshops (GC Wkshps), IEEE*, 2012, pp. 714–718.

[6] J. F. Botero, X. Hesselbach, M. Duelli, D. Schlosser, A. Fischer, and H. D. Meer, "Energy efficient virtual network embedding," *Communications Letters, IEEE*, vol. 16, no. 5, pp. 756–759, 2012.

[7] E. Ghazisaeedi, N. Wang, and R. Tafazolli, "Link sleeping optimization for green virtual network infrastructures," in *Globecom Workshops (GC Wkshps), IEEE*, 2012, pp. 842–846.

[8] J. F. Botero and X. Hesselbach, "Greener networking in a network virtualization environment," *Computer Networks, Elsevier*, vol. 57, no. 9, pp. 2021–2039, 2013.

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

[10] S. Even, A. Itai, and A. Shamir, "On the complexity of time table and multi-commodity flow problems," in *16th Annual Symposium on Foundations of Computer Science, IEEE*, 1975, pp. 184–193.

[11] B. M. Waxman, "Routing of multipoint connections," *Selected Areas in Communications, IEEE*, vol. 6, no. 9, pp. 1617–1622, 1988.