www.ietdl.org

Published in IET Networks
Received on 6th January 2014
Revised on 25th June 2014
Accepted on 29th July 2014
doi: 10.1049/iet-net.2014.0005

ISSN 2047-4954

Robustness and energy efficiency – a logical multi-topology scheme for time-varying traffic in IP networks

Jin Wang, Sheng Wang, Shizhong Xu

Key Laboratory of Optical Fiber Sensing and Communication, Education Ministry of China, University of Electronic Science and Technology of China, Chengdu 611731, People's Republic of China
E-mail: wangjin80uestc@gmail.com

Abstract: This study deals with a strategy to save energy in an Internet protocol (IP) network allowing different subsets of IP links to be put in the sleep mode during different traffic periods. The proposed solution involves a three-phase algorithm. In the first phase, time-varying critical traffic matrices from a large collection of real traffic matrices are obtained. In the second phase, a local search heuristic algorithm is utilised that is based on the link weight setting for the energy-saving optimisation corresponding to the critical traffic matrices. In the last phase, the real traffic flows are routed with the corresponding logical topologies. According to the performance studies that are conducted in a synthetic network and actual IP networks, this approach can achieve significantly improved energy efficiency while certain max link utilisation performance specifications and requirements are satisfied.

1 Introduction

Recent years have witnessed a rapid increase in the development of Information & Communication Technology (ICT) industry. Meanwhile, significant increase in the energy consumption and environmental challenges has led to more attention and interest by the scientific community. For an Internet Service Provider (ISP) the network power consumption is practically constant, irrespective of the traffic fluctuations. However, actual traffic follows the pattern of significant day/night variations and changes [1]. Many devices are not efficiently utilised during the off-peak hours when the traffic flow is low. Apparently this represents an opportunity for energy saving policies [2, 3], since many devices (such as routers and links) are powered on without being fully utilised. Carefully selecting subset of these resources to be switched off has little negative impact on the provided and guaranteed Quality of Service (QoS).

In the literature, various approaches have been proposed to reduce the energy consumption in communication and traffic networks. (see e.g. [4, 5] for an overview). The proposed approaches can be divided into two main categories: namely power proportional techniques and sleep mode approaches. The former approaches adapt the capacity (and thus consumption) of the devices to the actual load, and the latter approaches shut down the underutilised devices or make devices in sleep mode as much as possible. Although the first approach will has deep modifications in the design of the hardware components, the second approach requires networking devices to carefully distribute the extra load that results from placing some devices into the sleep mode and do not make network unstable and make the QoS declined.

1.1 Related work

Many approaches tackle the minimisation of the power consumption by placing network elements, such as routers and links, into the sleep mode, resulting in large energy savings, starting from the pioneering work of Gupta and Singh [6] or more recently in [7–11].

Cianfrani *et al.* [7] classified network nodes into different types, calculated the shortest paths for these different types of nodes and then obtained the result that which links can be closed. That is, by modifying the routing protocol, using as few links as possible to meet the traffic demands, and putting the idle link ports into the sleep mode, the energy-saving purpose is achieved. However, Cianfrani *et al.* [7] did not consider how to avoid congestion by waking up these link ports when new traffic demands are coming.

For backbone networks, the energy consumption is mainly concentrated in the node devices. Simply turning off idle or underutilised links will destroy the connectivity of topology, and then the overall network performance will be affected. Considering that the backbone link is often physically one bundle of cables, literature [8] established an integer linear programming (ILP) model and obtained the result that closing part of the cables can save energy and will not destroy the connectivity of topology.

Bonetto *et al.* [9] divided the day into smaller time periods. Each period was characterised by a traffic matrix (TM) used to design the logical topology (LT) for the current period. Least flow algorithm of [12] and genetic algorithm proposed in [13] were used, respectively, for configuring the network for each of these periods allowing to efficiently utilise resources, and consequently power can be saved by switching off some devices. One issue of this approach is that the network has to be reconfigured between two consecutive time periods frequently. Another issue is that the TMs are directly sampled from the original data which do not represent all TMs in a day.

Bianzino *et al.* [10, 11] are distributed on-line algorithms, which do not rely on the knowledge of the current TM is assumed, while they have the same issue with [9], that is, frequently adjusting link state which make these strategies not so robust as the off-line algorithms.

All these sleep mode methods can be classified into two groups: link-sleeping (only switching off links) and node–link-sleeping (switching off nodes and links). Bianzino *et al.* [14] proposed a mathematics model for node–link-sleeping method but it only evaluated the link-sleeping method in its experiment. GreenTE [15] maximised the number of links that can be put into sleep under given performance constraints such as link utilisation and packet delay. Energy saving IP routing (ESIR) [16] reduced the overall set of active network links by sharing a shortest path tree between neighbour routers. Francois *et al.* [17] is a robust algorithm to determine the window size of the configuration duration of the reduced topology, which was produced by removing network links using a greedy heuristic algorithm. Francois *et al.* [18] is an enhancement version of [17]. Francois *et al.* [18] combined the link-removing and link weights adjusting together to achieve a near optimal solution. Mumey *et al.* [19] minimised the total power consumption by turning off unused cables in bundled links and nodes and presented an ILP to provide optimal solutions. Chiaraviglio *et al.* [20] turned off network nodes and links while still can guarantee full connectivity and maximum link utilisation constraints. It derived complex formulations that can scale up large networks to middle-sized networks and furthermore provided an efficient heuristics algorithm. Matsuura [21] creates bypass routes between nodes on the Steiner tree to reduce traffic congestion between nodes when given a certain TM. Lee *et al.* [22] are methods, which making nodes and links sleeping. In all these methods, Francois *et al.* [18] and Lee *et al.* [22] are the most similar ones with our proposal. However, Francois *et al.* [18] did not take the node state into account and while Lee *et al.* [22] just only tackled the power optimal problem subjected to assuming there is only one knowing TM, so it cannot be adapted to the time varying traffic demands.

1.2 Contributions of this work

In this paper, we further extend the above off-line approaches by proposing a robust time-varying energy-save LT design (TV-ES-LTD) methodology that is belonged to node–link-sleeping mode. In our method, we assume that the network hardware elements, such as routers, can auto check the link utilisations connected to them. When the link is idle, that is, the link utilisation is zero, the corresponding linecard will be shut down. For the two bi-direction links, only when both links are idle, the linecards can be put into sleep. When all links connected to the router are idle, this router can auto change it state into sleeping.

Specifically, we demonstrate that (a) critical traffic matrices are determined from a large collection of real traffic matrices; (b) a local search heuristic algorithm can be utilised based on the link weight setting for the energy-saving optimisation according to the critical traffic matrices and (c) large collection of real traffic matrices are routed with the different weight settings. The effectiveness of the proposed approach is assessed by considering a real test case to illustrate that the power saving can indeed be significant, for example, as large and as high as 50% under normal max link utilisation (MLU) request.

The remainder of this paper is organised as follows. The mathematical formulation of the problem and the proposed algorithms are presented in Sections 2 and 3, respectively. Section 4 presents the time complexity analyse. Section 5 details the obtained results. Finally, conclusions are included in Section 6.

2 Problem formulation

The problem of obtaining the network configuration that corresponds to the minimum power consumption under certain traffic conditions and QoS specifications and constraints can be formalised as an ILP problem [20]. We represent an IP network as a directed graph $G=(N, A)$, where N is the set of vertices representing the network nodes, and A is the set of edges representing the interconnection arcs. Let c_{ij} denotes the capacity of the arc (i, j) while $(i, j)\in A$. $T(s, t)$ denotes traffic demand from node s to node t. With each pair (s, t) and each arc (i, j), we associate (i, j) variable $m_{ij}^{(s,t)}$ specifies the portion of the traffic flowing from s to t over (i, j). The variable l_{ij} represents the total load on arc (i, j), that is, the sum of the flows going over (i, j), φ_{ij} and ϕ_i are used to capture the energy cost function of the arc (i, j) and node i, respectively. Therefore the ILP model in this paper can be represented as follows

$$\mathrm{Min}\Psi = \frac{1}{2}\sum_{(i,j)\in A}\varphi_{ij} + \sum_{i\in N}\phi_i \tag{1}$$

subject to:

$$\sum_{x,(x,y)\in A} m_{(x,y)}^{(s,t)} - \sum_{z,(z,y)\in A} m_{(y,z)}^{(s,t)} = \begin{cases} -T(s,t), & \text{if } y=s \\ T(s,t), & \text{if } y=t \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$$l_{ij} = \sum_{(s,t)\in N\times N} m_{ij}^{(s,t)} \quad (i,j)\in A \tag{3}$$

$$\varphi_{ij} = \begin{cases} E_{ij}, & \text{for } 0\le l_{ij}/c_{ij}\le\rho \text{ and } l_{ji}/c_{ji}\le\rho \\ 0, & \text{for } l_{ij}/c_{ij}=0 \text{ and } l_{ji}/c_{ji}=0 \\ \infty, & \text{for otherwise} \end{cases} \tag{4}$$

$$\phi_i = \begin{cases} P_i & \text{for } \sum_{(i,j)\in A} l_{ij} + \sum_{(j,i)\in A} l_{ji} = 0 \\ 0 & \text{for otherwise} \end{cases} \tag{5}$$

$$0\le\rho\le 1 \tag{6}$$

$$m_{ij}^{(s,t)}\ge 0, \quad (i,j)\in A \tag{7}$$

The objective function (1) is the sum of energy cost of links and nodes. As links are full duplex and they are considered entirely powered as soon as one direction conveys traffic and in the above graph formulation the two directions are separately modelled, the link load is the sum of both directions load, the first sum needs to be divided by a factor 2 in order to avoid counting links twice.

IET Netw., 2015, Vol. 4, Iss. 3, pp. 169–174
doi: 10.1049/iet-net.2014.0005

The constrain (2) represents the flow conservation constraints that ensure the desired traffic flow is routed from s to t, the constraint (3) defines the load on each arc. The constraint (4) is the function of the energy cost of link (i, j). E_{ij} is the energy consumption of the link (i, j). The constraint (5) is the function of the energy cost of each node and P_i is the energy consumption of the node i. ρ (6) is the MLU, which is specified by the ISPs. The constraint (7) implies that there are no negative link flows.

As the formulation above indicates, the optimisation goal is to minimise the energy cost while simultaneously avoid congestion and make the links and node idle as much as possible. In this manner, the links will either be idle or busy and the busy link utilisation will be as high as feasible. The link utilisations will be bi-polarity distributed.

3 Algorithm description

3.1 Getting critical traffic matrices

During the day time, the traffic flows centralises in the working area and during the night, the traffic flow centralises in the residential area. The same pattern is also occurs in large areas that have significant time differences from the west to the east of the country. Therefore a day can be divided into several time horizons or periods. In each period, one can measure a large number of traffic flows. Therefore, one obtains a dominant traffic flow [23], which can represent these traffic flows in certain time periods.

In [24], a method named criticalness aware clustering (CritAC) is proposed to obtain a small number of critical traffic matrices from hundreds or thousands of measured traffic matrices. The main idea of CritAC is clustering those matrices, which have the smallest distance with each other into a given amount of clusters and obtaining a cluster head for each cluster. The elements of cluster head matrix are the biggest one in all matrices in this cluster.

We use this method here to derive critical traffic matrices corresponding to different periods of time in a given day or the same period of time in consecutive days. The distance function we used here is the Euclidean distance. The given K is set to be 1 because we only need one matrix for one period of time and for each period we use the method, respectively. Therefore this is a simplified version of CritAC in this paper.

3.2 Heuristic local search algorithm

Traditional network optimisation problems generally deal with the objective or goal of load balancing or maximising the link utilisation capability. Once the optimisation is performed the traffic demands are distributed equally and the link utilisations are closed to an average value. Consequently, the maximum network bandwidth is used, the congestion is avoided and the QoS guarantees are satisfied.

Compared with the load balancing optimisation problem in [25, 26], the energy consumption saving problem in this paper has only one major difference, that is, the cost function. The former balances the link utilisations while the latter diverges these utilisations. Therefore, the same optimal method can be used to obtain different objective.

We start by making slight modifications to the scheme in Fortz and Thorup [25] algorithm. The cost function that is used in this paper is defined by (4) and (5). The allowed maximum link utilisation ρ is a constant that is given initially as the QoS constraint. We are only concerned with the link state that is busy or idle. To avoid conflicting with ISP's MLU constraint, φ_{ij} is set to a very large number in our algorithm when the link utilisation is above ρ.

```
ALGORITHM
W^opt = W_1;   // the initial weight vector and each element in W_1 is set to 1
C^opt = C_1 = ReturnCost(W_1);   // Get the initial Energy Cost
i = 1;
Diversication Count = 0;
while i <= K   // K is a constant iteration time
    W_i = RamdonlyWeight(1, w_max, W_i);
    // w_max is a constant and the max value one element of W_i can be set.
    H_i = Hash(W_i);   // Hash(W_i) calculates the new hash value
    if IsInHash list(H_i) == FALSE
        C_i = ReturnCost(W_i);
        if C_i < C^opt
            W^opt = W_i;
            C^opt = C_i;
            Diversication Count = 0;
        else
            Diversication Count++;
    i++
    if Diversication Count == MAX Diversication Count
        DiversicationFunction();
        Diversication Count = 0;
```

Fig. 1 *Heuristic local search algorithm pseudo-code*

The heuristic local search algorithm pseudo-code in this paper is given in Fig. 1.

This algorithm (see Fig. 1) uses a weight vector $\boldsymbol{W}_1 = \{1, 1,\ldots, 1\}$ to be the initial weight setting of the iteration. The first cost C_1 is calculated by function *ReturnCost*(W_1). Then we search the neighbour W_2 of W_1 using function *RamdonlyWeight*. If C_2 is smaller than C_1, then W_2 is an optimal setting than W_1. As this local search algorithm may be trapped into a local optimal result, which is not global optimal, diversification must be done when Diversication_Count++ reaches the threshold MAX_ Diversication_Count.

3.3 Multiple LT

Following the use of the local search algorithm, period weight configurations for critical matrices are obtained. For example, weight 1 is associated with time period 1, weight 2 is associated with time period 2 and so on. Configuring the network for each of these periods allows one to utilise resources to a higher extent, and consequently power can be saved by switching off certain devices.

One of the concerns with our proposed approach is that the network has to be reconfigured between any two consecutive time periods. This issue may cause two problems. One is that the old traffic flow belonging to the last time period may not have yet been disposed while the new traffic flow has arrived. This may lead and cause packet drop-out losses. However, this problem is not seriously critical since the topology switching is not in a strict sense synchronous. Routers select the corresponding routing table according to the label in the packet head. Therefore old and new traffic flows can be transferred well together. Another concern deals with the fact that old and new traffic flows may have significant differences between them, which may cause and lead to congestion on the links. Assuming that there is a link l in the working state during two time periods, the old traffic flow has still not been disposed completely while the new traffic flow has arrived. Both flows are assumed to be of large size. By mixing the traffic together may cause the link to become congested. This paper does not dispose and fully address this problem. To a certain degree, our scheme just switches configurations for one or two times a day that is not so frequently as [9–11]. One mean of handling this challenging issue is through assigning traffic priorities. We will tackle this issue in our future work.

4 Time complexity

CritAC [24] used in step 1 of TV-ES-LTD, requires $O(kn^2)$ time to generate one critical TM for k traffic matrices in a network with n nodes and m links for each time period. In step 2, Buriol *et al.* [27] used to generate all pairs shortest paths in time $O(mn + mlgn + nT(m^*, n))$, where m^* is the number of different edges contained in shortest paths and $T(m^*, n)$ is a running time to solve a single-source shortest path problem. Thus step 2 requires $O(l(mn + mlgn + nT(m^*, n)))$ since we set the iteration time is l. Therefore the total complexity of TV-ES-LTD is $O(km^2) + O(l(mn + mlgn + nT(m^*, n))) = O(km^2 + l(mn + mlgn + nT(m^*, n))) = O(km^2 + lmn + lnT(m^*, n)) = O(kn^4 + ln^3)$ since $m = O(n^2)$.

5 Performance evaluation

To evaluate the performance in normal conditions, such as nodes can be shut down, and in worst case conditions such as node cannot be shut down, we use a synthetic topology and a real topology with real traffic data measured. A traffic engineering toolbox known as TOTEM [28] is used to complete the local search algorithm and the static simulation. Omnet++ 4.3 [29] is used to complete the logical switching simulation.

In synthetic scenario, we use a topology (shown in Fig. 2) with six nodes eight links. The capacity's unit is Mbps. We use the data shown in Table 1 [14] for the calculation of energy consumption. For the purpose of simplification, we assume that energy consumption is M when the network device is in used and the consumption is zero when the network device is sleeping.

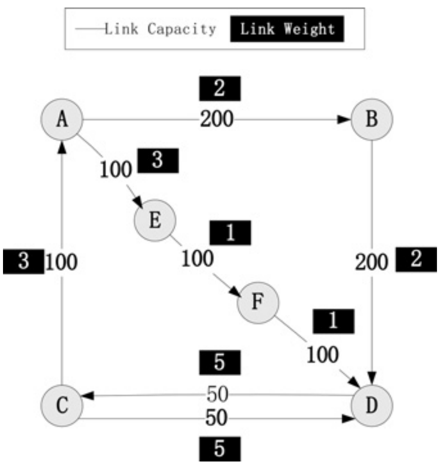

Fig. 2 *Synthetic topology with initialised link weight setting*

Table 1 Energy consumption parameters in watts, for the different network elements

Network element	E_0, W	M, W	Ref.
nodes	$0.85C^{3/2}$	$C^{3/2}$	[30]
(0–100) Mbps links	0.48	0.48	[31, 32]
(100–600) Mbps links	0.90	1	[31, 32]
(600–1000) Mbps links	1.70	2	[33]

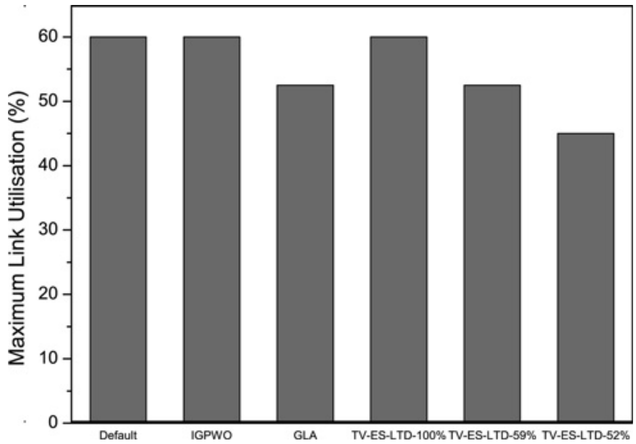

Fig. 3 *Maximum link utilisation between different algorithms*

Given a simple TM composed of only traffic demands between two source–destination pairs: A–D and C–D of 75 and 30 Mbps, respectively, the MLU with default weight setting, the MLU using GLA [18] and the MLU using the method in this paper are presented in Fig. 3. Fig. 4 presents energy efficiency between these different routing settings. The result shows that TV-ES-LTD can achieve almost the same energy consumption even given lower allowed maximum link utilisation.

Using this synthetic network, we can see the difference between the link-sleeping mode and the node–link-sleeping mode of this algorithm. Using the link-sleeping mode, the energy efficiency is not so good as the node–link-sleeping mode. We illustrate this in Fig. 2. In the link-sleeping mode, the nodes will not be considered in the algorithm, so the traffic demand from A to D will be transferred on the path A–E–F–D with the smallest link energy consumption 1.44 W. However, actually when the nodes energy consumption is added, the total value is larger than using the node–link-sleeping mode because there is extra one node on the path A–E–F–D compared with the node B on the path A–B–D. Although the links energy consumption is higher than on the path A–E–F–D, but the node's energy consumption is much higher than the link's.

The synthetic scenario is a simple topology with a sparse TM. How about the performance in a large-scale network

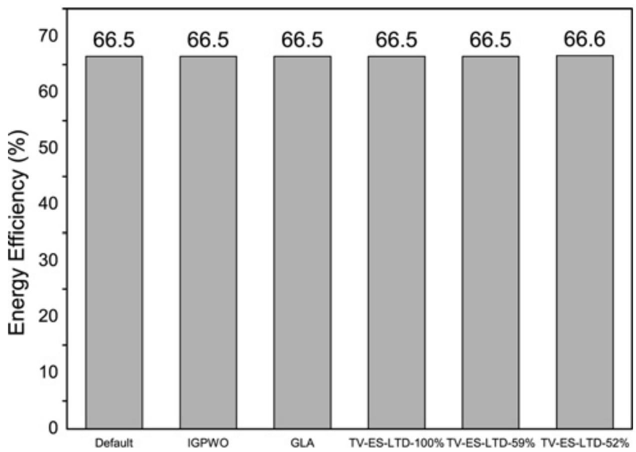

Fig. 4 *Energy efficiency between different algorithm*

IET Netw., 2015, Vol. 4, Iss. 3, pp. 169–174
doi: 10.1049/iet-net.2014.0005

Table 2 Power used because of an active router

Router element	Number	Energy consumption, W	Ref.
CPU	22	38	[35]
mainframe-box	22	64	[35]
linkcard	36 × 2	150	[35]

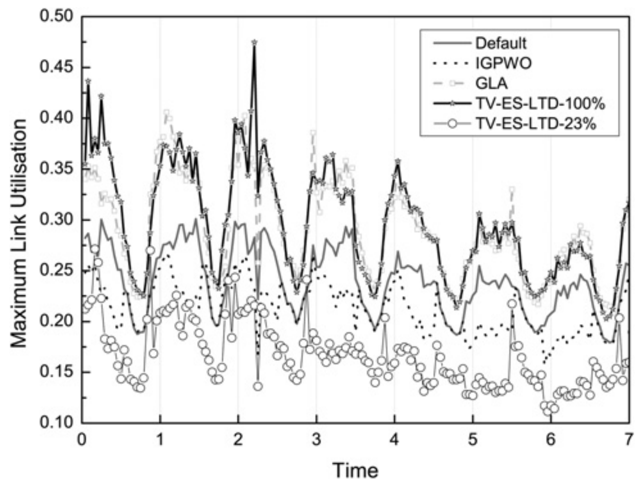

Fig. 5 *MLU variation across 7 days for default, IGPWO, GLA, TV-ES-LTD-100% (ρ is set to 100%) and TV-ES-LTD-23% (ρ is set to 23%)*

with much more traffic demands? We use 673 TMs originating from measurements in the 22-node and the 72-link (the capacity is 40 Gbps) GÉANTnetwork [34] from 12:00 p.m. on 06-06-2005 MDT to 12:00 p.m. on 06-13-2005 MDT (we assume UTC for the original traffic matrices in [34]). The original TMs are measured every 15 min.

For calculating the energy consumption in this experiment, we select the Cisco 12008 router with 40 Gbps link capacity as the node in the network. The energy consumption of this router is shown in Table 2.

We separate 673 TMs into two groups. Group one from 00:00 p.m. to 8:00 p.m. for each day and Group two

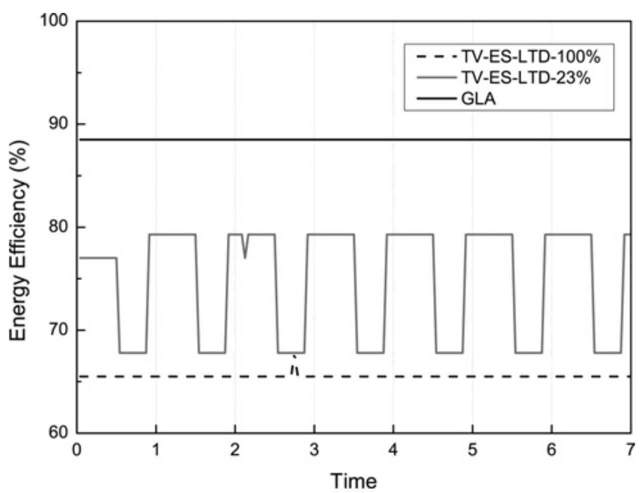

Fig. 6 *Energy efficiency variation across 7 days for default, IGPWO, GLA, TV-ES-LTD-100% (ρ is set to 100%) and TV-ES-LTD-23% (ρ is set to 23%)*

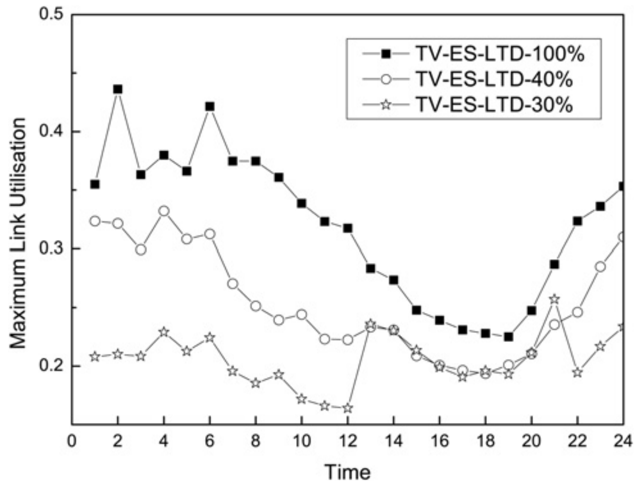

Fig. 7 *Number of working links subjected to different MLU QoS*

from 8:00 p.m. to 00:00 p.m. for each day. From each group, we obtain the corresponding critical matrices. We implement the local search algorithm for the TOTEM. Then, we use the TOTEM and the critical matrices to obtain sets of weight configurations. Corresponding to these weight configurations, using the Omnet++ and the critical traffic matrices the results that are shown in Figs 5 and 6.

Figs. 5 and 6 depict that the actual energy efficiency and MLU performance across the 7 days when GLA and TV-ES-LTD based on the different MLU requests. When Default and IGPWO [25], there is no idle link in the network and the energy consumption is the largest, which are not shown in the figures. When allowed MLU is set to 100%, the energy efficiency is about 65.5% and do not change with time. When allowed MLU is below 100%, such as 40, 30 and 23%, the energy efficiency is fluctuating periodically during peak-off time periodically. Although this is a worst case (all the nodes in this network are in the source–destination pairs of the traffic demands. So all the nodes cannot be switched off), TV-ES-LTD apparently performs better than GLA. As the GLA just only changes link's weight from one direction and it does not consider the duplex links and the nodes state.

The test data of one day are depicted in Fig. 7 specially. For simplicity, we extract 24 TMs from the original TMs on each hour. Fig. 7 demonstrates that since the weight configuration is obtained with respect to the critical matrices and the elements of the critical matrices are larger than the sample matrices, therefore the actual MLU is lower than the requested MLU.

6 Conclusion

In this paper, we have presented and developed a TV-ES-LTD problem for the IP networks, in which the objective and goal is to reduce the total power consumption and to ensure that the requested MLU as network devices are switched off. We have evaluated different MLUs to address and solve the original problem. The results demonstrate and illustrate that large power savings are possible, while limiting the network MLU request. Future work includes will include investigating the means by which one can avoid congestion that is caused by configuration switching and also dynamically save energy saving in IP networks.

7 References

1 Labovitz, C.: ‘What Europeans do at night’, *Secur. Core*, 2009
2 ECONET Project: funded by the European 7th framework programme. http://www.econet-project.eu
3 Kilper, D.: ‘Greentouch consortium: building the roadmap’, *GreenTouch Consortium*, 2012
4 Bolla, R., Bruschi, R., Davoli, F., Cucchietti, F.: ‘Energy efficiency in the future internet: a survey of existing approaches and trends in energy-aware fixed network infrastructures’, *IEEE Commun. Surv. Tutorials*, 2011, **13**, (2), pp. 223–244
5 Bianzino, A.P., Chaudet, C., Rossi, D., Rougier, J.-L.: ‘A survey of green networking research’, *IEEE Commun. Surv. Tutorials*, 2012, **14**, (1), pp. 3–20
6 Gupta, M., Singh, S.: ‘Greening of the internet’. Proc. of the 2003 Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communications, 2003, pp. 19–26
7 Cianfrani, A., Eramo, V., Listanti, M., Marazza, M., Vittorini, E.: ‘An energy saving routing algorithm for a green ospf protocol’. INFOCOM IEEE Conf. on Computer Communications Workshops, 2010, 2010, pp. 1–5
8 Fisher, W., Suchara, M., Rexford, J.: ‘Greening backbone networks: reducing energy consumption by shutting off cables in bundled links’. Proc. of the first ACM SIGCOMM Workshop on Green Networking, 2010, pp. 29–34
9 Bonetto, E., Chiaraviglio, L., Cuda, D., Idzikowski, F., Neri, F.: ‘Exploiting traffic dynamics in power-aware logical topology design’. European Conf. and Exposition on Optical Communications. Optical Society of America, 2011
10 Bianzino, A.P., Chiaraviglio, L., Mellia, M.: ‘Grida: a green distributed algorithm for backbone networks’. IEEE Online Conf. on Green Communications (GreenCom) 2011, 2011, pp. 113–119
11 Bianzino, A.P., Chiaraviglio, L., Mellia, M.: ‘Distributed algorithms for green ip networks’. IEEE Conf. on Computer Communications Workshops (INFOCOM WKSHPS) 2012, 2012, pp. 121–126
12 Chiaraviglio, L., Mellia, M., Neri, F.: ‘Reducing power consumption in backbone networks’. IEEE Int. Conf. on Communications, 2009. ICC’09, 2009, pp. 1–6
13 Ahmad, A., Bianco, A., Bonetto, E., Cuda, D., Gavilanes Castillo, G., Neri, F.: ‘Power-aware logical topology design heuristics in wavelength-routing networks’. 15th Int. Conf. on Optical Network Design and Modeling (ONDM) 2011, 2011, pp. 1–6
14 Bianzino, A.P., Chaudet, C., Larroca, F., Rossi, D., Rougier, J.: ‘Energy-aware routing: a reality check’. IEEE GLOBECOM Workshops (GC Wkshps) 2010, 2010, pp. 1422–1427
15 Zhang, M., Yi, C., Liu, B., Zhang, B.: ‘Greente: power-aware traffic engineering’. 18th IEEE Int. Conf. on Network Protocols (ICNP), 2010, 2010, pp. 21–30
16 Cianfrani, A., Eramo, V., Listanti, M., Polverini, M., Vasilakos, A.V.: ‘An ospf-integrated routing strategy for qos-aware energy saving in ip backbone networks’, *IEEE Trans. Netw. Serv. Manag.*, 2012, **9**, (3), pp. 254–267
17 Francois, F., Wang, N., Moessner, K., Georgoulas, S.: ‘Optimization for time-driven link sleeping reconfigurations in isp backbone networks’. IEEE Network Operations and Management Symp. (NOMS), 2012, 2012, pp. 221–228
18 Francois, F., Wang, N., Moessner, K., Georgoulas, S., Xu, K.: ‘Green igp link weights for energy-efficiency and load-balancing in ip backbone networks’. IFIP Networking Conf., 2013, 2013, pp. 1–9
19 Mumey, B., Tang, J., Hashimoto, S.: ‘Enabling green networking with a power down approach’. ICC, 2012, pp. 2867–2871
20 Chiaraviglio, L., Mellia, M., Neri, F.: ‘Minimizing isp network energy cost: formulation and solutions’, *IEEE/ACM Trans. Netw. (TON)*, 2012, **20**, (2), pp. 463–476
21 Matsuura, H.: ‘Energy-saving routing algorithm using Steiner tree’. IFIP/IEEE Int. Symp. on Integrated Network Management (IM 2013), 2013, 2013, pp. 378–386
22 Lee, S.S.W., Tseng, P.-K., Chen, A.: ‘Multi-topology design and link weight assignment for green ip networks’. IEEE Symp. on Computers and Communications (ISCC), 2011, 2011, pp. 377–382
23 Oriolo, G.: ‘Domination between traffic matrices’, *Math. Oper. Res.*, 2008, **33**, (1), pp. 91–96
24 Zhang, Y., Ge, Z.: ‘Finding critical traffic matrices’. Proc. Int. Conf. on Dependable Systems and Networks, 2005. DSN 2005, 2005, pp. 188–197
25 Fortz, B., Thorup, M.: ‘Internet traffic engineering by optimizing ospf weights’. Proc.Nineteenth Annual Joint Conf. of the IEEE Computer and Communications Societies, INFOCOM 2000, 2000, vol. 2, pp. 519–528
26 Fortz, B., Thorup, M.: ‘Optimizing ospf/is-is weights in a changing world’, *IEEE J. Sel. Areas Commun.*, 2002, **20**, (4), pp. 756–767
27 Buriol, L.S., Resende, M.G.C., Thorup, M.: ‘Speeding up dynamic shortest-path algorithms’, *INFORMS J. Comput.*, 2008, **20**, (2), pp. 191–204
28 Lepropre, J., Balon, S., Leduc, G.: ‘Totem: a toolbox for traffic engineering methods’. 2006
29 Pongor, G.: ‘Omnet: objective modular network testbed’. Proc. of the Int. Workshop on Modeling, Analysis, and Simulation on Computer and Telecommunication Systems, 1993, pp. 323–326
30 Tucker, R., Baliga, J., Ayre, R., Hinton, K., Sorin, W.: ‘Energy consumption in ip networks’. ECOC Symp. on Green ICT, 2008, p. 1
31 Hays, R.: ‘Energy efficient Ethernet using active/idle toggling’. US Patent App. 11/936,327, 7 November 2007
32 Gunaratne, C., Christensen, K., Nordman, B.: ‘Managing energy consumption costs in desktop pcs and lan switches with proxying, split tcp connections, and scaling of link speed’, *Int. J. Netw. Manag.*, 2005, **15**, (5), pp. 297–310
33 Gunaratne, C., Christensen, K., Suen, S.W.: ‘Ngl02-2: ethernet adaptive link rate (alr): analysis of a buffer threshold policy’. Global Telecommunications Conf., 2006. GLOBECOM’06, 2006, pp. 1–6
34 Orlowski, S., Wessäly, R., Pióro, M., Tomaszewski, A.: ‘Sndlib 1.0 survivable network design library’, http://www.sndlib.zib.de/home.action, 2010, **55**, (3), pp. 276–286
35 Cisco: ‘Cisco 12000 series routers – data sheets’. http://www.cisco.com/c/en/us/products/routers/12000-series-routers/datasheet-listing.html

IET Netw., 2015, Vol. 4, Iss. 3, pp. 169–174
doi: 10.1049/iet-net.2014.0005