http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (4): 811-816, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.811.816

TM-EAR: An Energy-aware Routing Algorithm Based on Traffic Matrix

[1]Ming Jiang, [1]Chengxing Zhou, [1]Min Zhang, [2]Chunming Wu and [3]Xiao Min
[1]Institute of Software and Intelligent Technology, Hangzhou Dianzi University, Hangzhou, 310018, China
[2]College of Computer Science and Technology, Zhejiang University, Hangzhou, 310027, China
[3]Mathematics and Information Engineering College of Jiaxing University, Jiaxing, 314001, China

Abstract: The power consumption of Internet is increasing year by year. How to decrease the consumption is of interest for both the research community and the public. Current network has poor efficiency because of the low link utilization and high path redundancy. The low link utilization and high path redundancy lead to high energy consumption. In order to decrease the energy consumption and improve the efficiency, this paper proposes an EAR algorithm called TM-EAR which is combined with the traffic matrix to improve the link utilization and reduce the number of path redundancy. TM-EAR can find a spanning tree which all traffic data are routed on. And we can switch off the links that are not contained in the spanning tree. The experiments show that it is possible to switch off about 50-20% of links. The TM-EAR has a good performance.

Key words: Energy-aware algorithm, green networks, power management, energy saving, Dijkstra algorithm

INTRODUCTION

With the intensification of global warming and the shortage of global energy, people are changing their lifestyle to reduce the demand of energy and decrease the excessive dependence on energy and relieve the greenhouse effect. The researchers in the field of Information and Communication Technology (ICT) are researching more effective ways to reduce the energy cost of ICT. For the increasing of the price of energy and the popularity of the Internet, it is important to find a high energy efficient way for Internet Service Providers (ISPs) and internet infrastructure providers (Zhang *et al*., 2008). In 2010, the internet used between three and five percent of the world's global electricity supply and it will consume more than triple rate of electricity by 2020 following the internet developing (Energy Saving Trust, 2010).

In view of the high energy consumption of network equipment, saving energy is urgent. There are many methods now applied to the field, such as replacing some original modules of the network equipment for energy-aware modules (Bolla *et al*., 2009, 2011) or simplified circuit or applying energy-aware router algorithm (Restrepo *et al*., 2009). Due to the large number of network devices, replacing the hardware is obviously too expensive to make come true. If routers are justly replaced the network routing protocol, the cost is obviously much smaller. Therefore, study on the algorithm is the trend for most researchers. However, some studies didn't consider the network traffic characteristics, which led to unnecessary energy consumption. For example, the EAR algorithm (Cianfrani *et al*., 2010) never considers the traffic matrix and the topology calculated has not related to the traffic matrix. As a result, some links have little flow but some have a high load.

This paper presents a algorithm combining network flow characteristics. According to the distribution of network traffic(e.g., traffic matrix, TM), it can find the minimum network topology to meet TM. Given traffic matrix, it can fix the traffic flow path and converge network traffic in fewer routers so as to close some free links. Experiments show that this algorithm in the case of low load can the 50% links, which is higher than EAR algorithm (Cianfrani *et al*., 2010) nearly 25%.

Existing researches focus on decreasing the energy consumption of network devices under the limit condition of insuring present traffic demand. Gupta and Singh (2003) asserted that it was reasonable to save energy by putting network interfaces and other router to sleep. Further, in order to maximize the amount of energy conservation, they note that some modifications to the Internet architecture may be needed. Cianfrani *et al*. (2010) presented a novel network-level strategy based on a modification of current link-state routing protocols, such as OSPF. By computing the Shortest Path Tree (SPT) and Modifying Path Tree (MPT), a subset of router SPTs are used to select the routing paths to reduce the number of links used to route traffic. Mingui *et al*. (2010) proposed

Corresponding Author: Ming Jiang, Institute of Software and Intelligent Technology, Hangzhou Dianzi University, Hangzhou, 310018, China

GreenTE, a hybrid OSPF/MPLS traffic engineering mechanism that reduces router's power consumption while maintaining network performance at a desired level. GreenTE relies on OSPF to route most of the traffic and uses MPLS tunnels to fine-tune traffic flows so that some links do not carry any traffic during certain periods of time, making it possible to put line-cards into sleep to conserve energy. Chiaraviglio *et al.* (2009) try to close router and switch in the whole network one by one with the goal of finding the best result on energy saving. However, the scheme proposed by Chiaraviglio *et al.* (2009) probably observes inefficient energy saving due to the overlong routing convergence time. Cianfrani *et al.* (2012) proposed an energy saving routing solution, called Energy Saving IP Routing (ESIR), which can switch off the selection of the links and avoid the negative effects of the IP topology reconfiguration procedures.

According to Gupta and Singh (2003) and Bolla *et al.* (2009), network energy saving can be categorized into three levels: circuit, devices and network. In the circuit respect, researches mainly improve the circuit by using techniques (e.g., DVS/DFS). As to devices (e.g., router/switch) in the network, physical links idle time is estimated to close some relevant interfaces. Besides, the inner components of network devices are able to modify their energy consumption according to their own traffic. Though these schemes make every routers and switches isolated from one another and using different scenarios possibly in theory, Gupta and Singh (2003) have not presented any detail solutions and the cost is probably large. For the network level, data packet delivering may need limited part of links instead of the whole in the network, especially at night. Thus, we can design a strategy by adjusting the data flow to switch off those links and routers which do not need. This scheme is easy to realize because present physical network devices (e.g., router, switch) do not have to be replaced. There has been a great deal of research on saving network energy by using this thought.

ALGORITHM DESCRIPTION

Before the description of the algorithm, we describe the physical network as following:

- Physical network is denoted by $G_p(V, E)$, where V is the set of nodes, E is the collection of unidirectional links. Let $N = \|V\|$, $L = \|E\|$ be the cardinalities of V and E, respectively. Let $e_{i,j}$ be the rest bandwidth of the link between router i and j. Let $b_{i,j}$ be the total bandwidth of the link between router i and j. And let B(i) be the links linked with router i. Let $p_{i,j}$ be the forwarding path from router i to j, $I, j \in V$
- The topology is denoted by $G_S(V_S, E_S)$ which switched off the links that calculated by TM-EAR, where V_S is the set of nodes, E_S is the collection of unidirectional links. Let $N_S = \|V_S\|$, $L_S = \|E_S\|$ be the cardinalities of V_S and E_S, respectively
- Traffic matrix $D_T = \{d_{i,j}:(i,j) \in V\}$ is denoted as a $N \times N$ matrix, where $d_{i,j}$ is the traffic flow between router i and j

TM-EAR algorithm: The goal of this paper is converging the traffic flow running on the network to reduce the number of active-links. In order to reduce the number of running links, TM-EAR is based on the theory in Graph Theory which is the minimum number of edges to ensure the graph connectivity. However, in practical, all of the spanning tree in the network may not meet the traffic matrix demand and need two or more trees to support the traffic matrix. For combining multi-tree into one, we define a virtual node linked with the roots of multi-tree which has infinity bandwidth.

Then two kinds of routers are defined as (G,D)-KeyRouter and virtual router. Through two sets of routers, energy saving problem is transformed to find the (G,D)-KeyRouter which is the spanning tree's root.

Definition 1: (G,D)-KeyRouter: According to given traffic matrix D and physical network G, the (G,D)-KeyRouter is the one router through which the maximum total of flows passed in network G.

Definition 2: Virtual router(VR): The virtual router is linked with all the (G,D)-KeyRouter, which has infinite bandwidth.

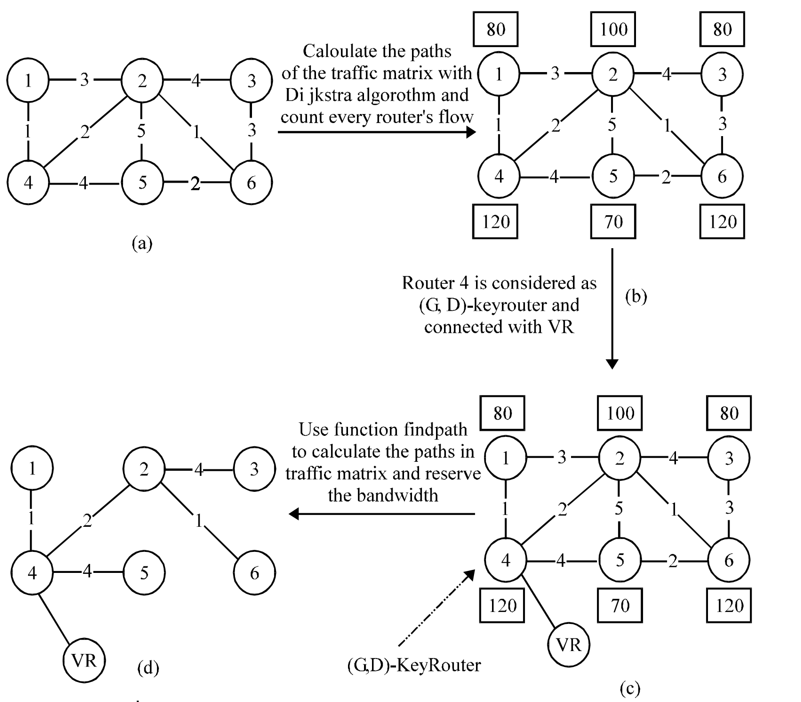
The TM-EAR algorithm (Table 1) is designed as following: Step 1, calculated (G,D)-KeyRouter when the

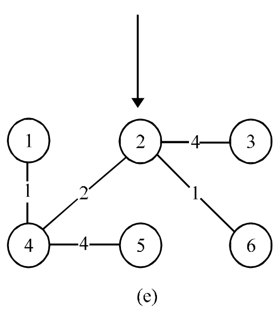
Table 1: The detail of TM-EAR

TM-EAR (D_p, D_T)
Key←getKeyRouter(D_p, D_T)
for $\forall d_{i,j} \in D_T$
$d_{m,n}$←maximum element in D_T, $m,n \in V$
if $c_{Key,i} > d_{m,n}$ and $c_{j,Key} > d_{m,n}$ $\forall I, j \in B(Key)$ then
$P_{m,n}$←findpath G_p, $d_{m,n}$,Key) where $p_{m,n} \subseteq V$
end if
if $p_{m,n}$ exist then
$C_{i,j}$←$c_{i,j}$-$d_{m,n}$ where $\forall i,j \in p_{m,n}$//deduct the reserved bandwidth of the links that $p_{m,n}$ pass through on G_p, so G_p has been changed to G'_p
$d_{m,n}$←-∞//change the value of $d_{m,n}$ to -∞, so D_T turn into D'_T
else
Key←getKeyRouter(D'_p, D'_T)
end if
end for
for each $e_{i,j}$ in E, $i,j \in V$/if the utilization of the links between router i and j is zero, then remove the links on G_p
if $c_{i,j} == b_{i,j}$ then
G_s←G_p-$e_{i,j}$
end if
end for

physical network is G_p and the traffic matrix is D_T. In case of exhausting the bandwidth of some links which link with (G,D)-KeyRouter, TM-EAR rules that if every link that links with (G,D)-KeyRouter can meet $d_{m,n}$, the largest element of the current traffic matrix(In order to achieve optimal result, TM-EAR applies the ideas of greedy algorithm), then search the path from router m to router n, called $p_{m,n}$ by using function findpath (Step 4); If $p_{m,n}$ exists, then the TM-EAR reserves the bandwidth for $p_{m,n}$ (the physical network G_p is changed as G'_p, Step 8) and change $d_{m,n}$ as -∞ to avoid being calculated again (so far, the D_T has been changed as D'_T, Step 9); if TM-EAR cannot find the $p_{m,n}$ (Step 10), we need to find the other (G,D)-KeyRouter on the changed physical network G'_p and traffic matrix D'_p to bear the traffic matrix with previous (G,D)-KeyRouter (Step 11).After Step 2-13, switch off the links that never used so we can get the topology G_s which is the final topology calculated (Step 14-18).

Example of TM-EAR: Figure 1 depicts a simple operation process of the TM-EAR algorithm. This example is a simple description of the TM-EAR idea. We assume that the physical network link bandwidth is 1 Gbps. Firstly, we calculate the (G,D)-KeyRouter by using function get KeyRouter(TM-EAR algorithm Step 1), e.g., Fig. 1b-c. Before calculating the (G,D)-KeyRouter, we must achieve the shortest paths of traffic matrix and count the every router's flow which is shown in the rectangular (Step 1-9 of function get KeyRouter). Then we can know the maximum load routers are router 4 and 6. Here router 4 is considered as (G,D)-KeyRouter and linked with VR, e.g., Fig. 1d. After obtaining the (G,D)-KeyRouter, the rest is to calculate the paths of traffic matrix by using the function findpath (Step 5 of TM-EAR algorithm) and reserve the bandwidth for those paths. When calculating the paths, the big element in traffic matrix has a higher priority to be calculated. If the element in traffic matrix has been





Router	1	2	3	4	5	6
1	0	10 M	0	10 M	0	40 M
2	20 M	0	10 M	0	0	0
3	0	0	0	20 M	30 M	0
4	0	20 M	0	0	0	0
5	0	0	20 M	0	0	0
6	0	0	0	20 M	0	0

The traffic matrix

Fig. 1: The example of TM-EAR, a-e show the flow of the system

calculated, the element is changed to $-\infty$ (Step 9 of TM-EAR algorithm). If one of links linked with (G,D)-KeyRouter cannot meet the demand of the maximum element in current traffic matrix, another (G',D')-KeyRouter is needed by repeating Fig. 1b-c. Until all elements are calculated, the topology G_S can be gotten by switching off the links that never be reserved bandwidth (Step 14-18 of TM-EAR algorithm).

Get KeyRouter: The function get key router (Table 2) aims to find the (G,D)-KeyRouter. In this function, we need two parameters G and D, where G denotes physical network and D denotes traffic matrix. Let f_m be the flow that pass through the router m. We assume that the routers in physical network forward the traffic based Dijkstra algorithm, so we can calculate the paths of all the element whose value is not equal $-\infty$ in traffic matrix (Step 1-4). For the maximum flow size router will be the (G,D)-KeyRouter, we need to statistic router m's flow (Step 5-7). Then the maximum flow router become the (G,D)-KeyRouter (step 10).

Findpath: The function findpath (Table 3) aims to find the path met $d_{i,j}$ between router i and j. This function takes five parameters as its input: G, i, j, $d_{i,j}$ and Key, where G denotes physical network; i and j are routers in physical network in traffic matrix; $d_{i,j}$ denotes the traffic demand between router i and j; the Key denote the (G,D)-KeyRouter. In order to combine multiple trees into a bigger tree, we need to link (G,D)-KeyRouter when it was calculated with VR and set the same weight to create a new topology denoted by G_{new} (Step 1). When calculating the path, we rule that all the flows must pass through one of the (G,D)-KeyRouter. Thus, we calculate the paths from router i to VR and VR to router j (Step 2, 3). However, if we follow the rule, some flows may pass through two unidirectional links connecting the same routers, which leads to wasting a lot of bandwidth.

According to this situation, we have optimized as following: if $p_{i,VR}$ and $p_{VR,j}$ contain same routers except VR e.g., $p_{i,VR} \cap p_{VR,j} - \{VR\} != \phi$, then we find the nearest router from router i and j in the set of $p_{i,VR} \cap p_{VR,j}$ based on shortest path algorithm denoted by m. So we can get the $p_{i,j}$ made up of $p_{i,m}$ and $p_{m,j}$ where $p_{i,m}$ and $p_{m,j}$ are contained in $p_{i,VR}$ and $p_{VR,j}$, respectively (Step 4, 5). If $p_{i,VR}$ and $p_{VR,j}$ have no same routers except VR, each path must have own (G,D)-KeyRouter, we calculate the shortest path from router Key_i to Key_j based on bandwidth-constrained shortest path algorithm, where Key_i and Key_j are belonged to $p_{i,\ VR}$ and $p_{VR,j}$, respectively. Finally, $p_{i,j}$ is made up of $p_{i,VR}$ and $p_{VR,j}$ (e.g., $p_{i,VR} \cup p_{VR,j} \cup pKey_i\ key_j - \{VR\}$, Step 7).

Table 2: The detail of function getKeyRouter

getKeyRouter(G, D)	
1.	$\forall M \in V\ f_m \leftarrow 0$
2.	for $\forall d_{i,j} \in D\ i,j \in V$
3.	if $d_{i,j} \neq -\infty$
4.	then the shortest path between router i and j denoted by $p_{i,j}$ is calculated by using Dijkstra algorithm
5.	for $m \in p_{i,j}$
6.	$f_m \leftarrow d_{i,j} + f_m$//statistic the traffic flow pass through router m
7.	end for
8.	end if
9.	end for
10.	$Key \leftarrow m, m = \max\{f_i : i \in V\}$

Table 3: The detail of function findpath

findpath (G,i,j,η_e,Key)	
1.	In physical network G, let Key link with VR to generate a new physical network denoted by G_{new}
2.	We denoted $p_{i,VR}$ and $p_{VR,j}$ as the shortest path from router i to VR and VR to router j calculated by bandwidth-constrained shortest path algorithm that the bandwidth is $d_{i,j}$
3.	if $p_{i,VR} \cap p_{VR,j} - \{VR\} != \phi$
4.	then $p_{i,j} \leftarrow p_{i,m} \cup p_{m,j}$ where $p_{i,m} \subset p_{i,VR}$, $p_{m,j} \subset p_{VR,j}$, $m \in p_{i,VR} \cap p_{VR,j}$ nearest router distance from router i and j based on bandwidth-constrained shortest path
5.	else
6.	we denote $p_{Keyi,\ Keyj}$ as the path from router Key_i to Key_j based on bandwidth-constrained shortest path algorithm, where Key_i and Key_j are belonged to $p_{i,VR}$ and $p_{VR,j}$, respectively. Then we can get the forward path from router i to j denoted by $p_{i,j} \leftarrow p_{i,VR} \cup p_{VR,j} \cup p_{Keyi,\ Keyj} - \{VR\}$
7.	end if

PERFORMANCE EVALUATION

In this section, we first describe the performance evaluation environment and then present our main evaluation results. Our evaluation focuses primarily on the percentage of links that are allowed to power off and the utilization of links' bandwidth. We also compare the TM-EAR algorithm with the EAR algorithm (Cianfrani *et al*., 2010). The evaluation results show that our TM-EAR algorithm can switch off more links than EAR algorithm.

Evaluation environment and performance metric:

Substrate network: The experiment is based on two different network topologies: one is Geant Network (http://www.geant.net/), whose topology is reported in Fig. 2. It has 23 nodes and 36 links., while black squared indicate the weight of links, whose number is reported as label. The other is AS7018 Network provided by the OspfOpt (http://research.microsoft.com/en-us/um/people/srikanth/software_ospfopt.html), which includes 35 nodes and 68 links. Its link's bandwidth obey a uniform distribution between 1 and 10.

Traffic matrix: The traffic matrix was generated by the OspfOpt (http://research.microsoft.com/en-us/um/people/srikanth/software_ospfopt.html). We define the traffic matrix as $D_{trafficmatrix} = \{d_{i,j}, \beta, \forall I, j \in V\}$, where, $\beta \in [1\%, 100\%]$

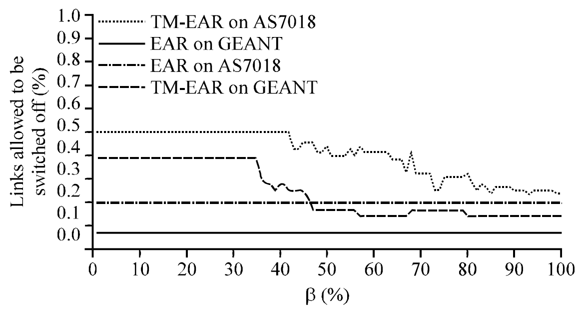


Fig. 2: The percentage of the links allowed to be switched off

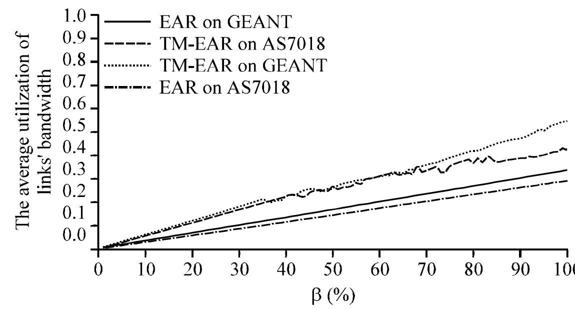


Fig. 3: The utilization of links' bandwidth

in order to evaluate the change trend of the link saving rate in different network loading conditions.

Performance metric: We use the percentage of links that allows to power off denoted by η_e and the utilization of links' bandwidth denoted by η_l as our performance metric, where:

$$\eta_e = \frac{|L - L_s|}{L}$$

and

$$\eta_l = \frac{\sum_{\forall i,j \in V} (b_{i,j} - c_{i,j})}{L}$$

Evaluation results: Figure 2 depicts the comparison of the percentage of the links allowed to be switched off between TM-EAR and EAR (Cianfrani *et al*., 2010) in different substrate network topologies. As can be seen from Fig. 2 TM-EAR has a better performance than EAR for each network topology. The percentage of the links allowed to be switched off is about 50% in the case of low load of traffic matrix. This is because the physical network only needs one spanning tree to satisfy the traffic matrix (which only requires one (G,D)-KeyRouter) in the case of low load of traffic matrix. When the load of traffic matrix is increasing, the saving link rate is declining. The reason is that a spanning tree can no longer meet the traffic matrix. So TM-EAR needs to calculate more (G,D)-KeyRouters in order to let more links to support the traffic matrix.

Figure 3 depicted the utilization of links' bandwidth. For the higher percentage of links that are allowed to power off, we can get higher utilization than EAR (Cianfrani *et al*., 2010). At the high load of traffic matrix, the utilization of TM-EAR increases faster than EAR (Cianfrani *et al*., 2010). It is because that most flows pass through much more links than EAR (Cianfrani *et al*., 2010).

So when the load of traffic matrix is 100%, the utilization of TM-EAR is still higher than EAR (Cianfrani *et al*., 2010) about 10-20%.

CONCLUSION

In this study, we proposed an TM-EAR algorithm to improve the link utilization and reduce the number of path redundancy. We find the (G,D)-KeyRouters and converge network flows in these routers so as to close some free links. The way to find the (G,D)-KeyRouter is important. We just consider the router which has maximum flow as the (G,D)-KeyRouter. Therefore, more in-depth and innovative studies which find a more conformable way are expected very much.

ACKNOWLEDGMENTS

This study is supported by the National Basic Research Program of China (973 Program) (2012CB315900(2012CB315903)), the National Natural Science Foundation of China (No. 61070157, No. 61070213), the Zhejiang Provincial Natural Science Foundation of China (No. Y1111192, LY12A01019) and the Zhejiang Provincial Technical Plan Project (No. 2011C13008).

REFERENCES

Bolla, R., R. Bruschi, F. Davoli and A. Ranieri, 2009. Performance constrained power consumption optimization in distributed network equipment. Proceedings of the IEEE International Conference on Communications Workshops, June 14-18, 2009, Dresden, pp: 1-6.

Bolla, R., R. Bruschi, F. Davoli and F. Cucchietti, 2011. Energy efficiency in the future internet: A survey of existing approaches and trends in energy-aware fixed network infrastructures. IEEE Commun. Surv. Tutorials., 13: 223-244.

Chiaraviglio, L., M. Mellia and F. Neri, 2009. Energy-aware backbone networks: A case study. Proceedings of the IEEE International Conference on Communications Workshops, June 14-18, 2009, Dresden, pp: 1-5.

Cianfrani, A., V. Eramo, M. Listanti, M. Marazza and E. Vittorini, 2010. An energy saving routing algorithm for a green OSPF protocol. Proceedings of the INFOCOM IEEE Conference on Computer Communications Workshops, March 15-19, 2010, San Diego, CA., pp: 1-5.

Cianfrani, A., V. Eramo, M. Listanti, M. Polverini and A.V. Vasilakos, 2012. An OSPF-integrated routing strategy for QoS-aware energy saving in IP backbone networks. IEEE Trans. Network Service Manag., 9: 254-267.

Energy Saving Trust., 2010. Internet's energy consumption growing, report finds. http://www.energysavingtrust.org.uk/About-us/Industry-news/Renewable-energy/Internet-s-energy-consumption-growing-report-finds

Gupta, M. and S. Singh, 2003. Greening of the Internet. Comput. Commun. Rev., 33: 19-26.

Mingui, Z., Y. Cheng, L. Bin and Z. Beichuan, 2010. Green TE: Power-aware traffic engineering. Proceedings of the 18th IEEE International Conference on Network Protocols, October 5-8, 2010, Kyoto, pp: 21-30.

Restrepo, J.C.C., C.G. Gruber and C.M. Machuca, 2009. Energy profile aware routing. Proceedings of the IEEE International Conference on Communications Workshops, June 14-18, 2009, Dresden, pp: 1-5.

Zhang, G.Q., Q.F. Yang, S.Q. Cheng and T. Zhou, 2008. Evolution of the internet and its cores. New J. Phys., Vol. 10.