Minimization of Network Power Consumption with Redundancy Elimination

Frédéric Giroire[1], Joanna Moulierac[1], Truong Khoa Phan[1]
and Frédéric Roudaut[2]

[1] Joint project MASCOTTE, I3S(CNRS-UNS), INRIA, Sophia-Antipolis, France
{frederic.giroire; joanna.moulierac; truong_khoa.phan}@inria.fr
[2] Orange Labs, Sophia Antipolis, France
{frederic.roudaut}@orange.com

Abstract. Recently, *energy-aware routing* has gained increasing popularity in the networking research community. The idea is that traffic demands are aggregated over a subset of the network links, allowing other links to be turned off to save energy. In this paper, we propose GreenRE - a new energy-aware routing model with the support of the new technique of *data redundancy elimination (RE)*. This technique, enabled within the routers, can identify and remove repeated content from network transfers. Hence, capacity of network links are virtually increased and more traffic demands can be aggregated. Based on our real experiments on Orange Labs platform, we show that performing RE consumes some energy. Thus, while preserving connectivity and QoS, it is important to identify at which routers to enable RE and which links to turn off so that the power consumption of the network is minimized. We model the problem as an Integer Linear Program and propose a greedy heuristic algorithm. Simulations on several network topologies show that GreenRE can gain further 30% of energy savings in comparison with the *traditional energy-aware routing* model.

Keywords: Green networking; Energy-efficient routing; Algorithm.

1 Introduction

Some recent studies [6][12] exhibit that the traffic load of routers only has small influence on their energy consumption. Instead, the dominating factor is the number of active elements on routers such as ports, line cards, base chassis, etc. Therefore, in order to minimize energy consumption, fewer network elements should be used while preserving connectivity and QoS. Intuitively, it is possible to have multiple paths between a pair of source-destination on the network. When traffic load on links is low, we can aggregate the traffic into fewer links so that the other links do not need to carry any traffic. Routers then can turn off idle links (or precisely, two ports on the two routers at the ends of the link) for energy reduction. Although today's routers cannot turn off the ports and bring them back to active state quickly, we believe that this advance will come in near future, especially if it offers big energy savings.

In general, link capacity is the main constraint when aggregating traffic flows to a subset of links. In this work, we use an assumption that routers can eliminate

redundant data traffic and hence virtually increase capacity of the links. As a result, more traffic flows can be aggregated and more links can be turned off to save energy. Although routers nowadays cannot remove repeated content from network transfers, we notice that there is WAN Optimization Controller (WOC) - a commercial device used in enterprises or small ISPs to eliminate traffic redundancy [5][10][14]. In order to identify the power consumption directly induced by RE, we perform real experiments on the WOCs. Because the principle idea of routers performing RE is similar to that of the WOC (section 2.2), we believe that when a router eliminates traffic redundancy, it also consumes more energy than usual. Therefore, in order to evaluate the global power consumption of the network, we should consider both the number of active links and the number of routers that perform RE. In summary, the contributions of this work are the following:

- We do real experiments to show power consumption of a WOC.
- We define and model the problem using Integer Linear Programing (ILP).
- We propose and evaluate a greedy heuristic algorithm that can be used for large-scale networks.
- By simulation, we present energy savings on real network topologies. In addition, we discuss the impact of GreenRE on route length (or delay also).

The rest of this paper is structured as follows. We summarize related works in Section 2. In Section 3, we model GreenRE using ILP, then propose a greedy heuristic algorithm. Simulation results are presented in Section 4. We have discussion in Section 5 and finally, we conclude the work in Section 6.

2 Related Works

2.1 Reduction of Energy Consumption

Measurement of energy consumptions. Several empirical measurements showed that energy consumption of network devices is largely independent of their load [6][12]. In addition, many experiments show that the power consumption depends on the number of active ports on routers [12]. Hence, explicitly turn off unused ports can reduce power consumption of routers.

Energy minimization. The work on energy consumption of the Internet has been first evoked as a hypothetical working direction in [11]. Recently (2008 - present), researchers have started to massively invest their efforts in this research area [4]. The authors in [7] modeled the problem using ILP and showed how much energy can be saved on different network topologies. In [9], the authors proved that there is no polynomial-time constant factor approximation algorithm for this problem. Then they give theoretical bounds for specific network topologies and present heuristics to find solution close to the optimal one.

2.2 Reduction of Traffic Load

Internet traffic exhibits large amount of redundancy when different users access the same or similar contents. Therefore, several works [1][2][3][10][16][17] have

explored how to eliminate traffic redundancy on the network. Spring et al. [17] developed the first system to remove redundant bytes from any traffic flows. Following this approach, several commercial vendors have introduced WAN Optimization Controller (WOC) - a device that can remove duplicate content from network transfers [5][10][14]. WOCs are installed at individual sites of small ISPs or enterprises to offer end-to-end RE between pairs of sites. As shown in Fig. 1,
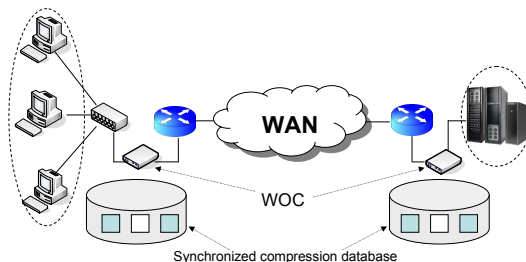

Fig. 1. Reduction of end-to-end link load using WOC

the patterns of previously sent data are stored in database of the WOCs at both sending and receiving sides. The technique used to synchronize the databases at peering WOCs can be found in [10]. Whenever the WOC at the sending side notices the same data pattern coming from the sending hosts, it sends a small signature instead of the original data (called *encoding process*). The receiving WOC then recovers the original data by looking up the signature in its database (called *decoding process*). Because signatures are only a few bytes in size, sending signatures instead of actual data gives significant bandwidth savings.

Recently, the success of WOC deployment has motivated researchers to explore the benefits of deploying RE in routers across the entire Internet [1][2][3][16]. The core techniques used here are similar to those used by the WOC: each router on the network has a local cache to store previously sent data which then is used to encode and decode data packets. Obviously, there are two key challenges that hinder the deployment of RE on routers. First, a significant number of memory accesses and heavy computation are required during various stages of RE. Second, a large amount of memory is required for the local cache at routers. However, Anand et al. [3] have introduced SmartRE which considers these challenges in the design. The authors show that on the desktop 2.4 GHz CPU with 1 GB RAM used for storing caches, the prototype can work at 2.2 Gbps for encoding and at 10 Gbps for decoding packets. Moreover, they believe that higher throughput can be attained if the prototype is implemented in hardware. Hence, the key limitations can be overcome and this technology should come in future's routers, especially if it offers significant bandwidth savings. Several real traffic traces have been collected from many networks such as at 11 corporate enterprises in US [2], at a large university in US [1] and at 5 sites of a large corporate network in North America [16]. The authors in [1][2][16] conclude that up to 50% of the traffic load can be reduced with RE support.

In this work, we propose GreenRE - the first model of *energy-aware routing* with RE support. We show that RE, which was initially designed for bandwidth savings, is also a potential technique to reduce power consumption of the Internet.

3 Energy-aware Routing with RE

Traditional energy-aware routing model (without RE) has been presented as a multicommodity integral flow problem with the objective of minimizing the number of links [9]. Our GreenRE model can be modeled in a similar way. However, when performing RE, routers consume some additional energy (section 4.1). Thus both the active links and the RE-routers (*routers that perform RE*) should be considered in the objective function. We model a network topology as an undirected weighted graph $G = (V, E)$, where the weight C_{uv} represents the capacity of an edge $(u, v) \in E$. We represent a set of demands by $D = \{D_{st} > 0; (s, t) \in V \times V, s \neq t\}$ where D_{st} denotes the amount of bandwidth required for a demand from s to t. A feasible routing of the demands is an assignment for each $D_{st} \in D$ a path in G such that the total amount of demands through an edge $(u, v) \in E$ does not exceed the capacity C_{uv}. We present in Fig. 2 a simple
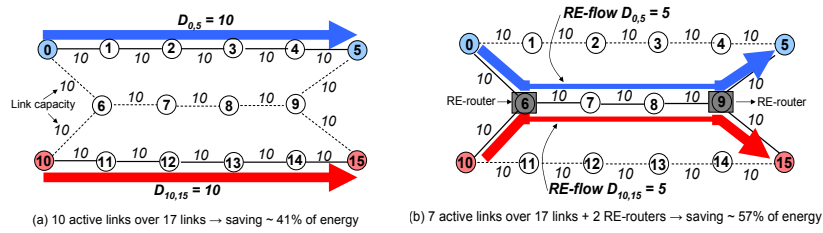

Fig. 2. Feasible routing solution (a) without RE-router and (b) with RE-routers

example to show the efficiency of using RE-routers in energy-aware routing. The objective is to find a routing solution for the two demands $D_{0,5}$ and $D_{10,15}$ (both with traffic volume requirement of 10) so that it satisfies the link capacity constraint and minimize power consumption of the network. As shown in Fig. 2a, the routing solution without RE-router requires 10 active links, then we can turn off 7 links and save 41% of energy consumption of the network. Meanwhile, when routers 6 and 9 are RE-routers, and assume that 50% of the traffic is redundant, then we have a better solution in which 57% of energy is saved (Fig. 2b). Moreover, if the two demands increase traffic volume to 20, there is no feasible routing solution without RE-router. But it is possible to find a solution if we have RE-routers at 0, 5, 10 and 15.

Energy-aware routing problems are known to be NP-hard [7][9], hence we first define and model the problem using Integer Linear Program (ILP), then we present a greedy heuristic algorithm for large scale networks.

3.1 Optimal solution - Integer Linear Program (ILP)

For each link $(u, v) \in E$, we use a binary variable x_{uv} to determine if the link is used or not. We consider a simplified architecture where each line card on router has one network interface. Therefore, when a link (u, v) is used ($x_{uv} = 1$), the two network interfaces at router u and router v are enabled, and the power consumption of this link (called PL_{uv}) is the sum of power consumption of the two interfaces at router u and router v. We note f_{uv}^{st} be the flow on edge (u, v) corresponding to the demand D_{st} flowing from u to v. We use an assumption that all routers on the network can perform RE, however they can disable RE when unused to save energy. Then, we define a binary variable w_u which is equal to 1 if router u perform RE (called RE-router and it consumes additional PN_u Watts). We also use a redundancy factor γ to represent amount of data that can be eliminated by the RE-routers. We differentiate the usual flow f_{uv}^{st} from the RE-flow fr_{uv}^{st} (the redundant data has been removed), on link (u, v) for a demand (s, t) and $fr_{uv}^{st} = f_{uv}^{st}/\gamma$. When a flow enters a RE-router u, it can happen:

- If the flow is RE-flow, the router u can decode it to usual flow or do nothing (just forward the RE-flow).
- If the flow is usual flow, the router u can perform RE for this flow or just forward it.

The *Objective Function* of the ILP is to find a routing solution that minimizes the total power consumption of the network:

$$\min \left[PL_{uv} \sum_{(uv) \in E} x_{uv} + PN_u \sum_{u \in V} w_u \right]$$

subject to:

"*Flow conservation constraint*": $\forall u, v \in V$, $N(u)$ - neighbors of u

$$\sum_{v \in N(u)} f_{vu}^{st} - \sum_{v \in N(u)} f_{uv}^{st} + \gamma \left(\sum_{v \in N(u)} fr_{vu}^{st} - \sum_{v \in N(u)} fr_{uv}^{st} \right) = \begin{cases} -D_{st} & \text{if } u = s, \\ D_{st} & \text{if } u = t, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

"*Capacity constraint*": $\forall (u, v) \in E$, C_{uv} - capacity of link (u, v)

$$\sum_{(s,t) \in D} \left(f_{uv}^{st} + f_{vu}^{st} + fr_{uv}^{st} + fr_{vu}^{st} \right) \leq x_{uv} C_{uv} \quad (2)$$

"*RE-router*": $\forall u, v \in V$,

$$M.w_u \geq \sum_{v \in N(u)} \left(fr_{uv}^{st} - fr_{vu}^{st} \right) \quad (3a)$$

$$M.w_u \geq \sum_{v \in N(u)} \left(fr_{vu}^{st} - fr_{uv}^{st} \right) \quad (3b)$$

Equation (1) states the flow conservation constraint and we differentiate between RE-flows and usual flows. Constraint (2) forces the link load to be smaller than the link capacity. Like some existing works [7][9], we consider a simplified model where link load is the total volume of all the flows passing through this link. In addition, to accommodate traffic bursts, we should limit the maximum utilization over any links in the network. For instance, the capacity using in equation (2) would be set to αC_{uv} where $\alpha = 50\%$. The constraints (3a) and (3b) (where M is a big constant number) are to make sure when the router u disables RE ($w_u = 0$), the difference between the sum of the RE-flows that enter and leave router u is equal to zero.

3.2 Heuristic solution

The heuristic algorithm, in the first step, tries to find a routing solution for all demands so that it minimizes the number of active links. Because all routers on the network are assumed to be RE-routers, we virtually decrease volume of all the traffic demands to $Dr_{st} = D_{st}/\gamma$. Then for the second step, based on the routing found in the previous step, we try to disable RE on as many routers as possible to save energy.

Algorithm 1: Finding a feasible routing
Input: An undirected weighted graph $G = (V, E)$ where each edge e has a capacity C_e, a residual capacity R_e, an initial metric w_e and a set of demands $Dr_{st} \in D$.

$\forall e \in E, R_e = C_e, w_e =$ total number of demands
Sort the demands in random order
while *Dr_{st} has no assigned route* **do**
    compute the shortest path SP_{st} with the metric (w_e)
    assign the routing SP_{st} to the demand Dr_{st}
    $\forall e \in SP_{st}, R_e = R_e - Dr_{st}, w_e = w_e - 1$
end
return the routing (if it exists) assigned to the demands in D

Starting with the *Algorithm 1*, we compute a feasible routing for the RE-demands. Initially, all links on the network are set up with the same metric w_e which is equal to the total number of demands. We compute the shortest path for each demand with the metric w_e on links. Then, the links that have carried the shortest path is updated with metric: $w_e = w_e - 1$. Using this metric in the shortest path, we implicitly set high priority to reuse links that have already been selected. Then, the *Algorithm 2 - Step 1* is used to remove in priority links that are less loaded. C_e/R_e is used as the load on a link where R_e is the residual capacity on link e when previous demands have been routed.

Algorithm 2: *Input:* An undirected weighted graph $G = (V, E)$ where each edge $e \in E$ has a capacity C_e and a residual capacity R_e

---

> *Step 1* - Removing less loaded links:
> **while** *edges can be removed* **do**
> > remove the edge e that has not been chosen and has smallest value C_e/R_e
> > compute a feasible routing with the Algorithm 1
> > if no feasible routing exists, put e back to G
>
> **end**
> return the feasible routing if it exists.
> *Step 2* - Enabling set of RE-routers:
> Consider normal demands (without RE) and routing solution in Step 1
> **while** *network is congested* **do**
> > find demands to perform RE first (details later in part (a))
> > enable set of RE-routers using end points congestion (details later in part (b))
>
> **end**

In *Step 2*, we use the routing solution found in the *Step 1* as the input of the algorithm. Then, we consider all the traffic demands as normal demands without RE. Hence, some links can be congested because the total traffic volume of demands may exceed the link capacity. The heuristic we use in *Step 2* is based on following observations:

(a) Which demand to perform RE first? In Fig. 3a, we can see that

(a) 9 active links + 3 RE-routers

(b) 6 active links + 4 RE-routers

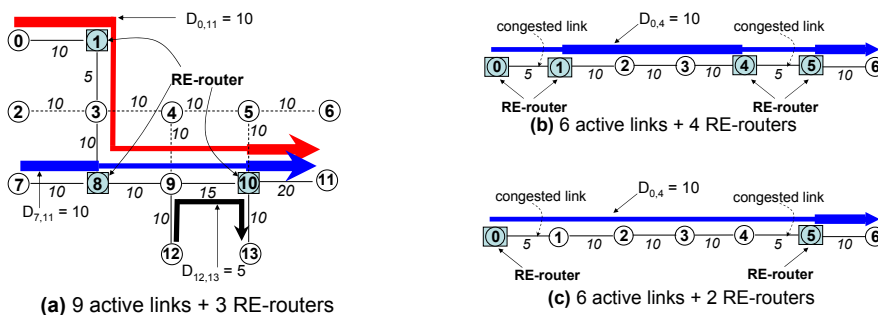**(c)** 6 active links + 2 RE-routers

Fig. 3. Congested links and RE-routers

when performing RE for $D_{0,11}$ on router 1 and 10, the amount of traffic passing all the congested links (links (1, 3), (8, 9) and (9, 10)) is also reduced. Then, the heuristic in step 2 decides to perform RE for this flow first. Assume that the redundancy factor $\gamma = 2$, then the links (8, 9) and (9, 10) are still congested. After removing $Dr_{0,11}$, the available capacity of links (8, 9) and (9, 10) are 5 and 10, respectively. Applying the same rule, the next demand to perform RE is $D_{7,11}$. Then, there is no congested link on the network since the link (9, 10) is

also released from congestion. Finally, only the routers 1, 8 and 10 are needed to enable RE. In summary, the algorithm will perform RE for the flows that passes through most of the congested links first.

(b) End Points Congestion: in Fig. 3b and Fig. 3c, the demand (0,6) has traffic volume of 10 and the number on links indicates link capacity. Therefore, the two links (0, 1) and (4, 5) are congested. Hence, a naive solution is to enable RE at the two end-point routers of each congested link as shown in Fig. 3b. However, a better solution with less RE-routers should be to enable RE only at the starting (router 0) and ending point (router 5) of all the congested links (Fig. 3c). In summary, the algorithm will look for the longest congested part of the flow to enable RE-routers.

4 Experiments and Simulation Results

4.1 Energy Consumption with WOC

Several results of bandwidth savings using WOC can be found in [10]. We have also performed experiments on the network platform of the project Network Boost at Orange Labs [1]. We installed two WOCs, each at the access links of the two sites (let's call site A and site B). These two sites are connected via a backbone composed of 4 routers. We setup FTP connections for uploading files from site A to site B. As shown in Fig. 4a, power consumption of the WOC
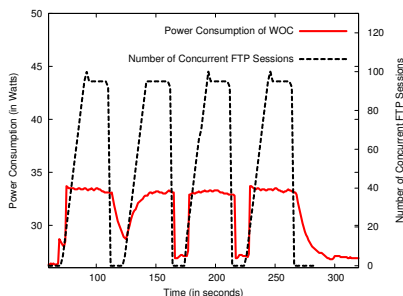

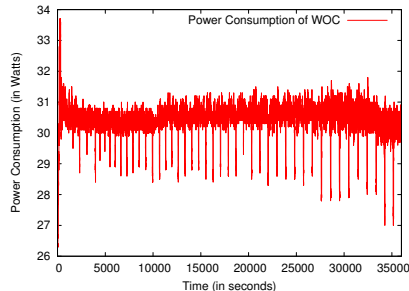Fig. 4a. With concurrent FTP sessions

Fig. 4b. With increasing size of uploaded files

Fig. 4. Power consumption of the WOC

is increased (from 26 Watts to 34 Watts) with the number of concurrent FTP sessions. For the next experiment, we keep only one FTP session and let the WOC perform RE for 10 hours in which the size of uploaded files is increased. The results show that the WOC consumes around 30 Watts on average (Fig. 4b). Therefore, for sake of simplicity, we use an average value of power consumption (30 Watts) to represent additional cost for the router to perform RE.

---

[1] The full figure of the test-bed can be found in [13]

4.2 Simulation Results with GreenRE

We studied ten classical real network topologies extracted from SNDLib [15]. Because CPLEX [8] takes several hours to find an optimal solution even for the smallest network (Atlanta network), we force CPLEX to stop after *two hours of execution*. According to the results of the works mentioned in Section 2.2, we use a redundancy factor $\gamma = 2$ for all the simulations. We present in Fig. 5 results for Atlanta network topology - optimal solutions without RE-routers (*OPT-Green*) given by [9], ILP results (within two hours) with RE-routers (*ILP-GreenRE*), heuristic with RE-routers (*H-GreenRE*) and shortest path routing with RE-routers (*SP-RE*) given by [1]. For worst-case scenario and for comparison with previous work [9], all links are set up with the same capacity C and the demands are all-to-all (one router has to send traffic to all remaining routers on the network) with the same traffic volume D for each demand. The $x-$axis in Fig. 5
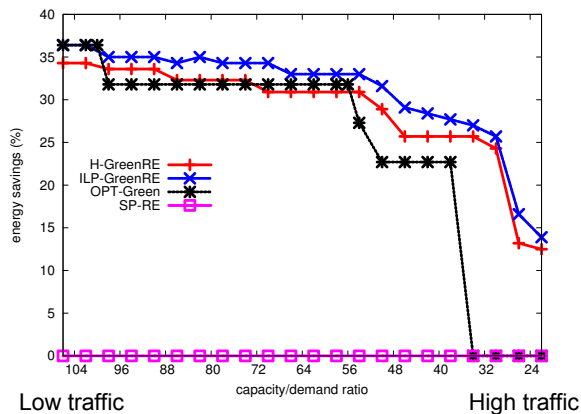

Fig. 5. Simulation results for Atlanta network

represents the *capacity/demand ratio* $(\lambda = C/D)$. Note that in the simulations, λ has the same value and it represents the traffic load on the network. Large value of λ means that the link capacity is much larger than the required bandwidth of traffic demands or we can say the traffic load on the network is low. The power consumption of a link and of a RE-router used in the simulations are 200 Watts [7] and 30 Watts (section 4.1), respectively. Power consumption of a network with RE-routers are calculated as [(the number of active links)*200+ (the number of RE-routers)*30]. Without considering energy-aware routing and without RE-router, all the links on the network are used. Hence, the power consumption is computed as [(the number of links)*200]. The difference between these two values of power consumption represents how much energy can be saved. As shown in Fig. 5, without RE-router (OPT-Green), there is no feasible routing solution and hence, no energy is saved if $\lambda < 38$. When λ increases, links have more bandwidth to aggregate traffic, hence the solutions with and without RE-

router converge to the same amount of energy savings. In general, the heuristic with RE-routers works well and approximate to the results of ILP. In addition, we also consider shortest path routing with RE-routers given by [1]. The results show that: without considering energy-aware routing, there is no energy saving even if all the routers have enabled RE. We present in Fig. 6a the number of
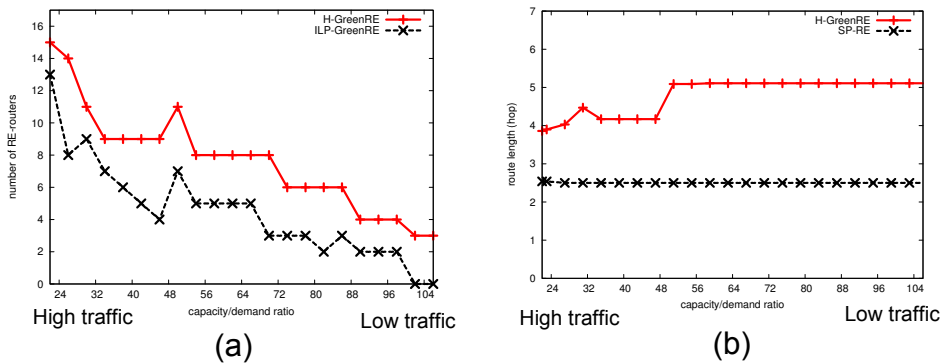

Fig. 6. (a) Number of RE-routers and (b) Average route length

RE-routers used by the heuristic and by the ILP. When λ increases, it is not necessary to have many RE-routers because the links have enough capacity for traffic aggregation. As shown in Fig. 6a, in general, the number of RE-routers is reduced when λ increases. However, there are some points where the number of RE-routers increases. It is because at these points, having more RE-routers allows to turn off more links and hence, reducing the global power consumption of the network.

We compare average route length (which also gives an idea of the delay) between the heuristic algorithm (*H-GreenRE*) and the shortest path routing with RE (*SP-RE*) in Fig. 6b. As we can see, when turning off links on the network, we save energy but the traffic demands need to route on longer paths. When λ increases, more traffic demands can be aggregated to a small number of links, resulting in longer routing paths. However, when this ratio is large enough, the routing solutions do not change much, and hence the average route length seems to be unchanged. Because *SP-RE* does not aggregate traffic, the route length is shorter than those in *H-GreenRE*.

We present in Table 1 energy gain for ten classical network topologies extracted from SNDLib [15]. Because these networks are too large to launch ILP, we compare heuristic results of H-GreenRE and H-Green (the heuristic without RE-router given by [9]). We use λ_{min} be the smallest value of capacity/demand ratio that allows to find a feasible route for all the demands without RE (found in [9]). In the simulations, a range of $\lambda = \{\lambda_{min}, 2\lambda_{min}, 3\lambda_{min}\}$ is used to represent high, medium and low traffic load on the network. As shown in Table 1, with RE-routers, it starts to save a large amount of energy (in average 32.77%)

Network topologies	|V|	|E|	λ_{min}	Traffic volume (capacity/demand ratio λ)					
				with RE-router			without RE-router		
				λ_{min}	$2\lambda_{min}$	$3\lambda_{min}$	λ_{min}	$2\lambda_{min}$	$3\lambda_{min}$
Atlanta	15	22	38	23.6%	30.1%	36.4%	0%	32%	36%
New York	16	49	15	52.2%	61.6%	65.2%	2%	59%	63%
Nobel Germany	17	26	44	23.9%	33.9%	36.8%	0%	35%	39%
France	25	45	67	33.8%	44%	45.4%	0%	42%	44%
Norway	27	51	75	36.2%	45%	47.5%	12%	43%	47%
Nobel EU	28	41	131	27.7%	30.9%	34.2%	12%	32%	34%
Cost266	37	57	175	25.3%	33.6%	35%	3.5%	32%	35%
Giul39	39	86	85	36.5%	48.4%	51.1%	0%	45%	50%
Pioro40	40	89	153	45.3%	53.2%	54.5%	0%	53%	54%
Zib54	54	80	294	23.2%	31.2%	32.7%	0%	30%	33%

Table 1. Gain of energy consumption (in %)

even with $\lambda = \lambda_{min}$. Recall that routing with RE-routers is possible even with $\lambda < \lambda_{min}$ meanwhile no feasible solution is found without RE-router. When λ is large enough, it is not necessary to have RE-routers on the network, hence both the solutions (with and without RE-router) converge to almost the same value of gaining in energy consumption.

5 Discussion

We propose in this paper the GreenRE model with an assumption that all routers on the network can perform RE. However, this model can be extended to support incremental deployment of RE on routers across the Internet. For instance, we have presented in [13] the models that work for the two following cases:

- Only a subset of routers on the network can perform RE. Hence, the model should utilize these RE-routers and find a routing solution so that power consumption of the network is minimized.
- Given a network topology and a maximum number of RE-routers, the model should find where to place these RE-routers and the corresponding routing solution to minimize network power consumption.

6 Conclusion

To the best of our knowledge, GreenRE is the first work which considers RE as a complementary help for energy aware routing. We formally define the model using Integer Linear Programming and propose a greedy heuristic algorithm. The simulations on several network topologies show a significant gain in energy savings. Moreover, in comparison with the traditional energy-aware routing, GreenRE works better especially when the traffic load on the network is high. We also prove by simulations that the heuristic algorithm works well and approximates to the solutions given by the ILP. As part of future work, we will carry

on more simulations with real trace redundancy factor on links and real traffic demands. We will also consider the impacts of GreenRE on the network such as network fault tolerance. Moreover, we plan to develop a distributed algorithm for GreenRE.

Acknowledgment

References

1. Ashok Anand, Archit Gupta, Aditya Akella, Srinivasan Seshan, and Scott Shenker. "Packet Caches on Routers: the Implications of Universal Redundant Traffic Elimination". *Proceedings of ACM SIGCOMM*, 2008.
2. Ashok Anand, Chitra Muthukrishnan, Aditya Akella, and Ramachandran Ramjee. "Redundancy in Network Traffic: Findings and Implications". In *Proceedings of ACM SIGMETRICS*, 2009.
3. Ashok Anand, Vyas Sekar, and Aditya Akella. 'SmartRE: an Architecture for Coordinated Network-wide Redundancy Elimination". In *Proceedings of ACM SIGCOMM*, 2009.
4. A. P. Bianzino, C. Chaudet, D. Rossi, and J. Rougier. "A Survey of Green Networking Research". www.cl.cam.ac.uk/teaching/1011/R02/papers/green-survey.pdf.
5. "BlueCoat: WAN Optimization". http://www.bluecoat.com/.
6. Joseph Chabarek, Joel Sommers, Paul Barford, Cristian Estan, David Tsiang, and Steve Wright. "Power Awareness in Network Design and Routing". In *Proceedings of IEEE INFOCOM*, 2008.
7. L. Chiaraviglio, M. Mellia, and Fabio Neri. "Minimizing ISP Network Energy Cost: Formulation and Solutions". In *IEEE/ACM Transactions on Networking*, 2011.
8. http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/.
9. F. Giroire, D. Mazauric, J. Moulierac, and B. Onfroy. "Minimizing Routing Energy Consumption: from Theoretical to Practical Results". In *Proceedings of IEEE/ACM GreenCom*, 2010.
10. T. Jr. Grevers and J. Christner. "Application Acceleration and WAN Optimization Fundamentals". In *Cisco Press*, 2007.
11. Maruti Gupta and Suresh Singh. "Greening of The Internet". In *Proceedings of ACM SIGCOMM*, 2003.
12. Priya Mahadevan, Puneet Sharma, and Sujata Banerjee. "A Power Benchmarking Framework for Network Devices". In *Proceedings of IFIP Networking*, 2009.
13. Truong Khoa Phan. "Minimization of Network Power Consumption with WAN Optimization". Master's thesis, University of Nice Sophia Antipolis, 2011. www-sop.inria.fr/mascotte/rapports_stages/KhoaPhan_internship-2011.pdf.
14. http://www.riverbed.com/solutions/optimize/.
15. SNDlib. http://sndlib.zib.de.
16. Yang Song, Katherine Guo, and Lixin Gao. "Redundancy-Aware Routing with Limited Resources". In *Proceeding of IEEE ICCCN*, 2010.
17. Neil T. Spring and David Wetherall. "A Protocol-Independent Technique for Eliminating Redundant Network Traffic". In *Proceedings of ACM SIGCOMM*, 2000.