

Robust Energy-Aware Routing with Uncertain Traffic Demands

Heng Lin
Tsinghua University
henglin1991@gmail.com

Mingwei Xu
Tsinghua University
xmwx@cernet.edu.cn

Yuan Yang
Tsinghua University
yyang@csnet1.cs.tsinghua.edu.cn

Abstract—Energy conservation has become a major challenge to the Internet. Switching part of components into sleep mode is an effective way for energy conservation. Many existing approaches compute routing based on traffic matrices, to balance energy saving and traffic engineering goals, e.g., the maximum link utilization ratio (MLUR). However, accurate traffic matrices are difficult to obtain and change frequently, resulting in lack of routing stability and robustness.

We propose to find one energy-aware routing robust to a set of traffic matrices. Such demand-oblivious routing problem has been studied without energy conservation, but it becomes more challenging when energy is considered. To overcome the challenges, we define a metric that reflects the MLUR distance from a routing to the optimal routing under certain energy conservation requirement. We model the problem of minimizing the metric, and analyze the upper bounds. Then, we propose a robust energy-aware routing scheme to solve the problem by selecting sleeping links and computing the routing, based on a classical demand-oblivious routing algorithm. We evaluate our algorithms by simulations on real topologies. The results show that REAR is much more robust than existing approaches while more than 35% line card power can be saved.

I. INTRODUCTION

Energy conservation has become a global concern nowadays. The Internet is one of the major energy consumers, and its rapid growth makes the green Internet a hot research topic. In the Internet backbone, energy is mainly drawn by routers and switches. Such devices consume almost full power even if the traffic load is small. Thus, an effective method to save energy in the Internet is to aggregate traffic into part of the routers when the traffic load is small, and switch the under-utilized components (routers or line cards) into off/sleep mode. Such a method is known as the energy efficient routing.

An important issue for energy efficient routing is to avoid network congestion after the traffic is aggregated. Many approaches have been proposed in existing works. Most approaches compute the routing based on real-time traffic matrices or link loads, to achieve a good load balancing and avoid congestion. However, such approaches come at a cost of obtaining real-time traffic data. Furthermore, the routing may shift frequently with the traffic changes, and a sudden traffic change may still induce congestions. To this end, we need to study the robustness of energy efficient routing.

Specifically, we study the energy efficient routing when traffic matrices cannot be obtained or predicted precisely, i.e., with uncertain traffic demands. To achieve robustness, we need

to find a routing, which can perform near optimally under a range of traffic matrices. The key technique that makes this possible is the advanced *demand-oblivious routing*. A seminal work [14] find that, the distance between the maximum link utilization ratio (MLUR) of a demand-oblivious routing and the MLUR of the optimal routing is bounded. Thus, no matter how the traffic demand changes, the demand-oblivious routing can guarantee certain performance. We note that this conclusion is in a network without sleeping components, and we need to consider the situation when energy conservation is required.

However, existing demand-oblivious routing algorithms cannot be directly applied to energy efficient routing. There are several challenges. First, we need to define a metric that can effectively measure the distance between a robust energy efficient routing and the optimal routing, because the existing metric for demand-oblivious routing fails in the situation when some components are switched into off/sleep mode. Second, we need to analyze whether the metric can be bounded, just like for demand-oblivious routing. If there exists no bound, then it is not feasible to find a robust energy efficient routing. Third, we need practical algorithms to compute the robust energy efficient routing. Specifically, we need to determine: 1) which routers or line cards should be switched into off/sleep mode, to achieve energy efficiency; and 2) in which path to forward the traffic for robustness, while the path does not traverse the sleeping components.

In this paper, we overcome the aforementioned challenges. First, we define a new metric, namely the oblivious performance ratio with energy constraint (OPRE). The OPRE reflects the MLUR distance from a routing to the optimal one when certain energy conservation requirement is satisfied. We model the problem of minimizing the OPRE. Second, we prove that there exists a robust energy efficient routing with the minimum OPRE, which has an upper bound given a network. Then, we propose Robust Energy-Aware Routing (REAR) scheme, which uses heuristic algorithms to solve the problem. We develop algorithm RLP, which chooses sleeping line cards in a way that the OPRE can be minimized potentially. We then develop algorithm RRA to compute the routing in the remaining topology, by extending existing optimal demand-oblivious routing algorithm. We evaluate our algorithms by simulations on real topologies and synthetic traffic demands with random fluctuations. The results show that REAR can

achieve an OPRE of 2.32 while 25% of line card power is saved.

The rest of the paper is organized as follows. Section II shows the related work. Section III presents metric OPRE and formally models the problem. We presents the bounds on OPRE in Section IV, and propose our algorithms in Section V. Section VI shows our simulation setup and results, and Section VII concludes our work.

II. RELATED WORK

There have been a number of studies on energy conservation in the Internet using energy efficient routing. Many approaches take link utilization as a key factor to avoid network congestions. GreenTE [1] is proposed to find a routing that can maximize the power saving by off/sleeping components, while the link utilization must be less than a given threshold. E^2 -MCRA [2] is proposed to find routing for new incoming flows, such that the number of active links and nodes are minimized while the unpredictable traffic increment is considered. ROD [3] computes the link weights using a Lagrange multiplier method, so as to balance the number of sleeping links and the link utilization under an OSPF-like routing. These approaches have to obtain the traffic matrix/demands in real-time. RE-sPoNse [4] uses MPLS tunnels to aggregate traffic and saves energy by switching under-utilized links into sleep mode; the sleeping links are activated when some link has an utilization ratio larger than a given threshold. This method of activating sleeping links is used in some other approaches [5] [6] [7] [8] [9]. Although these approaches do not need a traffic matrix, they still need the traffic volume on each link in the network, and the routing may shift frequently with the traffic change. Our study in this paper is different from the above approaches, because we assume the traffic can be uncertain, and we try to find a static routing that minimizes the distance to the optimal routing with respect to link utilization.

Great efforts have been made to optimize the network performance of delivering traffic. Generally, there are two types of approaches for such traffic engineering targets. The first type assumes the awareness of the traffic matrix/demands, while the other deals with uncertain traffic demands [11]. In this paper, we are interested in the second category, which is called the oblivious routing. Racke [12] proposes an oblivious routing algorithm with a polylog competitive ratio with respect to congestion. Azar et al. [13] propose a polynomial time algorithm to solve the oblivious routing based on the ellipsoid method. Applegate [14] et al. further simplify the problem into a polynomial-size linear programming model. Li et al. [15] propose a penalty method to improve the quality of oblivious routing with respect to path dispersion and path variation. Kodialam et al. [16] study the problem of providing bandwidth guarantee in VPN using oblivious routing. They further extend the application to a set of networking scenarios [17], and to provide resiliency against link failures [18]. Our work in this paper applies oblivious routing in the scenario of energy conservation. We need a new metric that is suitable for our scenario, and new routing algorithms as well.

III. PROBLEM STATEMENT

A. Background of Demand-Oblivious Routing

As mentioned above, demand-oblivious routing aims at finding one routing that performs near optimally under a range of traffic demands. In traffic engineering, a typical metric to evaluate routing performance is the maximum link utilization ratio (MLUR). Clearly, the MLUR is corresponded with a specified traffic matrix (TM). Demand-oblivious routing defines oblivious performance ratio (OPR) to evaluate the routing performance without the knowledge of TM. We briefly present the background.

Given a TM, the distance between a routing to the optimal one is defined as the ratio between their MLURs. Formally, a network is modeled as undirected graph $G(V, E)$, where V is the set of vertices (nodes), and E is the set of edges (links). Let cap_{ij} denote the capacity of the link $(i, j) \in E$. Let d_{ab} denote the traffic demand from origin node a and destination node b , and m denote the TM that contains d_{ab} for all $a, b \in V$. Let $f_{ab}^r(i, j)$ be the fraction of d_{ab} that is routed on link (i, j) ($0 \leq f_{ab}^r(i, j) \leq 1$), using routing r . Routing r is specified by $f_{ab}^r(i, j)$ for all $a, b \in V$ and $(i, j) \in E$, and we will formally define routing consistency later in Section III.C. Let $U_{r,m,G}$ be the MLUR of routing r under TM m . We have

$$U_{r,m,G} = \max_{(i,j) \in E} \frac{\sum_{a,b} d_{ab} f_{ab}^r(i, j)}{cap_{ij}}. \quad (1)$$

Let $P(\{r\}, \{m\}, G)$ be the *performance ratio* of routing r under TM m , which reflects how far from the routing to the optimal one, and is defined as

$$P(r, \{m\}, G) = \frac{U_{r,m,G}}{\min_{r'} U_{r',m,G}}. \quad (2)$$

The *oblivious performance ratio* (OPR) for routing r is defined by extending TM m to a set of TMs M , where M can be the set of all TMs. We have

$$P(r, M, G) = \max_{m \in M} P(r, \{m\}, G). \quad (3)$$

The target of demand-oblivious routing is to find routing r that minimizes OPR $P(\{r\}, M, G)$. Such a “robust” routing is independent of a specific TM, but can perform near optimally. A seminal work [14] uses linear programming to find the routing that minimizes the OPR, and finds that the optimal solution exists, which means that the OPR is bounded.

B. Oblivious Performance Ratio with Energy Constraint

With energy constraint, a network may have to switch part of line cards into off/sleep mode to save the total energy consumption. This changes the network topology and makes metric OPR fail to evaluate the robustness of a routing. We use an example to show this. Assume that G is a cycle with n unit capacity links. Then, the minimal OPR of G is $2 - 2/n$ [14]. Now we pruning one link from G to save energy, and the topology changes to G^* . Because there is only one routing feasible in G^* , OPR $P(\{r\}, M, G^*)$ equals 1. Since 1 is less than $2 - 2/n$ for $n > 2$, it means that the routing after

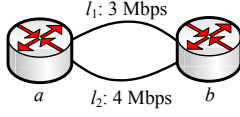


Fig. 1. An example of the OPRE, where the traffic demand is d_{ab} .

pruning one link is more robust than before. However, it is false because there are less links and the network is more likely to be congested.

The intrinsic reason for such a “fake robust” is that the topology is not changing when performance ratio is computed in Eq. (2). To address this issue, we extend the definition of OPR. Formally, let G^* be a sub-graph of G that satisfies the energy constraint (We will formally define the energy constraint in Section III.C, i.e., Ineq. (9)). Note that in this case, a path cannot traverse a sleeping link, so routing r is limited by G^* (Eq. (7) in Section III.C). The extended performance ratio is defined as

$$P(r, \{m\}, G, G^*) = \frac{U_{r,m,G^*}}{\min_{r'} U_{r',m,G}}. \quad (4)$$

Let $P^*(r, M, G, G^*)$ be the *oblivious performance ratio with energy constraint* (OPRE) for routing r . We define $P^*(r, M, G, G^*)$ by extending m to M . Similar to Eq. (3), we have

$$P^*(r, M, G, G^*) = \max_{m \in M} P(r, \{m\}, G, G^*). \quad (5)$$

Note that G^* is used to achieve certain energy conservation target. For an energy conservation target, there may exist a set of different G^* , which result in different OPRE. We will show an example below. When there is no energy constraint, G^* equals G , and Eq. (5) naturally reduces to Eq. (3).

C. Example

We show an example of our new metric OPRE. The network is shown in Fig. 1, where two nodes a, b are connected by two links l_1 and l_2 .¹ The capacities of the links are 3 Mbps and 4 Mbps, respectively. To forward a traffic demand of d_{ab} , the optimal routing is to put $\frac{3}{7}d_{ab}$ on link l_1 and $\frac{4}{7}d_{ab}$ on link l_2 , which results in the MLUR of $\frac{1}{7}d_{ab}$. This routing is also the optimal demand-oblivious routing, because the routing is optimal no matter how d_{ab} changes. Thus, the OPR equals 1.

Now, assume that links l_1 and l_2 consume the same power, and one link has to be switched off/sleep to save energy. Let us check the OPRE after pruning l_1 and l_2 , respectively. If l_1 is pruned, d_{ab} is put on link l_2 , and the MLUR is $\frac{1}{4}d_{ab}$, so the OPRE is $\frac{1}{4}d_{ab}/\frac{1}{7}d_{ab} = 7/4 = 1.75$. If l_2 is pruned, d_{ab} traverses link l_1 , resulting in the MLUR of $\frac{1}{3}d_{ab}$, so the OPRE is $\frac{1}{3}d_{ab}/\frac{1}{7}d_{ab} = 7/3 = 2.33$. The results tell us that pruning link l_1 is more robust than pruning link l_2 . This is consistent with the intuition that link l_2 has a larger capacity and is stronger against congestion.

¹When there are parallel links, we can add a virtual node in each parallel link to keep $G(V, E)$ as a simple graph.

D. Problem Formulation

Now we formally model the problem. Our objective is to minimize OPRE $P^*(r, M, G, G^*)$, where M and G are inputs, and routing r and sub-graph G^* are decision variables. The Min-OPRE problem is as follows.

$$\text{Minimize } P^*(r, M, G, G^*)$$

$$\text{s.t. (1), (4), (5)}$$

$$\forall a, b, i \in V:$$

$$\sum_{j, s.t. (i,j) \in E} f_{ab}^{r'}(i, j) - \sum_{j, s.t. (j,i) \in E} f_{ab}^{r'}(j, i) = \begin{cases} 1, & i = a \\ -1, & i = b \\ 0, & i \neq a, b \end{cases} \quad (6)$$

$$\forall a, b, i \in V:$$

$$\sum_{j, s.t. (i,j) \in E^*} f_{ab}^r(i, j) - \sum_{j, s.t. (j,i) \in E^*} f_{ab}^r(j, i) = \begin{cases} 1, & i = a \\ -1, & i = b \\ 0, & i \neq a, b \end{cases} \quad (7)$$

$$\forall a, b \in V, \forall i, j \in E:$$

$$0 \leq f_{ab}^r(i, j), f_{ab}^{r'}(i, j) \leq 1 \quad (8)$$

$$\sum_{l \in E^*} p(l) / \sum_{l \in E} p(l) \leq 1 - \theta \quad (9)$$

Constraints (1), (4), and (5) follow our definition of the OPRE. Eq. (6) means that the optimal routing r' in G must be a valid routing, i.e. with routing consistency. Specifically, for the source/destination node, all traffic must be routed/received; and for an intermediate node, the traffic flowing out must equal the traffic flowing in. Similarly, Eq. (7) means that the robust routing r in G^* must be a valid routing. Ineq. (8) means that the range of $f_{ab}^r(i, j)$ and $f_{ab}^{r'}(i, j)$ is from 0 to 1. Ineq. (9) means that the power consumption of links in G^* must be less than a given threshold. In Ineq. (9), θ denotes the power saving ratio we must achieve by using G^* . Here, we use $p(l)$ to denote the power consumption of link l , which is mainly drawn by the line cards at the two ends of the link. Note that $p(l)$ is independent of the traffic volume, and this assumption is based on the fact that in current stage, the power of a line card changes little with the traffic volume [20].

Note that if $G^* = G$ satisfy the energy constraint Ineq. (9) and M is the set of all TMs, the Min-OPRE problem reduces to the demand-oblivious routing problem, and can be transformed into a linear program and solved in polynomial time [14]. However, we must find the optimal G^* to achieve the required power saving ratio in other cases, and Min-OPRE is an MILP and is NP-hard in general.

Also note that we put the power saving in the constraint of our problem, instead of the objective function. In such a way, we can balance the trade off between energy conservation and the OPRE, by setting different values of power saving ratio θ . We can even achieve the maximum power saving ratio by a binary search on θ .

IV. UPPER BOUNDS OF THE MINIMUM OPRE

In this section, we analyze the upper bounds of the minimum OPRE. We are interested in two extreme types of networks, i.e., cycles and cliques, because general networks can be seen as intermediate cases between them. We will present the upper bounds with small and large values of θ . We find that, though the minimum OPR of cycles and cliques is similar and bounded by 2 [14], the minimum OPRE is much more different and has a larger upper bound with a larger θ . These theoretical results tell us that the robust energy efficient routing exists, and tell us how close a robust routing can get to the optimal routing.

Lemma 1. $\min P^*(r, M, G, G^*) \geq \min P^*(r, M, G)$ if $G^* \subseteq G$, where the equation holds if $G^* = G$.

Proof. We assume exist a G^* , s.t. $\min P^*(r, M, G, G^*) < \min P^*(r, M, G)$. Because G^* is the subset of G , we directly use the robust routing of G^* to G , then $P^*(r, M, G)$ will equal to $P^*(r, M, G, G^*)$, and arise contradiction. Particularly, when $G^* = G$, G^* is the subset of G , and G is the subset of G^* , so there is both $\min P^*(r, M, G, G^*) \geq \min P^*(r, M, G)$ and $P^*(r, M, G) \geq P^*(r, M, G, G^*)$, so they are equal to each other when $G^* = G$. This ends our proof. \square

Theorem 2. The minimum OPRE of C_n (the cycle on n vertices with unit capacity links) and K_n (the complete graph on n vertices with unit capacity links) is $2 - 2/n$ if $\theta = 0$.

Proof. According to Lemma 1, the minimum OPRE of C_n is the minimum OPR, i.e., $\min P^*(r, M, C_n)$, when $\theta = 0$. Similarly, the minimum OPRE of K_n is $\min P^*(r, M, K_n)$ when $\theta = 0$. On the other hand, we know from [14] that the minimum OPR of C_n is $2 - 2/n$, and the minimum OPR of K_n is also $2 - 2/n$. This ends our proof. \square

Theorem 2 shows the minimum OPRE when θ equals 0 and all link capacities are the same. In a more general case when there are different link capacities, we have

Theorem 3. The upper bound of the minimum OPRE for a cycle/complete graph on n nodes is $1 + \frac{\max cap_{ij}}{\min cap_{ij}}$ if $\theta = 0$.

Proof. See Appendix A. \square

Theorem 3 tells us that the minimum OPRE is related to the link capacity difference in the network. The above results are for $\theta = 0$. Now let us see the results for a large θ . Since there is not much difference between the power consumptions for links with different capacities [8], we consider the spanning trees of a graph, which can achieve a near-optimal θ . We have

Theorem 4. Let G^* be a spanning tree of a cycle on n nodes, and then the upper bound of the OPRE is $1 + \frac{\max cap_{ij}}{\min cap_{ij}}$.

Proof. See Appendix B. \square

Theorem 5. Let G^* be a spanning tree of a complete graph on n nodes, and then the upper bound of the OPRE is $\frac{n^2}{2} \frac{\max cap_{ij}}{\min cap_{ij}}$.

Proof. See Appendix C. \square

Theorem 4 and Theorem 5 present a large gap between the OPRE upper bounds in the two types of networks, because when G^* is a spanning tree, more links are pruned in the complete graph. This implies that in order to achieve more energy conservation, the OPRE may become larger quickly, and the routing becomes less robust, in the worst case. However, we can still develop effective algorithms to achieve a small OPRE and save energy in general cases, i.e., in real-world networks which have much less link densities than a complete graph.

V. ROBUST ENERGY-AWARE ROUTING ALGORITHMS

In this section, we present our REAR approach to solve the Min-OPRE problem. We first develop the robust link pruning (RLP) algorithm. RLP selects sleeping links that we will prune from the topology, to achieve the required power saving ratio and potentially reduce the OPRE. Then, we develop the robust routing adjustment (RRA) algorithm. RRA adjusts the demand-oblivious routing that has the minimum OPR, to detour the traffic around the sleeping links.

A. Algorithm RLP

Several heuristics have been proposed to select sleeping links without the knowledge of a specific TM, with various considerations [5] [6] [7] [10]. However, we need an algorithm that can potentially minimize the OPRE. To this end, we first find the demand-oblivious routing that minimizes the OPR, using the linear programming approach in [14]. Based on the routing, we define the extended robust link utilization (ERLU), to evaluate the contribution of a link to the OPRE.

Intuitively, a link is less preferred to be pruned if there are more flows traversing the link. Though we are not aware of the TM, the traffic volume is likely to fluctuate largely if there are many flows. This means that if we prune this link, these flows have to be carried by other links in the network, and the OPRE may increase largely. On the other hand, a link is less preferred to be pruned if the link has a less capacity. This is because in the optimal demand-oblivious routing where the OPR is minimized, such a link is not likely to be heavy-loaded, and we should keep the link to maintain the robustness. Formally, let r_0 be the optimal demand-oblivious routing, and u_{ij}^e be the ERLU of link (i, j) , we have

$$u_{ij}^e(r_0) = \frac{\sum_{a,b} f_{ab}^{r_0}(i, j)}{cap_{ij}} \quad (10)$$

With the ERLU, we iteratively select the link with the smallest ERLU as the sleeping link. If the topology after pruning the selected link is not connected, we pass the link and continue to search. We stop the process once we have sufficient sleeping links to achieve the required power saving ratio. We develop the RLP algorithm.

The inputs of RLP include graph $G(V, E)$ and required power saving ratio θ , and the output is the sleeping link set S . The ERLU of each link is computed in Steps 1 to 3. S is initialized to be empty (Step 4). In each iteration of the outer

Algorithm RLP()

Input: $G(V, E)$, θ ;
Output: the set of sleeping links S ;
1: $r_0 \leftarrow$ the optimal demand-oblivious routing [14] on G ;
2: **for** each link l in E
3: compute ERLU $u_l^e(r_0)$, using Eq. (10);
4: $S \leftarrow \emptyset$; $goon \leftarrow \text{true}$;
5: **while** $goon$
6: $goon \leftarrow \text{false}$;
7: **for** each link l in $E - S$ with increasing order based on $u_l^e(r_0)$
8: $G^* \leftarrow (V, E^* = E - S - \{l\})$;
9: **if** G^* is connected and $\sum_{l' \in E^*} p(l') / \sum_{l' \in E} p(l') > 1 - \theta$
10: $S \leftarrow S \cup \{l\}$;
11: $goon \leftarrow \text{true}$;
12: **break**;
13: **return** S ;

loop (Steps 5 to 12), a feasible sleeping link is selected (Steps 7 to 12). If a feasible sleeping link is found and the power saving ratio is not achieved yet (Step 9), the selected link is added into set S (Step 10), binary variable $goon$ will be true (Step 11), and the process continues (Step 12). Or else, binary variable $goon$ will remain false (Step 6), and the algorithm ends. The computation complexity of the RLP algorithm is $O(LP + |E||V|^2 + (|E| - |V|)|V||E| \log |E|)$ in the worst case, where $O(LP)$ is for solving the linear programming in [14], $O(|E||V|^2)$ is for computing the ERLU, $O(|E| - |V|)$ is the maximum number of sleeping links, and $O(|V||E| \log |E|)$ is for the connectivity check and the sleeping link search.

B. Algorithm RRA

After pruning the links, we get graph G^* (if exists) that satisfies the energy conservation requirement. Now we should compute the robust routing that minimizes the OPRE. Note that we cannot simply apply the algorithm in [14] on G^* , because the algorithm minimizes OPR but not OPRE. We observe that the optimal demand-oblivious routing r_0 minimizes the OPR in G , and minimizes OPRE when $G^* = G$ according to Theorem 2. Thus, to obtain a robust routing in G^* , we take a heuristic that adjusts routing r_0 to detour the traffic around the sleeping links in G^* .

Note that there may be multiple paths in r_0 between a source-destination pair. We first decompose the paths by a standard procedure given in [21]. Formally, let p_{ab}^h denote the h -th path between nodes a and b , and t_{ab}^h be the fraction of the traffic that is routed on p_{ab}^h . For each p_{ab}^h that traverses a sleeping link, we must find some detour paths to absorb t_{ab}^h . Such detour paths should not increase the OPRE too much. Recall that ERLU $u_{ij}^e(r)$ reflects the link utilization ratio without the knowledge of the TM, under routing r . Thus, we prefer a path whose maximum $u_{ij}^e(r)$ is small. On the other hand, the detour paths should not be too long, in order to limit the path stretch. To this end, we use YEN's algorithm [22] to compute the k -th shortest path in graph G^* . We develop Algorithm RRA.

The inputs of RRA include graph $G(V, E)$, sleeping link set S , optimal demand-oblivious routing r_0 in G , and N which will be used latter. The output is the robust energy efficient routing r . G^* and r are initialized in Step 1. The main loop

Algorithm RRA()

Input: $G(V, E)$, sleeping link set S ,
optimal demand-oblivious routing r_0 , N ;
Output: robust energy-aware routing r ;
1: $G^* \leftarrow G(V, E - S)$; $r \leftarrow r_0$;
2: **for** each link l in S
3: **for** each $s, d \in V$
4: $t \leftarrow f_{sd}^r(l)$;
5: $paths \leftarrow$ the extracted paths for s, d ;
6: **for** each path $p_{sd}^h \in paths$
7: **if** $l \in path$
8: $paths.remove(p_{sd}^h)$;
9: $yen_paths \leftarrow yens_algorithm(G^*, s, d)$;
10: **for** $counter = 1$ to N
11: compute $u_{ij}^e(r)$;
12: $p \leftarrow$ the path in yen_paths with the min. $\max_{ij} u_{ij}^e(r)$;
13: $p.fraction = t/N$;
14: $paths.add(p)$;
15: $r \leftarrow paths.merge()$;
16: **return** r ;

from Step 2 to Step 14 finds the detour paths for each sleeping link in S , and for each source-destination pair. The traffic fraction that should be detoured is denoted by t (Step 4), and the original paths are extracted and removed from routing r (Steps 5 to 8). Then, the k -th shortest paths are computed in Step 9. The traffic fraction of t is detoured by N times evenly (Steps 10 to 14), and the detour path with the minimum value of the ERLU is selected (Steps 11 and 12). In Step 15, the duplicate paths are merges and new routing r that does not traverse link l is obtained. The computation complexity of the RRA algorithm is polynomial. We omit the detail due to the space limitation.

VI. PERFORMANCE EVALUATION

We evaluate the REAR approach by simulations on three real-world network topologies, including Abilene [23], Geant [24], and CERNET2 [25]. The numbers of nodes and links of the topologies are listed in Table I. The traffic matrices we used in our simulations are generated as follows. We first generate a baseline traffic matrix using the gravity model [19], where the baseline traffic demand d_{ab} is proportional to the total capacity of nodes a and b . Then, the traffic demand is generated randomly in the range from d_{ab}/ω to $d_{ab}\omega$, where $\omega \geq 1$ is a margin factor used to control the fluctuation range of the traffic. We generate 1000 random TMs for each topology with each ω . The power consumption of different line cards is set as constant in the simulations, as shown in Table II, following [1].

TABLE I
TOPOLOGIES

Topology	Nodes	Links	Links can be Removed
Abilene	12	15	4
Geant	23	37	15
CERNET2	20	22	3

For comparison, we simulate a recent energy efficient routing approach, namely DMP [6], which also does not use the TMs as inputs. DMP always selects the most power-hungry links to switch into off/sleep mode, and can achieve a large

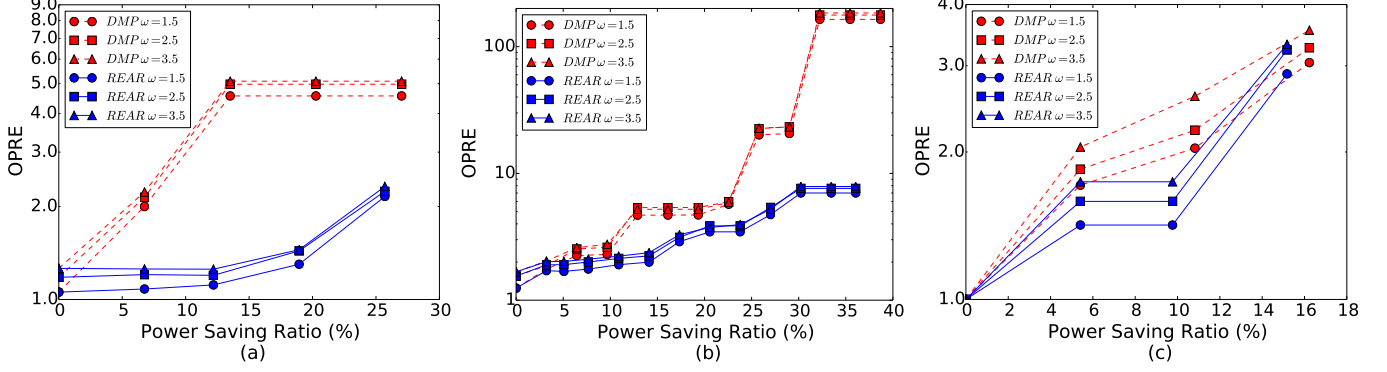


Fig. 2. OPRE in (a) Abilene. (b) Geant. (c) CERNET2.

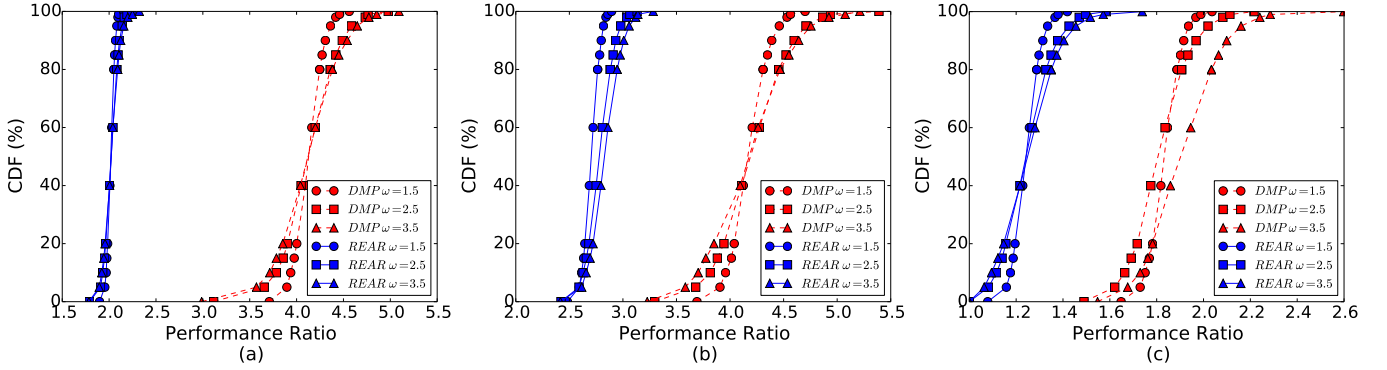


Fig. 3. Performance ratio in (a) Abilene with 4 links pruned. (b) Geant with 6 links pruned. (c) CERNET2 with 2 links pruned.

power saving ratio, but does not consider routing robustness. For the approaches, we evaluate the OPRE, the power saving ratio, the distribution of the performance ratio, and the path stretch.

TABLE II
LINE CARD POWER CONSUMPTION

Line-Card	Speed(Mbps)	Power(Watts)
1-Port OC3	155.52	60
8-Port OC3	1244.16	100
1-Port OC48	2488.32	140
1-Port OC192	9953.28	174

A. OPRE versus Power Saving

Fig. 2 (a) shows the OPRE (evaluated by the maximum performance ratio under all 1000 TMs), as a function of power saving ratio θ , in the Abilene topology. We see that the OPRE increases when more power is saved. This is not surprising because the routing becomes less robust when more links are switched into off/sleep mode. Our REAR has an OPRE ranging from 1.06 to 2.32, which is much less than the results of DMP. This is because REAR takes routing robustness into consideration when it prunes links and computes routing. For instance, when $\omega = 3.5$ and 25% power is saved, the OPRE of REAR is 2.32, while the OPRE of DMP is 5.09. This implies that REAR is more robust against traffic uncertainty. We also see that a larger ω results in a larger OPRE, because the

traffic volume fluctuates in a wider range. For instance, when ω increases from 1.5 to 3.5, the OPRE of REAR increases about 0.17, while the OPRE of DMP increases about 0.53.

Fig. 2 (b) shows the OPRE as a function of power saving ratio θ in Geant. We find that the results are similar to that in Abilene. However, the OPRE is larger in Geant than in Abilene for both REAR and DMP. One reason for this result is that Geant has a larger link density ($\frac{37}{23} = 1.61$) than Abilene ($\frac{15}{12} = 1.25$). Another reason is that the link capacity difference is large in Geant. The maximum link capacity is 9920 Mbps while the minimum link capacity is 155 Mbps. Such results are consistent with our analysis in Section IV.

Fig. 2 (c) shows the results in CERNET2. We observe that the difference between REAR and DMP is small. This is because the CERNET2 topology has a small link density, and only a small number of links can be pruned.

B. Performance Ratio Distribution

Fig. 3 (a) shows the CDF of random TMs as a function of the performance ratio, in Abilene with 4 links pruned. The curves of REAR are on the left of DMP, which means that REAR has a lower performance ratio under all TMs. We observe that the performance ratio has a larger range with a larger ω , but the performance ratio of REAR has a small range in total (e.g., from 1.74 to 2.32 when $\omega = 3.5$), unlike DMP (e.g., from 2.97 to 5.09 when $\omega = 3.5$). This implies that REAR is more robust.

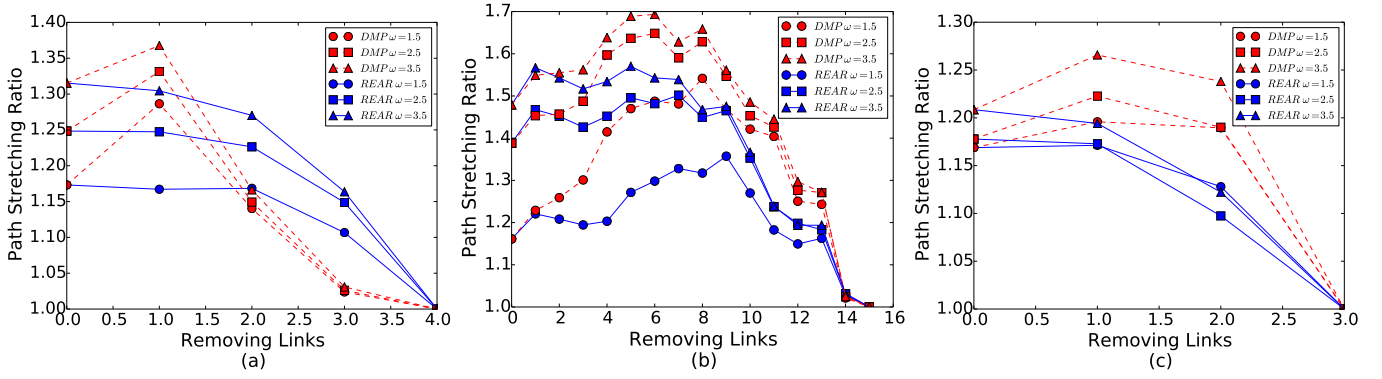


Fig. 4. Path Stretch in (a) Abilene. (b) Geant. (c) CERNET2.

Fig. 3 (b) shows the results in Geant with 6 links pruned, and Fig. 3 (c) shows the results in CERNET2 with 2 links pruned. The results are similar to that in Abilene.

C. Path Stretch

Now we evaluate the path stretch of the approaches, which reflects the end-to-end delay increment due to link pruning and robust routing. Here, path stretch is the ratio of the path length of a routing to the shortest path length, in the topology after pruning links. Note that there may be multiple paths in the robust routing for a source-destination pair. In such a case, we compute the weighted path stretch of these paths according to the traffic fraction they carry.

Fig. 4 (a) shows the path stretch as a function of pruned link number in Abilene. Generally, the path stretch decreases when more links are pruned, and equals 1 when 4 links are pruned. This is because the topology becomes less connected when more links are pruned, and reduces to a spanning tree when 4 links are pruned. The path stretch of REAR is less than 1.32, because we use YEN's algorithm to find the k -th shortest path when searching for paths to detour the traffic. We think this path stretch is small and will not induce large end-to-end delay increment. The path stretch of REAR is a little larger than that of DMP in some cases. This is because most links in Abilene has the same capacity and consumes the same power, and in such a case, DMP randomly selects sleeping links, which may result in less path stretch. XXXXXX We also observe that the path stretch increases with ω , because the demand-oblivious routing is computed based on ω , and when ω increases, the routing will use more paths to increase the robustness. XXXXXX

Fig. 4 (b) and (c) show the path stretch in Geant and CERNET2, respectively. We see that our REAR has a less path stretch than DMP. For instance, REAR has the largest path stretch of 1.35 in Geant when $\omega = 1.5$, while the result of DMP is 1.54.

VII. CONCLUSION AND FUTURE WORK

In this paper, we studied energy efficient routing under uncertain traffic demands. We proposed metric OPRE, which defines the distance between a routing and the optimal routing

with respect to the maximum link utilization ratio under energy conservation requirements. We modeled the problem of minimizing the OPRE and analyzed the upper bounds of the solution. Then, we developed the REAR scheme to solve the problem. We did simulations using real network topologies and showed that REAR can saving energy without knowing traffic matrices, and can achieve a smaller OPRE than existing approaches.

In the future, we plan to improve our algorithms, because solving the linear programming does not scale well to large networks. We are also interested in realizing the robust energy-aware routing in a hop-by-hop fashion, without fractional traffic and MPLS tunnels.

REFERENCES

- [1] M. Zhang, C. Yi, B. Liu, et al. "GreenTE: Power-Aware Traffic Engineering," in Proc. of IEEE ICNP, October 2010.
- [2] S. Avallone, G. Ventre. "Energy efficient online routing of flows with additive constraints," Computer Networks, vol. 56, 2368–2382, 2012.
- [3] M. Shen, H. Liu, K. Xu, et al. "Routing On Demand: Toward the Energy-Aware Traffic Engineering with OSPF," in Proc. of IFIP Networking, 2012, 232–246.
- [4] N. Vasic, P. Bhurat, D. Novakovic, M. Canini, S. Shekhar, and D. Kostic, "Identifying and using energy-critical paths," in Proc. of CoNext'11, 2011.
- [5] A. Cianfrani, V. Eramo, M. Listanti, and M. Polverini, "An OSPF enhancement for energy saving in IP networks," in IEEE INFOCOM Workshop on Green Communications and Networking, April 2011.
- [6] A.P. Bianzino, L. Chiaraviglio, M. Mellia, "Distributed algorithms for green IP networks," in: IEEE INFOCOM Workshop on Green Networking and Smart Grid, 2012
- [7] A. P. Bianzino, L. Chiaraviglio, M. Mellia, et al. "GRiDA: GReen Distributed Algorithm for energy-efficient IP backbone networks," Computer Networks, vol. 56, 3219–3232, 2012.
- [8] Y. Yang, M. Xu, Q. Li. "Towards fast rerouting-based energy efficient routing," Computer Networks, vol. 70(9), 1–15, 2014.
- [9] Q. Li, M. Xu, Y. Yang, L. Gao, Y. Cui, and J. Wu. "Safe and Practical Energy-Efficient Detour Routing in IP Networks," IEEE/ACM Transactions on Networking, vol. 22(6), 1925–1937, 2014.
- [10] F. Cuomo, A. Abbagnale, A. Cianfrani, and M. Polverini, "Keeping the Connectivity and Saving the Energy in the Internet," in IEEE INFOCOM Workshop on GCN, April 2011.
- [11] N. Goyal, N. Olver, F. B. Shepherd. "Dynamic vs. Oblivious Routing in Network Design," Algorithmica, vol. 61(1), 161–173, 2011.
- [12] Harald Räcke. "Minimizing congestion in general networks," In Proceedings of the 43rd IEEE Symposium on Foundations of Computer Science (FOCS), 2002. Co-Winner of Best Paper Award.
- [13] Y. Azar, E. Cohen, A. Fiat, H. Kaplan, and H. R. acke. "Optimal oblivious routing in polynomial time," In Proceedings of the 35th ACM Symposium on the Theory of Computing, 2003.

- [14] D. Applegate and E. Cohen, "Making Intra-Domain Routing Robust to Changing and Uncertain Traffic Demands: Understanding Fundamental Tradeoffs," in *Proceedings of ACM SIGCOMM 03*, Karlsruhe, Germany, August 2003.
- [15] Y. Li, J. Harms, R. Holte, "A simple method for balancing network utilization and quality of routing," in *Proc. of ICCCN*, 2005.
- [16] M. Kodialam, T. V. Lakshman, S. Sengupta, "Traffic-Oblivious Routing for Guaranteed Bandwidth Performance," *IEEE Communications Magazine*, vol. 45(4), 46–51, 2007.
- [17] M. Kodialam, T. V. Lakshman, J. B. Orlin, S. Sengupta, "Oblivious Routing of Highly Variable Traffic in Service Overlays and IP Backbones," *IEEE/ACM Trans. on Netw.*, vol. 17(2), 459–472, 2008.
- [18] M. Kodialam, T. V. Lakshman, J. B. Orlin, S. Sengupta, "End-to-End Restorable Oblivious Routing of Hose Model Traffic," *IEEE/ACM Trans. on Netw.*, vol. 19(4), 1223–1236, 2011.
- [19] M. Roughan, A. Greenberg, C. Kalmanek, M. Rumsewicz, J. Yates, and Y. Zhang, "Experience in measuring backbone traffic variability: models, metrics, measurements, and meaning," *InProceedings of the 2nd Internet Measurement Workshop*. ACM, 2002.
- [20] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsang, and S. Wright, "Power Awareness in Network Design and Routing," in *Proc. of IEEE INFOCOM*, 2008, pp. 457–465.
- [21] R. K. Ahuja, T. L. Magnanti, J. B. Orlin. *Network flows: theory, algorithms, and applications*. Prentice Hall, Feb., 1993.
- [22] Yen, Jin Y. "Finding the k Shortest Loopless Paths in a Network," *Management Science* 17 (11), 1971.
- [23] "Abilene Core Topology," [Online], Available: <https://itservices.stanford.edu/service/network/internet2/abilene>
- [24] "Geant Topology," [Online], Available: <http://geant3.archive.geant.net/Network/NetworkTopology/pages/home.aspx>
- [25] "CERNET2 Topology," [Online], Available: <http://www.edu.cn/20060111/3170220.shtml>

APPENDIX A PROOF OF THEOREM 3

Proof. We compute the optimal routing for some arbitrary TM and forward the demands on the topologies. Let d_l be the traffic on link l , and cap_l be the capacity of link l . For circles and cliques are highly homomorphic, without loss of generality we say the link l_1 is the bottleneck, i.e. the link with MLUR.

Routing can be adjusted. We image a simple situation that forward the traffic from one link to another path base on the routing computed above. The link utilization of links on the path will increase, while accordingly the link utilization of adjusted link will reduce to 0. Obviously, the adjusted routing must not be better than the optimal robust routing, so its upper bound also is the bound of minimum OPRE. Following proof is according to which the adjusted link is, and which the bottleneck is after adjusting routing.

For Case I.I, the adjusted link is l_2 , the new bottleneck still be l_1 . Because l_1 is the bottleneck before adjusting, so we have inequation:

$$\frac{d_2}{cap_2} \leq \frac{d_1}{cap_1} \iff \frac{d_2}{d_1} \leq \frac{cap_2}{cap_1} \quad (11)$$

we compute OPRE for Case I.I combined with Ineq. (11):

$$\frac{\frac{d_1+d_2}{cap_1}}{\frac{d_1}{cap_1}} = 1 + \frac{d_2}{d_1} \leq 1 + \frac{cap_2}{cap_1} \quad (12)$$

For Case I.II, the adjusted link also be l_2 , but the new bottleneck may be l_3 . Because l_1 is the bottleneck before adjusting, so we have inequations:

$$\frac{d_3}{cap_3} \leq \frac{d_1}{cap_1} \iff \frac{d_3 * cap_1}{d_1 * cap_3} \leq 1 \quad (13)$$

$$\frac{d_2}{cap_2} \leq \frac{d_1}{cap_1} \iff \frac{d_2 * cap_1}{d_1} \leq cap_2 \quad (14)$$

We compute OPRE for Case I.II combined with Ineq. (13) and Ineq. (14):

$$\frac{\frac{d_3+d_2}{cap_3}}{\frac{d_1}{cap_1}} = \frac{d_3 * cap_1}{d_1 * cap_3} + \frac{d_2 * cap_1}{d_1 * cap_3} \leq 1 + \frac{cap_2}{cap_3} \quad (15)$$

For Case II, the adjusted link is l_1 , namely the old bottleneck link, the new bottleneck maybe l_3 . Computing the OPRE combined with Ineq. (13):

$$\frac{\frac{d_3+d_1}{cap_3}}{\frac{d_1}{cap_1}} = \frac{d_3 * cap_1}{d_1 * cap_3} + \frac{cap_1}{cap_3} \leq 1 + \frac{cap_1}{cap_3} \quad (16)$$

To sum up, we have inequation from Ineq. (12), Ineq. (15) and Ineq. (16):

$$1 + \max\left\{\frac{cap_2}{cap_1}, \frac{cap_2}{cap_3}, \frac{cap_1}{cap_3}\right\} \leq 1 + \frac{\max_{l \in E} cap_l}{\min_{l \in E} cap_l} \quad (17)$$

□

APPENDIX B PROOF OF THEOREM 4

Proof. The spanning tree of circle is the topology which lose one link. We regard this link as the adjusted link mentioned in Appendix. A. The complete proof is almost the same with Appendix. A.

APPENDIX C PROOF OF THEOREM 5

Proof. Suppose clique has n nodes, every link in the spanning tree will split the tree to two components. Let l_3 be the bottleneck in the spanning tree, Let S, D be the node sets of two components, and every flow between S and D always trace on l_3 . Without loss of generality, we say l_1 is the bottleneck link for the optimal routing on the original clique.

Because the l_1 is the bottleneck link, we have:

$$\frac{d_l}{cap_l} \leq \frac{dem_1}{cap_1} \iff \frac{d_l * cap_1}{d_1} \leq cap_l, \forall l \in (S, D) \quad (18)$$

The l_3 is the bottleneck finally, we have OPRE combined with Ineq. (18):

$$\frac{\sum_{l \in (S, D)} \frac{d_l}{cap_3}}{\frac{d_1}{cap_1}} = \frac{cap_1}{cap_3} \sum_{l \in (S, D)} \frac{d_l}{d_1} \leq \sum_{l \in (S, D)} \frac{cap_l}{cap_3} \quad (19)$$

And we know that, the amount of l must less than $\frac{n^2}{2}$ (equal to $\frac{n^2}{2}$ if $|S| = |D| = \frac{n}{2}$), so OPRE must lower than:

$$\sum_{l \in (S,D)} \frac{cap_l}{cap_3} \leq \frac{n^2}{2} \frac{\max_{l \in (S,D)} cap_l}{\min_{l \in (S,D)} cap_l} \tag{20}$$

□