GreenTE: Power-Aware Traffic Engineering

Mingui Zhang*‡ zmg06@mails.tsinghua.edu.cn
Cheng Yi† yic@email.arizona.edu
Bin Liu* liub@tsinghua.edu.cn
Beichuan Zhang† bzhang@arizona.edu

***Abstract*—Current network infrastructures exhibit poor power efficiency, running network devices at full capacity all the time regardless of the traffic demand and distribution over the network. Most research on router power management are at component level or link level, treating routers as isolated devices. A complementary approach is to facilitate power management at network level by routing traffic through different paths to adjust the workload on individual routers or links. Given the high path redundancy and low link utilization in today's large networks, this approach can potentially allow more network devices or components to go into power saving mode. This paper proposes an intra-domain traffic engineering mechanism, *GreenTE*, which maximizes the number of links that can be put into sleep under given performance constraints such as link utilization and packet delay. Using network topologies and traffic data from several wide-area networks, our evaluation shows that GreenTE can reduce line-cards' power consumption by 27% to 42% under constraints that the maximum link utilization is below 50% and the network diameter remains the same as in shortest path routing.**

I. INTRODUCTION

The Internet as an indispensable communication system in our society also has its share in energy consumption. Research on energy management has traditionally focused on battery-operated devices, and more recently, stand-alone servers and server clusters in data centers. The underlying network infrastructure, namely routers, switches and other network devices, still lacks effective energy management solutions. Epps *et al.* [1] from Cisco report that a high-end router CRS-1 with maximum configuration can consume as much as one MegaWatt. The same report also points out that driven by exponential growth of Internet traffic, router system requirements are outpacing silicon and cooling technologies. In addition to electricity bills, the large power consumption by network devices also puts a lot of stress on power delivery to and heat removal from router components as well as the hosting facility. With the advent of cloud computing and large data centers, the problem will only get worse. In short, router power consumption has become an increasing concern for Internet Service Providers (ISPs), Internet Exchange Points (IXPs), and data centers.

Existing research on router power management treats routers as isolated devices and focuses on reducing power consumption at hardware component level. Recently Gupta *et al.* [2] suggested to consider routers in the network context and create more power saving opportunities by adjusting the amount of traffic going through routers, but they did not propose specific solutions. There are link-level solutions which put line-cards to sleep when there is no traffic on the link [3], however, the power saving from opportunistic sleeping is limited by the inter-arrival time of packets.

Complementary to component-level and link-level solutions are network-level solutions. Today's networks are designed and operated to carry the most traffic in the most reliable way without considerations of energy efficiency. A network usually builds many redundant links and aggressively over-provisions link bandwidth to accommodate potential link failures and traffic bursts. While these redundant links and bandwidth greatly increase the network reliability, they also greatly reduce the network's energy efficiency as all network devices are powered on at full capacity 24x7 but highly under-utilized most of the time. Rule of thumb states that today's backbone links are used by 40% or lower [4] in their capacity. The high path redundancy and low link utilization provide unique opportunities for power-aware traffic engineering. Intuitively, when there are multiple paths between the same origin-destination (OD) pair, and the traffic volume on each path is low, one can move the traffic to a fewer number of paths so that the other paths do not carry any traffic for an extended period of time. Routers that have idle links can then put the links to sleep for energy conservation. This approach can be combined with component-level and link-level solutions to achieve higher network energy efficiency.

Network-level solutions require network-wide coordination of routers. The challenges are two-fold, namely how to manipulate the routing paths to make as many idle links as possible to maximize the power conservation, and how to achieve power conservation without significantly affecting network performance and reliability. Since power-aware traffic engineering uses fewer number of links at any moment, it is important to make sure that links are not overloaded and packets do not experience extra long delays.

This paper proposes GreenTE, a power-aware traffic engineering mechanism that reduces network power consumption while still maintaining network performance at desired levels. GreenTE is formulated as a Mixed Integer Programming (MIP) problem with the total power saving as the objective to be maximized. Performance requirements such as maximum link utilization (MLU) and network delay are considered as constraints in the problem. While the problem formulation bears similarity to that of traditional traffic engineering research, the main contribution of this work is the solution results. Traditional traffic engineering and power-aware traffic engineering

---

*Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China.

†Computer Science Department, The University of Arizona, Tucson, AZ 85721, USA.

‡This work was done during Mingui Zhang's visit to the University of Arizona. He will join Huawei Technologies Co. Ltd.

This work is partially supported by US NSF award CNS-0721863, China NSFC (60625201, 60873250), 973 project (2007CB310701) and Tsinghua University Initiative Scientific Research Program.

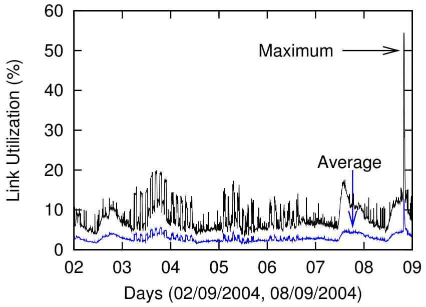


Fig. 1. Maximum and average link utilization in the Abilene network

have two opposite optimization goals: the former tries to spread traffic evenly to all the links, while the latter tries to concentrate traffic to a subset of the links. It is unclear whether one can achieve significant power saving while still maintaining acceptable link utilization in real networks. We solve the power-aware traffic engineering problem using real network topologies and traffic data, demonstrating that it is both promising and feasible.

Using network topologies and traffic data from two wide-area research networks, Abilene and GÉANT, we evaluate GreenTE in terms of power saving, link utilization, packet delay and routing stability. Results show that GreenTE can achieve 27% to 42% power saving on line-cards under the constraints that maximum link utilization is below 50% and the network diameter remains the same as that in pure OSPF routing. The number of MPLS tunnels needed is small compared with full mesh, and routing is largely stable as more than 70% of the MPLS tunnels remain unchanged from one adjustment period to the next one. We also show that GreenTE can be applied to large commercial networks such as Sprint and AT&T and achieve similar power savings too.

The rest of the paper is organized as follows. Section II gives an overview of the basic idea and its assumptions. Section III formulates the power saving problem as a traffic engineering problem and presents the GreenTE model. Section IV discusses potential implementation issues. Section V evaluates GreenTE using topologies and traffic matrices from several real networks with collected traces and synthesized data. Section VI reviews the previous work and Section VII concludes the paper.

II. BASIC IDEA AND ASSUMPTIONS

Today's wide-area networks usually have redundant and over-provisioned links, resulting in low link utilization during most of the time. Figure 1 shows the maximum and average link utilization under OSPF routing in Abilene, a large US education backbone, during a typical week. The average link utilization is only about 2%, the maximum fluctuates mostly between 10% and 20%, and only one rare event pushes the maximum over 50%. Such behavior is common in large commercial networks as well.

High path redundancy and low link utilization combined also provide a unique opportunity for power-aware traffic engineering as illustrated by the example in Figure 2. Traditional traffic engineering spreads the traffic evenly in a network (Figure 2(a)), trying to minimize the chance of congestion induced by traffic bursts. However, in power-aware traffic engineering (Figure 2(b)), one can free some links by moving their traffic onto other links, so that the links without traffic can go sleep for an extended period of time. This should result in more power saving than pure opportunistic link sleeping because the sleep mode is much less likely to be interrupted by traffic.

TABLE I
THE CONFIGURATION OF A CISCO 12000 ROUTER [5]

Slot	Cardtype	Watts
1	OC3-4-POS-X	90
2	GE-4	106
6	OC3-POS-16	100
7	OC12-ATM-4	122
8	OC3-4-POS-X	90
5, 9	GSRP	38
16, 17	CSC10	19
18∼22	SFC10	64
24, 25	ALARM10	33
29	BLOWER16	178

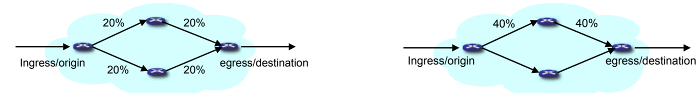
TABLE II
THE POWER BUDGET OF A CISCO 12000 ROUTER [5]

Slot	Category	Watts
1, 2, 6, 7, 8	line cards	508 W
5, 9	Route processors	76 W
16∼22, 24, 25, 29	Chassis components	602 W
	Total inuse power	1186 W

In this paper, we focus on saving power by turning off links, or interchangeably, putting line-cards (or their ports) into sleep mode. Line-cards contribute a significant portion to the total power consumption of a router. Table I shows a typical configuration of a Cisco 12000 series router with low to medium interface rates and Table II shows its budget of in-use power consumption. All the line-cards together consume 508 Watts, about 43% of the router's total power budget. This particular configuration uses relatively low rate interfaces (less than 1Gb/s) and the router is also of an old model. With faster interfaces (10Gb/s or even 40Gb/s) in newer routers, line-cards' power consumption will constitute an even larger part of the entire system's power consumption. Besides direct power savings, turning off links may also give indirect savings, *e.g.*, one of the router's blowers may be able to shut down due to less heat. There can also be different ways to reduce power consumptions of line-cards, *e.g.*, using slower line rates for less traffic, but they are out of the scope of this paper.

GreenTE, like any other power saving mechanisms, needs support from the underlying hardware. We make the following assumptions based on today's typical router architectures and hardware in designing GreenTE. However, most of them can be relaxed to take advantage of better hardware support in the future without impacting the basic GreenTE problem formulation and solution.

First, a line-card can have multiple ports, and each port may connect to a link. The multiple ports of one line-card may



(a) traditional (b) power-aware

Fig. 2. Different traffic engineering goals

connect to the same remote router, making it a bundled link, or connect to different remote routers. When a link is put to sleep, the port that connects to the link can go sleep; when all ports on a line-card are asleep, the entire line-card can be put to sleep, resulting in more power saving due to the line-card's base power consumption. A comprehensive analysis of the power consumption of different network components can be found in [6]. Second, a link can be put to sleep only when there is no traffic in *both* inbound and outbound directions, which is based on the fact that the transceivers of inbound and outbound traffic do not have separate power control in most hardware. The GreenTE model can be easily adjusted to allow turning on/off links unidirectionally should hardware supports it, which will bring even more power saving than what we show in this paper. Third, a port or a line-card can go into sleep or wake up quickly, in the order of milliseconds, controlled by its host router [3].

III. GREENTE MODEL

To generalize the basic idea illustrated in Figure 2(b), we develop the GreenTE model, which, given the network topology and traffic matrix, finds a routing solution (*i.e.*, the links to be used and the traffic volume to be carried on each link) that maximizes the power saving from turning off line-cards as well as satisfying performance constraints including link utilization and packet delay.

A. The General Problem Formulation

We model the network as a directed graph $G = (V, E)$, where V is the set of nodes (*i.e.*, routers) and E is the set of links. A port can be put to sleep if there is no traffic on the link, and a line-card can be put to sleep if all its ports are asleep. Let M be the set of line-cards in the network. For a single line-card $m \in M$, its base power consumption is B_m, its set of ports is S_m, and each port $l \in S_m$ consumes power P_l, then the power saving from turning of one port is P_l, and the power saving from turning off the entire line-card is $B_m + \sum_{l \in S_m} P_l$. The objective is to find a routing that maximizes the total power saving in the network. This general power-aware traffic engineering problem can be formulated based on the Multi-Commodity Flow (MCF) model as follows. Please see Table III for the notation used in this paper.

TABLE III
SUMMARY OF NOTATION USED IN THIS PAPER

Notation	Meaning
S_m	Set of links connected to line-card m
P_l	Power consumption of the port connected to link l
B_m	Base power consumption of line-card m
x_l	1 if link l is sleeping, 0 otherwise
y_m	1 if line-card m is sleeping, 0 otherwise
$f_l^{s,t}$	Traffic demand from s to t that is routed through link l
H_l	l's head node
T_l	l's tail node
I_l^v	1 if v is the head node of link l, 0 otherwise
O_l^v	1 if v is the tail node of link l, 0 otherwise
$D_{s,t}$	Traffic demand from s to t
C_l	Capacity of link l
u_l	Utilization of link l
$r(l)$	Reverse link of l
k	Maximum number of candidate paths for each OD pair
U_T	Threshold for the MLU
$Q_i^{s,t}(l)$	1 if the ith candidate path from s to t contains link l, 0 otherwise
$\alpha_i^{s,t}$	Ratio of traffic demand from s to t that is routed through the ith candidate path

Equation 1 computes the objective, the total power saving in the network. Equation 2 states the flow conservation constraints. Let $|S_m|$ be the cardinality of S_m, then equation 3 ensures that a line-card is put to sleep only when all its ports are asleep. Equation 4 calculates the link utilization. Equation 5 ensures that links are put to sleep in pairs, *i.e.*, there is no inbound traffic nor outbound traffic. Equation 6 states that a link can be put to sleep only if there is no traffic on it, and when it is on, it does not carry traffic more than its capacity. Solving this problem gives which links to be turned off, and how much traffic each remaining link should carry.

$$\text{maximize} \quad \sum_{l \in E} P_l x_l + \sum_{m \in M} B_m y_m \tag{1}$$

$$\text{s.t.} \quad \sum_{l \in E} f_l^{s,t} O_l^i - \sum_{l \in E} f_l^{s,t} I_l^i = \begin{cases} D_{s,t}, & i = t \\ -D_{s,t}, & i = s \\ 0, & i \neq s,t \end{cases}, \quad s,t,i \in V, s \neq t \tag{2}$$

$$|S_m| \, y_m \leq \sum_{l \in S_m} x_l \tag{3}$$

$$u_l = \frac{1}{C_l} \sum_{s,t \in V, s \neq t} f_l^{s,t}, \quad l \in E \tag{4}$$

$$x_l = x_{r(l)}, \quad l \in E \tag{5}$$

$$x_l + u_l \leq 1, \quad l \in E \tag{6}$$

The binary (integer) variables x_l and y_m that denote the power state of link l and line-card m make the model a MIP problem. Generally speaking, MIP problems are NP-Hard, thus its computation time for networks with medium and large sizes is a concern. This model, though maximizes power saving in the network, does not consider some practical constraints. For example, packet delay could be much longer than that of current shortest path routing, and links may operate at unacceptably high link utilization, making them vulnerable to any traffic bursts.

B. A Practical Heuristic

To consider the practical constraints and reduce computation time, we refine the problem formulation as follows.

$$\text{maximize} \quad \sum_{l \in E} P_l x_l + \sum_{m \in M} B_m y_m \tag{7}$$

$$\text{s.t.} \quad f_l^{s,t} = \sum_{0 \le i < k} Q_i^{s,t}(l) D_{s,t} \alpha_i^{s,t},$$

$$s,t \in V, l \in E, s \neq t \tag{8}$$

$$\sum_{0 \le i < k} \alpha_i^{s,t} = 1, \quad s,t \in V, s \neq t \tag{9}$$

$$|S_m| \, y_m \le \sum_{l \in S_m} x_l \tag{10}$$

$$u_l = \frac{1}{C_l} \sum_{s,t \in V, s \neq t} f_l^{s,t}, \quad l \in E \tag{11}$$

$$x_l = x_{r(l)}, \quad l \in E \tag{12}$$

$$x_l + u_l \le 1, \quad l \in E \tag{13}$$

$$u_l \le U_T, \quad l \in E \tag{14}$$

One change is the addition of the bound on maximum link utilization in a network. Equation 14 states that MLU must be no greater than a configured threshold U_T. In this paper, we use 50% as the default value of U_T.

Another change is the use of candidate paths instead of searching the solution in all possible paths. The candidate paths are chosen based on the k-shortest paths; therefore each OD pair has at most k candidate paths. Equation 8 and 9 are equivalent to the flow conservation constraints under this change. It reduces overall computation time as well as adds path length as another constraint. The general model introduced in the previous subsection considers all possible paths for each OD pair, making the search space extremely large. To reduce search space and computation time, for each OD pair, we pre-compute its set of candidate paths and only search solutions within this set. Since the k-shortest paths are pre-computed with network topology as the only input, they do not change with the traffic matrix and the computation does not add run-time overhead. Note that when k is set to be large enough, we can actually consider all possible paths for each OD pair, which will give the maximal power saving under the MLU constraint. However, the computation time increases with the value of k; therefore there is a tradeoff between the precision of the heuristic and the computation time. Our evaluation later will show that a reasonably large k can achieve near optimal results.

Searching solutions only within the candidate paths also avoids very long paths. In practice, network operators can have their own definitions of link delays and path lengths, and choose the set of candidate paths accordingly. In this paper we add up link propagation delays to get path lengths, and consider two different constraints in selecting the candidate paths. One is that any candidate path should not be longer than the diameter of the network, *i.e.*, the length of the shortest path between the farthest pair of nodes in the topology. The other is that between any OD pair, a candidate path's length should not be greater than twice that of the shortest path.

Depending on how the candidate paths are chosen, in this paper, we will evaluate three different combinations:

- *basic*: The candidate paths are the k-shortest paths. MLU bound is applied.
- *basic+nd*: The candidate paths are the k-shortest paths which also satisfy the network diameter constraint. MLU bound is applied.
- *basic+e2e*: The candidate paths are the k-shortest paths which also conform to the OD-pair end-to-end delay constraint. MLU bound is applied.

With these changes, the GreenTE model now has practical constraints on link utilization and path length, and also can be solved within reasonable time.

C. Load Balancing

In conventional traffic engineering, load balancing is the main objective, usually formulated as minimizing the MLU in a network. Though it may not be a good idea to combine it with power-aware traffic engineering in the same problem formulation, we can still do load balancing on top of the routing resulted from power-aware traffic engineering.

From solving the problem formulated in the previous subsection we can obtain the set of links to be put to sleep. Excluding paths containing these links from the original set of candidate paths, we get a new set of paths Q', onto which the traffic load will be balanced by solving the following problem:

$$\text{minimize} \quad \max_{l \in E} u_l \tag{15}$$

$$\text{s.t.} \quad f_l^{s,t} = \sum_{0 \le i < k} Q_i'^{s,t}(l) D_{s,t} \alpha_i^{s,t},$$

$$s,t \in V, l \in E, s \neq t \tag{16}$$

$$\sum_{0 \le i < k} \alpha_i^{s,t} = 1, \quad s,t \in V, s \neq t \tag{17}$$

$$u_l = \frac{1}{C_l} \sum_{s,t \in V, s \neq t} f_l^{s,t}, \quad l \in E \tag{18}$$

The above formulation minimizes the MLU over links that remain on, and solving it gives the traffic load that each such link should carry. We use *basic+lb*, *basic+nd+lb* and *basic+e2e+lb* to denote the models under different performance constraints after performing load balancing.

In summary, the GreenTE model maximizes power saving, considers constraints on link utilization and path length, and also balances load over the links.

IV. IMPLEMENTATION ISSUES

Realizing GreenTE in operational networks requires coordination among all routers in the network. In this section we outline such coordination and discuss its different aspects. Our basic principle in GreenTE design is to use existing protocols and mechanisms as much as possible for the benefits of compatibility and deployability. We also assume that networks run both OSPF (or any link state routing protocol) and MPLS.

A. Overview

As in conventional traffic engineering, GreenTE relies on a logically centralized controller in the Network Operation Center (NOC) to make decisions on traffic engineering. The physical implementation of such a controller can have hardware redundancy and/or replica in different locations for better reliability. The controller collects input information (*i.e.*, network topology and traffic matrix) from routers, solves the GreenTE problem to get new routing configurations (*i.e.*, which links are up and how much traffic on each link), and disseminates the results to routers. Each router will then turn on/off some line-cards or ports according to the GreenTE solution and set up MPLS tunnels for data forwarding if needed.

As traffic demand changes over time and sometimes unpredictably, the process described above needs to be done periodically. The frequency of such routing adjustment depends on how often the traffic matrix changes and by how much. Adjusting routing too often will result in more control overhead and more disturbance to data forwarding (*e.g.*, packet loss and re-order), but too few will leave the routing non-optimized for too long as the traffic matrix may have changed significantly. In our experiments, we adjust the routing every 5-15 minutes. However, our result shows that route selection by GreenTE is relatively stable, *i.e.*, most MPLS tunnels remain unchanged from one routing configuration to the next, which means that the negative impacts of routing adjustment is rather limited each time.

B. Gathering Input Information for GreenTE

The controller collects network topology and traffic matrix from OSPF's Link State Advertisements (LSAs). In OSPF, each router floods its LSAs whenever its link state changes. Thus the controller can readily collect all the link state information and compile the up-to-date network topology.

Directly measuring traffic matrix in real-time is still expensive in large networks. Instead, the GreenTE controller collects link load information from routers and computes the traffic matrix locally. The link load information is part of the Traffic Engineering Link State Advertisement (TE-LSA) defined in RFC3630 [7]. As an extension to the basic OSPF LSA, TE-LSA is also flooded in the network. TE-LSA reports a link's maximum bandwidth and unreserved bandwidth, and the difference between them is the link load. A router sends out TE-LSA when there is a significant change in its bandwidth usage. Once the link load information is collected, the controller computes the network traffic matrix using the tomogravity method, which is lightweight, accurate, and can be done within a few seconds for large ISP networks [8].

Both the network topology and link load information are collected by the controller passively. The controller does not poll any specific router, nor has any explicit point-to-point conversation with any individual router. All information is announced via LSAs. This design choice is compatible with existing mechanisms, simplifies operations, and also inherits the delivery reliability provided by LSA flooding.

C. Distributing GreenTE Results

With the network topology and traffic matrix, the controller solves the GreenTE problem to get which links to be turned on or off, and distributes this information to routers via the Traffic Engineering Metric (TE-Metric) attribute, another extension to OSPF defined in RFC 3630 [7]. The GreenTE convention is that if a link's TE-Metric is set to be equal to its OSPF weight, the link should to be turned on; if a link's TE-Metric is set to be the maximum value allowed, it should be turned off. Note that both TE-LSA and TE-Metric messages are flooded in the network as regular OSPF LSAs; therefore they can reach all routers as long as the network is connected and require no separate interfaces or links to be reserved. The flow conservation constraints in GreenTE formulation guarantee that the solution does not partition the network.

To minimize packet loss during routing transitions, extra care is needed when routers are turning on/off links. When a router has a link to turn off, it should not do so immediately, because otherwise some on-the-fly packets may be lost. Ideally it should wait for all the alternative paths have been set up before actually turning off a link. In practice a router may turn off a link after the link has been idle for more than a certain period of time. The network diameter can serve as a rough threshold for this purpose. When a router has a link to wake up, it should turn on the link immediately but not transmit data onto this link until both ends of the link are ready. Two routers can exchange messages to confirm that they are ready. Such messages can be MPLS signaling messages, OSPF Hello messages, or simple messages designed specifically for this purpose. An alternative is to simply use a timer.

D. Data Forwarding under GreenTE

In GreenTE, data packets are forwarded along either OSPF paths or MPLS tunnels (*i.e.*, Label Switching Path, LSP). Solving the GreenTE model gives the paths that data traffic should take. If a GreenTE path happens to be the shortest path according to OSPF, the traffic is simply transmitted as native IP packets; otherwise an LSP is set up, by either the Constraint-based Routing Label Distribution Protocol (CR-LDP) [9] or Resource Reservation Protocol-Traffic Engineering (RSVP-TE) [10], to implement the non-shortest path to carry traffic.

In the case that the traffic between an OD pair takes multiple paths in GreenTE solution, the traffic split ratio among the multiple paths will also be part of the solution. Such traffic split is usually supported in today's commercial routers by a hash-based mechanism [11]. Basically at the ingress routers, regardless of whether its from MPLS or OSPF, one or multiple next-hop interfaces will be associated with a destination prefix in the forwarding table. When there are multiple next-hop interfaces, a hashing mechanism is employed to determine which flow of traffic will take which next-hop, so that traffic of the same flow will always take the same path. The hashing mechanism can be configured with different weights to realize different traffic split ratios.

The OSPF/MPLS hybrid approach has been shown in previous work (*e.g.*, [12]) to have two main advantages. First, the number of MPLS tunnels is much less than what would be required in a full-mesh configuration because the majority of the traffic actually takes the shortest path. Second, it causes much fewer OSPF convergence than pure-OSPF traffic engineering because each time routing adjustment is achieved by changing a few MPLS tunnels and the traffic split ratio instead of changing OSPF link weights. These advantages are confirmed in our evaluations.

E. Impacts on Other Protocols

In conventional networking, a link has two states, either up or down. An up link is able to transmit packets while a down link cannot. With power-aware networking, a link has a third state: sleeping. A sleeping link is not used to transmit packets for the moment but can do so if needed. Introducing sleeping links has no or minimal impacts on end-to-end protocols such as TCP and UDP since GreenTE has set up alternative paths for data delivery, but it may affect the operation of protocols that depend on link-level information. A typical example is OSPF, which uses periodic HELLO messages to detect the existence of a link and its state. Simply putting a link into sleep will make OSPF believe that the link is down, which will trigger LSAs and network-wide OSPF convergence process. Generally speaking there are three different approaches that a protocol can use:

- Explicitly handling sleeping links. For example, in GreenTE, the information of sleeping links is flooded via TE-Metric attribute in LSAs. A router, after missing a few HELLO messages from a link, can check whether this link is supposed to be sleeping, and if yes, it can label this link as sleeping and handle it differently from down links. This is the cleanest way to support power-aware networking, but it requires changes to protocol specification or implementation.
- Adjusting protocol parameters. For example, one can use a larger interval for OSPF HELLO messages to avoid a sleeping link from being detected as a down link. This does not require any changes to the protocol, but may discover actual link failures much later than it would with original parameters.
- Waking up the link. The link is awaken on-demand whenever there is a packet for it. This is fully compatible with current networking but results in less power saving.

Exactly which approach to take is a tradeoff between backward compatibility and network energy efficiency, and may evolve over time as we see different solutions take place.

The impact of GreenTE on OSPF convergence time is limited. Most of the time GreenTE does not change OSPF link weights or its routing paths; it only adjusts the traffic split ratio and/or MPLS tunnels. Thus GreenTE usually does not trigger OSPF convergence. When OSPF convergence does happen, one factor in its convergence time is how quick the LSA is flooded to reach all routers. Sleeping links may make this time longer since there are less links for flooding. However, as we have delay bound built in GreenTE, such propagation delay of routing messages should not be much longer than in non-GreenTE networks. For example, under the network diameter constraint, GreenTE network maintains the same network diameter, thus it should take about the same time for a routing message to reach all routers in the network.

Another issue with sleeping links is network robustness to link failures and traffic bursts. When a link fails or a burst of traffic arrives, the network needs to find alternative paths to accommodate the traffic if the traffic being affected is of very high volume. This problem exists in power-aware networking as well as in any IP networks, although in the former the problem can be more severe as the sleeping links are not readily available. In this case some sleeping links need to wake up on-demand in order to handle the extra traffic demand. One of our future work is to further investigate this issue.

V. EVALUATION

In this section, we evaluate GreenTE and show that it is able to achieve considerable power savings in real networks with minor impact on network performance.

A. Experiment Setup

TABLE IV
NETWORK TOPOLOGIES USED IN THE EVALUATION

Network	Usage	Location	Nodes	Links
Abilene	Research	US	12	30
GÉANT	Research	Europe	23	74
Sprint	Commercial	US	52	168
AT&T	Commercial	US	115	296

We use different network topologies in the evaluation, including Abilene, GÉANT and selected topologies from Rocketfuel [13] as listed in Table IV. These topologies vary in size and usage. For Abilene, the router-level topology (*i.e.*, link connectivity, weights, lengths and capacities) and measured traffic matrices are available at [14]. The non-anonymized topology and traffic matrices of GÉANT are provided by the authors of [15]. The traffic matrices are measured every 5 minutes for Abilene and every 15 minutes for GÉANT.

While Abilene and GÉANT are both research networks, Rocketfuel provides PoP-level topologies of commercial ISPs. We assume that each node in the topology corresponds to a router. Since link capacities and traffic matrices are not available for the Rocketfuel topologies, we assign capacities to links using the method described in [16] and generate traffic matrices using the gravity model [17] [18].

Given the above information, we are able to pre-compute the candidate paths for each OD pair and solve the GreenTE model using CPLEX [19]. From the solution of the model, we can obtain the power saving for the network as well as the utilization for each link. In addition, the solution also gives which paths to use for each OD pair and how to split traffic among these paths.

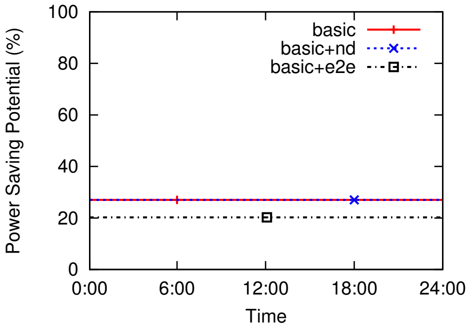


Fig. 3. Power saving potential of Abilene on Sep. 5th, 2004

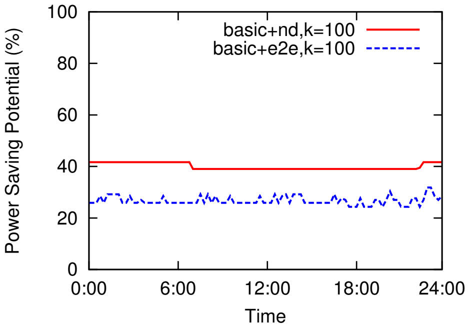


Fig. 4. Power saving potential of GÉANT on May 5th, 2005

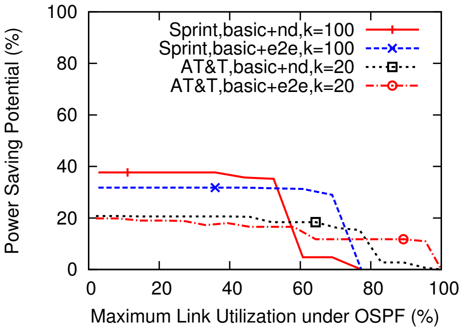


Fig. 5. Power saving potential of Sprint and AT&T

In the evaluation, we assume that each line-card is connected to a single link; therefore a line-card can be put to sleep when there is no traffic on the link. We make this assumption because the information of physical connections among line-cards is not available in the data set. However, the GreenTE model can be applied to the situation when line-cards have multiple ports.

We reconfigure the network every 5-15 minutes based on the availability of traffic matrices. We assume that traffic matrix does not change significantly within 5-15 minutes, and this is confirmed by real traffic data as follows. We analyze the traffic matrices of Abilene from one typical day (Sep. 4th, 2004), and evaluate how traffic between each OD pair changes over time. Results show that for more than 74% of the OD pairs, the change of traffic volume within 5 minutes is less than 30%.

To evaluate the impact of GreenTE on queuing delay, we implement GreenTE in *ns2* [20] for packet-level simulations. We set up OSPF paths as well as MPLS tunnels, assign traffic split ratio to the paths, and generate traffic based on measured traffic matrices. Specifically, we generate self-similar traffic for each OD pair using a mix of Pareto flows to simulate real Internet traffic [21]. All the experiments are conducted on machines with 8 GB of RAM and a Quad-Core Intel Q9650 processor (3.0 GHz).

B. *Power Savings*

TABLE V
POWER CONSUMPTIONS OF LINE-CARDS [5]

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

We explore the power saving potential under GreenTE using different network topologies and traffic matrices. We compute the power saving ratio as the total power of sleeping line-cards over the total power of all line-cards in the network. As noted in Section II, line-cards all together account for more than 40% of a router's total power budget; therefore it is meaningful to measure the power saving ratio of line-cards. The power consumption of line-cards we use in the evaluation is specified in Table V.

1) Abilene: Figure 3 shows the power saving potential of Abilene on Sep. 5th, 2004 under different performance constraints. The power saving ratio under basic OSPF is not shown here because it is always zero. We still use 50% as the MLU threshold. In this experiment, we set k to be large enough so that all paths that satisfy the delay constraints are included; therefore the results shown in the figure are actually optimal.

GreenTE is able to achieve about 27% power savings under *basic* and *basic+nd*. The two curves overlap because *basic+nd* includes sufficient candidate paths to achieve the maximal power saving. The power saving ratio under *basic+e2e* is about 20%, lower than the other two because less number of candidate paths are considered. The power saving ratio does not change over time because the traffic volume is relatively small throughout the day so that GreenTE is always able to put the maximum number of links to sleep while conforming to the constraints.

2) GÉANT: Figure 4 presents the power saving potential of GÉANT on May 5th, 2005. We set $k = 100$ in this experiment to include most of the candidate paths that conform to the delay constraints. We show how the value of k affects the power saving potential in Section V-G. The result for *basic* is not shown here because the problem cannot be solved within reasonable time, and we expect *basic+nd* and *basic+e2e* to be more commonly used in practice. Similar to Abilene, GreenTE achieves more power savings under *basic+nd* because it is less restrictive than *basic+e2e*; but the power saving ratio under *basic+e2e* is still considerable (more than 20%).

3) Rocketfuel Topologies: For large topologies such as Sprint or AT&T, CPLEX is unable to get the optimal solution within reasonable time. To resolve this problem, we force CPLEX to terminate after 300 seconds. The results, though not optimal, are guaranteed to be correct as they satisfy all the constraints. We show in Section V-G that this method actually yields satisfactory results within acceptable time limit.

Figure 5 presents the power saving potential of Sprint and AT&T under various traffic conditions. We generate one traffic matrix for each network using the gravity model, which is then scaled to obtain different traffic loads. The figure shows that the power saving ratio decreases as the traffic load increases; but in a normally operated network where the MLU is below 40%, GreenTE is able to achieve stable power savings. Sprint

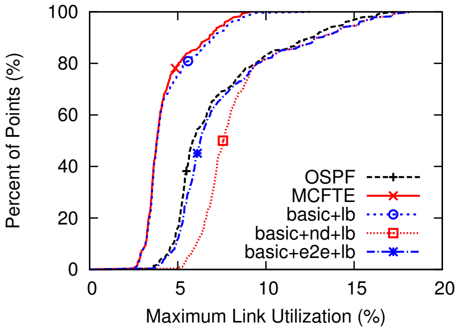


Fig. 6. CDF of MLU of Abilene on Sep. 5th, 2004

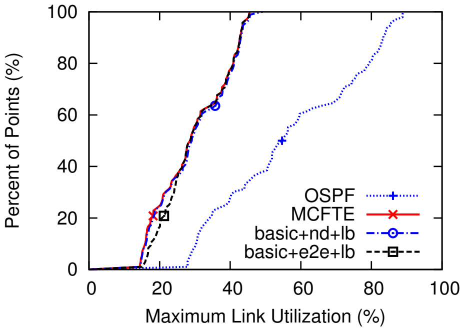


Fig. 7. CDF of MLU of GÉANT on May 5th, 2005 (k = 100)

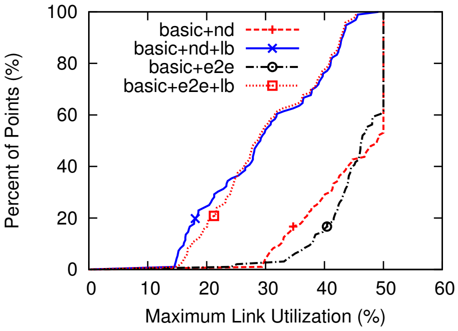


Fig. 8. CDF of MLU of GÉANT on May 5th, 2005 (k = 100), before and after load balancing

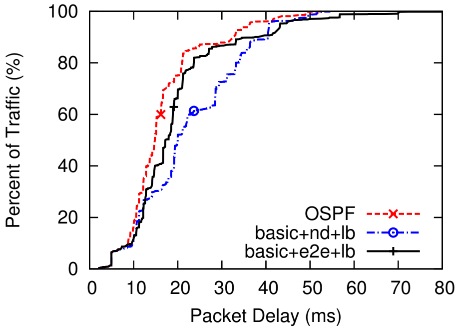


Fig. 9. CDF of Packet Delay for GÉANT (k = 100)

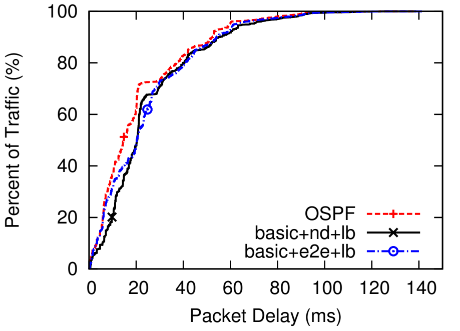


Fig. 10. CDF of Packet Delay for Sprint (k = 100)

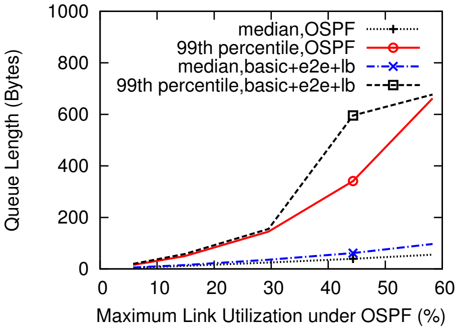


Fig. 11. Queue length for Abilene under different traffic conditions

exhibits higher power saving potential because it has relatively higher link redundancy than AT&T (3.23 *vs.* 2.57 links per node).

C. Link Utilization

Intuitively, GreenTE would affect the utilization of links as fewer links are used to carry traffic. In this subsection, we evaluate the impact of GreenTE on link utilization. Specifically, we show how the maximum link utilization of the network is affected by different routing mechanisms.

1) Abilene: We draw the CDF of MLU of Abilene throughout one day in Figure 6. Since the traffic load is light, the MLU is always under 20% for all routing mechanisms. MCFTE achieves optimal load balancing and thus acts as the lower bound for the MLU [12] [22]. *basic+lb* is very close to MCFTE because it has sufficient candidate paths for load balancing. The figure also shows that *basic+nd+lb* and *basic+e2e+lb* are able to achieve similar MLU as OSPF when the network is lightly loaded.

2) GÉANT: The CDF of MLU of GÉANT throughout one day is shown in Figure 7. The MLU as high as about 90% under OSPF is caused by a single hot-spot link, and GreenTE is able to shift traffic away from that link to avoid congestion. GreenTE achieves similar MLU as MCFTE while obtaining considerable power savings.

Figure 8 shows the comparison of MLU before and after the load balancing optimization. Before load balancing is performed, more than 40% of links have utilization of 50%. This is because the solver only focuses on putting links to sleep as long as the MLU is no greater than 50%. The load balancing optimization effectively reduces the MLU of the network.

D. Delay

Since part of the traffic is routed through non-shortest paths, GreenTE may also increase the packet delay. In this subsection, we evaluate propagation delay which dominates packet delay when the network is not congested. Queuing delay will be considered in the next subsection.

Figure 9 and 10 show the CDF of packet delay for GÉANT and Sprint. The results for other topologies are similar and thus not shown here. For GÉANT, we choose a traffic matrix whose MLU under OSPF is about 50%; for Sprint we scale the generated traffic matrix so that the MLU under OSPF is about 50%. Since link weight reflect link length in all the experiments, OSPF actually gives a lower bound for packet delay. The figures show that GreenTE is able to effectively bound the packet delay within a desired level.

For GÉANT, *basic+e2e+lb* is closer to OSPF than *basic+nd+lb* for most of the traffic, while *basic+nd+lb* successfully bounds the worst case to be no greater than the longest delay in OSPF. For Sprint, the three curves are closer to each other because the topology is larger; hence there are more paths for each OD pair that have delay close to the shortest path.

E. Queue Length

As GreenTE uses fewer number of links to carry the same amount of traffic, queuing delay experienced by packets is also

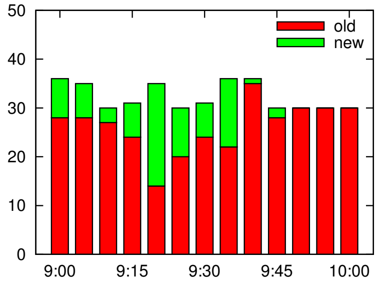


Fig. 12. Number of MPLS tunnels for Abilene under *basic+nd+lb*

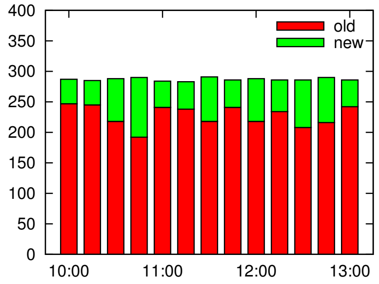


Fig. 13. Number of MPLS tunnels for GÉANT under *basic+nd+lb*

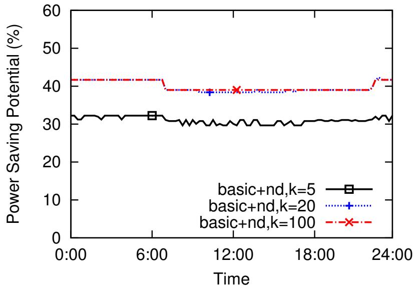


Fig. 14. Power saving potential of GÉANT with different k values

likely to be affected. We use ns2 to evaluate the queuing delay under OSPF and GreenTE. We choose the same traffic matrix from Abilene as in Section V-D and scale it to produce different traffic loads. We run each experiment for 5 minutes, and collect the average queue length of each link per second. Figure 11 shows the median and 99th percentile of queue lengths for Abilene under different traffic conditions.

The queue lengths under GreenTE and OSPF are very close when the MLU under OSPF is lower than 30%. As the network becomes more heavily loaded, the queue length under GreenTE becomes obviously larger than that under OSPF. However, since the absolute values of queue lengths are actually very small, the impact of GreenTE on queuing delay is minor.

F. Routing Stability

Transition from one route configuration to another may cause problems such as packet loss and re-ordering. In this subsection, we show that route selection by GreenTE is relatively stable so that negative impacts caused by routing adjustments can be limited.

We choose traffic matrices from peak hours during the day (9:00-10:00 for Abilene and 10:00-13:00 for GÉANT). Figure 12 and 13 show that on average more than 70% of MPLS tunnels stay unchanged during routing transitions under *basic+nd+lb*. This is because traffic matrix does not change significantly between two contiguous routing configurations. The figures also show that the number of MPLS tunnels needed in the network is much less than that in a full mesh. The same conclusion also applies to other topologies and constraints.

G. Precision of Heuristics

TABLE VI
POWER SAVING POTENTIAL OF AT&T UNDER *basic+e2e* WITH 21% MLU UNDER OSPF

k value	Computation Time	Status	Power Saving Potential
5	65s	Optimal	11.90%
10	5747s	Optimal	17.54%
20	100892s	Optimal	19.79%
20	300s	Non-optimal	18.99%

Figure 14 shows that the power saving potential grows as the value of k increases. However, increasing k also increases the computation time. When k is large enough (20 in this example), increasing k only improves the power saving potential by a negligible amount. Therefore, GreenTE is able to achieve near optimal power savings as long as k is reasonably large.

As the computation time is too long for large topologies such as Sprint and AT&T, we force CPLEX to stop after 300 seconds. Table VI shows that computation time increases dramatically as the value of k grows. However, when $k = 20$, we can obtain about 96% of the optimal power saving if we limit the computation time to be 300 seconds.

VI. RELATED WORK

Gupta *et al.* identify the power saving problem in the Internet, and propose sleeping as the approach to conserve energy [2]. Specifically, they suggest two options - *uncoordinated sleeping* which works at link level and *coordinated sleeping* which operates at network level. In their follow-up works [23] [24] [25], the authors study *uncoordinated sleeping* in Local Area Networks (LANs). This approach works effectively in LANs because of its specific traffic patterns; however, it might not be applicable to backbone networks where inter-packet time is too short for the links to sleep.

In [3], Nedevschi *et al.* propose the *buffer-and-burst* approach which shapes traffic into small bursts to create greater opportunities for network components to sleep. The same work also brings up the idea of *rate-adaptation*, which adjusts operating rates of links according to the traffic condition. This work is also focused on link level solutions.

Chabarek *et al.* explore power-awareness in the design of networks and routing protocols in wire-line networks [26]. The authors reveal the significant power saving potential in operational networks by including power-awareness, but they do not come up with a specific power-aware routing design. In this paper, we propose GreenTE to achieve power-aware routing through traffic engineering.

Heller *et al.* propose ElasticTree [27], which optimizes the energy consumption of Data Center Networks by turning off unnecessary links and switches during off-peak hours. ElasticTree also models the problem based on the MCF model, but is focused on Fat-Tree or similar tree-based topologies. ElasticTree takes link utilization and redundancy into consideration when calculating the minimum-power network subset, and is implemented using OpenFlow.

Vasić *et al.* propose EATe [28] to reduce Internet power consumption through traffic engineering. EATe considers sleeping of links and routers as well as link rate adaption. EATe achieves its routing decisions in a distributed fashion via router coordination and thus requires routers to be able to send announcement and feedback to each other. In contrast, GreenTE is mostly compatible with current operation practice.

Internet traffic engineering is a widely studied topic. Fortz and Thorup first propose the idea of IGP link weight optimization for the purpose of traffic engineering [29] [30]. However, frequent changes to link weights would cause problems such as network-wide routing convergence and traffic shift.

MATE [31] and TeXCP [16] perform traffic engineering by splitting traffic among multiple MPLS paths. MPLS-based traffic engineering can achieve optimal routing, but does not scale well as the size of network grows. In [12], Zhang *et al.* propose MCFTE, which performs traffic engineering through hybrid OSPF/MPLS routing. MCFTE achieves optimal routing with only a small number of MPLS tunnels, and thus alleviates the scalability problem. Other works on hybrid routing include [32] [33] [34].

VII. CONCLUSION

High path redundancy and low link utilization in today's large networks provide unique opportunities for power-aware traffic engineering. By switching traffic onto fewer number of paths, one can free some links from carrying data traffic and put them to sleep for energy conservation. The GreenTE model maximizes the number of links that can be put to sleep under the constraints of link utilization and path length, and also balances the network load afterwards. Evaluations based on real network topologies and traffic matrices show that GreenTE is able to achieve considerable power savings with minor impacts on the network performance. Our future work includes investigating the implication of multiple ports per line-card on power savings, handling link failures and sudden traffic bursts, taking advantages of recent advances in hardware energy management mechanisms.

REFERENCES

[1] G. Epps, D. Tsiang, and T. Boures, "System Power Challenges," 2006. [Online]. Available: http://www.slidefinder.net/c/cisco_routing_research/seminar_august_29/1562106

[2] M. Gupta and S. Singh, "Greening of the Internet," in *Proceedings of ACM SIGCOMM*, 2003.

[3] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall, "Reducing Network Energy Consumption via Sleeping and Rate-Adaptation," in *Proceedings of USENIX NSDI*, 2008.

[4] J. Guichard, F. L. Faucheur, and J.-P. Vasseur, *Definitive MPLS Network Designs*. Cisco Press, 2005.

[5] "Power Management for the Cisco 12000 Series Router." [Online]. Available: http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/12spower.html

[6] P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan, "A Power Benchmarking Framework for Network Devices," in *Proceedings of the 8th International IFIP-TC 6 Networking Conference*, 2009.

[7] D. Katz, K. Kompella, and D. Yeung, "RFC 3630: Traffic Engineering (TE) Extensions to OSPF Version 2," Sept 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3630.txt

[8] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast Accurate Computation of Large-Scale IP Traffic Matrices from Link Loads," in *Proceedings of ACM SIGMETRICS*, 2003.

[9] B. Jamoussi *et al.* , "RFC 3212: Constraint-based LSP Setup Using LDP," Jan 2002. [Online]. Available: http://www.ietf.org/rfc/rfc3212.txt

[10] D. Awduche *et al.* , "RFC 3209: RSVP-TE: Extensions to RSVP for LSP tunnels," Dec 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3209.txt

[11] "Using Hash-Key Load Balancing for LSP Traffic." [Online]. Available: https://www.juniper.net/techpubs/en_US/junos10.0/information-products/topic-collections/nog-mpls-frr/mpls-load-balancing-hash-key.html

[12] M. Zhang, B. Liu, and B. Zhang, "Multi-Commodity Flow Traffic Engineering with Hybrid MPLS/OSPF Routing," in *Proceedings of IEEE GLOBECOM*, 2009.

[13] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, "Measuring ISP Topologies with Rocketfuel," *IEEE/ACM Transactions on Networking*, vol. 12, no. 1, pp. 2–16, February 2004.

[14] "Yin Zhang's Abilene TM." [Online]. Available: http://www.cs.utexas.edu/~yzhang/research/AbileneTM/

[15] S. Uhlig, B. Quoitin, J. Lepropre, and S. Balon, "Providing Public Intradomain Traffic Matrices to the Research Community," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 83–86, January 2006.

[16] S. Kandula, D. Katabi, B. Davie, and A. Charny, "Walking the Tightrope: Responsive Yet Stable Traffic Engineering," in *Proceedings of ACM SIGCOMM*, 2005.

[17] M. Roughan, A. Greenberg, C. Kalmanek, M. Rumsewicz, J. Yates, and Y. Zhang, "Experience in Measuring Backbone Traffic Variability: Models, Metrics, Measurements and Meaning," in *in ACM SIGCOMM Internet Measurement Workshop*, 2002.

[18] D. Applegate and E. Cohen, "Making Intra-domain Routing Robust to Changing and Uncertain Traffic Demands: Understanding Fundamental Tradeoffs," in *Proceedings of ACM SIGCOMM*, 2003.

[19] "CPLEX." [Online]. Available: http://www.ilog.com/products/cplex/

[20] "Network Simulator ns-2." [Online]. Available: http://www.isi.edu/nsnam/ns/

[21] M. Yuksel, B. Sikdar, K. S. Vastola, and B. Szymanski, "Workload Generation for NS Simulations of Wide Area Networks and the Internet," in *Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference*, 2000.

[22] M. Roughan, M. Thorup, and Y. Zhang, "Traffic Engineering with Estimated Traffic Matrices," in *Proceedings of ACM IMC*, 2003.

[23] M. Gupta, S. Grover, and S. Singh, "A Feasibility Study for Power Management in LAN Switches," in *Proceedings of IEEE ICNP*, 2004.

[24] M. Gupta and S. Singh, "Dynamic Ethernet Link Shutdown for Energy Conservation on Ethernet Links," in *Proceedings of IEEE ICC*, 2007.

[25] ——, "Using Low-Power Modes for Energy Conservation in Ethernet LANs," in *Proceedings of IEEE INFOCOM*, 2007.

[26] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright, "Power Awareness in Network Design and Routing," in *Proceedings of IEEE INFOCOM*, 2008.

[27] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. McKeown, "ElasticTree: Saving Energy in Data Center Networks," in *Proceedings of USENIX NSDI*, 2010.

[28] N. Vasić and D. Kostić, "Energy-Aware Traffic Engineering," in *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking*, 2010.

[29] B. Fortz and M. Thorup, "Internet Traffic Engineering by Optimizing OSPF Weights," in *Proceedings of IEEE INFOCOM*, 2000.

[30] ——, "Optimizing OSPF/IS-IS Weights in a Changing World," *IEEE Journal On Selected Areas In Communications*, vol. 20, no. 4, pp. 756–767, May 2002.

[31] A. Elwalid, C. Jin, S. Low, and I. Widjaja, "MATE: MPLS Adaptive Traffic Engineering," in *Proceedings of IEEE INFOCOM*, 2001.

[32] E. Mulyana and U. Killat, "An Offline Hybrid IGP/MPLS Traffic Engineering Approach under LSP Constraints," in *Proceedings of the 1st International Network Optimization Conference, INOC*, 2003.

[33] H. Pham and B. Lavery, "Hybrid Routing for Scalable IP/MPLS Traffic Engineering," in *Proceedings of IEEE ICC*, 2003.

[34] F. Skivée, S. Balon, and G. Leduc, "A Scalable Heuristic for Hybrid IGP/MPLS Traffic Engineering - Case Study on an Operational Network," in *Proceedings of IEEE ICON*, 2006.