1-1-2012

# Power-aware routing in networks with delay and link utilization constraints

Gongqi Lin
*Curtin University of Technology*

Sieteng Soh
*Curtin University of Technology*, s.soh@curtin.edu.au

Mihai Lazarescu
*Curtin University of Technology*

Kwan-Wu Chin
*University of Wollongong*, kwanwu@uow.edu.au

## Recommended Citation

Lin, Gongqi; Soh, Sieteng; Lazarescu, Mihai; and Chin, Kwan-Wu, "Power-aware routing in networks with delay and link utilization constraints" (2012). *Faculty of Engineering and Information Sciences - Papers: Part A*. 241.
https://ro.uow.edu.au/eispapers/241

# Power-aware routing in networks with delay and link utilization constraints

## Abstract

Current network infrastructures are over-provisioned and thus exhibit poor power efficiency at low traffic load. In this paper, we consider networks comprising of bundled links, whereby each link has one or more physical cables that can be switched off independently. The problem at hand is then to switch off redundant cables during off peak periods, while retaining the QoS provided to existing traffic demands. Unfortunately, the problem to maximally shutdown redundant cables is an NP-complete problem. Henceforth, we design a fast heuristic, called Multiple Paths by Shortest Path First (MSPF), that aims to maximize the number of switched-off cables subject to satisfying maximum link utilization (MLU) and end-to-end delay requirements. We have extensively evaluated the performance of MSPF on both real and synthetic topologies and traffic demands. Further, we have compared its performance against two state-of-the-art techniques: GreenTE usable only when each link has one cable, and FGH that supports bundled links but usable only for networks without MLU and delay constraints. MSPF improves the energy saving on average by 5% as compared to GreenTE incurring only 1% the CPU time. While yielding equivalent energy savings, MSPF requires only 0.35% of the running time of FGH. Finally, for MLU at most 50% and end-to-end delay no longer than the network diameter, MSPF reduces the power usage of the GÉANT topology up to 91% and bundled links consisting of ten cables.

## Keywords

networks, delay, utilization, routing, power, aware, constraints, link

## Disciplines

Engineering | Science and Technology Studies

## Publication Details

G. Lin, S. Soh, M. Lazarescu & K. Chin, "Power-aware routing in networks with delay and link utilization constraints," in IEEE 37th Conference on Local Computer Networks (LCN), 2012, pp. 272-275.

# Power-Aware Routing in Networks with Delay and Link Utilization Constraints

Gongqi Lin, Sieteng Soh, Mihai Lazarescu
Department of Computing
Curtin University of Technology
Perth Australia
Gongqi.lin@postgrad.curtin.edu.au

Kwan-Wu Chin
School of Electrical, Computer and Telecommunications Engineering
University of Wollongong
Wollongong Australia
kwanwu@uow.edu.au

*Abstract*—**This paper addresses the NP-hard problem of switching off bundled links whilst retaining the QoS provided to existing applications. We propose a fast heuristic, called Multiple Paths by Shortest Path First (MSPF), and evaluated its performance against two state-of-the-art techniques: GreenTE, and FGH. MSPF improves the energy saving on average by 5% as compared to GreenTE with only 1% CPU time. While yielding equivalent energy savings, MSPF requires only 0.35% of the running time of FGH. Finally, for Maximum Link Utilization (MLU) below 50% and delay no longer than the network diameter, MSPF reduces the power usage of the GÉANT topology by up to 91%.**

***Keywords - power savings; routing; multiple paths; maximum link utilization; bundled links; shortest path***

## I. INTRODUCTION

Current backbone networks are over-provisioned to accommodate traffic bursts, and route/link failures. However, they consume unnecessary amount of energy during off-peak periods since the power consumption of routers and their line cards is independent of link load [3]. To this end, Chiaraviglio *et al.* [4] proposed a solution to the problem of finding the minimum set of routers and links that can accommodate a given traffic demand. However, their solution does not consider the effects of traffic delays and maximum link utilization *after* these routers are switched off, which may reduce a network's fault tolerance capability. Vassos *et al.* [15] studied the impact of power in interconnection networks, and explored the design space for shutting down links. However, they did not consider the impact of delay and link utilization on network performance. Other works such as Zhang *et al.* [5] proposed a traffic engineering technique, called GreenTE, to reduce energy expenditure by turning off unused links while considering maximum link utilization (MLU) and delay constraints. This optimization problem is known to be NP-complete. GreenTE is effective in selecting unused links, but its computation is slow for large networks. Fisher *et al.* [6] observed that each network link may comprise of two to twenty cables [14]. They then propose three algorithms, e.g., FGH, to turn-off unused cables. While FGH is effective in reducing energy, it does not guarantee both MLU and delay constraints. Further, like GreenTE [5], its running time is prohibitive on large networks.

Our contributions in this paper are twofold. First, we propose an optimization problem to maximally turn-off unnecessary cables in a network with bundled links while meeting two performance constraints: MLU and traffic delay. Each link $e_{ij}$ comprises of $w_{ij}\geq 1$ cables that can be turned off independently and the delay can be either the network's diameter or λ times the delay of its original shortest path, for a given delay multiplier $1.0\leq\lambda\leq 2.0$. The NP-complete problem generalizes those in [5] and [6]; *i.e.*, for $w_{ij}=1$ and $\lambda=2.0$, it reduces to that in [5], and it becomes that in [6] if we ignore the two constraints. Second, we design a heuristic, called Multiple Paths by Shortest Path First (MSPF) that solves the problem more efficiently and as effective, if not more, than the solutions in [5] and [6]. MSPF runs on average 99% faster than GreenTE [5] while improving its energy savings by 5%. Further, MSPF uses only 0.35% of the running time of FGH [6], while yielding equivalent energy savings.

## II. PROBLEM FORMULATION

### A. Network Model

Consider a network modeled by a directed graph $G(V, E)$, where $V(E)$ is the set of $n$ nodes ($m$ links). Each node represents a router and each link $e_{ij}$ between nodes $v_i$ and $v_j$ represents a bundled link as a communication channel with capacity $c_{ij}>0$. Each link $e_{ij}$ consists of $w_{ij}\geq 1$ cables, *i.e.*, each link $e_{ij}$ corresponds to $w_{ij}$ copies of cable $b_{ij}$. Our model generalizes that of [5] which assumes equal bundle size $w_{ij}$. Each $b_{ij}$ that can be turned-off independently has the same bandwidth and consumes the same energy $p_{ij}$. Let $n_{ij}\leq w_{ij}$ be an integer that represents the total number of powered-on cables in $e_{ij}$. Let $D=\{D_\alpha=(s, t, f^\alpha) \mid$ a demand $D_\alpha$ from a source node $s=1,\ldots,n$ to a terminal node $t=1,\ldots, n$ that has traffic flow $f^\alpha$ $\}$, and $CP_\alpha=\{cp_{\alpha,q} \mid$ a candidate path $q=1, 2, \ldots, |CP_\alpha|$ that can be used to route demand $D_\alpha$ with delay no more than $d_{T,\alpha}$ $\}$. The variable $\delta_{ij}^{\alpha,q}$ has a value of 1 if the $q^{\text{th}}$ candidate path for $D_\alpha$ contains link $e_{ij}\in E$, and it is 0 otherwise. Let $d_\alpha$ be the length of the shortest path. When a traffic demand is routed through multiple $(s,t)$ paths, we set its traffic delay to the maximum hop count among the paths. We denote $f_{ij}^{st}$ or $f_{ij}^{\alpha}$ as the flow on link

$e_{ij}$ of demand $D_\alpha$, and $f_{ij}$ the total flow on $e_{ij}$. Lastly, the remaining/spare capacity on link $e_{ij}$ is $r_{ij} = c_{ij} - f_{ij}$.

## *B. Problem Statement*

Given a network $G(V, E)$ and a traffic demand set $D$, the problem is to generate (i) the minimum number of powered on cables, and (ii) the path set $MP_\alpha$ that can be used to route each traffic in $D_\alpha$ while using only the powered-on cables, subject to two constraints: (C1) the utilization of each link $e_{ij}$ is no larger than a given threshold $u_T$, *i.e.*, $u_{ij} \leq u_T$, and (C2) the length of each path $cp_{\alpha,q} \in CP_\alpha$ is no longer than a given constraint $d_{T,\alpha}$. In other words, the problem is to find as many cables as possible that can be switched off while satisfying all traffic demands in $D$ under constraints (C1) and (C2). Similar to [5], we set the MLU to $u_T \leq 50\%$; this over-provisioning is necessary to maintain network fault tolerance and performance. For delay, we consider two path length constraints when routing each demand $D_\alpha$ with powered-off cables: (C2.1) each $D_\alpha$ is routed through one or more paths with a bounded delay $d_{T,\alpha} \leq \mathrm{ND}$; ND is the network diameter of the original network, or (C2.2) each $D_\alpha$ is routed through one of more paths with threshold delay $d_{T,\alpha} \leq d_\alpha * \lambda$, for a multiplier $1.0 \leq \lambda \leq 2.0$. Formally, we have,

$$\text{Min} \quad \sum_{e_{ij} \in E} p_{ij} n_{ij} \tag{1}$$

$$\text{s.t.} \quad \sum_{j \in V} f_{sj}^{\alpha} = \sum_{j \in V} f_{jt}^{\alpha} = f^{\alpha}, \forall D_\alpha \in D \tag{2}$$

$$\sum_{j \in V} f_{ij}^{\alpha} = \sum_{j \in V} f_{ji}^{\alpha}, \forall D_\alpha \in D, i \neq s, t \tag{3}$$

$$\sum_{D} \sum_{CP_\alpha} \delta_{ij}^{\alpha,q} f_{ij}^{\alpha} \leq (n_{ij} / w_{ij}) * c_{ij}, \forall e_{ij} \in E \tag{4}$$

$$(w_{ij} * \sum_{D} f_{ij}^{\alpha}) / (n_{ij} * c_{ij}) \leq u_T, \forall e_{ij} \in E \tag{5}$$

$$| cp_{\alpha,q} | \leq d_{T,\alpha} = d_\alpha * \lambda, \forall D_\alpha \in D \tag{6}$$

Eq. (1) quantifies the total energy consumed by all active cables. Eq. (2) ensures that the sum of flows leaving a source or entering a destination equal to $f^\alpha$. Eq. (3) ensures that no flow is lost, while Eq. (4) computes the flow in each link and restricts each link to carry flow no more than its capacity. Eq. (5) computes the link utilization and limits it to at most $u_T$. Finally, Eq. (6) restricts each path delay to be no more than $d_{T,\alpha}$. The resulting formulation is a MIP problem, which is NP-hard, due to the integer variables $n_{ij}$.

## III. GREEN ROUTING ALGORITHMS

Fig. 1 describes the main steps of MSPF. Step 1 uses Yen's algorithm [8] to generate $k$ shortest paths, $CP_\alpha$, for each demand $D_\alpha$, each of which has delay no more than $d_{T,\alpha}$. Since we set each link's weight to one, the shortest path in the $k$ paths has the smallest hop count; the next shortest has the second smallest, and so forth. Step 2 uses the function ***D-Flow*()** to distribute the traffic flow of each demand $D_\alpha$ through one or more candidate paths in $CP_\alpha$. The function aims to distribute the flow starting from the shortest candidate path $cp_{\alpha 1}$. If $cp_{\alpha 1}$ cannot satisfy all flow of $D_\alpha$, it uses the second candidate path $cp_{\alpha 2}$ to carry the remaining flow, and so forth, until the flow of each demand $D_\alpha$ is supported by the network. The function returns false if the flow in $D_\alpha$ cannot be routed through the candidate paths. Otherwise, it returns true and creates a set of $MP_\alpha$ that contains all paths used to route demand $D_\alpha$. Notice that in Step 2 the function would always return true since we assume that the original network has sufficient capacity to meet the demand requirements. Step 3 calculates the total flow $f_{ij}$ for each link $e_{ij}$, and computes the remaining link capacity $r_{ij} = c_{ij} - f_{ij}$ which in turn is used to calculate the maximum number of redundant cables $\lfloor r_{ij} \rfloor$ to shut down.

Step 4a) repeatedly selects a candidate cable $b_{ij}$ to switch off; it targets the cable whose link $e_{ij}$ has the largest remaining capacity $r_{ij}$, as calculated using (7).

$$r_{ij} = u_T * n_{ij} * c_{ij} / w_{ij} - f_{ij}, \forall e_{ij} \in E \tag{7}$$

Step 4b) uses function ***Reroute-D*()**, shown in Fig. 2, to reroute the flow in one or more paths in $MP_\alpha$ that are affected by the removal of $b_{ij}$. If rerouting is possible, it deletes $b_{ij}$ and puts it in the set of powered off cables $B_d$. Otherwise, it knows that the cable must be switched on to ensure the feasibility of satisfying all demand flow; therefore it sets *fix*($e_{ij}$)=*TRUE*. When feasible, the function generates a new set $MP_\alpha$ for each demand $D_\alpha$ affected by the cable's deletion. Steps 4a) and 4b) are repeated until each *fix*($e_{ij}$) is *TRUE*.

We define the set $B_d$ = ($b_{ij}$ | all powered off cables in $\forall e_{ij} \in E$). Step 1 of function ***Reroute-D*()** repeatedly finds each $D_\alpha$ affected by the deletion of edge *del_e*; $D_\alpha$ is affected if any path in $MP_\alpha$ contains *del_e*. The step places all paths that contain *del_e* in the set $DP_\alpha$. Then, for each $e_{ij} \in P$ and each $P \in DP_\alpha$, Step 2 increases $u_{ij}$ by the flow of path $P$. This step is needed since the function wants to redistribute the flow of each path in $DP_\alpha$. However, as shown in Step 4, the function will revert to each $u_{ij}$'s capacity if redistributing the flow in Step 3 is not feasible. In Step 3, the function aims to distribute the affected flow of $D_\alpha$, *i.e.*, $f^\alpha$-*flow*($DP_\alpha$), where *flow*($DP_\alpha$) denotes the total flow of all paths in $DP_\alpha$, through the remaining candidate paths, *i.e.*, ($CP_\alpha$-$DP_\alpha$**)**. If function ***D-Flow*()** returns false for any $D_\alpha$, deleting *del_e* is not feasible, and therefore function ***Reroute-D*()** returns false.

```
1) For each demand $D_\alpha \in D$, generate $CP_\alpha$;
2) For each demand $D_\alpha \in D$, Call D-Flow($f^\alpha$,$CP_\alpha$);
3) For each $e_{ij}$, calculates $r_{ij}$, remove the maximum cables such that all
   flows are still satisfied and set fix($e_{ij}$) $\leftarrow$ FALSE for each link $e_{ij}$;
4) Repeat
    a) Find a candidate edge $e_{ij} \in E$ using (8), remove $b_{ij}$, and put the
       cable in $B_d$.
    b) Call Reroute-D($e_{ij}$)
        (i) If feasible go to Step 3.
        (ii) If not feasible, retain $b_{ij}$, remove it from $B_d$, and set
             fix($e_{ij}$)$\leftarrow$TRUE,
   Until fix($e_{ij}$)=TRUE for every $e_{ij} \in E$.
```

Figure 1. MSPF algorithm

```
For each $D_\alpha \in D$
1) Place all paths in $MP_\alpha$ that contain del_e in $DP_\alpha$
2) Increase $u_{ij}$ of $e_{ij}$ in path $P \in DP_\alpha$ by the flow in $P$
3) Call D-Flow(($f^\alpha$-flow($DP_\alpha$), ($CP_\alpha$ - $DP_\alpha$))
4) If not feasible, retain the original value of each $u_{ij}$ in Step 2 and
   return false; else return true.
```

Figure 2. Function Reroute-D(*del_e*)

## IV. EVALUATION

### A. Experiment Setup

To evaluate MSPF's performance, we used four topologies, *i.e.*, Abilene [10], GÉANT [11], Sprint [7] and AT&T [7]. For each network and each link $e_{ij}$, we consider bundle size $w_{ij}$ ranging from 1 to 10 and MLU $u_T \leq 50\%$.

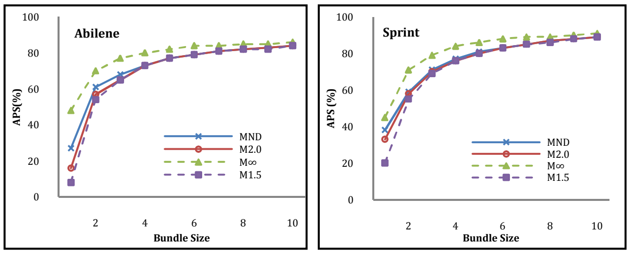
We used the Abilene topology and traffic matrices measured on Sep. 5th, 2004 for every five minutes, which are provided by the authors of [10]. For GÉANT, its traffic matrices were collected on May 5th, 2005 for every 15 minutes; we obtained both the topology and traffic matrices from the authors of [11]. For Sprint and AT&T, we randomly generate a traffic matrix using the gravity model [12], and scaled the traffic to obtain 40 different traffic matrices. Simulation runs were carried out on a Linux PC with 3.07GHz CPU and 8GB RAM. We ran source codes of [5] and [6], provided by their respective authors, using the CPLEX [13] LP solver.

### B. Power Savings

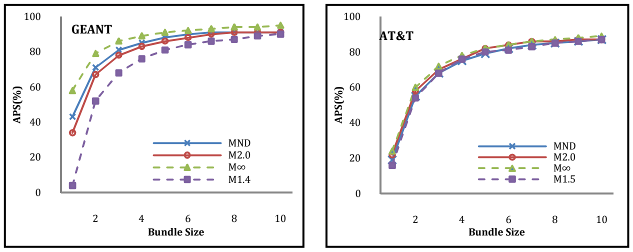
We compute the power saving ratio as the total power of sleeping cables over the total power of all cables in the network. The power consumption of line-cards we use in the evaluation is specified in [9]. Let $M_{ND}$ and $M_{\lambda}$ represent the energy savings generated by MSPF when the delay constraints (C2.1) and (C.2.2) are set to $d_{T,\alpha} \leq$ND and $d_{T,\alpha} \leq d_{\alpha} * \lambda$, respectively. Further, $M_{\infty}$ denotes the upper bound on energy saving when the delay constraint is set to infinity. We used the LP solution in [5] to find the minimum delay multiplier λ that allows a feasible solution for Abilene, GÉANT, Sprint and AT&T, which require a minimum λ of 1.5, 1.4, 1.5 and 1.5 respectively. We used the pre-computed λ in MSPF to produce the lower bound energy saving of the networks. In other words, $M_{1.5}$, $M_{1.4}$, $M_{1.5}$, $M_{1.5}$ are the lower bound energy savings on the respective networks produced by our MSPF.

Fig. 3(a) shows the average power savings for Abilene over the 288 traffic matrices for $w_{ij}$=1,2, …,10. For $w_{ij}$=1, $M_{ND}$=27% is better than $M_{2.0}$=15% because, for each $D_{\alpha}$, there are more paths with $|cp_{\alpha,q}| \leq$ND than $|cp_{\alpha,q}| \leq 2.0 * d_{\alpha}$; thus MSPF can use more candidate paths for $M_{ND}$ than for $M_{2.0}$. It also shows that the average power savings increases sharply when the bundle size increases from 1 to 2, and 2 to 3 for both $M_{ND}$ and $M_{2.0}$. Notice that MSPF produces the best energy saving $M_{ND}$=$M_{2.0}$=84% for $w_{ij}$=10. For $M_{\infty}$ and $M_{1.5}$, they have the same trend as $M_{ND}$ and $M_{2.0}$ when the bundle size increases from 1 to 10. $M_{\infty}$ starts from the 46% to 86% while $M_{2.0}$ is from 8% but still to 84%.

Fig. 3(c) shows the power saving of GÉANT averaged over the 96 traffic matrices for $w_{ij}$=1, 2, …,10. For $w_{ij}$=1, $M_{2.0}$=34% is lower than $M_{ND}$=43% because the network contains fewer paths that has length $|cp_{\alpha q}| \leq 2 * d_{\alpha}$ than $|cp_{\alpha q}| \leq$ND; thus MSPF has a smaller search space on the former than the latter constraint. Notice the significant jump in energy savings, i.e., $M_{ND}$=71% and $M_{2.0}$=67%, when the bundle size increases to $w_{ij}$=2. Both $M_{ND}$ and $M_{2.0}$ reach their peak at 91% when $w_{ij}$=10. The gap between $M_{\infty}$ and $M_{1.5}$ is very large; in fact, it exceeds 50% for $w_{ij}$=1 but less than 5% for $w_{ij}$=10.

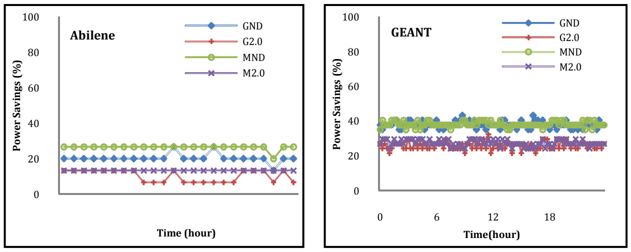


(a) Abilene (b) Sprint

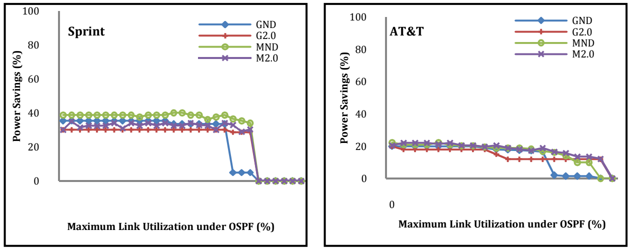


(c) GÉANT (d) AT&T

Figure 3. Power Savings of MSPF



(a) Abilene (b) GÉANT



(c) Sprint (d) AT&T

Figure 4. Comparison between MSPF and GreenTE[5]

Fig. 3(b) and (d) show the average power savings of Sprint and AT&T for $w_{ij}$=1 to $w_{ij}$=10. For Sprint, MSPF uses the first 100 shortest paths to reroute each demand, *i.e.*, $k$=100; we set $k$=20 for AT&T. As shown in Fig. 6 and 7, the power savings for Sprint and AT&T also increase sharply as we increase the bundle size from 1 to 2; their peak also occurs when $w_{ij}$=10. For Sprint, the upper bound $M_{\infty}$=42% is more than twice that of the lower bound $M_{1.5}$=19%. For the AT&T, the upper bound is very close to the lower bound, *i.e.*, $M_{\infty}$=22% versus $M_{1.5}$=19%.

### C. MSPF versus FGH

While FGH [6] guarantees that its result would provide sufficient powered on cables for rerouting the given traffic demands, the length/delay of rerouted traffics might exceed their upper bound. Further, the solution may increase the utilization of each link to be above a threshold that may affect

TABLE I. AVERAGE RUNNING TIME (CPU SECONDS); $u_T$=100% AND $\lambda=\infty$

| Algorithm | Abilene | GÉANT | Sprint | AT&T |
|---|---|---|---|---|
| MSPF_NC | 0.037 | 1.71 | 3.526 | 10.35 |
| FGH | 5.1 | 63.9 | 1184.3 | 2965.2 |

TABLE II. AVERAGE POWER SAVING (%); $u_T$=100% AND $\lambda=\infty$

| Algorithm | Abilene | GÉANT | Sprint | AT&T |
|---|---|---|---|---|
| MSPF_NC | 53.3 | 58.1 | 45.29 | 23.64 |
| FGH | 53.3 | 58.5 | 43.02 | 25.3 |

the network's resilience against failures or network congestion during peak hours. To compare the performance of MSPF against FGH, we set $u_T$=100% and $\lambda=\infty$, *i.e.*, a scenario where there is no upper limit on link utilization nor traffic delay; the results are outlined in Table III and IV. As shown in Table III and IV, MSPF runs significantly faster than FGH while producing similar energy savings. MSPF requires only 0.73%, 2.68%, 0.3%, and 0.35% of the computation time of FGH for the Abilene, GÉANT, Sprint, and AT&T networks respectively. Notice that MSPF produces equivalent or better energy savings for Abilene and Sprint. Further, MSPF is more efficient as compared to FGH on larger networks, *i.e.*, Sprint and AT&T.

### *D. MSPF versus GreenTE*

GreenTE [5] assumes a hierarchical topology, which is typical of a Wide Area Network (WAN) where all links are assumed to be bidirectional. Thus, each pair of directional links from $v_i$ to $v_j$, *i.e.*, link $e_{ij}$ and $e_{ji}$, must be turn on or off together. GreenTE aims to maximally switch off paired directional links. Further, their model considers $d_{T,\alpha}\leq$ND or $d_{T,\alpha}\leq d_\alpha$*2.0, $w_{ij}$=1; the model does not consider links with bundled cables. To ensure fair comparison, we set the same values for $u_T$, $d_{T,\alpha}$, and $w_{ij}$ for both GreenTE and MSPF. Let $G_{ND}$ and $G_{2.0}$ represent the energy saving produced by GreenTE when its delay constraint is set to the network diameter and twice of the shortest path, respectively.

Fig. 4(a) and (b) show a comparison of power savings with $u_T\leq$50% on Abilene and GÉANT over 24 hours. As shown in Fig. 4(a), for Abilene, MSPF can shut down more cables than GreenTE, resulting in energy saving of almost $M_{ND}$=27%, a 7% improvement over GreenTE. For delay constraint C2.2, MSPF consistently obtained $M_{2.0}$=13.33%, better than GreenTE whose $G_{2.0}$ ranges between 8% and 13.33%. In Fig. 8, for GÉANT, the average power savings of running MSPF is always larger than GreenTE with $\lambda = 2.0(G_{2.0} \leq M_{2.0})$; *i.e.*, around 25%. In terms of running time, MSPF requires only about 2-3 CPU seconds to produce its results, significantly faster than GreenTE, which required 300 CPU seconds while producing results that incur higher energy expenditure.

Fig. 4(c) and (d) compare the performance of MSPF against GreenTE when the MLU under Open Shortest Path First (OSPF) increases from 2 to 100% for Sprint and AT&T, using the traffic matrices generated as described in Section IV.A. As shown in Fig. 9, for $u_T\leq$70%, MSPF outperforms GreenTE, on average about 5% in power saving for delay constraint C2.1; see $M_{ND}$ and $G_{ND}$. Similarly, MSPF achieves power saving $M_{2.0}$ on average 3% better than $G_{2.0}$ generated by GreenTE. Notice that GreenTE produces the results for these large topologies in 300 seconds; CPLEX [13], used in GreenTE, was unable to produce the optimal solution, and therefore, as suggested in [5], we stopped CPLEX after it ran for 300 seconds. In contrast, MSPF uses approximately 10 seconds while producing better energy savings for Sprint and AT&T's networks.

## V. CONCLUSION

We have described a problem to reduce the energy usage of networks comprising links with bundled cables. Our MSPF turns off unused cables during off-peak periods such that the remaining powered on cables have sufficient capacity to support the given traffic demands. Further, each demand is only re-routed through one or more paths with lengths no longer than a given constraint, and each link's utilization does not exceed a given threshold. Our results show that MSPF is superior against two state-of-the-art techniques. We will extend our work so that the resulting network also provides a lower bound on reliability.

## REFERENCES

[1] Global Action Plan, "An inefficient truth". http://www.globalactionpaln.rog.uk/, December, 2007.

[2] K. W. Roth, F. Goldstein and J. Kleinman, "Office and telecommunications equipment in commerical buildings-volume I: Energy consumption baseline". In Tech. Report 7289500, Arthur D. Little, Inc., 2002.

[3] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang and S. Wright, "Power awareness in network design and routing". In INFOCOM, 2008.

[4] L. Chiaraviglio, M. Mellia, and F. Neri, "Energy-aware Backbone Networks: a Case Study". In GreenComm, 2009.

[5] M. Zhang, C. Yi, B. Liu, and B. Zhang, "GreenTE: Power-Aware Traffic Engineering". In ICNP, 2010.

[6] W. Fisher, M. Suchara, and J. Rexford, "Greening backbone networks: reducing energy consumption by shutting off cables in bundled links". In Green Networking, 2010.

[7] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, "Measuring ISP Topologies with Rocketful" In IEEE/ACM Transactions on Networking, vol. 12, no. 1, pp. 2-16, 2004.

[8] J. Y. Yen, "Finding the K Shortest Loopless paths in a network". In Management Science, 17(11), 1971.

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

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

[11] S. Uhlig, B. Quoitin, J. Lepropre, and S. Balon, "Providing Public Introdomain Traffic Matrices to the Research Community". ACM SIGCOMM Computer Communication Review, Vol. 36, no. 1, pp. 83-86, January 2006.

[12] 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 ACM SIGCOMM Internet Measurement Workshop, 2002.

[13] "CPLEX". [Online]. Available: http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/.

[14] R. Doverspike, K. K. Ramakrishnan, and C. Chase, "Structual overview of ISP networks". In Guide to Reliable Internet Services and Applications , Springer, 2010.

[15] V. Soteriou and L.S. Peh, "Design-Space Exploration of Power-Aware On/Off Interconnection Networks". In ICCD 2004: 510-517.