

Routing for Energy Minimization in the Speed Scaling Model

Matthew Andrews
Bell Labs
Murray Hill, NJ
andrews@research.bell-labs.com

Antonio Fernández Anta
U. Rey Juan Carlos
Móstoles, Madrid, Spain
anto@gsyc.es

Lisa Zhang
Bell Labs
Murray Hill, NJ
ylz@research.bell-labs.com

Wenbo Zhao
UCSD
La Jolla, CA
w3zhao@ucsd.edu

Abstract—We study network optimization that considers energy minimization as an objective. Studies have shown that mechanisms such as *speed scaling* can significantly reduce the power consumption of telecommunication networks by matching the consumption of each network element to the amount of processing required for its carried traffic. Most existing research on speed scaling focuses on a single network element in isolation. We aim for a network-wide optimization.

Specifically, we study a routing problem with the objective of provisioning guaranteed speed/bandwidth for a given demand matrix while minimizing energy consumption. Optimizing the routes critically relies on the characteristic of the energy curve $f(s)$, which is how energy is consumed as a function of the processing speed s . If f is superadditive, we show that there is no bounded approximation in general for integral routing, i.e., each traffic demand follows a single path. This contrasts with the well-known logarithmic approximation for subadditive functions. However, for common energy curves such as polynomials $f(s) = \mu s^\alpha$, we are able to show a constant approximation via a simple scheme of randomized rounding.

The scenario is quite different when a non-zero startup cost σ appears in the energy curve, e.g. $f(s) = \begin{cases} 0 & \text{if } s = 0 \\ \sigma + \mu s^\alpha & \text{if } s > 0 \end{cases}$. For this case a constant approximation is no longer feasible. In fact, for any $\alpha > 1$, we show an $\Omega(\log^{\frac{1}{4}} N)$ hardness result under a common complexity assumption. (Here N is the size of the network.) On the positive side we present $O((\sigma/\mu)^{1/\alpha})$ and $O(K)$ approximations, where K is the number of demands.

I. INTRODUCTION

Energy conservation is emerging as a key issue in computing and networking as the ICT sector (Information and Communications Technologies) significantly steps up the energy efficiency of its products and services, in response to growing energy bills, government mandates, as well as societal pressures to minimize the carbon emissions by the sector [5], [30]. Many methods for avoiding waste and improving energy efficiency are being developed. One opportunity to reap significant potential saving in data networking is achieving *energy proportionality* [12]. Energy proportionality refers to a goal in which the amount of energy consumed by a network element in proportion to the carried traffic load. We use network element as a generic term that represents a computing and communication resource such as a router, switch, CPU or a link connecting this equipment.

As indicated in a study conducted by the US Department of Energy [4], the current network elements and telecommunication networks are not designed with energy optimization as an objective or a constraint. They are often designed for

peak traffic, for reasons such as accommodating future growth, planned maintenance or unexpected failures, or quality-of-service guarantees. At the same time, the energy consumption of network elements is often defined by the peak profile and varies little for typical traffic, which can be a small fraction of the peak. By a conservative estimate in the same study, at least 40% of the total consumption by network elements such as switches and routers can be saved if energy proportionality is achieved. This translates to a saving of 24 billion kWh per year attributed to data networking [4].

Two popular methods for effectively matching power consumption to traffic load are via *speed scaling* or *powering down*. The former refers to setting the processing speed of a network element according to traffic load, and the latter refers to turning off the element. Both methods are the subject of active research, though most of the work focuses on optimizing an individual element in isolation [26], [21], [37], [31], [11], [16], [27], [28], [25]. In addition, enabling sleep modes and sizing power to traffic are also features in some commercial products such as the Intel pentium processors [3], standards like ADSL2 and ADSL2+ [22], or proposals to the IEEE 802.3az task forces [20], [34]. Our goal is to examine the optimization problems that arise in a network consisting of multiple network elements.

We focus on the speed scaling model in this paper, and study the power down model in a separate effort [7]. We assume each network element e has the speed scaling capability, characterized by an energy curve $f_e(s)$, which is how e consumes energy as a function of its processing speed s . We propose a routing problem with the objective of provisioning for a long-timescale traffic matrix and minimizing the total energy consumption by the network elements over the entire network. Routing determines the traffic load on each network element and this in turn determines the energy consumption specified by the energy curve $f_e(\cdot)$.

The algorithmic aspect of this routing problem critically relies on the nature of the energy curve $f_e(s)$. For example, if the curve should be linear $f_e(s) = \mu s$, then shortest path routing is optimal. Unfortunately, situations in reality are far more complex. For example, several preliminary studies on Ethernet links, edge routers, e.g. [15], [24] and the well-accepted understanding of optical links and equipment suggest that the energy consumption for network elements as such may grow *subadditively* with the speed. That is, doubling speed less than doubles the energy, or more formally $f_e(s_1) + f_e(s_2) \geq$

$f_e(s_1+s_2)$. In this case, the routing problem corresponds to the well-studied problem of buy-at-bulk network design (BAB), e.g. [10], [17], [18], [6]. BAB has a logarithmic approximation and almost matching logarithmic hardness.

On the other hand, the energy consumption of a microprocessor grows *superadditively* with the speed. That is, doubling the speed more than doubles the energy consumption, or more formally $f_e(s_1) + f_e(s_2) \leq f_e(s_1 + s_2)$. Furthermore, the energy curve is often modeled by a polynomial function $f_e(s) = \mu_e s^\alpha$ where μ_e and α are parameters associated with the device. (While α has been usually assumed to be around 3 [14], it has been recently estimated to be much smaller. In particular its value is 1.11, 1.66, and 1.62 for the Intel PXA 270, a TCP offload engine, and the Pentium M 770, respectively [36].) Note that if *fractional* routes are allowed, i.e. a demand may be carried on multiple paths between its source and destination, then the problem falls into the realm of convex optimization since μs^α is convex, and therefore is solvable in polynomial time [13]. However, for *integer routes* where each demand must be carried on one single path, the problem has to the best of our knowledge not received much attention before. Integral routing can be important for a number of reasons, e.g. if we wish to avoid problems associated with packet reordering.

In addition, a more accurate but more complex energy curve for a microprocessor may be $f_e(s) = \begin{cases} 0 & \text{if } s = 0 \\ \sigma_e + \mu_e s^\alpha & \text{if } s > 0 \end{cases}$ where σ_e represents the non-negligible power consumption by leakage currents, see e.g. [3]. This energy function is neither superadditive nor subadditive, and little is known about routing optimization subject to such functions. For the rest of the paper we explore these two well-motivated energy functions by showing how to approximate the optimal solution as well as the limit to which approximation can be accomplished.

Previous Work: Speed scaling has been widely studied to save energy at the single element level. Yao et al. [37] were the first to study speed scaling in processors, in the form of task scheduling problems. They assumed that the energy to run a processor at a speed s grows superlinearly with s , and explored the problem of scheduling a set of tasks with the smallest amount of energy. Speed scaling has also been combined with powering down in the same context of power-efficient task scheduling [27]. (The survey of Irani and Pruhs [26] reviews results and open problems under the speed scaling and powerdown models for processors.) In networks, most effort has been invested in reducing the consumption at the edge of the Internet (edge links and routers). For instance, Gunaratne et al. [24] have proposed a Markovian model to optimize single Ethernet link usage with speed scaling.

To the best of our knowledge, only a few papers study energy saving at a global wired network level. For instance, Nedeveschi et al. [33] explore both speed scaling and powerdown as techniques to globally reduce energy consumption. When using speed scaling, they consider two alternative models, one in which only the frequency of the transmission can

be scaled, and one in which also the operational voltage can be scaled. The authors propose heuristics for these models, that are evaluated empirically. While energy saving has always been a concern in wireless networks [29] (since mobile devices work on limited-energy batteries), they are intrinsically different from wireline networks, and are not considered here.

II. MODEL AND RESULTS

We are given a network modeled by an undirected graph G and a set of demands. Each demand i requests d_i integer units of bandwidth between a source node s_i and a destination node t_i . We are also given a cost function $f_e(s)$ on each link¹ e that represents the energy consumption for routing s units of demand through link e . Our aim is to route all of the demands on integral routes with minimum cost. Not surprisingly, the routing problem is NP hard for most functions $f_e(\cdot)$. We therefore consider approximation algorithms. A polynomial-time algorithm is a β -approximation if for all instances it returns a solution at most β times the optimal. A problem has no β -approximation if no polynomial-time algorithm can guarantee a β -approximation for all instances under complexity assumptions such as $P \neq NP$.

Formally, the min-energy routing problem can be formulated as the following program. Let binary variable $y_{i,e}$ indicate whether demand i passes through link e and x_e be the total load on e . Our route optimization problem can be formulated as follows.

$$\begin{aligned} (P_1) \quad & \min \sum_e f_e(x_e) \\ \text{subject to} \quad & x_e = \sum_i y_{i,e} d_i \quad \forall e \\ & y_{i,e} \in \{0, 1\} \quad \forall i, e \\ & y_{i,e} : \text{flow conservation} \end{aligned}$$

Let $I_i(v)$ and $O_i(v)$ be the amount of demand i entering and leaving node v , respectively, and $F_i(v) = O_i(v) - I_i(v)$. Flow conservation means that $F_i(s_i) = d_i$, $F_i(t_i) = -d_i$, and $F_i(v) = 0$ for any other node v . As mentioned in the Introduction, if the cost function $f_e(\cdot)$ is subadditive then this corresponds to the well-studied buy-at-bulk network design problem. The following summarizes the main results for buy-at-bulk.

Theorem 1 (BAB). *For subadditive cost functions $f_e(\cdot)$, (P_1) has $O(\log N)^2$ approximation ratio [10] and $\Omega(\log^{1/4} N)$ hardness bound [6] if $f_e(\cdot)$ is uniform over all e ; (P_1) has $O(\log^3 N)$ approximation ratio and $\Omega(\log^{1/2} N)$ hardness bound if $f_e(\cdot)$ is different from edge to edge. Here N is the size of the network.*

In this paper we are interested in less studied functions such as superadditive, and mixed sub and superadditive $f_e(\cdot)$. To provide a contrast with subadditive functions, we first show

¹Cost functions for other network elements is left for future study.

²All logarithms are to the base 2.

via a simple reduction that extremely simple superadditive functions, such as $f_e(x) = \max\{0, x - 1\}$, lend to unbounded approximations.

Lemma 2. *If a monotone function $f_e(\cdot)$ satisfies $f_e(1) = 0$ and $f_e(2) > 0$ for all e , then there is no polynomial time algorithm to the min-energy routing problem with any finite approximation ratio unless $P=NP$.*

Proof: The reduction is from the edge-disjoint path (EDP) problem, which is known to be NP-hard. Given a network and a set of demand, EDP decides if all demands can be routed along edge-disjoint paths. If EDP has a solution, then the resulting load on each edge is at most 1, which implies a solution of cost 0 for the min-energy problem. In contrast, if EDP has no solution, in any solution some link must have load at least 2, which implies an optimal min-energy solution of cost at least $f(2) > 0$. Hence, a bound-approximation to the min-energy problem would return a zero solution iff EDP has a solution. ■

We focus on two non-subadditive functions in this paper, both because they do not have the issue stated in Lemma 2 and because they closely model energy consumption of certain network elements, as discussed earlier. We state the following main results.

- In Section III we consider polynomial functions of the form $f_e(s) = \mu_e s^\alpha$. For uniform demands where d_i is the same for all i , we prove a γ -approximation where γ only depends on α . Since, as mentioned before, α is very small in practice (less than 2), we consider this to be a constant approximation. This result generalizes to an approximation that is logarithmic on $D = \max_i d_i$ for nonuniform demands.
- In Section IV we consider polynomial functions with a startup cost, $f_e(s) = \begin{cases} 0 & \text{if } s = 0 \\ \sigma_e + \mu_e s^\alpha & \text{if } s > 0 \end{cases}$. In contrast to polynomial cost functions where $\sigma_e = 0$, we show that there is no $O(\log^{\frac{1}{4}} N)$ -approximation algorithm under a common complexity assumption. This lower bound even holds when all $d_i = 1$ and the cost functions $f_e(\cdot)$ are identical for all e . On the positive side, we present an $O(K)$ approximation for unit demands, where K is the number of demands. We also show an $O((\max_e \{\sigma_e/\mu_e\})^{1/\alpha} + 1)$ -approximation, independent of K , for uniform demands. Again, for nonuniform demands, an additional factor logarithmic in D appears in the approximation ratios.
- In Section V we evaluate our proposed approximation algorithms via simulations. For polynomials without a startup cost, randomized rounding performs superbly. When the startup is large, both approximations from Section IV are less than satisfactory. However, we present a heuristic that appears to rectify the situation.

III. APPROXIMATION FOR POLYNOMIAL COST FUNCTIONS

In this section we use randomized rounding on the convex program (P_1) to approximate the optimal cost for polynomial

cost functions $f_e(s) = \mu_e s^\alpha$.

We first relax the binary constraint on flow variables $y_{i,e} \in \{0, 1\}$ to $y_{i,e} \in [0, 1]$. As a result, for polynomial cost functions $f_e(\cdot)$, the routing problem is convex programming and is optimally solvable [13]. From the optimal fractional routing, we round the fractional flow in the Raghavan-Thompson manner [35] as follows. We first decompose the fractional solution defined by $y_{i,e}$ into weighted flow paths for each demand i via the following standard procedure. We repeatedly extract paths connecting the source and destination nodes of demand i from the subgraph defined by links e for which $y_{i,e} > 0$. If p is extracted, then the weight of p is $w_p = \min_{e \in p} y_{i,e}$ and the $y_{i,e}$ value of every link along p is reduced by w_p . The flow conservation constraint on $y_{i,e}$ guarantees that when the last path is extracted for demand i , every $y_{i,e}$ is zero. Following the flow decomposition, we randomly choose one path from the potentially multiple paths for each demand, using the path weight as the probability. At the end of the rounding, every demand follows one single path.

Obviously, the fractional optimal solution is a lower bound of the integral optimal solution. If we could bound the difference between the rounded solution and the fractional optimal, we would have bounded the difference between the rounded solution and the integral optimal. Unfortunately, the direct application of randomized rounding as described above does not guarantee a good approximation. For example, consider a network with two nodes u, v and m parallel links connecting them, one unit-demand with source u and destination v , and a uniform cost function $f_e(x) = x^\alpha$. The optimal fractional solution to (P_1) distributes the demand evenly among the m links, resulting in a cost of $m \cdot f_e(1/m) = 1/m^{\alpha-1}$. The optimal integral solution has to send the demand along one of the edges, resulting in cost $f_e(1) = 1$. Hence, the integrality gap is $m^{\alpha-1}$. However, we now show how to adapt this procedure in order to overcome this difficulty.

A. Uniform Demands

The essence of the previous example stems from the behavior of $f_e(\cdot)$ in the interval $[0, 1]$. We observe that, in fact, for $x \in [0, 1]$ we can use the cost function $f_e(x) = \mu_e x$ since $\mu_e x$ and $\mu_e x^\alpha$ agree on $x = 0$ and $x = 1$. More importantly, if we do this the integrality gap in the aforementioned example disappears. Formally, for unit demands, i.e. $d_i = 1$, we define the cost function

$$g_e(x) = \mu_e \max\{x, x^\alpha\}. \quad (1)$$

Note that minimizing $\sum_e g_e(x_e)$ has the same integral optimal as the original program (P_1) since $f_e(\cdot)$ and $g_e(\cdot)$ agree on all integral values. In addition, the optimal fractional solution with respect to $g_e(\cdot)$ can still be obtained by convex programming as $g_e(\cdot)$ is still convex after linearizing $f_e(\cdot)$ in the interval $[0, 1]$. We use this observation to show that randomized rounding gives a constant factor approximation for unit demands. Let x_e^* be the flow on link e under the optimal fractional routing and let \hat{x}_e be the resulting rounded flow. We show,

Lemma 3. For unit demands, randomized rounding the optimal fractional solution x_e^* with respect to the cost function $g_e(x)$ guarantees that $E[g_e(\hat{x}_e)] = E[f_e(\hat{x}_e)] \leq \gamma g_e(x_e^*)$, for some constant γ and all links e .

Proof: Observe that $E[\hat{x}_e] = x_e^*$. We consider two cases $x_e^* \leq 1$ and $x_e^* > 1$.

Case 1: $x_e^* \leq 1$. We show that $E[f_e(\hat{x}_e)] \leq \gamma_1 g_e(x_e^*)$ for some constant γ_1 . We partition the possible values of \hat{x}_e into the ranges $[0, 1)$, $[1, 2)$, $[2, 4)$, \dots . We have,

$$\begin{aligned} E[f_e(\hat{x}_e)] &\leq f_e(\hat{x}_e = 0)Pr[\hat{x}_e < 1] + \\ &\quad \sum_{j \geq 0} f_e(\hat{x}_e = 2^{j+1})Pr[\hat{x}_e \geq 2^j] \\ &\leq 0 + \sum_{j \geq 0} \mu_e (2^{j+1})^\alpha \left(\frac{e^{\frac{2^j}{x_e^*} - 1}}{\left(\frac{2^j}{x_e^*}\right)^{\frac{2^j}{x_e^*}}} \right)^{x_e^*} \\ &= \mu_e x_e^* \sum_{j \geq 0} \frac{2^{\alpha(j+1)}}{x_e^*} \frac{x_e^{*2^j}}{e^{x_e^*}} \left(\frac{e}{2^j}\right)^{2^j} \\ &\leq g_e(x_e^*) \sum_{j \geq 0} 2^{\alpha(j+1) - 2^j(j - \lg e)}. \end{aligned} \quad (2)$$

The first inequality follows from the definition of expectation. The second follows from a Chernoff bound [32, Theorem 4.4(1)]. We obtain the third inequality via algebraic manipulation and the fact that $0 \leq x_e^* \leq 1$. Let $j_0 = \lceil 2 \lg(\alpha + 4) \rceil$. Via further algebraic manipulation we can show that all the terms in (2) for which $j \geq j_0$ add up to at most 1. Hence, there is a constant γ_1 (dependent on α) such that $E[f_e(\hat{x}_e)] \leq \gamma_1 g_e(x_e^*)$.

Case 2: $x_e^* > 1$. We partition the possible values of x_e into the ranges $[0, x_e^*)$, $[x_e^*, 2x_e^*)$, $[2x_e^*, 4x_e^*)$, \dots . By the definition of expectation, we have

$$\begin{aligned} E[f_e(\hat{x}_e)] &\leq f_e(\hat{x}_e = x_e^*)Pr[\hat{x}_e \geq 0] + \\ &\quad \sum_{j \geq 0} f_e(\hat{x}_e = 2^{j+1}x_e^*)Pr[\hat{x}_e \geq 2^j x_e^*] \\ &\leq g_e(x_e^*) + \sum_{j \geq 0} 2^{\alpha(j+1)} g_e(x_e^*) \left(\frac{e^{2^j - 1}}{(2^j)^{2^j}} \right)^{x_e^*} \\ &\leq g_e(x_e^*) \left(1 + \sum_{j \geq 0} 2^{\alpha(j+1) - (j - \lg e)2^j - \lg e} \right), \end{aligned}$$

where the second inequality follows from a Chernoff bound [32, Theorem 4.4(1)], and the third inequality follows from $x_e^* > 1$. The summation for $j \geq 2$ can be bounded similar to the one in Eq (2). Hence, there is a constant γ_2 (dependent on α) such that $E[f_e(\hat{x}_e)] \leq \gamma_2 g_e(x_e^*)$. Combining both cases we have that, for $\gamma = \max\{\gamma_1, \gamma_2\} > 0$, every link e satisfies that $E[f_e(\hat{x}_e)] \leq \gamma g_e(x_e^*)$. ■

It is easy to see that Lemma 3 for unit demands also applies to uniform demands in which $d_i = d$ for all demands i . By linearizing $f_e(\cdot)$ in the range of $[0, d]$ instead of $[0, 1]$, we define

$$g_e(x) = \mu_e \max\{d^{\alpha-1}x, x^\alpha\}. \quad (3)$$

By randomized rounding the fractional optimal solution with respect to $g_e(\cdot)$, we can easily derive the following parallel to Lemma 3.

Corollary 4. For uniform demands, $E[f_e(\hat{x}_e)] \leq \gamma g_e(x_e^*)$ for all e where $g_e(\cdot)$ is defined in (3).

The previous results only examine the expected value of the solution. We now show how to convert this into a result that holds with high probability.

Theorem 5. For uniform demands in which all d_i are equal, randomized rounding guarantees a γ -approximation in the expected value of the total energy cost, where γ is the constant in Lemma 3. Further, For any constant c , randomized rounding guarantees a $c\gamma$ -approximation with probability at least $1 - 1/c$.

Proof: The expected total cost after randomized rounding is $E[\sum_e f_e(\hat{x}_e)] = \sum_e E[f_e(\hat{x}_e)] \leq \gamma \sum_e g_e(x_e^*) \leq \gamma Opt$, where Opt is the integral solution to (P_1) . By Markov's inequality, the probability that a rounded solution is more than $c\gamma \cdot Opt$ is upper bounded by $1/c$. ■

B. Non-uniform Demands

We now prove an $O(\log^{\alpha-1} D)$ approximation for non-uniform demands with cost function $f_e(x) = \mu_e x^\alpha$, where $D = \max_i d_i$ is the maximum demand bandwidth. Note that in the application of interest, $f_e(\cdot)$ represents the energy curve for which the typical value of α is under 3. Hence, the logarithmic approximation ratio has a small exponent.

Theorem 6. For nonuniform demands, randomized rounding can be used to achieve a $O(\log^{\alpha-1} D)$ -approximation, where $D = \max_i d_i$.

Proof: We partition the demands into $\log D$ groups, where group $j \geq 0$ consists of demands whose d_i is in the range of $[2^j, 2^{j+1})$. We treat each group separately. For group j , we assume each demand requests bandwidth of exactly 2^{j+1} and invoke the randomized rounding algorithm for those demands. Let $x_e^{(j)*}$ be the load on link e due to the optimal fractional solution, and let $\hat{x}_e^{(j)}$ be the load after the rounding. Both $x_e^{(j)*}$ and $\hat{x}_e^{(j)}$ are calculated with respect to demand bandwidth rounded up to 2^{j+1} . Let $Opt^{(j)}$ be the optimal solution with respect to demands in group j , and Opt be the optimal solution with respect to demands in all groups. Note that both $Opt^{(j)}$ and Opt are with respect to actual demand bandwidth.

We have

$$\begin{aligned} \sum_e E \left[f_e \left(\sum_j \hat{x}_e^{(j)} \right) \right] &\leq \sum_e E \left[(\log D)^{\alpha-1} \sum_j f_e(\hat{x}_e^{(j)}) \right] \\ &\leq \sum_e (\log D)^{\alpha-1} \gamma \sum_j g_e^{(j)}(x_e^{(j)*}) \\ &\leq (\log D)^{\alpha-1} \gamma \sum_j 2^\alpha Opt^{(j)} \\ &\leq (\log D)^{\alpha-1} \gamma 2^\alpha Opt \end{aligned}$$

The first inequality is due to the convexity of $f_e(x) = \mu_e x^\alpha$, namely $f(\sum_{j=0}^{\log D} x_e^{(j)}) \leq (\log D)^{\alpha-1} \sum_j f(x_e^{(j)})$. In the second inequality $g_e^{(j)}(\cdot)$ refers to (3) for $d = 2^{j+1}$, the linearization of the function $f_e(\cdot)$ for demand bandwidth 2^{j+1} . The second inequality is due to Corollary 4. The third inequality holds since as before $\sum_e g_e(x_e^*)$ is a lower bound on the optimal integral solution and each demand bandwidth has been rounded up by a factor of at most 2. The last inequality holds due to the superadditive nature of $f_e(\cdot)$.

Finally, we note that $\sum_e E \left[f(\sum_j \hat{x}_e^{(j)}) \right]$ upperbounds the expected total cost since $\hat{x}_e^{(j)}$ is calculated based on bandwidth that is rounded up. This completes the proof. ■

IV. POLYNOMIAL FUNCTIONS WITH STARTUP COST

We now turn our attention to energy curves that are polynomials with a startup cost. These functions have the form $f_e(x) = \begin{cases} 0 & \text{if } x = 0 \\ \sigma_e + \mu_e x^\alpha & \text{if } x > 0 \end{cases}$. Note that for $\alpha \leq 1$, such a function is concave and Theorem 1 summarizes its approximability. When $\alpha > 1$, the function is neither convex nor concave, and therefore convex programming cannot obtain an optimal fractional solution to (P_1) .

In Section IV-A we provide two approximations. The first one is based on rounding a newly formed convex program defined in (P_2) . The resulting approximation ratio depends on the number of demands. The second one replaces the neither convex nor concave function $f_e(\cdot)$ with a convex function $h_e(\cdot)$ that ‘‘resembles’’ $f_e(\cdot)$, and then uses randomized rounding on the problem (P_1) with objective function $h_e(\cdot)$. The resulting ratio depends on the parameters σ_e and μ_e .

It remains a challenge to come up with approximations that are independent of the demands and cost functions and are small with respect to the network size. In Section IV-B, we first discuss why existing techniques for buy-at-bulk (in which $\alpha \leq 1$) can only guarantee approximation ratios polynomial in the size of the network when $\alpha > 1$. We then turn to the intrinsic hardness. In particular, for every $\alpha > 1$, we show there is a function $f_e(\cdot)$ uniform over all links e for which no algorithms and no techniques can guarantee an $O(\log^{1/4} N)$ approximation. Due to space limitation, we present a proof sketch.

A. Approximation results

1) Approximation with Respect to the Number of Demands:

The following is a natural formulation that handles polynomial functions with a startup cost.

$$\begin{aligned} (P_2) \quad & \min \quad \sum_e \sigma_e z_e + g_e(x_e) \\ & \text{subject to} \\ & x_e = \sum_i y_{i,e} \cdot d_i \\ & y_{i,e} \leq z_e \\ & y_{i,e}, z_e \in \{0, 1\} \\ & y_{i,e} : \text{flow conservation,} \end{aligned}$$

where in the integer formulation z_e represents whether or not the startup cost on link e is paid for, i.e. whether or not we route any demand on it. The second constraint enforces the condition that we cannot route any demand on a link unless its startup cost is paid for. In the objective function $g_e(x_e)$ is a linearization of $\mu_e x^\alpha$ as in Section III. For example, $g_e(x_e)$ is defined as in (3) for uniform demands. Again, the optimal integral solution to (P_2) is the same as to the objective of minimizing $\sum_e \sigma_e z_e + f_e(x_e)$, and its continuous relaxation is convex.

Theorem 7. *For uniform demands, randomized rounding of the optimal fractional solution to (P_2) guarantees an $(K + \gamma)$ -approximation to the optimal integral solution in expectation, where K is the number of demands and γ is the constant in Corollary 3.*

(We remark that this ratio can be better than the naive ratio that would be obtained by simply routing each demand along the minimum hop path since that solution could route all demands along a single edge whereas the optimum solution might route all demands along separate edges. Using this fact it is easy to construct examples where the cost of minimum hop is a factor $O(K^{\alpha-1})$ away from optimal.)

Proof: Let z^* , y^* and x^* be the optimal fractional solution and let \hat{z} , \hat{y} and \hat{x} be the solutions that we get from the rounding. From Lemma 3 we have that $E[g_e(\hat{x}_e)] \leq \gamma g_e(x_e^*)$ for some constant γ . It remains to relate $\sum_e \sigma_e \hat{z}_e^*$ and $\sum_e \sigma_e z_e^*$. We have,

$$\begin{aligned} E[\hat{z}_e] &= Pr(\hat{z}_e = 1) = 1 - Pr(\hat{y}_{i,e} = 0 \text{ for all } i) \\ &= 1 - \prod_i (1 - y_{i,e}^*) \leq \sum_i y_{i,e}^* \leq K z_e^*. \end{aligned}$$

The last inequality comes from the fact that in the fractional solution, each $y_{i,e}^*$ is constrained to be at most z_e^* . Putting everything together, we have that the rounded solution has expected value at most $(K + \gamma)$ times higher than the optimal solution. ■

The theorem above can be generalized to non-uniform demands. Combining the analysis for Theorems 6 and 7 we have the following.

Theorem 8. *For nonuniform demands, randomized rounding can be used to achieve a $O(K + \log^{\alpha-1} D)$ -approximation.*

2) *Approximation with Respect to σ_e and μ_e :* For the second approximation, we use a convex function $h_e(\cdot)$ in place of $f_e(\cdot)$ and randomized rounding on the optimal fractional solution with respect to $h_e(\cdot)$. To obtain an approximation ratio, we need to bound the difference between fractional and integral solutions, and the difference between $h_e(\cdot)$ and $f_e(\cdot)$. It is intuitive that we would like $h_e(\cdot)$ to be close to $f_e(\cdot)$.

The function $h_e(\cdot)$ starts with a line through the origin and switches to be $f_e(\cdot)$ at some point. Let $(s_e, f_e(s_e))$ be the point at which the line tangent to the curve $f_e(\cdot)$ goes through the origin. If $s_e \leq 1$ then $h_e(\cdot)$ begins with the straight line through origin to $(1, f_e(1))$ and continues on $f_e(\cdot)$, as shown in Figure 1 (left). Otherwise, $h_e(\cdot)$ begins with the tangent line

up to the tangent point $(s_e, f_e(s_e))$ and continues on $f_e(\cdot)$, as shown in Figure 1 (right).

More formally, let $s_e = (\sigma_e / ((\alpha - 1)\mu_e))^{1/\alpha}$. We define the parameter β_e and the function $h_e(x)$, for each edge e , as follows.

$$\beta_e = \begin{cases} \sigma_e + \mu_e & \text{if } s_e < 1, \\ \alpha\mu_e(\sigma_e / ((\alpha - 1)\mu_e))^{1-1/\alpha} & \text{if } s_e \geq 1. \end{cases}$$

$$h_e(x) = \begin{cases} \beta_e x & \text{if } x \in [0, \max(1, s_e)), \\ \sigma_e + \mu_e x^\alpha & \text{if } x \geq \max(1, s_e). \end{cases}$$

It can be observed that the function $h_e(x)$ is continuous, convex, and satisfies $h_e(x) \leq f_e(x)$, for all integral $x \geq 0$.

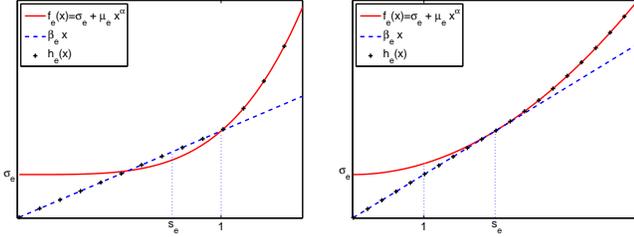


Fig. 1. $f_e(\cdot)$ and its approximation $h_e(\cdot)$. (Left) $s_e \leq 1$. (Right) $s_e > 1$.

Theorem 9. For unit demands, applying randomized rounding to the fractional solution obtained from the convex program (P_1) minimizing $\sum_e h_e(x_e)$, guarantees $E[f_e(\hat{x}_e)] \leq O(1 + (\sigma_e/\mu_e)^{1/\alpha}) \cdot h_e(x_e^*)$, for each link e .

Proof: Let us fix an edge e . We break the proof in three cases, $x_e^* \leq 1$, $x_e^* \geq \max(1, s_e)$, and $x_e^* \in (1, s_e)$. Case 1: $x_e^* \leq 1$. We partition the possible values of \hat{x}_e into the ranges $[0, 1)$, $[1, 2)$, $[2, 4)$, \dots . By the definition of expectation, we have

$$\begin{aligned} E[f_e(\hat{x}_e)] &\leq f_e(\hat{x}_e = 0)Pr[\hat{x}_e < 1] + \\ &\quad \sum_{j \geq 0} f_e(\hat{x}_e = 2^{j+1})Pr[\hat{x}_e \geq 2^j] \\ &\leq 0 + \sum_{j \geq 0} (\sigma_e + \mu_e (2^{j+1})^\alpha) Pr[\hat{x}_e \geq 2^j] \\ &\leq \beta_e x_e^* \frac{\sigma_e + \mu_e}{\beta_e} \sum_{j \geq 0} \frac{2^{\alpha(j+1)} x_e^{*2^j}}{x_e^* e^{x_e^{*2^j}}} \left(\frac{e}{2^j}\right)^{2^j} \end{aligned}$$

where the last inequality follows from a Chernoff bound. The sum was shown in the proof of Lemma 3 to be bounded by a constant γ_1 . If $s_e < 1$, then $\beta_e = \sigma_e + \mu_e$ and hence $\frac{\sigma_e + \mu_e}{\beta_e} = 1$. Otherwise, $s_e \geq 1$, and then $\beta_e = \Theta(\mu_e^{1/\alpha} \sigma_e^{1-1/\alpha})$. Since $s_e = \Theta((\sigma_e/\mu_e)^{1/\alpha})$, then $\mu_e/\sigma_e = O(1)$. In either case, we get

$$\frac{\sigma_e + \mu_e}{\beta_e} = O(1 + (\sigma_e/\mu_e)^{1/\alpha}), \quad (4)$$

and then, $E[f_e(\hat{x}_e)] \leq O(1 + (\sigma_e/\mu_e)^{1/\alpha}) \cdot h_e(x_e^*)$.

Case 2: $x_e^* \geq \max(1, s_e)$. In this case we have $E[f_e(\hat{x}_e)] \leq \gamma_2 \cdot h_e(x_e^*)$, from a proof identical to case 2 in Lemma 3.

Case 3: $x_e^* \in (1, s_e)$. Note that this case can only occur if $s_e \geq 1$. We partition the possible values of \hat{x}_e into the ranges $[0, x_e^*)$, $[x_e^*, 2x_e^*)$, $[2x_e^*, 4x_e^*)$, \dots . By the definition of expectation, we have

$$\begin{aligned} &E[f_e(\hat{x}_e)] \\ &\leq f_e(\hat{x}_e = x_e^*)Pr[\hat{x}_e \geq 0] + \\ &\quad \sum_{j \geq 0} f_e(\hat{x}_e = 2^{j+1}x_e^*)Pr[\hat{x}_e \geq 2^jx_e^*] \\ &\leq \beta_e x_e^* \left(\frac{\sigma_e + \mu_e (x_e^*)^\alpha}{\beta_e x_e^*} + \right. \\ &\quad \left. \sum_{j \geq 0} \frac{\sigma_e + \mu_e (2^{j+1}x_e^*)^\alpha}{\beta_e x_e^*} Pr[\hat{x}_e \geq 2^jx_e^*] \right) \\ &\leq h_e(x_e^*) \frac{\sigma_e + \mu_e}{\beta_e} \cdot \left(1 + \sum_{j \geq 0} 2^{\alpha(j+1)} Pr[\hat{x}_e \geq 2^jx_e^*] \right) \end{aligned}$$

where the third inequality follows from the fact that $\frac{\sigma_e + \mu_e x^\alpha}{\beta_e x}$ is non-increasing for $x \in (1, s_e)$. From Eq. 4, $\frac{\sigma_e + \mu_e}{\beta_e} = O(1 + (\sigma_e/\mu_e)^{1/\alpha})$, and the other factor of $h_e(x_e^*)$ was shown in the proof of Lemma 3 to be bounded by a constant γ_2 . Therefore, $E[f_e(\hat{x}_e)] \leq O(1 + (\sigma_e/\mu_e)^{1/\alpha}) \cdot h_e(x_e^*)$. ■

The approximation above also applies to uniform demands. Similar to Theorem 6, we also have the following for nonuniform demands.

Theorem 10. For polynomial functions with startup costs, randomized rounding can be used to achieve a $O(1 + (\max_e \{\sigma_e/\mu_e\})^{1/\alpha}) (\log D)^{\alpha-1}$ -approximation, where $D = \max_i d_i$.

B. Hardness of Approximation

The results of the previous section work well when σ_e or K are small but give less good bounds when these parameters are large. Recall from Theorem 1 that we have a range of techniques that guarantee a poly-logarithmic approximation for the Buy-at-Bulk problem (i.e. the problem where the cost functions are subadditive, e.g, when $\alpha \leq 1$). We briefly comment on why these techniques cannot produce approximation ratios better than polynomial in the network size when $\alpha > 1$. In fact all these techniques fail on an example similar to the one at the beginning of Section III. Let us revisit the example: the network has 2 nodes u and v , m parallel links, and m unit demands between u and v . Suppose the cost function is $f_e(x) = m + x^2$ for $x > 0$ for all e . It is easy to compute that the optimal solution routes over \sqrt{m} links each carrying \sqrt{m} demands. The optimal cost is $\sqrt{m}(m + \sqrt{m}^2) = 2m\sqrt{m}$.

One technique for uniform-cost buy-at-bulk is due to Awerbuch and Azar [10] and always returns a solution in which the routes form a tree. (In particular, the tree is taken from a distribution that approximates the underlying distance metric.) If the solution in the above example is restricted to a tree, i.e. a single link, the cost would be $f(m) = m + m^2$, which is $\Omega(\sqrt{m})$ times the optimal. Therefore, restricting a solution to a tree sets a lower bound of \sqrt{N} in approximation ratio where N is the network size.

The second technique of buy-at-bulk involves rounding a linear relaxation of a problem formulation in the spirit of (P_2) , e.g. [23], [18]. The optimal fractional solution to the above example sets $z_e = 1/m$ for all e and therefore the total is $m(1/m+1) = 1+m$. This yields an integrality gap of $\Theta(\sqrt{m})$. Hence, rounding cannot be expected to give better than \sqrt{N} approximations.

The third approach by Charikar and Karagiozova [17] does not always produce a tree and their analysis does not compare against the optimal fractional solution. Their solution operates by first ordering the demands in a random order and then for each m greedily routing m/i times the i th demand, each along the path that incurs the least extra cost. In our example, it is less expensive for a demand to use a link with a load in $[1, m/2)$ than to start a new link. The particular scaling of the Charikar-Karagiozova algorithm routes $m + m/2 + m/3 + \dots \approx m \log m$ units of demands and therefore uses $\log m$ links. Therefore, the total cost is at least $m^2/\log m$, which again creates a polynomial gap from the optimal.

So far we have shown that known techniques cannot give an approximation ratio better than polynomial. It is an intriguing open problem whether or not there exists a polylogarithmic approximation ratio. However, we now show that we cannot hope for better than a polylogarithmic ratio, since we have the following intrinsic hardness result.

Theorem 11. *For any $\alpha > 1$, there is a uniform polynomial cost function with startup cost such that no algorithm can guarantee an $O(\log^{1/4} N)$ approximation unless $NP \subseteq ZPTIME(n^{\text{polylog} n})$.*

Recall that $ZPTIME(n^{\text{polylog} n})$ is the class of languages for which there is a randomized algorithm that always gives the correct answer and whose expected running time is $n^{\text{polylog} n}$. The proof of the theorem is motivated by the hardness for buy-at-bulk [6], [8]. The construction of the hardness reduction and its analysis are somewhat lengthy. Hence, instead of presenting a self-contained proof here, we give a high-level sketch.

We start with a 3CNF(5) formula ϕ which is a boolean formula in conjunctive normal form in which each clause contains exactly 3 literals and each variable appears in exactly 5 clauses. The Probabilistically Checkable Proof (PCP) theorem [9], [19] implies that there is a constant ε such that it is NP-hard to distinguish between the case where ϕ is satisfiable and the case where at most a $(1 - \varepsilon)$ -fraction of the clauses can be simultaneously satisfied.

From ϕ we can use results of [6], [8] to construct a routing instance such that if ϕ is a yes-instance, namely more than $(1 - \varepsilon)$ -fraction of the clauses can be satisfied, then the optimal routing cost is at most a low value of ℓ . Otherwise, ϕ is a no-instance, and the optimal routing cost is at least a high value of h with high probability. If we should be able to approximate the routing instance to better than h/ℓ , we would then be able to tell a yes-instance from a no-instance. This contradicts the PCP theorem. However, our reduction is not polynomial. In fact the size of the routing instance is $n^{\text{polylog} n}$ and therefore the complexity assumption of $NP \not\subseteq ZPTIME(n^{\text{polylog} n})$.

V. EXPERIMENTAL RESULTS

In this section, we provide the detailed experimental findings. We associate cost functions like those previously presented to the links of real networks, implement the approximation algorithms presented in Sections III and IV, and compare the approximate solutions against both the optimal and the straight-forward shortest-paths solution. The reason for comparing against the latter is to show that routing without energy in mind can be wasteful. As we shall see, we observe a consistent savings of 10% or more over shortest-paths. This gives initial evidence that a non-negligible percentage of energy saving could come from global network planning such as routing.

We obtain the optimal integral solutions by solving the relevant integer programs use CPLEX solver [1]. For our approximation algorithms, we use the CVX solver [2] to obtain the optimal fractional solutions before applying randomized rounding. Most of our experiments are conducted on the Abilene Research network which consists of 10 nodes and 13 links, and the NSF Network which consists of 14 nodes and 20 links. See Figure 2. We also test scalability on larger networks.

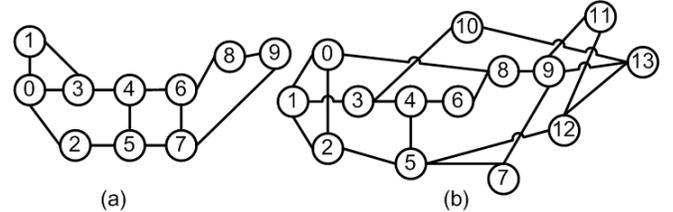


Fig. 2. (a) Abilene Research network and (b) NSF network

A. Polynomial cost function without startup: $f_e(x) = \mu_e x^\alpha$

We use a quadratic function $f_e(x) = x^2$ for our experiments. For each network, we perform the routing algorithm with different number of demands, where the number ranges from twice the number of nodes to six times the number of nodes. The source and sink nodes of each demand are chosen uniformly at random. We concentrate on unit demands. For each routing instance, we compare 4 values of interest, the optimal integral solution from CPLEX, the optimal fractional solution from CVX, the rounded integral solution and the short-paths solution. The four curves in Figure 3 (a) and (b) correspond to the ratio of these 4 values all normalized by the optimal integral. We observe the following.

i) The optimal fractional values are very close to the integral optimal. The difference is at most 0.84% in the Abilene Research network and at most 1.2% in the NSF network. This suggests that optimal fractional solution (which is polynomially obtainable) can be a good lower bound in the absence of optimal integral solution (which is NP hard).

ii) The randomized rounding solutions are within 4% of the integral optimal in the Abilene Research network and within 0.5% in the NSF network. This suggests that randomized

rounding performs even better in practice than the approximation ratio analyzed in Lemma 3.

iii) The randomized rounding solutions are consistently at least 10% better than the shortest-path solutions.

To explain these findings, we examine the *link load*, which is the total demand flow going through e . We observe that the maximum link load as a result of the integral optimal, fractional optimal and randomized rounding are quite close to one another. However, the maximum load of the shortest-path solutions is often significantly higher, as shortest-path routing does not intend to balance the link load and therefore incurs high cost.

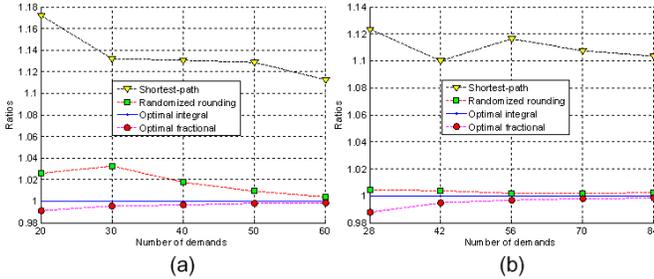


Fig. 3. From top to bottom, values due to shortest-paths, randomized rounding, integral optimal and fractional optimal, all normalized by the integral optimal. (a) Abilene Research network and (b) NSF network

B. Polynomial function with startup: $f_e(x) = \sigma_e + \mu_e x^\alpha$

We use $f_e(x) = \sigma + x^2$ as the cost function, where $\sigma \in \{4, 16, 64, 256, 1024\}$. Again the number of demands varies from twice the number of nodes to six times the number of nodes. We compare a number of routing strategies here. Two of the strategies correspond to the $O(K)$ -approximation and $O(\sqrt{\sigma})$ -approximation, as shown in Theorems 7 and 9. Not surprisingly the $O(\sqrt{\sigma})$ -approximation performs poorly for large σ , as the approximate function $g_e(\cdot)$ deviates significantly from $f_e(\cdot)$ for large σ . The $O(K)$ approximation is also less than satisfactory for large σ . The difficulty for the large startup cost is that a large σ encourages aggregating traffic to minimize the number of *active* links, namely those carrying non-zero traffic. On the other hand, the convex nature of x^2 encourages load balancing traffic to avoid paying quadratic cost on high loads. The balance between these contradicting objectives is challenging.

We offer a heuristic Greedy_ActiveLinks (see Figure 4) that helps to shrink the set of active links. Initially, we assume every link is active, namely the active link set $E' = E$. We minimize $\sum_{e \in E'} x_e^2$ to a value, say S' . (We know from the previous findings that randomized rounding performs extremely well for $\min \sum_{e \in E'} x_e^2$.) The total cost $\sum_{e \in E} f_e(x_e)$ is therefore $S' + \sigma|E'|$. Note that the routes may not use every link E , but we nevertheless pay σ for all. During each subsequent iteration, we aim to remove one link from the active set so that total cost is reduced. Again, for the current active set E' we minimize $\sum_{e \in E'} x_e^2$. The process

```

Greedy_ActiveLinks(  $G = (V, E)$  ):
Let  $E' \leftarrow E$ ,  $E'' \leftarrow \emptyset$ , and  $S \leftarrow \infty$ 
while (true):
  For  $e \in E'$  begin:
    1. If  $E'' = \emptyset$ , then  $E'' \leftarrow E'$ , o.w.  $E'' \leftarrow E' / \{e\}$ 
    2. Let  $S' = \min \sum_{e \in E''} x_e^2$ 
       via randomized rounding on  $G(V, E'')$ 
    3. If  $S' + \sigma|E''| < S$ ,
       then  $E' \leftarrow E''$ ,  $S \leftarrow S' + \sigma|E''|$  and break
  end
  If no improvement for all  $e \in E'$ , break
end

```

Fig. 4. Pseudocode for the Greedy_ActiveLinks heuristic.

stops when we can no longer shrink the active set, either due to disconnectivity or increasing cost.

We refer to the solution from the above Greedy_ActiveLinks heuristic as *Greedy_RR*, where RR refers to randomized rounding. From Table I, we observe that Greedy_RR improves a great deal over the $O(K)$ and $O(\sqrt{\sigma})$ approximations for large σ ; whereas for small σ , the $O(K)$ and $O(\sqrt{\sigma})$ approximations continue to have an advantage. We therefore have a *Combined* strategy which is to run all three strategies for each instance and keep the best.

Again, we compare against the shortest-paths solution (*SP*). Like the case in which $\sigma = 0$, shortest-paths is worse off than the $O(K)$ and $O(\sqrt{\sigma})$ approximation for small σ . For large σ , all three strategies perform poorly. If we combine the greedy heuristic combine with shortest-paths, specifically by running shortest paths in line 2 of Greedy_ActiveLinks, the improvement of the resulting solution *Greedy_SP* is less than satisfactory.

Due to space consideration we only present numbers for the NSF network in Table I. Again, all the values are normalized by the optimal integral values.

NSF Network		Startup cost σ				
Solutions	dem.	4	16	64	256	1024
<i>SP</i>	28	1.107	1.053	1.137	1.314	1.389
<i>Greedy_SP</i>	28	1.107	1.053	1.123	1.259	1.247
<i>K</i> -approx	28	1.017	1.120	1.270	1.434	1.516
$\sqrt{\sigma}$ -approx	28	1.005	1.022	1.142	1.292	1.382
<i>Greedy_RR</i>	28	1.129	1.071	1.071	1.133	1.099
<i>Combined</i>	28	1.005	1.022	1.071	1.133	1.099
<i>SP</i>	56	1.091	1.074	1.046	1.152	1.380
<i>Greedy_SP</i>	56	1.091	1.074	1.046	1.130	1.252
<i>K</i> -approx	56	1.001	1.004	1.143	1.271	1.479
$\sqrt{\sigma}$ -approx	56	1.001	1.005	1.020	1.146	1.371
<i>Greedy_RR</i>	56	1.167	1.124	1.056	1.064	1.162
<i>Combined</i>	56	1.001	1.004	1.020	1.064	1.162
<i>SP</i>	84	1.077	1.065	1.038	1.103	1.249
<i>Greedy_SP</i>	84	1.077	1.064	1.038	1.103	1.212
<i>K</i> -approx	84	1.001	1.001	1.012	1.212	1.374
$\sqrt{\sigma}$ -approx	84	1.002	1.002	1.003	1.084	1.260
<i>Greedy_RR</i>	84	1.175	1.141	1.074	1.068	1.091
<i>Combined</i>	84	1.001	1.001	1.003	1.068	1.091

TABLE I
NSF NETWORK.

C. Running time and larger networks

The average running time for the CVX solver is around 2-3 seconds for obtaining optimal fractional solutions to all the instances presented so far. The CPLEX solver is also fast for obtaining the integral optimal to all the instances with small startup values, namely $\sigma = 0, 4, \dots, 64$. The running times vary from 30 seconds to 3 minutes. However, for larger startup cost $\sigma = 256, 1024$, CPLEX takes significantly longer. For example when $\sigma = 1024$ and the number of demand pairs is 6 times the number of nodes, it took CPLEX longer than 17 hours to get a solution with relative error within 2.1% on the NSF network. For larger networks with at least 25 nodes, CPLEX has trouble even for $\sigma = 0$.

We repeated our experiments on random sparse networks with 100 nodes and expected node degree of 4. Although we cannot obtain optimal integral solutions, our findings of the performance of other algorithms and heuristics are consistent with our findings on the Abilene Research network and the NSF network.

VI. CONCLUSION

In this paper we consider a min-cost integer routing problem where the cost function represents the energy curve of a network element. Subadditive cost functions are well studied. We focus on the less-studied polynomial functions and polynomials with a startup cost. The problem is interesting for two reasons. First, the cost function closely models the energy consumption of some network elements and network-wide optimization is a well-motivated but under-explored direction for energy minimization. Second, it brings light to a challenging combinatorial optimization problem. We have presented positive and negative results for polynomial functions and polynomial functions with startup cost. For the latter, techniques to accomplish better-than-polynomial approximation ratios independent of demands and cost function remains a challenging problem.

REFERENCES

- [1] <http://www.ilog.com/products/cplex>.
- [2] <http://www.stanford.edu/~boyd/cvx/>.
- [3] Enhanced intel speedstep technology for the intel pentium m processor. Intel White Paper 301170-001, 2004.
- [4] In *Proceedings of the Vision and Roadmap Workshop on Routing Telecom and Data Centers Toward Efficient Energy Use*, October 2008.
- [5] Smart2020: Enabling the low carbon economy in the information age. www.smart2020.org, 2008.
- [6] M. Andrews. Hardness of buy-at-bulk network design. In *Proceedings of the 45th Annual Symposium on Foundations of Computer Science*, pages 115 – 124, Rome, Italy, October 2004.
- [7] M. Andrews, A. Fernandez Anta, L. Zhang, and W. Zhao. Routing and scheduling for energy and delay minimization in the powerdown model. In *Proceedings of IEEE INFOCOM 2010*, San Diego, CA, 2010.
- [8] M. Andrews and L. Zhang. Bounds on fiber minimization in optical networks with fixed fiber capacity. In *Proceedings of IEEE INFOCOM '05*, Miami, FL, March 2005.
- [9] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998.
- [10] B. Awerbuch and Y. Azar. Buy-at-bulk network design. In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science*, pages 542 – 547, Miami Beach, FL, October 1997.
- [11] Nikhil Bansal, Tracy Kimbrel, and Kirk Pruhs. Speed scaling to manage energy and temperature. *J. ACM*, 54(1), 2007.
- [12] L.A. Barroso and U. Hözlze. The case for energy-proportional computing. *IEEE Computer*, 40(12):33, 2007.
- [13] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- [14] David Brooks, Pradip Bose, Stanley Schuster, Hans M. Jacobson, Prabhakar Kudva, Alper Buyuktosunoglu, John-David Wellman, Victor V. Zyuban, Manish Gupta, and Peter W. Cook. Power-aware microarchitecture: Design and modeling challenges for next-generation microprocessors. *IEEE Micro*, 20(6):26–44, 2000.
- [15] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsang, and S. Wright. Power awareness in network design and routing. *INFOCOM 2008. The 27th Conference on Computer Communications*. IEEE, April 2008.
- [16] Ho-Leung Chan, Wun-Tat Chan, Tak Wah Lam, Lap-Kei Lee, Kin-Sum Mak, and Prudence W. H. Wong. Energy efficient online deadline scheduling. In Nikhil Bansal, Kirk Pruhs, and Clifford Stein, editors, *SODA*, pages 795–804. SIAM, 2007.
- [17] Moses Charikar and Adriana Karagiozova. On non-uniform multicommodity buy-at-bulk network design. In Harold N. Gabow and Ronald Fagin, editors, *STOC*, pages 176–182. ACM, 2005.
- [18] Chandra Chekuri, Mohammad Taghi Hajiaghayi, Guy Kortsarz, and Mohammad R. Salavatipour. Approximation algorithms for non-uniform buy-at-bulk network design. In *FOCS*, pages 677–686. IEEE Computer Society, 2006.
- [19] Irit Dinur. The PCP theorem by gap amplification. *J. ACM*, 54(3):12, 2007.
- [20] IEEE P802.3az Energy Efficient Ethernet. Task force public area. <http://grouper.ieee.org/groups/802/3/az/public/index.html>, 2008.
- [21] Matthew Garrett. Powering down. *Commun. ACM*, 51(9):42–46, 2008.
- [22] George Ginis. Low-power modes for adsl2 and adsl2+. White Paper, Broadband Communications Group, Texas Instruments, Jan. 2005.
- [23] S. Guha, A. Meyerson, and K. Munagala. A constant factor approximation for the single sink edge installation problems. In *STOC '01: Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 383–388, New York, NY, USA, 2001. ACM.
- [24] Chamara Gunaratne, Kenneth J. Christensen, Bruce Nordman, and Stephen Suen. Reducing the energy consumption of ethernet with adaptive link rate (alr). *IEEE Trans. Computers*, 57(4):448–461, 2008.
- [25] Sandy Irani, Rajesh K. Gupta, and Sandeep K. Shukla. Competitive analysis of dynamic power management strategies for systems with multiple power savings states. In *DATE*, pages 117–123. IEEE Computer Society, 2002.
- [26] Sandy Irani and Kirk Pruhs. Algorithmic problems in power management. *SIGACT News*, 36(2):63–76, 2005.
- [27] Sandy Irani, Sandeep K. Shukla, and Rajesh Gupta. Algorithms for power savings. *ACM Transactions on Algorithms*, 3(4), 2007.
- [28] Sandy Irani, Sandeep K. Shukla, and Rajesh K. Gupta. Online strategies for dynamic power management in systems with multiple power-saving states. *ACM Trans. Embedded Comput. Syst.*, 2(3):325–346, 2003.
- [29] Christine E. Jones, Krishna M. Sivalingam, Prathima Agrawal, and Jyh-Cheng Chen. A survey of energy efficient network protocols for wireless networks. *Wireless Networks*, 7(4):343–358, 2001.
- [30] Patrick Kurp. Green computing. *Commun. ACM*, 51(10):11–13, 2008.
- [31] Minming Li, Becky Jie Liu, and Frances F. Yao. Min-energy voltage allocation for tree-structured tasks. In Lusheng Wang, editor, *COCOON*, volume 3595 of *Lecture Notes in Computer Science*, pages 283–296. Springer, 2005.
- [32] Michael Mitzenmacher and Eli Upfal. *Probability and Computing : Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, January 2005.
- [33] Sergiu Nedevschi, Lucian Popa, Gianluca Iannaccone, Sylvia Ratnasamy, and David Wetherall. Reducing network energy consumption via sleeping and rate-adaptation. In Jon Crowcroft and Michael Dahlin, editors, *NSDI*, pages 323–336. USENIX Association, 2008.
- [34] Prachi Patel-Predd. Energy-efficient ethernet. *IEEE Spectrum*, page 13, May 2008.
- [35] P. Raghavan and C.D. Thompson. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7:365 – 374, 1991.
- [36] Adam Wierman, Lachlan L. H. Andrew, and Ao Tang. Power-aware speed scaling in processor sharing systems. In *INFOCOM*, 2009.
- [37] F. Frances Yao, Alan J. Demers, and Scott Shenker. A scheduling model for reduced cpu energy. In *FOCS*, pages 374–382, 1995.