McMaster University

Advanced Optimization Laboratory

Title:

Packing Trees in Communication Networks

Authors:

Mohamed Saad, Tamás Terlaky, Anthony Vannelli and Hu Zhang

AdvOl-Report No. 2005/13

July 2005, Hamilton, Ontario, Canada

Packing Trees in Communication Networks*

Mohamed Saad[†] Tamás Terlaky[‡] Anthony Vannelli[§]
Hu Zhang[‡]

Abstract

Given an undirected edge-capacitated graph and given subset of vertices, we consider the problem of selecting a maximum (weighted) set of Steiner trees, each tree spanning a subset of vertices without violating the capacity constraints. This problem is motivated by applications in multicast communication networks. We give an integer linear programming (ILP) formulation for the problem, and observe that its linear programming (LP-) relaxation is a fractional packing problem with exponentially many variables and a block (sub-)problem that cannot be solved in polynomial time. To this end, we take an r-approximate block solver to develop a $(1-\varepsilon)/r$ approximation algorithm for the LP-relaxation. The algorithm has a polynomial coordination complexity for any $\varepsilon \in (0,1)$. To the best of our knowledge, this is the first approximation result for fractional packing problems with only approximate block solvers and a coordination complexity that is polynomial in the input size. This leads also to an approximation algorithm for the underlying tree packing problem. Finally, we extend our results to an important multicast routing and wavelength assignment problem in optical networks, where each Steiner tree is also to be assigned one of a limited set of given wavelengths, so that trees crossing the same fiber are assigned different wavelengths.

*Research supported by a MITACS grant for all the authors, the NSERC Discovery Grant #5-48923 for the second and fourth author, NSERC Grant #15296 for the third author, and the Canada Research Chair Program for the second author.

[†]Department of Electrical and Computer Engineering, University of Toronto, Canada, `saadme@comm.utoronto.ca`.

[‡]Department of Computing and Software, McMaster University, Canada, `{terlaky,zhanghu}@mcmaster.ca`.

[§]Department of Electrical and Computer Engineering, University of Waterloo, Canada, `a.vannelli@ece.uwaterloo.ca`.

1 Introduction

Multicast is an efficient approach to deliver data from a source to multiple destinations over a communication network. This approach is motivated by emerging telecommunication applications, e.g., video-conferencing, streaming video and distributed computing. In particular, a multicast session is established by finding a Steiner tree in the network that connects the multicast source with all the multicast destinations.

In this paper we address the following *Steiner tree packing* problem, that is fundamental in multicast communications. We are given a communication network represented by an undirected graph, a capacity associated with every edge in the graph, and a set of multicast *requests* (each defined by a subset of vertices to be connected, called *terminals*). A feasible solution to this problem is a set of Steiner trees, each Steiner tree spanning a multicast request, such that the number of Steiner trees crossing the same edge is bounded by the capacity of that edge. The goal is to maximize the total *profit/throughput* (the weighted sum of the successfully routed requests). It is worth noting that some requests may not be successfully routed due to the edge capacity. This problem arises in the communication network that provides multicast communication service to multiple groups of users in order to realize the routing that attains the maximum global profit for the whole network with limited bandwidth resources.

A special case in which the same request is spanned by the maximum number of edge-disjoint Steiner trees was studied in [14]. The authors presented $4/|S|$-asymptotic approximation algorithm, where S is the terminal set to be connected. This problem was further studied in [16] and a polynomial time algorithm was proposed to find $\lfloor \lambda_S(G)/26 \rfloor$ edge-disjoint Steiner trees, where $\lambda_S(G)$ is the size of a minimum S-cut in G. Another generalization is to find the maximum collection of Steiner forests spanning different requests [17]. There are also many applications in this category (see [16]). A related problem of realizing all given multicast requests as to minimize the maximum edge congestion was studied from theoretical and experimental aspects in [3, 6, 7, 13, 18, 25]. This is essentially equivalent to a routing problem in VLSI design [21]. Another related problem is that of realizing all given multicast requests at the minimum cost. This problem was studied in [5] and [15] for the special case of all Steiner trees connecting the same set of vertices, and in [24] for the general case where for each Steiner tree a different set of vertices is given.

We show that the relaxation of the Steiner packing problem is a *fractional packing problem* in Section 2. Fractional packing problems have attracted considerable attention in the literature [9, 19, 27]. In general, a block solver is called to play a similar role to the separation oracle in the ellipsoid methods in [11]. The approximation algorithm in [9] is only for the case that the block problem is required polynomial time solvable. In addition, the approximation algorithms in [19, 27] have coordination complexity that depends on the input data, and are thus not polynomial in the input size. A problem related to fractional packing is the convex min-max resource-sharing problem, which is studied in [10, 26]. If the block problem is $\mathcal{NP}$-hard, an approximation algorithm is designed in [12] with polynomial coordination complexity. It is the first polynomial time approximation result for the convex min-max resource-sharing problem.

To date, we are not aware of any approximation results for the Steiner tree packing problem in its full generality (where for each Steiner tree is required to connect a set of vertices). Furthermore, we are not aware of any approximation algorithm for fractional packing problems with coordination complexity polynomial in the input size while the block problem is $\mathcal{NP}$-hard.

The contribution of this paper can be summarized as follows. We formulate the Steiner tree packing problem in its full generality as an ILP, and observe that its LP-relaxation is a fractional packing problem with exponentially many variables and a block problem that cannot be solved in polynomial time. We thus develop a $(1-\varepsilon)/r$-approximation algorithm for fractional packing problems with polynomial coordination complexity, each iteration calling an r-approximate block solver, for $r \geq 1$ and any given $\varepsilon \in (0,1)$. This is the first result for fractional packing problems with only approximate block solvers and a coordination complexity strictly polynomial in input size. In fact, the coordination complexity of our algorithm is exactly the same as in [9] where the block problem is required to be polynomial time solvable. Then we present an algorithm for the Steiner tree packing problem and also apply our approximation algorithm for integer packing problems to establish a method to directly find a feasible solution. We extend our results to an important multicast routing and wavelength assignment problem in optical networks, where each Steiner tree is also to be assigned one of a limited set of given wavelengths, so that trees crossing the same fiber are assigned different wavelengths.

The remainder of this paper is organized as follows. In Section 2 we give an ILP formulation of the Steiner tree packing problem. Then we present and analyze the approximation algorithm for fractional packing problems in Section 3 and use it to develop an approximation algorithm for the integer Steiner tree packing problem in Section 4. The approach to directly find integer approximate solutions is discussed in Section 5. The multicast routing and wavelength assignment problem in optical networks is studied in Section 6. Finally, Section 7 concludes the paper.

2 Mathematical Programming Formulation

We are given an undirected graph $G = (V, E)$ representing the input multicast communication network, and a set of multicast requests $S_1, \ldots, S_K \subseteq V$ to be routed by Steiner trees. Each edge $e_i \in E$ is associated with a capacity c_i indicating the bandwidth of the corresponding cable. Denote by $\mathcal{T}_k$ the set of all Steiner trees spanning S_k for request S_k, $k \in \{1, \ldots, K\}$. The number of trees $|\mathcal{T}_k|$ may be exponentially large. Furthermore, we define an indicator variable $x_k(T)$ for each tree as follows:

$$x_k(T) = \begin{cases} 1, & \text{if } T \in \mathcal{T}_k \text{ is selected for routing } S_k; \\ 0, & \text{otherwise.} \end{cases}$$

In addition, each request S_k is associated with a weight w_k to measure its importance in the given multicast communication network. Therefore, the Steiner tree packing problem can be cast as the following ILP:

$$\begin{array}{rll} \max & \sum_{k=1}^{K} w_k \sum_{T \in \mathcal{T}_k} x_k(T) & \\ \text{s.t.} & \sum_{k=1}^{K} \sum_{T \in \mathcal{T}_k \& e_i \in T} x_k(T) \leq c_i, & \forall e_i \in E; \\ & \sum_{T \in \mathcal{T}_k} x_k(T) \leq 1, & k = 1, \ldots, K; \\ & x_k(T) \in \{0, 1\}, & \forall T \,\&\, k = 1, \ldots, K. \end{array} \tag{1}$$

The first set of constraints in (1) means that the congestion of each edge is bounded by the edge capacity. The second set of constraints shows that at most one tree is selected to realize the routing for each request. It is possible that in a feasible solution, no tree is chosen for some requests, i.e., some requests may not be realized, due to the edge capacity constraints. In fact our goal is to maximize the weighted sum of successful routings according to the given importance of the multicast requests.

The special cases of the Steiner tree problem studied in this work have been shown $\mathcal{APX}$-hard [14, 16, 17], so is our underlying problem. Indeed the integrality of the variables in (1) implies the $\mathcal{NP}$-hardness of the Steiner tree packing problem. Furthermore, there may be exponentially many variables in (1). Thus many exact algorithms such as standard interior point methods can not be applied to solve its LP-relaxation. The LP-relaxation of (1) may be solved by the volumetric-center [1] or the ellipsoid methods with separation oracle [11]. However, those approaches will lead to a large amount of running time. In addition, different from previous models for unicast routing, here the routing is realized by trees instead of paths, which increases the hardness of the problem. We show in Section 4 that the block problem of the Steiner tree packing problem is the *minimum Steiner tree problem* in graphs, which is $\mathcal{APX}$-hard [2, 4].

As usual, we first solve the LP-relaxation of (1), and then apply rounding techniques to obtain a feasible solution. We call the linear relaxation of the Steiner tree packing problem as the *fractional Steiner tree packing problem*, and its solution as the *fractional solution* to the Steiner tree packing problem. The LP-relaxations of (1) is in fact a fractional packing problem [9, 19, 27]. However, the approximation algorithm in [9] is only for the case that the block problem is polynomial time solvable. Unfortunately, it is not the case for the Steiner tree packing problem as it is the minimum Steiner tree problem. In addition, the approximation algorithms in [19, 27] both lead to complexity bounds that depend on the input data, and only result in pseudo polynomial time approximation algorithms. Thus, we need to study approximation algorithms for fractional packing problems with approximate block solvers and input data independent complexity.

3 Approximation Algorithm for Fractional Packing Problems

In this section, we develop an approximation algorithm for fractional packing problems based on the approach in [9]. Our algorithm allows that the block problem can only be approximately solved. Our complexity is still strictly polynomial in the input size, which is superior to the methods in [19, 27].

We consider the following fractional packing problem:

$$\max\{c^T x | Ax \leq b, x \geq 0\}. \tag{2}$$

Here A is a $m \times n$ positive matrix, and $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$ are positive vectors. In addition, we assume that the (i, j)-th entry $A_{i,j} \leq b_i$ for all i and j. The corresponding dual program is:

$$\min\{b^T y | A^T y \geq c, y \geq 0\}. \tag{3}$$

Similar to the strategies in [9, 10, 12, 19, 26, 27], an (approximate) block solver is needed, which is equivalent to the separation oracle for the ellipsoid methods. For a given $y \in \mathbb{R}^m$, the block problem is to find a column index q that $(A_q)^T y/c_q = \min_j (A_j)^T y/c_j$, In our algorithm, we assume that we are given the following approximate block solver $ABS(y)$ that finds a column index q that $(A_q)^T y/c_q \le r \min_j (A_j)^T y/c_j$, where $r \ge 1$ is the approximation ratio of the block solver. It is worth noting that in [9] it is required that $r = 1$, i.e., the block problem is polynomial time solvable.

```
δ = 1 − √(1 − ε), u = (1 + δ)((1 + δ)m)^{−1/δ};
k = 0, x^k = 0, f^k = 0, y_i^k = u/b_i, D^k = um;
while D^k < 1 do {iteration}
   k = k + 1;
   call ABS(y^{k−1}) to find a column index q;
   p = arg min_i b_i/A_{i,q};
   x_q^k = x_q^{k−1} + b_q/A_{p,q};
   f^k = f^{k−1} + c_q b_p/A_{p,q};
   y_i^k = y_i^{k−1}[1 + δ(b_p/A_{p,q})/(b_i/A_{i,q})];
   D^k = b^T y^k;
end do
```

Table 1: Approximation algorithm for fractional packing problems.

Our algorithm is an iterative method. We first maintain a pair of a primal feasible solution x to the fractional packing problem (2) and a dual infeasible solution y. At each iteration, based on the current dual solution y, the algorithm calls the approximate block solver once. Then the algorithm increases the component of the primal solution x corresponding to the returned column index by a certain amount and multiples the dual solution y by a factor larger than 1. This iterative procedure does not stop until the dual objective value is more than 1 (though the dual solution may be still infeasible). The algorithm is shown in Table 1. In the algorithm, D^k is in fact the dual objective value for the dual vector y_k at the k-th iteration, though it can be infeasible. Let $\mathcal{OPT}$ denote the optimum dual value (also the optimum objective value of the primal program according to the duality relation). In addition, we assume that the algorithm stops at the t-th iteration. We have the following bound:

Lemma 3.1 *When the algorithm stops,* $\mathcal{OPT}/f^t \le r\delta/\ln(um)^{-1}$.

Proof: Denote by $\alpha_*^k = \min_j \sum_{i=1}^m A_{i,j} y_i^k / c_j$, and by $\alpha^{k-1} = \sum_{i=1}^m A_{i,q} y_i^{k-1}/c_q$ for the index q returned by ABS. From the definition of D^k, the property of the approximate block solver ABS, and the increment of y^k according to the algorithm, we have:

$$
\begin{aligned}
D^k &= \sum_{i=1}^m b_i y_i^k = \sum_{i=1}^m b_i y_i^{k-1} + \delta \sum_{i=1}^m b_i \frac{b_p/A_{p,q}}{b_i/A_{i,q}} y_i^{k-1} \\
&= \sum_{i=1}^m b_i y_i^{k-1} + \delta \frac{b_q}{A_{p,q}} \sum_{i=1}^m A_{i,q} y_i^{k-1} \\
&= D^{k-1} + \delta(f^k - f^{k-1})\alpha^{k-1} \\
&\leq D^{k-1} + r\delta(f^k - f^{k-1})\alpha_*^{k-1}.
\end{aligned} \tag{4}
$$

Since (4) is valid for all $l = 1, \ldots, k$, summing them up yields:

$$
D^k = D^0 + r\delta \sum_{l=1}^k (f^l - f^{l-1})\alpha_*^{l-1}. \tag{5}
$$

It is clear that $\sum_{i=1}^m A_{i,j} y_j$ is the left hand side of the j-th constraint in the dual program (3), and c_j is the corresponding right hand side. If the former is at least the latter, then the constraint is satisfied. Therefore if we scale any positive vector $y = (y_1, \ldots, y_m)^T$ by an appropriate value, then we can obtain a feasible solution to the dual program. Indeed, because $\alpha_*^l = \min_i \sum_{i=1}^m A_{i,j} y_j^l / c_j$, we conclude that y^l/α_*^l is a feasible dual solution and the corresponding dual value is D^l/α_*^l. Since $\mathcal{OPT}$ is the optimal dual objective value, we have $\mathcal{OPT} = \min_{y \geq 0} D/\alpha_* \leq D^l/\alpha_*^l$ for all $l = 1, \ldots, k$. Then from (5) and the initial value $D^0 = um$, we have $D^k \leq um + r\delta \sum_{l=1}^k (f^l - f^{l-1}) D^{l-1}/\mathcal{OPT}$. We notice that in order to obtain the largest possible D^k, D^l must be as large as possible for all $l = 0, \ldots, k-1$. Denote by $(D^k)_{\max}$ the largest possible value of D^k. Then the sequence $(D^0)_{\max}, \ldots, (D^k)_{\max}$ dominates the sequence $D^0, \ldots, D^k$ where $(D^0)_{\max} = D^0$, and $(D^k)_{\max} = um + r\delta \sum_{l=1}^k (f^l - f^{l-1})(D^{l-1})_{\max}/\mathcal{OPT}$. Since the above equality holds for all $k = 0, \ldots, t$, we have $(D^k)_{\max} = (D^{k-1})_{\max} + r\delta(f^k - f^{k-1})(D^{k-1})_{\max}/\mathcal{OPT} \leq (D^{k-1})_{\max} e^{r\delta(f^k - f^{k-1})/\mathcal{OPT}}$. The last inequality comes from the elementary inequality $1 + v \leq e^v$ for any v. Since $(D^0)_{\max} = D^0 = um$ and $f^0 = 0$, we obtain that $D^k \leq (D^k)_{\max} \leq (D^0)_{\max} e^{r\delta f^k/\mathcal{OPT}} = ume^{r\delta f^k/\mathcal{OPT}}$. Thus, when the algorithm stops at the t-th iteration, the stopping rule $1 \leq D^t$ is satisfied, i.e., $1 \leq ume^{r\delta f^t/\mathcal{OPT}}$, which yields the claimed bound. □

The solution x_t delivered by the algorithm could be infeasible and some packing constraints may be violated. Thus we need to scale the solution by an appropriate amount to obtain a feasible solution. In the following we give the feasibility result:

Lemma 3.2 *The scaled solution* $x_S = x^t / \log_{1+\delta}((1+\delta)/u)$ *is feasible for the fractional packing problem (2) and the corresponding objective value is* $f^t / \log_{1+\delta}((1+\delta)/u)$.

Proof: We consider the relation between the increments of the primal solution and the dual solution. For the primal problem, we consider an equivalent LP as follows:

$$\max\{c^T x | A_i x / b_i \leq 1, x \geq 0, i = 1, \ldots, m\}. \tag{6}$$

At each iteration of the algorithm, the q-th variable x_q increases by an amount of $b_p / A_{p,q}$. Thus, the left hand side of the i-th constraint of (6) increases by an amount of $z_i = (b_p / A_{p,q}) / (b_i / A_{i,q})$. Meanwhile, the dual variable y_i also increases by a factor $1 + \delta z_i$. Since $z_i \leq 1$ for all $i = 1, \ldots, m$ according to the definition of p, if the left hand side of the i-th constraint increases by 1 in (6), then the dual variable y_i increases by a factor of at least $1 + \delta$. At the k-th iteration, $y_i^k = y_i^{k-1}(1 + \delta z_i) \leq y_i^{k-1}(1 + \delta)$. So the dual objective value increases by a factor at most $1 + \delta$, i.e., $D^k \leq D^{k-1}(1 + \delta)$. Because at the last iteration before the algorithm stops $D^{t-1} < 1$, the final dual objective value $D^t \leq 1 + \delta$. Denote by $y_l^t = \max_i y_i^t$. Then $y_l^t \leq D^t / b_l \leq (1+\delta)/b_l$. Notice that the initial value of the dual variable y_l is u/b_l. Because y_l increases by a factor at least $1 + \delta$, then the number of increments for y_l (also the number of increments of y) is bounded by $N = \log_{1+\delta}((1+\delta)/u)$. We notice that N is also an upper bound on the number of increments of the left hand side of the constraints in (6), and each time the left hand side increases by at most an amount of $z_i \leq 1$ for $i = 1, \ldots, m$. Since initially the left hand side is zero, scaling final solution x^t by the factor $\log_{1+\delta}((1+\delta)/u)$ yields a feasible solution and the proof is completed. □

Now we are ready to show the performance bound of the solution for our algorithm:

Theorem 3.1 *When the algorithm stops, the scaled solution* x_S *is a* $(1 - \varepsilon)/r$*-approximate solution to the fractional packing problem (2).*

Proof: According to the duality relation, the optimum dual value $\mathcal{OPT}$ is also the optimum objective value of the primal problem (2). Thus we need to examine the objective value corresponding to the feasible solution x_S. According to the definition $u = (1+\delta)((1+\delta)m)^{-1/\delta}$, we have $\ln(um)^{-1} = (1-\delta)\ln[m(1+\delta)]/\delta$ and $\ln((1+\delta)/u) = \ln[m(1+\delta)]/\delta$. Denote by $\mathcal{ALG}$ the objective value of the solution delivered by our algorithm. From the above relations and Lemma 3.1, the approximation ratio r_{ALG} of our algorithm has the following bound:

$$\begin{aligned} r_{ALG} &\geq \frac{\mathcal{ALG}}{\mathcal{OPT}} = \frac{f^t}{\mathcal{OPT}\log_{1+\delta}((1+\delta)/u)} \\ &\geq \frac{\ln(um)^{-1}}{r\delta}\frac{\ln(1+\delta)}{\ln((1+\delta)/u)} \\ &= \frac{(1-\delta)\ln(1+\delta)}{r\delta} \end{aligned}$$

According to the elementary inequality $\ln(1+z) \geq z - z^2/2$ for any $0 \leq z \leq 1$, we have $r_{ALG} \geq (1-\delta)(\delta - \delta^2/2)/(r\delta) \geq (1-\delta)^2/r = (1-\varepsilon)/r$, which completes the proof. □

Theorem 3.2 *There exists a $(1-\varepsilon)/r$-approximation algorithm for the fractional packing problem (2) that performs $O(m\varepsilon^{-2}\ln m)$ iterations, calling an r-approximate block solver once per iteration, for any $\varepsilon \in (0,1]$.*

Proof: The correctness is guaranteed by Theorem 3.1 if the algorithm stops within a finite number of iterations. Hence we need to count the bound on the number of iterations. At each iteration, we choose one element in the selected column with approximate minimum value of $\sum_{i=1}^{m} A_{i,j} y_i / c_j$ and increase the dual variable by an appropriate amount. As analyzed in the proof of Lemma 3.2, one dual variable increases at most $\log_{1+\delta}((1+\delta)/u)$ times. Therefore the overall number of iterations is bounded by $N \leq m\log_{1+\delta}((1+\delta)/u)$. We now estimate this bound. From the definition $\delta = 1 - \sqrt{1-\varepsilon}$ and $\varepsilon \in (0,1]$, it is clear that $\delta \leq 1$. Again, from the definition of δ, we have $1-\varepsilon = (1-\delta)^2 = 1 - 2\delta + \delta^2$. Therefore, $2\delta = \varepsilon + \delta^2 \geq \varepsilon$, i.e., $\delta \geq \varepsilon/2$. According to the definition of u, we have that $N \leq m\log_{1+\delta}((1+\delta)/u) = m\ln m/[\delta\ln(1+\delta)] \leq m\ln m/[\delta(\delta - \delta^2/2)] \leq 4m\ln m/[\varepsilon(\varepsilon - \varepsilon^2/4)] \leq 16m\ln m/(3\varepsilon^2)$. □

Thus we have developed an algorithm that find a $(1-\varepsilon)/r$-approximate solution to fractional packing problems (2) with a complexity polynomial in the input sizes, provided an approximate block solver. It is a generalization of the approximation algorithm in [9], and is the first approximation algorithm for (2) with complexity independent of the input data with an approximate block solver. In fact, our coordination complexity is exactly the same as that in [9]. We believe that this algorithm can find more applications in combinatorial optimization such as routing in communication networks and VLSI design.

4 Approximation Algorithm for the Steiner Tree Packing Problem

As mentioned in Section 2, our strategy is to find an approximate fractional solution to (1) and then round it to a feasible solution. First we study the LP-relaxation of (1).

Theorem 4.1 *There is a $(1-\varepsilon)/r$-approximation algorithm for the fractional Steiner tree packing problem with complexity $O((m+K)K\varepsilon^{-2}\beta\ln(m+K))$, where r and β are the approximation ratio and the complexity of the minimum Steiner tree solver called as an oracle, respectively.*

Proof: We use the generalized approximation algorithm for fractional packing problems developed in Section 3 to solve the LP-relaxation of (1). The only problems are to identify the block problem and to find an (approximate) solver. Consider the LP-relaxation of (1). Notice that the dual vector $y = (y_1, \ldots, y_m, y_{m+1}, \ldots, y_{m+K})^T$ consists of two types of components. The first $m = |E|$ components $y_1, \ldots, y_m$ corresponds to the edges $e_1, \ldots, e_m$, respectively. In fact they correspond to the first set of constraints in (1). The remaining K components $y_{m+1}, \ldots, y_{m+K}$ in y corresponds to the second set of constraints in (1) indicating that at most one tree is selected for routing each request. It is easy to verify that the block problem is as follows: to find a tree T that $\min_k \min_{T\in\mathcal{T}_k}(\sum_{e_i\in T} y_i + y_{m+k}\delta_{k,T})/w_k$. Here the indicator $\delta_{k,T} = 1$ if $T \in \mathcal{T}_k$, and otherwise $\delta_{k,T} = 0$. To solve the block problem, one can search for K trees corresponding to the K requests separately, such that each tree routes a request with the minimum of $\sum_{e_i\in T} y_i$. Afterwards, for each of these K trees, the additional term y_{m+k} is added, and the sums are divided by w_k respectively. Thus the tree with the minimum value of $(\sum_{e_i\in T} y_i + y_{m+k}\delta_{k,T})/w_k$ over all K trees is selected, which is the optimum solution to the block problem. Since the value y_{m+k} is fixed for a fixed request k at each iteration, the block problem is in fact equivalent to finding a tree spanning the request S_k that $\min_{T\in\mathcal{T}_k} \sum_{e_i\in T} y_i$, for all $k = 1, \ldots, K$. Regarding y_i the length associated to the edge e_i for $i = 1, \ldots, m$, the block problem is in fact the minimum Steiner tree problem in graphs. Thus, we can use the approximation algorithm developed in Section 3 with an approximate minimum Steiner tree solver to obtain a feasible solution to the LP-relaxation of (1), and the theorem follows. □

Unfortunately, the minimum Steiner tree problem is $\mathcal{APX}$-hard [2, 4]. The best known lower and upper bounds on the approximation ratio are

$96/95 \approx 1.0105$ [8] and $1 + (\ln 3)/2 \approx 1.550$ [22], respectively. Thus, the approximation algorithm in [9] is not applicable in this case.

With the fractional solution to the Steiner tree packing problem (1), we apply randomized rounding [20, 21] to find a feasible (integer) solution. As indicated in [20, 21], to guarantee non-zero probability that no constraint is violated, a scaling technique is necessary to be employed. Denote by c the minimum edge capacity. Suppose that there exists a scalar v satisfying $(ve^{1-v})^c < 1/(m+1)$. The left hand side of the above inequality is in fact the Chernoff-type bound on the probability that the weighted sum of Bernoulli trials with expectation exceed c [20]. Based on the bounds on the deviation, and the rounding results for the maximum multicommodity flow problem in [20], we can immediately obtain the following bound:

Theorem 4.2 *There is an approximation algorithms for the Steiner tree packing problem such that the objective value delivered is at least*

$$\begin{cases} (1-\varepsilon)vOPT/r - (\exp(1)-1)(1-\varepsilon)v\sqrt{OPT\ln(m+1)/r}, & \textit{if } OPT > r\ln(m+1); \\ (1-\varepsilon)vOPT/r - \dfrac{\exp(1)(1-\varepsilon)v\ln(m+1)}{1+\ln\left(r\ln(m+1)/OPT\right)}, & \textit{otherwise}, \end{cases}$$

where OPT is the optimal objective value of (1).

In (1) there are exponential number of variables. However, by applying our approximation algorithm for fractional packing problems in Section 3, we just need to generate K approximate minimum Steiner trees for the K requests at each iteration corresponding to the current dual vector. Thus there are only $O((m+K)K\varepsilon^{-2}\ln(m+K))$ Steiner trees generated in total. This is similar to the column generation technique for LPs, and the hardness due to exponential number of variables in (1) is overcome.

5 Integrality

We have presented an approximation algorithm for the Steiner tree packing problem based on our generalized approximation algorithm for fractional packing problems. The coordination complexity of the approximation algorithm is strictly polynomial in the input size m, n, K and given accuracy ε^{-1}. In this section, we investigate the possibility to directly find an integer solution to (1) near the optimal integer solution in a larger amount (maybe not polynomial) of time, similar to [9].

A solution to the fractional packing problem (2) has *integrality* w if each components in the solution is a non-negative integer multiple of w. In this case we can modify our approximation algorithm for fractional packing problems slightly to find a solution which has a small integrality.

Theorem 5.1 *If* $w \leq \min_{i,j} b_i/A_{i,j}$ *in the fractional packing problem (2), then there exists an algorithm that finds a* $(1-\varepsilon)/r$*-approximate solution to (2) with integrality* $w\delta/(1+\log_{1+\delta} m)$ *within* $O(m\varepsilon^{-2}\rho \ln m)$ *iterations, where* $\rho = \max_{i,j} b_i/A_{i,j}$.

Proof: We modify the approximation algorithm in Table 1 as follows: At the k-th iteration, after calling the approximate block solver, x_q^k increases by an amount w and y_i^k increases by a factor $[1+\delta w/(b_i/A_{i,q})]$ for all i. Because $w \leq \min_{i,j} b_i/A_{i,j}$, the increments of the primal vector x and the dual vector y are less than those in the original algorithms. So this strategy leads to a feasible solution and the correctness analysis in Section 3 is still valid. Thus this modified algorithm generates a $(1-\varepsilon)/r$-approximate solution with certain integrality. Since we need to scale the primal variables by $\log_{1+\delta}((1+\delta)/u)$, at each iteration there is an amount of increment of $w/\log_{1+\delta}((1+\delta)/u)$ in the feasible solution. Thus the integrality of the feasible solution is $w/\log_{1+\delta}((1+\delta)/u) = w\delta/(1+\log_{1+\delta} m)$ according to the definition of u.

The number of iterations now depends on the value of $b_i/A_{i,j}$ for all i and j with nonzero $A_{i,j}$, because in the worst case the algorithm needs to increase the variable $\lceil b_i/A_{i,j}w)\rceil$ times before choosing another index q. Since $\rho = \max_{i,j} b_i/A_{i,j}$, the number of iterations now is bounded by $O(m\varepsilon^{-2}\rho \ln m)$. □

Corollary 5.1 *If* $b_i/A_{i,j} \geq (1+\log_{1+\delta} m)/\delta$ *for all* i *and* j*, then there exists an algorithm that finds a* $(1-\varepsilon)/r$*-approximate solution to integer packing problems within* $O(m\varepsilon^{-2}\rho \ln m)$ *iterations.*

Corollary 5.2 *If all edge capacities are at least* $(1+\log_{1+\delta}(m+K))/\delta$*, then there exists an algorithm that finds a* $(1-\varepsilon)/r$*-approximate integer solution to the Steiner tree packing problem (1) within* $O((m+K)K\varepsilon^{-2}c_{\max}\beta \ln(m+K))$ *time, where* r *and* β *are the approximation ratio and the complexity of the minimum Steiner tree solver called as the oracle, and* $c_{\max}$ *is the maximum edge capacity.*

We have presented an approximation algorithm for integer packing problems and applied it to the Steiner tree packing problem. It is only a pseudo polynomial time approximation algorithm due to the existence of ρ in the complexity, which depends on the input data. However, this approach is still useful, as it can directly lead to an integer solution and can avoid the rounding stage. We believe that there exist instances of integer packing problems such that the actual running time using our modified approximation algorithm does not increase too much compared with that of our approximation algorithm for the corresponding fractional packing problems.

6 Multicast Routing and Wavelength Assignment in Optical Networks

Now we turn to the multicast routing and wavelength assignment problem in optical networks. In this problem, we are given an undirected graph $G = (V, E)$, a set of multicast requests $S_1, \ldots, S_K \subseteq V$, and a set $\mathcal{L} = \{1, \ldots, L\}$ of wavelengths. It is assumed that every edge represents a bundle containing multiple fibers in parallel. In particular, we let $c_{i,l}$ denote the number of fibers of edge $e_i \in E$ that have wavelength $l \in \mathcal{L}$. Note that wavelengths that are not available in a fiber are assumed to be pre-occupied by existing connections in the network. The goal is to find a routing with the maximum total profit/throughput, such that every selected request is realized by a Steiner tree and assigned one of the given wavelengths, and that trees crossing the same fiber are assigned different wavelengths.

We now give the ILP formulation for the multicast routing and wavelength assignment problem in optical networks. Denote by $\mathcal{T}_k$ the set of all trees spanning the request S_k, for all $k = 1, \ldots, K$. Here $|\mathcal{T}_k|$ could be exponentially large. Then we define an indicator variable $x_k(T, l)$ as follows:

$$x_k(T,l) = \begin{cases} 1, & \text{if } T \in \mathcal{T}_k \text{ is selected for routing } S_k \text{ and is assigned wavelength } l; \\ 0, & \text{otherwise.} \end{cases}$$

In addition, each request S_K is associated with a weight w_k indicating its importance in the given multicast optical network. Thus the ILP of the problem is as follows:

$$\begin{array}{rll} \max & \sum_{k=1}^{K} w_k \sum_{l=1}^{L} \sum_{T \in \mathcal{T}_k} x_k(T,l) & \\ \text{s.t.} & \sum_{k=1}^{K} \sum_{T \in \mathcal{T}_k \& e_i \in T} x_k(T,l) \le c_{i,l}, & \forall e_i \in E \,\&\, l \in \mathcal{L}; \\ & \sum_{l=1}^{L} \sum_{T \in \mathcal{T}_k} x_k(T,l) \le 1, & k = 1, \ldots, K; \\ & x_k(T,l) \in \{0,1\}, & \forall T,\, l \in \mathcal{L} \,\&\, k = 1, \ldots, K. \end{array} \tag{7}$$

The first set of constraints mean that the number of trees using a specific edge and assigned the same wavelength should be bounded by the number of fibers on that edge. This ensures that each of these trees can be routed through a separate fiber. The second set of constraints indicate that the we just need to route each request by at most one tree and assign at most one wavelength to it.

We use similar strategy as in the previous sections. First, for the fractional multicast routing and wavelength assignment problem, we have the following result:

Theorem 6.1 *There is a $(1-\varepsilon)/r$-approximation algorithm for the fractional multicast routing and wavelength assignment problem in optical networks with complexity $O((mL+K)KL\varepsilon^{-2}\beta\ln(mL+K))$, where r and β are the approximation ratio and the complexity of the minimum Steiner tree solver called as the oracle, respectively.*

Proof: We just need to consider the block problem. There are two types of components in the dual vector y. The first type of components corresponding to the first set of constraints (capacity constraints) in (7) are $y_1, \ldots, y_m, y_{m+1}, \ldots, y_{2m}, \ldots, y_{m(L-1)+1}, \ldots, y_{mL}$. The remaining components $y_{mL+1}, \ldots, y_{mL+K}$ corresponds to the second set of constraints in (7). It is easy to verify that the block problem of the LP-relaxation of (7) is to find a tree T that $\min_k \min_l \min_{T \in \mathcal{T}_k}(\sum_{e_i \in T} y_{ml+i} + y_{mL+k}\delta_{k,T})/w_k$, where the indicator variable $\delta_{k,T} = 1$ if $T \in \mathcal{T}_k$, and otherwise $\delta_{k,T} = 0$. To find the minimum, we can just search a number of KL trees to attain $\min_{T \in \mathcal{T}_k} \sum_{e_i \in T} y_{ml+i}$, for all $k = 1, \ldots, K$ and $l = 1, \ldots, L$. Then we can find the minimal objective value of the block problem over all these KL trees. Consider a graph $H = (V_H, E_H)$, which has L components and without edges between the components. Each component has the same vertex set and edge set as G. For the l-th component, its edge lengths are $y_{m(l-1)+1}, \ldots, y_{ml}$. Now the block problem is in fact the minimum Steiner tree problem in each component of graph H. With an r-approximate minimum Steiner tree solver and using our approximation algorithm for fractional packing problems, the theorem follows. □

Similar to Section 4, for any real number v satisfying $(ve^{1-v})^c < 1/(m+1)$, where $c = \min_{i,l} c_{i,l}$ is the minimal capacity, we can obtain a bound for the integer solution by randomized rounding [20, 21]:

Theorem 6.2 *There is an approximation algorithms for the multicast routing and wavelength assignment problem in optical networks such that the objective value delivered is at least*

$$\begin{cases} (1-\varepsilon)vOPT/r - (\exp(1)-1)(1-\varepsilon)v\sqrt{OPT\ln(m+1)/r}, & \textit{if } OPT > r\ln(m+1); \\ (1-\varepsilon)vOPT/r - \dfrac{\exp(1)(1-\varepsilon)v\ln(m+1)}{1+\ln\left(r\ln(m+1)/OPT\right)}, & \textit{otherwise}, \end{cases}$$

where OPT is the optimal objective value of (7).

Furthermore, we can apply the modified approximation algorithm for integer packing problems described in Section 5 to (7) and directly obtain an integer solution:

Theorem 6.3 *If all edge capacities are at least $(1+\log_{1+\delta}(mL+K))/\delta$, then there exists an algorithm that finds a $(1-\varepsilon)/r$-approximate integer solution to the multicast routing and wavelength assignment problem in optical networks (7) within $O((mL+K)LK\varepsilon^{-2}c_{\max}\beta\ln(mL+K))$ time, where r and β are the approximation ratio and the complexity of the minimum Steiner tree solver called as the oracle, and $c_{\max}$ is the maximum capacity.*

7 Conclusion

in this paper, we have addressed the problem of maximizing a Steiner tree packing, such that each tree connects a subset of required vertices without violating the edge capacity constraints. This problem is motivated by applications in multicast communication networks. We have developed a $(1-\varepsilon)/r$ approximation algorithm to solve the LP-relaxation provided an r-approximate block solver. This is the first approximation result for fractional packing problems with only approximate block solvers and a coordination complexity that is polynomial in the input size. This generalizes the well-known result in [9] while the complexity is the same, which is superior to many other approximation algorithms for fractional packing problems, e.g., [19, 27]. In this way we have designed approximation algorithms for the Steiner tree packing problem. Finally, we have studied an important multicast routing and wavelength assignment problem in optical networks.

We are further interested in both theoretical and practical extensions of this work. An interesting problem is to develop approximation algorithms for fractional/integer packing problems with other properties, e.g., with negative entries or with block structure in the coefficient matrix. From the practical point of view, we aim to develop better realistic models for routing problems arising in communication networks and design strategies to (approximately) solve them efficiently. We have applied our approximation algorithm for fractional packing problems to the global routing problem in VLSI design [23], and we are exploring other application areas. In addition, we are working on implementation of our approximation algorithms with challenging benchmarks to explore their power in computational practice.

References

[1] K. M. Anstreicher, Towards a practical volumetric cutting plane method for convex programming, *SIAM J. Optimization*, 9 (1999), 190-206.

[2] S. Arora, C. Lund, R. Motwani, M. Sudan and M. Szegedy, Proof verification and hardness of approximation problems, *J. ACM*, 45 (1998), 501-555.

[3] A. Baltz and A. Srivastav, Fast approximation of minimum multicast congestion - implementation versus theory, *RAIRO Oper. Res.*, 38 (2004), 319-344.

[4] M. Bern and P. Plassmann, The Steiner problem with edge lengths 1 and 2, *Inf. Process. Lett.*, 32 (1989), 171-176.

[5] M. Cai, X. Deng and L. Wang, Minimum k arborescences with bandwidth constraints, *Algorithmica*, 38 (2004), 529-537.

[6] R. Carr and S. Vempala, Randomized meta-rounding, *Proceedings of STOC 2000*, 58-62.

[7] S. Chen, O. Günlük and B. Yener, The multicast packing problem, *IEEE/ACM Trans. Networking*, 8 (3) (2000), 311-318.

[8] M. Chlebík and J. Chlebíková, Approximation hardness of the Steiner tree problem, *Proceedings of SWAT 2002*, LNCS 2368, 170-179.

[9] N. Garg and J. Könemann, Fast and simpler algorithms for multicommodity flow and other fractional packing problems, *Proceedings of FOCS 1998*, 300-309.

[10] M. D. Grigoriadis and L. G. Khachiyan, Coordination complexity of parallel price-directive decomposition, *Math. Oper. Res.*, 2 (1996), 321-340.

[11] M. Grötschel, L. Lovász and A. Schrijver, The ellipsoid method and its consequences in combinatorial optimization, *Combinatorica*, 1 (1981), 169-197.

[12] K. Jansen and H. Zhang, Approximation algorithms for general packing problems with modified logarithmic potential function, *Proceedings of TCS 2002*, 255-266.

[13] K. Jansen and H. Zhang, An approximation algorithm for the multicast congestion problem via minimum Steiner trees, *Proceedings of ARACNE 2002*, 77-90.

[14] K. Jain, M. Mahdian and M.R. Salavatipour, Packing Steiner trees, *Proceedings of SODA 2003*, 266-274.

[15] X. Jia and L. Wang, A group multicast routing algorithm by using multiple minimum Steiner trees, *Comput. Commun.* 20 (1997), 750-758.

[16] L. C. Lau, An approximate max-Steiner-tree-packing min-Steiner-cut theorem, *Proceedings of STOC 2004*, 61-70.

[17] L. C. Lau, Packing Steiner forests, *Proceedings of IPCO 2005*, 362-376.

[18] Q. Lu and H. Zhang, Implementation of approximation algorithms for the multicast congestion problem, *Proceedings of WEA 2005*, LNCS 3503, 152-164.

[19] S. A. Plotkin, D. B. Shmoys and E. Tardos, Fast approximation algorithms for fractional packing and covering problems, *Math. Oper. Res.*, 2 (1995), 257-301.

[20] P. Raghavan, Probabilistic construction of deterministic algorithms: approximating packing integer programs, *J. Comput. Syst. Sci.*, 37 (1988), 130-143.

[21] P. Raghavan and C. D. Thompson, Randomized rounding: a technique for provably good algorithms and algorithmic proofs, *Combinatorica* 7 (4) (1987), 365-374.

[22] G. Robins and A. Zelikovsky, Improved Steiner tree approximation in graphs, *Proceedings of SODA 2000*, 770-779.

[23] M. Saad, T. Terlaky, A. Vannelli and H. Zhang, A provably good global routing algorithm in multilayer IC and MCM layout designs, *manuscript.*

[24] T. Terlaky, A. Vannelli and H. Zhang, On routing in VLSI design and communication networks, *manuscript.*

[25] S. Vempala and B. Vöcking, Approximating multicast congestion, *Proceedings of ISAAC 1999*, LNCS 1741, 367-372.

[26] J. Villavicencio and M. D. Grigoriadis, Approximate Lagrangian decomposition with a modified Karmarkar logarithmic potential, *Network Optimization, P. Pardalos, D. W. Hearn and W. W. Hager (Eds.), Lecture Notes in Economics and Mathematical Systems 450, Springer-Verlag, Berlin*, (1997), 471-485.

[27] N. E. Young, Randomized rounding without solving the linear program, *Proceedings of SODA 1995*, 170-178.