Provisioning a Virtual Private Network Under the Presence of Non-Communicating Groups

Friedrich Eisenbrand[1] and Edda Happ[2]

[1] University of Dortmund, Otto Hahn Str. 14,
D-44221 Dortmund, Germany,
Friedrich.Eisenbrand@cs.uni-dortmund.de
[2] Max-Planck-Institut fr Informatik, Stuhlsatzenhausweg 85,
D-66123 Saarbrcken, Germany,
edda@mpi-sb.mpg.de

Abstract. Virtual private network design in the hose model deals with the reservation of capacities in a weighted graph such that the terminals in this network can communicate with one another. Each terminal is equipped with an upper bound on the amount of traffic that the terminal can send or receive. The task is to install capacities at minimum cost and to compute paths for each unordered terminal pair such that each valid traffic matrix can be routed along those paths.
In this paper we consider a variant of the virtual private network design problem which generalizes the previously studied symmetric and asymmetric case. In our model the terminal set is partitioned into a number of groups, where terminals of each group do not communicate with each other.
Our main result is a 4.74 approximation algorithm for this problem.

1 Introduction

Suppose that a large globally operating company wants to connect all of its branch-offices into a common network to ensure communication between the offices. One approach to do so is to build the network on top of an existing public network by buying a certain amount of link capacities which is then reserved exclusively for the use of this company. In this way the company has established a *virtual private network*. The capacity reservation on links comes with certain costs which we assume to be linear in the amount of reserved capacity.

The network can be modeled as an undirected graph $G = (V, E)$ with edge costs $c : E \longrightarrow \mathbb{R}_+$ reflecting the cost of reserving one unit of capacity on an edge. The branch-offices are a subset $T \subseteq V$ of the nodes which are the *terminals* of this network design problem. A solution to the problem is an assignment of capacities to the edges and paths P_{ij} for each unordered pair $\{i, j\} \subseteq T$ of terminals such that all possible traffic between the terminals can be routed along those paths over the network.

Predicting the amount of traffic that pairs of terminals exchange is often illusive. In the so-called hose model [1, 2] the knowledge of the exact amount of traffic which is exchanged between the terminal pairs is relaxed into a prediction of how much traffic occurrs at each terminal. Here, each terminal $v \in T$ has an threshold $b(v) \in \mathbb{Z}_{\geq 0}$ which is an upper bound on the amount of network traffic that this terminal can interchange with other terminals.

A *traffic matrix* $D \in \mathbb{Q}_{\geq 0}^{TT}$ is a symmetric rational matrix which represents the amount of communication between terminals. The traffic matrix is *valid*, if it respects the upper bounds, i.e., if the following holds for each terminal $i \in T$

$$\sum_{j \in T, j \neq i} D(i,j) \leq b(i) \ . \tag{1}$$

Virtual private network design is the optimization problem that searches a minimum cost assignment of capacities to the edges and specifies for each unordered terminal pair $i, j \in T$ a path P_{ij} in the network such that each valid traffic matrix can be routed along these paths without exceeding the capacities.

This virtual private network design problem has received a considerable amount of attention. Gupta et al. [3] provided a 2-approximation algorithm for this problem and showed that it can be solved in polynomial time when the graph stemming from the edges with nonzero capacity reservation is supposed to form a tree. It is a well known conjecture that there always exists an optimal tree reservation. Hurkens, Keijsper and Stougie [4] have recently shown that this is the case in ring networks. Computational evidence that it also holds in arbitrary networks is for example presented in [5, 6].

In the *asymmetric* variant of virtual private network design, one distinguishes between traffic which is *sent* and traffic which is *received* by a terminal. A traffic matrix then has to respect these upper bounds on each vertex in order to be valid. Via duplicating each terminal into two terminals, where one copy can only send and the other can only receive traffic, the asymmetric variant can be formalized as follows. The terminal set T is partitioned into two sets $\mathcal{R}$ and $\mathcal{S}$, representing *receivers* and *senders*, respectively. The terminals are equipped with upper bounds $b(v)$ as above. A traffic matrix is valid, if it satisfies (1) and if $D(i, j) = 0$ whenever i and j are both senders or both receivers.

The asymmetric virtual private network design problem is NP-hard [3] which follows from a reduction to the steiner tree problem. Gupta et al.[3] gave the first constant factor approximation algorithm for this problem. Gupta, Kumar and Roughgarden [7] presented a randomized approximation algorithm. Their algorithm samples terminals which are then connected into a high bandwidth core. The remaining terminals are connected along their shortest paths to this core. The approximation ratio of this algorithm is 5.55. This result was refined to a 4.74 approximation [8] which also finds a tree solution. The first non-tree approximation algorithm achieves an approximation factor of 3.55 [9]. Italiano, Leonardi and Oriolo [10] consider the setting in which the sums of the sender and receiver thresholds are equal.

1.1 A Setting in Which Some Terminals Do Not Communicate

In this paper we consider a variant of the virtual private network design problem which generalizes both the symmetric and asymmetric version of virtual private network design.

The terminals T are partitioned into disjoint sets $T_1, \dots, T_k$. Network traffic only occurs between terminals i and j if i and j are in different sets $T_i \neq T_j$. This means that a traffic matrix Q is now valid if $D(i,j) = 0$ for all $i, j \in T_\ell$ and all $1 \leq \ell \leq k$ and

$$\sum_{j \in T, j \neq i} D(i,j) \leq b(i) \text{ for all } i \in T \ .$$

The goal is now to determine paths between each unordered pair of terminals belonging to different sets and to reserve capacities on the edges such that each valid traffic matrix can be routed and the capacity reservation has minimum cost. In the following we refer to this combinatorial optimization problem as *virtual private network design* (VPND).

If the terminal sets $T_1, \dots, T_k$ are singletons, then we are dealing with the symmetric virtual private network design problem. If the terminals are partitioned into two sets only, then this is the setting of the asymmetric case. Thus, our model is flexible enough to capture both variants of network design which have previously been studied in the literature.

A possible application scenario where this more general model is relevant is as follows (Fig. 1). Some companies want to cooperate and to connect all their branch-offices via a common virtual private network. The companies themselves are already connected.

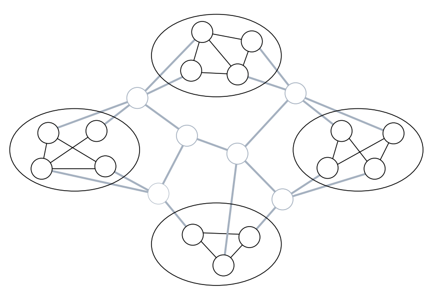

Fig. 1. Companies having internal networks have established a joint network.

One possible approach would be to use these connections and to treat the existing small networks as one terminal. Thus, all communication leaving a company network would have to be collected in one selected node and then sent outside. This might cause congestion in the small networks and might lead to a necessary renegotiation for new contracts with the providers of the small networks. It could be cheaper to consider the VPND problem in which the terminal sets correspond to the companies which are already connected.

We would also like to mention that the network design problem of building a minimum cost virtual private network connecting one terminal of each company is hard to approximate with a factor of less than $\log(n)$ since it is a generalization of the group steiner tree problem [11] which is known to have this bound [12, 13].

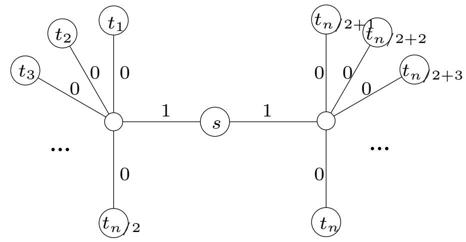

Fig. 2. A network which demonstrates that the optimal solution can differ considerably depending on the partitioning of the terminals. The edges are labeled with their costs.

The following example (Fig. 2) shows that the optimal solution can differ considerably on the same graph and the same terminals depending on the partitioning of the terminals. The terminals are the set $\{s, t_1, \ldots, t_n\}$. The threshold on each node is one.

A solution to the corresponding symmetric problem requires a reservation of $\frac{n}{2}$ on the edges adjacent to s which has cost n while a solution to the asymmetric problem where the set of senders is $\{s\}$ and the set of receivers is $\mathcal{R} = \{t_1, \ldots, t_n\}$ requires only a reservation of 1. It is easy to see there exists a partitioning of the terminals for each even natural number $i \in \{1, \ldots, n\}$ such that the optimum value of the corresponding VPND problem is exactly i.

Contribution of This Paper

The above example shows that an arbitrary reduction to the symmetric or asymmetric case does not yield a constant factor approximation. Our main result however is a proof that ignoring the terminal partitions, i.e. solving the corresponding symmetric case, yields a constant factor approximation to VPND unless the problem is unbalanced. This is the case if the size of one terminal partition is larger than the sum of the remaining partitions. In this case we show that an optimal tree solution of the asymmetric problem stemming from identifying the large terminal partition as the set of receivers and collecting the remaining terminals from the other partitions into a set of senders yields a constant factor approximation.

Assume without loss of generality that the terminal sets are ordered in decreasing cardinality, i.e., $|T_i| \geq |T_j|$ for $i < j$. We call the VPND instance *unbalanced* if $|T_1| \geq \sum_{i>1} |T_i| - 1$.

We show that the following algorithm is a 4.74 approximation algorithm for VPND.

Algorithm 1 VPND $(G, \bigcup_{i=1}^{k} T_i, c)$

---

1. If the VPND-instance is unbalanced then return an approximate tree solution for the asymmetric problem with senders T_1 and receivers $T_2 \cup \ldots \cup T_k$.
2. Otherwise output an approximate solution of the symmetric VPND-instance with terminal set $T_1 \cup \ldots \cup T_k$.

We show that if we use the randomized approximation algorithm [8] in step 1 and the algorithm [3] in step 2, then we achieve an overall approximation ratio of 4.74 which coincides with the approximation ratio of the algorithm in [8].

2 Subinstances and Their Optimal Solutions

By duplicating terminals we can assume that $b(i) = 1$ for all $i \in T$. Suppose that the paths $\mathcal{P}$ are given along which the flow has to be routed. We can compute the corresponding necessary capacity assignment as follows. Consider the complete k-partite graph $B = (T_1 \cup \cdots \cup T_k, E^B)$ and the set of matchings $\mathcal{M}$ of B. Each $M \in \mathcal{M}$ corresponds to a valid traffic matrix. We have to make sure that for all M all paths can be packed. Therefore we compute the capacity $u(e)$ of an edge e as

$$u(e) = \max_{M \in \mathcal{M}} |\{P_{rs} \in \mathcal{P} \mid e \in P_{rs} \text{ and } rs \in M\}| \ . \tag{2}$$

The following is a generalization of a similar statement for the asymmetric case [9].

Lemma 1. *Let $H_1, \ldots, H_\ell$ be a partitioning of the terminals T. We denote the* VPND*-instance on graph G with Terminals $T \cap H_i$ and corresponding partitioning $T_1 \cap H_i, \ldots, T_k \cap H_i$ by I_i. Then one has*

$$\sum_{i=1}^{k} \mathrm{OPT}_i \leq \mathrm{OPT} \ ,$$

where OPT_i is the optimum cost of instance I_i.

Proof. Let $\mathcal{P}$ be an optimal set of paths for the original VPND-instance with resulting capacity reservation $u : E \to \mathbb{Z}_+$. The subset $\mathcal{P}_i \subseteq \mathcal{P}$ of paths with both endpoints in H_i defines a solution to instances I_i with the corresponding capacity reservation $u_i : E \to \mathbb{Z}_+$. It suffices to show that $\sum_{i=1}^{k} u_i(e) \leq u(e)$ for each edge $e \in E$.

It follows from (2) that for each $i = 1, \ldots, k$

$$u_i(e) \;=\; \max_{M_i \in \mathcal{M}_i} \left|\{P_{rs} \in \mathcal{P}_i \mid e \in P_{rs} \text{ and } rs \in M_i\}\right| \;.$$

Let $\tilde{M}_i$ denote the matching for which the maximum is attained. Then, the disjoint union $\tilde{M} := \bigcup_{i=1}^{k} \tilde{M}_i$ is a matching of B. It thus follows from (2) that

$$\begin{aligned}\sum_{i=1}^{k} u_i(e) = & \sum_{i=1}^{k} \left|\{P_{rs} \in \mathcal{P}_i \mid e \in P_{rs}, rs \in \tilde{M}_i\}\right| \\ = & \left|\{P_{rs} \in \mathcal{P} \mid e \in P_{rs}, rs \in \tilde{M}\}\right| \;\leq\; u(e)\end{aligned}$$

for each edge $e \in E$. This concludes the proof. □

3 An Unbalanced Terminal Set

Let us first consider unbalanced instances of VPND with $|T_1| \geq \sum_{i>1} |T_i| - 1$ which is the case in step 1 of the algorithm.

Theorem 1. *Let $(G, \bigcup_{i=1}^{k} T_i, c)$ be an unbalanced* VPND *instance. Then any tree solution to the corresponding asymmetric virtual private network design problem with $\mathcal{R} = T_1$ and $\mathcal{S} = \bigcup_{i>1} T_i$ is a valid solution to the* VPND*-instance.*

Proof. Assume the opposite. Then there exists a valid traffic matrix corresponding to a k-partite matching M that cannot be routed on the tree solution to the asymmetric virtual private network design problem. Since any bipartite matching on $\mathcal{S} \cup \mathcal{R}$ can be routed, M contains matched pairs $t_i t_j$ with $t_i, t_j \notin T_1$. Let M^* be a non-routable matching having the *minimal number* of such pairs. Consider matching $M' := M^* \setminus \{t_i t_j\}$ where $t_i, t_j \notin T_1$. Since $|T_1| \geq \sum_{i>1} |T_i| - 1 > |\bigcup_{i>1} T_i \setminus \{t_i, t_j\}| = \sum_{i>1} |T_i| - 2$ there is at least one terminal $t^* \in T_1$ which is idle in M'. So $M' \cup \{t_i t^*\}$ and $M' \cup \{t_j t^*\}$ must be routable since the number of pairs with neither terminal in T_1 is smaller than in M^*. That means that on the path from t_i to t^* and on the path from t_j to t^* one unit of capacity must be free, and therefore also on the unique path from t_i to t_j. So M^* is routable. □

We have to require tree solutions to guarantee an unambiguous path from t_i to t_j independent of t^*. Figure 3 shows a non-tree solution to the asymmetric virtual private network design problem with $\mathcal{R} = \{t_1^*, t_2^*\}$ and $\mathcal{S} = \{t_1, t_2, t_3\}$. If we consider the corresponding VPND with $T_1 = \mathcal{R}$, $T_2 = \{t_1, t_2\}$, and $T_3 = \{t_3\}$, then the condition $|T_1| \geq |T_2| + |T_3| - 1$ holds, but however we fix the path between t_1 and t_3 (e.g. as given in gray in Fig. 3) there is a valid traffic matrix (e.g. $t_2^* t_2, t_1 t_3$) that is not routable even though there is a path between t_1 and t_3 with free capacity (but it is not the fixed path).

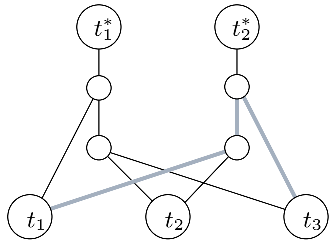

Fig. 3. Non-tree solution of asymmetric problem version not sufficient for VPND.

The current best tree approximation algorithm to the asymmetric virtual private network design problem has an approximation factor of 4.74 [8]. Since any solution to the VPND $(G, \bigcup_{i=1}^{k} T_i, c)$ is also a solution to the problem when we replace some sets by their union we have $\mathrm{OPT}_{\mathrm{asym}} \le \mathrm{OPT}_{\texttt{VPND}}$ implying that the above is also a 4.74 approximation to VPND for unbalanced instances.

4 A Balanced Terminal Set

In the following we denote the shortest path distance between i and j in the Graph $G = (V, E)$ with edge costs $c : E \longrightarrow \mathbb{R}_+$ by $\ell(i, j)$. Finding the terminal t that minimizes $\sum_{t' \in T} \ell(t, t')$ and adding one unit of capacity along each shortest path gives a valid solution to the symmetric virtual private network design problem (G, T, c) where every terminal can communicate with any other [3]. Obviously, it is also a solution to VPND where we restrict the communication to be only between terminals of different sets.

Let us now consider VPND instances where $|T_1| \le \sum_{i>1} |T_i| - 2$ which we will call *balanced*. We use the following theorem which is proven in [9, Theorem 2] to show that in this case the cheapest shortest path tree is a factor 3 approximation to the optimum solution.

Theorem 2 ([9]). *Consider an instance of* VPND *with two terminal sets T_1, T_2 with $|T_1| = |T_2|$. Let M be an arbitrary matching of the complete graph on $T_1 \cup T_2$. Then*

$$\sum_{uv \in M} \ell(u, v) \le \mathrm{OPT} \ .$$

To prove the approximation factor for VPND we use the following lower bound.

Theorem 3. *Let* OPT *be the optimal cost of a balanced* VPND *and M an inclusion-wise maximal matching of the terminals. Then*

$$\sum_{t_1 t_2 \in M} \ell(t_1, t_2) \le \frac{|M|}{|M| - 1} \cdot \mathrm{OPT} \ .$$

Proof. Recall that $|T_1| \leq \sum_{i>1} |T_i| - 2$. If the number of terminals is odd, then one terminal node is free and we have $|T_1| \leq \sum_{i>1} |T_i| - 3$. Thus, if we discard the one possibly free node and the endpoints of the lightest edge in the matching from the terminal set, we obtain a new instance where $|T_1| \leq \sum_{i>1} |T_i|$. Thus, we can assume that $|T_1| \leq \sum_{i>1} |T_i|$ holds, $|M|$ is even and all terminals are matched. We will now show that we have

$$\sum_{t_1 t_2 \in M} \ell(t_1, t_2) \leq \mathrm{OPT} \ .$$

The assertion then follows since we removed the lightest edge from the matching.

We proceed by showing that the edges of M can be paired in such a way that each pair of edges t_1t_2, t_3t_4 satisfies one of the following conditions.

(i) There exists a terminal set T_i such that $t_1, t_2 \in T_i$ and $t_3, t_4 \notin T_i$, or
(ii) there does not exist a terminal set T_i such that $t_1, t_2 \in T_i$ or $t_3, t_4 \in T_i$.

Let $M_i \subseteq M$ be the set of those edges of M having both endpoints in T_i. In other words

$$M_i = \{uv \in M \mid u, v \in T_i\}$$

and assume that the cardinality of M_l is maximal. The endpoints of the remaining edges of M belong to different sets. We partition them into the set $\overline{M}$ containing edges having one node in T_l and the set $\widetilde{M}$ containing edges that do not comprise nodes of T_l.

We now distinguish two cases. Suppose first that $|M_l| > \sum_{j \neq l} |M_j|$. Since $|T_l| \leq \sum_{j \neq l} |T_j|$ this implies

$$|M_l| \leq \sum_{j \neq l} |M_j| + |\widetilde{M}| \ .$$

This allows us to pair each edge of M_l with an edge from $\bigcup_{j \neq l} M_j \cup \widetilde{M}$ such that all edges of $\bigcup_{j \neq l} M_j$ are paired. These pairs will satisfy Condition (i). The remaining edges of $\widetilde{M}$ and the edges of $\overline{M}$ are then paired arbitrarily and satisfy Condition (ii).

In the second case one has

$$|M_l| \leq \sum_{j \neq l} |M_j| \ . \tag{3}$$

Consider an ordering $e_1, \ldots, e_\mu$ of the edges of $M_1 \cup \cdots \cup M_k$ in which the elements of M_i precede the elements of M_j whenever $i < j$. If μ is odd, we can find an edge e_i that can be paired with an edge from $\overline{M} \cup \widetilde{M}$ such that Condition (i) holds and (3) still holds. So assume that μ is even. The pairings $\{e_i, e_{\mu/2+i}\}$ for $1 \leq i \leq \mu/2$ satisfy Condition (i) and are of the left type in Figure 4. The remaining edges can be paired arbitrarily.

Let $\{t_1t_2, t_3t_4\}, \ldots, \{t_{4s+1}t_{4s+2}, t_{4s+3}t_{4s+4}\}$ be such a pairing where each pair satisfies either condition (i) or condition (ii). We now partition T into the sets $H_i = \{t_{4i+1}, t_{4i+2}, t_{4i+3}, t_{4i+4}\}$ for $i = 0, \ldots, s$ and show that

$$\ell(t_{4i+1}t_{4i+2}) + \ell(t_{4i+3}t_{4i+4}) \leq \mathrm{OPT}_i \ , \tag{4}$$

where we use the terminology of Lemma 1.

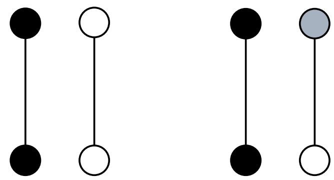

Fig. 4. The possible colorings of nodes in paired edges satisfying condition (i).

If the paired edges satisfy condition (ii), then the edges correspond to a valid traffic matrix and (4) clearly holds.

There are two possibilities on how the terminals in a paired set of edges satisfying Condition (i) can be colored, see Fig. 4. Here membership to a terminal set is interpreted as a color. In the first case, the endpoints of the first edge share the same color as well as the endpoints of the second edge. The assertion then follows from Theorem 2. We further constrain the problem of the second case by recoloring the gray node white. In other words, we forbid communication between the gray and white node. The optimal solution to this problem is at most as expensive as the optimal solution to the original one. This settles (4).

Applying Lemma 1 concludes the proof. □

Consider the complete graph $K = (T, E^K)$ on the terminals T with edge costs equal to the shortest path distances between the terminals in the original graph G. The cost of the shortest path tree of a terminal t in G is equal to the cost of the star of t in the graph K. The edges E^K can be covered by at most $|T|$ matchings. Therefore there exists a star whose cost is bounded by $2 \cdot \ell(M)$, where M is a maximum weight matching of K. By Theorem 3 one has

$$\ell(M) \leq \frac{\lfloor \frac{|T|}{2} \rfloor}{\lfloor \frac{|T|}{2} \rfloor - 1} \cdot \mathrm{OPT} \leq \frac{3}{2} \cdot \mathrm{OPT}$$

for $|T| \geq 6$ which is the case whenever any $|T_i| \geq 2$. If $T_i = 1$ for all i it is a symmetric virtual private network design problem for which this tree is known to be a 2 approximation.

This implies the following theorem.

Theorem 4. *Let* $(G, \cup_{i=1}^{k} T_i, c)$ *be a* balanced `VPND`*-instance. The cheapest shortest path tree yields a tree-solution whose cost is at most* $3 \cdot \mathrm{OPT}$*.*

By combining Theorem 1 and Theorem 4 we obtain our main result.

Theorem 5. *There exists a 4.74 randomized approximation algorithm for* VPND.

References

1. Duffield, N.G., Goyal, P., Greenberg, A., Mishra, P., Ramakrishnan, K.K., van der Merive, J.E.: A flexible model for resource management in virtual private networks. In: Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication, ACM Press (1999) 95–108
2. Fingerhut, J.A., Suri, S., Turner, J.S.: Designing least-cost nonblocking broadband networks. Journal of Algorithms **24**(2) (1997) 287–309
3. Gupta, A., Kleinberg, J., Kumar, A., Rastogi, R., Yener, B.: Provisioning a virtual private network: a network design problem for multicommodity flow. (2001) 389–398
4. Hurkens, C., Keijsper, J., Stougie, L.: Virtual private network design: A proof of the tree routing conjecture on ring networks. In: Proceedings of the eleventh Conference on Integer Programming and Combinatorial Optimization, IPCO XI. (2005) to appear.
5. Erlebach, T., Regg, M.: Optimal bandwidth reservation in hose-model vpns with multi-path routing. In: INFOCOM. (2004)
6. Altin, A., Amaldi, E., Belotti, P., Pinar, M.: Provisioning virtual private networks under traffic uncertainty. In: Proceedings of the 3rd Cologne Twente Workshop on Graphs and Combinatorial Optimization (CTW'04). Volume 17 of Electronic Notes in Discrete Mathematics. (2004) 19–22
7. Gupta, A., Kumar, A., Roughgarden, T.: Simpler and better approximation algorithms for network design. In ACM, ed.: Proceedings of the Thirty-Fifth ACM Symposium on Theory of Computing, San Diego, CA, USA, June 9–11, 2003, New York, NY, USA, ACM Press (2003) 365–372
8. Eisenbrand, F., Grandoni, F.: An improved approximation algorithm for virtual private network design. In: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms, SODA 05. (2005) 928–932
9. Eisenbrand, F., Grandoni, F., Oriolo, G., Skutella, M.: New approaches for virtual private network design. (to appear in the proceedings of ICALP 05)
10. Italiano, G., Leonardi, S., Oriolo, G.: Design of networks in the hose model. In: Proceedings of ARACNE 2002. (2002) pp 65–76
11. Reich, G., Widmayer, P.: Beyond Steiner's problem: A VLSI oriented generalization. In: Graph-Theoretic Concepts in Computer Science WG-89. Volume 411 of Lecture Notes in Computer Science. (1990) 196–210
12. Ihler, E.: The complexity of approximating the class Steiner tree problem. Technical report, Institut fr Informatik, Albert-Ludwigs-Universitt Freiburg (1991)
13. Feige, U.: A threshold of ln n for approximating set cover. In: Proc. of the ACM sympository on the Theory of Computing. (1996) 314–318