Set Cover with Requirements and Costs Evolving over Time

Milena Mihail

College of Computing and Department of Industrial and Systems Engineering
Georgia Institute of Technology, Atlanta GA 30332
mihail@cc.gatech.edu

Abstract. We model certain issues of future planning by introducing time parameters to the set cover problem. For example, this model captures the scenario of optimization under projections of increasing covering demand and decreasing set cost. We obtain an efficient approximation algorithm with performance guarantee independent of time, thus achieving planning for the future with the same accuracy as optimizing in the standard static model.
From a technical point of view, the difficulty in scheduling the evolution of a (set cover) solution that is "good over time" is in quantifying the intuition that "a solution which is suboptimal for time t may be chosen, if this solution reduces substantially the additional cost required to obtain a solution for $t' > t$". We use the greedy set picking approach, however, we introduce a new criterion for evaluating the potential benefit of sets that addresses precisely the above difficulty.
The above extension of the set cover problem arose in a toolkit for automated design and architecture evolution of high speed networks. Further optimization problems that arise in the same context include survivable network design, facility location with demands and natural extensions of these problems under projections of increasing demands and decreasing costs; obtaining efficient approximation algorithms for the latter questions are interesting open problems.

1 Introduction: Set Cover with Time Parameters

Fluctuations in cost and demand is a natural phenomenon of free markets and the relevance of cost effective schedules under such fluctuations is fundamental for both clients and service providers. When cost and demand fluctuations are totally unpredictable the models are necessarily on-line and, by now, we have a rich theory of on-line algorithms with heuristics that have found concrete practical applications [2] [4] [14]. However, in many cases, fairly accurate projections for the evolution of cost and demand over time are known in advance and the performance measures of on-line models no longer apply. This paper focuses on the latter context.

The concern that motivated this paper involves the evolution of networks like SONET, ATM, FRAME RELAY, WDM, e.t.c. which are experiencing a sharp increase in service demand (e.g. bandwidth) together with a substantial decrease in cost for the upgrade of the infrastructure that will cover this demand. Tools for automated design of such networks input service demand and equipment cost projected over several points in the future, and output a network solution that evolves over these points in the future. At the algorithmic core of such tools it is typical to find problems reminiscent of set cover and its variants. Very roughly, elements represent bandwidth demand and sets represent systems that can cover a collection of demands. For example, this approach was explicitly taken in Bellcore's SONET planning tool [6]; see Figure 1. We thus need adaptations of the set cover problem that capture the following intuition: If equipment cost (cost of sets) is expected to drop as bandwith demand (requirements of elements) is expected to rise, we wish to explore the option of bying equipment at a later point for a smaller cost. On the other hand, it may also be beneficial to buy slightly more equipment than what is necessary at present, if this additional equipment will cover a large amount of future demand at very low extra cost. The set cover problem with parameters evolving over time of Section 2 captures these considerations. Note that this is not an on-line model. Here the future is known in advance, but it introduces an additional dimension of complexity to the problem. We measure our performance by *comparing the approximation factor of the time variant of the problem to the best known approximation factor of the static problem.*

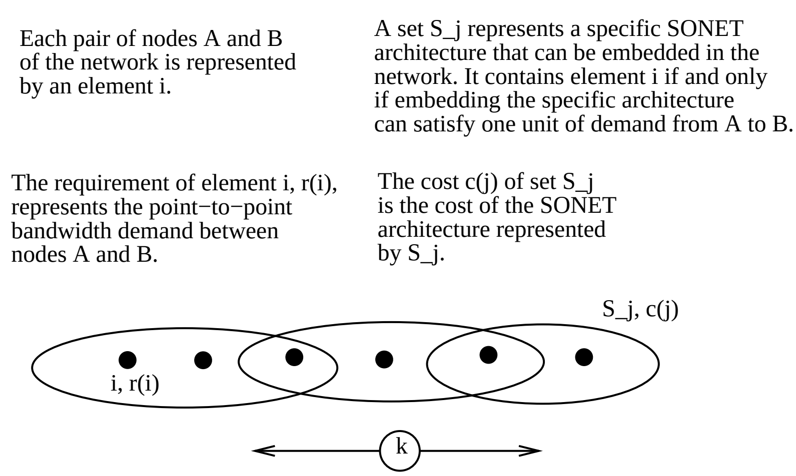

Figure 1 [6] describes a commercial SONET planning tool implementing an adaptation of the classical greedy algorithm for set cover. At a suitable level of abstraction, the heuristic of the planning tool can be viewed as follows: while there exists point-to-point bandwidth demand not covered by the SONET architectures selected so far, select a new architecture that is most efficient for the current iteration.

In Section 2 we give a reduction of set cover with time parameters to the standard set cover problem. This reduction carries over approximations and suggests a factor $\mathcal{H}(kT)$ algorithm for set cover with time parameters (e.g. following Chvatal [5]), where k is the cardinality of the largest set in the original problem, T is the number of points in time and kT is the cardinality of the largest set in the set cover instance that arises in the reduction.

In Section 3 we improve this factor to an optimal $\mathcal{H}(k)$ by introducing a new criterion for picking sets and adapting the standard duality based performance guarantee accordingly. The new criterion captures the following intuition: (a) cost suboptimal solutions should be considered for time t, if such solutions reduce substantially the additional cost required to obtain a solution for $t' > t$, (b) there is benefit in postponing picking sets that are not necessary for time t, if the cost at time $t' > t$ drops substantially and (c) while there is potential benefit in sets whose cost drops over time, this benefit should be counter-measured against the potential redundancy of such sets, if their effectiveness will be eventually covered by other sets that are necessary at earlier times.

The survivable network design problem and versions of the facility location problem are further examples of combinatorial optimization problems that arise repeatedly in automated network design [1] [9] [12] [13] [15]; major progress for these problems has been reported recently [3] [8] [10] [11] [16]. The survivable network design problem and the facility location problem have natural extensions with time parameters which remain open. We give an outline in Section 4.

2 Reduction to Set Cover and a factor $\mathcal{H}(kT)$ Approximation

The formal definition of SET COVER WITH REQUIREMENTS AND COSTS EVOLVING OVER TIME is as follows. There is a universe of n elements and a set system of m sets denoted by S_j, $1 \leq j \leq m$. Let k denote the maximum cardinality of a set $k = \max_{1 \leq j \leq m} |S_j|$. As in the standard set cover problem, elements have covering requirements and sets have costs. In this extended model however, requirements and costs evolve over T discrete points in time. In particular, for each time $t \, : \, 1 \leq t \leq T$, each element i needs to be covered by $r(i,t)$ sets that have been picked on or before time t, while picking one copy of set S_j at time t has cost $c(j,t)$. We assume that the future evolution of requirements and costs are known in advance and we consider a "buying" scenario where, if a set is picked at some point in time, it is never removed — for example, the purchase and installation of SONET architectures incurs tens or hundreds of millions of cost in equipment and management; once installed, such architectures are not removed. We wish to pick sets that satisfy the requirements at every point in time and are of minimal total cost. Formally, where $x(j,t)$ denotes the number of copies of S_j picked at time t, we have to solve:

$$\begin{array}{rll} \min & \sum_{t=1}^{T}\sum_{j=1}^{m} c(j,t)x(j,t) & \\ \text{subject to} & \sum_{t'=1}^{t}\sum_{j:i\in S_j} x(j,t') \geq r(i,t) & 1\leq i\leq n\ ,\ 1\leq t\leq T \\ & x(j,t)\in\ \aleph_0 & 1\leq j\leq m\ ,\ 1\leq t\leq T \end{array}$$

We first give a reduction to the standard SET COVER problem. In particular, for a universe of n elements, a set system of m sets denoted by S_j, $1\leq j\leq m$ and where $k=\max_{1\leq j\leq m}|S_j|$, in the SET COVER problem each element i has a covering requirement $r(i)$, picking one copy of set S_j has cost $c(j)$ and we wish to find a minimum cost collection of sets that satisfy the covering requirements:

$$\begin{array}{rll} \min & \sum_{j=1}^{m} c(j)x(j) & \\ \text{subject to} & \sum_{j:i\in S_j} x(j)\geq r(i) & 1\leq i\leq n \\ & x(j)\in\ \aleph_0 & 1\leq j\leq m \end{array}$$

Recall also the classical GREEDY ALGORITHM FOR SET COVER which repeatedly picks sets that reduce the total number of requirements at minimum average cost per unit of covered requirement. More specifically, for each element i with requirement $r(i)$ consider a stack of $r(i)$ chips labeled p_{ir}, $1\leq r\leq r(i)$, and consider a further labeling of each chip as either *covered* or *uncovered*. Define the *potential* of a set S_j with respect to such a labeling as the average cost at which the set covers uncovered chips: $\mathrm{P}(S_j)=c(j)/|\{i : \exists p_{ir}\in S_j \text{ and } p_{ir} \text{ is uncovered}\}|$. The algorithm then is:

GREEDY ALGORITHM FOR SET COVER
$x(j)=0,\ \forall j$;
label chip p_{ir} "uncovered", $\forall i,r$;
while there exist uncovered chips do
    set $\mathrm{P}(S_j)=|\{i : \exists p_{ir}\in S_j \text{ and } p_{ir} \text{ is uncovered}\}|,\ \forall j$;
    for some S_{j_0} that minimizes $c(j)/\mathrm{P}(S_j)$ set $x(j_0)=x(j_0)+1$;
    for all i,
        if some uncovered chip $p_{ir}\in S_{j_0}$ then
            label chip p_{ir} "covered" for exactly one such r;
        set $cost(p_{ir})=c(j_0)/\mathrm{P}(S_{j_0})$;

Now the following performance guarantee is well known and follows by duality considerations [5]:

Theorem 1. [Chvatal]. *The cost of the* GREEDY ALGORITHM FOR SET COVER *is within a $\mathcal{H}(k)$ multiplicative factor of the cost of any optimal solution: $\sum_{j=1}^{m} c(j)x(j)\leq\mathcal{H}(k)\cdot\mathrm{OPT}$.*

We may now give the reduction from SET COVER WITH PARAMETERS EVOLVING OVER TIME to SET COVER. See Figure 2. For each element i and each time t of SET COVER WITH PARAMETERS EVOLVING OVER TIME we introduce an element I_{it} with requirement $r(i,t)$ for SET COVER, and for each set S_j and each time t of SET COVER WITH TIME PARAMETERS we introduce a new set $S_{jt} = \{I_{it'} : i \in S_j \ , \ t' \geq t\}$ of cost $c(j,t)$ for SET COVER. Realize that the maximum set cardinality is kT, thus Chvatal's Theorem suggests that the greedy algorithm for set cover achieves a $\ln kT$ approximation factor. In the next Section we will modify the criterion for picking sets and achieve approximation factor $\ln k$. This is optimal in view of Feige's bound [7].

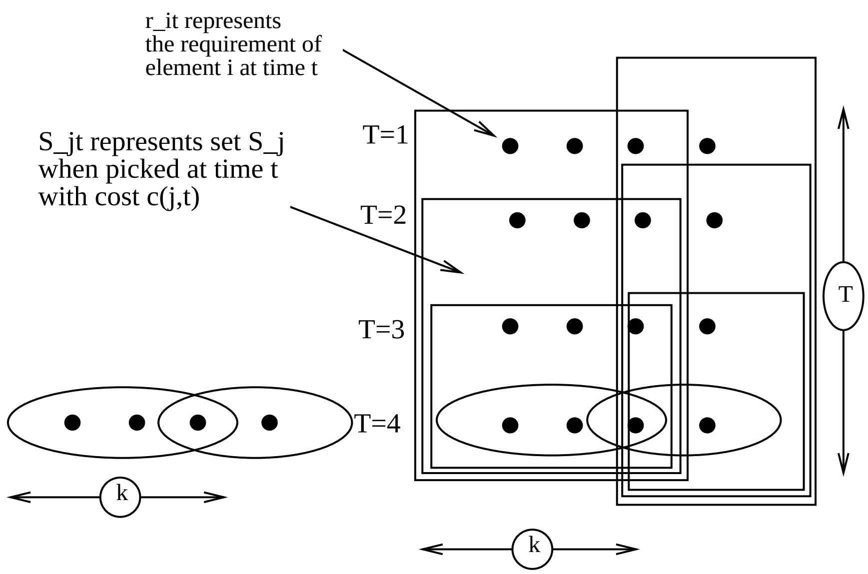

Figure 2 Indicating the the reduction of SET COVER WITH PARAMETERS EVOLVING OVER TIME to SET COVER. This reduction increases the size of the problem by a factor T.

3 A factor $\mathcal{H}(k)$ Modified Greedy Algorithm

How can we improve the GREEDY ALGORITHM of Section 2 when applied to SET COVER instances that arise from the reduction from SET COVER WITH REQUIREMENTS AND COSTS EVOLVING OVER TIME? Realize that a good heuristic for the latter set cover problem should capture the following: (a) cost suboptimal solutions must be considered for time t, if such solutions reduce substantially the additional cost required to obtain a solution for $t' > t$, (b) there is benefit in postponing picking sets that are not necessary for time t, if the cost at time $t' > t$ drops substantially and (c) while there is potential benefit in sets whose cost drops over time, this benefit should be counter-measured against the potential redundancy of such sets, if their effectiveness will be eventually covered by other sets that are necessary at earlier times. Realize further that the potential of sets S_{jt} arising in the reduction indeed capture (a) and (b). In particular, for (a), note that a set S_{jt} includes elements representing requirements for times $t' \geq t$ which may increase the potential of S_{jt}, while for (b), note that a substantial drop of the cost of a set S_j at time t' is represented by the cost of the set $S_{jt'}$ which must consequently become of high potential. However, the reduction does not capture (c). In particular, a set $S_{jt'}$ of very low cost could be chosen at first to satisfy the requirement of an element at time t'. On the other hand, this element may also have requirements at time $t < t'$ which will eventually result in the choice of sets $S_{j't}$, thus making the choice of $S_{jt'}$ redundant. The Modified Greedy Algorithm below modifies the set picking criterion to take into account (c).

For the descreption of the Modified Greedy Algorithm we need the following notation. See Figure 3. For each element I_{it} with requirement $r(i,t)$ consider a stack of $r(i,t)$ chips labeled p_{itr}, $1 \leq r \leq r(i,t)$. Define a *line* as a set of chips where i and r are fixed and t varies arbitrarily, and denote such a line by $L_{ir} = \{p_{itr} : 1 \leq t \leq T\}$. Say that $L_{ir} \in S_{jt}$ if and only if $t \leq \min\{t' : p_{it'r} \in L_{ir}\}$. Consider a further labeling of each line as either *covered* or *uncovered*. Define the *potential* of a set S_{jt} with respect to such a labeling as the average cost at which the set covers uncovered lines: $\mathrm{P}(S_{jt}) = C(j,t)/|\{i : \exists L_{ir} \in S_{jt} \text{ and } L_{ir} \text{ is uncovered}\}|$.

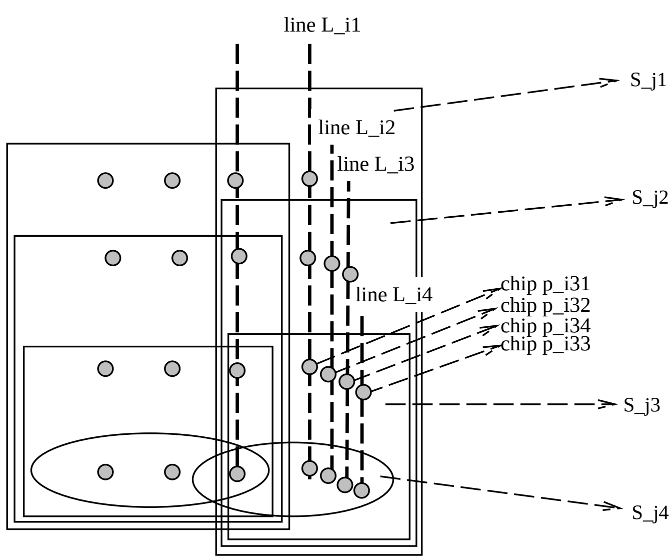

Figure 3 Indicating sets covering entire lines. For example, S_{j3} covers line L_{i4}, but S_{j3} does not cover lines L_{i3}, L_{i2} and L_{i1}.

Now we modify the set picking criterion as follows:

Modified Greedy Algorithm for Set Cover with Parameters Evolving over Time
$x(j,t) = 0,\ \forall j,t$;
label line L_{ir} "uncovered", $\forall i, r$;
while there exist uncovered lines do
    set $\mathrm{P}(S_{jt}) = |\{i : \exists L_{ir} \in S_{jt}$ and L_{ir} is uncovered$\}|$, $\forall j,t$;
    for some $S_{j_0 t_0}$ that minimizes $c(j,t)/\mathrm{P}(S_{jt})$ set $x(j_0,t_0) = x(j_0,t_0)+1$;
    for all i,
        if some uncovered line $L_{ir} \in S_{j_0 t_0}$ then
            label line L_{ir} "covered" for exactly one such r;
        set $cost(L_{ir}) = c(j_0,t_0)/\mathrm{P}(S_{j_0 t_0})$;

For the performance guarantee observe:

Lemma 2. *For all sets S_{jt}, $\sum_{i\in S_j} \max_{L_{ir}\in S_{jt}} cost(L_{ir}) \le \mathcal{H}(k)\cdot c(j,t)$.*

Proof. Assume without loss of generality that $S_j = \{1,\ldots,k'\}$. Also assume without loss of generality that for fixed t, among all lines $L_{ir} \in S_{jt}$ (as r varies), L_{ir_i} was the last line to be covered by the Modified Greedy Algorithm. Finally assume without loss of generality that, for all $1 \le i' \le i \le k'$, L_{ir_i} was covered at a previous or at the same iteration of the Modified Greedy Algorithm as line $L_{i'r_{i'}}$. Then, since S_{jt} could have covered L_{ir_i} at cost no more than $c(j,t)/i$ we have:

$$cost(L_{ir}) \le \frac{c(j,t)}{i} \quad , \quad 1 \le i \le k'$$

Thus,

$$\begin{aligned}\sum_{i\in S_j} \max_{L_{ir}\in S_{jt}} cost(L_{ir}) &= \sum_{i\in S_j} cost(L_{ir_i}) \\ &\le \left(\frac{1}{k'} + \frac{1}{k'-1} + \ldots + 1\right) c(j,t) \\ &\le \mathcal{H}(k)\cdot c(j,t)\ .\end{aligned}$$

Theorem 3. Performance Guarantee. *The cost of the solution of* Modified Greedy Algorithm *is within a* $\ln k$ *multiplicative factor of the cost of any optimal solution:* $\sum_{t=1}^{T}\sum_{j=1}^{m} c(j,t)x(j,t) \le \mathcal{H}(k)\cdot \mathrm{OPT}$

Proof. Follows by applying the Lemma to the sets of some optimal solution. In particular, let us fix an optimal solution, and suppose that it contains $x^*(j,t)$ copies of set S_{jt}. Then

$$\begin{aligned}\mathcal{H}(k)\cdot \mathrm{OPT} &= \mathcal{H}(k)\sum_{t=1}^{T}\sum_{j=1}^{m} c(j,t)x^*(j,t) \\ &\ge \sum_{t=1}^{T}\sum_{j=1}^{m} x^*(j,t) \sum_{i\in S_j} \max_{L_{ir}\in S_{jt}} cost(L_{ir})\ , \text{ by the Lemma}\end{aligned}$$

$$\geq \sum_{i=1}^{n} \sum_{r=1}^{\max_t r(i,t)} cost(L_{ir}) \text{ , by counting}$$

$$= \sum_{t=1}^{T} \sum_{j=1}^{m} c(j,t)x(j,t) \text{ , also by counting}$$

4 Survivabe Network Design and Facility Location

In the survivable network design problem we are given a weighted undirected graph and a requirement function over the cuts of the graph. We wish to pick a minimum cost subgraph such that each cut is crossed by at least as many edges as its requirement. Formally, for an undirected graph $G(V, E)$, $n = |V|$, cost function c on its edges: $E \xrightarrow{c} \mathcal{Q}_+$, cut requirement function f: $2^V \xrightarrow{f} \aleph_0$ and where $\delta(S)$ is the set of edges in E with exactly one endpoint in S, the survivable network design problem is expressed by the integer program below.

$$\begin{array}{ll} \min \sum_{e \in E} c(e)x(e) & \\ \sum_{e \in \delta(S)} x(e) \geq f(S) & \forall S \subseteq V \\ x(e) \in \{0, 1\} & \forall e \in E \end{array}$$

The survivable network design problem models "survivability/reliability" considerations and has a long history in practice and in theory [1] [4] [8] [10] [12] [13] [16]. The extension of this problem with time parameters involves a cut requirement function which increases with time and costs of edges which decrease with time. Formally, for an undirected graph $G(V, E)$, $n = |V|$, $[T] = \{1, 2, \ldots, T\}$ points in time, cost function c on the edges: $E \times [T] \xrightarrow{c} \mathcal{Q}_+$ and cut requirement function f: $2^V \times [T] \xrightarrow{f} \aleph_0$, the network design problem with requirements and costs evolving over time is expressed by the integer program below.

$$\begin{array}{ll} \min \sum_{t=1}^{T} \sum_{e \in E} c(e,t)x(e,t) & \\ \sum_{t'=1}^{t} \sum_{e \in \delta(S)} x(e,t') \geq f(S,t) & \forall S \subseteq V \text{ , } \forall t \in [T] \\ \sum_{t=1}^{T} x(e,t) \in \{0, 1\} & \forall e \in E \end{array}$$

Obtaining an approximation for the above problem is open. Such an approximation would be of concrete practical importance, for example, in upgrading the architectures of survivable CCSN, SONET, and WDM telecommunications networks.

The facility location problem has many variants; here we outline a representative one. There is a collection of facilities: each facility i can be opened at cost $f(i)$ for each unit of capacity $u(i)$. There is also a collection of cities: each city j has demand $d(j)$ which must be routed to open facilities. One unit of demand from city j can be routed to facility i at cost $c(i, j)$. We wish choose facilities that minimize the total cost:

$$\begin{array}{ll} \min \sum_{i \in F, j \in C} c(i,j)x(i,j) + \sum_{i \in F} y(i)f(i) & \\ \sum_{i \in F} x(i,j) \geq d(j) & \forall j \in C \\ u(i)y(i) \geq \sum_{j \in C} x(i,j) & \forall i \in F \ , \ \forall j \in C \\ x(i,j), y(i) \ \in \ \aleph_0 & \forall i \in F \ , \ \forall j \in C \end{array}$$

Now in the facility location problem with parameters evolving over time we have the demands increasing with time, and the costs of opening facilities and routing demands decreasing with time. We may write:

$$\begin{array}{ll} \min \sum_{t=1}^{T} \sum_{i \in F, j \in C} c(i,j,t)x(i,j,t) + \sum_{i \in F} y(i,t)f(i,t) & \\ \sum_{t'=1}^{t} \sum_{i \in F} x(i,j,t') \geq d(j,t) & \forall j \in C \ , \ \forall t \in [T] \\ u(i) \sum_{t'=1}^{t} y(i,t') \geq \sum_{t'=1}^{t} \sum_{j \in C} x(i,j,t') & \forall i \in F \ , \ \forall j \in C \ , \ \forall t \in [T] \\ x(i,j,t), y(i,t) \ \in \ \aleph_0 & \forall i \in F \ , \ \forall j \in C \ , \ \forall t \in [T] \end{array}$$

Obtaining an approximation for the above problem (or some suitable variant) is open. Such an approximation would be of importance, for example, in upgrading the architectures of ATM and Frame Relay telecommunications networks [15].

References

1. M. Ball, T. Magnati, C. Monma, and G Hemhauser, *Handbook in Operations Research and Management Science*, Vol 8, North-Holland (1992).
2. A. Borodin and R. El-Yaniv, Online Computation and Competitive Analysis, Cambridge University Press, 1998.
3. M. Charikar, S. Guha, E. Tardos, and D. Shmoys, "A constant-factor Approximation Algorithm for the k-median Problem", to appear in STOC Proc., 1999.
4. D. Hochbaum, Approximation Algorithms for NP-Hard Problems, PSW Publishing Company, Boston MA, 1997.
5. V. Chvatal, *A Greedy Heuristic for the Set Covering Probl em*, Mathematics of Operations Research, 4 (1979), pp. 233-235.
6. S. Cosares, D.Deutch, I. Saniee, and O. Wasem, "SONET Toolkit: A Decision Support System for the Design of Cost-Effective Fiber Optic Networks, *Interfaces* Vol25, Jan-Feb 1995, pp.20-40.
7. U. Feige, "A Threshold of $\ln n$ for Approximating Set Cover", in Proceedings of *STOC* 1996.
8. M. Goemans, A. Goldberg, S. Plotkin, D. Schmoys, E. Tardos, and D. Williamson, *Improved Approximation Algorithms for Network Design Problems*, Proc. SODA 94.
9. 3rd *INFORMS Telecommunications Conference*, Special Sessions on "Network Design Aspects about ATM" and "Design and Routing for Telecommunications Networks", May 1997.
10. K. Jain, "A Factor 2 Approximation Algorithm for the Generalized Steiner Network Problem", FOCS Proc., 1998.
11. K. Jain and V.V. Vazirani, "Primal-Dual Approximation Algorithms for Metric Facility Location and k-Median Problems", submitted, also in http://www.cc.gatech.edu/fac/Vijay.Vazirani.
12. T. Magnati and R.T. Wong, "Network Design and Transportation Planning": Models and Algorithms, *Transportation Science* 18, pp. 1-55, 1984.

13. M. Mihail, D. Shallcross, N. Dean, and M. Mostrel, *A Commercial Application of Survivable Network Design*, Proc. SODA 96.
14. R.Motwani and P. Raghavan, Randomized Algorithms, Cambridge University Press, 1995.
15. Iraj Saniee and Dan Bienstock, "ATM Network Design: Traffic Models and Optimization Based Heuristics", 4th *INFORMS Telecommunications Conference*, March 1998.
16. D. Williamson, M. Goemans, M. Mihail, and V. Vazirani, *A Primal-Dual Approximation Algorithm for Generalized Steiner Network Problems*, *Combinatorica* 15:435-454, December 1995.