Automated Antenna Positioning for Wireless Networks

Amit Dvir, Yehuda Ben-Shimol, Yoav Ben-Yehezkel, Michael Segal
Department of Communication Systems Engineering, Ben Gurion University, Israel
Boaz Ben-Moshe
School of Computing Science, Simon Fraser University, Canada

***Abstract*— This article addresses a real-life problem - obtaining communication links between multiple base stations sites, by positioning a minimal set of fixed-access relay antenna sites on a given terrain. Reducing the number of relay antenna sites is considered critical due to substantial installation and maintenance costs. Despite the potential significant cost saving by eliminating even a single antenna site, a hardly optimal manual approach is employed due to the computation complexity of the problem. We suggest several alternative automated heuristics, relying on terrain preprocessing to find educated potential points for positioning relay stations. A large-scale experiment was conducted showing that the saving potential increases when more BSs are required to be interconnected and in any case is better than the one obtained by a human expert.**

I. Introduction

In wireless fixed access networks (WFAN) a set of base stations (BSs) provide communication services to remote fixed users. Each BS antenna is planned to serve only its local potential subscribers. There is no need for continuous coverage of a given area by the set of BSs, especially in rural areas. However, interconnections between the BSs are required in order to form a connected communication network where subscribers are able to communicate one with each other. Here we assume that the interconnections between the BSs are obtained partially by positioning additional relay stations (RSs) at specific sites and using radio communication links. This work is targeted especially rural areas with sparse population, where the use of wired links between BSs is considered impractical.

The installation and maintenance costs of the RSs motivate the minimization of the number of RSs sites. Despite the significant cost saving by eliminating even a single antenna site, a hardly optimal manual approach is employed due to the computation complexity of the problem (Max-SNP-hard). Practical solutions to problems of antenna positioning of WFAN are usually semi-automated. A professional system designer, experienced in the essentials of installing WFAN, suggests an antenna-positioning strategy. Interconnecting microwave antennas (as done in this work) requires both Line Of Sight (LOS) and a distance limit between any two antennas. Some antenna types may even work when LOS is broken, although path loss increases significantly. As the authors are aware (after long discussion with industrial experts) for a given rural area of 50 Km^2, a preliminary design is obtained in several days, mixing human and computerized decision. The final design is obtained after a work of additional several weeks with much higher human involvement due to the lack of highly automated decision supporting tools.

An informal definition of the discussed problem follows: Given a terrain T and a set of BSs, connect all BSs by positioning the smallest set of RSs. An RS can connect any number of BS/RS and a BS can also serve as a RS. Terrain properties need to be considered. This problem is referred to as CMBS (Connecting Multiple fixed access BSs.)

Some solutions for automation of antenna-positioning problems in cellular networks were suggested [1], [2], [3], [4], [5]. In addition, the CMBS problem is often raised in the context of minimizing the number of BSs that can serve all (or most of) the network clients [6]. Solutions to the problem of serving a maximum surface of a geographical area with a minimum number of Base Transceiver Stations were also given [7].

Most of our suggested heuristics are based on variations of minimal spanning trees, and on shortest paths. These methods have been studied extensively in the context of facility location problems [8], [9], [10], [11]. We suggest two other heuristics based on a new grading scheme defined specifically for solving the CMBS problem.

The CMBS problem resembles the Steiner Tree Problem (for summary and references see [12]). The analogous problem in this paper was to minimize the total number of Steiner nodes used to solve a Steiner Tree Problem. To the best of our knowledge there are no previous studies on the CMBS problem as defined above. However, one should be aware not to confuse between the CMBS problem and the Steiner problem in networks. This is discussed in more detail in section 3.

To solve the design issues that were raised above, the goal of the present research is to establish a fully-automated antenna positioning algorithm for WFAN, outperforming human approaches, in terms of both selecting fewer RS antennas, and achieving this strategy in a much shorter time. One should notice that solutions of equal number of BSs serving the same number of subscribers may still be subjected to a significant difference from their CMBS point of view (i.e., the number of RSs). Therefore, we believe that CMBS algorithms should be used as complementary objectives of GIS-RF applications [13], [14]. Usually, these applications focus on optimization algorithms that attempt to minimize the number of BSs antennas while providing service to the subscribers. However, interconnecting BS antenna sites directly or through a minimum set of relay stations is not dealt with at all using these applications.

The rest of the paper is organized as follows. Section 2 presents a solution for connecting two BS antenna sites using the smallest set of RSs. Section 3 discusses the NP-hardness of the CMBS problem. We show that the CMBS problem is ‘set-cover’ hard to approximate. This section is also dedicated to heuristics that connect multiple (more than two) BSs, as well as to some examples of the CMBS solver application we developed; Section 4 provides details of the large scale experiment we conducted. Recommendations regarding future work are given in section 5.

II. The Point-To-Point (P2P) Case

Here we discuss the simplest CMBS case: find a path with a minimal number of RSs that connects a source BS - BS_s to a target BS - BS_t. In spite of the “simplicity” of the problem, many fundamental issues arise in this problem as well as in the more general case where more than two BSs should be interconnected.

A. Terrain Preprocessing

Since the proportion of the terrain models in our problem is huge (thousands of square km), it is necessary to preprocess it in order to reduce the input size by an order of magnitude, before applying various facility location algorithms on it. To determine whether two antennas, positioned on two arbitrary points on the map can communicate directly, it is required to calculate the accumulated path-loss between these two points and to compare it with the sensitivity threshold of the receiver. Calculating all pairs of points on a large-scale map is infeasible. Therefore, we applied a preprocessing algorithm that attempts to preserve the LOS property between potential antenna locations.

Two methods for finding potential RSs’ positions were used. In the first method a network of $N \times N$ grids was overlaid on the map, where N is a user-defined parameter. From each grid, the two highest grid points among all N^2 points in that square are selected as potential antenna positions. The second method is based on a variation of using the map as an input to a convex hull algorithm.

B. Positioning remote stations

The result of terrain preprocessing with one of the methods described above, and then considering path loss, a visibility graph $VG = (V, E)$ is constructed, with V is the set of all potential points for relay stations together with BS_s and BS_t. An edge $e = (v_i, v_j)$ denotes that the Euclidian distance between nodes v_i and v_j is smaller then a given threshold and they maintain a line of sight (LOS).

All the shortest paths are given as suitable alternative solution to select from, since when testing these paths in the field, additional constraints, such as restricted military zones, deep water reservoirs, swamps, etc., which cannot be used or are impractical for antenna positioning, are posed. The set of all shortest paths from BS_s to BS_t is discovered by a combined variation of the BFS and DFS algorithms, reffered as Breadth First Depth Next (BFDN) algorithm.

III. Establishing Communication Links Among Multiple BS

In this section we deal with solutions to the more general problem: connecting multiple BSs with a minimal set of RSs (i.e., CMBS). One should be aware not to confuse between the CMBS and the minimal Steiner tree problem. The minimal Steiner tree problem is concerned with finding a tree of minimal weight that spans a predefined set of terminals (BSs for our case) with optional additional non-terminal nodes called Steiner nodes (RSs in our case) [13,14]. The weight of the tree is the sum of edge weights of the edges in the tree. The CMBS problem is concerned with minimizing the number of non terminal nodes. The intractability of the CMBS problem is proved to be NP-complete with a formal reduction from the CMBS problem to the general set cover problem. This reduction shows that the CMBS problem is ‘set cover hard’. Several heuristics are developed and implemented in an attempt to resolve the CMBS problem and present solutions to one scenario using the application we developed for the present research.

A. NP-Completeness (Max SNP-hard) Proof

The general set-cover problem is defined as follows: let S be set of all integers from 1 to n, let $S^* = \{S_1, ..., S_m\}$ be a set of m subsets of S. Our goal is to find the smallest subset of S^* such that its union is S. The k-set cover problem is a set-cover problem for which the size of the maximal set in S^* is bounded by k. The k-set cover problem is known to be NP complete for $k \geq 3$ [15], and unapproximatable within $(1-\epsilon)\ln(k)$ for any fixed $\epsilon > 0$, that is, MAX SNP-hard [16] (assuming P is not NP).

Lemma: The CMBS problem can be reduced from the set-cover problem using a ‘gap-preserving’ reduction that preserves the approximation ratio (using visibility over the terrain as a connection condition).

Any valid solution to the CMBS problem implies a valid solution to the original set cover problem, moreover any subset of $S_1, ..., S_m$ which does not connect all S (i.e., not a valid CMBS solution), also does not cover all S. To conclude, if there was an algorithm, which has a constant factor approximation for the CMBS problem; it could have been used to compute a constant factor approximation for the set-cover problem – contradiction.

Theorem: (proof ommited) the CMBS problem is MAX-SNP-hard, and therefore cannot have an approximation ratio better than $O(1)\ln(k)$ (where k is the maxima over the size of subsets $S_1, ..., S_m$. Namely, the CMBS is set-cover hard.

B. Heuristics for the CMBS problem

We will address the CMBS problem in terms of graph theory: given a set $B = \{b_1, ..., b_n\}$ of n BSs and a set of $R = r_1, ..., r_m$ of RSs, select the smallest subset $SRS \subseteq R$, such that $b_i \in B$, are in a single connected component.

C. Minimal Spanning Tree Based Heuristics

The first class of heuristics that we developed and tested was based on minmal spanning tree (MST) like algorithms. In some of these heuristics we calculate on VG all the shortest paths of potential RSs between any two BSs using *BFDN*, as a basis for our solution. Simplicity, ease of implementation and short running time are the main advantages of these heuristics.

Trivial Minimal Spanning Tree (*T-MST*): Start with the minimal shortest path distance (in terms of number of RSs in the path) among any two BSs. Then select the BS with the shortest path to any of the already connected BS nodes. Add that distance to the total sum of the distances and add the new BS to the set of connected BSs. Repeat adding unconnected BSs in an order of increasing shortest paths until all BS nodes are connected. The result is an upper bound for the number of required RS, because the summation of distances includes RS positioned on more than one of the shortest paths more than once. *T-MST* is used as a fast upper-bound approximation for the number (not positions) of RSs required to solve the CMBS problem.

The following set of heuristics for the CMBS problem are concerned with the positioning of the RSs.

The *Simple Minimal Spanning Tree* (*S-MST*) is the first heuristic that actually allows to position RS antennas on the given terrain. The *S-MST* has three variations:

1) Select a predefined initial BS to begin with. Then calculate the shortest path to its nearest neighbor. If several such paths are available, pick the first one on the list. In each step henceforth, connect the first BS closest to the already connected component (any already connected RS or BS). 2) The same as variant 1 with the following distinctions: instead of predefining the initial BS, select it randomly, and, if more than one shortest path to the closest connected component exists, randomly select one of these shortest paths. 3) This variation resembles the second variation, however, rather than connecting the entire shortest path to the already connected component, add only a single RS to the solution, until all BS are connected.

Random Batch (*RB-MST*): *RB-MST* simply apply the *S-MST* heuristic several times: each time the initial BS, one of the shortest paths or the added RS and are selected randomly. *RB-MST* returns the best results.

Greedy Improvement (*GI-MST*): Start with a randomly selected BS (e.g., BS_a) and find all the shortest paths to its nearest BS (e.g., BS_b) using the *BFDN* algorithm. Iterate through these shortest paths, one at a time, and grade these paths where all the RSs on that path are considered as BSs. The greedy grading function can be the number of required RS by using *T-MST*, *S-MST* or a combination of them. The highest graded is selected.

Since the number of shortest paths can be large in practical problems another shorter version of *GI-MST* was implemented, where the grading function is carried out only on the RS-min-cut set between any two BSs on VG.

Greedy & Random (*GR-MST*): *GR-MST* runs *GI-MST* several times, selecting the *GI-MST* parameters at random. The initial BS, which RS-min-cut is set to select if there are several of the same size, the BS destination to apply *GI-MST* if several shortest paths reach more than one BS, serve as examples to the random selection we can make. In general, whenever a selection is made, it is done randomly. *GR-MST* returns the best results.

D. Relay Stations Grading (RSG) Heuristics

The other class of heuristics that was developed and tested is based on grading RSs. The following two heuristics are based on the fact that the optimal solution might contain a RS, which does not reside on any of the shortest paths from any BS to any other (closest) BS (see Figure 1).

Basic Relay Station Grading (*B-RSG*): *B-RSG* enables to locate RSs that are not necessarily on any shortest path (between two BSs). *B-RSG* grades candidate RSs based both on their reuse property and on their path distances. The fact that the number of paths on which the RS can be used improves its grade and allows it to break loose from shortest path based solutions. Several grading functions were tested and compared. The result of this comparison following to provide the required effect.

The *B-RSG* algorithm works as follows:

While all BSs are not yet connected:

1) *Initialize the grades of all RSs.*
2) *Grade all RSs.*
3) *Select the best-graded RS for the solution.*
4) *In the next iteration, connect the stations that were directly connected to the best-graded RS*

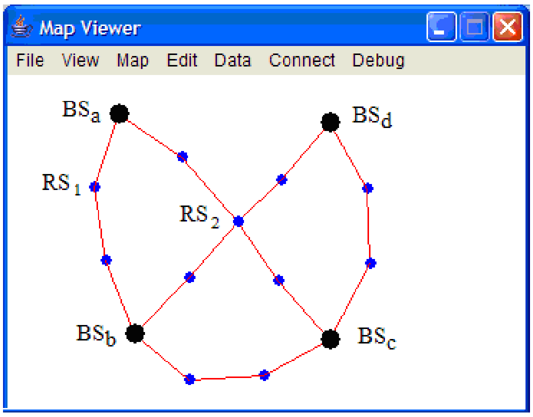

Fig. 1. Four BSs (big circles) and 11 RS (small circles) example: The optimal solution is 5 (using the middle RS), while any solution based on shortest path like method is 6 (using the outer RSs).

It should be noted that *B-RSG* usually selects RSs lying on shortest paths, in which case it does not improve the MST-based heuristics (but rather produce similar results). In addition, its run-time is longer than MST-based heuristics. However, *B-RSG* is exceptionally useful in rare situations where a solution can be found on none of shortest paths.

Hybrid-RSG Heuristic (*H-RSG*): *H-RSG* combines both *B-RSG* and *GI-MST* heuristics. It is based on the idea that the *B-RSG* can serve as a filtering mechanism to select a relatively small and meaningful set of candidates (e.g. the RS whose grade is greater than a given percentage of the highest grade). Then, the *GI-MST* is applied on that small set of RS candidates. Both grades are weighted and the RS with the highest weighted grade is marked as an RS in the solution. This grading method is repeated and terminates when all BS are connected. As with other random based heuristics, this heuristic also runs several times and returns the best results. When more than one RS receives the same weighted grade we use a partial version of the "Branch & Bound" algorithm [17]. Figure 2 illustrates how the VG is displayed with and without a DEM background. Figure 3 displays a single scenario with 25 BSs and the solutions obtained by each heuristic.

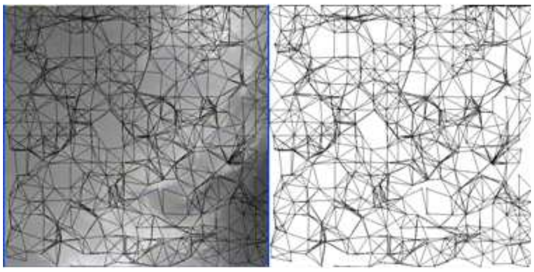

Fig. 2. VG of 450 RSs (10m antenna heights and 10 km transmission range). On the left: VG displayed over $100 \times 100km^2$ DEM (the lighter the higher). On the right: the same VG displayed with the DEM hidden.)

IV. Large-Scale Experiment

A. Experiment Description

The objectives of the experiment were to test whether the suggested heuristics actually work, and to compare the results they obtained. An experiment scenario is described by a vector of one permutation of the scenario parameters (i.e., number of BS, number of potential RSs, antenna heights and transmission ranges). Many different possible sets of scenarios were tested. The experiment consisted of a total of approximately 7,000 scenarios (about one quarter of the possible permutations) using all the heuristics detailed in Section 3. Non-realistic cases were eliminated.

17 high-resolution elevation maps (100×100 km^2), each representing a different type of terrain: flat, hills, dunes, mountains, lakes, etc. were used. We positioned the selected number of BSs on the map either by selecting a subset of the potential RSs (which is a reasonable strategy, because BSs are usually positioned on high altitude points) or at random. For each permutation of the parameters we tested 10 different positions for BSs.

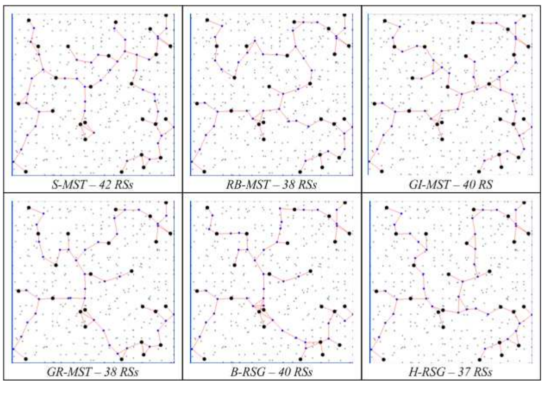

Fig. 3. An example of a single scenario solved by all the suggested heuristics

B. Experimental Results

In order to compare between the performances of the heuristics a useful method was implemented, called the *Best Of All* (*BOA*). *BOA* picks out for each scenario the best result obtained by any of the heuristics. Using this method we can compare results obtained by each heuristic to the minimal number of RSs obtained by any of the other heuristics.

A reasonable assumption is the decrease in the average number of required RSs per BS while increasing the number of BSs. The rational behind this assumption is the decrease in the average distance between BSs and also because BSs can also serve as RSs.

Comparing between Heuristics: To compare between heuristics we calculated the normalized effectiveness of each heuristic solution for each scenario. Normalization can be performed because both upper and lower solution bounds are available (*T-MST* and *BOA*, respectively). We used $Grade(h) = 1 - (\mathrm{RS}(h) - \mathrm{RS}(BOA))/(\mathrm{RS}(\text{T-MST}) - \mathrm{RS}(BOA))$ as the grading function, where h denotes a heuristic and $\mathrm{RS}(x)$ denotes the number of RS obtained by the parenthesized heuristic. This grading scheme ensures that the more RSs in a heuristic's solution, the lower the grade it gets. The grades of the best and worst heuristics are always 1 and 0, respectively. Then, we calculated the Cumulative Density Function (CDF), with the random variable X being the grade.

Figure 4 presents a bar-chart showing the mean number of RSs required to connect 10, 25, 50 and 100 BSs. The graph compares between results obtained by each heuristic (not ordered by their presentation order in Section 3, but rather by their mean effectiveness). The graph indicates that H-RSG was the most effective heuristic and is only slightly less effective than the BOA. B-RSG produced almost always the best results compared to all the single iteration based heuristics (T-MST, S-MST and GI -MST). However, B-RSG also suffers from the longest runtime among the single iteration based heuristics. It takes the same time to run B-RSG once or to run RB-MST many times and usually obtain better results. A reasonable assumption is the decrease in the average number of required RSs per BS while increasing the number of BSs, for the same given area under investigation. The rational behind this assumption is the decrease in the average distance between base stations and also because BSs also serve as RSs. This assumption was found accurate, using all heuristics.

Figure 5 indicates that for example, *S-MST* grade is lower than 30% in approximately 70% of the cases, whereas *H-RSG* grade is lower than 90% in approximately 30% of the cases (suggesting the more meaningful notion that in the complementary 70%, its grade is greater than 90%). In general, a heuristic whose line is lower is better. An interesting notion is that *B-RSG* and *GI-MST* intersect at a grade of approximately 80%, though *B-RSG* seems better for most of the tested cases.

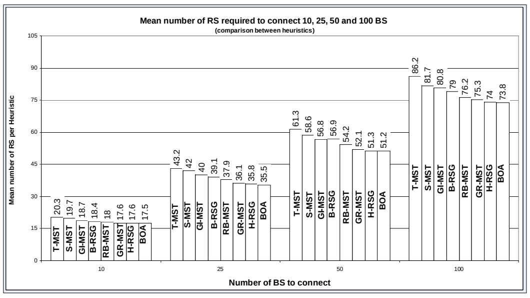

Fig. 4. Mean number of RSs required to connect 10, 25, 50 and 100 BSs - comparison between heuristics.

This, however, might explain the fact that for connecting 50 BSs, the solution obtained by *B-RSG* required 0.1 more RSs than the solution obtained by *GI-MST* (that was the only case where *B-RSG* required more RSs than *GI-MST*). The grades of the random heuristics were the highest, and *H-RSG* is the best of them all (which correlates to the fact that it required the lowest number of RSs for any number of BSs to connect. *Potential Savings*: The results suggest that

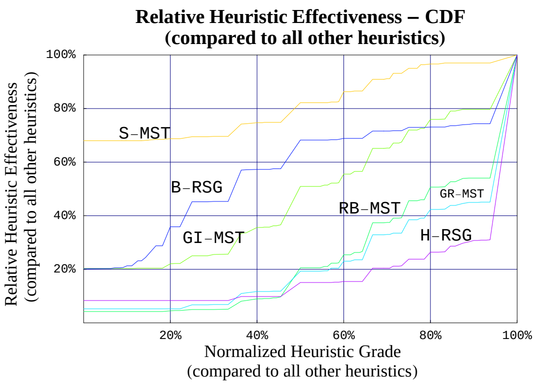

Fig. 5. Relative Effectiveness of Heuristics - CDF.

a significant saving potential can be obtained by applying the presented heuristics for positioning RS antennas. For example, in positioning RSs to support communication between 50 BSs, there was an average difference of up to 10.1 RSs when comparing positioning strategies of *T-MST* and *BOA* (potential saving of approximately 20% in the number of RSs). Table 1 summarizes the average running time (in $mSec$) of the different algorithms for various numbers of BSs and various number of potential RSs positions. As can be seen, even the best algorithm that iterates multiple times (i.e., H-RSG), finishes its computation for the most demanding task in about 6 seconds. Thus, running time is not serving as a limiting factor.

Analysis Usefulness: We wish to emphasize that the analysis suggested in this article provided solutions to approximately 98% of the tested cases. This statement is valid even when BSs are located randomly We usually do not find a solution, when we use too little potential RS with low antenna heights and short transmission ranges. Using more RSs, higher antennas or longer transmission ranges usually overcomes this problem.

Algorithm	Number of BSs/Number of potential RSs			
	10/500	25/1000	50/2500	100/5000
S-MST	< 10	< 30	52	140
GI-MST	35	107	287	532
B-RSG	72	183	530	1236
RB-MST	402	1255	2743	5943
GR-MST	393	1214	2717	5901
H-RSG	424	1250	2813	6012

TABLE I

Comparison of running times of the different proposed algorithms for different cases of number of BSs and number of potential RSs.

V. Future Work

Further research may include additional heuristics and fine-tuning to the ones suggested here. Generalizing the CMBS problem to a weighted CMBS problem, in which the objective function is weighted, subject to the same constraint of connecting all BS seems as a good direction for research. One obvious application for this generalization is to solve the CMBS problem while minimizing the installation cost, for example by assuming different costs to different RS types. We also think that the algorithm for selecting potential RS antenna positions can be improved in terms of runtime complexity and approximation factor. Another aspect of the CMBS problem is to compute a lower bound for a given input by employing continuous linear programming based methods.

References

[1] M. Vasquez and J. K. Hao. A heuristic approach for antenna positioning in cellular networks. *Journal of Heuristics*, 7:443–472, 2001.

[2] H.R. Anderson and J.P. McGeehan. Optimizing Microcell Base Station Locations Using Simulated Annealing Techniques. In *Proc. of the IEEE Vehicular Technology Conference*, 1994.

[3] K. Tutschku. Demand-based radio network planning of cellular mobile communication system. In *Infocom 98*, 1998.

[4] H. D. Sherali, C. M. Pendyala, and T. S. Rappaport. Optimal Location of Transmitters for Micro-Cellular Radio Communication System Design. *IEEE JSAC*, 14:662–673, 1996.

[5] J. K. Hao, R. Dorne, and P. Galinier. Tabu Search for Frequency Assignement in Mobile Radio Networks. *Journal of Heuristics*, 4:47–72, 1998.

[6] B. Ben-Moshe. *Geometric Facility Location Optimization*. PhD thesis, Ben Gurion University, Israel, 2004.

[7] P. Calegari, F. Guidec, and P. Kuonen. Combinatorial optimization algorithms for radio network planning. *Theoretical Computer Science*, 263:235–245, 2001.

[8] J. S. B. Mitchell. *Handbook of Computational Geometry*, chapter 15. Elsevier, 1998.

[9] P. K. Agarwal, S. Har-Peled, and M. Sharir. Approximate Shortest Paths on a Convex Polytope in Three Dimensions. *J. ACM*, 44:567–584, 1997.

[10] S. Har-Peled. Constructing approximate shortest path maps in three dimensions. *SIAM J. Comput.*, 28:1182–1197, 1999.

[11] X. Tan. Approximation algorithms for the watchman route and zookeeper's problems. *J. Discrete Appl. Math.*, 136:363–376, 2004.

[12] P. Winter F. K. Hwang, D. S. Richards. *The Steiner tree problem*. Elsevier, 1992.

[13] *Optimizing the Wireless World*. http://www.schema.com.

[14] Nir – radio planning and optimization tool", 2004. http://www.hexagonltd.com.

[15] R. M Karp. *Reducibility among combinatorial problems, Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972. Enter text hereNew York,.

[16] U. Feige. A threshold of ln n for approximating set cover. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 314–318, 1996.

[17] M. A. Efroymson and T. L. Ray. A branch and bound algorithm for plant location. *Operations Research*, 14, 1996.