EXPERIMENTAL ANALYSIS OF LOCAL SEARCH ALGORITHMS FOR OPTIMAL BASE STATION LOCATION

Bhaskar Krishnamachari, Stephen B. Wicker
Wireless Multimedia Laboratory
School of Electrical Engineering
Cornell University
Ithaca, NY 14853, U.S.A
{bhaskar, wicker}@ee.cornell.edu

Abstract

Search techniques such as Genetic Algorithms, Simulated Annealing, Tabu Search and Random Walk Algorithms have been used extensively for global optimization. This paper presents an experimental analysis of the performance of these algorithms for the problem of selecting the optimal location of base stations in a mobile communication system. The effect of varying the value of important parameters for each algorithm is investigated to determine suitable values. The algorithms are then compared to each other using a common neighborhood definition to ensure fairness. Intuitive explanations are provided for the results.

1 Introduction

In the planning stages of a new mobile communications network, the locations of the radio base stations need to be determined. Information concerning the radio propagation characteristics of the service area, and a list of potential sites where base stations can be located is used to design cells in such a way as to minimize the cost of equipment (i.e. number of base stations used), while maximizing service (i.e. the radio coverage provided in the area). This problem has been shown to be NP-complete by analogy to the Minimum Dominating Set (MDS) problem [1].

Local search techniques[1], such as Genetic Algorithms (GA), Simulated Annealing (SA), Tabu Search (TS) and Random Walk Algorithms (RW) are all known to yield near-optimal solutions within a reasonable time for such difficult optimization problems [3]-[5]. However due to the complex interactions between optimization problems, corresponding neighborhood definitions, and various algorithm parameters, their performance and relative suitability for a problem can only be determined via experimental analysis. Genetic Algorithms and Simulated Annealing have been independently applied for optimal base station location using slightly different problem formulations in [1] and [2] respectively. In this paper the performance of GA, SA, TS and RW is examined for various parameter values and compared on a sample problem instance.

2 Problem Description and Parameters

A square service area is selected and divided into a grid of 100x100 points spaced 1 unit distance apart. From this set of 100,000 possible points, 51 locations where base transmitting stations could be potentially be located were generated randomly with a uniform distribution. These 51 possible sites can be seen in Figure 1. For real world problems, this initial list of potential base station locations would depend upon geographical considerations, physical constraints and the ease with which radio transmitters can be installed in those locations.

[1] There is some confusion in the literature regarding nomenclature. These techniques have also been referred to as global, stochastic, meta-heuristic, or neighborhood-based search by different authors.

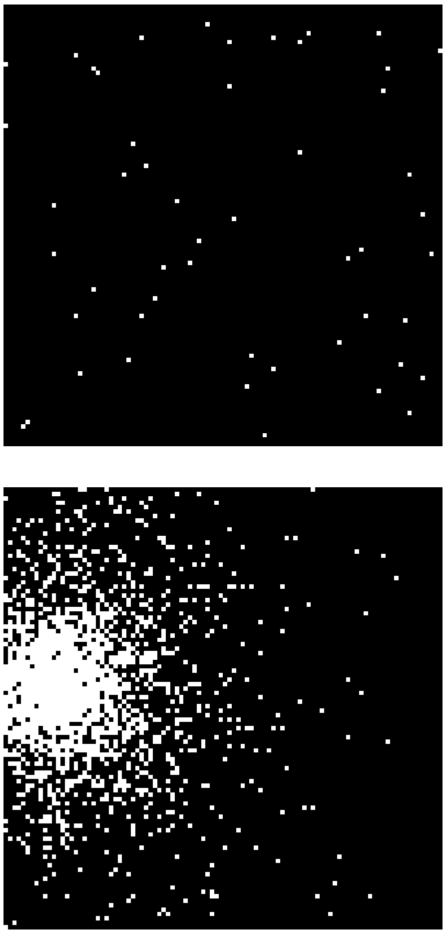

Figure 1: 51 Potential Base Station Locations (top) and Sample Radio Coverage for One Potential Location (bottom)

In a real scenario, the radio propagation characteristics of the area would have to be determined via sophisticated ray-tracing simulations or field measurements. For the purposes of analyzing the performance of optimization techniques, a service coverage area is obtained for each potential location using an uncorrelated log-normal shadow fading model for radio propagation. In this model, the power loss in dB at a distance d from the base station is given by the equation

$$P_{loss}(d) = A + Blog(d) + N \tag{1}$$

where N is a zero-mean Gaussian random Variable with a variance σ^2. The values of these parameters used in this paper are: A = 50, B = 40, σ^2 = 10. From each possible base station location, P_loss, the radio path loss, is computed for all 100,000 points. Then, a cutoff value $P^* = 100dB$ is chosen for the path loss, such that if $P_{loss} \geq P^*$ for any point it is deemed that the radio coverage is not sufficient. The bottom of figure 1 shows the radio coverage for a transmitter located at one of the 51 possible locations (white pixels indicate areas with radio coverage from the transmitter). If all 51 possible locations were selected, the net coverage would be 100% but there would be a significant overlap between different base stations. Therefore far fewer than 51 base stations are actually required to provide good coverage in the area.

For this optimization each point in the search space would represent which subset of the 51 potential locations are actually chosen for installing base stations. This can be represented by a 51 bit binary string with each bit corresponding to one of the potential locations. Each bit is one if a base station is placed at the corresponding location and zero otherwise. The cost function chosen is similar to that used in [1]:

$$f = k\frac{N_{BS}}{R^{\beta}} \tag{2}$$

where N_{BS} is the number of base stations selected, R is the radio coverage provided by these selected stations (the percentage of the service area that is covered by at least one base station), β reflects the weight attached to maximizing coverage as opposed to minimizing the number of base stations, and k is a scaling factor. The values of the parameters used in this paper are $\beta = 3$, $k = 10^4$.

All local search algorithms are characterized by the existence of a neighborhood definition which describes how the search proceeds from one point in the search space to another. In the algorithms described here, the neighbor is generated by randomly inverting one of the 51 bits that define the current point. Thus the neighborhood of a given solution consists of all points at a Hamming distance of 1.

3 Effect of Algorithm Parameters on Performance

The best parameter for each algorithm is determined by comparing different settings and running each for 10 runs consisting of 1000 cost function evaluations. All runs are started at the same initial point in the search space with a cost function evaluation of 0.284

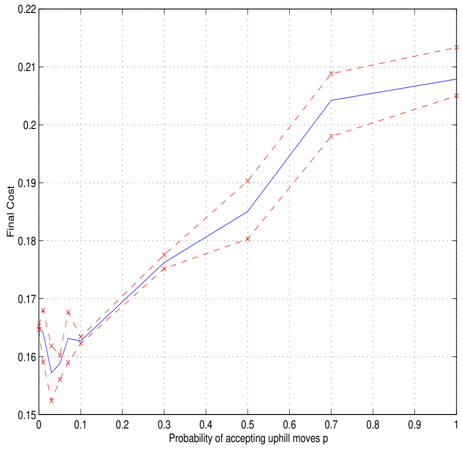

Figure 2: Performance of RW

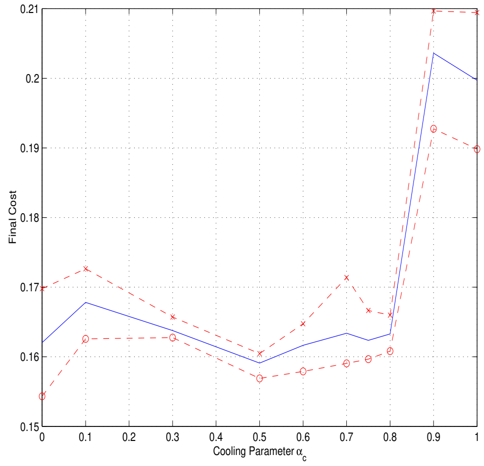

Figure 3: Performance of SA

(26 base stations providing a net coverage of 97.1%). For more details on the implementations of these algorithms, see [6].

Random Walk: The Random Walk Algorithms are perhaps the simplest of all local search techniques. At each iteration a neighbor that is generated is accepted unconditionally if it has a lower or same cost function as the current point, and conditionally with a probability p if it has a higher cost function. Hence these algorithms range from a purely greedy search ($p = 0$) to a completely random search ($p = 1$). Figure 2 shows the average (solid), minimum and maximum (dotted) final costs obtained. An acceptance probability level $p = 0.03$ is found to be optimal, which implies the existence of local minima that may be escaped by allowing a small percentage of uphill moves. If p is too high, on the other hand, the search becomes unfocused and the algorithm is less likely to locate minimum cost solutions efficiently.

Simulated Annealing: An SA algorithm with a geometric cooling schedule ($T_{new} = \alpha_c T_{old}$) and the Metropolis acceptance criterion is used. The initial temperature $T_o = 10\sigma_\infty$, where σ_∞ is the standard deviation of costs obtained by a purely random search. The number of different temperature levels is fixed at 20, with 50 iterations of the algorithm at each temperature. If the cooling parameter α_c is too low, then the SA would go down to zero temperature quickly and perform a greedy search. If, on the other hand, α_c is too high, the algorithms behaviour would be more like that of a purely random search. To identify the optimum value of α_c the algorithm was tested for values ranging from 0 to 1.

The results are shown in figure 3 which shows the average, minimum and maximum final costs for each value of α_c. As expected, for low values of α_c, SA performs rather like greedy search. For high values of α_c (greater than 0.8), the performance of SA deteriorates significantly. The value of α_c for which SA performs well is 0.5.

Tabu Search: A simple TS with the following Tabu definition is tested: any base station location that is selected within the last K iterations is considered Tabu and may not be selected at the current iteration. The value of the tabu tenure K is chosen to be 1. An important parameter that can be varied is the the size of the Candidate List v, which determines how many neighbors are considered at each step in the search. One might expect that there would some improvement as the size of the candidate list increases since one can better sample the neighborhood of the current point in order to determine a good point to move to. Figure 4 shows the minimum, average and maximum final costs obtained by TS for different values of v ranging from 1 to 20. As expected, the algorithm is seen to perform progressively better as v is increased, reaching the best performance for a candidate list size of 10.

Genetic Algorithms: The Genetic Algorithms considered here do not implement crossover and utilize a mutation operator that is equivalent to the neighborhood definition described before. Figure 5 shows the average performance of Genetic Algorithms for population sizes ranging from 10 to 50 with three different selection mechanisms - Rank-based, Propor-

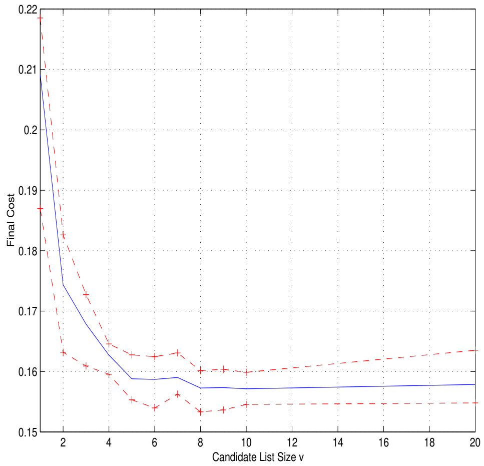

Figure 4: Performance of TS

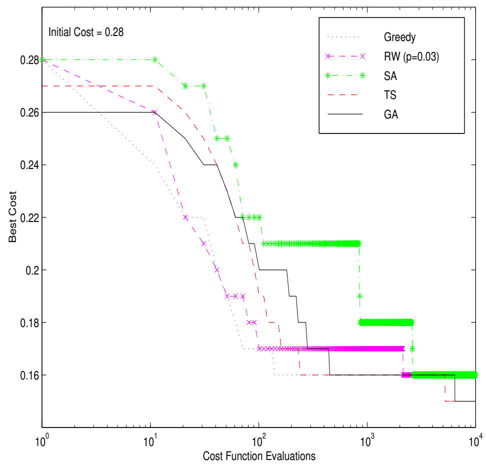

Figure 6: Comparison of Algorithms (1 run)

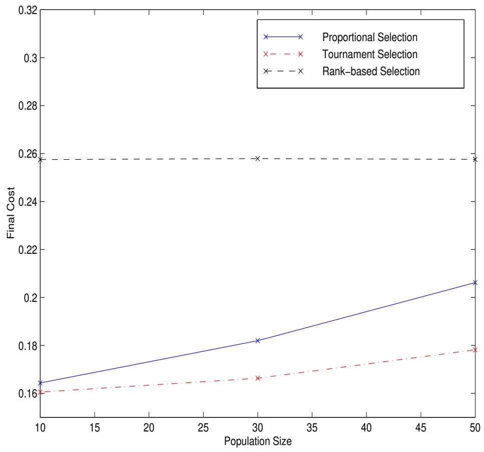

Figure 5: Performance of GA

tional and Tournament Selection. Rank-based selection is seen to yield uniformly poor results irrespective of population size - this is chiefly due to the fact that this selection mechanism ignores the actual values of the cost functions at each generation, focusing on their relative rank instead. Tournament Selection with a low population size of 10 appears to provide the best performance, suggesting that the cost function surface is such that the extra parallelism obtained by using a large-population GA is not particularly beneficial for this problem.

4 Comparison of Algorithms

The performance of Random Walk with $p = 0$ (Greedy Search), Random Walk with $p = 0.03$, Simulated Annealing, Tabu Search, and Genetic Algorithms is compared using the best parameters determined above. To ensure fairness in comparison, each algorithm is run for exactly 10,000 cost functions – the GA with population size of 10 is run for 1000 generations and the TS with a candidate list size of 10 is run for 1000 iterations, while RW and SA are each run for 10000 iterations. Also, as mentioned before, all algorithms utilize a common neighborhood definition.

Figure 6 shows the progress of each algorithm, plotting the best cost seen at each step with respect to the number of cost function evaluations for a single run. Each algorithm was then run 10 times. Figure 7 shows the maximum, average and minimum final cost values obtained by each search technique. The results of the greedy search algorithm that only accepts downhill moves shows the poorest performance with a higher variance than others indicating that it has a tendency to get trapped in local minima. In order of decreasing average final cost, the algorithms are: SA, RW, GA and TS (with the last two providing about the same average final cost). Tabu Search provides final costs with the lowest variance, which indicates that it is able to obtain the best solution in nearly every run. The best solution for the sample base station location problem investigated in this paper was also obtained by Tabu Search and yielded 86.82% coverage with only 10 base stations (with a

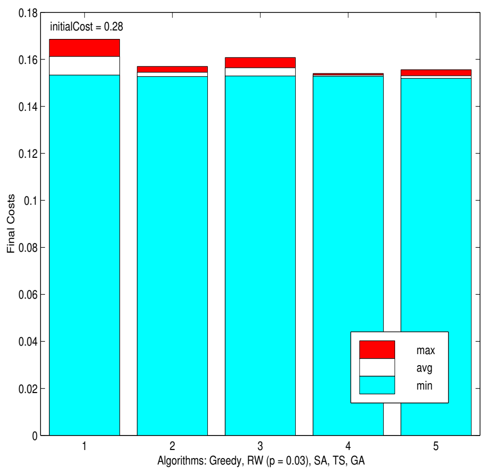

Figure 7: Comparison of Algorithms (10 runs)

cost function evaluation of 0.153). This best solution is shown in figure 8.

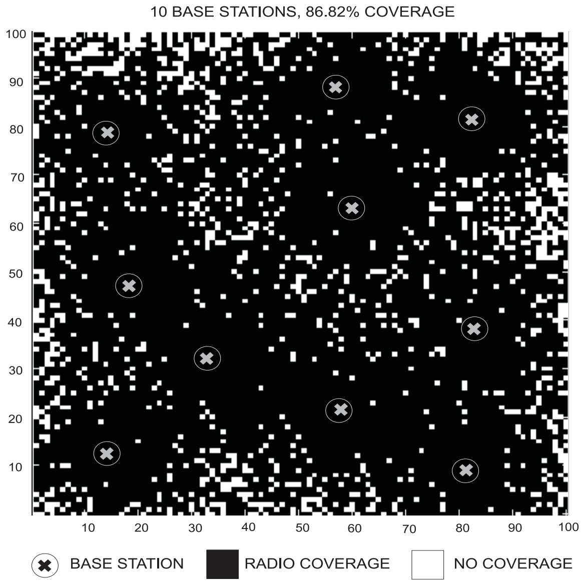

Figure 8: Best Final Solution for Base Station Location Problem

5 Conclusions

This paper has presented an experimental analysis of the performance of four search algorithms for the problem of designing the optimal location of base stations in a mobile communication system - Random Walk, Simulated Annealing, Tabu Search and Genetic Algorithms. We investigated the effect of varying the value of important parameters for each algorithm to determine suitable values. We then compared the performance of these algorithms using a common neighborhood definition, the same number of cost function evaluations, and the same initial conditions. It is found that Genetic Algorithms and Tabu Search both perform well, with Tabu Search providing the most consistent final cost value on multiple runs.

References

[1] P. Calegari et al., "Genetic approach to radio network optimization for mobile systems," *IEEE 47th Vehicular Technology Conference*, vol. 2, pp. 755-9, 1997.

[2] H.R. Anderson, and J.P. McGeehan, "Optimizing Microcell Base Station Locations Using Simulated Annealing Techniques," *IEEE 44th Vehicular Technology Conference*, vol. 2, pp. 858-862, 1994.

[3] E. Aarts and J. K. Lenstra, *Local search in Combinatorial Optimization*, Wiley, 1997.

[4] B. Selman, H.A. Kautz, and B. Cohen, "Noise Strategies for Improving Local Search," *Proceedings of Twelfth National Conference on Artificial Intelligence, AAAI-94*, vol.1, pp. 337-43, July 1994.

[5] B. Krishnamachari and S.B. Wicker, "Global search techniques for problems in mobile communications," *Telecommunications Optimisation: Adaptive and Heuristic Methods*, ed. D. Corne et al., John Wiley and Sons, (expected) 1999.

[6] B. Krishnamachari, *Global Optimization in the Design of Mobile Communication Systems*, Master's Thesis, Electrical Engineering, Cornell University, 1999.