OPEN ACCESS

University of BRISTOL

Molina, A., Athanasiadou, GE., & Nix, AR. (1999). *The automatic location of base-stations for optimised cellular coverage : a new combinatorial approach*. 606 - 610. https://doi.org/10.1109/VETEC.1999.778212

Peer reviewed version

Link to published version (if available):
10.1109/VETEC.1999.778212

Link to publication record in Explore Bristol Research
PDF-document

University of Bristol - Explore Bristol Research
General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available: http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/

THE AUTOMATIC LOCATION OF BASE-STATIONS FOR OPTIMISED CELLULAR COVERAGE: A NEW COMBINATORIAL APPROACH

A. Molina, G.E. Athanasiadou, A.R Nix

Centre for Communication Research, University of Bristol
Merchant Ventures Building, Woodland Road, Bristol BS8 1UB, UK
Fax: +44 (0) 117 954 5206 Tel: +44 (0) 117 954 5123
E-mail: Araceli.Molina@bristol.ac.uk

Abstract – The cost and complexity of a network is closely related to the number of base-stations (BSs) required to achieve the system operator's service objectives. The location of BSs is not an easy task and there are numerous factors that must be taken into account when deciding the optimum position of BSs. This paper discusses the performance of three different algorithms developed to solve the BS location problem: the greedy algorithm (GR), the genetic algorithm (GA) and the combination algorithm for total optimisation (CAT). These three methods are compared and results are given for a typical test scenario.

I. INTRODUCTION

UMTS will provide advanced multimedia services to customers. These new networks will bring additional requirements, such as more advanced services operating in different propagation environments (e.g. microcells, which are already being installed in the second generation networks). New features are expected that will increase the complexity of the resource dimensioning and the expected benefits of these new networks will heavily depend on the capabilities of next generation planning tools.

One of the main cellular network-planning tasks is the optimum location of BSs. The search for an optimisation tool that can solve the BS location problem is not new. In recent years, many authors have investigated the application of different algorithms to solve this problem [1,2,3]. This paper concentrates on the analysis of three algorithms: the greedy algorithm (GR), the genetic algorithm (GA) and the combination algorithm for total optimisation (CAT). The latter is a new algorithm explored for the first time in this paper. To show the efficiency of the CAT algorithm, results are compared with the GR and GA algorithm for a typical deployment scenario. A number of other algorithms were initially considered, such as the Simulated Annealing (SA) [7] and the Simplex algorithm [8], however the number of restrictions necessary to make these methods meet our specifications seriously limited their efficiency and applicability.

II. THE PROBLEM

The aim of the research presented in this paper is to find an algorithm that can automatically identify an optimum solution to the problem of BS deployment. The following section describes in detail the problem considered in this paper.

Our solution is based on the following assumptions. The surface to cover is represented by a set of user supplied points or *control nodes*. The maximum number of allowable control nodes is unbounded. The initial number of possible BS locations is user supplied (i.e. locations where planning permission and/or agreements have been made). The possible position of BSs are pre-set to make the algorithms more efficient. An algorithm that does not have pre-set possible locations could position the solution anywhere in the test area, including non-sensible solutions such as the middle of a lake. This could make the entire solution completely useless even if the rest of the requirements are fully satisfied. Different algorithms will have different restrictions regarding the possible number of BSs that can be used. Note the initial selection of the user supplied sites will determine to a great extent, the final outcome of the radio planning procedure. This arises since no additional locations will be identified during the optimisation process, and user selected positions will not be modified.

A typical problem is shown in figure 1, where a number of control nodes have been distributed over an area. A number of possible BS locations are also displayed in the map. The algorithm must provide coverage to all control nodes using the smallest sub-set of possible BSs. The bounds required for the algorithms are provided by the use of modules that supply the restricting variables. These modules are currently based on simple existing models. The propagation module is based on the Okumura-Hata model [4]. The traffic module is based on the simple assumption of equal demand from every control node.

0-7803-5565-2/99/$10.00 © 1999 IEEE

More complex ray-tracing propagation models and a non-uniform traffic model are planned in the future.

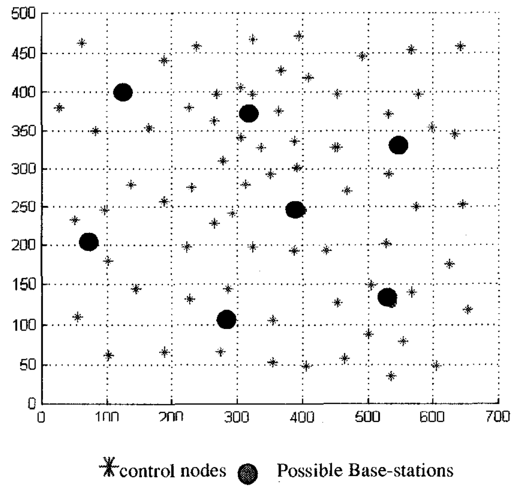

Figure 1.Location of BSs and control nodes

The nature of the optimisation problem is reflected in figure 2. For an area of approximately 500m by 700m, 70 control nodes and 7 possible BS locations have been randomly distributed. The idea is to find the optimum group of BSs that provide coverage to the control nodes dispersed in the area of interest.

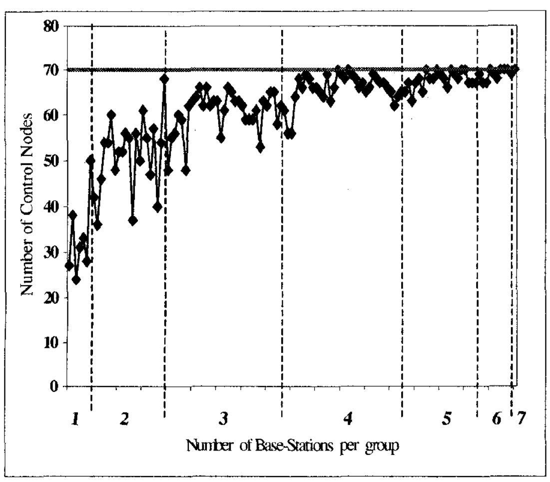

Figure 2. Number of control nodes covered per group of BSs

In figure 2, the horizontal line represents the optimum solution (i.e. all 70 control nodes covered). When the curve reaches this optimum, a solution has been found. The curve represents all the different solutions based on all the different BS groups. In the x-axis "1" represents a single BS, "2" represents combinations containing two BSs and so on, up to 7 where all the BSs are included. Note different groups will contain different numbers of combinations. It can be observed that the probability of finding an optimum solution increases with the number of BSs per group. However it is not guaranteed that a better solution is found if the number of BSs is increased. In fact, in this case some of the solutions reached with a group of five BSs are worse that the best solution for two BSs (based on the number of control nodes covered).

The complexity of the problem is only partially reflected in figure 2. Even so, it is obvious that the BS location problem is not an easy optimisation problem. With this in mind, attention is now given to the development of the algorithms which solve this problem.

IV. ALGORITHMS CONSIDERED

A. The Greedy Algorithm (GR)

The GR is a simple algorithm that can be implemented to solve the BS location problem. The version of the GR implemented in this paper follows the restrictions and assumptions described in section III.

The GR algorithm is based on the following idea. Given a number of BSs and control nodes, the algorithm first selects the BS that covers the most control nodes. The BS and the control nodes are then removed from the area of study and the same operation is repeated until there are no control nodes left to cover [2].

The speed of the GR algorithm is obviously a function of the number of possible BSs, which is set in the planning area by the user. In general, the run-time of the GR algorithm is lower than that required for the rest of the algorithms presented in this paper.

The GR code has been developed to allow the user maximum flexibility. It permits the setting of different parameters, such as the number of control nodes and the number of possible BSs. In future versions, the user will be able to control a wider range of parameters that affect the environment. However, at this stage the inclusion of such parameters would not contribute to the algorithm comparison.

B. The Genetic Algorithm (GA)

The GA is a popular optimisation method that has been studied as a possible algorithm to solve the BS placement problem by several research groups [3] and individuals [5]. The GA optimisation is based on the election of a group of possible solutions or set of individuals that evolve toward an optimum solution, under the selective pressure of the fitness function [5]. The implementation of the GA carried out in this paper, follows the standard definition that can be found in the available literature [3,5].

The major drawback of the GA is its run-time, which becomes unpredictable (very high in the majority of cases) when the size of the population is large [5]. For the placement problem the population is equivalent to the number of possible BSs in the area of study. Some research groups have overcome this problem by modifying the GA. For example, the ACTS STORMS project uses the *island concept* [3]. They claim that this method reduces the run-time of the algorithm. However, it is not clear whether the use of this concept improves the efficiency of the algorithm. In this paper a standard version of the GA has been implemented and compared with results from the GR and CAT algorithms. Methods such as the island concept have been considered as a possible enhancement to improve the algorithm run-time. The objective of this research is to find an algorithm that identifies the best deployment solution within a reasonable run-time.

C. The Combination Algorithm for Total Optimisation (CAT)

The CAT is a new algorithm that follows a combinatorial approach. The initial idea is simple and was partially illustrated at the beginning of this paper. If we can combine all the possible BS locations in the area of study, the optimum combination for this set of BSs can be found. In some cases, more than one combination of suitable BSs can be found. In such a case, more strict bounds can be used to achieve the optimum solution for that group of BSs.

The combination approach fits perfectly with the idea of pre-selected BSs (located by the system operator). Unfortunately, as figure 4 shows, the number of combinations increases dramatically when the number of possible BSs is increased. As a consequence, the computational time tends to infinity [6]. Equation 1 shows the total number of combinations, C_T, that can be obtained when all the possible groups are formed for a given number of elements.

$$C_T = \sum_{G=1}^{G=B} \frac{B!}{G! * (B-G)!} \qquad \text{Eq. 1}$$

Where B is the total number of possible BSs in the area of study and G is the number of elements per group.

Although, the combination idea could provide an optimum solution, the limitations regarding run-time must be seriously taken into account. For clarification, when every possible combination of elements is performed based on equation 1, the method is referred to as OCA (Original Combination Algorithm).

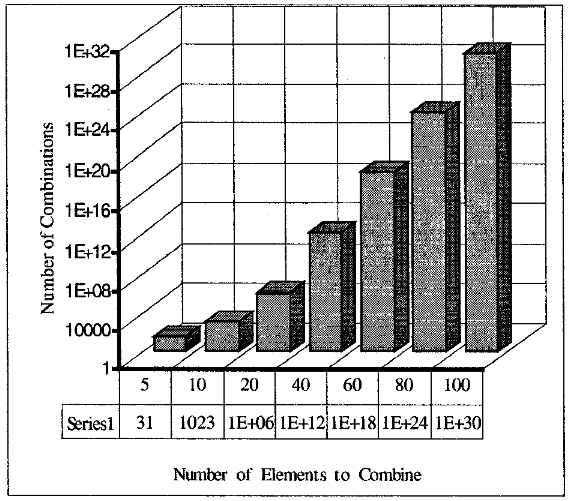

Figure 4. Number of combinations per BSs group .

The CAT algorithm uses the principle explained above combined with the following idea: splitting the possible BS locations into a number of groups and using the GR algorithm as a prediction tool. Note: the use of the GR algorithm is not necessary, although its use as a pre-processor method dramatically reduces computational time.

The CAT algorithm segments the total number of possible BSs into smaller groups, which are randomly selected. The number of elements per group must be small enough to allow the OCA to be performed. As a consequence, a number of solutions are found in every group. The number of solutions is often very high (recall that applying the idea of combinations means the BSs are grouped in every possible way). The CAT algorithm selects the solutions that offer the lowest cost. The best solutions are stored and merged together in a unique group, after that, the process described above is repeated until the number of solutions cannot be further reduced.

The problem frequently found when implementing this approach is that the best solutions, or solutions with the lower cost, are always kept. This implies that in some cases, after a number of iterations the number of possible BSs cannot be reduced any further because all the groups provide the same best solutions. The alternative solution depends on the final number of possible BSs. If the size of the group is within the limits of the OCA, then this can be applied and the problem will be solved. However, if the size of the final group of possible BSs is still too large, then the OCA cannot be applied. In such a case, the MCA-US (Multi-Combinations Algorithm Using Unique Solution) algorithm is introduced.

MCA-US was developed to solve the problem raised when the number of possible BSs cannot be reduced using the method described above. MCA-US uses every combination that is classified as a "best solution" and which are supplied by the combinations performed in every group. The MCA-US algorithm only uses one of the best possible solutions per group, chosen randomly. In this way, the problem of keeping all the best possible solutions is broken, and the number of possible BSs decreases to a final and smaller number. The idea is illustrated in figure 5.

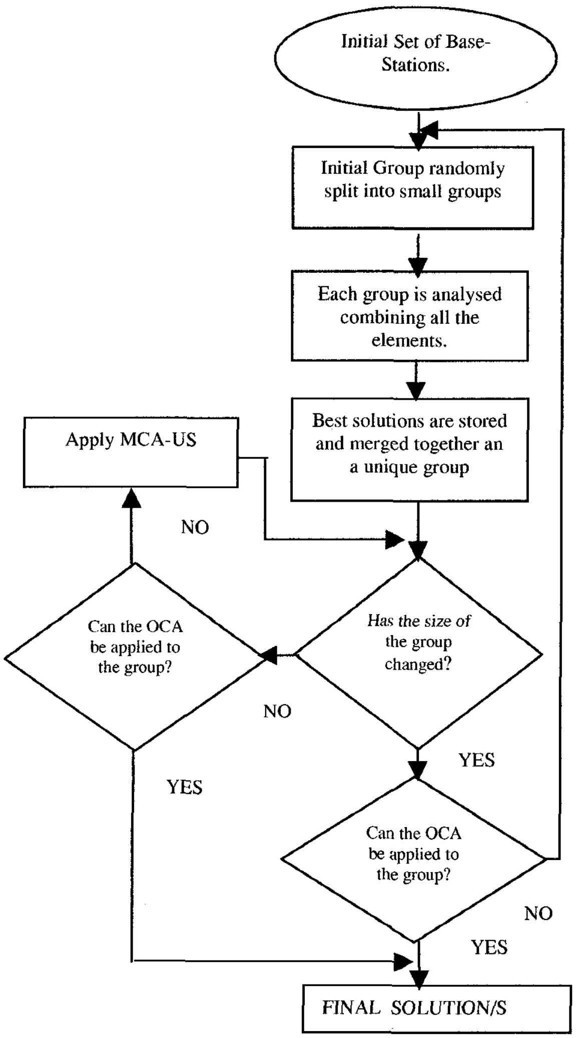

Figure 5. Flow chart for the CAT algorithm.

The CAT algorithm can also be used in conjunction with the GR algorithm, to reduce run-time. The GR algorithm offers an initial solution, which contains the number of possible BSs. This initial number would be introduced as a alternative parameter in the CAT algorithm reducing the run-time required to find a solution or set of solutions.

The number of possible solutions offered by CAT is generally greater than one. This condition allows the user to introduce more tighter bounds or restrictions to find the best possible solution between the groups that contain the same number of elements or BSs. For example it may select the BSs that provide more equal coverage to the control nodes. This means that the CAT algorithm offers the best possible solution according to the user's need, not just the first solution found, which is the case for the other two algorithm described in this paper.

The speed and complexity of the CAT algorithm is a function of the number of possible BSs. In general, the run-time is shorter than that required by our GA but longer than the GR. However, the run-time depends on the technique used, in other words, it depends on the use/non-use of the GR algorithm. The quality of the solutions is unaffected by the use of the GR solutions, but in some cases, the run-time is shorter if the GR algorithm is not used (e.g. for a small number of BSs, <30).

V. CASE STUDY RESULTS

In this section the performance of the three methods are now compared. The most effective way to evaluate the accuracy of each algorithm is to compare its results with measurements taken in a real environment. However, given the complex nature of the environment, this is not possible at this stage of the research. For this reason, a generic problem has been configured, in which all the algorithms are evaluated under the same conditions so as to compare their efficiency and performance.

The results shown here are a representative sample of the work performed in this study. The algorithms were tested under many different conditions and the results tend to follow the same trend in efficiency. The case presented here is shown in figure 6.

The surface covers an area of approximately $1Km^2$. The initial selection of possible locations for the BSs is user supplied, as discussed in section II. The total number of possible BS sites is 60. The control nodes are distributed evenly over the area of study (or in the locations where the operator needs coverage/capacity). The different control node densities can represent capacity requirements in different areas. The total number of control nodes is 140. Table 1 shows the different solutions found by each algorithm. A number is associated with each BS, this number reflects the order in which each BS was supplied by the user.

As can be seen, the solutions supplied by each algorithm are different, both the GR algorithm and the GA provide one unique solution, which is automatically selected. This selection obeys the necessity of finding the minimum cost solution, which in these cases is related to the minimum number of BSs needed to cover the control nodes supplied.

The CAT algorithm behaves in a different way, finding a number of solutions with the same cost, or number of BSs, but goes further by selecting the best possible solution (among the solutions with the minimum number of BSs). This solution is found by applying a tighter bound as described in section IV.C. Figure 6 shows the CAT solution, which is represented by larger and shaded BSs.

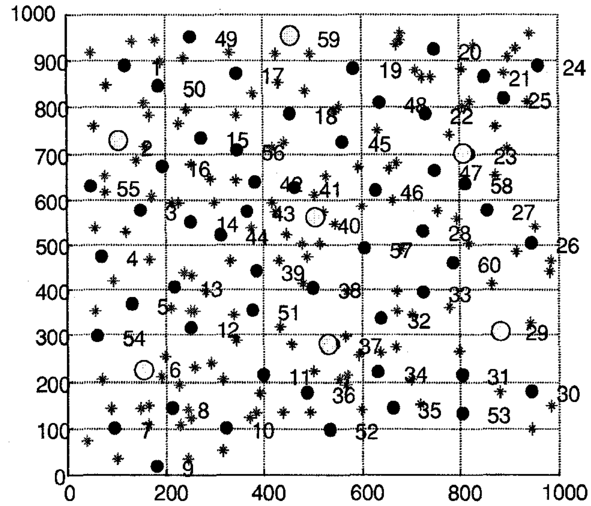

Figure 6. Illustration of test problem.

Algorithm	GR	GA	CAT
Solution/s Found	2,6,15,18, 20,22,29,3 2,51	8,15,22,29, 30,32,38, 44	2,6,23,29, 37,40, 59 2,6,14,22, 29,36,45 ...
Number of solutions	1	1	8
Solution Selected	2,6,15,18, 20,22,29,3 2,51	8,15,22,29, 30,32,38, 44	2,6,23,29, 37,40, 49
Number of BSs needed	9	8	7

Table 1. Solutions offer by the different algorithms.

VI. CONCLUSION

Three different algorithms were presented in this paper. They were tested under the same conditions and shown to perform in different ways with different run-times.

For the previous case study, the solution reached in every case, covered 100% of the control nodes. However every algorithm found different sets of BSs to cover the control nodes effectively. The cost of the solutions depends on the algorithm used. The GR algorithm found the solution with the higher cost (9 BSs) followed by the GA (8 BSs) and the CAT (7 BSs).

The GR algorithm is relatively easy to implement. Although the results were not the best, this is compensated by the fact that the run time required is short. The GA performs well, and offers quality solutions in every case, however its major drawback is the run time. There are numerous studies looking to achieve lower and more predictable run times. The CAT algorithm performed better than the other two algorithms presented here. The structure of the algorithm is flexible due to the way it links to the different modules (coverage and capacity). The performance and flexibility of this new algorithm makes it a very appropriate for solving the BS placement problem.

This paper has shown that automated BS deployment is possible providing user supplied site and control node information is supplied. Of the algorithms considered, the CAT is currently providing the most reliable solutions, and offers the flexibility required for future enhancement and modification.

VII. ACKNOWLEDGEMENTS

The authors would like to thank Nortel Networks, Harlow, UK for their financial support. The authors would also like to thank Dr Dean Kitchener for his useful help and encouragement during the development of this research.

VIII. REFERENCES

[1] S.F. Fortune, D.M. Gay, B.W. Kernighan, O. Ladron, R. A. Valenzuela, M. H. Wright. AT&T Bell Laboratories, "WISE Design of Indoors Wireless Systems: Practical Computation and Optimization". IEEE Computational Science & Engineering. Spring 95, pp 58-68

[2] W.J.Cook, W.H Cunningham, W.R.Pulleyblank,A.Schrijver "Combinatorial Optimizatio". John Wiley & Sons,inc

[3] ACO16 STORMS CD, April 1998.

[4] J.D. Parsons, " The Mobile Radio Propagation Channel", Pentech Press 1992.

[5] J. Michael & Y. Rahmat-Samui. "Genetic algorithms in engineering electromagnetic". IEEE Antennas and Propagation Magazine. Vol. 39-4, August 1997, pp 7-25.

[6] E. Kryszig. "Advance Engineering Mathematics". 7th Edition. John Wiley & Sons.

[7] E. Aarts and J. Korst. "Simulated Annealing and Bolzmann Machines, A Stochastic Approach to Combinatorial Optimisation and Neural Computing". John Wiley & Sons 1989.

[8] J.A. Nelder and R. Mead. " A Simplex Method for Function Minimisation". The Computer Journal, Vol. 7, April 1964 to January 1965. Pp 308