Cross-Layer Clustering Optimization in Mobile Networks Using Evolutionary Algorithms

L. Carro-Calvo, S. Maldonado-Bascon, A. Portilla-Figueras,
S. Lafuente-Arroyo, and S. Salcedo-Sanz

Universidad de Alcalá, Spain
sancho.salcedo@uah.es

Abstract. In this paper we present an evolutionary algorithm to tackle the aggregation network design in a mobile communication system. The design and optimization of this part of the network involves four different tasks: the determination of the number and location of the Base Station Controller (BSC) or Radio Network Controllers (RNC), the assignment of Base Stations (BTS) or B-Nodes to the controllers, the definition of the tree structure that links all the nodes with the controllers and, finally, the system assignment in the links between the different hops of the tree. The novel evolutionary heuristic proposed deals with all these sub-problem together and it is able to obtain good solutions, as will be shown in several real scenarios.

Keywords: Aggregation network, multi-layer optimization, evolutionary algorithms.

1 Introduction

Traditionally, the main problem in the design of mobile telecommunication networks is related to the cell deployment [2], i.e., to the definition of the type, number and location of BTSs in 2G systems, or B-Nodes in 3G ones, that are required to provide coverage to a specific area.

The aggregation network (transport network between the terminal nodes, the BTS and the BSC) design has been considered sometimes a low relevancy task, implemented by leased lines of a fixed operator. However as the data services have become popular, and the required bandwidth upstream the BSC has increased, the design of this part of the network has gained higher relevance again, becoming one of the main problems in the access network planning.

The design of the aggregation network involves three different tasks:

1. Logical layer design: It consists on the cluster definition, i.e., the determination of the number and location of the Base Station Controller and the definition of which BTS are assigned to each BSC.
2. Physical layer design: For each of the clusters defined in the logical layer, the physical topology that links all BTS with the BSC is determined, usually forming a tree structure. Note that some new network elements, as Hubs to

P. Chatzimisios et al. (Eds.): MobiLight 2010, LNICST 45, pp. 638–649, 2010.

concentrate links up-streams or link repeaters (with no logical function) has to be considered in this task.

3. System Assignment: Once the previous tasks have been carried out, this task calculates the type and number of network equipments (ports, link systems, etc.) to provide enough capacity to the network previously designed.

Traditionally aggregation network optimization has been tackled trying to optimize each task separately. The Logical layer design could be considered as a terminal assignment problem with capacity restrictions, [3]. Intense research has been carried out in field, using different approaches, from traditional ones, see [4], to evolutionary or hybrid meta-heuristics, [6], [5]. The same situation could be observed in the physical layer design. There are quite broad literature about tree topology design with capacity, distance and reliability restrictions [7], [8]. Finally, when the network is completely designed, the system assignment is just a deterministic procedure using the most suitable equipment for each network element [9].

Although these studies offer interesting results in terms of network investment cost, the problem of network optimization has to be considered under a global perspective. Note that small changes in the clustering procedure obtained in the aggregation network design may produce dramatic changes in the network topology, which may result in important cost reductions. This means that the optimization of the whole aggregation network has to use global procedures under a multi-layer approach. As far as we know, there are not specific works over this point in the literature.

This paper proposes a novel heuristic to tackle the multi-layer optimization problem in the aggregation part of a mobile network. The proposed heuristic is based on an evolutionary algorithm that optimizes the investment cost of the calculated network. This is done by considering traffic and physical ports capacity constraints in the links and nodes, and also distance restrictions in the links.

The rest of the paper is organized as follows. Next section provides a mathematical description of the problem. Section 3 presents in detail the proposed evolutionary algorithm. Section 4 provides a description of the experiments carried out to test the performance of the proposed algorithm. Finally the conclusions section discusses the applicability of the proposed heuristic and some future work lines.

2 Problem Definition

Let us consider a set of N BTSs from $n = 1, \ldots, N$ and a set of M BSCs from $m = 1, \ldots, M$. Each BTS n handles a traffic of A_n Erlangs, which has to be routed from the BTS up to the BSC where it is assigned. Each BSC m has a maximum capacity in terms of traffic Ca, and also in the number of radio links that could be physically connected to it, Crl.

We consider that, as usually, the logical connections between the BTSs and the BSCs are done considering a star structure, where all the BTSs are directly linked to the BSC. On the other hand, the physical communication between the

BTSs and the BSC is done using a tree-structured network. Each branch of the tree is connected to the main trunk by a BTS-hub which concentrates the traffic of the BTSs that are under it in the branch. To save costs, the BTS-hub is placed in the same location as an existing BTS. The BTS-hub has a limited capacity, defined by Crl_{hub}, involving the number of radio links of the BTSs that could be connected to it.

The connections between network elements, BTS-hubs, BTS-BSC or hub-BSC, are implemented by radio links RL. The maximum distance each radio link can reach is defined by d_{RL} and the traffic flow as A_{RL}. Each radio link implements a specific transmission system TS_i from a set of systems $i = 1, \ldots, I$ which has a maximum capacity in terms of the traffic it can handle, C_i and an associated cost Q_i. Furthermore, the reliability of each radio link decreases as a function of the radio link length RL_d, let us name it $f(RL_d)$.

Each network element has a corresponding associated cost. The cost of the BSC and the hub are Q^{BSC} and Q^{Hub} respectively. For each radio link, there are three different costs. The first one is associated with the required physical port to implement in the hub or in the BSC, Q^{port}. The second cost depends on the type of system that has to be implemented to carry the traffic of the BTSs in the branch of the tree. Systems which support higher capacity will have higher costs Q^S. Finally if we need to link two network elements separated more than d_{RL} an additional repeater equipment will be required, and hence we will have to add the corresponding Q^R.

To find the complete network configuration we have identified the following two subproblems:

2.1 First Sub-problem: Capacitated Clustering Problem

The first subproblem consists of the logical assignment of BTSs to BSCs, to define the M clusters of the network, see Figure 1 (a). Each BTS is assigned to its nearest BSC with enough free capacity. This is a classical terminal assignment problem with capacity limits in the concentrator node. The fitness function tries to minimize the total aggregated distances. Let us consider that K_m, $m = 1 \cdots M$, is the set of BTSs assigned to the BSC m. Let us also consider that d_n^{BTS} is the distance between the BTS n and its corresponding BSC. Therefore the objective function may be defined as follows.

$$\min \left(\sum_{i=1}^{N} d_i^{BTS} \right) \tag{1}$$

Subject to:

$$\sum_{\forall \; BTS \; \in \; K_m} A_n \leq Ca, \forall K_m, \, m = \{1, \ldots, M\}. \tag{2}$$

2.2 Second Sub-problem: Tree Structured Physical Topology

The second subproblem consists in the definition of the tree structure between the BSC and the BTSs associated to it, see Figure 1 (b). To do this, we need to determine the location of each hub h, $h = 1 \cdots H$ in the cluster K_m and the physical links between the BTSs and hubs or BSCs. Note that the value of H changes for each cluster. Furthermore, we need to calculate the aggregated traffic flow from the BTSs upstream to the BSC in order to determine the type of transmission system that has to be implemented. There are also some constrains related to the capacity of the BSC, Crl, the maximum capacity of the hub, Crl_{hub}, the maximum length and reliability of the radio links, d_{RL} and $f(RL_d)$ and the maximum capacity of the transmission system C_i. The objective is to minimize the total cost of the tree topology of each cluster K_m, Q_{K_m} following the equation:

$$\min\left(Q_{K_m}\right) = Q^{BSC} + Q^{Links} + NHubs_{K_m} \cdot Q^{Hub}, \tag{3}$$

where $NHubs_{K_m}$ is the number of hubs required in the BSC cluster K_m and Q_{Links} is the total cost of all radio links RL in the BSC cluster given by the following:

$$Q_{Links} = \sum_{\forall RL \in K_m} Q^{Port} + Q_i + N_{Repeaters}\left(Q^R + Q_i\right) \tag{4}$$

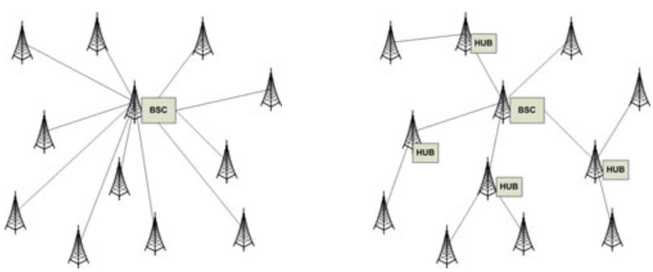
The constraints for this tree structure are:

$$|RL_{BSC} \in K_m| \leq Crl \tag{5}$$

$$|RL_{Hub}| \leq Crl_{hub} \, \forall Hub \in K_m \tag{6}$$

$$A_{RL} \in K_m \leq Ca_i \, \forall RL \in K_m \tag{7}$$

$$\min_{\forall \text{Branchs} \in K_m} \left(\prod_{\forall \text{links in Branch}} f(RL_d) \right) \geq f_{min} \tag{8}$$



(a) (b)

Fig. 1. Problem definition; (a) Logical layer (star structure topology); (b) Physical layer (tree structure topology)

Equation (5) specifies that the number of radio-links connected to the BSC do not exceed the capacity of the BSC. Equation (6) applies the same criteria for the hubs in the cluster. Equation 7 assures that the capacity of the transmission system implemented in each radio-link RL is enough to carry the traffic flow of the link. Finally, Equation (8) defines the requirements in terms of reliability. It analyzes the less reliable branch in the three and compares it with a minimum threshold defined by f_{min}.

3 Proposed Evolutionary Algorithm

This section presents the proposed meta-heuristic algorithm to solve this problem. Specifically, it is an evolutionary algorithm adapted to tackle with the different sub-problems of the aggregation network optimization problem tackled. This section is divided into three subsections: the first one deals with the solutions encoding into the algorithm, the second one explains the initialization of the algorithm and the fitness function used, and finally the third one shows the evolutionary operators used.

3.1 Solution Encoding

The complete network is composed of M clusters corresponding to the M BSCs of the problems. Therefore the physical structure will be composed of M trees. Let L be the maximum possible number of BTSs that can be allocated in a single cluster. Note that L will be chosen as a value that ensures a feasible encoding of our algorithm. Then, each tree is codified in the evolutionary algorithm as an integer $2 \times L$ matrix S_m (note that an individual in the evolutionary algorithm is formed by M matrices S_m). The coding of the tree is done from the leafs to the trunk, what means that the lower level BTSs are on the leftmost positions of the matrix. Each position of the first row of the matrix stores either a BTS identifier, $i = 1 \ldots K_m$, or the value -1. The second row stores the jump to the position in the matrix where the preceding BTS in the tree is located. Let us illustrate this encoding method with a graphical example. Consider a single cluster with nodes $n = \{1, 5, 2, 11, 6\}$ in it, in hierarchical order from leafs to trunk. The maximum number of nodes per cluster is $L = 9$. Figure 2 shows the corresponding matrix encoding and the resulting decoded tree. Note that node 1 of the cluster m in position $S_m(1, 1)$ of the matrix is linked to node 5 in the tree. Then, the value in position $S_m(2, 1)$ of the matrix is 1 which is the jump to the position of the node 5 in the matrix. The same happens with node 5 and node 11. The value in the position $S_m(2, 2)$ is 3, the jump to the position of node 11 in the matrix.

As we are using an evolutionary algorithm, to encourage the diversity of the search space we also allow jumps to non-feasible positions in the matrix. In this case, the tree decoding will use the next feasible position where a BTS is stored. Thus, Figure 3 shows a different encoding for the previous example which is also valid. In this case, the jump in position $S_m(2, 2)$ indicates a position in the

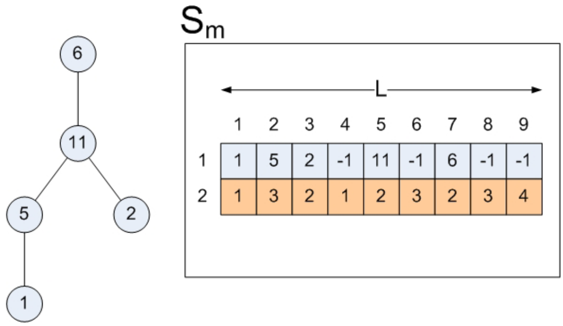


Fig. 2. Example of tree structure encoding

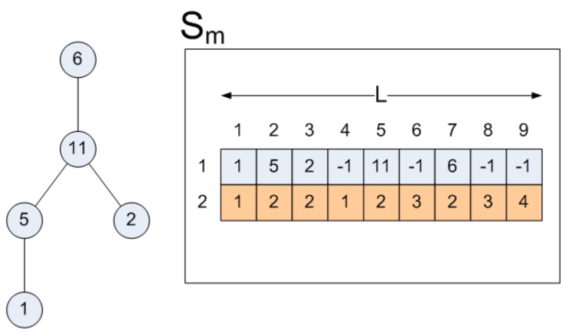


Fig. 3. Example of tree structure encoding with jumps to non-valid positions

matrix with no BTS, denoted by a -1. Therefore, the decoding of the tree will look for the next feasible position, which is again the position corresponding to node 11. In both cases, when the decoding algorithm reaches to the root node (BSC of the cluster, in the example node number 6), no further jumps are allowed in the matrix.

3.2 Initialization

The proposed heuristic will determine the optimum clustering distribution by means of crossover and mutation procedures that will be explained in subsection 3.7. However for the algorithm initialization we will use a $p-median$ algorithm which is described in [5].

The $p-median$ algorithm provides from the complete set of nodes of the experiment the division in the M clusters with the corresponding M BSC or root nodes. This algorithm minimizes the total distance between the nodes in the cluster under the capacity constraint of the BSC. The tree structure of each cluster defined by the $p-median$ algorithm is randomly generated, $S_m(1,j)$ for $i = 1 \ldots L$. Note that although the $p-median$ algorithm defines a root node (the BSC) for each cluster, our algorithm does not consider them to proceed to a further optimization of the total network structure.

The second row of the matrix S_m, which represents the jumps in the tree structures, is initialized as follows: each position $S_m(2, j)$, $j = 1, \ldots, L$ takes a value that is randomly generated following a uniform distribution in the interval $[1, p]$. Parameter p models the initial topology of the cluster. If $p \approx 1$ the cluster structure will tend to form a chain. Opposite if $p \approx L$ the structure of the cluster will tend to be a star. Therefore we have fixed the value of p to $L/2$.

3.3 Reliability of the Branches

An important point in this implementation is the reliability of the links. We consider that each link has a reliability of $f_{RL} = 0.994$ that could be considered constant in the distance range $[0, d_{RL}]$. The reliability is calculated from the leaves to the root of the tree. Each node in the branch has a reliability value, f_{node} that is the minimum over of the reliability values of all branches under it. The value of the reliability of each branch is calculated multiplying the current reliability value, by the value of the next link upstream in the tree. When the current value of the reliability in a node is lower than a minimum threshold f_{min} we consider a penalty value P_f that is linearly incremented depending on the number of nodes upstream the current one. Of course this solution will not be feasible and hence will be surely discarded by the evolutionary algorithm. This procedure is graphically explained in Figure 4.

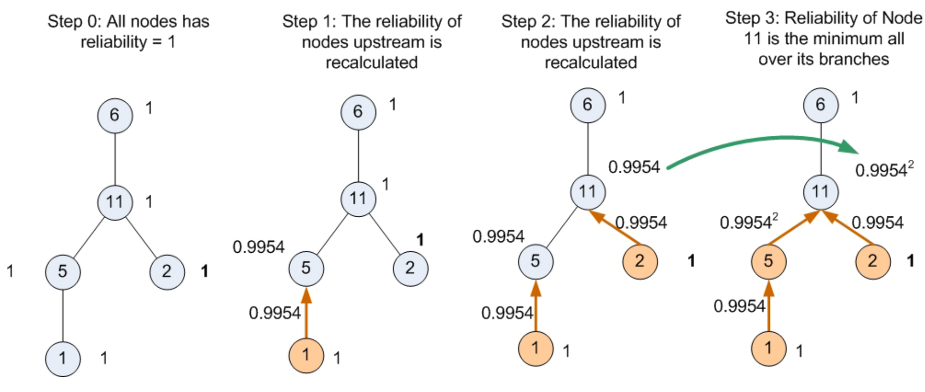


Fig. 4. Example reliability calculation procedure

3.4 Capacity of the Hubs and BSC

In Section 2.2 we have considered the capacity constraints of the hubs and BSC related to the number of physical interfaces. The solution obtained by the evolutionary algorithm will not be feasible if these capacities are exceeded. Therefore, we introduce a penalty factor P_p in the fitness function that linearly depends on the exceeding ports in the hubs and BSC of the current solution. Regarding the BSC, there is an additional constraint related with the traffic capacity. This constraint is related to the logical capacity of the BSC and it could not be applied to the hubs. The maximum traffic handled by the BSC has to be lower than a maximum value Ca. If this value is overcome, the solution is again unfeasible, and the corresponding penalty, P_c has to be applied in the fitness function.

3.5 Large Distance Penalties

Early experiments performed with the algorithm showed that in some cases, there were some crossed links in the clusters. To solve this point and to reduce the length of the cluster links (hence to minimize the cost function), we have introduced a penalty for large distance links, P_d. The value of this penalty is obtained from a stepwise function, where each step has a different slope as it is shown in Figure 5.

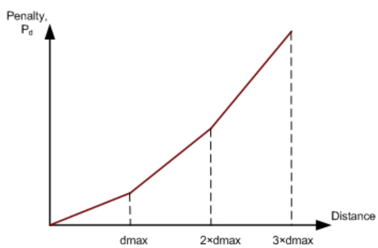


Fig. 5. Example of stepwise distance penalty function with different slopes

3.6 Fitness Function

The proposed evolutionary algorithm uses a fitness function closely related to the one shown in Equation (3). Note that the proposed algorithm performs a multi-layer optimization. This means that we try to simultaneously optimize, the cluster organization in the logical layer (see subsection 3.7), the cluster topology in the physical layer and finally the system assignment. A very difficult problem when tackling this multi-layer problem is to calculate the traffic flow upstream in the tree, because it depends on the physical topology. To overcome this difficulty, our fitness function is calculated at the same time that the trees are being decoded. To do this, we define a new vector O_m, with length L, that stores, for each node in the current tree, the number of radio links downstream to the next level in the tree. We also define a cost $Qagg_m$ which represents the aggregated cost of the network elements and systems that is calculated as we climb the tree towards the root node.

Summarizing, the final cost of the cluster Q_{K_m} is:

$$Q_{K_m} = Qagg_m + \alpha \cdot P_f + \beta \cdot P_c + \delta \cdot P_p + \eta \cdot P_d \quad (9)$$

And the value of the fitness function of each individual

$$\sum_{\forall K_m} Q_{K_m} \quad (10)$$

3.7 Evolutionary Operators

Selection operator. Once we have calculated the fitness value for each individual, the algorithm calculates the average value of all individuals. All those individuals with a fitness value above the average are discarded. Note that the algorithm tries to optimize the total cost of the network so lower values of the fitness function, that is lower cost, are better than higher values. Each discarded individual is replaced by a new individual resulting from the crossover and mutation operators, applied over the reduced population, that are explained below.

Crossover Operator. The crossover operator implemented in our evolutionary algorithm considers only intra-cluster changes. From the complete set of individuals, we randomly select two of them to be the parents of one individual in the next generation. Remember that each individual is a set of M trees, where each cluster is codified using the matrix S_m. Each position in the matrix of the new individual, for example $S_m(1,5)$ and $S_m(2,5)$ is randomly selected from the equivalent positions $S_m(1,5)$ and $S_m(2,5)$ in the parent matrix with probability 0.5. The crossover procedure has a list of all nodes that have been added to the new individual, so it guarantees that nodes previously added to the new individual are not available for following assignment.

At the end of this procedure, there could be some nodes that are not assigned to any cluster in the new individual. To repair this solution, these nodes are included in the first free position of the same cluster they were in the first parent. To guarantee the coherency, in the second line of the cluster matrix, S_m, the value of the jump is modified to point to the same position as in the first parent. If the node is after that position, the value of the jump is fixed to 1.

Mutation operator. The proposed heuristic uses three type of mutations, two of them act inside the same cluster and the third one induces changes between two different clusters. The first intra-cluster mutation swaps two randomly selected nodes in the structure of the tree. This mutation is applied to 1/8 of the total number of nodes with probability 0.1.

The second intra-cluster mutation modifies the value of the jump in the second line of the matrix S_m of a randomly selected node of $M/10$ randomly selected clusters. The value of the jump of this randomly selected node is fixed to 1 with probability 1/5. This means an enlargement of the tree because this node will be now connected to a node closer to the leafs. With probability 4/5 the jump will take the value required to be linked to the node above its current parent node. This means a shortening of the tree.

The last mutation operator performs a swap between nodes from different clusters, randomly selected. Note that only the node identifier j in the first row of matrix $S_m(1, j)$ is swapped in order to keep the mathematical structure of the tree. To avoid dramatic changes in the geographical structure of the tree, the swapping probability is higher for the nodes of the surrounding clusters.

4 Experiments and Results

To test the performance of the algorithm we have developed a set of three experimental examples. Each one is composed of a subset of the most important districts in Spain. The complete scenario consists of 881 districts, with a population greater of equal to 1000 inhabitants. This information has been obtained from the Spanish National Statistic Institute. For the experiments we consider the case of an mobile operator with 25% of market share. The Spanish penetration of the mobile service is 110% on average. We consider that a single BTS provides service to 5200 inhabitants following the results in [1]. The individual traffic per user is 12.5 mErlangs as it results from [10]. Using the Erlang B formulation, and considering 16Kbps circuits in the Abis interface [11] between the BTS and the BSC, the total throughput of the district is calculated. We consider that all BTS's links in the district are multiplexed into a single one that is offered to the network structure.

Table 4 shows the main parameters of the experiments carried out. Note that the ratio between the number of BTSs and the number of predefined clusters is 128, that is a typical capacity of a BSC. The cost and capacities of each network equipment is shown in Table 4.

Table 1. Experiments definition

Experiment	Number of Nodes	Total Number of BTS	Number of Clusters
Exp1	300	14190	111
Exp2	600	17134	134
Exp3	881	18131	142

Note that as we have not found any similar work in the literature for comparing with, so a lower bound obtained by considering no capacity constraints neither in the BSC nor in the Hubs is used to validate our approach. We have run each experiment 10 times and we offer the values of the best and average solution and the standard deviation. These results are shown in Table 4. Note that even for the largest experiment, Exp3, the final cost of the best solution obtained is quite close to the lower bound, only 3% higher.

The purpose of this algorithm is to be used as a planning tool in real applications on mobile communications regulatory processes. Therefore an important

Table 2. Cost of the network elements in k€

Element	Cost
Increment BTS/Hub (Q^{Hub})	42.0
HUB card (Q^{port})	2.5
Repeater (Q^{R})	21.6
2Mb system (Q^{i})	15.0
8Mb system (Q^{i})	25.0
34Mb system (Q^{i})	38.0
140Mb system (Q^{i})	47.0

Table 3. Experimental results

Experiment	Min Cost	Avg Cost	Std. Desv	Lower Bound
Exp1	5177.1	5261.3	17.0 (0.32%)	5081.93
Exp2	11872.0	12105.0	87.0 (0.71%)	11696.12
Exp3	45991.0	47132.0	381.0 (0.80%)	44453.80

feature is the computation time. A large processing time means that the repetition of each experiment becomes a hard task, so it is important to observe the evolution of the algorithm versus time (wall clock time is used in this case). In the experiments carried out, the stopping condition of the evolutionary algorithm was fixed using computation time, with a maximum of 5 minutes of computation (wall clock time). Table 4 shows this evolution. Note that the value of the investment cost obtained at time=1 min is less than 3% worse than the solution obtained at time=4 min even in the Exp3. This error percentage falls inside the tolerance margin of any possible real implementation, so the solution at time=3 min could be a good compromise. The final best solution is obtained at time=5 min.

Table 4. Computational time of the proposed algorithm

Experiment	1 min	2 min	3 min	4 min	5 min (max time)
Exp1	5202.37	5188.72 (-0.26%)	5188.56 (-0.26%)	5095.06 (-2.06%)	5081.93
Exp2	12034.5	11909.3 (-1.04%)	11837.3 (-1.63%)	11759.8 (-2.28%)	11696.12
Exp3	47094.7	46705.8 (-0.82%)	46524.7 (-1.21%)	45497.1 (-1.26%)	44453.80

5 Conclusions

In this paper we have proposed a novel evolutionary heuristic for the multi-layer optimization problem in tree topology access networks. The proposed approach has some innovative characteristics such as the possibility of moving nodes between different clusters, changing the location of the BSC inside each cluster and the form of encoding the tree topology of the clusters. We have run several experiments with increasing number of nodes, from 300 to 881 and the results are very close to a lower bound introduced for the problem. Another key point is the processing time, obtaining good results within short times, about 180 seconds. This makes this algorithm an interesting tool to be used in network planning tasks. In fact the final objective of this work is to substitute the old access network planning algorithms used in previous regulation projects as [1] by the one proposed in order to obtain much more optimized structures.

References

1. Brinkmann, M., Hackbarth, K., Ilic, D., Neu, W., Neumman, K.H., Portilla-Figueras, A.: Mobile Termination Cost Model for Australia. Final Report, `http://www.accc.gov.au`
2. Krishnamachari, B., Wicker, S.: Base station location optimization in cellular wireless networks using heuristic search algorithms. Soft Computing in Communications, 201–219 (2003)
3. Quintero, A., Pierre, S.: Assigning cells to switches in cellular mobile networks: a comparative study. Computer Communications 26, 950–960 (2003)
4. Riis, M.: Deployment of mobile switching centers in a telecommunications network: a stochastic approach. Telecommunication Systems 26(1), 93–109 (2004)
5. Salcedo-Sanz, S., Portilla-Figueras, J.A., Ortiz-Garcia, E.G., Pérez-Bellido, A.M., Thraves, C., Fernández-Anta, A., Yao, X.: Optimal switch location in mobile communication networks using hybrid genetic algorithms. Applied Soft Computing 8, 1486–1497 (2008)
6. Pierre, S., Houeto, F.: Assigning cells to switches in cellular mobile networks using taboo search. IEEE Trans. Syst. Man. Cyber. 32, 351–357 (2002)
7. Lee, Y.J., Atiquzzaman, M.: Exact algorithm for delay-constrained capacitated minimum spanning tree network. IET Communications 1(6), 1238–1247
8. Charnsripinyo, C., Wattanapongsakorn, N.: A Model for Reliable Wireless Access Network Topology Design. Proceedings of the IEEE TENCON 2, 561–564 (2004)
9. Cox, L.A., Sanchez, J.: Designing Least-Cost Survibale Wireless Backhaul Networks. Journal of Heuristics 6(4), 525–540 (2000)
10. Comisión del Mercado de las Telecomunicaciones (Spanish Telecommunication National Regulator), Informe Anual (2008), `http://www.cmt.es`
11. Boucher, N.J.: The cellular radio handbook. J. Willey & Sons, England (2001)