WiMAX Network Planning Using Adaptive-Population-Size Genetic Algorithm

Ting Hu, Yuanzhu Peter Chen, and Wolfgang Banzhaf

Department of Computer Science, Memorial University, St. John's, Canada
{tingh,yzchen,banzhaf}@mun.ca

Abstract. IEEE 802.16, also known as WiMAX, is a new wireless access technology for currently increasing demand of wireless high-speed broadband service. Efficient and effective deployment of such a network to service an area of users with certain traffic demands is an important network planning problem. In this article, we resort to a Genetic Algorithm in order to yield good approximation solutions. In our method, individual representation and genetic variation operations are specifically designed to incorporate the feature of this application problem. Moreover, an adaptive population size approach inspired by neutral theory in molecular biology is applied in our algorithm to enhance its search ability. Simulation results show that our algorithm is fairly effective and robust to different scenarios of the network planning problem. By comparing to a conventional fixed population size scheme, our method is further verified to be able to accelerate the search process.

1 Introduction

WiMAX (Worldwide Inter-operability for Microwave Access) is a telecommunication technology based on the IEEE 802.16 standard in order to provide broadband wireless networks at the metropolitan scale. It intends to replace the more expensive wireline-based access technologies such as TV cable and ADSL [7]. As the standard evolves, WiMAX supports a variety of data transmission methods. It originated from the first 802.16 standard in 2002, also called WirelessMAN, where a cellular-like *point-to-multipoint* (PMP) operation is adopted. In the PMP mode, all communications are limited to be between a base station (BS) and a subscriber station (SS). With a serial of amendments later on, currently, a new working group, 802.16j, is focusing on multi-hop extensions so that the network can operate in a *mobile multi-hop relay* (MMR) mode. With the relay stations (RSs) to help, the coverage of the BSs can be increased significantly, which alleviates the line-of-sight (LOS) problem further [3].

In this article, we focus on the PMP mode of WiMAX and leave the more general model which incorporates the MMR mode as our next step work. In the PMP mode of WiMAX, there can be two types of entities to form the wireless component of the network, the BSs and SSs. The BSs form the infrastructure for the SSs. An SS is allowed to communicate to a BS directly if the channel quality is sufficient for the given data rate. A network planning problem in this case is

an optimization problem to cover the SSs in a geographical area using a small number of BSs. The BSs can only be placed on a subset of pre-selected candidate sites. Typically, the locations of the SSs and their bandwidth requirements are given. In addition, the channel gains between the locations of the SSs and all BS candidate sites can also be obtained. Thus, for a given candidate site, the set of SSs that can be serviced by this site is known as well. Note that, in practice, since every BS has a capacity upper limit, it may not necessarily service all these SSs within range. We assume that there is no power control mechanism at either end of the channel.

The network planning problem considered in our work can be formulated as an *unsplittable capacitated facility location* problem. It is the most difficult flavor among the various facility location problem variants (see Section 2.2). In a broader context, the constraints enforced by this problem render the techniques used to solve the bin-packing and p-median problems not suitable here.

Given these inherent challenges of the network planning problem, we resort to an Genetic Algorithm (GA) to utilize the renowned advantages of evolutionary computing. Compared to exact algorithms, evolutionary algorithms are fairly robust to varying problem instances, and can provide a set of near-optimal solutions with similar or identical utility for multiple options. A specifically customized GA with variable population size is proposed here to embrace the properties of the WiMAX network planning problem. The effectiveness of this algorithm is further verified by simulation.

2   Background

2.1   Problem Formulation

The *network planning problem* can be modeled as a minimization problem on a weighted graph $G = (V, E)$. There are two types of vertices in the graph, i.e., $V = B \cup S$, where B represents the candidate BS sites and S represents the SSs. For each $s \in S$ and $b \in B$, there is an edge between them if the channel gain $g(s, b)$ between s and b is greater than or equal to a given threshold δ for data reception. Therefore, graph G in this case is a bi-partite graph, where there is no edge within B or S themselves. Every $s \in S$ is associated with a capacity requirement of bandwidth c_s. The candidate BS sites each have a capacity limit of C, which caps the total amount of bandwidth of its connected SSs.

A *feasible plan* is a mapping $M : S \mapsto B$ that satisfies the following constraints.

1. For each $s \in S$,
$$g(s, M(s)) \geq \delta. \tag{1}$$

2. We define the *load* of a BS $b \in B$ with an enforced capacity limit as
$$l(b) = \sum_{M(s)=b, s \in S} c_s, \, l(b) \leq C. \tag{2}$$

The total infrastructure cost of the network lies in the number of BSs in service. Therefore, our goal is to minimize $|M(S)|$ over all feasible plans.

2.2   Related Works

WiMAX network planning as an optimization problem in different flavors, has attracted research interests recently. When a BS has a capacity limit, the problem is called *capacitated*; otherwise, it is *uncapacitated*. Amaldi et al. [2] study the problem in uncapacitated UMTS cellular networks by formulating the problem as an Integer Program (IP), and resort to randomized greedy search and tabu search. In Yu et al. [11], a two-tier assignment variant is considered to model 802.16j MMR with the assumptions that the RSs and BSs are uncapacitated. In their solution, a fixed number of BSs is considered so that the top-level assignment can be treated by a p-median clustering. Generally, when an SS is allowed to be serviced only by one BS (or RS), we say the problem is *unsplittable*, as in the work discussed above. Alternatively, with more sophisticated scheduling and channel assignment, an SS may be serviced by multiple BSs (or RSs) equivalently. This is called *splittable*. In Lin et al. [8], a flow-based heuristic is devised to relax the capacitated IP formulation essentially to a splittable variant. This is a generalization of the problem of *capacitated facility location* [6], where an SS can be potentially serviced by all BSs with different transportation costs.

Evolutionary algorithms have also been applied to variants of the network planning problem. For instance, in the context of 802.11 access point (AP) deployment where a wired line is needed to connect every AP, the problem appears to be a multi-objective optimization to minimize the costs of both placing APs and laying down cables to join them. Moreover, the locations of APs are not constrained as in our problem formulation. Toward this AP deployment problem, both GA [10] and GP [4] have been applied and shown to achieve successful solutions. Although on different problem formulations, the literature suggests that evolutionary algorithms can be effective and robust applied to such problems.

For the more general models of p-median and bin-packing problems, there are also successful examples of using evolutionary algorithms [1]. However, the problem variant that we consider in this work is the more difficult *unsplittable capacitated facility location* problem, where a BS can only service the SSs within range. When user demands are not allowed to be split, flow-based solutions are not useful any more. It is also more difficult than the better studied bin-packing and p-median problems. First, the geographical locations of the devices (BSs and SSs) and their communication range dictate that each BS has a different set of SSs in range. Therefore, they are not equivalent in terms of capability of servicing the users. This is distinctive from bin-packing where all bins are equal in pursuit of using a minimum number of them. Second, rather than assigning the SSs to a fixed number of BSs as in p-median, our goal here is to use as few BSs as possible to minimize the infrastructure costs. To the best of our knowledge, this is the first attempt to use GA to solve such a variant.

3   Adaptive-Population-Size Genetic Algorithm

For constrained combinatorial optimization problems, genetic variation operations in evolutionary algorithms are usually destructive to invalidate an individual

as a candidate solution. Simply applying general and conventional genetic operations without specific heuristics could not be able to exploit the automatic search power of evolutionary algorithms. There is an increasing need for customizing evolutionary methods to closely incorporate the feature of a problem.

We propose a GA tailored to specifically solve the network planning problem. The individual representation and genetic variations are specifically designed suited to the characteristics of the problem. Furthermore, we incorporate a population adjustment method to enhance its search ability. The framework of our APS-GA (Adaptive-Population-Size GA) is described with a view on four specific aspects. We start out with a description of how to represent a solution to the network planning problem using a two-tier genetic structure in order to encode the BS selection and SS assignment separately (Section 3.1). Next, we outline the iterative operations applied to the population to approach the optimum (Section 3.2). Then, Section 3.3 explains the incorporation of an adaptive population size scheme proposed by us in previous work [5]. Note that the fitness of an individual is defined as the number of BSs in service. Thus, there can be many tied solutions with the same fitness but not necessarily the same set of activated BSs and associated SSs. Although this neutral diversity is not observable at the fitness level, it plays an important role in expanding the genotypic search space. The adaptive population size scheme allows a system to dynamically enhance neutral search during different stages of the evolution by population size adjustment. Last, Section 3.4 describes evolutionary operations, including crossover, mutation and a repair heuristic, which are specifically tailored to the problem.

3.1   Individual Representation

Given a set of subscriber stations S and a set of base stations B with their location information, we encode a mapping M from S to B as a two-tier chromosome. At the higher level, i.e., the *BS activation level*, we use an array of length $m = |B|$ to represent the BSs. Each locus i of this chromosome stands for a BS b_i, referring to its service list containing all the SSs assigned to it. If there is no SS connected to a BS (i.e., this BS is not needed), its service list is $\emptyset$. This is referred to as the *SS assignment level*. Such a two-tier representation is depicted in Figure 1.

For a feasible solution, the total length of the service lists should add up to $|S|$, and for each b_i the total capacity demand in the list must not exceed the BS capacity. The division of information into two tiers separates the semantics embedded in an individual. That is, the activation of a BS and the assignment of an SS to an activated BS are encoded in two separate domains. This allows us to control the genetic variations at these two levels independently, which turns out to be fairly powerful as indicated by our experiments. This two-tier genotype is distinctive from the most common representations of GA solutions to combinatorial optimization problems. In such works, the genotype usually takes a fixed form to resemble a biological gene sequence. In particular, a genotype would consist of $|S|$ loci, each of which refers to the index of the BS servicing this SS. Alternatively in the fixed-structure genotype camp, a genotype would
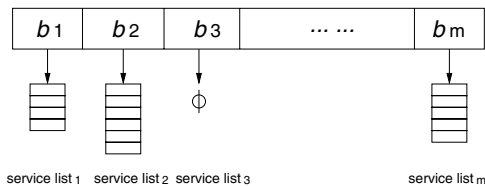
| b_1 | b_2 | b_3 | ... ... | b_m |

service list 1   service list 2   service list 3   service list m

Fig. 1. Two-tier chromosome representation

represent a solution by an indicator matrix $\{0,1\}_{|M|\times|S|}$, where each column i ($i = 1, 2, \ldots, |S|$) contains exactly one 1 and $|B|-1$ 0's. One noticeable exception to this is the "multi-level encoding" in Meunier at al. [9]. In their model, the BS site activation, antenna type selection, and antenna configuration are encoded as three levels. However, the separation in our model is based on a more inherent difference of the information embedded in a solution, i.e., site activation and user assignment.

3.2   Evolution Framework

We evolve a population of individuals with adaptive size in the generational mode to approach the optimum. The process starts with randomly generating a population P_0 of a given size. The value of $|P_0|$, i.e., the initial population size, and other configuration parameters will be detailed in Section 4. Next, each individual's fitness in this initial population is evaluated. Then, the process enters a generational iteration outlined as follows.

1. Randomly pair up individuals of population P_t ($t = 0$ at the start);
2. Crossover each pair of individuals to generate $|P_t|$ offspring;
3. Repair the offspring of previous step;
4. Mutate the offspring;
5. Repair the output of previous step;
6. Evaluate offspring;
7. Calculate the next population size $|P_{t+1}|=f(|P_t|)$ (see Section 3.3);
8. Choose by truncation selection the next population P_{t+1} from the competition pool consist of $|P_t|$ parent and $|P_t|$ offspring individuals;
9. Go to Step 1 if termination criterion is not met.

The iterative process stops when the best fitness in the population has remained the same for s (stagnation threshold) individual evaluations. This termination condition will signal if the evolution stagnates. We measure how fast the algorithm leads the process to a possibly global/local optimum before stagnation by recording the number of individual evaluations elapsed so far.

3.3   Adaptive Population Size Adjustment

Adjusting population size to enhance neutral rearch has been shown by us to be able to accelerate GP evolution [5]. This idea is adopted here in the context of a

GA framework. The central idea is to measure the "rate of accepting nonsynonymous to synonymous genetic changes k_a/k_s" and to use this indicator to adjust the population size dynamically during evolution. Population size is thought to be important for accepting new genetic variations in a system. That is, if the rate of accepting nonsynonymous changes is greater than that of the synonymous changes, i.e., $k_a > k_s$, increasing the population size facilitates accepting new variations, especially the neutral or nearly neutral ones that seem to have little effect on fitness. On the other hand, if $k_a < k_s$, decreasing the population size encourages to accept more genetic variations. If $k_a = k_s$, population size is independent of such an effect. The k_a/k_s ratio is obtained for each generation for population size adjustment. This adjustment is inspired by the neutral theory from molecular biology.

Knowing the importance of neutral search, here for the network planning problem, we apply this adaptive population size approach to our GA. From one generation to the next, N_a denotes the number of *attempted* nonsynonymous changes and N_s for *attempted* synonymous changes. Specifically, for a crossover, if a valid offspring alters its fitness from either parent, this crossover is regarded as a non-synonymous change. A mutation is regarded nonsynonymous if it changes the fitness of an individual. In evolutionary algorithms, not all genetic variations can be favored and accepted by selection. We use M_a and M_s to denote the number of *accepted* nonsynonymous and synonymous changes. Then according the definition in [5], k_a (k_s resp.) is obtained by dividing the accepted nonsynonymous (synonymous resp.) genetic changes by the attempted nonsynonymous (synonymous resp.) genetic changes.

For truncation selection (Section 3.2), the population size of a new generation is at most twice of its previous generation, and an absolute upper and lower limits of the population size is enforced, as described in Section 4.

3.4   Evolutionary Operations

Crossover. Crossover applied to two parents we denote by $x = \langle x_1, x_2, \ldots, x_m \rangle$ and $y = \langle y_1, y_2, \ldots, y_m \rangle$, to obtain two children, $x' = \langle x'_1, x'_2, \ldots, x'_m \rangle$ and $y' = \langle y'_1, y'_2, \ldots, y'_m \rangle$. It is non-trivial to design an efficient crossover operation since this operator has substantial effects on the performance of the algorithm. Here, we propose a Bi-polar Blend crossover that appropriately incorporates the feature of the network planning problem.

The Bi-polar Blend crossover strives to move the SS assignment from less loaded BSs to more loaded ones so that some will eventually no longer be needed and can be de-activated. The crossover operator is therefore a force to drive the activated BSs towards two extremes of being either very heavily or very lightly loaded. Thus, more BSs are expected to be released. To do that, we define that x' inherits the greater load from its parents and y' inherits the less load. Specifically, for each locus i ($1 \leq i \leq m$), we define $x'_i = \begin{cases} x_i \text{ if } l(x_i) \geq l(y_i) \\ y_i \text{ otherwise} \end{cases}$

and $y'_i = \begin{cases} x_i \text{ if } l(x_i) < l(y_i) \\ y_i \text{ otherwise.} \end{cases}$

Repair heuristic. An individual can become infeasible after the application of a genetic variation. Therefore, we conduct the following greedy repair procedure on a modified individual x. For each $s \in S$, we consider all BSs in x that service it, denoted by $\bar{B}$. We first remove all overloaded elements in $\bar{B}$, i.e., with load greater than capacity C. If $\bar{B} \neq \emptyset$, we keep the most loaded element in $\bar{B}$ and release the rest of $\bar{B}$. Otherwise, i.e., s is not serviced by any BS, we search through all BSs within range to find the *best fit* if any. Here, by best fit we mean, when s is added, the BS that has the least residual capacity. If such a best fit exists, s is added to its load. Otherwise, however, we claim that x cannot be repaired, the current genetic variation is aborted and the evolutionary process seeks to produce a new variant in the next iteration.

This repair procedure is applicable to the output of both crossover and mutation operators. Note that it also works in such a trend that the loads of activated BSs are driven towards two extremes.

Mutation. An individual is subject to a point mutation at the BS activation level. Specifically, we select an activated BS uniformly at random and simply clear its service list. We adopt such a mutation scheme for the following reasons.

First, a mutation at the BS activation level, as opposed to the SS assignment level, yields sufficient genetic alteration for the exploration of solutions. A mutation at the SS assignment level, in contrast, would yield a change which is usually too mild. Second, selecting a BS as a unit of mutation confines the changes to one locus of the network. It is, therefore, well modularized. Last, random selection of an activated BS rather than deterministic selection, say of the least loaded BS, has proved to be less directive and more effective in broadening the exploration space as of our preliminary tests.

Because this mutation inevitably invalidates the solution, the previously mentioned repair procedure is also needed.

4   Simulation

We are interested in the effectiveness and efficiency of our APS-GA to the WiMAX planning problem. In addition, we would like to verify the observation that neutral search is critical in our model by investigating the performance improvement yielded by population size adjustment.

4.1   Network Layouts and Algorithm Configuration

Considering that the size and configuration of a network layout may affect the performance of a network planning algorithm, we study two network sizes. The bandwidth demands of an SS is assumed to be 1 unit and the capacity limit for a BSs is 30 units, i.e., at most 30 SSs can be connected to any BS. The deployment area is a 1.0×1.0 2-dimensional space. We consider two network sizes with 30 (300, resp.) and 60 (600, resp.) BS candidate sites (SSs, resp.). All sites and nodes are distributed in the space uniformly at random. The coverage range of
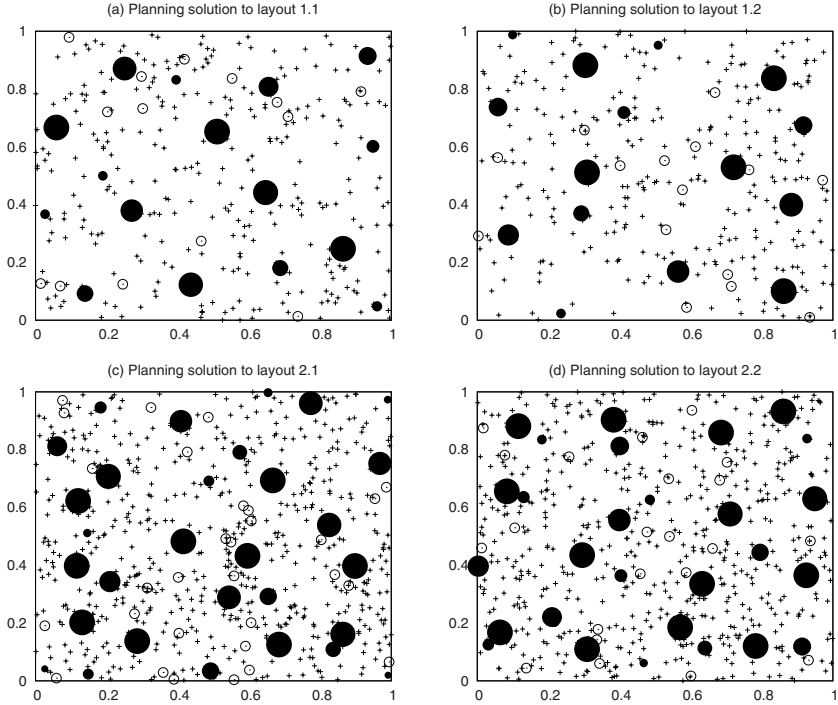
Fig. 2. Network layouts with example optimal solutions. Crosses represent SSs and circles stand for candidate BS sites. Each solid circle means that its BS is activated and its size indicates the load of this BS. Open circles represent inactivated candidate sites. For instance, in (a), the loads of 16 BSs in service vary from 5 to 30 SSs.

each BS is set to 0.2 for the first and to 0.15 for the second network size, such that a BS always has approximately the same number of SSs within range. Two layout instances are generated for each network size denoted by layout 1.1 and 1.2 (of size 1) and layout 2.1 and 2.2 (of size 2).

In all cases, we set the initial population size $|P_0|$ to 200 and the termination stagnation threshold s to 10,000 (evolution is terminated if the best fitness of the population remains unchanged for 10,000 evaluations). Further, we limit the population size to between 100 and 500 when it is varied.

4.2   Results

For each network layout, the results of 100 APS-GA runs are recorded. Figure 2 shows four network layouts with example best solutions. Observe that the loads of the BSs tend to settle on the two extremes, as expected from our Bi-polar Blend crossover and its corresponding mutation and repair operations. We also notice that although some BSs are very lightly loaded, they are indispensable to service the entire network.

Table 1. Results of APS-GA (average data over 100 runs)

	Layout 1.1	Layout 1.2	Layout 2.1	Layout 2.2
Mean of best fitness	16.0	15.9	30.3	29.1
Mean of evaluations	1862	3450	5082	5315
Median of evaluations	1810	3334	4947	4727
95% confidence interval	[1799,1925]	[3176,3723]	[4805,5360]	[4889,5740]
Mean of population size	310	346	234	250

Table 2. Results of FPS-GA (average data over 100 runs)

	Layout 1.1	Layout 1.2	Layout 2.1	Layout 2.2
Mean of best fitness	16.0	16.1	30.2	29.1
Mean of evaluations	2179	3736	5995	5609
Median of evaluations	2170	3459	5732	5250
95% confidence interval	[2157,2201]	[3510,3971]	[5571,6419]	[5199,6019]

Over 100 runs for each layout, the best solutions found by APS-GA are 16 (layout 1.1), 15 (layout 1.2), 30 (layout 2.1), and 28 (layout 2.2). These show that our method is fairly effective since about half of the candidate BSs can be retired and the average load of active BSs can be as high as 70% of the capacity limit. There also can be more than one best solution for each problem instance.

To verify that the adaptive population size scheme has indeed improved the performance of our algorithm, we compare APS-GA to a conventional fixed-population-size GA (FPS-GA) with the same operations and parameter configurations. Since population size fluctuates in APS-GA we average it during an entire evolutionary process over 100 runs for each problem instance. This number is set as the fixed population size for a FPS-GA. Therefore, it is possible and fair to compare these two algorithms.

Tables 1 and 2 show the results of these two algorithms. We collect the mean best fitness achieved at the end of evolution. Recall that evolution terminates when the best fitness of a population does not improve for 10,000 evaluations. Evaluations before stagnation are recorded as the computational cost for a population to reach its best solution. The means, medians, and the 95% confidence intervals of the number of individual evaluations are shown. It can be observed that the two algorithms perform equally well at achieving best solutions. However, APS-GA is noticeable more efficient since it always incurs smaller computational cost.

5   Concluding Remarks and Future Research

A specifically designed adaptive-population-size GA is proposed in this article to solve the WiMAX network planning problem. Computer simulation verifies its effectiveness and efficiency. This work emphasizes that it is critical to thoroughly consider the properties of a problem when applying evolutionary algorithms.

A number of interesting extensions of this work present themselves. First, other features of the solution should be considered, such as resilience. If many BSs are fully loaded, the network might be fragile to changes of SSs. Resilience to demand fluctuation (e.g., fluctuating numbers of SSs in different areas of the 2D map) can also be useful to address user mobility to a degree. Further we can see how our algorithm can adapt in such a scenario, perhaps in real time. Second, an extension of our method to solve the WiMAX network planning problem in MMR mode should be possible. That is, another layer of relay stations are used to extend network coverage. It will be interesting to see how these two levels of mapping affect each other in the framework.

Acknowledgements

References

1. Alp, O., Erkut, E., Drezner, Z.: An efficient genetic algorithm for the p-median problem. Annals of Operations Research 122(1-4), 21–42 (2003)
2. Amaldi, A., Capone, A., Malucelli, F.: Planning UMTS base station location: optimization models with power control and algorithms. IEEE Transactions on Wireless Communications 2(5), 939–952 (2003)
3. Ghosh, D., Gupta, A., Mohapatra, P.: Scheduling in multihop WiMAX networks. ACM Mobile Computing and Communications Review 12(2), 1–11 (2007)
4. Hu, J., Goodman, E.: Wireless access point configuration by genetic programming. In: Proceedings of the IEEE Congerss on Evolutionary Computation (CEC), pp. 1178–1184 (2004)
5. Hu, T., Banzhaf, W.: The role of population size in rate of evolution in genetic programming. In: Vanneschi, L., Gustafson, S., Moraglio, A., De Falco, I., Ebner, M. (eds.) EuroGP 2009. LNCS, vol. 5481, pp. 85–96. Springer, Heidelberg (2009)
6. Levi, R., Shmoys, D.B., Swamy, C.: LP-based approximation algorithms for capacitated facility location. In: Bienstock, D., Nemhauser, G.L. (eds.) IPCO 2004. LNCS, vol. 3064, pp. 206–218. Springer, Heidelberg (2004)
7. Li, B., Qin, Y., Low, C.P., Gwee, C.L.: A survey on Mobile WiMAX. IEEE Communications Magazine 46(12), 70–75 (2007)
8. Lin, P., Ngo, H., Qiao, C., Wang, X., Wang, T., Qian, D.: Minimum cost wireless broadband overlay network planning. In: Proceedings of the IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM), pp. 228–236 (2006)
9. Meunier, H., Talbi, E.G., Reininger, P.: A multiobjective genetic algorithm for radio network optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC), pp. 317–324 (2000)
10. Tang, K.S., Man, K.F., Kwong, S.: Wireless communication network design in IC factory. IEEE Transactions on Industrial Electronics 48(2), 452–459 (2001)
11. Yu, Y., Murphy, S., Murphy, L.: A clustering approach to planning base station and relay station locations in IEEE 802.16j multi-hop relay networks. In: Proceedings of the IEEE International Conference on Communications (ICC), pp. 2586–2591 (2008)