Executive Overview

TITLE:	D2.3.2 - TSBS Optimal Location Problem (test and evaluation)

SUMMARY: This deliverable aims at solving the problem of placing Traffic Service Base Stations (TSBSs) in order to maximize the coverage of the Carlink wireless platform. Our proposal is to apply metaheuristics techniques for obtaining good quality solutions as soon as possible.

GOALS:

1. Outline of the optimization techniques used for solving the problem.
2. Presentation of the different regions where the optimization will be applied.
3. Report of the experimental results.

CONCLUSIONS:

1. The results reveal that is possible to find good quality solutions which cover a wide road region using a low number of TSBSs. The multi-objective approach is our advised proposal because it allows the users to select the solution which best adjust to its requirements.

D2.3.2 - TSBS Optimal Location Problem (test and evaluation)

CARLINK::UMA

February 28, 2008

1 Introduction

The aim of the CARLINK project is to develop an intelligent wireless traffic service platform for communicating vehicles. The platform is divided in three different parts (consult [3] for further information): the Traffic Service Central Unit (TSCU), the Traffic Service Base Stations (TSBS) and the Mobile End Users (MEUs). The TSCU and MEUs interchange information while the TSBSs act as bidirectional data transceivers.

The TSCU is placed in a specific place, while the TSBSs must be placed along the road. This deliverable aims at solving the problem of placing TSBSs in a specific area. A complete definition of this problem can be found in [4]. This task is very important since the most part of the road must be covered trying to use the lesser number of TSBSs. When the area to optimize is large and the number of TSBSs is high, it is very difficult of placing them manually, so it is necessary the automation in the solution of the problem. Our purpose for solving this problem is based on metaheuristics [2] techniques, concretely by means of evolutionary algorithms. In short, a metaheuristic can be defined as a top-level general strategy which guides other heuristics to search for good solutions in a reasonable time. They are approximated and non-deterministic.

This deliverable is structured as follows: Section 2 describes the most important aspects of our metaheuristic approach. Section 3 presents the different results. Finally, the conclusions of this work are shown in Section 4.

2 The Algorithm

This section outlines the used algorithm for solving the TSBS Optimal Location Problem (TSBS-OLP). Section 2.1 presents the kind of metaheuristic that we have used for solving the problem: Evolutionary Algorithms (EAs [1]). In Section 2.2 we describe the way of representing the solutions and the operators which alter them with the goal of obtaining new ones (offspring). We have developed two different approaches: considering TSBS-OLP as a mono-objective problem and as a multi-objective problem. Section 2.3 comments both approaches.

2.1 Evolutionary Algorithms

Evolutionary Algorithms (EAs) [1] are search methods that take their inspiration from natural selection and survival of the fittest individuals in the biological world. In these algorithms, a set of solutions (population of individuals) is iteratively improved by means of the evolutionary cycle (see Figure 1).

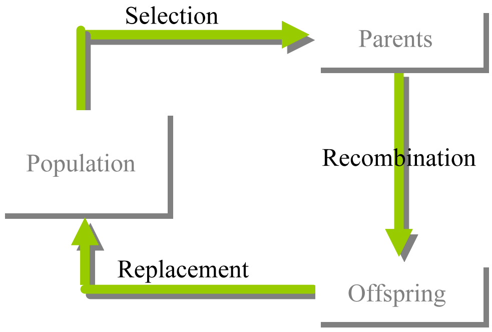


Figure 1: The evolutionary cycle illustrates the basic idea of the EAs: from a set of solutions (population), new solutions are created iteratively. These new solutions compete with the others for staying in the population. The goal is to improve iteratively the quality of the solutions in the population

This cycle starts with the selection phase, where a certain number of individuals (parents) are selected from the population using a specific criterion. After that, in the recombination step, the selected individuals are combined among them with the goal of creating new ones. The new individuals compete against the others for staying in the population (replacement). This way, in each iteration of the cycle, the population can contain better individuals.

2.2 Representation of the solutions and operators

This subsection presents the way of representing the solutions and the operators which modify them. In the TSBS-OLP, we consider a set of coordinates as candidate places (positions in the crossroads where there can be road signals or traffic lights to place TSBSs). The solution must contain information about the final selected points for placing TSBSs in the optimization area. This way, we represent a solution as a binary vector of size N (being N the total number of candidate places). The '1' value indicates the presence of a TSBS in the corresponding place ('0' otherwise). Figure 2 illustrates an example of solution where three TSBSs are placed in the optimization area.

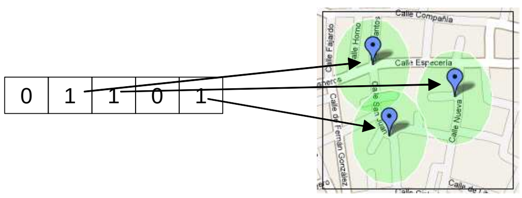


Figure 2: Example of a solution. In the binary vector, each '1' represents the presence of a TSBS (illustrated in the right picture using drawing pins) in the corresponding place

We propose two operators to modify solutions in our model:

- **Bit Flip Mutation**: This operator selects one bit randomly in the vector and changes its value (Figure 3).
- **Single Point Crossover**: given two solutions, one point is selected randomly (cross point). The binary string from the beginning of the crossover point is copied from one solution, while the rest is copied from the other solution (Figure 4).

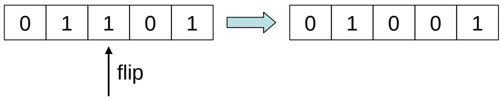


Figure 3: Example of bit flip operator

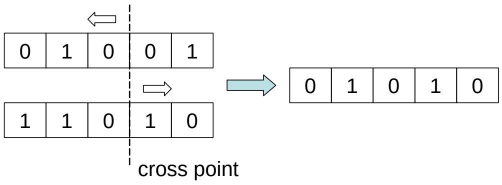


Figure 4: Example of single point crossover

2.3 Two approaches: Mono-Objective and Multi-Objective overview

The generated solutions in the evolutionary cycle (Figure 1) compete with the existing ones in the population for surviving. We need to determine the quality of all the generated solutions, that is, we have to design a function (or functions) which evaluates a solution. The difference between the mono and multi-objective approach lies in the number of functions to be optimized.

- **Mono-objective**: there is only one function, which is defined as follows:

$$f(\vec{x}) = \frac{|M'(\vec{x})|}{Coverage(\vec{x})^2} \tag{1}$$

  where $\vec{x}$ represents the encoded solution and $M'(\vec{x})$ the set of all the used TSBSs. The function is the defined in [4], but in this case it is inverted since we are using a software library (jMetal [7]) which works by minimizing the objective functions. This function includes both the reached road coverage and the number of used TSBSs in the solution. This way, in order to compare two solutions, that one with lower value in the previous function is the best. The algorithm will return the solution with the lowest value found in the function.

- **Multi-objective**: In this approach, the coverage and number of used TSBSs are considered as independent functions (equations 2 and 3). This way, multi-objective optimization does not restrict to find a single solution, but a set of solutions called *nondominated solutions* [5] (Figure 5 illustrates an example). Each point in the graphic represents the cost of a solution. This approach allows the final users to select a solution which best adjust to its requirements.

$$f_1(\vec{x}) = |M'(\vec{x})| \tag{2}$$

$$f_2(\vec{x}) = Coverage(\vec{x}) \tag{3}$$

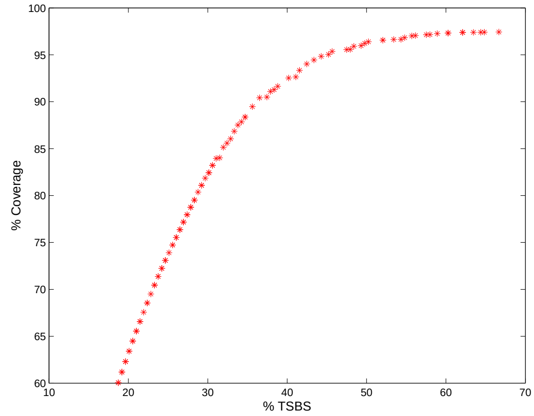


Figure 5: Example of a graphic presenting the result of a multi-objective algorithm: the set of non-dominated solutions

3 Results

The algorithms are applied to three different regions placed in spanish cities: Madrid, Valencia and Málaga. In each region, we have selected a certain number of candidate points for placing TSBSs. Table 1 shows the main characteristics of the optimization areas while Figure 6 illustrates a Google Maps[1] snapshot on them.

Figure 6: Map view of the three optimization areas. The square points represent the places where TSBSs can be placed

Table 1: Instance characteristics

	Madrid	Valencia	Málaga
Optimization area size (m)	749.13 x 1520	804 x 1570	308 x 410
Candidate places for TSBSs	86	219	91
Average minimum distance (m) among nearest candidate points	89.39	41.57	23.09
Size of square grid discretization	413 x 384	850 x 437	427 x 321

[1] http://maps.google.com/

As we commented in the previous section, we have developed two different approaches. A Steady-State Genetic Algorithm (SSGA [8]) is the selected choice for the mono-objective approach, while the Non-dominated Sorting Genetic Algorithm II (NSGA-II [6]) is the proposal in the multi-objective approach. The parameterization of the tests is shown in Table 2. Thirty independent runs have been executed for each instance and algorithm. The algorithms work with a set of 100 solutions. For avoiding configurations with low coverage, all the solutions which cover a road percentage lower than 60% will be discarded by the algorithm. As we are using WiFi-based TSBSs, we assume a radio coverage of 75 m (this parameter is configurable).

Table 2: Test Parameterization

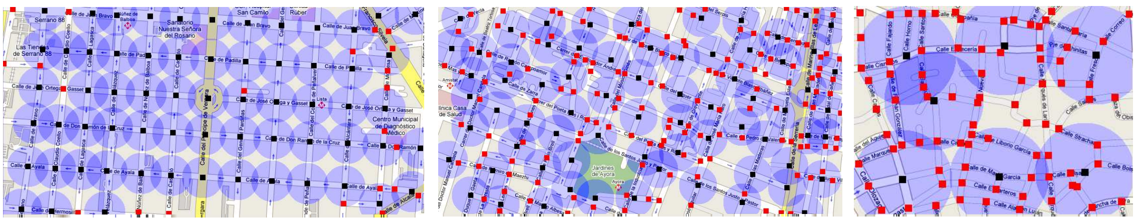
Independent runs	30
Number of evaluations	100000
Population size	100
Selection	Binary tournament [9]
Bit-Flip probability	0.3
Single point crossover probability	0.7
Minimum coverage restriction	60%
TSBS radio coverage	75 m

Table 3 presents the obtained results using the mono-objective approach. The **Avg.** column shows the average value of the evaluation function in all the independent runs and the standard deviation. The cost of the best found solution is written in the **Best** column. Finally, the column **Time** shows the average required time for the algorithm in minutes. The execution time of the algorithm depends on the size of the optimization area and the number of candidate points for placing TSBSs. This way, Madrid and Valencia instances present the longest computational times. The standard deviation values reveal the low degree of variation among different executions of the algorithm, so the quality of the final solutions are very similar among them.

Table 3: Mono-objective results

	Avg.	Best	Time (min)
Madrid	8.2e-3$\pm 4.1e-5$	8.1e-3 (64 TSBSs 88.39%)	21.84
Valencia	9.7e-3$\pm 8.6e-5$	9.5e-3 (71 TSBSs 86.22%)	25.17
Málaga	1.07e-3$\pm 2.9e-5$	1.01e-3 (7 TSBSs 83%)	5.72

Figure 7 shows the geographical representation of best found solutions in each city using the mono-objective approach. For example, in Málaga, due to the high density of candidate points, it has been possible to cover the 83% of road using only 7 TSBSs. In Madrid and Valencia, more TSBSs are needed for covering the 88.39% and 86.22% of road, respectively. In Valencia, the percentage of used TSBSs is 32%, while in Madrid, the average distance between the candidate points forces to use a high percentage of TSBSs (74%).



Madrid Valencia Málaga

Figure 7: Best set of solutions in the three cities using the mono-objective approach

In the multi-objective approach, the execution times are a little longer (included between 6 and 23 minutes) than the obtained ones in the mono-objective approach because multi-objective optimization requires specific operations (e.g. ranking and crowding [6]). For comparing the quality of the resulting set of solutions given by this approach, we use the hypervolume metric [10]. This metric assigns a value between 0 and 1 to each set of solutions (the solution set with the highest hypervolume value is the best one). This way, Figure 8 shows the best set of solutions according to the hypervolume. In the graphics, each point represents the cost of a resulting solution. The X-axis shows the percentage of candidate places used for placing TSBSs, while the Y-axis represents the resulting coverage.

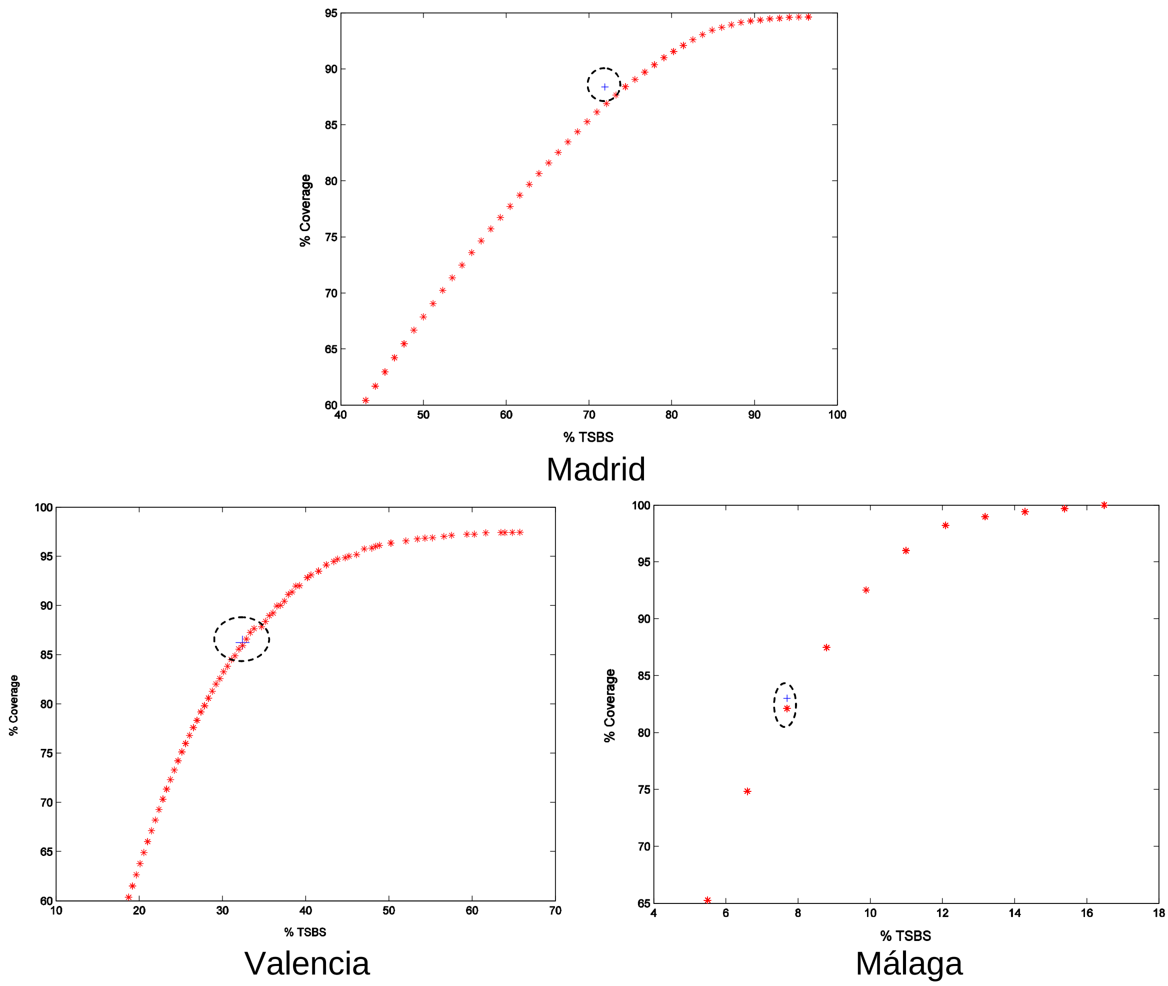
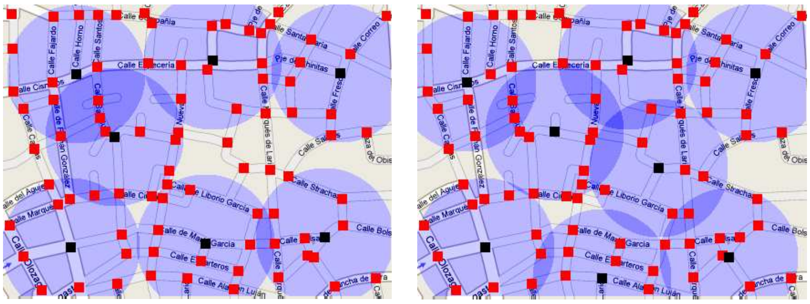


Figure 8: Best fronts according to the hypervolume in Málaga, Madrid and Valencia. The cross point represents the cost of the best mono-objective found solution

In Málaga it has been possible to cover all the road and, in spite of the topology of the instance (a high density of TSBSs), the full coverage is reached using a low percentage of TSBSs (16.6%). The relation between the reached coverage and the percentage of used TSBSs in the Madrid area is almost linear due to the low density of candidate places. In the Valencia area, the set of solutions shows as from the 50% of used TSBSs, it is very difficult to increase the coverage.

In Figure 8, the cross points represent the cost of the best found mono-objective solution. If we compare the best mono-objective solutions with the associated ones in the multi-objective approach which use the same percentage of TSBSs, the resulting mono-objective solutions reach a little higher coverage (this fact is more clear in Málaga). However, the multi-objective algorithm allows the user to select solutions of a wide range of values. This way, the final user can select the best solution according to his budget.

The mono-objective approach has an additional drawback, which it is illustrated in Figure 9. We show 2 different solutions in the figure, the first one uses 7 TSBSs for covering the 81.3% of road, while the second one needs 8 TSBSs being the resulting coverage of 86.2%. According to the evaluation function used in the mono-objective algorithm, the first solution has a lower cost. So, if both solutions are compared, the algorithm will select the solution which uses 7 TSBSs. However, it is very probably that the final user prefers to use one additional TSBS for covering a wider road region. It is possible to solve this problem applying a higher weight to the coverage in the evaluation function, but we have not made this decision because the number of TSBSs in a solution would not have influence in the function.



(a) (b)

Figure 9: Two solutions in Málaga which use 7 (a) and 8 (b) TSBSs. According to the function evaluation in the mono-objective approach, the first solution is the best one. However, the final user can prefer to use one additional TSBS for covering a wider road region

4 Conclusions

This deliverable aims at solving the problem of maximizing the coverage of road and minimizing the number of TSBSs in a geographical area. We have used approximated techniques (metaheuristics) for solving the problem. Two different approaches have been developed: the mono-objective version returns the best solution found with respect to an evaluation function while the multi-objective algorithm returns a set of solutions, being the final user who must decide the best solution according to his own opinion. The algorithms have been applied to three geographical areas placed in spanish cities, where we have identified the candidate points for placing TSBSs in the crossroads. The results reveal that is possible to find high quality solutions which cover a wide road region using a low percentage of TSBSs.

References

[1] T. Back, D.B Fogel, and Z. Michalewicz. Handbook of evolutionary computation. *Oxford University Press. 1997.*

[2] Christian Blum and Andrea Roli. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Comput. Surv.*, 35(3):268–308, 2003.

[3] CARLINK. CARLINK D2.2 - Platform Specification. CARLINK consortium, February 2008.

[4] CARLINK::UMA. CARLINK::UMA D2.3.1- TSBS Optimal Location Problem (definition). University of Malaga, February 2008.

[5] C. A. Coello, G. B. Lamont, and D. A. Van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems.* Genetic and Evolutionary Computation Series. Springer, second edition, 2007.

[6] K. Deb, A. Pratab, S. Agarwal, and T. Meyarivan. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 2002.

[7] Juan J. Durillo, Antonio J. Nebro, Francisco Luna, Bernabé Dorronsoro, and Enrique Alba. jMetal: a Java Framework for Developing Multi-objective Optimization Metaheuristics. Technical Report ITI-2006-10, Departamento de Lenguajes y Ciencias de la Computación, University of Málaga, E.T.S.I. Informática, Campus de Teatinos, 2006.

[8] D.E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1989.

[9] B.L. Miller and D.E. Goldberg. Genetic Algorithms, Selection Schemes, and the Varying Effects of Noise. *Evolutionary Computation*, 4(2):113–131, 1996.

[10] E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: a comparative case studyand the strength Pareto approach. *Evolutionary Computation, IEEE Transactions on*, 3(4):257–271, 1999.