

Document downloaded from:

<http://hdl.handle.net/10251/176241>

This paper must be cited as:

Jordán, J.; Palanca Cámara, J.; Del Val Noguera, E.; Julian Inglada, VJ.; Botti, V. (2021). Localization of charging stations for electric vehicles using genetic algorithms. *Neurocomputing*. 452:416-423. <https://doi.org/10.1016/j.neucom.2019.11.122>



The final publication is available at

<https://doi.org/10.1016/j.neucom.2019.11.122>

Copyright Elsevier

Additional Information

# Localization of Charging Stations for Electric Vehicles using Genetic Algorithms

Jaume Jordán<sup>a</sup>, Javier Palanca<sup>a</sup>, Elena del Val<sup>b</sup>, Vicente Julian<sup>a</sup>, Vicente Botti<sup>a</sup>

<sup>a</sup>*Institut Valencià d'Investigació en Intel·ligència Artificial (VRAIN)  
Universitat Politècnica de València  
Camino de Vera S/N 46022 Valencia (Spain)  
{jjordan,jpalanca,vinglada,vbotti}@dsic.upv.es*

<sup>b</sup>*Departamento de Informática e Ingeniería de Sistemas  
Universidad de Zaragoza, Escuela Universitaria Politécnica de Teruel  
Calle Atarazana 2, 44003 Teruel (Spain)  
edelval@unizar.es*

---

## Abstract

The electric vehicle (EV) is gradually being introduced in cities. The impact of this introduction is less due, among other reasons, to the lack of charging infrastructure necessary to satisfy the demand. In today's cities there is no adequate infrastructure and it is necessary to have action plans that allow an easy deployment of a network of EV charging points in current cities. These action plans should try to place the EV charging stations in the most appropriate places for optimizing their use. According to this, this paper presents an agent-oriented approach that analyses the different configurations of possible locations of charging stations for the electric vehicles in a specific city. The proposed multi-agent system takes into account data from a variety of sources such as social networks activity and mobility information in order to estimate the best configurations. The proposed approach employs a genetic algorithm (GA) that tries to optimize the possible configurations of the charging infrastructure. Additionally, a new crossover method for the GA is proposed considering this context.

*Keywords:* genetic algorithm, crossover, multi-agent system, charging station, electric vehicle

---

## 1. Introduction

Electric vehicles (EV) are an alternative to fossil-fuel vehicles. Among the advantages of EV, it is worth to mention that EV reduce the carbon emissions, the air pollution and the noise [1]. The market potential of electric vehicles could be limited by the current charging infrastructure. One of the factors that can reduce the adoption of electric vehicles in cities is the infrastructure that would limit the length of journeys to be made by users. In addition, the charging time is another factor that might influence in the adoption of EV. Although the cost of charging an electric vehicle is less than the cost of a full tank, if charging times are relatively long, users will prefer the immediate charging of traditional vehicles. For example, charging a battery can take between 4 and 8 hours [2].

There are many advocates who argue that good planning of charging stations has a relevant impact on the service quality and operation efficiency and would increase consumer adoption rates [3, 4]. This would reduce the psychological effect of "anxiety" [5] (i.e., fear of not having enough battery charge to reach the destination) as they could rely on having accessible "nearby" stations along the daily journeys made in the city. However, the necessary investment in infrastructure is very costly, it is therefore necessary to manage investments in order to implement electric recharging stations mainly in those areas with maximum impact. It is important to be able to determine the optimal infrastructure planning needed to provide an acceptable charging service for consumers. There is a need for joint and comprehensive analysis taking into account information from multiple data sources. It is important to take into account which areas of cities are most heavily traveled by vehicles (traffic), what population there is in each area of the city, what type of activities are carried out in each area of the city (land use), or how much activity there is in a given area (social networks). This problem is more complex if, in addition, when planning, it is necessary to take into account forecasts for future extensions where necessary.

To help city policy-makers to allocate public resources efficiently to support the deployment of charging infrastructure, it is essential to provide a systematic approach to quantify the benefit of providing public charging opportunities, as well as to determine where to locate charging stations subject to limitations on the range of vehicle travel [6, 7].

This work addresses the problem of optimizing the placement of new infrastructures for EV charging. We design a system, made up of a set of

agents, which gathers information from heterogeneous data sources of the city such as open data web portals to obtain data on traffic, population in different parts of a city, data from Google applications that provide information on the average time spent in certain points of the city or geolocalized information from social networks to estimate user activity. Finally, a search for the optimal solution is made by means of a genetic algorithm. This optimization checks the possible locations of the charging stations and tries to distribute throughout the city the required stations. Moreover, it has to satisfy constraints such as the maximum poles per station, it must guarantee the supply of electricity for stations, and it has to consider urban information such as population and traffic per neighborhood to optimize the investment. The genetic algorithm can be configured by the user of the system to adapt it to the specific conditions of each city.

This work is an extension of a previously published paper [8], which also adds a crossover operator specifically designed for a geolocalized domain. Additionally, some experiments about this operator are presented and show a remarkable improvement of performance for these type of domains. The paper is structured as follows: Section 2 analyzes previous works related to the placement of electrical stations. Section 3 presents a multi-agent system that gathers data and executes the genetic algorithm proposed in this work. Section 4 presents a case study to illustrate how the system works. Then, an experimental evaluation focused on a new crossover operator for the genetic algorithm is performed in Section 5. To conclude, Section 6 presents some conclusions and future works.

## 2. Related Work

Currently, there is a paradigm shift in transport in cities from traditional vehicles to electric vehicles. This is partly due to increased environmental awareness in society and the observation of some of the effects of pollution. As a result, many cities have begun to popularize the use of the electric vehicle and have begun to investigate further the most appropriate location of charging stations to service new users of these vehicles [9].

There are different perspectives to cover the problem of placing electric stations. Some initiatives try to provide support, through implementation guidelines, to the selection problem of the most appropriate location of charging stations in a city [10, 11]. There are other approaches that consider different information sources and optimization algorithms for the solution of

the problem (see Table 1). Erbacas et al. [12] propose the combination of geographic, economic and urbanity information to address the Electric Vehicle Charging Station (EVCS) site selection. The authors use fuzzy techniques and preference by similarity to ideal solution to choose the optimal EVCS sites. However, it is not always possible to know the optimal solution.

Lin et al. [13] propose a Mixed Integer Linear Programming (MILP) model to identify the optimal location and size of EVCS in cities. They consider traffic flow data, aggregated charging profiles and land-use classifications as inputs of their MILP model to identify the optimal location and size of EVCS in cities. The goal of their proposal is to maximize the total profits of new charging stations. Li et al. [14] also consider the users' charging behavior data to decide the distribution of charging stations. Their proposal is mainly oriented to understand and satisfy the real needs of the current EV drivers. However, providing a planning that goes beyond the current users would have to consider other aspects such as traffic, uses of different areas of the city or population in each area. A similar approach is proposed in [15]. The authors considered that charging occurred more likely at the end rather than in the middle of a trip. They propose a genetic algorithm to maximize the charge quantity, optimize the layout of public charger and decide which type of chargers should be installed at each location. Similarly, Nie et al. [7] propose an analysis of journeys made by EV using an optimization model. The objective of the model described here is to determine the most appropriate charging power for each station and how many stations would be needed to cover the journeys made, satisfying a particular level of service and minimizing cost. In the paper presented by Wood et al. [4] the authors focus on establishing an approximate number of charging stations to increase vehicle utility on the one hand and how the stations can be strategically located to maximize the future benefit of EV users on the other. This approach makes use of driver behavior, vehicle performance, travel profiles, battery attributes, environmental conditions and charging infrastructure to optimize EV and charging station performance. All these works are based on an existing EVCS infrastructure and assume that the models are allowed to access to user profiles. These facts may limit the application of the proposed models to contexts where there is no previous infrastructure and therefore no information on the behavior patterns of EV users.

In the paper presented in [16] authors propose to take into account the destination of journeys made by EV users (i.e. restaurants, shops or banks) to decide the location of a charging station. Points of interest (POIs) are

these destinations. Specifically, the paper describes a model that allows these POIs to be classified depending on how attractive they are to users of electric vehicles. To locate a station, the work considers two approaches: i) use a method based on obtaining maximum coverage of demand; ii) use an iterative method that penalized the location of a new recharging point if it was near another existing point. The main drawback of this proposal is that it only uses travel information for EV users. Similar work was proposed in [17], where authors transformed the problem of locating stations into a problem of maximum coverage of a weighted network where the weight of the arcs was the number of cars going from one origin to one destination.

Genetic algorithms (GAs) have been also proposed as a suitable technique to deal with the problem of determining locations for charging stations. Wei et al. [18] present a tool based on a GA to model the demand for taxis, stations and electric taxis. In their proposal, the radius of action of a taxi, the charging time and the capacity of the EV stations are taken into account as input parameters. In the work proposed by Dong et al. [5] they also make use of a GA to determine the location and type of charging stations. The algorithm aims to minimize lost trips taking into account budgetary constraints. The city is modeled as a grid in which each cell stores information on the number of journeys ending in it. To select station locations, the most visited destinations are taken into account.

GAs consist of several elements, including genome coding, population generation, fitness function and selection procedure, crossover, mutation and stop criteria. In the work presented by Wei et al. [18] the crossover was performed by the single-point crossing operator. Dong et al. [5] decided to use an existing commercial tool to carry out their experiments, however, their work does not detail which operators they use. In our work, we analyzed the main components involved in the proposed GA and propose a new crossover operator that improves the performance of the algorithm.

### 3. Multi-Agent System Application

In this work we propose an application that evaluates a set of points of interest throughout a city in order to optimize the best distribution of charging stations for EVs that minimizes the called *range anxiety* that is mainly determined by the infrastructure of the charging stations [19]. For this purpose, a multi-agent system manages a genetic algorithm designed in this work. This system tries to reduce the search problem by introducing

	[7]	[4]	[16]	[18]	[5]	[17]	[12]	[13]	[17]	[15]	<b>Proposal</b>
Traffic, frequent routes	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Social data											✓
Population											✓
Time spent in a PoI	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Cost per station	✓				✓	✓	✓				✓
Demand per station		✓	✓	✓	✓	✓					✓
GIS							✓	✓			✓
Land use							✓	✓			✓
Charging profiles								✓		✓	

Table 1: Comparison of the types of information considered by the approaches that deal with the EV charging station location.

information about the city and a heuristic that makes the search faster and prevents premature convergence to a less optimal solution. The SPADE<sup>1</sup> agent platform [20] was used to develop the MAS. SPADE allowed us an easy prototyping and further development of the MAS by means of the instant messaging support provided by the XMPP<sup>2</sup> protocol [21] in SPADE. All messages are sent in real-time between the agents described below using the presence notification mechanism that the platform provides.

### 3.1. Data collecting and processing

Before running the GA there is a set of SPADE agents of the application’s MAS that collect information related with the city we are working on. This allows our application to be run and reused in other municipalities with not too much effort. The information collected by the application is:

- **Points of Interest:** The GA starts with a set of Points of Interest (PoIs) that are candidates for hosting a charging station. These points are manually selected according to the city’s urban development plan. Subsequently, the GA will be responsible for reducing this set, but the *PoI agent* is in charge of detecting and classifying PoIs candidates to be selected for the input of the algorithm.
- **Population information:** The *Urban agent* gathers information about the population living in each area of the city. It queries census sections of open data portals and obtains the population in the different neighborhoods or blocks of the city under study.

<sup>1</sup><https://github.com/javipalanca/spade>

<sup>2</sup><http://www.xmpp.org>

- **Traffic information:** The amount of traffic that an area of a city has is obtained by the *Traffic agent*.
- **Popularity information:** The influx of an area determines its popularity. Based on the number of people that visits each PoI and how much time they spend in the area, we can estimate its popularity. The *Popularity agent* uses an exhaustive search on third party services (such as Google cards on the search engine) to locate this data.
- **Social Networks information:** Geo-tagged information of social networks can be used to estimate the activity that occurs in that area. The *Social Network agent* tracks some networks (Twitter and Instagram) to collect all the geolocated items in an area.

Once the data is collected, the *Data Processing Agent* aggregates all the information obtained and serves it through geo-queries that simplify obtaining the information around an individual PoI.

### 3.2. Application's data flow

The application is run in six stages that include the selection of PoIs, transforming the PoIs to a Voronoi diagram, the extraction of city data, the characterization with this data of the polygons representing PoIs, the execution of the GA and the visualization of the results (see Figure 1).

The output of the first stage is a set of PoIs  $P$  that are candidates for the location of a station  $s_i$ . The PoI agent may use as first PoI set the location of public parking and garages, which have a large number of visitors.

At the second stage the PoI agent builds a Voronoi diagram, using the selected PoIs as centroids, to determine the area of influence of each point. This area allows us to better determine a full polygon, and not only a single point, to consider the location of a station. It also helps to calculate which data collected from the city at the next stage belongs to which polygon.

Third and fourth stages collect data from the city and aggregate it to the Voronoi polygons, which get characterized with population density, traffic intensity and social networks activity. These stages are performed by the Data Processing Agent and the collecting agents described above.

The fifth stage is performed by the Emplacement Optimizer Agent, which is described in depth in the next section. This agent uses the characterized polygons and the constraints of the problem, such as the maximum number of stations to install in the city, to make a heuristic search using a GA.



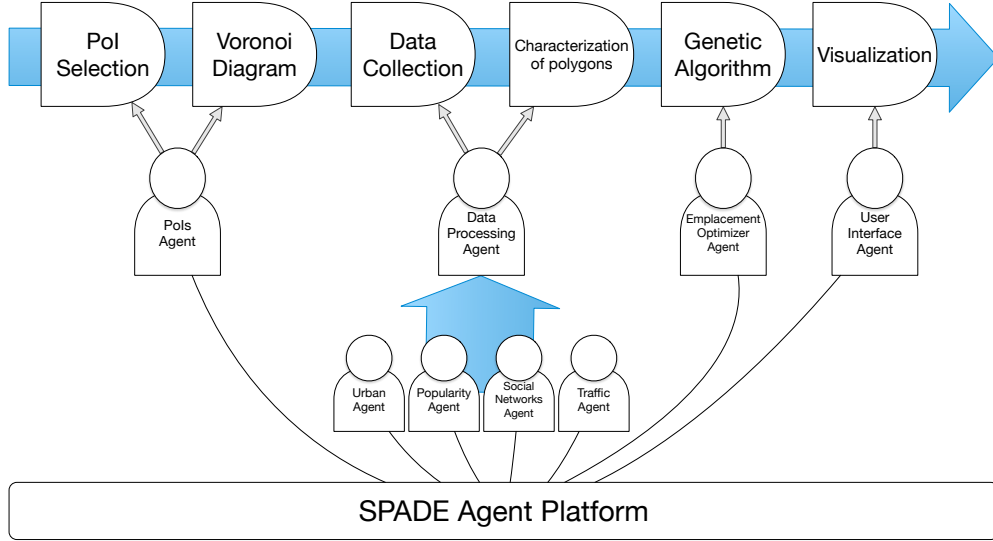


Figure 1: Application's data flow

The last stage of the application is performed by the User Interface Agent, which runs a website where the results of the application can be examined on a map, query the properties of each proposed station and run new executions.

### 3.3. The Emplacement Optimizer Agent

This agent is responsible for determining the most appropriate locations for the emplacement of electric charging stations. In this subsection, we describe the formalization of the problem of finding the most suitable configuration for the location of the stations and the GA used.

#### 3.3.1. Location of electric charging stations problem

The goal of the Emplacement Optimizer agent is to analyze a set of possible configurations and select the most appropriate according to a utility function and a set of pre-defined PoIs. We consider a set of possible locations (i.e., PoIs)  $P = \{p_1, \dots, p_n\}$  belonging to the city under study. Each PoI  $p_i$  is

described by a set of attributes  $p_i = \{a_1, a_2, \dots, a_n\}$ . Specifically, we consider the following attributes: (1)  $a_{population}$ , population in the area around  $p_i$ ; (2)  $a_{traffic}$ , average traffic in the area; (3)  $a_{time}$ , average time spend by citizens in public places in the area; (4)  $a_{social}$ , geolocated social networking activity in the area; (5)  $cost\_area$ , cost depending on the area covered by the stations; (6)  $cost\_per\_charger$ , cost per each charging station.

Besides the set of possible locations, the agent considers a set of charging stations  $S = \{s_1, \dots, s_n\}; 0 < s_i \leq max\_chargers\_per\_poi$ , that could be deployed in the city. The number of chargers per station in a PoI ranges from 0 to a constant value  $max\_chargers\_per\_poi$  (4 in our tests).

Considering the set of predefined PoIs  $P$  and the set of charging stations  $S$ , the agent is able to provide the most appropriate configuration  $C_i$  for the location of stations in PoIs. A configuration  $C_i = \{\{p_1, s_1\}, \{p_2, s_2\}, \dots, \{p_n, s_n\}\}$  consists of a set of pairs PoI-stations. Each configuration has associated a fitness value  $V(C_i)$  according to its suitability.

### 3.3.2. Genetic Algorithm

Our Genetic Algorithm (GA) [22] is implemented using the *DEAP*<sup>3</sup> library. We propose a GA that will generate solutions where each individual is a possible configuration of the charging stations. Initially, the algorithm considers a random population of  $N$  individuals. Each individual is a feasible solution  $C_i$  to the problem. Specifically, a chromosome is composed of a set of locations  $P$  and the number of stations per location  $S$ . An example of the chromosome encoding and functions of the GA are shown in Figure 2.

To evaluate the suitability of each chromosome (i.e., solution), the agent uses a fitness function that evaluates the quality of the solution considering the suitability of placing charging stations in the selected points using the parameters defined above. The fitness function considered is as follows:

$$V(C_i) = \sum_{\forall p_i \in C} ((\omega_p \cdot a_{population} + \omega_{tr} \cdot a_{traffic} + \omega_t \cdot a_{time} + \omega_s \cdot a_{social}) - (\omega_a \cdot cost\_area + \omega_c \cdot cost\_per\_charger \cdot |s_i|)), \quad (1)$$

where  $a_{population}$  denotes the population that is covered by the charging stations located in  $p_i$ ;  $a_{traffic}$  refers to the traffic generated in the area of  $p_i$ ;  $a_{time}$  refers to the average time citizens spend in public/commercial places

---

<sup>3</sup><https://github.com/DEAP/deap>

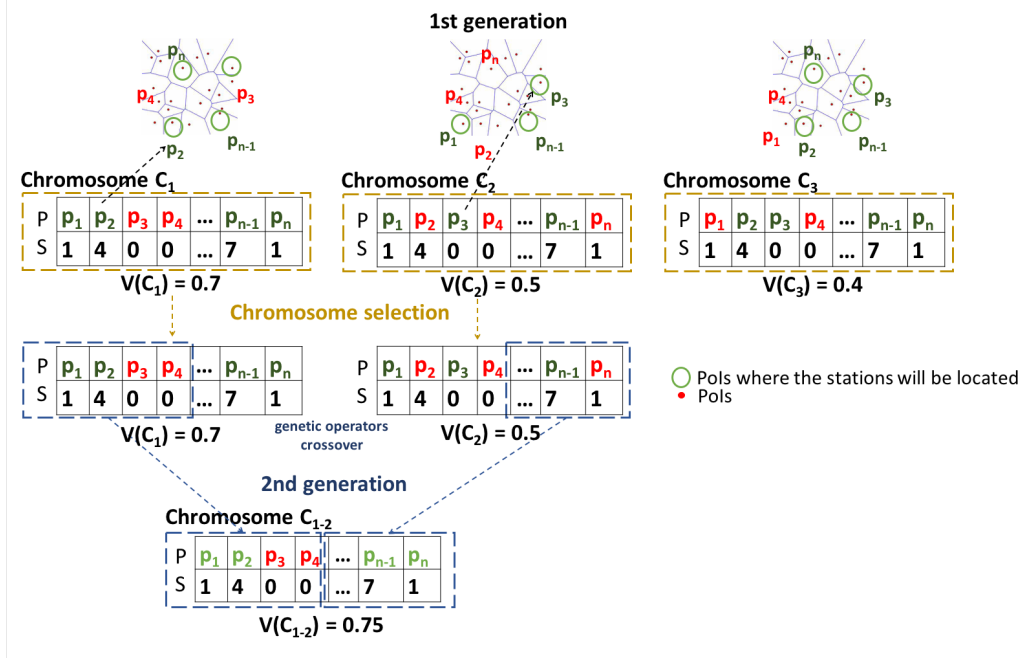


Figure 2: The encoding of individuals (i.e., chromosomes) and genetic algorithm.

in the area of  $p_i$ ;  $a_{social}$  refers to the average social networks activity in the area of  $p_i$ ;  $cost\_area$  refers to the cost of locating stations in  $p_i$  that covers the demand of that area; and  $cost\_per\_charger$  is a constant cost per each charger ( $|s_i|$ ) located in  $p_i$ . The value of these parameters ranges in the interval  $[0,1]$ . Each parameter has associated a weight value  $\omega$  established by the users of the system depending on the characteristics of the city (in our tests, these values are  $\omega_p = 0.4$ ;  $\omega_{tr} = 0.3$ ;  $\omega_t = 0.2$ ; and  $\omega_s = 0.1$ ).

In our implementation, we used the *tournament selection* method, which makes several random groups of individuals, called tournaments, and selects the best one of each group (in our tests, the size of groups is 3). The selected chromosomes are *combined* with others (crossover) and/or *mutated*.

We used a new *crossover* algorithm that considers the geolocated domain of this problem. This crossover will be presented in next section.

We use the uniform integer technique as a *mutation* operator (with a mutation probability of 0.05), which generates a new integer value within a provided range with an independent probability (also fixed to 0.05) of each

attribute to be mutated. At the end of the GA process, the *Emplacement Optimizer Agent* sends the obtained results to the *User Interface Agent*.

### 3.4. A new crossover algorithm for geolocated domains

If we analyze our domain, we find that there is a relationship between each of the characteristics of a chromosome. This is because they are geolocated in a map and they have neighbors, so we can look at the quality of a portion of the solution that includes some characteristics that are close to each other. With this scenario, representing the chromosome as a list of characteristics does not show the relations between those characteristics.

We propose in this work a better representation of the chromosome as a graph, where the nodes are the characteristics of the chromosome and the edges represent that two nodes are close to each other in a map. Thus, if instead of randomly exchanging characteristics of the chromosome we do it by selecting subgraphs of the graph, we will be bringing the semantics of the proximity between nodes to the crossover algorithm. To create the graph we start from the Voronoi tessellation created by our GA and use it to create a Delaunay triangulation. Then the nodes representing the locations on the map are joined by edges to their neighbors. The crossover algorithm randomly selects a node and their neighbors, and these are the nodes that are exchanged between the selected parents, checking that the new generated individual is feasible, i.e., that it satisfies the constraints of the problem. This process can be done with as many subgraphs  $g$  as we want and with a variable depth in the selection of neighbors  $n$ .

Therefore, an individual is the set of vertices  $V$  of a connected graph  $G$  and a set of edges  $E$  that connect the vertices, where  $G = (V, E)$ , for  $v \in V, N(v) = v' \in V | (v, v') \in E$ , which identifies the set of vertices that are connected to  $v$ . Then, a subgraph  $S$  that is going to be exchanged between two parents is (assuming  $g = 1, n = 1$ )  $S = (v_0, v_1, \dots, v_L)$ , where  $v_0$  is selected randomly to be the center of the subgraph and  $L$  is the length of  $N(v_0)$ . Thus,  $N(v_0) = (v_1, \dots, v_L)$ , which represents the neighbors of  $v_0$ . For bigger values of  $n$  this process is repeated for each neighbor of  $N(v_0)$ . For bigger values of  $g$  multiple  $S$  are selected.

Figure 3 shows how a subgraph of the a geolocated graph is exchanged using the graph crossover and how it would be represented in a linear chromosome. The blue node is the randomly selected center of the subgraph and the red nodes are their neighbors at level one. In Section 5 we will present some experimental results about this new crossover.

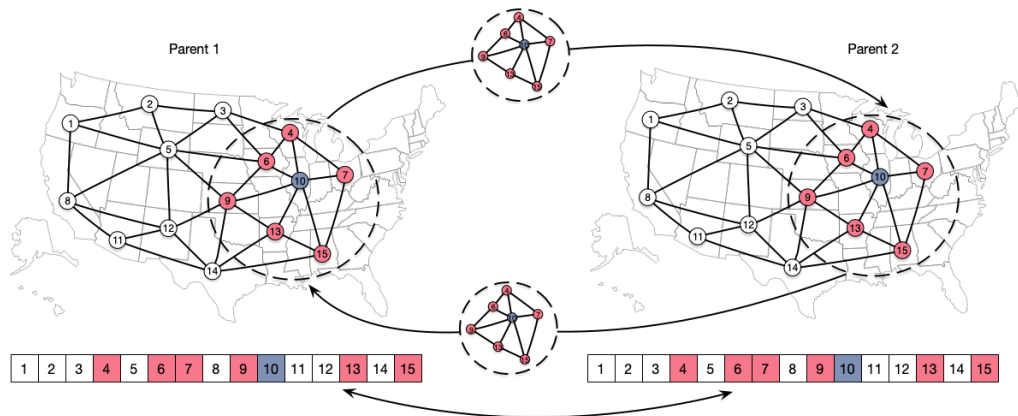


Figure 3: Example of a graph crossover exchanging one subgraph with one neighbor level.

#### 4. Example: City of Valencia

This section presents a case study in the city of Valencia using data from the open data portal supported by the city council<sup>4</sup>. The goal is to determine the most suitable locations in Valencia to locate EV charging stations using the system presented above. Currently, there are 76 charging points in the province of Valencia<sup>5</sup> and 24 of these are located in the city of Valencia.

In the first phase, the MAS determines the  $P$  potential PoIs for the location of a charging station  $s_i$  taking into account data from the General Urban Development Plan. The system determines the area of influence of each of the PoIs creating a Voronoi diagram around the selected points.

After this, in a second phase, the MAS collects data about different aspects from the city of Valencia: (i) Information about the level of traffic in each street of the city. (ii) Information about the population that lives or works in each zone of the city. (iii) The average time spent in commercial and public spaces. This information is obtained from statistics published by the city council and extracted from Google cards. (iv) Information about the geolocated social activity from social networks. This information is obtained using uTool [23], which performs a real-time analysis on the activity of a city through the messages that users exchange inside a social network.

This data is collected and aggregated for each of the polygons around a

<sup>4</sup><http://gobiernoabierto.valencia.es/es/data/>

<sup>5</sup><https://www.electromaps.com/puntos-de-recarga/espana/valencia>

Pol using the agents described in Section 3.1. The *Emplacement Optimizer Agent* receives all this data in order to determine possible solutions.

Finally, in the third phase, the *Emplacement Optimizer Agent* returns solutions through the proposed GA. The best individual in the population is chosen based on the *fitness* value obtained by Equation 1.

The proposed system has been tested with different data and configurations for the particular case study of Valencia in order to compare solutions of different quality. Figure 4 represents two example computed solutions. Figure 4(a) is a solution computed with an initial population of 250 with a fitness value of 0.563. The solution of Figure 4(b) is computed with an initial population of 4000 that yields in a fitness of 0.639.

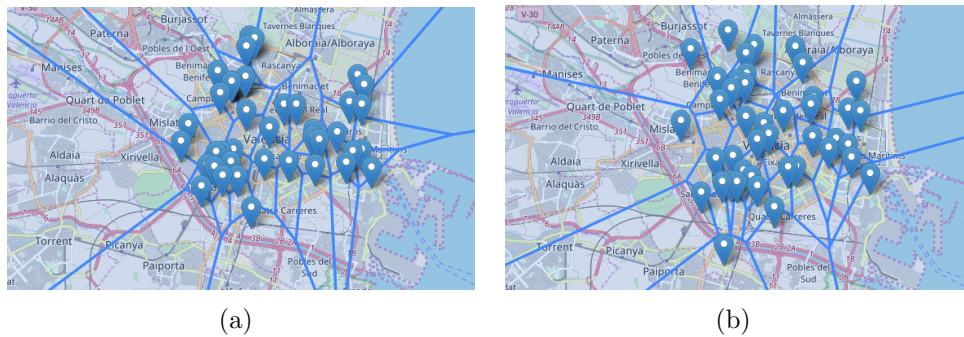


Figure 4: GA solutions for EV charging stations locations in Valencia

At a glance, both solutions are very similar. There is only a difference of 2 stations between the two proposals (solution of Figure 4(a) has 40 charging stations, while Figure 4(b) has 42). However, the quality of a solution is given by the disposition of the stations throughout the city. For instance, Figure 4(a) places a charging station in the far south of the city because there is some activity there. However, this activity is not significant enough and it would be a waste of resources because that area does not need to be covered (in our scenario). The solution of Figure 4(b) places the charging stations more uniformly in the city, covering the full area where the main activity occurs. Concretely, there are several charging stations covering the center and north of the city which are not present in Figure 4(a). Bearing this in mind, the solution proposed in Figure 4(b) is more appropriate if we take into account the characteristics of the city.

## 5. Experimental Evaluation

In these experiments, we analyze the graph crossover explained in Section 3.4 from different points of view. Firstly, we analyze the maximum fitness obtained by the graph crossover with different values of the number of subgraphs  $g$  and the depth of neighbors  $n$ . Then, we show the evolution of maximum fitness of a selection of graph crossover with different values of  $g$  and  $n$  compared to the uniform crossover. Finally, we compare the evolution of maximum fitness of different crossover methods.

For the experiments presented in this section, we use an initial population of 250 individuals, which evolve in the GA through 200 generations. There are 926 PoIs in the city of Valencia after performing the clustering. The initial PoIs of these experiments is significantly higher than those in the previous example. The goal is to locate 100 charging points in the provided PoIs, considering a *max\_chargers\_per\_poi* of 4. With these settings, the search space is significantly large, which justifies the use of a GA.

The weights for the different parameters that characterize the city are  $\omega_p = 0.4$ ;  $\omega_{tr} = 0.3$ ;  $\omega_t = 0.2$ ; and  $\omega_s = 0.1$ ; which represent population, traffic, time spent in a place and social activity, respectively. The weights for the costs *cost\_area* and *cost\_per\_charger* are  $\omega_a = 0.5$  and  $\omega_c = 0.5$ , respectively. Finally, the probability of performing a crossover operation is 0.5, while the probability of performing a mutation operation is 0.05.

The first experiment of the graph crossover consists of a series of tests to evaluate the maximum fitness of the best individual that the GA can achieve using different values of  $g$  and  $n$ . The  $g$  value represents the number of subgraphs (centered on one random PoI) that will be switched between the two individuals to perform the crossover. The value  $n$  is the depth level of neighbors of the selected center PoI to be switched in the crossover.

The combinations of the values  $g$  and  $n$  are restricted to a maximum that depends on the number of genes which are finally switched between the parents. So any combination of  $g$  and  $n$  that provokes all genes (the number of PoIs) or more to be switched is not interesting since genes would be switched more than one time. Therefore, we use an approximation of the maximum values of  $g$  and  $n$  depending on the values of the problem to solve ( $g \cdot avg\_neighbors^n = |PoIs|$ ) where  $g$  and  $n$  are the parameters of the graph crossover corresponding to the number of subgraphs and the depth level of neighbors; *avg\_neighbors* is the average number of neighbors per PoI in the problem to solve, which in our experiments is 5.974; and  $|PoIs|$  is

the total number of PoIs in the problem, which is 926 for these experiments in Valencia. For instance, if we consider  $g = 200$  and  $n = 1$ , the graph crossover will switch approximately  $200 \cdot 5.974^1 = 1493.5$  PoIs (genes), which is significantly higher than the actual number of PoIs in these experiments. Hence, this combination is not interesting and we must decrease  $g$ . Another example with  $g = 1$  and  $n = 5$  implies that  $1 \cdot 5.974^5 = 7608.97$ ; which is more than 8 times the number of PoIs of the problem.

Table 2 shows the maximum fitness achieved by the best individual of an instance of the GA using the graph crossover. Each cell represents the average of 10 different repetitions for the same values of  $g$  and  $n$ . Cells with an **X** symbol represent combinations of  $g$  and  $n$  which are out of the valid values for these experiments as we explained above. Each column represents a number of subgraphs  $g$ , and each row a depth level of neighbors  $n$ .

$n \backslash g$	1	2	5	10	15	20	25	40	50	75	100	125	150	175
1	7.73E-6	3.08E-5	2.68E-5	4.60E-5	4.21E-5	6.08E-5	5.09E-5	<b>1.04E-4</b>	9.85E-5	6.84E-5	7.74E-5	4.97E-5	6.41E-5	5.61E-5
2	1.62E-5	2.62E-5	4.88E-5	3.99E-5	6.31E-5	3.41E-5	6.50E-5	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
3	2.55E-5	4.58E-5	5.51E-5	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
4	2.14E-5	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>

Table 2: Maximum fitness for different values of  $g$  and  $n$  of the graph crossover

The results depicted in Table 2 show that the maximum fitness is achieved by the experiments with 40 subgraphs (in bold), followed by the combinations of 50, 75, and 100 subgraphs. Additionally, the combinations of 15 and 25 subgraphs with 2 depth levels of neighbors  $n$  also achieve a high fitness. Therefore, we can distinguish a pattern which indicates that the results are not good if the gene swaps at the crossover are few, as well as the results are also not good if most of the genes are switched between the parents.

We must note that the values of fitness of all these experiments are significantly lower than the ones shown in the previous example of Section 4 since the setup of the problem is completely different with much more PoIs.

In order to perform a deeper analysis of these results, we select the most significant combinations of  $g$  and  $n$  to show the evolution of the maximum fitness in Figure 5. We also used the results of the uniform crossover as a baseline to compare with the maximum fitness obtained by different parameter combinations of our graph crossover.

As we can see in Figure 5, the uniform crossover depicted with blue  $x$  marks (baseline to compare with graph crossover) tends to stabilize at around generation 125, achieving minor improvements thence onward. However, the graph crossover maximum fitness stabilize later at around 150 or 175



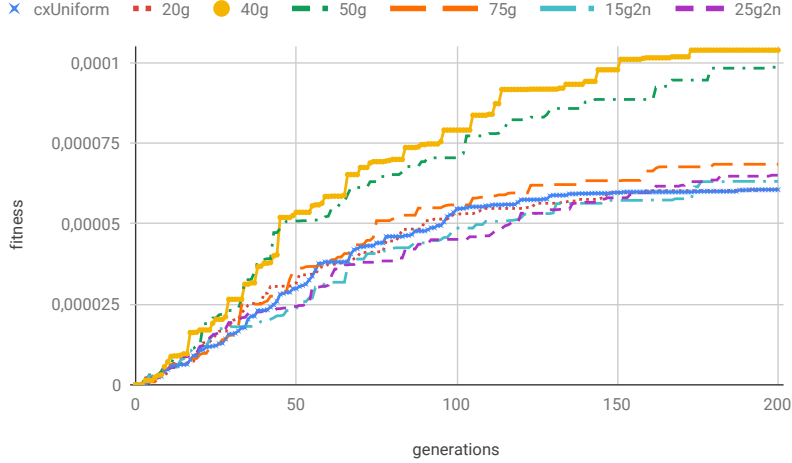


Figure 5: Maximum fitness for baseline crossover and different values of graph crossover

generations. Hence, our graph crossover needs some more generations to reach the top values of fitness compared to the uniform crossover.

Some of the graph crossover values are near the baseline during the evolution, or even below it (see  $20g$ ,  $15g2n$ ,  $25g2n$ , in Figure 5) to finally reach a higher fitness value in the final generations. However, the values reached by these combinations, including the fitness values of  $75g$ , are not significantly higher than the fitness values of the baseline. Finally, the values of maximum fitness achieved by the combinations of  $g = 40$  and  $g = 50$ , as well as the values during the evolution, are significantly higher than the values of the baseline. Concretely, the final value of those combinations are 1.72 and 1.63 times greater than the baseline, respectively.

The last of these experiments is a comparison between four different crossover methods. Figure 6 shows the maximum fitness reached by the best individual for the one-point crossover, two-points crossover, uniform crossover, and graph crossover. As in previous experiments, these results are the average of 10 repetitions for each crossover. In the graph crossover, we plot the combination of  $g = 40$  which achieved the highest fitness.

Figure 6 shows that the one-point crossover stabilizes before arriving to 100 generations, while the two-points crossover stabilizes at 100 generations. Regarding the uniform crossover and the graph crossover, as we mentioned

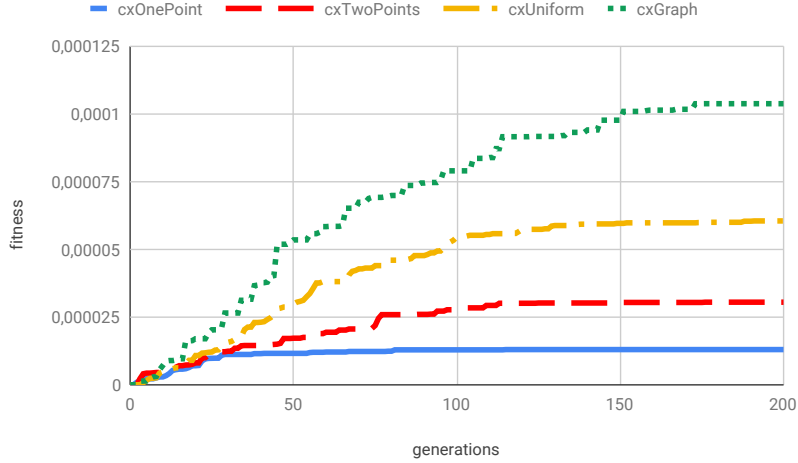


Figure 6: Maximum fitness evolution for alternative crossovers

before, these results stabilize at around 125 and 150, respectively.

Our graph crossover has a significantly higher maximum fitness than the rest of crossover methods as dotted green line shows in Figure 6. The uniform crossover is the next with higher fitness followed by the two-points crossover, being the one-point crossover the one that reaches the lowest fitness. The graph crossover reaches a maximum fitness which is 1.72 times greater than the maximum fitness of the uniform crossover; 3.4 times greater than the fitness of the two-points crossover; and 7.99 times greater than the fitness reached by the one-point crossover. So we can claim that our graph crossover clearly outperforms the other crossover methods in these experiments.

The fact of using a graph structure that considers the neighbor PoIs in the map of the city has proven to be the best technique compared with the standard crossover methods for GAs since it considers context information. Therefore, our graph crossover is the superior approach for the GA applied to the problem of locating EV charging stations, thus confirming our hypothesis.

## 6. Conclusions

A new multi-agent system has been proposed in order to facilitate the analysis of possible locations of EV charging stations. The proposed MAS integrates information from heterogeneous data sources as a starting point to

characterize the areas where charging stations could be located. The core of the system is a genetic algorithm that takes such data as input and generates a proposal of possible locations taking into account several restrictions. The system has been evaluated with real data from the city of Valencia. Additionally, the proposed algorithm incorporates a new crossover method which is specially designed for geolocated domains. Experiments have also showed a good performance of the new proposed crossover method.

As future works, it is considered to define a planning of the installation by phases or years between the different proposed sites. In this way, it would be possible to define a phased installation plan. We also consider an in-depth analysis of the expected energy consumption of the charging stations, especially at times of greatest demand.

**Acknowledgments:** This work was partially supported by MINECO/FEDER RTI2018-095390-B-C31 and MODINVECI project of the Spanish government. Vicent Botti and Jaume Jordán are funded by UPV PAID-06-18 project. Jaume Jordán is funded by grant APOSTD/2018/010 of GVA-FSE.

## References

- [1] P. Wolfram, N. Lutsey, Electric vehicles: Literature review of technology costs and carbon emissions, Tech. rep., The International Council on Clean Transportation (2016).
- [2] F. Millo, L. Rolando, R. Fuso, F. Mallamo, Real co2 emissions benefits and end user's operating costs of a plug-in hybrid electric vehicle, *Applied Energy* 114 (2014) 563 – 571. doi:<https://doi.org/10.1016/j.apenergy.2013.09.014>.
- [3] S. Brown, D. Pyke, P. Steenhof, Electric vehicles: The role and importance of standards in an emerging market, *Energy Policy* 38 (7) (2010) 3797 – 3806. doi:<https://doi.org/10.1016/j.enpol.2010.02.059>.
- [4] E. Wood, J. S. Neubauer, E. Burton, Measuring the benefits of public chargers and improving infrastructure deployments using advanced simulation tools, Tech. rep., SAE Technical Paper (2015).
- [5] J. Dong, C. Liu, Z. Lin, Charging infrastructure planning for promoting battery electric vehicles: An activity-based approach using multiday travel data, *Transportation Research Part C: Emerging Technologies* 38 (2014) 44–55.

- [6] A. Shukla, J. Pekny, V. Venkatasubramanian, An optimization framework for cost effective design of refueling station infrastructure for alternative fuel vehicles, *Computers & Chemical Engineering* 35 (8) (2011) 1431–1438.
- [7] Y. M. Nie, M. Ghamami, A corridor-centric approach to planning electric vehicle charging infrastructure, *Transportation Research Part B: Methodological* 57 (2013) 172–190.
- [8] J. Jordán, J. Palanca, E. del Val, V. Julian, V. Botti, Using genetic algorithms to optimize the location of electric vehicle charging stations, in: *International Joint Conference SOCO'18-CISIS'18-ICEUTE'18*, Springer International Publishing, 2019, pp. 11–20.
- [9] S. Deb, K. Tammi, K. Kalita, P. Mahanta, Review of recent trends in charging infrastructure planning for electric vehicles, *Wiley Interdisciplinary Reviews: Energy and Environment* 7 (6) (2018) e306.
- [10] EV Infrastructure Corridor Development Toolkit, <http://altfueltoolkit.org/ev-infrastructure-corridor-development-toolkit/>, Accessed: 29-10-2019.
- [11] Zap Map, <https://www.zap-map.com/live/>, Accessed: 29-10-2019.
- [12] M. Erbaş, M. Kabak, E. Özceylan, C. Çetinkaya, Optimal siting of electric vehicle charging stations: A gis-based fuzzy multi-criteria decision analysis, *Energy* 163 (2018) 1017–1031.
- [13] H. Lin, C. Bian, H. Li, Q. Sun, R. Wennersten, Optimal siting and sizing of public charging stations in urban area, in: *International Conference on Energy, Ecology and Environment (ICEEE) and International Conference on Electric and Intelligent Vehicles (ICEIV)*, DEStech Publications, INC, 2018, pp. 1–7. doi:10.12783/dteees/iceee2018/27843.
- [14] J. Li, X. Sun, Q. Liu, W. Zheng, H. Liu, J. A. Stankovic, Planning electric vehicle charging stations based on user charging behavior, in: *2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI)*, IEEE, 2018, pp. 225–236.
- [15] Y. Tao, M. Huang, L. Yang, Data-driven optimized layout of battery electric vehicle charging infrastructure, *Energy* 150 (2018) 735–744.

- [16] S. Wagner, M. Götzinger, D. Neumann, Optimal location of charging stations in smart cities: A points of interest based approach, in: 34th International Conference on Information Systems (ICIS), 2013.
- [17] Z. Li, X. Cui, Research on Location Problem of Electric Vehicle Charging Station, *Journal of Applied Science and Engineering Innovation* 2 (12) (2015) 495–498.
- [18] W. Tu, Q. Li, Z. Fang, S.-l. Shaw, B. Zhou, X. Chang, Optimizing the locations of electric taxi charging stations: A spatial–temporal demand coverage approach, *Transportation Research Part C: Emerging Technologies* 65 (2016) 172–189.
- [19] Z. A. Needell, J. McNERney, M. T. Chang, J. E. Trancik, Potential for widespread electrification of personal vehicle travel in the united states, *Nature Energy* 1 (9) (2016) 16112.
- [20] M. Escriva, J. Palanca, G. Aranda, A. Garcia-Fornes, V. Julian, V. Botti, A Jabber-based Multi-Agent System Platform, in: *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS06)*, ACM Press, 2006, pp. 1282–1284.
- [21] P. Saint-Andre, K. Smith, R. Tronçon, R. Troncon, *XMPP: the definitive guide*, " O'Reilly Media, Inc.", 2009.
- [22] D. E. Goldberg, *Genetic Algorithms*, Pearson Education India, 2006.
- [23] E. del Val, J. Palanca, M. Rebollo, U-tool: A urban-toolkit for enhancing city maps through citizens' activity, in: *Advances in Practical Applications of Scalable Multi-agent Systems. The PAAMS Collection*, Springer, 2016, pp. 243–246.