An optimization location scheme for electric charging stations

Conference Paper · June 2013

DOI: 10.1109/SaCoNeT.2013.6654565

2 authors:

Sara Mehar
University of Burgundy

Sidi-Mohammed Senouci
University of Burgundy

An Optimization Location Scheme for Electric Charging Stations

Sara Mehar, Sidi Mohammed Senouci
DRIVE Labs, University of Burgundy, Nevers, France
{Sara.Mehar, Sidi-Mohammed.Senouci}@u-bourgogne.fr

Abstract— ***Due to environmental issues, electric mobility is one of the mobility alternatives that are receiving a huge attention nowadays. In fact, in the last few years electric vehicles have entered the world's car market. This revolutionary technology requires a fast deployment of electric charging stations since the key issue in this system is recharging the batteries. In this work, we propose an optimized algorithm to locate electric-vehicles charging stations. Different factors and limitations are considered and a real case study is given as an application. We first determine the appropriate strict constraints and cost of charging stations' location; and then we propose a mathematical formulation of the problem before solving it using our optimized algorithm named OLoCs (Optimized Location Scheme for electric charging stations). This latter is a heuristic solution; in which we adapt a genetic algorithm to solve the charging stations' location problem. We add a new operator to the classical genetic algorithm to prevent premature convergence and improve the efficiency of the algorithm. OLoCs determines the necessary number of charging stations and their best opening placement. Finally, we evaluate OLoCs performances by analyzing its convergence time and depicting the graphic placement results on a studied map.***

Keywords: smart-grid, placement optimization, charging station, electric vehicle, Dendrogram, genetic algorithm, investment cost, capacity constraint.

I. Introduction

Thanks to advances in technology, new opportunities for sustainable transportation are now possible. Several projects are under research as ridesharing, fleet management, electric vehicles, bio-fuel resources productions… These new alternatives are studied to cope with the transportation sector pressure regarding fossil fuel dependency.

The introduction of alternative vehicle technologies such as Electrical Vehicles (EV) poses questions regarding the impacts of these alternative technologies in people's driving behavior and patterns. In fact, electric mobility brings with it new challenges, such as vehicle recharging and management. It has long been understood that supportive infrastructure must be in place prior to the introduction of a new energy technology. For this reason, contemporary of charging infrastructure is paramount for the imminent organization of large-scale EV systems. So, a lot of efforts are done to install the necessary infrastructure such as charging station, electricity factory and solar panel.

In our work we focus on electric vehicles' charging stations deployment. We consider in our study several constraints (opening cost, distance between charging stations and clients, the city power grid capacity, etc). We first propose a mathematical formulation of the problem. Then, we solve it using an optimized genetic algorithm with an objective of calculating the necessary number of charging stations and the best position to locate them in order to satisfy the clients demand. As the location problem is Np-hard problem, we use heuristic solution instead of an exact solution, which is not effective for a big number of variables and constraints.

Instead of using a big number of clients in our mathematical model, we pre-process the traffic area and group clients (cars) into clusters using a hierarchical clustering algorithm in order to reduce the number of initial variables. After that, we calculate the energy demand for each cluster. In the genetic algorithm, we introduce a new operator "permute" that helps to find the optimal solution and avoid premature convergence, using some intelligent permutation. The pre-processing and the optimized genetic algorithm reduce the execution time and get the optimal solution rapidly. Also, even the demand is not equitable in the area; our optimized genetic algorithm named OLoCs is efficient and gives the optimal solution with the minimum investment cost and transportation costs.

This paper is structured in five Sections: In the first Section, we introduce the most important related work about charging stations location problem. Then, we give the mathematical model of the charging stations location in Section 3. After that, we describe the optimization algorithm OLoCs and the pre-processing phase. In Section 4, we give a case study to evaluate the algorithm performances. Section 5 summarizes and concludes the paper.

II. Related Work

Charging stations placement is a location optimization problem. In this field, many works were proposed and the most known are facilities location and covering location. In facilities location problem, each facility [1][2] has to be placed optimally in order to distribute goods to the clients with a minimum cost. The considered criteria are relative to the client position, the investment cost. In covering location problem, the goal is to find the optimal positions to cover all the clients demand (e.g. road side unit - RSU location in a vehicular network). In [3][4][5], RSU distribution aims to improve connectivity and usually considers intersection as a potential location where information is propagated in all the directions of a crossroad. Generally, the most important parameters are: connectivity delay, coverage, density, quality-of-service QoS, and investment cost. In both problems, the authors propose to solve location problem using heuristic or meta-heuristic algorithms such as: swarm optimization (SO) [3], genetic algorithms (GA) [2], balloon expansion analog heuristic (BEH) [5], Binary Integer Programming (BIP) [5], Mimetic algorithms [6].

For the charging stations placement problem, Shaoyun [7] proposed a method of locating and sizing charging stations for electric vehicles. He divides a Manhattan area into a small identical grid. In each grid, he calculates the traffic sum of all intersection, and then deduces the cost of the transportation power consumed toward the charging station. He estimates the number of initial partitions based on the charging demand of the electric vehicles in the planning area and the maximum and minimum capacity of a charging station. He supposes the cost of power loss equal to the sum of weighted distance from the charging station to the demand nodes in the service area. A genetic algorithm (GA) is used to get the best solution to minimize the loss. In each step of GA, it adjusts the partition depending on demand. At the last iteration, the GA gives the final number of charging stations with the best distribution. However, this solution doesn't take into account the investment constraint. Also, the traffic density of the road is not considered in the formation of the initial partition. This implies that the solution is not global and depends on the chosen scenario.

In [8] the goal is to find the optimum placement of portable charging stations to serve Electric Vehicles in case of peak hours and overload demand or outage. The architecture of the solution is composed of portable energy storage stations and the operation center to manage the distribution of these stations. Each EV sends the battery's state to the operation center periodically or in demand. Based on the collected information, the operation center calculates the optimum point for deploying a single portable charging station that has the maximum total reward (successful travel). The portable charging stations are equipped with sensors to measure their remaining energy level to be replaced when they are out of energy. The exchanged messages between operator center and electric vehicle are transmitted with optional communication protocols, using a peer-to-peer or centralized approach. This method allows EVs to a better load balancing of the power. However, this solution is not well detailed; it suggests using sufficient number of mobile charging stations and then calculating the rate of electric vehicles' travel without draining. It supposes that the best position to place charging stations is the points that maximize the rate of successful travel. However, no method is proposed to get these optimal points.

In [9], an optimization model of charging station planning for electric vehicles is proposed. It combines the global searching ability of the particle Swarm Optimization and weighted Voronoi diagram. The area is partitioned with Voronoi diagram and then the Swarm Optimization is applied to get the best distribution of charging stations.

We notice that most of these works focus on some parameters and ignore others. Usually only the distance between clients and the charging station parameter is considered. In addition, the details of the solutions are not sufficient and no performance evaluations are given.

III. Optimization Problem Modeling

For our problem, we aim to minimize the deployment costs of new charging stations in order to satisfy the customer demands. We take into account many factors and constraints to have a real model that can be used in any area or topology. In our work, we consider the following parameters:

- Area traffic density (helps to deduce the energy Demand),
- Land cost, infrastructure cost, investment cost,
- Transportation cost toward the Charging station,
- Charging station's capacity (Capacity),
- Electric grid capability (Total Capacity).

A. Idea and Modeling

In our work, we need to answer these questions: how many charging stations are needed in a given area? Where can we deploy them? How to assign clients to the charging stations and respect constraints? So, we can model this problem as follows:

Let I={1,…,m} be the set of possible locations to establish a CS, J={ 1,…,n} be the set of customers, $\mathbf{C_{ij}}$ denoting the amount of transportation from charging stations i to customer j, $\mathbf{d_j}$ be the demand of customer j and $\mathbf{a_i}$ be the opening cost of Charging stations i.

Let $\mathbf{Y_i}$ be a decision variable that is not null if the charging stations i is opened and $\mathbf{X_{ij}}$ a binary variable not null if the client j is assigned to charging station i, and $\mathbf{W_i}$ is the i^{th} charging station capacity. These variables are summarized in the following table.

TABLE I. NOTATION

Variable	***Notation***
Investment cost to build charging station i	a_i
i^{th} Charging station capacity	W_i
Binary decision variable of affecting client j to charging station i	Xij
Decision variable to open or not charging station i	yi
Transportation Cost of client j toward charging station i	Cij
Client j demand	dj
Max charging station	m
Client number (Demand cluster number in our case)	n

In order to optimize the placement of charging stations, we must minimize two objectives functions: F1 minimizes the investment cost (2) and F2 minimizes the transportation cost (3). We combine the two functions into a single objective function F:

$$F = \alpha F1 + \beta F2 \qquad (1)$$

Where α and β represent the weight of each function. We choose α=β=1 to give equitable importance to the investment and transportation costs.

$$F1 = \min \sum_{i=1}^{m} a_i y_i \tag{2}$$

$$F2 = \min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{3}$$

The solution feasibility depends on different constraints represented by the following equations: (4) ensures that client j is affected only to one charging station; (5) guarantees that the sum of energy demand dj is smaller than the charging station capacity; finally, (6) and (7) are the integrity constraints.

$$\sum_{i=1}^{m} x_{ij} = 1 \tag{4}$$

$$\sum_{j=1}^{n} d_j x_{ij} \leq W_i y_i \tag{5}$$

$$x_{ij} \in \{0,1\} \tag{6}$$

$$y_i \in \{0,1,\ldots,n\} \tag{7}$$

In the next section, we explain our algorithm .

IV. OLOCS ALGORITHM

In this section, we propose an optimized genetic algorithm called OLoCs (Optimization Scheme for Locating Electric-Vehicles' Charging Stations) to solve the problem of charging stations placement. We use genetic algorithm since the location decision is considered as Np-hard problem. In fact, genetic algorithm is one of the evolutionary algorithms developed to resolve optimization complex problems. It encodes a specific problem on a simple chromosome structure, applies to the population genetic operators (crossover, mutation, recombination) to generate new offspring individuals, and using a fitness function it preserves the best individuals. At the end, it gives the best solution to the problem.

We explain here our algorithm with the different adaptation introduced to resolve charging stations location problem. We give below the suitable genetic code for our problem and we introduce also a new permute operator to the classical genetic algorithm to prevent premature convergence.

A. Genetic code

An important step in genetic algorithms is the individual code (chromosome). As depicted in Fig. 1, the chromosome is divided into two parts of information, the first part is the Charging Station (CS) state and the second part is the Clients' iDentifier (CD).

Figure 1. Chromosome (genetic code)

When the state of CS[i] is null, this means that the CS is closed. Otherwise, CS[i] gives the index of the first assigned client to the CS_i. Each CS_i has many clients; the number of clients is deduced when we subtract the current index value from the smallest index bigger than the current one. For example in Fig.1, CS1= 3, so the first client is in CD3, the next smallest index bigger than the current is CS4=5 so the two clients CD3 and CD4 are assigned to CS1. In the same way CD1 and CD2 are assigned to CS2.

B. Crossover

The cross over is done on a selected part of population. We use the roulette wheel [10] selection method to stochastically select from one generation to create the basis of the next generation. We chose to use the two-point crossover [11].

C. Mutation

This operation is a random change in the population. In our solution, we use a Gaussian mutation [12] that modifies one or more gene values in a chromosome to have a new chromosome value in the pool.

D. Permute

As we use two-point crossover on non-ordered chromosomes, the offspring chromosomes may contain some individual with repeated affections. We add this step to delete the repeated affectations and ensure that each client is affected at most one time to any charging station.

This operator also corrects the non-feasible solution, when the capacity constraints were not respected or when some clients are not affected at all. To do this, we use permutation and change the clients' assignments to ensure that the charging station's capacity is not violated and that the clients are affected to the nearest CS as possible.

E. Recombination

Recombination combines the chromosomes from the initial population and the new offspring chromosomes.

F. Objective function

In order to calculate the cost of each chromosome configuration, we use equation (1), which is equal to (8) by replacing F1 and F2 by equations (2) and (3) respectively. The algorithm keeps the chromosomes with the minimum fitness value.

$$F = \min\left(\sum_{i=1}^{m} a_i y_i + \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \right) \tag{8}$$

After the definition of all operations, we apply the OLoCs algorithm.

G. OLoCs algorithm

The OLocs algorithm is summarized below:

Algorithm 2 *OLoCs algorithm*

< Clusters of demand, Area coordinates,… >

```
Population.Create ("random", initial population)
Population. calculateFitness()
While (!stop condition)
```

```
{ Offspring = Select ("Roulette- wheel",Population, Nb-chromosoms)
  Offspring.Crossover ("2-point" , Pc) ;
  Offspring.Mutation("Gaussian", Pm) ;
  Offspring.Permut () ;
  Recombination("Keep-Best", Offspring, Population)
} Population.BestSolution() <Nb CS, position CS>
```

With P_c (resp. P_m) is the crossover probability (resp. mutation probability).

The fact that the number of clients is very big, this makes the optimization overly complex. Hence, before applying OLoCs algorithm, we pre-process the clients' traffic information and calculate the charging demands. This step is essential to reduce the execution time and to get the best solutions rapidly. We get real time traffic in a peak hour, we subdivide the studied area into n square (e.g. 1Km²), we use Matlab [13] Dendrogram to split data and form n hierarchical clusters, after that we calculate the one hour demands for each cluster using (9) and table II. This demand depends on the rate of electric vehicles, traffic, charging station capacity and daily driving trip [14][7].

$$d_i = \sum_{k=1}^{nb} T_k \times E\_r \times Cs\_Cap \times dur \times D\_r \qquad (9)$$

Where d_i is the demand in cluster i, and nb is the number of sub-zone in the cluster i.

We summarize the area partitioning and energy demand calculation in the following algorithm 2:

Algorithm 2 Area partitioning and *Charging demand calculation*
<Cars traffic, Area coordinates ,... >

1- Get the usual traffic for the considered area
2- Subdivide the area in a small sub-grid
3- Merge the sub-grid into hierarchical clusters
4- Calculate the Energy demand in each Demand cluster (9)

V. EXPERIMENTAL RESULTS

In this section, we expose the results of "Tapas Cologne" town scenario. Here we calculate using OLoCs the best position to locate the charging stations.

The "TAPAS Cologne" simulation scenario describes the traffic within the city of Cologne (Germany) from 6:00 to 8:00 am, which is considered as peak hour. This scenario is generated based on information about traveling habits [15].

Hereafter the parameters used in our simulation and tests.

TABLE II. PARAMETERS TABLE

Parameter	*Value*
Electric vehicles rate (E_r)	6%
Charging demand rate (D_r)	3%
Charging station capacity (CS_Cap)	22Kva
Daily driving	30km
sufficient charging duration (dur) for daily driving	1/4 hours
One hour traffic in sub zone k	T_k

We subdivide the studied area (64Km²) using algorithm 2 in order to reduce the number of variables. Instead 26 000 clients, we use 30 clusters (the number of clusters is variable and depends on the maximum allowed capacity by a cluster). To calculate the number of charging stations, we divide the total charging demand by the minimum charging stations capacity. In this example, we find that the maximum charging station is equal to 16.

In the last step, we use our OLoCs to get the best location, as shown in Fig.2. We can see that 11 stations out of 16 were chosen.

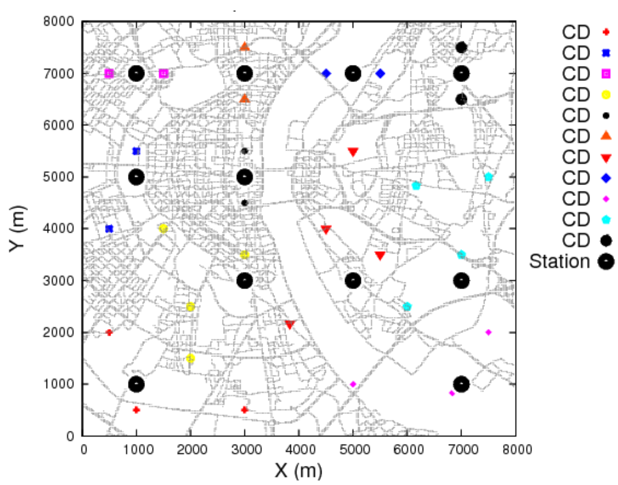

Figure 2. The CS optimized location with aggregated clients (clusters)

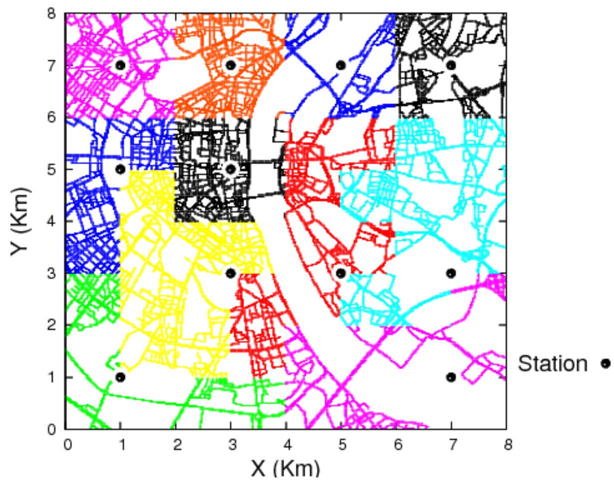

Figure 3. The CS optimized location with real clients (clusters zoom)

Fig.3 shows the final partition where we mark out the clusters assigned to the solution charging station.

All the tests were carried out on Intel i5 64Bits with 4 Go of memory. We summarized in table III the results of 100 simulations for each scenario. Each scenario is represented by 3 rows: Best solution (Bi), Average solution (Ai) and Worst solution (Wi). For example, B1_16_30 is the best solution to plan 16 charging stations for 30 demand clusters. Column Fitness represents the final cost of the solution; Time column is the execution time, NbIter column is the number of necessary iteration and NB_CS_final is the number of charging station to deploy. As no benchmark scenarios are available in the Internet for charging station deployment, we generate different scenarios, where we vary the chromosomes size, the cost constraints' value and charging capacity. We repeated the tests 100 times and in each test we run the algorithm for 4000 to 9000 iterations or less if we get the same fitness after 400 iterations. The size of initial population is 100 chromosomes.

TABLE III. RESULTS

Sol	*NB_CS_final*	*Nb_Iter*	*Fitness*	*Time*
B1_16_30	12	183	4620	0.06
A1_16_30	12	682	5354	0.25
W1_16_30	13	3850	5909	1.27
B2_25_64	16	350	9858	0.24
A2_25_64	16	4240	11720	2.3
W2_25_64	17	8055	13325	4.46

Even if the initial population is completely random, the results show that OLoCs algorithm is able to find the optimal solution after a short time. The variation in the results is due to the fact in our scenario the demand clusters have no equitable load, which results to more computation.

To validate our approach, we compare our approach to an exact solution generated by Cplex [16]. The results, represented in table IV, correspond to a new scenario with 9 charging stations for 15 demand clusters.

TABLE IV. COMPARISON

Approach	*NB_CS_final*	*Fitness*	*Type*	*Time*
OLoCs	7	2732	Optimal	0.01
CPLEX: MIP	7	2732	Optimal	0.03

For this simple scenario, we can notice that the exact solution is slower and we can clearly state that it is expected to worsen with more complex scenarios. These results confirmed also that OLoCs finds the optimal fitness more rapidly than the exact solution, which evaluates all possible solutions.

VI. CONCLUSION AND FUTURE WORK

In this work, we propose an optimized genetic algorithm to place optimally the electric vehicles' charging stations. The pre-process phase creates clusters from traffic data, which helps to reduce the complexity of the problem and saves execution time. The "Permutation operator" introduced to genetic algorithm helps to respect the problem's constraint and to converge to the optimal solution. The result shows OLoCs efficiency in terms of time and optimality.

The optimized placement of electric vehicle charging station is very important for new generation of electric cars. As future work, we want to develop an extension of OLoCs for deploying a new charging station or to use a portable charging station in case of overload of the existing deployed charging station. Also, we plan to study the complexity of our contribution and compare it with the existing work. Furthermore, we develop a fleet management system for electric vehicle to inform drivers about charging station's availability and provide the most economic path for their trips.

REFERENCES

[1] M. Adel El-Baz, "A genetic algorithm for facility layout problems of different manufacturing environments" Computers & Industrial Engineering, Vol. 47 No.2, pp.233-246, 2004.

[2] M. Maric "An efficient Genetic algorithm for solving The multi-Level uncapacitated Facility Location Problem", In Computing and Informatics. Vol. 29, no. 2 , pp 183-201, 2010.

[3] L. Wu , X. Zhang, J. Zhang , "Capacitated facility location problem with general setup cost", Computers & Operations Research 33, pp.1226–1241, 2006.

[4] C. Lochert, B. Scheuermann, C. Wewetzer, A. Luebke and M. Mauve, "Data aggregation and roadside unit placement for a VANET traffic information system", Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking (VANET'08), pp.58-65, 2008.

[5] B. Aslam, F. Amjad and C.C. Zou, "Optimal Roadside Units Placement in Urban Areas for Vehicular Networks", Computers and Communications (ISCC), 2012 IEEE Symposium on ISCC 2012, Cappadocia, Turkey, pp.423-429, 2012.

[6] C. Prodhon, "location-routing problem", Ph.D. dissertation, Troyes university - France, 2006.

[7] S. Ge, "The Planning of Electric Vehicle Charging Station Based on Grid Partition Method" - Electrical and Control Engineering (ICECE), 2011 International Conference on , pp. 2726 - 2730 , 2011.

[8] Z. Li, "Electric Vehicles Network with Nomadic Portable Charging Stations", Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd, Ottawa , Canada, pp. 1-5 , 2010.

[9] X. Tang "Electric Vehicle Charging Station Planning Based on Weighted Voronoi Diagram" Transportation, Mechanical, and Electrical Engineering (TMEE), 2011 International Conference on, Changchun, China , pp.1297-1300, Dec. 2011.

[10] A. Lipowski and D. Lipowska, "Roulette-wheel selection via stochastic acceptance", Physica A: Statistical Mechanics and its Applications, Vol. 391, no. 6, PP. 2193 - 2196, 2012.

[11] Neubauer, A. , "The circular schema theorem for genetic algorithms and two-point crossover", Genetic Algorithms in Engineering Systems: Innovations and Applications, 1997. GALESIA 97. Second International Conference On (Conf. Publ. No. 446), pp. 209-214, 1997.

[12] R. Tinós, "Evolution Strategies with q-Gaussian Mutation for Dynamic Optimization Problems", Neural Networks (SBRN), 2010 Eleventh Brazilian Symposium on , Sao Paulo, Brazil , pp. 223 – 228, 2010.

[13] Matlab, www.mathworks.com

[14] F. Koyonagi, T. Inzuka, and Y. Uriu, "Monte Carlo Simulation on the Demand Impact by Quick Chargers for electric vehicles". Power Engineering Society Summer Meeting, Edmonton, Alta , pp. 1031-1036, 1999.

[15] Sumo http://sumo.sourceforge.net/doc/current/docs/userdoc/Data/Scenarios/TAPASCologne.html

[16] Cplex, http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/