Application Design Aiming to Minimize Drivers' Trip Duration through Intermediate Charging at Public Station Deployed in Smart Cities †

Ibrahim El-Fedany [1,*], Driss Kiouach [1] and Rachid Alaoui [2,3]

[1]   Modeling, Systems and Technologies of Information Team, High School of Technology, Ibn Zohr University, Agadir 80070, Morocco; d.kiouach@uiz.ac.ma

[2]   Research Laboratory in Computer Science and Telecommunications, Faculty of Sciences, Mohammed V University, Rabat 10100, Morocco; rachid.alaoui@uiz.ac.ma

[3]   Multimedia, Signal and Communications Systems TEAM, National Institute of Posts and Telecommunication, Rabat 10100, Morocco

*   Correspondence: fedany82@gmail.com

†   This paper is a modified version of our paper "Framework for optimizing the charging time of electric vehicles in public supply station deployed in smart cities". In Proceedings 2018 IEEE 5th International Congress on Information Science and Technology (CiSt), Marrakesh, Morocco, 21–27 October 2018.

Abstract: Today, smart cities are turning to electric transport, carpooling and zero emission zones. The growing number of electric vehicles on the roads makes it increasingly necessary to have a public charging infrastructure. On the other hand, the main limitations of electric vehicles are the limited range of their batteries and their relatively long charging times. To avoid having problems to recharge, electric vehicle drivers must plan their journeys more thoroughly than others. At the goal of optimizing trip time, drivers need to automate their travel plans based on a smart charging solution, which will require the development of new Vehicle-to-Grid applications that will allow at the charging stations to dynamically interact with the vehicles. In this paper, we propose an architecture based on an algorithm allowing the management of charging plans for electric vehicles traveling on the road to their destination, in order to minimize the duration of the drivers' journey including waiting and charging times. The decision taken by the algorithm based on the exploration of the data of each public supply station according to its location, number of vehicles in the queue, number of charging sockets, and rates of service.

Keywords: algorithm; electric vehicle; queue; public supply station; smart city; trip duration; waiting time

---

1. Introduction

Today, cities occupy 2% of the world's surface; they shelter 50% of the world population, consume 75% of the energy produced and are responsible for 80% of CO_2 emissions. Fuel-consuming is the main source of CO_2 emissions, the city and its inhabitants are the first to be concerned by the dangers of global warming. The success of the transition to a low-carbon society depends largely on what the cities will decide. Their rapid transition to smart cities, which takes steps to reduce emissions of pollutants and irritants, is essential to ensure better management of environmental resources based on the principles of sustainable development and to improve environmental performance of urban areas.

Many cities are implementing pollution reduction projects also through improvements in urban planning and transport, which are the main contributors to CO_2 emissions and other pollutants. Electric vehicles (EVs) are therefore ideal for creating an eco-sustainable urban environment, and thus the number of electric vehicles (EVs) is expected to increase significantly in the near future [1].

Electric vehicles have been seen by several countries around the world as a plausible and appropriate solution that can both solve transport sector problems and protect the environment. The problem facing them is essentially the limitation of battery manufacturing technologies, which makes their driving range too limited and more precisely, it may not exceed 500 km. Moreover, the recharging time is long, ranging from 30 min to 10 h or more, which is considered by the users very long compared to the gasoline vehicle that is capable to refuel in 10 min maximum. On the other hand, the availability and development of fast charging infrastructures [2] can prolong the waiting time of electric vehicles in charging stations (CSs) and influence the acceleration of EVs adoption.

In order to help EV drivers and to contribute to a wide diffusion of VEs, most of the previous work [3–6] presents many categories of applications that divided as following: applications that inform drivers on the way to the next charging station (CS) and battery level, applications assisting drivers to know the status of charging stations, route planning applications, and applications that collect the data from the EVs to purpose of an effective route. A solution proposed by the authors in [7], to place charging stations in a smart city in optimal ways. Thus, it presents a model for EVs, their battery depletion, and the mobility of EVs. In this model, they assume that the charging process only starts when the EVs have a low battery and the communication between the stations and the EVs performed directly using the vehicular ad-hoc network. In [8], the authors propose a model to help EV drivers locate the nearest charging station, they implement an interactive application developed via SQL and PHP platforms to assign EVs their charging locations. In order to make a vision of the smart city via dispersed detection devices, the Internet of Things (IoT), the authors in [9] discusses a data base, big data, based on the history of demand for energy, the location of EVs, and the time of connection. A reservation application based on the parameters set by the EV driver to calculate the optimal path to its destination is implemented in [10]. A framework that shares information, on the real-time, between EVs and charging stations proposed in [11]. EV charging stations can provide the peak/off-peak pricing of chargers, average service times, and customer service index for charging locations. The EV user can select the optimal charging stations based on a weighted combination of the cost of charging and the distance to the charging station.

Some works address the problem on where to charge, primarily, by minimizing the waiting time for EV charging. This cannot be overlooked, as it is the most important feature of a vehicle in future smart city, especially for fast charging. The works in [12] implement charging plans for all EVs based on the minimized queuing time. In [13], the authors proposed a navigation system where the drivers are expected to choose the option that is the least time consuming, where this time includes the driving time, waiting time, and charging time. A communication framework based on the publish/subscribe mechanism to disseminate the necessary information from the CSs' waiting times, to the electric vehicles, to make their individual decision to select the desired CS [14]. Results in [15] show that considering the number of other EVs parking at the CS outperforms that considering the distance to the CS, achieves a shorter charging waiting time particularly given a high EV density.

In [16], optimal online scheduling was implemented by using a price-responsive charging scheme and a solution is proposed in [17] to improve the charge planning of electric vehicles by combining the cost and waiting time in CSs. An Android mobile application developed in [18] that provides a list of CSs in the closest way to EV. Although the app provides an estimate of the distance traveled and a list of available CSs, this a blind search that does not take into account the real-time status of CSs, i.e., the availability of slots. Article [19], based on an estimation of traction power to provide the driver with a recommendation on the most suitable charging station. However, the problem for the static system presented lies in the fact that the driver cannot obtain all the information about the charging station and whether it is working or not and the number of vehicles waiting. Our interest is to determine "where" electric vehicles should move to recharge their batteries while traveling. Indeed, drivers of electric vehicles have their own courses. On one hand, drivers may not be willing to wait long enough to charge their electric vehicles while they are parked at home. In addition, it is also not recommended to

choose a host far from the destination. However, inappropriate charging taking place during journeys may degrade users' quality of experience, as drivers prefer to reach trip destinations as soon as possible.

To facilitate the interaction between charging stations and electric vehicles. Therefore, the electric vehicle drivers find an optimal solution to their charging requirements by reducing trip time to their destination. We proposed and implemented an application design for efficient management of charging requests. This solution minimizes the trip time of electric vehicle drivers by guaranteeing the charging of the EV during the journey.

The rest of this paper is organized as follows. In Section 2, we present the design of an application to optimize trip duration, including charging time in an electric vehicle public supply station (EVPSS). In Section 2, we evaluated the algorithm proposed in this paper with respect to what was proposed in [12]. Finally, Section 4 concludes the document.

2. Application Design

There are 3 types of charging, the normal charge, the accelerated charge and finally, the fast charge. The main difference between the three types is the loading speed, which depends on other parameters. Normal charging takes longer than others to charge an entire battery between 6 and 8 h. It is usually suitable for long periods of breaks such as a night at home or a day's work. The second possible type of charge takes between 1 and 2 h. Indeed, the stops of type car parks, shopping center or for a lunch break are ideal for this type of load. the latter type is the fastest option to charge an electric car at public charging stations. It may be necessary to quickly equip a vehicle with the necessary autonomy to make or continue a journey. This type of charge requires on average 20 to 30 min. So, the waiting time for charging, then the trip time, remains among the big obstacles to a wide use of EVs.

In this work, we implement a communication architecture between the EVs and the EVPSS. This architecture allows the EV driver to obtain, in real time, the information concerning the location of the charging station through which he will spend the minimum trip duration. To achieve this goal, EVs and EVPSS need connected to the internet networks to ensure the transmission of data between them and the server that will store the CSs data and manage the EVs requests Figure 1.
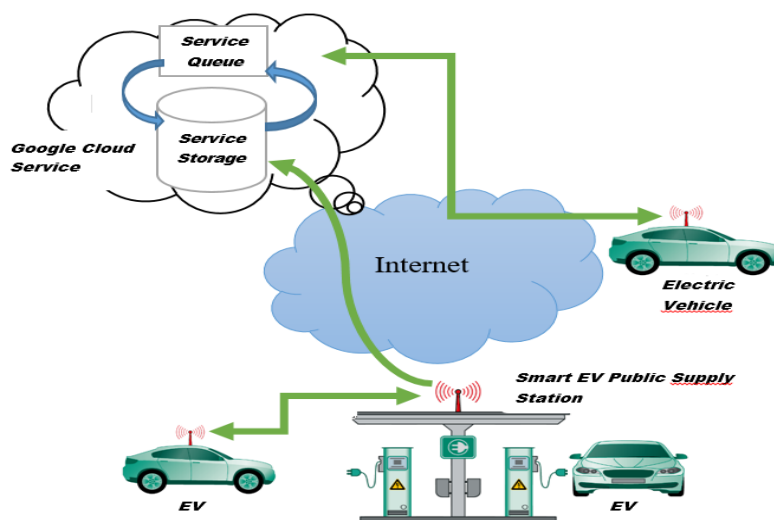
Figure 1. General architecture of smart grid. Reproduced with permission from [12].

In order to reduce design complexity, have a clear and efficient design, and save maintenance time and system evolution, we have implemented the distributed structure based on the design pattern MVC ("model view controller") which is development architecture to separate the source code into modules. Indeed, this widely used model consists of separating data access (databases), the view displayed to the user and the business logic separately. As well as the MVC model, our system has been divided into three layers, one of which is the display layer or view consisting of an application for

EVs and one for EVPSS, the logic layer or controller contains all the services allowing the optimization of charging time, and the model layer manages the status or data of EVPSS Figure 2.
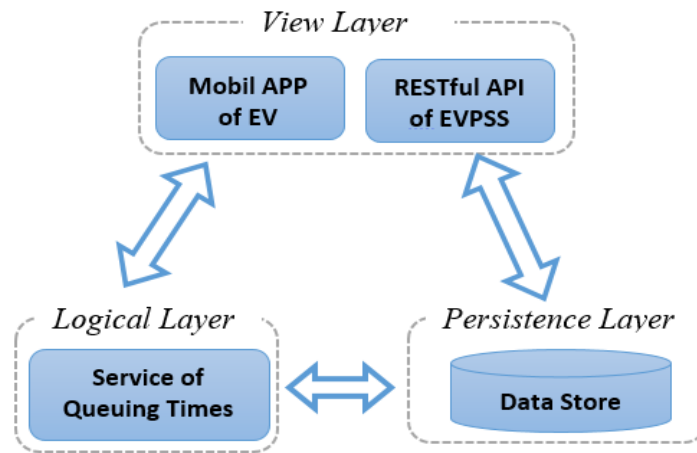
Figure 2. Three-layer framework of application. Reproduced with permission from [12].

The development of the MVC model of our system needs an application protocol allowing web accessibility and exchange of information between EVs, EVPSS, and the server. Thus, the protocol must have non-functional aspects such as the transport of information that is delegated to lower layers, Transmission Control Protocol (TCP) or even HyperText Transfer Protocol (HTTP). For these latter reasons, we have adopted the Representational State Transfer (REST) application protocol which makes it possible to guarantee the interoperability of our system and to extend it by facilitating the integration of new applications. In addition, we have based on the JSON format (JavaScript Object Notation), text format, to exchange the data between the components of our system in an ideal way and independent of any programming language.

2.1. View Layer

The visualization layer represents a communication interface with the system. It consists of a display control and data sending. At each change of EVPSS state, the application server receives the new data. Thus, it responds to the demands of EVs drivers looking for a CS at which trip duration will be minimized. This layer consists of two mobile applications, one for the EVs and the other for the public charging stations.

The mobile application of EVs is an interface designed for their drivers. It allows sending a recharge request. The latter contains the information about the geographical location of the EV and its destination on the map, in order to find the EVPSS by which the EV driver will spend the minimum trip duration.

When the server receives the request from the EV, it uses the queuing service, executes Algorithm 1, to determine the state of each EVPSS and to extract the data from EVPSS in gals that will be send by the server to the EV's mobile application. The latter is responsible for drawing the path to follow on the map using the Google Maps API Figure 3.

The flow management at EVPSS is performed by a mobile application. It allows communicating with the incoming EVs by assigning them the serial number of charging queue. Thus, it sends to the server the new state with each input or output of the EVs to the station Figure 3.
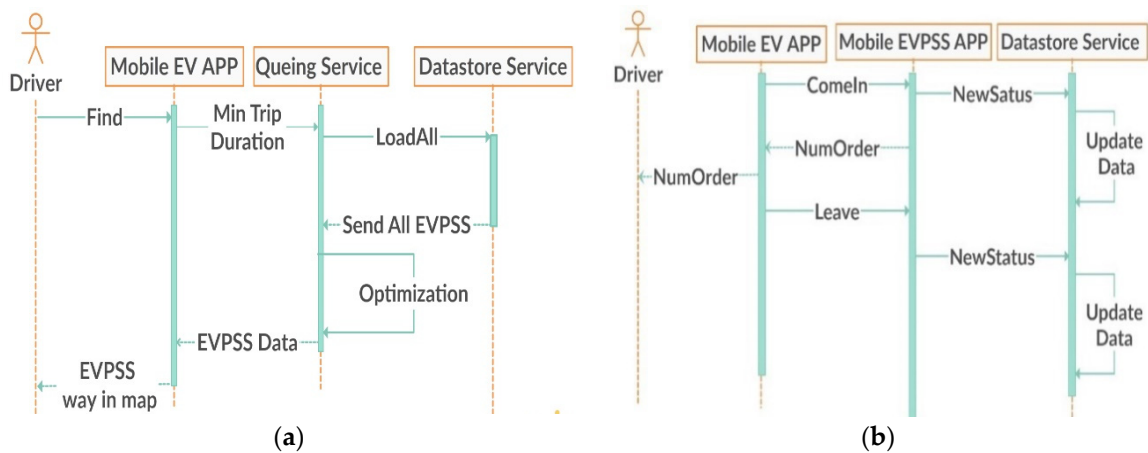
Figure 3. Communication scenario: (**a**) between electric vehicles (EV) and Server; (**b**) between electric vehicle public supply station (EVPSS) and Server. Reproduced with permission from [12].

2.2. Logical Layer

The logical layer or controller is for the essential role of coordinating the sequences of actions/reactions of the proposed system. It receives requests or data sent by EVs and EVPSS, triggers a logic based on business objects to which Algorithm 1 has been deployed, and finally to delegate the path drawing on EV application that constitutes the answer to the request. Thus, when executing Algorithm 1, the controller taking into account the following assumptions:

- EVPASS is continuously connected to the electrical grid, and each has several charging sockets, and such that a number of EVs can be charged in parallel.
- EVPSS are deployed geographically in a city.
- EVs are equipped with Global Positioning System (GPS) [20,21] and wireless communication devices such as 3G to determine their current position and to provide charging request/response services.
- First-come-first-served (FCFS) policy applied in EVPSS. So, if an EV entered at EVPSS where all its charging sockets are currently be used, the EV must wait for one of charging sockets to be free.
- Each EVPSS contains an Ns number of charging sockets, the latter being identical and functioning according to an exponential service rate μ.
- Each EV input into EVPSS sends, input data, the parameters of battery status and when it is output from the station, their battery is fully charged.

2.3. EVPSS Selection Decision Algorithm

The execution of the Algorithm 1 by the server is performed when an EV driver requests an estimate of the trip duration, including the charging time. The result of this execution is to select a CS based on a list (LIST) of the CSs deployed in the smart city.

The criterion selection is based on the minimization of trip duration which consists of the arrival time from EV to CS (T^{arr}), the waiting time, and the charging time of EV in CS (T^{wait}), and the trip time of EV between CS selected and its destination ($T^{cs,\ evd}$). Line 1 indicates the loading of the list (LIST) by all CSs from the database. In line 2, the CS^{min} is initialized by the first CS of the list (LIST) in order to initializing the minimum trip duration, line 3. To obtain the station that meets our objective, the algorithm traverses all the CSs registered in list (LIST), between line 4 and line 17. At each iteration, it extracts the parameters of the following CS in order to calculate T^{trip} of EV through this CS, between line 5 and line 12. In line 15, the CS replaces the CS_{min} if the obtained T^{trip} is less than the $T^{tripMin}$.

Algorithm 1: Decision of Selection the EVPSS

1:　　　Load all active EVPSS to LIST
2:　　　CS_{min} = LIST.getCS(0)
3:　　　$T^{minTrip}$ = init (CS_{min})
4:　　　While (LISTCS.hasNext()) do
5:　　　　　CS = LIST.getCS()
6:　　　N_{ev} = CS.getN()
7:　　　　　N_s = CS.getS()
8:　　　　　μ = CS.getM()
9:　　　T^{wait} = $N_{ev}/(N_s \times \mu)$
10:　　　T^{arr} = distance(ev,cs)/S_{ev}
11:　　　$T^{cs,evd}$ = distance(cs, d_{ev})/S_{ev}
12:　　　T^{Trip} = T^{arr} + T^{wait} + $T^{cs,evd}$
13:　　　　　if ($T^{minTrip} > T^{trip}$)
14:　　　$T^{minTrip} = T^{trip}$
15:　　　CS_{min} = CS
16:　　　　　end if
17:　　　end while
18:　　　send to EV the CS_{min} location

After traversing all the CSs, the algorithm returns the CS to which the EV driver will spend a minimum trip time, including the charging time, line 18. The Table 1 contains the meaning of the variables used in the algorithm.

Table 1. List of nomenclatures.

Abbreviation	Full Meaning
LIST	List of EVPSS loaded from data base
CS_{min}	EVPSS has a minimum waiting time
$T^{minWait}$	Minimum waiting times
N_{ev}	Number of EV at the EVPSS
N_s	Number of charging slot at the EVPSS
μ	Exponential service rate of EVPSS
T^{wait}	waiting times
$T^{cs,evd}$	Minimum travelling time from a CS to EV trip destination
T^{arr}	The EV's arrival time to a CS
$T^{minTrip}$	The minimum time for EV arrives at their destination
S_{ev}	Moving speed of EV

2.4. Data Base Layer

The growing demand for electric vehicles in the future requires a powerful and efficient information storage tool. So, we are relying on the Big Table, which is integrated in Google's Data Store tool to develop the database layer. This tool adopts the concept of entities, which helps us to structure the data sent by EVPSS and EVs via the REST architecture that we have chosen as a communication protocol between the system actors. Moreover, it is a distributed tool on several data centers, very fast in execution of queries, able to handle heavy loads, and very resistant to errors.

3. Performance Evaluation

We used programming with JAVA on Android to develop the architecture of the proposed system and to simulate the communication between the EV, the public charging stations, and the global aggregator (GA) in which we deployed the server of the business layer.

The mobile interface of the EVPSS allows the management of the input and output flows of electric vehicles in a charging station. Figure 4 shows the extraction of the ticket during the entry and the mobile interface that has implemented to allow EV drivers to search for the charging station with the minimum waiting time. We used the Google Maps API to indicate the location of the charging station by specifying the appropriate route.

(**a**)　　　　　　　　　　　　　　　　　　　　(**b**)
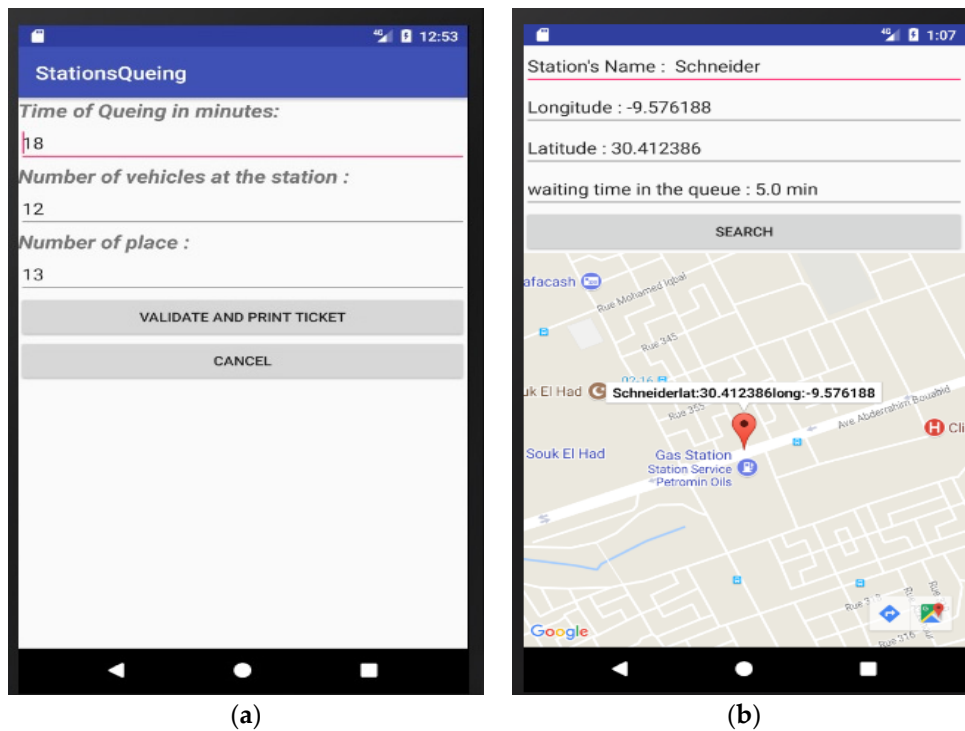
Figure 4. The mobile interface of: (**a**) EVPSS; (**b**) EV. Reproduced with permission from [12].

On the server side, we created a project on the Google App Engine platform to publish our server application.

We implemented a custom simulator in Java to evaluate Algorithm 1 (Alg.1) of the proposed system and compare it with Algorithm 2 (Alg.2) proposed in [12], the latter algorithm allows to determine the charging station where waiting time is minimal without taking into account EV driver destination.

As a case study, we simulated a network of electric vehicle drivers covering an area of $8000 \times 18{,}000$ m^2, which was the center of Agadir in Morocco. As shown in Figure 5, there are ten fast charging stations with the parameters listed in Table 2. In this table, we have listed a number of virtual EVPSS stations, for example Wanma with coordinates of latitude 30.398747 and longitude—9.197610. We assume that the station above contains 10 charging slot and each slot serves three vehicles per hour which represents the parameter μ, service rate.

Figure 5. Simulation Scenario of Agadir City.

Table 2. Parameters of electric vehicle public supply station (EVPSS).

Name of CS	Latitude	Longitude	Slot	μ
Wanma	30.398747	−9.197610	10	3
Bosch	30.43533	−9.641677	10	3
GE	30.360539	−9.526556	10	3
ABB	30.357669	−9.537414	10	3
Hager	30.354614	−9.512201	10	3
Schneider	30.412386	−9.576188	10	3
IES	30.37996	−9.256297	10	3
AddEnergie	30.44643	−9.647556	10	3
DBT	30.390899	−9.514025	10	3
Leviton	30.428318	−9.563399	10	3

We dispose two global aggregator (GA) in the city center in Agadir, the first one (GA1) uses ALs1 and the second one (GA2) based on ALs2, to manage charging requests of EVs.

Algorithm 2: Decision of Selection the EVPSS

1:    Load all EVPSS to LIST
2:    CS_{min} = LIST.getCS(0)
3:    $T^{minWait}$ = init (CS_{min})
4:    While (LISTCS.hasNext()) do
5:        CS = LIST.getCS()
6:    N_{ev} = CS.getN()
7:        N_s = CS.getS()
8:        μ = CS.getM()
9:    T^{wait} = $N_{ev}/(N_s \times μ)$
10:       if ($T^{minWait}$ > T^{wait})
11:   $T^{minWait}$ = T^{wait}
12:   CS_{min} = CS
13:       end if
14:   end while
15:   send to EV the CS_{min} location

In addition, 200 electric vehicles with a maximum speed of movement in cities fixed at 16 m/s are initialized in the network, 1000 electric vehicles are assigned to charging stations in a random way by initiating their queues. Thus, each EV needs charging must send to the server their current coordinates and the coordinates of their destination.

The comparison schemes in this evaluation are:

- The trip duration estimated by Algorithm 2 is equal to the trip duration estimated by Algorithm 1, named A1EA2.
- The trip duration estimated by Algorithm 2 is greater than the trip duration estimated by algorithm 1, named A1GA2
- The trip duration estimated by algorithm 1, named Alg.1.
- The trip duration estimated by Algorithm 2, by adding to the waiting time given by Algorithm 2 the arrival time of the EV at the selected charging station and the duration of the trip between this station and the EV destination, named Alg.2.
- The evaluation measures are as follows:
- Average trip duration: average duration of a VE experiences for his trip, through the recharge service to an intermediate CS.
- *The average number of EVs in percent*: the average number of EVs that checked the patterns A1EA2 and A1GA2.

When the number of charging stations deployed in the smart city is rather small or tends towards a single station, the average number of EVs having the trip duration estimated by Alg.1 is equal to that estimated by Alg.2 Figure 6. Similarly, the average trip duration determined by the two algorithms are almost equal, Figure 7. Therefore, in this case both algorithms achieve a near performance.
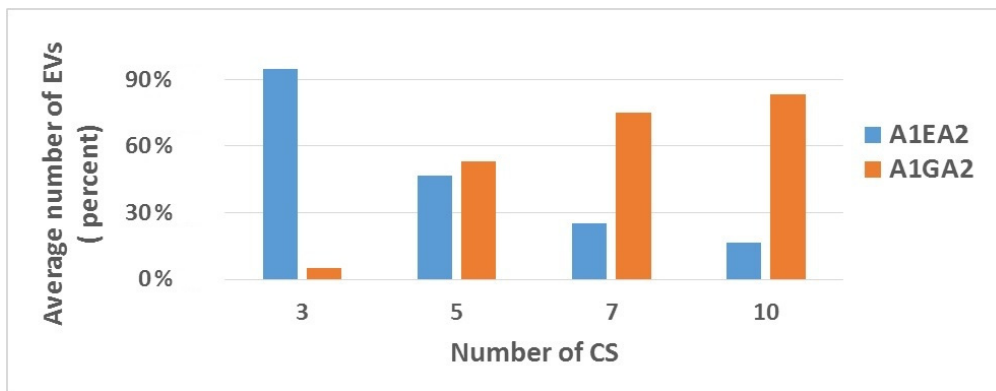
Figure 6. Number of EVs.
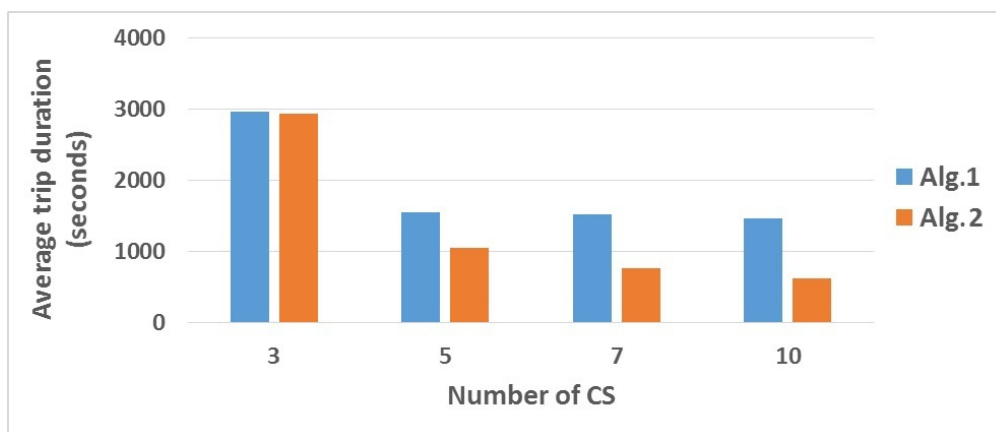
Figure 7. Average trip duration.

On the contrary, as long as the number of stations increases, the average number of EVs with a trip duration estimated by the Alg.1 is lower than that estimated by the Alg.2 increases. Similarly, if the number of stations increases, the average travel time given by the Alg.1 decreases compared to the average estimated by the Alg.2. In general, the Alg.1 is more efficient than the Alg.2 and more precisely when the number of charging station is large enough.

4. Conclusions

In this article we have dealt with the waiting times problem of electric vehicle charge in public charging stations in order to minimize total trip time including waiting time. Our proposed solution is the development of a communication system based on the MVC design pattern concept in order to satisfy the EVs driver recharge request. The system developed is composed of two mobile applications, one for the drivers of the EVs and the other for the EVPSS, and an application installed in a global aggregator, server, that is including the decision algorithm. Furthermore, this paper also contains the communication scenarios between the EVs, the server, and the EVPSS. Thus, the flowchart of the algorithm implemented in the logical layer makes it possible to determine the address of the EVPSS, which the EV drivers spends the minimum trip duration. In order to evaluate the effectiveness of the algorithm proposed in relation to the previous algorithm, we have programmed a simulator in java language in a test scenario realized in Agadir city.

Author Contributions: D.K. and R.A. mainly proposed the idea that's developed in this paper, and they contributed to the system design, the interpretation of the results and the revision of the document. I.E.-F. contributed to the steps of this publication process: data collection, data processing, system design and programming application, simulator programming, statistical analysis, interpretation of results and document writing.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Shuai, W.; Maille, P.; Pelov, A. Charging electric vehicles in the smart city: A survey of economy-driven approaches. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 2089–2106. [CrossRef]
2. Boulanger, A.; Chu, A.; Maxx, S. Vehicle electrification: Status and issues. *Proc IEEE* **2011**, *99*, 1116–1138. [CrossRef]
3. Frank, R.; Castignani, G.; Schmitz, R.; Engel, T. A novel eco-driving application to reduce energy consumption of electric vehicles. In Proceedings of the 2013 International Conference on Connected Vehicles and Expo (ICCVE), Las Vegas, NV, USA, 2–6 December 2013; pp. 283–288.
4. Alizadeh, M.; Wai, H.T.; Scaglione, A.; Goldsmith, A.; Fan, Y.; Javidi, T. Optimized path planning for electric vehicle routing and charging. In Proceedings of the 2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL, USA, 30 September–3 October 2014; pp. 25–32.
5. Ferreira, J.; Monteiro, V.; Afonso, J. Vehicle-to-anything application (V2Anything App) for electric vehicles. *IEEE Trans. Ind. Inform.* **2013**, *10*, 1927–1937. [CrossRef]
6. Hoch, N.; Zemmer, K.; Werther, B.; Siegwart, R.Y. Electric vehicle travel optimization: Customer satisfaction despite resource constraints. In Proceedings of the 2012 IEEE Intelligent Vehicles Symposium, Alcalà de Henares, Spain, 3–7 June 2012; pp. 172–177.
7. Hess, A.; Malandrino, F.; Reinhardt, M.B.; Casetti, C.; Hummel, K.A.; Barcelo-Ordinas, J.M. Optimal deployment of charging stations for electric vehicular networks. In Proceedings of the First Workshop on Urban Networking, Nice, France, 10 December 2012; pp. 1–6.
8. Chokkalingam, B.; Padmanaban, S.; Siano, P.; Krishnamoorthy, R.; Selvaraj, R. Real-Time Forecasting of EV Charging Station Scheduling for Smart Energy Systems. *Energies* **2017**, *10*, 377. [CrossRef]
9. Soares, J.; Borges, N.; Ghazvini, M.A.F.; Vale, Z. Scenario generation for electric vehicles uncertain behavior in a smart city environment. *ELSEVIER Energy* **2016**, *111*, 664–675. [CrossRef]

10. Bedogni, L.; Bononi, L.; Di Felice, M. A Route Planner Service with Recharging Reservation: Electric Itinerary with a Click. *IEEE Intell. Transp. Syst. Mag.* **2016**, *8*, 75–84. [CrossRef]

11. Lokesh, B.T.; Hui Min, J.T. A Framework for Electric Vehicle (EV) Charging in Singapore. *Energy Procedia* **2017**, *143*, 15–20. [CrossRef]

12. El-fedany, I.; Kiouach, D.; Alaoui, R. Framework for optimizing the charging time of electric vehicles in public supply station deployed in smart cities. In Proceedings of the 2018 IEEE 5th International Congress on Information Science and Technology (CiSt), Marrakesh, Morocco, 21–27 October 2018; pp. 537–541.

13. Guo, Q.; Xin, S.; Sun, H.; Li, Z.; Zhang, B. Rapid-Charging Navigation of Electric Vehicles Based on Real-Time Power Systems and Traffic Data. *IEEE Trans. Smart Grid* **2014**, *5*, 1969–1979. [CrossRef]

14. Cao, Y.; Wang, N.; Kamel, G. A Publish/Subscribe Communication Framework For Managing Electric Vehicle Charging. In Proceedings of the 2014 International Conference on Connected Vehicles and Expo (ICCVE), Vienna, Austria, 3–7 November 2014.

15. Gharbaoui, M.; Valcarenghi, L.; Bruno, R.; Martini, B.; Conti, M.; Castoldi, P. An Advanced Smart Management System for Electric Vehicle Recharge. In Proceedings of the 2012 IEEE International Electric Vehicle Conference, Greenville, SC, USA, 4–8 March 2012.

16. Shaobing, Y. Price-responsive early charging control based on data mining for electric vehicle online scheduling. *Electr. Power Syst. Res.* **2019**, *167*, 113–121.

17. Savari, G.F.; Krishnasamy, V.; Sathik, J.; Ali, Z.M.; Aleem, S.H.A. Internet of Things based real-time electric vehicle load forecasting and charging station recommendation. *ISA Trans.* **2019**. [CrossRef] [PubMed]

18. Bedogni, L.; Bononi, L.; D'Elia, A.; Di Felice, M.; Rondelli, S.; Cinotti, T.S. A mobile application to assist electric vehicles' drivers with charging services. In Proceedings of the 2014 Eighth International Conference on Next Generation Mobile Apps, Services and Technologies, Oxford, UK, 10–12 September 2014; pp. 78–83.

19. Sarrafan, K.; Muttaqi, K.M.; Sutanto, D.; Town, G.E. An intelligent driver alerting system for real-time range indicator embedded in electric vehicles. *IEEE Trans. Ind. Appl.* **2017**, *53*, 1751–1760. [CrossRef]

20. Zhou, B.; Chen, Q.; Xiao, P.; Zhao, L. On the spatial error propagation characteristics of cooperative localization in wireless networks. *IEEE Trans. Veh. Technol.* **2017**, *66*, 1647–1658. [CrossRef]

21. Zhou, B.; Chen, Q.; Xiao, P. The error propagation analysis of the received signal strength-based simultaneous localization and tracking in wireless sensor networks. *IEEE Trans. Inf. Theory* **2017**, *63*, 3983–4007. [CrossRef]