

# A Novel Software-defined Network Based Approach for Charging Station Allocation to Plugged-in Electric Vehicles

Raj Mani Shukla\* and Shamik Sengupta†

Department of Computer Science and Engineering, University of Nevada, Reno, USA

Email: \*rshukla@unr.edu, †ssengupta@unr.edu

**Abstract**—This paper proposes a novel approach to integrate plugged-in electric vehicles (PEVs), electric vehicle supply equipments (EVSEs), and smart grid (SG) infrastructure using state of the art software-defined network (SDN) technology, which has a potential to provide unprecedented flexibility to smart grid communication network. We further present set-cardinality based search algorithms for assigning charging station to PEVs to reduce their average charging time. Simulation results show a considerable improvement in the average charging time of PEVs as compared to conventional minimum distance based charging station selection.

## I. INTRODUCTION

The emission of green house gases by automobiles have spurred the demand for the alternative sources of power for their operation. Depletion of non-renewable fossil fuel reserves, rising petroleum prices throughout the world and government regulations to adopt sustainable energy sources have further added the demand for adoption of green energy sources. Plugged-in electric vehicles (PEVs) or plugged-in hybrid electric vehicles (PHEVs) which use electricity either as their primary or alternative source of energy can alleviate the problem of high carbon emission. However, a major limitation of PEVs is that they require a considerably large amount of time for their battery charge as compared to gasoline based refueling. There has been work done in literature on PEV charging which can be categorized into two main parts: 1) Development of charging infrastructure [1] [2] [3] 2). Scheduling of PEV charging [4] [5] [6]. However, the problem of dynamic allocation of charging station to PEVs en-route considering variable charging (or discharging) rates, and wait time of a vehicle at a charging station is still need to be investigated.

Software-defined network (SDN) paradigm offers extensive promises in networking domain because of its simplified control of the network by abstracting application, control and physical layers [7]. Since network applications can be accommodated according to the demand in an SDN framework, we developed a novel SDN architecture for the integration of Plugged-in Electric Vehicles (PEVs), Electric Vehicle Supply Equipments (EVSEs) and Smart Grid (SG). The proposed integrated architecture addresses the aforementioned problems of allocating charging station to moving PEVs to reduce their

average charging time. An SDN controller helps in efficient coordination between different units because of its global view of the network which in this case is spatially located over a large geographical distance. In addition, PEVs selecting charging station can cause imbalance wait time at charging stations such that one PEV may be charged very quickly and another may take a long time to charge at a station. The main contributions of this work are as follows.

- Proposal of a novel SDN architecture for efficient coordination among PEVs, EVSEs, and SG.
- Proposal of set-cardinality based algorithms for charging station allocation to PEVs with the objective of reducing their charging time.

## II. SYSTEM MODEL

Figure 1 depicts the proposed system model. In the context of the proposed model we assumed that the charging rate of a PEV is independent of its battery and depends only on EVSE's charging rate. We also considered that the charging rate of all EVSEs installed in a single charging station is same, but it varies over different charging stations. PEV battery discharge depends upon the distance traveled.

The proposed system consist of moving PEVs on a road which may requests for their battery recharge if their charging status is low. Charging stations are also located along the road. Each charging station contains multiple EVSEs. Electric supply comes from various generation units which include main grid, local grid, and stored energy grid. The street is divided into multiple sections/zones. Each zone contains an SDN controller which is responsible for coordination among PEVs, charging stations, local grids (in the same zone) and main grid.

The smart grid includes main grid, local generation unit, and storage unit. The main grid supplies energy to various zones. Local grids provide energy within a zone, and if this energy is abundant then it is stored in storage units. There can be multiple local grids and storage units in a single zone. Figure 2 represents charging station architecture. A power distribution unit located at a charging station collects electric supply from various units. The collected power, if it is AC, is converted to DC power using AC to DC converter to provide fast DC charging to PEVs. The DC power is supplied to a bus which supplies power to all EVSEs in charging station.

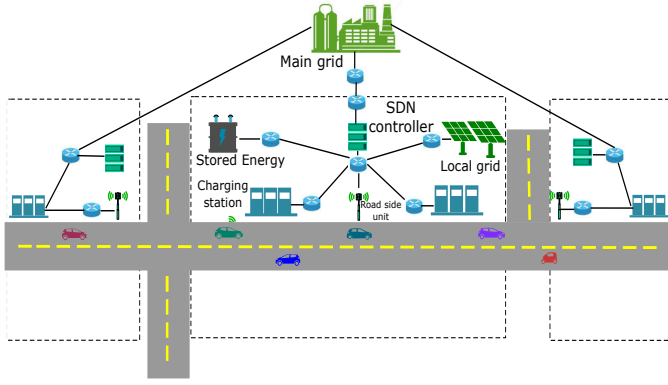


Fig. 1. System model

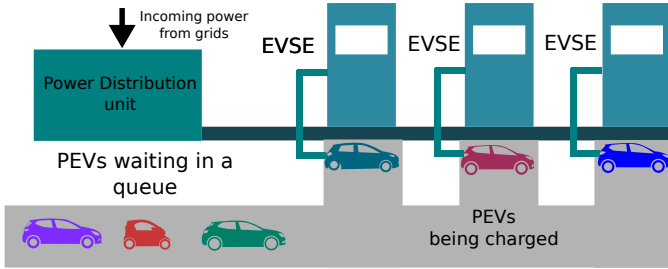


Fig. 2. Charging station model

EVSE establishes a communication link with power distribution unit and PEV to ensure that suitable amount of electricity is supplied to PEV. To track the amount of electricity supplied and spent on charging, the smart meters are placed in power distribution unit and EVSE. At power distribution unit, meter records the amount of power collected from the grid. At EVSE, smart meters track the amount of electricity supplied to a PEV. The PEVs arriving at a charging station wait in a queue. As soon as the EVSE slot is empty, the PEV first in the queue occupies that space for charging.

### III. METHODOLOGY

We describe the optimization unit running on top of the SDN controller to allocate charging station to PEVs. We consider optimization interval  $\tau$  for making optimization decision. The optimization period  $\tau$  is of the order of 15 minutes. Optimization unit receives PEV charging requests in an optimization interval  $\tau_i$ . The requests collected in interval  $\tau_i$  are allocated charging stations at the beginning of the next interval  $\tau_{i+1}$ . The optimization objective is to minimize average *effective charging time* at an EVSE, which we define as the sum of the wait time of PEV in queue and charging time at EVSE (Figure 2). Since PEV charging requests in interval  $\tau_i$  are allocated some charging stations, the wait time of the queue at those charging stations are updated. The updated wait time is used for making optimization decision for requests collected in the next optimization interval  $\tau_{i+1}$ .

1) *Problem statement*: We consider  $N$  vehicles and  $M$  charging stations. Charging station are represented by  $CS_1,$

$CS_2, CS_3 \dots CS_m$ . The location of  $j$ th charging station is given by  $l_j^c$ . An  $i$ th PEV charging request at time  $t$ ,  $R_i^t$ , is represented by  $(t_i, l_i, d_i^r, s_i^{avg}, SOC_i^{curr}, SOC_i^{min}, SOC_i^{req}, r_i^{dc})$ , where  $t_i, l_i, d_i^r$  and  $s_i^{avg}$  are the time at which request is made, location of the vehicle when the request is made, PEV's direction of travel and its average speed. The current, minimum and required status of charge (SOC) and rate of PEV battery discharge per unit of distance are represented by  $SOC_i^{curr}, SOC_i^{min}, SOC_i^{req}$  and  $r_i^{dc}()$ . The threshold distance a PEV can travel based upon its SOC level is given by equation 1.

$$d_{th}^i = \frac{SOC_i^{curr} - SOC_i^{min}}{r_i^{dc}}. \quad (1)$$

For the optimization period  $\tau$ , the time slot  $\tau_n$  at which the PEV reaches  $j$ th charging station is given by equation 2. If the wait time at  $j$ th station at time slot  $\tau_n$  is  $t_w^{j,n}$ , then equation 3 represents the sum of the time to travel and effective charging time of  $i$ th vehicle at  $j$ th charging station, where  $r_{c,j}$  is the charging rate of EVSE at station  $j$ .

$$\tau_n = \left\lfloor \frac{t_i + \frac{||l_j^c - l_i||}{s_i^{avg}}}{\tau} \right\rfloor \quad (2)$$

$$T_{i,j}^c = \frac{||l_j^c - l_i||}{s_i^{avg}} + t_w^{j,n} + (SOC_i^{req} - \frac{SOC_i^{curr}}{r_i^{dc} \times ||l_j^c - l_i^t||}) \times r_{c,j} \quad (3)$$

The objective functions to minimize is given by equations 4.

$$\min\{t_w^{j,n} + (SOC_i^{req} - \frac{SOC_i^{curr}}{r_i^{dc} \times ||l_j^c - l_i^t||}) \times r_{c,j}\} \quad (4)$$

subject to the constrains given by equations 5 and 6.

$$\{l_i \leq l_j^c \leq l_i + d_{th}^r\} \times d_i^r \quad (5)$$

$$\{l_i - d_{th}^l \leq l_j^c \leq l_i\} \times (1 - d_i^r) \quad (6)$$

where,  $d_i^r$  a binary variable representing the direction of travel (1 if we consider highway as an X-axis and PEV is moving from left to right). Equation 5 and 6 specify the constraints regarding the location of charging stations with respect to the location of PEV at the time it has made the charging request.

2) *Set-cardinality based search*: We use algorithms 1 and 2 to find an optimal solution of the problem mentioned above. Algorithm 2 makes use of algorithm 3 for updating charging time after charging station is allocated to a PEV request.

Based on the constraints defined by equations 5 and 6, each vehicle can be charged at a set of potential charging stations to meet its SOC requirements. We consider a set  $M_a^i$ , for PEV charging request  $R_i^t$ , of potentials charging stations where a PEV can be charged. Algorithm 1 finds the set  $M_a^i$  for every charging request  $R_i^t$ . In algorithm 1, steps 2 - 13 are repeated for each charging request  $R_i^t$ . Step 2 finds threshold distance upto which a PEV can travel while maintaining SOC level

---

**Algorithm 1** Charging station set for PEVs

---

**Input:**  $R_a$ - Set of charging requests in an optimization interval

$l_j^c$  - Location of charging stations

**Output:** Set  $M_a^i$  - Potential charging station set for PEVs

```
1: for  $i \leftarrow 1$  to  $\|R_a\|$  do
2:    $d_{th}^i = \frac{SOC_i^{curr} - SOC_i^{min}}{r_i^{dc}(t)}$ 
3:   for  $j \leftarrow 1$  to  $M$  do
4:     if  $d_i^r == 1$  then
5:       if  $l_j^c > l_i$  and  $l_j^c < l_i + d_{th}^i$  then
6:          $M_a^i.append(CS_j)$ 
7:       end if
8:     else
9:       if  $l_j^c < l_i$  and  $l_j^c > l_i - d_{th}^i$  then
10:         $M_a^i.append(CS_j)$ 
11:      end if
12:    end if
13:  end for
14: end for
```

---

above than the minimum required SOC level  $SOC_i^{min}$ . Steps 4-12 are repeated for every charging stations. In step 4, we find the direction of travel of PEV. If the location of the charging station is between the current vehicle location and threshold distance PEV can travel  $d_{th}^i$  (steps 5 and 9), then charging station  $CS_j$  is appended (steps 6 and 10) to the set of potential charging station  $M_a^i$  for the request  $R_i^t$ .

Algorithm 2 presents the charging station allocation to PEVs based on the potential charging stations  $M_a^i$  for the request  $R_i^t$ . We use set  $M_a$  which includes all possible sets  $M_a^1, M_a^2, \dots, M_a^n$  of potential stations such that set  $M_a^i \in M_a$ . Since the PEV charging request with less number of potential charging stations has less options to find a charging station, the set  $M_a$  is sorted (increasing order) according to the cardinality of the set  $M_a^i$  (Step 1).

In steps 2-4, charging stations in set  $M_a^i$  are sorted in increasing order according to their effective charging time. For instance, if there are three charging requests in a scheduling period  $\tau_i$ , with the first request having two potential charging stations, second request having four potential charging stations and the third request having three potential charging stations then after the sorting process the sorted set can be represented as follows:

$\{\{R_2^t : CS_1, CS_5\}, \{R_1^t : CS_1, CS_5, CS_7\}, \{R_3^t : CS_5, CS_7, CS_1, CS_2\}\}$ , such that the effective charging time at charging station  $CS_1$  is less than the effective charging time at charging station  $CS_5$  for request  $R_2^t$ . For request  $R_1^t$ , the effective charging time are in the order  $CS_1, CS_5, CS_7$ , and for the request  $R_3^t$ , the effective charging time are in the order  $CS_5, CS_7, CS_1$  and  $CS_2$ . For the above sorted set, it seems that  $CS_1$  with the lowest effective charging time is a feasible assignment for the request  $R_2^t$ .

However, once  $CS_1$  is assigned to request  $R_2^t$ , the effective charging time in sets  $M_a^i$  for all subsequent charging requests

---

**Algorithm 2** Optimal Charging station allocation to PEVs

---

**Input:**  $M_a$ - Potential charging station set

$l_j^c$  - Location of charging stations

**Output:** Set  $S_a$  - Set of assigned requests

```
1:  $M_a.sort(cardinality)$ 
2: for  $i \leftarrow 1$  to  $\|M_a\|$  do
3:    $M_a^i.sort(chargingTime)$ 
4: end for
5: for  $i \leftarrow 1$  to  $\|M_a\|$  do
6:   if  $M_a^i == NULL$  then
7:     continue
8:   end if
9:    $tmpStation = M_a^i(1)$ 
10:   $k = 1$ 
11:  for  $j \leftarrow k$  to  $\|M_a^i\|$  do
12:     $P = M_a^i$ 
13:     $d1 = \sum_{i+1}^{\|M_a^i\|} t_c^{i,1}$ 
14:     $UpdateEffChargingTime(tmpStation)$ 
15:     $d2 = \sum_{i+1}^{\|M_a^i\|} t_c^{i,1}$ 
16:     $difFirstEle = d2 - d1$ 
17:     $M_a = P$ 
18:    for  $l \leftarrow k + 1$  to  $\|M_a^i\|$  do
19:       $d1' = \sum_{i+1}^{\|M_a^i\|} t_c^{i,1}$ 
20:       $UpdateEffChargingTime(tmpStation)$ 
21:       $d2' = \sum_{i+1}^{\|M_a^i\|} t_c^{i,1}$ 
22:       $difFirstEle' = d2' - d1'$ 
23:       $M_a = P$ 
24:       $FirstEleCh = difFirstEle' - difFirstEle$ 
25:       $difSameSet = t_c^l - t_c^k$ 
26:      if  $difSameSet < FirstEleCh$  then
27:         $tmpStation = M_a^i(l)$ 
28:         $k = l$ 
29:      end if
30:    end for
31:  end for
32:   $N_a^i = tmpStation$ 
33:   $UpdateEffChargingTime(N_a^i)$ 
34:   $S_a.append(N_a^i)$ 
35: end for
```

---

in sorted set  $M_a$  need to be updated and the sets need to be resorted. This is because assigning a request to some charging station may affect effective charging time at charging stations for subsequent requests in the set  $M_a$ .

The updated set, after assigning  $CS_1$  to  $R_2^t$  and re-sorting according to effective charging time, may have following possible representation:

$\{\{R_1^t : CS_5, CS_7, CS_1\}, \{R_3^t : CS_5, CS_7, CS_1, CS_2\}\}$ . For the given possible case, assigning  $CS_1$  to  $R_2^t$  affects the sorted set of potential station for  $R_1^t$ , but the assignment has no effect for  $R_3^t$ . This is because PEV with request  $R_1^t$  reaches  $CS_1$  between the time while PEV with request  $R_2^t$  is being waiting or being charged at allocated charging station  $CS_1$ . On the other hand, PEV with request  $R_3^t$  reaches charging station

$CS_1$  at some different time for the above mentioned case. On the other hand, assigning  $CS_5$ , with higher effective charging time, to  $R_2^t$  may result in following possible sorted set:  $\{\{R_1^t : CS_1, CS_7, CS_5\}, \{R_3^t : CS_7, CS_5, CS_1, CS_2\}\}$  Assigning charging station  $CS_1$  to  $R_2^t$  is a feasible solution if combined effective charging time at  $CS_1$  for  $R_2^t$ ,  $CS_5$  for  $R_1^t$  and  $CS_5$  for  $R_3^t$  is less than the combined effective charging time for assigning  $CS_5$  to  $R_2^t$ ,  $CS_1$  to  $R_1^t$ , and  $CS_5$  to  $R_3^t$ . Otherwise, assigning  $CS_5$  with higher effective charging time to  $R_2^t$  is a feasible solution since it affects the effective charging time of subsequent requests to a lesser extent.

To solve the above assignment problem, we compared the extent to which assigning a charging station to a request affects the effective charging time of subsequent requests. However, rather than an exhaustive search in subsequent requests, which will result the search to grow exponentially in terms of its complexity, we compare how much assigning a charging request affects the first element of subsequent requests because the first element is the element having lowest effective charging time for a given request. After a charging station is assigned to a request, the effective charging time is updated for every charging station in subsequent requests and the sets are resorted according to the updated effective charging time at charging stations.

Steps 6-34 in algorithm 2 are repeated for every charging request.  $\|M_a\|$  represents the number of elements in set  $M_a$ . In step 6 - 8, algorithm finds if the set  $M_a^i$  of potential charging stations contains any charging station. In step 9, we assign the first element of the set  $M_a^i$  as a temporary charging station to the given request. The variable  $k$ , which keeps track of the index of the temporary charging station in set  $M_a^i$ , is assigned an initial value of 1 (step 10). Steps 11- 29 are repeated for every potential charging station for the given request. Variable  $P$  saves the status of set  $M_a$  (Step 12). Step 13 finds the sum of effective charging time of the first element of subsequent requests ( $t_c^{i,j}$  is the effective charging time of  $i_{th}$  request in  $j_{th}$  charging station). In step 14, the effective charging time of subsequent requests is updated with a temporary station allocated to the request according to the algorithm 3. Step 15 finds sum of the effective charging time of the first element of the subsequent request with updated effective charging time. Step 16 determines the increase in effective charging time after assigning temporary charging station. Step 17 reassigns the status of set  $M_a$  with the saved state  $P$ . Steps 19-23 are similar to steps 12 - 16 with the station being checked are those with index higher than  $k$  in set  $M_a^i$ . The difference in charging time of temporarily assigned station with next stations with higher effective charging time in the same set  $M_a^i$  is found in step 25. If the increase in effective charging time by assigning a station with higher effective charging time in set  $M_a^i$  is less than the difference between the extent to which lower and higher effective charging time station assignment have on the first element of subsequent requests, then the value of temporarily assigned station and variable  $k$  are updated (steps 26-28) to the station with higher effective charging time and its position in the set respectively. In step 32, the final

---

### Algorithm 3 Update effective charging time

---

**Input:**  $M_a$ - Potential charging station set

$R_a^t$ - Assigned request

$CS_a$ - Station allocated

**Output:**  $M_a$  - Updated charging time

```

1: for  $i \leftarrow R_a^t.index() + 1$  to  $\|M_a\|$  do
2:   for  $j \leftarrow 1$  to  $\|M_a^i\|$  do
3:     if  $M_a^i(j) = CS_a$  then
4:       if  $t_r^a + t_t^a < t_r^i + t_t^{i,j} < t_r^a + t_t^a + t_w^a + t_c^a$  then
5:          $t_w^{i,j} = t_w^{i,j} + t_r^a + t_t^a + t_w^a + t_c^a - t_r^j - t_t^{i,j}$ 
6:       end if
7:     end if
8:   end for
9: end for
10: for  $i \leftarrow R_a^t.index() + 1$  to  $\|M_a\|$  do
11:    $M_a^i.sort(chargingTime)$ 
12: end for

```

---

temporary station is assigned to the request. After assignment, the effective charging times in set  $M_a$  is updated (Step 33) according to the algorithm 3. The assigned station is appended to the list  $S_a$  in step 34. The final updated effective charging time is used for making optimization decision for the next optimization interval.

Algorithm 3 first updates the effective charging time in set  $M_a$  and then it sorts the updated sets after each PEV charging request is assigned to charging station. The algorithm takes the station assigned to a request  $R_a^t$ . Steps 2-8 are repeated for every subsequent requests in set  $M_a$ . For every subsequent requests, algorithm checks every potential charging stations (Steps 3-7). Step 3 checks whether given charging station matches assigned charging station. If the station matches with the assigned charging station, then in step 4, algorithm checks whether the wait or charge time for the two requests, for the given charging station, overlaps. Wait time or charge time overlaps if the arrival time of the PEV request being checked ( $t_r^i$  and  $t_t^{i,j}$  are the PEV charging request time and travel time of  $i_{th}$  PEV in  $j_{th}$  charging station respectively) is in between the arrival time of the allocated PEV and the time at which charging of allocated PEV's request is complete ( $t_r^a$ ,  $t_t^a$ ,  $t_w^a$ ,  $t_c^a$  are the PEV charging request time, travel time, wait time at station, and charge time of assigned request in station  $j$  respectively) in the given charging station. The difference in time of arrival of PEV at the station being checked and the time at which allocated request completes its charging is added to the effective time of the station being checked (step 5). If in future the particular station is to be assigned for any request then the updated effective charging time is used. Steps 10 - 12 sorts the subsequent requests in the new updated set according to their effective charging time.

## IV. RESULTS

We conducted a set of experiments in a python based framework to determine the improvement in the average effective charging time of PEVs. We also investigated the implemen-

TABLE I  
SIMULATION PARAMETERS

Parameter	Value
SOC (current)	8 - 15 kWh
SOC(minimum)	2 - 5 kWh
SOC(required)	15 - 20 kWh
PEV location	0 - 100 mile
PEV Request time	0 - 15 minute
PEV average speed	50 -60 miles/hour
PEV discharge rate	0.1 - 2.5 kWh/mile
PEV direction of travel	0 - 1
Waiting time at charging station queue	5 - 10 minutes
Charging rate at station	1 - 5 kWh
Charging station location	20 - 120 miles

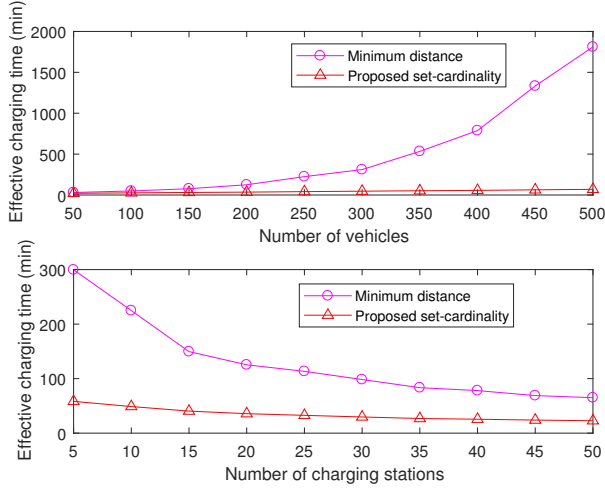


Fig. 3. Average effective charging time as number of vehicles and charging stations are varied

tation feasibility of the proposed algorithms by comparing its run time on various CPU configurations.

1) *Simulation parameters:* Table I presents the simulation parameters used for the experimentation. We used parameters similar to that of PEVs available in the market [8]. For each PEV and charging station, the parameters are randomly selected from the specified range. Optimization period was selected as 15 minutes.

2) *Analysis of charging time:* Figure 3 presents the performance of the proposed algorithm as the number of vehicle requests en-route and charging stations are varied. Figure depicts that the average effective charging time of the proposed set cardinality based algorithm is considerably less than the minimum distance based approach. The average effective charging time due to minimum distance approach increases exponentially as the number of vehicle requests are increased while it increases linearly for the case of the proposed scheme. The effective charging time also varies approximately linearly with the charging stations due to the proposed algorithm. The average effective charging time becomes almost constant as the number of charging stations increases.

3) *Timing analysis:* Figure 4 presents the execution time of the proposed algorithm as the number of vehicles or

charging stations are varied. We run proposed algorithms for two different CPU configurations (8 cores, 3.6 GHz and 4 cores, 2.4 GHz) in ubuntu operating system. Figure depicts that the run time of the algorithm is less than 30 seconds for both configurations. Since optimization period is proposed to be order of 15 minutes, the execution time of the algorithm is negligible as compared to the optimization period. Thus the proposed algorithms are feasible to implement in an SDN controller.

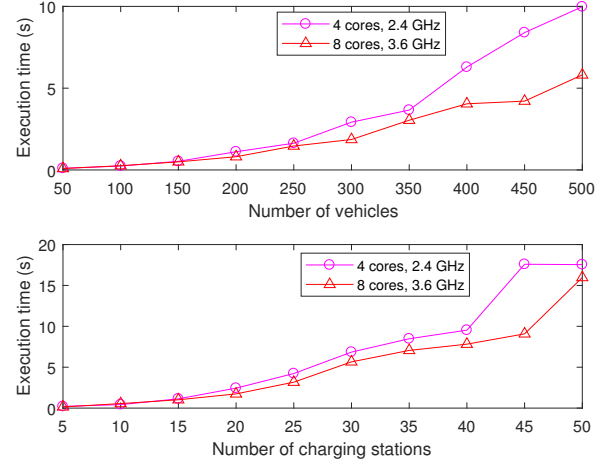


Fig. 4. Execution time of the proposed algorithm as the number of vehicles or charging stations are varied with constant number of charging stations

#### ACKNOWLEDGEMENT

This research is supported by NSF CAREER grant CNS #1346600.

#### REFERENCES

- [1] S. Chen and L. Tong, "iems for large scale charging of electric vehicles: Architecture and optimal online scheduling," in *2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm)*, Nov 2012, pp. 629–634.
- [2] I. S. Bayram, A. Tajer, M. Abdallah, and K. Qaraqe, "Capacity planning frameworks for electric vehicle charging stations with multiclass customers," *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 1934–1943, July 2015.
- [3] H. Zhang, S. Moura, Z. Hu, and Y. Song, "Pev fast-charging station siting and sizing on coupled transportation and power networks," *IEEE Transactions on Smart Grid*, vol. PP, no. 99, pp. 1–1, October 2017.
- [4] N. S. Nafi, K. Ahmed, M. Datta, and M. A. Gregory, "A novel software defined wireless sensor network based grid to vehicle load management system," in *2016 10th International Conference on Signal Processing and Communication Systems (ICSPCS)*, Dec 2016, pp. 1–6.
- [5] H. Lu, G. Pang, and G. Kesidis, "Automated scheduling of deferrable pev/phev load by power-profile unevenness," in *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Oct 2013, pp. 235–240.
- [6] S. Deilami, A. S. Masoum, P. S. Moses, and M. A. S. Masoum, "Real-time coordination of plug-in electric vehicle charging in smart grids to minimize power losses and improve voltage profile," *IEEE Transactions on Smart Grid*, vol. 2, no. 3, pp. 456–467, Sept 2011.
- [7] W. Xia, Y. Wen, C. H. Foh, D. Niyato, and H. Xie, "A survey on software-defined networking," *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 27–51, June 2015.
- [8] Electric vehicle charging. [Online]. Available: www.nfpa.org