ChaseMe: A Heuristic Scheme for Electric Vehicles Mobility Management on Charging Stations in a Smart City Scenario

Neetesh Kumar, *Member, IEEE*, Rashmi Chaudhry, *Member, IEEE*, Omprakash Kaiwartya, *Senior Member, IEEE*, and Neeraj Kumar, *Senior Member, IEEE*

***Abstract*— Towards achieving the goal of green transportation, the usage of battery powered electric vehicles (BEVs) has been continuously growing across the globe. However, considering the limited number of Charging Stations (CSs) in the cities, electric vehicle charging problem has become a challenging task, especially, due to the constraints of longer waiting time and dynamic pricing at the CHs. This issue has led to the degradation in Quality of Experience (QoE) for BEV drivers. Moreover, Charging Point (CP) service providers in the cities also suffer from lack of space which causes higher congestion at the CSs. In this context, we propose ChaseMe, a heuristic scheme for optimizing CS management by scheduling BEVs based on availability and type (fast/ultra-fast) of CPs by considering delay and charging time for CPs reservation. The proposed heuristic scheme consists of two soft computing techniques i) Harris Hawk Optimization (HHO) and ii) Fuzzy Inference System (FIS). Former technique is used to map the CP reservation requests to the best-suited CS by considering Quality of Service (QoS) parameters and acting as a global optimizer. FIS locally manages CPs at a particular CS in coordination with proposed meta-heuristic technique. The experimental results prove the benefits of the proposed ChaseMe framework as compared to the state-of-the-art techniques considering various charging metrics for BEVs.**

***Index Terms*— Battery electric vehicles (BEVs), intelligent transportation system (ITS), optimization, reservation policy, soft computing.**

I. Introduction

BATTERY powered Electric Vehicles (BEVs) have been successfully adopted as commercial, individual and public vehicle fleets across the globe to offer green mobility towards pollution free transportation [1], [2]. BEVs can significantly reduce the carbon based pollutants emitted by traditional vehicles and mitigate the transportation dependency on conventional energy resources such as oil and gas [3]. Generally, BEVs are charged at a Charging Station (CS) connected to an electricity supplying grid network [4]. The multiple Charging Points (CPs) are available at a CS with fast and ultra-fast charging capabilities [5]. As of now, BEVs need frequent charging, particularly, for longer journey due to finite battery capacity constraint. In order to improve market penetration of BEVs, one of the challenging research question that needs to be answered is how to effectively manage the charging processes of BEVs [6] to improve driver's QoE with range of traffic parameters and vehicular communication [7].

In this context, the issue of charging point scheduling has been investigated in [8] for finding the suitable CSs and reserving them for the requested BEVs. Most of the state-of-the-art studies on CS selection are based on traditional client-server centralized system architecture without considering real-time traffic parameters [9]. However, researchers overlooked the optimization of the schedule in case of more number of charging requests. Existing CS selection schemes have attempted to minimize waiting time at CS [10]–[12]. The Minimum Queuing Time (MQT) at local CS is considered as the major prioritization criteria in the CS selection [13]. However, impact of real time traffic conditions including travel distance towards CS, traffic density, average travelling speed were overlooked which result in poor QoE for drivers due to last minute congestion at CSs in case of BEVs requesting a particular CS at the same time [14]. Conclusively, time and efforts spent in search of CSs availability towards travel direction and waiting for the service in a long queue degrade driver's QoE. Therefore, an optimal pre-reservation charging policy can potentially improve BEV driving QoE considering CS Service Provider (SP) centric and vehicular dynamic QoS parameters.

In particular, the limitations of the state-of-the-art approaches on CS selection can be summarized as follows:

- Traditional solutions based on client-server architecture limit the CS accessibility and availability due to absence of cloud or edge supported ubiquitous architecture.
- The real time vehicular dynamics representing QoS parameters have been overlooked, and on the contrary, which rely on localized CS statistics.

Manuscript received January 31, 2021; revised August 13, 2021, September 29, 2021, and December 28, 2021; accepted January 19, 2022. This work was supported by IIT Roorke, India, under FIG Project ChaseMe: A Heuristic Scheme for Electric Vehicles Mobility Management on Charging Stations in a Smart City Scenario under Grant EED/FIG/100892. The Associate Editor for this article was B. Singh. *(Corresponding author: Neetesh Kumar.)*

Neetesh Kumar is with the Department of Computer Science & Engineering, Indian Institute of Technology Roorkee (IIT-R), Roorkee 247667, India (e-mail: neetesh@cs.iitr.ac.in).

Rashmi Chaudhry is with the Department of Computer Engineering, Netaji Subhash University of Technology, Delhi 110078, India (e-mail: rashmi@nsut.ac.in).

Omprakash Kaiwartya is with the School of Science and Technology, Nottingham Trent University, Nottingham NG1 4FQ, U.K. (e-mail: omprakash.kaiwartya@ntu.ac.uk).

Neeraj Kumar is with CSED, Thapar Institute of Engineering and Technology, Patiala, Punjab 147004, India, also with the Department of Computer Science and Information Engineering, Asia University, Taichung 413, Taiwan, also with the King Abdul Aziz University, Jeddah 21589, Saudi Arabia, and also with the School of Computer Science, University of Petroleum and Energy Studies, Dehradun 248007, Uttarakhand (e-mail: neeraj.kumar@thapar.edu).

Digital Object Identifier 10.1109/TITS.2022.3147685

- In CPs scheduling solutions, there are lack of optimization approaches for accounting QoS based load balancing and traffic constraints at CS, vehicular dynamics, heterogeneous infrastructure, and driver's preference.

Towards aiming to address the aforementioned challenges, this paper proposes a CS selection framework *ChaseMe* focusing on optimization of real time traffic centric QoS parameters and cloud or edge based distributed system architecture. The four major contributions of the proposed CS selection framework are as follows:

1) A system model is presented as the two-level charging scheduling optimization problem formulation considering vehicular dynamics, driver as well as service provider centric QoS parameters.
2) An Optimized Reservation Policy (ORP) solution of the problem is developed focusing on optimal mapping the charging reservation requests to the CSs using Harris Hawk Optimization (HHO).
3) A resource efficient fitness function is designed using Fuzzy Inference System (FIS) for the ORP solution enabling it as a second level local optimizer considering light, medium and heavy BEVs requirements of fast/ultra-fast charging points at the charging stations.
4) The performance behaviour of the proposed ORP solution is evaluated through realistic simulation study, and the results are compared with several state-of-the-art approaches to attest its benefits.

Rest of the paper is organized as follows. The literature review is presented in Section II. The problem is formulated in Section III. The proposed *ChaseMe* is detailed in Section IV. The simulation study of the model is discussed in Section V. Finally, paper is concluded in Section VI.

II. Literature Review

Most of the research efforts for CS-reservation policies are based on the centralized systems, and major efforts have been made to optimize the BEVs' charging waiting time [9]. The work in [15] utilized the idea of a centrally connected global controller across CSs in which all EVs send charging requests to obtain the status information. The work in [16] relayed the information such as BEV reservation status or queuing time for charging, scheduling by considering route information for optimizing the performance metrics. In [10], a BEV charging policy was proposed in which high capacity CS accepts charging for on-the-move BEV by advertising its service on a higher frequency, and the BEV with a less battery can frequently sense such services of the CS. Further, a CS selection scheme was proposed to maximize the profit by increasing and decreasing charging rates by accounting the arrival and departure rate of the BEVs at the CS [17]. Architecture considering smart grid communication was also introduced for energy management of BEVs considering battery replacement and traffic congestion on the road [18].

Besides, few reservation-based schemes have been proposed to enhance the reservation intelligence utilizing anticipated BEVs mobility information [19] and city scenario [20]. The work in [6], investigated a technology named as battery switching to improve BEV driver's comfort benefiting from fully recharged switchable batteries at CSs (CSs). Researchers in [21], introduced a framework for communication on-the-move BEV charging scenarios that was based on the subscription to broadcast essential CS information to BEVs to optimize decisions of charging. With advancements in the research, a BEV charging management system [22] was proposed focusing on decision making on CS-selection and to manage BEV's charging plans to decrease driver'S trip duration via midway charging at CSs. In this work [23], a simulation model was described for routing and reservation system of CP based on infrastructure deployed in Ireland. Authors utilized extensive monte-carlo simulations to measure irish population density and to estimate trip length. Few research efforts have also been made on charging cost optimization via minimal peak loads utilizing decentralized control strategies [24], [25] for BEV charging which establish a charging schedule to meet the overnight demand. Further, Furkan *et al.* [26] also presented a model of Battery Swapping Station (BSS) including dynamicity of energy price and EVs load using a forecasting technique. An error analysis was also done with the case study of a fleet of 100 EVs to assess the suitability of the forecasting method. Devendiran *et al.* [27] proposed an intelligent mechanism for electric vehicle scheduling using exponential HHO.

Unlike others, in *ChaseMe*, we identified two level CP reservation requests scheduling problem. Hence, the proposed ORP is composed of two soft computing techniques; HHO [28] is used to map charging point requests to the CSs at the global level, and FIS is used to schedule CP reservation requests at local CS level considering heterogeneous charging demands.

III. System Model and Problem Formulation

The system model includes a vehicular network consisting of Road Side Units (RSUs) equipped with the Multi-access Edge Computing (MEC) servers for the networking connection if public Internet connection is not available. In this scenario, RSU units are responsible for receiving the BEVs charging requests and forwarding them to other RSUs/BEVs/Cloud server. On the other hand, RSUs are also responsible for receiving the Cloud server booked slots from Charging Stations (CS) and send them to the respective BEVs. BEVs are assumed to be distributed using the Homogeneous Poisson Point Process (HPPP) considering the RSUs service radius R.

ORP modelling for BEVs charging includes a group of BEVs with Charging Requests (CRs), CSs comprised of multiple Charging Points (CPs) as the major entities. BEV's charging request maintains a State of Charge (SoC) to reach the destination, and it contains available time window (charging start time, charging time) and priority. Priority of the BEV is computed based on its usage i.e., Emergency (ambulance, fire brigade, patrolling vehicles etc.), personal and warehouse related vehicles have higher, less and lowest

TABLE I

MAIN NOTATIONS AND THEIR DESCRIPTIONS

Notation	Description
BEV	Set of Battery Electrical Vehicles.
CS_i	The $i-th$ charging station.
m	The total number of charging stations.
y	The total number of charging points in the city.
n	The total number of BEVs among all CSs.
$CP_{i,j}$	The $j-th$ charging point at $i-th$ CS.
$p_{i,j}$	The total number of charging points at $i-th$ CS.
$B_{i,j}$	Set of BEVs requesting at $j-th$ CP of $i-th$ CS.
R^f	The charging rate for $fast$ CP.
R^{uf}	The charging rate for $ultra-fast$ CP.
$a_{g,i,j}$	Arrival time of $g-th$ BEV at $j-th$ CP of $i-th$ CS.
p_g	Preferred start time of $g-th$ BEV.
$ft_{g,i,j}$	Finish time of $g-th$ BEV at $j-th$ CP of $i-th$ CS.
$c_{g,i,j}$	Charging duration of $g-th$ BEV at $j-th$ CP of $i-th$ CS.
$R_{i,j}$	Current charging rate at $j-th$ CP of $i-th$ CS.
$\xi_{g,i,j}$	The maximum distance travelled by $g-th$ BEV before running out of battery.
SoC_g	State of Charge of *g-th* BEV.
E_{cg}	Current battery state of $g-th$ BEV.
E_{lb_g}	Lower bound for battery state of $g-th$ BEV.
$\psi_{g,i,j}$	Sum of charging time of each $g-th$ BEV at $j-th$ CP of $i-th$ CS.
$\rho_{i,j}$	Finish charging time of last scheduled BEV at $j-th$ CP of $i-th$ CS. .
$TCT_{g,i,j}$	Total charging time of $g-th$ BEV at $j-th$ CP of $i-th$ CS.
X_h	Position vector of $h-th$ hawk in the population.
s_h	Smallest position vector of $h-th$ hawk w.r.t. X_h.
$Fval_h$	Fitness value of $h-th$ hawk.
ζ	A factor of charging depletion rate for fast charger.
$F^k_{g,i,j}$	if *k-th* BEV precedes *g-th* BEV at *j-th* CP of *i-th* CS.
γ_g	1 if BEV g has achieved the required SoC level.
$\delta_{g,i,j}$	1 if *g-th* BEV can reach *j-th* CP of *i-th* CS otherwise it is 0.
v_g	velocity of *g-th* BEV.
r_1, r_2, r_3, r_4	random numbers.

priorities respectively. While sending the CRs to the ORP, BEVs add preferred start charging time which is proportional to the given priority, and private information is kept separated from other CSs. In order to meet the dynamism of the network, CSs offer CPs with different charging capacities. ORP accounts waiting time, preferred start charging time and charging duration while making its scheduling decisions. Considering smart city CS infrastructure, smart transport infrastructure network includes m CSs, where $CS = \{CS_1, CS_2, \dots CS_m\}$, given CS_i $(1 \leq i \leq m)$ is equipped with p_i $(1 \leq j \leq p_i)$ number of charging points *i.e.*, $CS_i = \{CP_{i,1}, CP_{i,2}, \dots, CP_{i,j}, \dots, CP_{i,p_i}\}$, denotes the set of charging points at i^{th} CS where $y = \sum_{i=1}^{m} p_i$. CSs are ready to serve n BEVs $BEV = \{b_1, b_2, \dots b_n\}$, $n \geq 2$, BEVs seek suitable CS to charge the batteries following residual battery and timing constraints. With this scenario, a graph (vehicular network) $G =< V, E >$ is formed where V represents set CS, and E represents the set of road lanes (edges) connecting CSs. The weight over the edges of the graph is the shortest Euclidean distance (km) between source BEV and destination CS that is computed through Floyd-warshall algorithm. Distance between BEV and any CP is identical to the parent CS. A pictorial representation (consisting physical world and cyber world) for the same is shown in Fig. 1. Further, CSs offer different charging rates

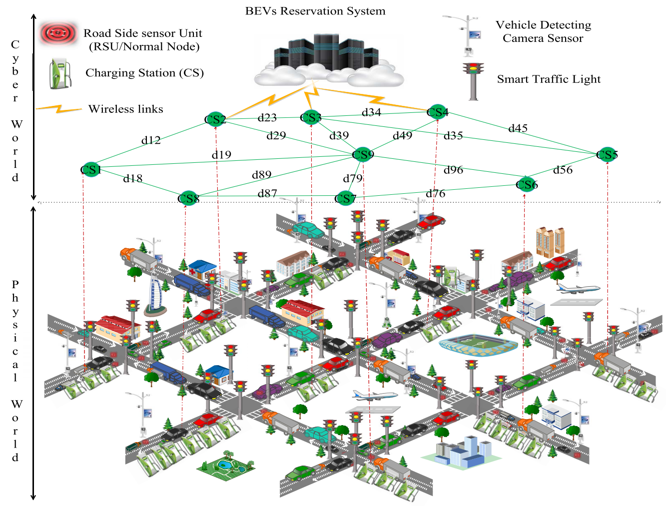


Fig. 1. Cyber-physical world.

$R = \{R^f, R^{uf}\}$, R^f and R^{uf} indicate rate of fast and ultra-fast CPs respectively. It is assumed that all BEVs are equipped with wireless and GPS devices for the real-time communication to make their charging request with the relevant information such as GPS coordinates, electricity consumption rate, residual battery capacity, velocity and priority. The sequence of BEVs at any queue is determined following the timestamp of arrival time at the CS. It is also assumed that each BEV can reach at least one CS with its residual battery capacity. To support this, arrival time and current SoC are used for calculating aggregated finish time by accounting queue waiting time and charging time. After collecting information from BEV and CS, a tuple $< a_{g,i,j}, pt_g, ft_{g,i,j}, c_{g,i,j} >$ is maintained at the Cloud server for g-th BEV $(1 \leq g \leq n)$. The value for the arrival time is dynamically estimated considering the respective charging station, and the different traffic parameters on the road towards the charging station. Here, $a_{g,i,j}$ represents the arrival time of g-th BEV at the j-th CP of the i-th CS, pt_g denotes preferred start time of g-th BEV, $ft_{g,i,j}$ is the finish time of the g-th BEV at the j-th CP of the i-th CS, and $c_{g,i,j}$ represents the charging duration of the g-th BEV at the j-th CP of the i-th CS to reach from current SoC to required SoC (SoC'), $a_{g,i,j} \leq pt_g \leq ft_{g,i,j} - c_{g,i,j}$.

$$c_{g,i,j} = \frac{E_g * (SoC'_g - SoC_g)}{R_{i,j}} \tag{1}$$

where, E_g represents the maximum battery power (Ah) of g-th BEV with lower bound E_{lb_g}, and $R_{i,j}$ is the charging rate at j-th CP of i-th CS i.e., $R_{i,j} \in \{R^f, R^{uf}\}$. The SoC'_g denotes the SoC required by the BEV to reach its respective destination, and SoC is the current SoC which is proportional to current battery (E_{c_g}). The maximum distance travelled by g-th BEV before running out of battery is calculated as,

$$\xi_g = v_g \frac{E_{c_g} - E_{lb_g}}{R_{i,j}} \tag{2}$$

where, v_g represents the velocity of the g-th BEV with $d_{g,i,j} \leq \xi_{g,i,j}$ ($d_{g,i,j}$ is the euclidean distance between g-th BEV and j-th CP of i-th CS).

Lets assume that $B_{i,j}$ is the set of BEVs which are generating charging request at j-th CP of i-th CS, such that, $n = \sum_{i=1}^{m} \sum_{j=1}^{p_i} |B_{i,j}|$. Further, assuming that the g-th BEV arrives at j-th CP of i-th CS, then the arrival time and actual charging time are represented as $a_{g,i,j}$ and $c_{g,i,j}$ respectively, where, the sum of charging time of each BEV at j-th CP of i-th CS i.e., $\psi_{i,j} = \sum_{g \in B_{i,j}} \psi_{g,i,j}$ where $\psi_{g,i,j}$ is calculated as,

$$\psi_{g,i,j} = \begin{cases} max(\rho_{i,j}, a_{g,i,j}) + c_{g,i,j}, & if\ g = 1 \\ max(\psi_{g-1,i,j}, a_{g,i,j}) + c_{g,i,j}, & otherwise \end{cases} \tag{3}$$

where, $\rho_{i,j}$ represents the finish charging time of the last scheduled BEV at j-th CP of i-th CS. Thus, Total Charging Time (TCT) time of the g-th BEV at j-th CP of i-th CS can be obtained as,

$$TCT_{g,i,j} = \sum_{i=1}^{m} \sum_{j=1}^{p_i} (max(\rho_j, a_{g,i,j}) + c_{g,i,j}) F^0_{g,i,j} + \sum_{k \in U_{g,i,j}} (max(\psi_{g,j}, a_{g,j}) + c_{g,j}) F^k_{g,i,j} \tag{4}$$

where, $U_{g,i,j}$ represents the set of BEVs which precedes g-th BEV at j-th CP of i-th CS, $F^0_{g,i,j}$ is 1 if g-th BEV is the first at j-th CP of i-th CS and 0 otherwise, and metric $F^k_{g,i,j}$ has value 1 if k-th BEV precedes g-th BEV at j-th CP of i-th CS otherwise it remains 0.

In view of this modeling, the proposed ORP acts as a two level optimization problem: firstly, problem is to schedule BEVs charging requests to the best suited CS considering BEV dynamics and CS constraints (current load) to optimize the total charging time and balanced BEVs distribution (load) to the CSs. Secondly, the problem is to further assign BEVs to CPs considering heterogeneous BEVs and heterogeneous CPs (at the same CS) scenarios to optimize queue length (charging time) in view of different charging capacities. Thus, the scheduling problem of ORP can be formulated as:

$$Minimize \sum_{g=1}^{n} TCT_{g,i^*,j^*} \tag{5}$$

$$\text{where } j^* = argmin_{j=1 \ldots p_i} TCT_{g,i,j} \tag{6}$$

$$\text{and } i^* = argmin_{i=1 \ldots m} TCT_{g,i,j} \tag{7}$$

$$\text{subject to } \sum_{j \in y} F_{g,i,j} \delta_{g,i,j} = 1 \tag{8}$$

$$d_{g,i,j} \le \xi_g \quad \forall g \in BEV \tag{9}$$

$$\gamma_g = \begin{cases} 0, & if\ SoC_g \le SoC'_g \\ 1, & otherwise \end{cases} \tag{10}$$

where, $\delta_{g,i,j}$ is 1 if g-th BEV can reach j-th CP of i-th CS otherwise it is 0. In other words, Eq. 8 ensures that BEV can reach at-least one CP of any CS, where y is the total number of charging points in the city. Eq. 9 checks that g-th BEV that has enough battery power to reach the selected CP. In Eq. 10, $\gamma_{g,i,j}$ is 1 if g-th BEV has reached the required level of SoC and ready to leave, otherwise it is 0. Thus, it mandates that if a BEV arrives at any CP, it can not leave before the required level of charging *i.e.*, this model is based on no-preemption approach.

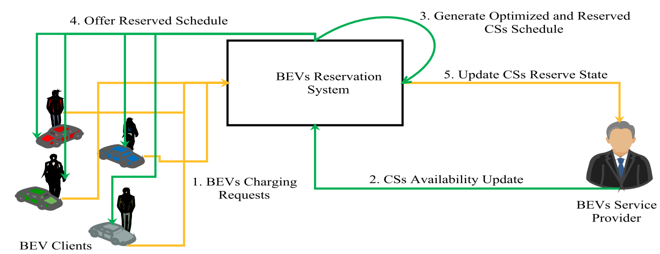


Fig. 2. Context diagram for the *ChaseMe*.

IV. The *ChaseMe* Framework

A. The System Overview

Fig. 1 shows the cyber physical world of BEVs optimized reservation policy. Smart city is considered with the well planned deployment of BEVs oriented CSs. Geographic location of each CS along with its CPs (with their capacity and current loads of BEVs charging requests) is maintained at the cloud server as a graph. Nodes of the graph represent CSs, and edges of the graph represent distance between two CSs. The deployment of the proposed BEVs reservation policy is done at the cloud server. Following this architecture of BEVs ORP deployment, Fig. 2 presents an abstract model of the proposed system. It consists of five steps as follow: **Step1:** BEV client sends its charging request to the cloud server where the proposed ORP maintains a queue of the charging requests. **Step2:** BEV service provider periodically updates the status of CPs about the current load of the respective CPs at the cloud server (as it contains the information of the CSs in the formed graph). **Step3:** Periodically, ORP collects a window of predefined number of charging requests from the queue. After that, it applies HHO meta-heuristic algorithm along with the coordination of FIS to optimize the scheduling of charging requests at the CSs/CPs to minimize TCT accounting other QoS parameters. It returns a BEV charging schedule for each charging request. **Step4:** After step 3, ORP generated schedule is offered to the respective requester. Based on his/her feasibility, CP requester accepts or rejects the schedule. **Step5:** After confirmation (accept/reject) about the charging schedule offer from the BEV charging requester, proposed model updates CSs reservation status.

Detailed functioning modules of the proposed ORP along with three verticals namely BEV clients, optimized reservation policy and BEV service provider are pictorially represented in Fig. 3. This representation determines an ordered logical flow among several modules of the system. The model initiates as soon as client makes a Charging Reservation Request (CRR) following the BEV client side module 1. The same CRR is collected by Request Queue Maintainer (RQM) Module; RQM maintains a queue along with time stamp of each client in First In First Out (FIFO) order. Set of requests are periodically forwarded to the Request Window Extractor (RWE) module (3). RWE extracts a window of the predefined number (that is decided based on frequency of the requests and server capability), all CRRs contained in this window will be scheduled/processed together by the HHO algorithm. Following this window of CRRs, Client Location

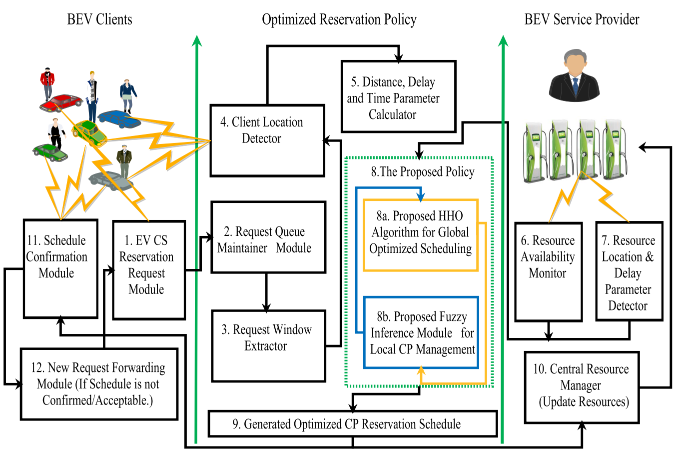


Fig. 3. *ChaseMe* system model.

Detector (CLD) detects GPS locations of all CRRs contained in window. Based on this GPS locations and CRRs inputs, module 5 calculates distance, charging time and travel delay related parameters; these parameters act as the inputs to the proposed ORP for the optimization. In parallel, Resource Availability Monitor (RAM) module activates and periodically sends the updates of the availability of CPs at the respective CS to ORP system (deployed at cloud server). This updated information of the CPs is sent along with the CSs location, CS distance matrix parameters which are calculated using module 7. Data/Information obtained by modules 5, 6 and 7 becomes the input to the optimized ORP as shown in module 8. ORP is composed of two sub-modules: 8a) it implements HHO algorithm to optimize the scheduling of CRRs on CSs at the global level based on input parameters, and 8b) module implements FIS at the local level to allocate best suited CP to CRR (scheduled by HHO to the respective CS) considering fast and ultra-fast requirements and BEV type. These two sub-modules cooperatively optimize the solution following the iterative procedure of HHO. Once, termination criteria of HHO is met then optimized schedule is produced in module 9. The same schedule is forwarded to Schedule Confirmation Module (SCM) (module 10). Based on the received inputs from the clients, Central Resources Manager (CRM) is updated at service provider side. Clients (rejecting schedule) re-submit their CRRs and repeat the same process.

B. The Proposed Charging Point Reservation Policy

The proposed ORP generates an optimized schedule of effective mapping of CRRs on CPs (at CS). To do this, the HHO is encoded to the BEV charging request reservation problem (Section III) along with an effective fitness function to map CRRs to the best suited CPs accounting QoS parameters (distance, delay and time). In general, HHO is competitive and effective nature-inspired optimization technique, it mimics all the stages used in hunting by Harris's hawks to solve the single-objective optimization problems. HHO adopts cooperative behaviour of one of the most intelligent birds, i.e., Harris' Hawks in hunting escaping prey (rabbit) [28].

The major tactic used by Harris hawks to capture a prey is "surprise pounce", that is also referred as "seven kills" strategy. With this intelligent strategy, numerous hawks cooperatively attack from multiple directions and instantaneously converge on an identified escaping rabbit outside the coverage range. This attack is successfully completed by capturing the prey in few seconds. Harris' hawk is able to demonstrate several chasing styles based on the dynamic circumstances and escaping patterns of the prey. Tactic switching takes place when leader hawk (best) stops at the prey and gets lost, and the same chase is continued by one of the flock members. Switching activities are observed in diverse situations as they are useful to confuse the escaping rabbit. Major benefit of such cooperative tactics is that the Harris' hawks can chase detected rabbit to the exhaustion, as it increases the vulnerability. Finally, rabbit cannot escape from confronted team besiege since one of the hawks that is most experienced one and powerful, which easily captures the rabbit and shares it with other flock members.

Following the same procedure, the proposed ORP (Algorithm 1) begins (0^{th} iteration) with the predefined number (H) of randomly generated initial population of hawks; $X_h^w(t)$, $(1 \leq h \leq H)$ is the initial location (position) of h^{th} search agent at t^{th} iteration in w^{th} dimension. If it is not the 0^{th} iteration, in each iteration, Smallest Position Value (SPV) rule (Section IV (D)) is used to the $X_h(t)$ vector and a respective smallest position vector $S_h(t)$ is generated. Next, SPV $S_h(t)$ is evaluated using the proposed fitness function (Section IV(E)), and the respective fitness value $(Fval_h(t))$ is calculated for each hawk. Based on minimum fitness value, X_R and X_{rabVal} are updated. Further, using Eq. 16, the energy and jump strength are updated. Based on the energy, Soft and Hard besiege with or without progressive rapid dives, location vector is updated using Eqs. 17, 19, 23 and 24 as per satisfying assertions. After each iteration, the two termination criterion of HHO is verified to halt the procedure: i) maximum number of iterations, ii) all Hawks convergence to the same point. At the end, X_R is returned which is the final solution.

C. Encoding

A Hawk (search agent) represents a location vector corresponding to BEVs, and the length of the vector is equal to the number of BEVs charging requests (at the instant) made to reserve the CPs. Let, the location of h^{th} search agent in w^{th} dimensional search space is denoted by the location vector i.e., $X_h = \{X_h^1, X_h^2, X_h^3, \ldots.., X_h^w\}$. The minimum and maximum ranges of the search agent for each hawk are defined considering multi-dimensional search space as $[X_{min}$, $X_{max}]$ respectively. The normalized location vector values are generated in the range of [0.0, 4.0]. At 0^{th} iteration for h^{th} search agent, the w^{th} BEV location is generated by Eq. 11.

$$X_h^w(0) = X_{min} + (X_{max} - X_{min}) * l \tag{11}$$

where, l variable signifies uniform random distribution between 0 and 1.

Algorithm 1 ORP

Input : $Network, Area, Nodes$
Output: X_R

begin
- H ← $PosSize$; $X_{rabVal} \leftarrow \infty$; $X_R \leftarrow [\ ]$; t ← 0;
- **while** $\neg$*TerminationCondition* **do**
  - **for** $h = 1$ *To* H **do**
    - **if** $t == 0$ **then**
      - $X_h(t) \leftarrow$ Randomly generated position vector using Eq. 11;
    - **else**
      - $S_h(t) \leftarrow$ SPV($X_h(t)$);
      - $Fval_h(t) \leftarrow Fitness(S_h(t))$;
      - $[PInd, PVal] \leftarrow min(FitVal)$;
      - **if** $X_{rabVal} > PVal$ **then**
        - $X_R \leftarrow X_{PInd}(t))$;
        - $X_{rabVal} \leftarrow PVal$;
      - E_0=2rand()-1, J=2(1-rand())
      - Update the E using Eq. 16
      - **if** $(|E| \geq 1)$ **then**
        - Update the location vector using Eq. 14 Exploration phase
      - **if** $(|E| < 1)$ **then**
        - **if** $(r \geq 0.5\ and |E| \geq 0.5)$ **then**
          - Update the location vector using Eq. 17 Soft besiege
        - **else if** $(r \geq 0.5\ and |E| < 0.5)$ **then**
          - Update the location vector using Eq. 19 Hard besiege
        - **else if** $(r < 0.5\ and |E| \geq 0.5)$ **then**
          - Update the location vector using Eq. 23 Soft besiege with progressive rapid dives
        - **else if** $(r < 0.5\ and |E| < 0.5)$ **then**
          - Update the location vector using Eq. 24 Hard besiege with progressive rapid dives
  - t=t+1;
- **Return** (X_R);

D. Smallest Position Value (SPV)

As discussed, encoding of hawks is in continuous valued vector; however, solution is required in discrete value vector as the charging requests of the vehicles are represented in the sequence of discrete values vector. There is need of a mechanism to convert these continuous values into discrete values vector. To do this, SPV operator is employed. SPV [29] is a heuristic rule which is used to convert the continuous value vector into discrete value vector for all class of sequencing problem. Same is utilized with the HHO which enables the conversion of continuous location value vector X of the hawks into the discrete valued location vectors S. Details of the SPV notion is discussed in [29] which uses a $sort()$ function which arranges the hawk's dimension indexes in an increasing order of position to produce the discrete value vector corresponding to its continuous location vector.

TABLE II

IF/ELSE RULE MAPPING W.R.T. ULTRA-FAST CHARGER

Rules
If ($H.F. \geq M.F.$ and $H.F. \geq \lambda$) Schedule to UF charger; CPtype = 1; elseif ($M.F. \geq H.F.$ and $M.F. \geq \lambda$) Schedule to UF charger; CPtype = 1; elseif ($L.F. \leq H.F.$ and $L.F. \leq M.F.$) Schedule to F charger; CPtype = 0; elseif ($L.F. \geq H.F.$ and $L.F \leq \lambda$) Schedule to F charger; CPtype = 0; elseif ($H.F. \geq L.F.$ and $M.F. \geq L.F.$ and $H.F. \leq \lambda$) Schedule to UF charger; CPtype = 1;

E. Fitness Function

As fitness function is very important component for any meta-heuristic algorithm's performance. The main objective of this work is to minimize the Total Charging Time (TCT) of the schedule (each Hawk) considering fast/ultra-fast CPs at the respective CS. TCT is significantly affected by the type of vehicles and charging rate of the selected CP. Therefore, FIS provides flexibility considering uncertainties. In FIS, absolute false value and absolute truth values are represented with 0.0 and 1.0, and support for intermediate values are also provided with the partial memberships w.r.t. the taken variable. Typical FIS consists of mainly four subsystems; Rule base, Fuzzification, Inference Engine and DeFuzzification. In the proposed FIS, rule base consists of IF-THEN rules pertaining type of BEVs. For the given scheduling problem, rules are tabulated in Table II. Second step is Fuzzification where crisp value is converted into fuzzy value with the assignment of a membership value. For the given scenario, BEV type is taken as ('Heavy','Medium','Light'). At any instant, BEVs arriving at CS are counted according to BEV type where $|H.BEV|$, $|M.BEV|$ and $|L.BEV|$ represents the respective counts for heavy, medium and light vehicles. To calculate the fuzzy values representing respective membership value, H.F., M.F. and L.F. are calculated using following equation.

$$\begin{aligned} H.F. &= \frac{|H.BEV|}{|H.BEV| + |M.BEV| + |L.BEV|} \\ M.F. &= \frac{|M.BEV|}{|H.BEV| + |M.BEV| + |L.BEV|} \\ L.F. &= \frac{|L.BEV|}{|H.BEV| + |M.BEV| + |L.BEV|} \end{aligned} \tag{12}$$

For the Defuzzification, λ cut method is used [30]. Further, to convert fuzzy values into crisp value (in 0 or 1), threshold value (λ) is taken into account, if membership value is greater than λ, true (1) is the output otherwise false (0) becomes the output. The output helps to select the respective CP at the CS, if final result is 1, ultra-fast charger is selected for BEV i, otherwise, fast-charger is selected. After getting its value for

BEV i, charging rate R_j will be calculated following Eq. (13).

$$R_j = \begin{cases} R, & CPtype = 1 \\ \zeta R, & otherwise \end{cases} \tag{13}$$

where R is the charging rate for ultra fast chargers, and ζ represents a factor of charging depletion rate for fast charger and its value lies in [0, 1] interval. Once, R_j is calculated, charging time of BEV for CP j can be calculated using the Eq. 1. After finding out the R_j, BEV is scheduled to j^{th} CP, and then TCT is calculated for the given scenario by following Eq. 5.

F. Location Updating Rules

In the proposed algorithm, position of hawks representing a solution of allocating BEV to various CS is updated by with the exploration phase, transition to exploitation and exploitation phase. These phases follow soft besiege and hard besiege.

1) Exploration Phase: Randomly generated search agents at the respective locations are the candidate solutions which consists of the BEV scheduled to CPs, and the best candidate solution (allocation in each step) is considered as the intended prey or nearly the optimum. Hawks detect a prey based on two strategies considering an equal chance q for each perching strategy, they perch based on positions of other members and the rabbit that is generated in Eq. (11) for the condition of $q < 0.5$, or perch on random tall trees, which is modeled in Eq. (14) for the condition of $q \geq 0.5$. For t^{th} iteration

$$X(t+1) = \begin{cases} X_{rdm}(t) - r_1|X_{rdm}(t) - 2r_2X(t)|, \\ \quad q \geq 0.5 \\ (X_R(t) - X_{avg}(t)) - r_3(LB + r_4(X_{UB} - X_{LB})), \\ \quad q < 0.5 \end{cases} \tag{14}$$

where, *X(t+1)* is the position vector of agents (BEV scheduling in next iteration) used in next iteration $t+1$, X_R denotes the position of rabbit (possible optimum BEV scheduling), *X(t)* represents the current position vector of agents, r_1, r_2, r_3, r_4 and q are the random values in the range (0,1), and X_{LB} and X_{UB} denote the upper and lower bounds of variables, $X_{rdm}(t)$ is a randomly selected agent from current population, and X_{avg} shows average of the position from current population scheduling agents. The average of position of hawks is obtained using Eq. (15):

$$X_{avg}(t) = \frac{1}{H}\sum_{i=1}^{H} X_i(t) \tag{15}$$

where, $X_i(t)$ indicates the location of i^{th} hawk in iteration t, and H represents the number of hawks.

2) Transition From Exploration to Exploitation: ORP transitions from exploration to exploitation to change the exploitative behaviours based on the prey's escaping energy. Therefore, the energy of a prey is modelled as:

$$E = 2E_0\left(1 - \frac{t}{T}\right) \tag{16}$$

where, E denotes escaping energy of the prey, T represents maximum number of iterations, and E_0 is the initial state of its energy which randomly changes between (−1, 1). When, the value of E_0 decreases from 0 to −1, it denotes that rabbit is physically flagging, and the value of E_0 increases (0 to 1), it indicates that rabbit's position is strengthening. This behavior helps to keep the variation in various generated BEV scheduling solutions so that complete spectrum of possible solutions is covered.

3) Exploitation Phase: Similar to the escaping behaviours of the prey and chasing tactics of Harris' hawks in HHO, four strategies are proposed in ORP to model the solution finding stage. If r is the probability of a prey in successfully escaping ($r < 0.5$) and not successfully escaping ($r \geq 0.5$) before surprising the pounce. Based on this, hawks (solution agents) perform hard or soft besiege in order to catch the prey.

a) Soft besiege: If $r \geq 0.5$ and $|E| \geq 0.5$ (pray has good energy and try to escape), the Harris' hawks encircle it softly to exhaust the rabbit and perform the surprise pounce. This behaviour for the same is modelled as:

$$X(t+1) = \Delta X(t) - E|JX_R(t) - X(t)| \tag{17}$$

where,

$$\Delta X(t) = X_R(t) - X(t) \tag{18}$$

where, $J = 2(1 - r_5)$ denotes random jump strength of rabbit throughout the escaping process, and it changes randomly (in each iteration) to simulate the behaviour of the rabbit motions to keep the variation in the solutions.

b) Hard besiege: If $r < 0.5$ and $|E| \geq 0.5$ (prey is exhausted and has low escaping energy), Harris' hawks hardly encircle the prey to perform the surprise pounce, and the current positions are updated using Eq. 19:

$$X(t+1) = X_R(t) - E|\Delta X(t)| \tag{19}$$

c) Soft besiege with progressive rapid dives: If $r \geq 0.5$ and $|E| < 0.5$, it denotes rabbit has good amount of energy to escape and still a soft besiege, this is more intelligent procedure than previous case. It is assumed that hawks evaluate (decide) their next move using the following rule in Eq. 20, they dive based on LF-based patterns using rule formulated in Eq. 21.

$$Y = X_R(t) - E|JX_R(t) - X(t)| \tag{20}$$

$$Z = X_R(t) - E|JX_R(t) - X(t)| + S \times LF(W) \tag{21}$$

where, W and S denote the dimension of the problem and random vector by size $1 \times W$ respectively, and LF is levy flight function as calculated using Eq. 21 [49]:

$$LF(w) = 0.01 \times \frac{u \times \alpha}{|vc|^{\frac{1}{\beta}}} \tag{22}$$

where, *u, vc* are random values between [0, 1], β denotes default constant and α variable is calculated same as in [28]. Therefore, the final strategy to update positions of the hawks can be done by Eq. 22.

$$X(t+1) = \begin{cases} Y, & if F(Y) < F(X(t)) \\ Z, & if F(Z) < F(X(t)) \end{cases} \tag{23}$$

d) Hard besiege with progressive rapid dives: If $r < 0.5$ and $|E| < 0.5$, rabbit does not have enough energy to escape, thus, a hard besiege constructed before surprise pounce to capture and kill the prey. Hawks try to decrease distance of their average location from the escaping prey, and the Eq. 24 is used to update the location in this phase:

$$X(t+1) = \begin{cases} Y, & if\, F(Y) < F(X(t)) \\ Z, & if\, F(Z) < F(X(t)) \end{cases} \tag{24}$$

where, Y or Z denotes the next location for the new iteration.

V. Simulation Study

This section provides the performance analysis of the proposed HHO meta-heuristic based ORP using simulation prepared in MATLAB tool. The objective of the ORP is to find an optimal schedule of the BEV CRRs to the CSs (comprised of fast and ultra-fast CPs) such that charging requests are fairly distributed. To analyze the effectiveness of the proposed ORP, various important performance metrics are identified as follows. Total Charging Time (TCT), Average Relative Imbalance (ARI), Average CRs served per CS and Average CS Utilization (ACSU). These QoS performance metrics are formulated as follow:

TCT can be calculated using Eq. 4.

A. Average CRs Served per CS

This metric represents the number of CRs served by the per CS per unit of time, formulated as,

$$AverageCRsServedperCS = \frac{\sum_{i=1}^{|CS|} TCT_i^{CS}}{|CR|} \tag{25}$$

where, TCT_i^{CS} represents the total charging time of i^{th} CS which consist of the summation of TCT of all the CRs served at that CS given as follows,

$$TCT_i^{CS} = \sum_{CR_j \in CS_i} TCT_j \tag{26}$$

B. Average CS Utilization (ACSU)

This metric indicates the efficient utilization of CS. The proposed methodology distributes the heavy BEVs to ultra-fast CS and other BEVs to fast CS. This metric depends on the obtained total charging time of BEVs in the network scenarios.

$$ACSU = \frac{\frac{\sum_{i=1}^{|CS|} TCT_i^{CS}}{MaxTime}}{|CS|} \tag{27}$$

C. Average Relative Imbalance (ARI)

Even though total charging time and utilization metric determine the performance of the algorithm, however, all CSs might not have balanced load distribution due to distance based distributed load. Due to long waiting queue on one CSs, it may be possible that BEVs may get charged early form other CSs. To avoid such discrepancy, ARI metric is used. This metric helps to achieve the balanced distribution, the Eq. 27 shows how ARI controls the highest and least load distributed among all CS.

$$ARI = \frac{MaxTime - MinTime}{MaxTime} \tag{28}$$

where $MinTime$ and $MaxTime$ represents the minimum time and maximum time taken for charging the assigned BEVs among all CSs which are calculated following Eq. 28 and Eq. 29 respectively.

$$MinTime = min_{1i \geq |CS|} TCT_i^{CS} \tag{29}$$

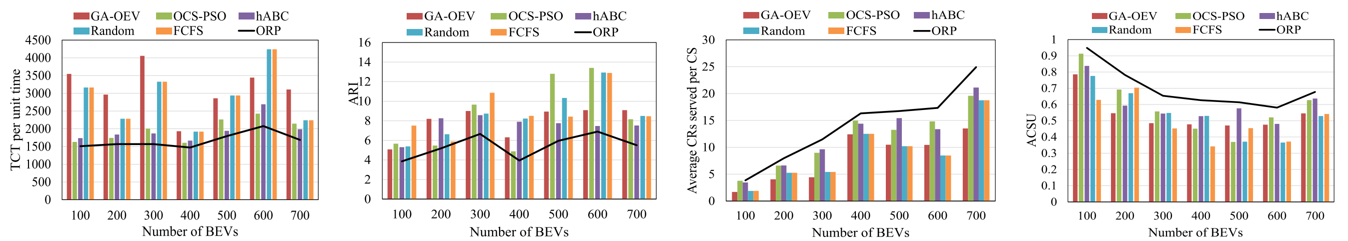
$$MaxTime = max_{1 \geq i \geq |CS|} TCT_i^{CS} \tag{30}$$

To verify the comparative performance behaviour of the proposed ORP algorithm, a set of diverse nature algorithms (First Come First Serve (FCFS) [21], Random [31], Genetic Algorithm (GA-OEV) [32], Particle Swarm Optimization (OCS-PSO) [33], hABC [34]) are identified and compared with the proposed ORP algorithm considering several QoS performance metrics. Additionally, to analyse the statistical behaviour of the algorithms in terms of average, best and standard deviation over a series of experiments, a study is also done.

1) Simulation Setup: All experiments are conducted on Windows 7 OS with 4GB RAM and Intel(R) $CORE_{TM}$ i5-4200U CPU of 2.6GHz. For conducting the experiments, the network scenarios of different number of BEVs and CSs are taken in to account. Number of BEVs are varied in the range (100, 700), and BEV window size is fixed to 100. The number of CSs is taken while maintaining the BEV density for balanced charging, therefore, number of CSs is varied from 1/10 to 1/30 of total number of BEVs. The heterogeneity among the BEVs is considered in three forms: low, medium and heavy BEVs.

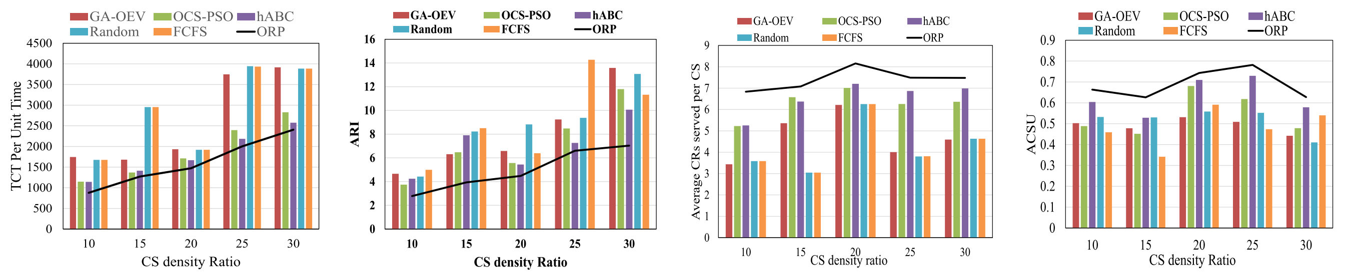
In the simulation, vehicular network's BEV location and CS locations are generated in an uniform square and their respective charging request demands and Number of charging stations in a zone are also generated randomly between [100, 700] and [10, 30] respectively. Accordingly, their respective needed charging duration is calculated on the basis of current and required SoC. Using this information to allocate CSs to the charging requests, euclidean distance between BEV and CS is taken into account. Finish time of charging for a BEV depends on travel time, waiting time, charging time and nature of CP (fast/ultra-fast). BEVs arrive in the network following Poisson distribution with a rate of $\tau \sim P(\lambda)$, where, $\lambda = 5$. It is assumed that each BEV can reach to at-least one CS with its current battery power. It is also assumed that maximum battery capacity of BEVs i is in the range of [20 80]Ah, and velocity is the direct ratio to the electricity consumption rate, it is taken in between 2 $3R_i^-$ and distance between each BEV and CS is in the range of 4-30 Km.

For HHO algorithm parameter settings, default constant β is taken as 1.5 and $J \in 0, 2$, rest of the parameters are taken same as suggested in [28]. Variables r_1, r_2, r_3, r_4, r_5 are randomly generated in range [0 1]. Population size and iteration numbers are varied from 20 to 80 and 10 to 50 respectively. Based on a series of experiments, it was analyzed that HHO produces its best fitness value with population size $= 40$ and number of



(a) Results on TCT (b) Results on ARI (c) Results on avg. CRs served per CS (d) Results on ACSU

Fig. 4. Comparative results of ORP with other state of the art algorithms on varied number of BEVs charging requests.



(a) Results on TCT (b) Results on ARI (c) Results on avg. CRs served per CS (d) Results on ACSU

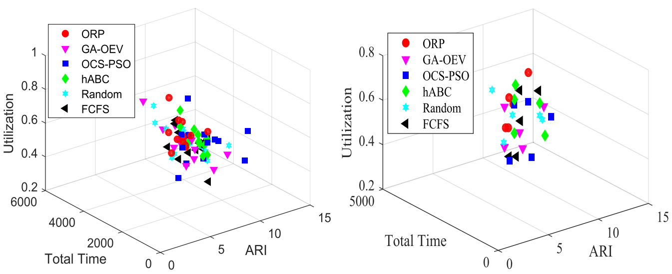
Fig. 5. Comparative results of ORP with other state of the art algorithms on varied number of CSs.

iterations = 30. Therefore, same parameters were considered for the experimentation. To avoid the high variability of the stochastic methods, 30 independent runs are taken for each experiment and average results are reported in this paper. The simulation data used is generated randomly to avoid user subjectivity in experimental results. We do agree that real driver pattern data would be able to show more clarity in result analysis for a particular city environment. However, for the general case study random driving pattern would give sufficient clarity about the result performance considering different metrics.

2) Testing Scenarios: The performance behaviour of the proposed ORP is analysed by conducting a set of experiments considering four QoS performance metrics (Eqs. 5, 22, 24 and 25) and three different scenarios: 1) varied BEVs charging requests, 2) varied CSs (in the city zone), and 3) varied multiple QoS performance metrics together (for both (first and second) scenarios).

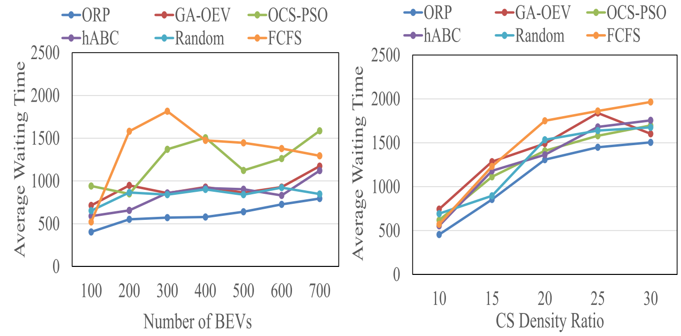
a) xtScenario 1: varied number of BEVs charging requests: In this scenario, the comparative performance behaviour of the algorithms is analysed by conducting a series of experiments where number of charging requests are varied in the range of (100, 700). The number of CSs per zone is fixed to 20 (where city is partitioned in four zones (north, south, east and west)). Results (in Figs. 4(a), 4(b), 4(c) and 4(d)) are on varied BEVs charging requests on TCT, ARI, Average CRs Served per CS and ACSU performance metrics respectively. 'Y' axis of the figure shows the respective performance metric, and 'X' axis denotes varied BEVs charging requests. From the Figs. 4(a) and 4(b) it is observed that the proposed ORP achieves minimal (optimal) TCT and ARI respectively, over other state of the art algorithms despite of increase in the BEVs charging requests. Further, results (in Figs. 4(c) and 4(d)) show that the proposed ORP also beats other algorithms for the Average CRs Served per CS and ACSU performance metrics respectively. The reason for this significant improvement in the performance of ORP (over other state of the art algorithms) is that ORP has an effective exploration and exploitation mechanism to refine the solution in an iterative fashion, and the proposed fitness function is built on the base of Fuzzy inference rules to automatic adjust the CPs based on BEVs requests during the run and effectively optimizes the respective QoS performance metric. Form this study, it is observed that the proposed ORP policy is sustainable to meet the desired QoS performance metrics for the case of when number of charging requests is increasing drastically. With this set of experiments, it is also observed that the proposed solution does not be much affected by increasing the number of BEV requests since it dynamically selects a window of BEV charging requests to process them together in an iteration. In this set of experiments, 100 BEV window size is considered by the RWE module. However, this window size can be increased with increased computational infrastructure.

b) Scenario 2: varied charging stations: Scenario 2 is designed to analyse the scalable performance behaviour of the proposed ORP. A set of experiments were conducted considering increased number of CSs in the city zone to analyse the comparative performance behaviour of the algorithms. Number of CSs per zones is varied in the range of (10, 30), and number of BEVs charging requests are fixed to 400 for this scenario. Results (in Figs. 5(a), 5(b), 5(c) and 5(d)) are obtained on varied CSs for TCT, ARI, Average CRs Served per CS and ACSU performance metrics respectively. Figs. 4(a) and 4(b) are the evidence for the optimal TCT and ARI respectively over other state of the art algorithms. Similarly, Figs. 4(c) and 4(d) prove the effectiveness of the



(a) Results for scenario 1 (b) Results for scenario 2

Fig. 6. Comparative results of ORP with other state of the art algorithms on multi parameters.



(a) Results on varied BEVs at the CP (b) Results on varied CS

Fig. 7. Comparative study for avg. waiting time per BEV by varying the number of BEVs charging requests and CSs.

TABLE III
ANALYSIS OF TOTAL TIME

Algorithm	Average (s)	Best (s)	STD (s)
ORP	1484.58	1085.03	219.81
GA-OEV	2490.66	1698.03	484.38
hABC	1696.27	1195.91	243.82
OCS-PSO	1746.02	1486.60	224.81
Random	2680.04	1747.34	573.08
FCFS	2880.04	1841.04	583.16

ORP to achieve better Average CRs Served per CS and ACSU respectively instead of increasing CSs in city. Noteworthy, this performance is enhanced due to the proposed fitness function as FIS is dynamically able to transfer the charging requests to fast/ultra-fast charging points by observing the BEVs charging requests, and following the fitness value, ORP iterative procedure converges in same direction to optimize QoS metrics. Study (in scenario 2) attests the scalability of the proposed ORP policy w.r.t. QoS performance metrics.

c) Scenario 3: Varied multiple QoS performance metrics: This scenario is to analyse the multi-parameter behaviour of the algorithms accounting the issues tackled in scenarios 1 and 2. Three parameters (TCT, ARI and ACSU) are considered to conduct this set of experiments. In this scenario, one can be interested to visualize the performance behaviour of the algorithms on multiple parameters. The testing parameters for this scenario are taken from the scenario 1 and scenario 2. Thus, Figs. 6(a) and 6 (b) show the results for the scenario 1 and scenario 2 respectively. From the results, it can be analysed that despite of contradicting nature of the multiple parameters, the proposed ORP achieves desired solution as it lies in most desired centroid region of 3D plots that is the most interested region. The statistical behaviour of the algorithms is also analysed to measure the performance metric (TCT), and results corresponding to best, average and standard deviation (STD) obtained by the algorithms are illustrated in Table III. From the results, it is observed that the proposed ORP also performs better than other state of the art algorithms in terms of average, best and STD of the results.

3) Summary of Observations: This simulation study is carried out with three test-case scenarios: 1) varied number of charging requests, and 2) varied number of charging stations for single performance metric i.e., TCT, ARI, avg. charging requests served per CS, and ACSU, and 3) varied multiple performance metrics. The results are compared with various state of the art methods i.e., GA-OEV, OCS-PSO, hABC, Random, and FCS. From this study, it is observed that the proposed ORP policy is quite effective to optimize the above mentioned performance metrics not only on single performance metric, it also significantly maintains the good trade-off among multiple performance metrics as shown in Fig. 6. With this, one is also interested to analyse the impact on average waiting time per BEV by varying the number of BEVs at the CP, and the number of charging stations in the simulated scenario. Thus, an experimental study is also done, and the comparative results are shown in Fig. 7. As vehicles arrive at the selected charging point, delay incurs while waiting for their respective turn that is shown in Fig. 7(a) and 7(b). Fig. 7(a) and (b) represent that waiting time of BEV utilizing ORP is lesser in comparison to other state of the art methods for varying number of BEVs charging request, and number of CSs respectively. The delay is incurred at the CP due to the previous ancestor BEVs getting their turn for arriving at the CP. Using ORP, this delay is mitigated because of comparatively much optimized resource allocation strategy. This study signifies that the proposed ORP policy is quite novel in terms of optimizing several user and service provider centric QoS parameters.

VI. CONCLUSION

In this paper, an Optimized Reservation Policy (ORP) framework is presented for scheduling charging request of BEVs to available CSs in the smart cities environment considering number of real time traffic parameters. The framework focuses on limited battery power of BEVs, traffic delays towards approaching CS and congestion at the CSs as important factors in decision making for charging. Initially, an optimization problem is identified considering the scheduling requirement of dense charging requests of BEVs on available Charging Points (CPs) at the several CSs using travel distance, traffic delay, travel time and fast/ultra-fast CP requirements in the CPs reservation. To solve the problem, an optimized CSs reservation policy for BEVs is proposed utilizing two soft computing techniques; i) Harris Hawk Optimization (HHO) and ii) Fuzzy Inference System (FIS). HHO enabled component in the framework offers an effective mapping between CP reservation requests and available CSs considering range of QoS parameters in traffic environment, and it acts as a global optimizer. FIS enabled component locally manages the CPs considering light, medium and heavy BEVs dynamically by observing the heterogeneity of BEVs at the respective CS. The comparative evaluation results attest the benefits of the proposed ORP framework against the state-of-the-art schemes. In future research, we will extended the experimental study considering specific city traffic environment, and related driver patter data for more precise and city specific recommendation. Further, the work will be extended considering next generation traffic environments such as drone enabled BEVs traffic management and availability of edge computing environment at drones. The consideration of consumer and service provider centric multiple optimisation parameters, and multi-objective algorithmic variants will also be the quest.

References

[1] J. C. Mukherjee and A. Gupta, "A review of charge scheduling of electric vehicles in smart grid," *IEEE Syst. J.*, vol. 9, no. 4, pp. 1541–1553, Dec. 2015.

[2] O. Kaiwartya and S. Kumar, "Geocast routing: Recent advances and future challenges in vehicular adhoc networks," in *Proc. Int. Conf. Signal Process. Integr. Netw. (SPIN)*, Feb. 2014, pp. 291–296.

[3] M. Aljaidi, N. Aslam, X. Chen, O. Kaiwartya, and Y. A. Al-Gumaei, "Energy-efficient EV charging station placement for E-mobility," in *Proc. 46th Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2020, pp. 3672–3678.

[4] M. M. Vazifeh, H. Zhang, P. Santi, and C. Ratti, "Optimizing the deployment of electric vehicle charging stations using pervasive mobility data," *Transp. Res. A, Policy Pract.*, vol. 121, pp. 75–91, Mar. 2019.

[5] H. Tu, H. Feng, S. Srdic, and S. Lukic, "Extreme fast charging of electric vehicles: A technology overview," *IEEE Trans. Transport. Electrific.*, vol. 5, no. 4, pp. 861–878, Dec. 2019.

[6] Y. Cao, T. Wang, X. Zhang, O. Kaiwartya, M. H. Eiza, and G. Putrus, "Toward anycasting-driven reservation system for electric vehicle battery switch service," *IEEE Syst. J.*, vol. 13, no. 1, pp. 906–917, Mar. 2019.

[7] A. U. Makarfi, K. M. Rabie, O. Kaiwartya, X. Li, and R. Kharel, "Physical layer security in vehicular networks with reconfigurable intelligent surfaces," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, May 2020, pp. 1–6.

[8] Y. Cao, T. Jiang, O. Kaiwartya, H. Sun, H. Zhou, and R. Wang, "Toward pre-empted EV charging recommendation through V2V-based reservation system," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 51, no. 5, pp. 3026–3039, May 2021.

[9] E. S. Rigas, S. D. Ramchurn, and N. Bassiliades, "Managing electric vehicles in the smart grid using artificial intelligence: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 1619–1635, Aug. 2015.

[10] F. Hausler, E. Crisostomi, A. Schlote, I. Radusch, and R. Shorten, "Stochastic park-and-charge balancing for fully electric and plug-in hybrid vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 2, pp. 895–901, Apr. 2014.

[11] M. Zhu, X.-Y. Liu, and X. Wang, "Joint transportation and charging scheduling in public vehicle systems—A game theoretic approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2407–2419, Aug. 2018.

[12] E. S. Rigas, S. D. Ramchurn, N. Bassiliades, and G. Koutitas, "Congestion management for urban EV charging systems," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, Oct. 2013, pp. 121–126.

[13] S. Anderson and V. J. Nair, "Electric vehicle charge scheduling on highway networks from an aggregate cost perspective," 2019, *arXiv:1901.03017*.

[14] X. Wang, C. Sun, R. Wang, and T. Wei, "Two-stage optimal scheduling strategy for large-scale electric vehicles," *IEEE Access*, vol. 8, pp. 13821–13832, 2020.

[15] F. Malandrino, C. Casetti, C.-F. Chiasserini, and M. Reineri, "A game-theory analysis of charging stations selection by EV drivers," *Perform. Eval.*, vols. 83–84, pp. 16–31, Jan. 2015.

[16] H. Qin and W. Zhang, "Charging scheduling with minimal waiting in a network of electric vehicles and charging stations," in *Proc. 8th ACM Int. Workshop Veh. Inter-Netw. (VANET)*, 2011, pp. 51–60.

[17] T. Zhang, W. Chen, Z. Han, and Z. Cao, "Charging scheduling of electric vehicles with local renewable energy under uncertain electric vehicle arrival and grid power price," *IEEE Trans. Veh. Technol.*, vol. 63, no. 6, pp. 2600–2612, Jul. 2014.

[18] J. Hu, S. You, M. Lind, and J. Østergaard, "Coordinated charging of electric vehicles for congestion prevention in the distribution grid," *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 703–711, Mar. 2014.

[19] N. Olsen and N. Kliewer, "Electric vehicle scheduling—A study on charging modeling for electric vehicles," in *Operations Research Proceedings 2016*. Cham, Switzerland: Springer, 2018, pp. 653–658.

[20] F. V. Cerna, M. Pourakbari-Kasmaei, R. A. Romero, and M. J. Rider, "Optimal delivery scheduling and charging of EVs in the navigation of a city map," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4815–4827, Sep. 2018.

[21] Y. Cao, N. Wang, G. Kamel, and Y.-J. Kim, "An electric vehicle charging management scheme based on publish/subscribe communication framework," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1822–1835, Sep. 2017.

[22] Y. Cao, T. Wang, O. Kaiwartya, G. Min, and A. H. Abdullah, "An EV charging management system concerning drivers' trip duration and mobility uncertainty," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 4, pp. 596–607, Apr. 2016.

[23] T. Conway, "On the effects of a routing and reservation system on the electric vehicle public charging network," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2311–2318, Sep. 2017.

[24] X. Tan, B. Sun, and D. H. K. Tsang, "Queueing network models for electric vehicle charging station with battery swapping," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, Nov. 2014, pp. 1–6.

[25] Y. Cao *et al.*, "A cost-efficient communication framework for battery-switch-based electric vehicle charging," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 162–169, May 2017.

[26] F. Ahmad, M. S. Alam, and S. M. Shariff, "A cost-efficient energy management system for battery swapping station," *IEEE Syst. J.*, vol. 13, no. 4, pp. 4355–4364, Dec. 2019.

[27] R. Devendiran, P. Kasinathan, V. K. Ramachandaramurthy, U. Subramaniam, U. Govindarajan, and X. Fernando, "Intelligent optimization for charging scheduling of electric vehicle using exponential Harris hawks technique," *Int. J. Intell. Syst.*, vol. 36, no. 10, pp. 5816–5844, Oct. 2021.

[28] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019.

[29] M. F. Tasgetiren, M. Sevkli, Y.-C. Liang, and G. Gencyilmaz, "Particle swarm optimization algorithm for single machine total weighted tardiness problem," in *Proc. Congr. Evol. Comput.*, vol. 2, Jun. 2004, pp. 1412–1419.

[30] S. N. Sivanandam, S. Sumathi, and S. N. Deepa, *Introduction to Fuzzy Logic Using MATLAB*, vol. 1. Berlin, Germany: Springer, 2007.

[31] A. Y. Zomaya and Y.-H. Teh, "Observations on using genetic algorithms for dynamic load-balancing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 12, no. 9, pp. 899–911, Sep. 2001.

[32] F. Alesiani and N. Maslekar, "Optimization of charging stops for fleet of electric vehicles: A genetic approach," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 3, pp. 10–21, Jul. 2014.

[33] J. An, B. Huang, Q. Kang, and M. Zhou, "PSO-based method to find electric vehicle's optimal charging schedule under dynamic electricity price," in *Proc. 10th IEEE Int. Conf. Netw., Sens. Control (ICNSC)*, Apr. 2013, pp. 913–918.

[34] J. G. Álvarez, M. González, C. R. Vela, and R. Varela, "Electric vehicle charging scheduling by an enhanced artificial bee colony algorithm," *Energies*, vol. 11, no. 10, p. 2752, 2018.