An Efficient Blockchain-based Electric Vehicle Charging Management System

Ahmed Khoumsi
Department Electrical and Computer Engineering, Université de Sherbrooke, Canada
Ahmed.khoumsi@usherbrooke.ca

2021 IEEE Symposium on Computers and Communications (ISCC) | 978-1-6654-2744-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISCC53001.2021.9631412

***Abstract*—Given a charging station (CS) infrastructure, we solve the problem of an electric vehicle (EV) that needs an efficient charging program that allows it to achieve a given objective which includes: get to its destination, and keep it as low as possible the overall cost and average waiting time of the EV near each CS. Besides developing a model and method to determine a charging program that allows the EV to meet its objective, we also propose a cryptocurrency-based mechanism to realize payments for the energy received. The determined charging program and corresponding payments are managed using a smart contract (SC) executed in the blockchain, to ensure that EV charging and payments comply with the rules and are recorded in a secure, immutable, transparent and decentralized manner.**

***Index Terms*—Electric vehicle (EV), charging station (CS), efficient charging program, linear programming, smart meters, payment, cryptocurrency, blockchain, smart contract, Ethereum.**

I. INTRODUCTION

To encourage people to adopt electric vehicles (EVs), it is essential to develop a dense and widespread charging station (CS) infrastructure and to keep charging costs as low as possible [1]. Let an infrastructure of public CSs and an EV that needs to be supplied with energy by one or more CSs during the trip that takes it to its destination. It is therefore necessary to determine a *charging program* which is intuitively specified by: the selected sequence of CSs to visit, the period of each visit, and the amount of energy to be received from each CS. Our *first contribution* is to determine automatically a charging program that respects the following 4-point objective:

Objective 1: 1) The total energy received from the selected CSs allows the EV to reach its destination and still have enough energy left to reach the CS closest to the destination. This condition prevents the EV from running out of energy after reaching it destination. 2) A CS is required to provide energy only in periods when it is available. 3) During its trip, the average waiting time of the EV near each CS before the charging starts, does not exceed a given threshold. 4) The global cost of the trip is as low as possible.

Each of the above four points i is referred to as *sub-objective* i. In Sect. II we develop a model to describe and analyze charging programs, and in Sect. III we develop a method to compute a charging program that ensures Objective 1.

Our *second contribution* (in Sect. IV) is to develop a blockchain-based framework to manage a selected charging program and corresponding payments for energy received from CSs. The use of blockchain and smart contract (SC) technology [2]–[5] ensures that EV charging and corresponding payments comply with the rules and are recorded in a *secure*, *immutable*, *transparent* and *fully decentralized* manner.

The rest of the paper is organized as follows: Sect. II presents the model developed to specify and analyze charging programs. In Sect. III, we suggest how to select an efficient charging program. Sect. IV shows how the blockchain and an SC are used to manage and record a charging program and the corresponding payments. A conclusion is given in Sect. V.

II. MODELING AND ANALYZING A CHARGING PROGRAM

Given a set of CSs, consider an EV that cannot reach sub-objective 1 using only its initial state of charge (i.e. without receiving energy from any CS). It is therefore paramount to select an adequate *charging program*, that is, a sequence of CSs the EV will visit, the period of each visit, and the amount of energy the EV will receive from each CS. The criterion used to select the charging program is to reach Objective 1. To solve this problem, we must first identify relevant characteristics to specify a CS and an EV.

A. Characteritics of CS and EV

A CS is noted CS_i and characterized by (l_i, π_i, T_i), which are distinguished in two categories: *constant* and *variable*. The constant characteristics of CS_i are: its location l_i and its energy unit price π_i. The characteristic of CS_i that varies with time is T_i which specifies the periods where CS_i is available. For example, the CS is not available in the periods already reserved for charging EVs. Another example: the CS is not available in some periods to respect its charging capacity.

An EV is characterized by $(\gamma, \lambda, \psi, a_{\max}, a, l, l_{\text{dest}}, \tau, \omega_{\max})$, which too are distinguished into *constant* and *variable* characteristics. The constant characteristics of the EV: γ is its wear cost per unit of distance; λ is its energy consumption per unit of distance; ψ is the time it takes to be charged with a unit of energy; $a_{\max}$ is its maximum autonomy, i.e. the distance it can travel with a full state of charge. The characteristics of the EV that vary at each charging request: a ($\leq a_{\max}$) is its initial autonomy, i.e. the distance it can still travel when it requests charging; l is its initial location, i.e. its location when it requests charging; l_{dest} is the location of its destination; τ is the time when it wants to start its trip after having made its request; and $\omega_{\max}$ is the maximum average time it accepts to wait at each CS before the charging starts.

978-1-6654-2744-9/21/$31.00 ©2021 IEEE

B. Charging and partial programs and their basis

In this section, we define a charging program, a partial program, and a basis of a (charging or partial) program.

1) Charging program: Consider an EV that needs to be charged by one or more CSs to reach its destination. A charging program of the EV must therefore be established, which specifies: the sequence $(\mathrm{CS}_i)_{i=1\cdots q}$ of CSs the EV must visit in order, the amount of energy e_i to be received from each CS_i, and the time ν_i at which charging by each CS_i will begin. A charging program of an EV is therefore defined by a sequence $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$.

Given $(\mathrm{CS}_i)_{i=1\cdots q}$ and an EV, the following can be deduced from their characteristics using a web mapping system (e.g. Google Maps): $d = \mathrm{dist}(l, l_{\mathrm{dest}})$ and $t = \mathrm{time}(l, l_{\mathrm{dest}})$ are the distance and time to travel *directly* from l to l_{dest}; $d_0 = \mathrm{dist}(l, l_1)$ and $t_0 = \mathrm{time}(l, l_1)$ are the distance and time to travel from l to CS_1; $d_i = \mathrm{dist}(l_i, l_{i+1})$ and $t_i = \mathrm{time}(l_i, l_{i+1})$ are the distance and time to travel from CS_i to CS_{i+1}, for $i = 1, \cdots, q-1$; $d_q = \mathrm{dist}(l_q, l_{\mathrm{dest}})$ and $t_q = \mathrm{time}(l_q, l_{\mathrm{dest}})$ are the distance and time to travel from CS_q to l_{dest}.

2) Partial program: It is the sequence $(\mathrm{CS}_i, e_i)_{i=1\cdots q}$, where the times $(\nu_i)_{i=1\cdots q}$ of a charging program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$ are omitted.

3) Basis of a program: Consider a charging program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$ or a partial program $(\mathrm{CS}_i, e_i)_{i=1\cdots q}$. The corresponding CS sequence $(\mathrm{CS}_i)_{i=1\cdots q}$ is called the basis of these programs.

C. Autonomy evolution

Consider a partial program $(\mathrm{CS}_i, e_i)_{i=1\cdots q}$ of an EV. During the execution of a corresponding charging program, when the EV has visited $\mathrm{CS}_1, \cdots, \mathrm{CS}_i$ and received respectively the energy $e_1, \cdots, e_i$, for $i \leq q$, its autonomy is updated as follows: it is reduced by the traveled distance $\sum_{k=0}^{i-1} d_k$, it is increased by $\frac{1}{\lambda}\sum_{k=1}^{i} e_k$, where $\sum_{k=1}^{i} e_k$ is the energy supplied by $(\mathrm{CS}_k)_{k=1\cdots i}$. Therefore, the autonomy a_i of EV just before leaving CS_i is computed by (1).

$$a_i = a - \sum_{k=0}^{i-1} d_k + \frac{1}{\lambda}\sum_{k=1}^{i} e_k \tag{1}$$

D. Feasible program

Intuitively, a feasible program of an EV is a realizable charging program that ensures sub-objective 1. More rigorously, a feasible program is a partial program $(\mathrm{CS}_i, e_i)_{i=1\cdots q}$ respecting the following three conditions:

- The initial autonomy of the EV allows it to move to CS_1.
- For each $i = 1 \cdots q-1$, after receiving the energy $(e_k)_{k=1\cdots i}$ from $(\mathrm{CS}_k)_{k=1\cdots i}$ respectively, the autonomy of the EV before leaving CS_i is sufficient to reach CS_{i+1}.
- After receiving $(e_i)_{i=1\cdots q}$ from all $(\mathrm{CS}_i)_{i=1\cdots q}$ respectively, the autonomy of the EV before leaving CS_q is sufficient to reach l_{dest} and still have enough energy to be able to move to the closest CS. Let $d_{l,\mathrm{cs}}$ be the distance from the destination to the closest CS.

The above three conditions are formally and respectively defined by the three lines of (2).

$$\left.\begin{array}{l} d_0 \leq a \\ \forall i \in \{1, \cdots, q-1\} : d_i \leq a_i \leq a_{\max} \\ (d_q + d_{l,\mathrm{cs}}) \leq a_q \leq a_{\max} \end{array}\right\} \tag{2}$$

E. Eligible CS sequence

Intuitively, a sequence $(\mathrm{CS}_i)_{i=1\cdots q}$ is *eligible* by an EV if there exists a realizable charging program using such CS sequence that ensures sub-objective 1. More rigorously, a CS sequence $(\mathrm{CS}_i)_{i=1\cdots q}$ is *eligible* by a given EV if there exists a feasible program that has $(\mathrm{CS}_i)_{i=1\cdots q}$ as its basis.

Proposition 1: $(\mathrm{CS}_i)_{i=1\cdots q}$ is eligible by a given EV, if and only if (3) is satisfied (note the similarity with (2)).

$$\left.\begin{array}{l} d_0 \leq a \\ \forall i \in \{1, \cdots, q-1\} : d_i \leq a_{\max} \\ (d_q + d_{l,\mathrm{cs}}) \leq a_{\max} \end{array}\right\} \tag{3}$$

The three lines of (3) respectively state that: the initial autonomy of the EV allows it to move to CS_1; from each CS_i $(i < q)$, the maximum autonomy of the EV allows it to reach CS_{i+1}; and from CS_q, the maximum autonomy of the EV allows it to reach its destination and still have enough energy to move to the closest CS.

F. Charging evolution for imposed charging times

Let us study the execution over time of a given charging program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$, hence times $(\nu_i)_{i=1\cdots q}$ are given. More precisely, the aim is to determine $(\tau_i, \omega_i)_{i=1\cdots q}$, where τ_i is the time when the EV reaches CS_i and ω_i is the waiting time of the EV before energy charging by CS_i begins (see Fig. 1). The series $(\tau_i, \omega_i)_{i=1\cdots q}$ is computed as specified in (4) from: τ and ψ (given in the characteristics of the EV), $(e_i, \nu_i)_{i=1\cdots q}$, and $(t_i)_{i=1\cdots q}$ (computed using a web mapping system as explained in Sect. II-B).

$$\left.\begin{array}{l} \tau_1 = \tau + t_0 \\ \forall i = 1 \cdots q-1 : \tau_{i+1} = \nu_i + \psi e_i + t_i \\ \forall i = 1 \cdots q : \omega_i = \nu_i - \tau_i \end{array}\right\} \tag{4}$$

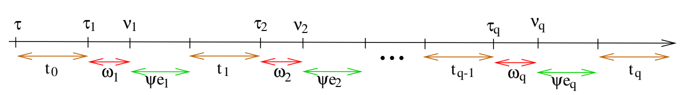

Fig. 1. Charging evolution over time.

G. Charging evolution respecting sub-objective 2

We now consider the case where $(\nu_i)_{i=1\cdots q}$ are not imposed, and we will see how to compute $(\tau_i, \omega_i, \nu_i)_{i=1\cdots q}$ so that sub-objective 2 is respected. This is realized by: rewriting line 3 of (4) as shown in (5), and adding (6) to compute ω_i as a value which ensures sub-objective 2, that is, CS_i supplies energy to the EV only when it is available.

$$\left.\begin{array}{l} \tau_1 = \tau + t_0 \\ \forall i = 1 \cdots q-1 : \tau_{i+1} = \nu_i + \psi e_i + t_i \\ \forall i = 1 \cdots q : \nu_i = \tau_i + \omega_i \end{array}\right\} \tag{5}$$

$$\forall i = 1 \cdots q : [\tau_i + \omega_i, \tau_i + \omega_i + \psi e_i] \subseteq T_i \tag{6}$$

Instead of accepting any values of $(\omega_i)_{i=1\cdots q}$ that respect (6), we may select the smallest ones, that is, respecting (7):

$$\forall i = 1 \cdots q : \omega_i = \text{ smallest } x \text{ s.t. } [\tau_i + x, \tau_i + x + \psi e_i] \subseteq T_i \tag{7}$$

H. Acceptable program

Intuitively, an acceptable program is a realizable charging program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$ that ensures sub-objectives 1-3. Formally, a charging program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$ is acceptable if: the corresponding partial program $(\mathrm{CS}_i, e_i)_{i=1\cdots q}$ is feasible (i.e. ensures sub-objective 1), and the series $(\tau_i, \omega_i)_{i=1\cdots q}$ computed using (4) respects (6) and (8), which ensure sub-objectives 2 and 3 respectively.

$$\frac{1}{q} \sum_{i=1}^{q} \omega_i \leq \omega_{\max} \tag{8}$$

I. Example

Fig. 2 shows two CSs CS_1 and CS_2 located in $(2,1)$ and $(7,1)$, and an EV located in $(0,0)$ and whose destination is in $(10,0)$. The CS closest to the destination is a third CS at location $(12,1)$. The characteristics of CS_1 and CS_2 are $(l_1, \pi_1, T_1) = ((2,1), 0.9, [3.5, \infty])$ and $(l_2, \pi_2, T_2) = ((7,1), 1.1, [11.5, \infty])$. The characteristics of the EV are $(\gamma, \lambda, \psi, a_{\max}, a, l, l_{\mathrm{dest}}, \tau, \omega_{\max}) = (1, 1, 0.5, 6, 3, (0,0), (6,0), 0, 0.5)$. For simplicity and without loss of generality, the routes of the EV to reach its destination (directly or via CS_1 and CS_2) are considered as the crow flies. So we obtain $d = \mathrm{dist}(l, l_{\mathrm{dest}}) = 10$, $d_0 = \mathrm{dist}(l, l_1) = \sqrt{5}$, $d_1 = \mathrm{dist}(l_1, l_2) = 5$, and $d_2 = \mathrm{dist}(l_2, l_{\mathrm{dest}}) = \sqrt{10}$, and $d_{l,\mathrm{cs}} = 2$. Let us assume that the times obtained by a web mapping service are $t_0 = \mathrm{time}(l, l_1) = 3$, $t_1 = \mathrm{time}(l_1, l_2) = 5$, and $t_2 = \mathrm{time}(l_2, l_{\mathrm{dest}}) = 4$.

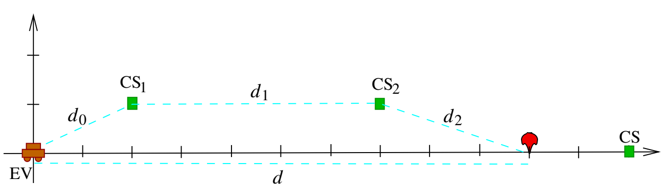

Fig. 2. Example of a charging program using two CSs.

Since the initial autonomy of the EV is 3 which is $< d = 10$, the EV cannot reach its destination without charging. Using (3) of Prop. 1, we find that each of CS_1 and CS_2 is not eligible when it is taken alone. The ineligibility of CS_1 is intuitively explained by the fact that the maximum autonomy ($= 6$) of the EV does not allow it to reach its destination from CS_1. The ineligibility of CS_2 is intuitively explained by the fact that the initial autonomy ($= 3$) of the EV does not allow it to reach CS_2 from its initial location. Using Prop. 1, we also find that $(\mathrm{CS}_1, \mathrm{CS}_2)$ is eligible while $(\mathrm{CS}_2, \mathrm{CS}_1)$ is ineligible. Hence, the only solution, if any, to reach sub-objective 1 is that the EV receives energy from CS_1 and then from CS_2.

From (1,2), we obtain a *feasible* program by associating to CS_1 and CS_2 respectively any charging e_1 and e_2 that satisfy: $\sqrt{5} + 2 \leq e_1 \leq \sqrt{5} + 3$ and $\sqrt{5} + \sqrt{10} + 4 \leq (e_1 + e_2) \leq \sqrt{5} + 8$. Intuitively, these conditions on (e_1, e_2) ensure sub-objective 1. In the rest of the example, we take $e_1 = \sqrt{5} + 3$ and $e_2 = \sqrt{10} + 1$, which respects the above constraints.

Let us complete the above feasible program $(\mathrm{CS}_i, e_i)_{i=1\cdots 2}$ to obtain an acceptable program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots 2}$, i.e. that respects sub-objectives 2 and 3 (in addition to sub-objective 1). For that purpose, we compute $(\tau_i, \nu_i)_{i=1\cdots 2}$ using (5), and $(\omega_i)_{i=1\cdots 2}$ using (7). So the EV starts moving at time $\tau = 0$ and reaches CS_1 at time $\tau_1 = \tau + t_0 = 0 + 3 = 3$. Since CS_1 is available in the time interval $T_1 = [3.5, \infty]$, the EV must wait $\omega_1 = 0.5$ before starting to be supplied with energy at time $\nu_1 = \tau_1 + \omega_1 = 3.5$. Charging time is $\psi e_1 = \frac{\sqrt{5}+3}{2} \approx 2.62$ and hence terminates at time $\nu_1 + \psi e_1 \approx 3.5 + 2.62 = 6.12$. Then EV travels during $t_1 = 5$ to reach CS_2 at time $\tau_2 = \nu_1 + \psi e_1 + t_1 \approx 6.12 + 5 = 11.12$. Since CS_2 is available in the time interval $T_2 = [11.5, \infty]$, the EV must wait $\omega_2 \approx 11.5 - 11.12 = 0.38$ before starting to be supplied with energy at time $\nu_2 = \tau_2 + \omega_2 = 11.5$. Charging time is $\psi e_2 = \frac{\sqrt{10}+1}{2} \approx 2.08$ and hence terminates at time $\nu_2 + \psi e_2 \approx 11.5 + 2.08 = 13.58$. Then EV travels during $t_2 = 4$ to reach its destination at time $\nu_2 + \psi e_2 + t_2 \approx 13.58 + 4 = 17.58$.

The obtained charging program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots 2}$ respects sub-objectives 2 and 3. Indeed, sub-objective 2 is respected from the fact that $(\tau_i, \omega_i)_{i=1\cdots 2}$ is computed using: (5) which is equivalent to (4), and (7) which is stronger than (6)). Sub-objective 3 is respected from the fact that the obtained $(\omega_i)_{i=1\cdots 2}$ respect (8) since $\frac{1}{2}(\omega_1 + \omega_2) \approx 0.44 < 0.5$.

III. Determination of efficient charging program

In Sect. II, we have considered acceptable programs, i.e. charging program that ensure sub-objectives 1-3. In this section, we will show how to compute an acceptable program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$ that ensures sub-objective 4 (in addition to sub-objectives 1-3). In other words, we have to determine an acceptable program where the cost for the EV is as low as possible. For that purpose, we need to develop a model to evaluate the cost of any acceptable program.

A. Cost and pricing models

Consider an acceptable program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$. In a first step, we take its corresponding feasible program $(\mathrm{CS}_i, e_i)_{i=1\cdots q}$, because we will use a cost model that is independent of timing aspects. A time dependent cost model is left for future work.

In the charging program, the EV travels the distance $\sum_{k=0}^{q} d_k$. The wear cost is therefore $\gamma \sum_{k=0}^{q} d_k$. Since each CS_i provides the amount e_i of energy at the price π_i per unit of energy, the cost of the transferred energy is $\sum_{i=1}^{q} \pi_i e_i$. The global cost of the charging program is therefore given by (9), and the EV must pay to each CS_i the amount given by (10).

$$\mathcal{C} = \gamma \sum_{k=0}^{q} d_k + \sum_{i=1}^{q} \pi_i e_i \tag{9}$$

$$\forall i = 1 \cdots q : \mathrm{pay}_i = \pi_i e_i \tag{10}$$

B. Selection of an efficient charging program

Given an EV that wants to be charged by one or more CSs to reach Objective 1, the problem is to determine an acceptable program whose cost is as low as possible for the EV. Such problem is solved using Algorithm 1; let us explain it w.r.t the four sub-objectives 1-4. Line 1 identifies every CS sequence $(\mathrm{CS}_i)_{i=1\cdots q}$ that is a basis of partial programs $(\mathrm{CS}_i, e_i)_{i=1\cdots q}$ respecting sub-objective 1. The for-loop (lines 2-6) is applied for each CS sequence identified in Line 1. In line 3, the least costly feasible program of an identified CS sequence is determined using linear programming which consists in minimizing an expression under constraints [6]. The expression to minimize is given by (9) where the unknown vector to be determined is $(e_i)_{i=1\cdots q}$. The constraints to respect are given by (2) in which a_i is replaced by its expression of (1). Line 4 completes the least costly feasible program found in line 3 to obtain a charging program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$ that respects sub-objective 2 (in addition to sub-objective 1). Lines 5-6 keep the charging program obtained in line 4 only if it respects sub-objective 3 (in addition to sub-objectives 1 and 2). Line 7 selects the least costly acceptable program among all acceptable programs obtained in the various instances of lines 5-6. Sub-objective 4 is ensured by lines 3 and 7.

Algorithm 1: Selection of efficient charging program

Input: all known CSs

Output: efficient acceptable program

```
1 Identify all eligible CS sequences
2 for every identified eligible sequence (CS_i)_{i=1...q} do
3     Determine the least costly feasible program
        (CS_i, e_i)_{i=1...q} that has (CS_i)_{i=1...q} as basis
4     Compute (τ_i, ω_i, ν_i)_{i=1...q} using (5,7)
5     if (ω_i)_{i=1...q} respects (8) then
6         the corresponding charging program
            (CS_i, e_i, ν_i)_{i=1...q} is acceptable
7 return the least costly acceptable program found
```

C. Example

We continue with our example of Sect. II-I. Since we only have two CSs CS_1 and CS_2, there are only 4 possible sequences of CSs: (CS_1), (CS_2), $(\mathrm{CS}_1, \mathrm{CS}_2)$, and $(\mathrm{CS}_2, \mathrm{CS}_1)$. In Sect. II-I, we have obtained that only $(\mathrm{CS}_1, \mathrm{CS}_2)$ is eligible, so it is the only sequence to consider. We have also obtained that sub-objective 1 is respected if and only the charging (e_1, e_2) associated to $(\mathrm{CS}_1, \mathrm{CS}_2)$ satisfies the following two conditions: $\sqrt{5} + 2 \leq e_1 \leq \sqrt{5} + 3$ and $\sqrt{5} + \sqrt{10} + 4 \leq (e_1 + e_2) \leq \sqrt{5} + 8$.

From (9), we compute that the global cost of the charging program is composed of the wear cost $1 \times (\sqrt{5} + 5 + \sqrt{10}) \approx 10.40$ and the cost of the transferred energy $0.9e_1 + 1.1e_2$, so that the global cost is $\mathcal{C} \approx 10.40 + 0.9e_1 + 1.1e_2$. After resolution using linear programming, the obtained charging that minimizes that cost while respecting the above two conditions (i.e. that respects sub-objective 4), is $e_1 = \sqrt{5} + 3$ and $e_2 = \sqrt{10} + 1$. Let us explain intuitively how this result is obtained: Since the global cost increases linearly with e_1 and e_2, we must select the *minimum* total charging $e_1 + e_2$ while satisfying the condition $\sqrt{5} + \sqrt{10} + 4 \leq (e_1 + e_2) \leq \sqrt{5} + 8$. Hence, we take $e_1 + e_2 = \sqrt{5} + \sqrt{10} + 4$. Note that the weights of e_1 and e_2 in the global cost are 0.9 and 1.1, respectively. Since $0.9 < 1.1$, we must select the *maximum* e_1 (or equivalently, the *minimum* e_2) while satisfying the condition $\sqrt{5} + 2 \leq e_1 \leq \sqrt{5} + 3$. Hence, we take $e_1 = \sqrt{5} + 3$. Then from $e_1 + e_2 = \sqrt{5} + \sqrt{10} + 4$ and $e_1 = \sqrt{5} + 3$, we deduce $e_2 = \sqrt{10} + 1$. For this optimal charging, the EV has to pay $0.9(\sqrt{5} + 3) \approx 4.71$ to CS_1, and $1.1(\sqrt{10} + 1) \approx 4.58$ to CS_2. The minimal global cost, including the wear cost, is therefore $\mathcal{C} \approx 10.40 + 4.71 + 4.58 = 19.69$. To recapitulate, we have obtained a partial program $(\mathrm{CS}_i, e_i)_{i=1\cdots 2}$ that respects sub-objectives 1 and 4.

As for the values of $(\nu_i, \omega_i)_{i=1\cdots 2}$, they have been computed in Sect. II-I as $\nu_1 = 3.5$, $\omega_1 = 0.5$, $\nu_2 = 11.5$ and $\omega_2 \approx 0.38$. We have verified in Sect. II-I that the resulting charging program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots 2}$ is acceptable, i.e. respects sub-objectives 1-3. Hence, Objective 1 is reached.

IV. Blockchain-based management of a charging program and corresponding payments

In this section, we show how blockchain [2]–[5] is used to manage a charging program specified and selected as shown in Sects. II and III, and make payments for energy received by the EV from CSs. Blockchain has been used for several applications like smart grids [7] and the Internet of Vehicles [8], [9]. Closer to our topic, several studies have been done on the use of blockchain for EV charging [10]–[18].

The two most widely known blockchain implementations are Bitcoin [2] and Ethereum [19]. In our study, we consider Ethereum as it provides a programming language, called Solidity [20], to program any given logic. The obtained program is called *smart contract* (SC) and is executed in Ethereum to enforce the corresponding logic. An SC is quite similar to a software object with attributes, a constructor and functions.

EVs and CSs can register as *external* entities of Ethereum. On the other hand, an SC is an *internal* entity of Ethereum from the time it is deployed on the blockchain. Every entity X (whether internal or external) has an account in Ethereum which is identified by an address that we will denote by $@_{\mathrm{X}}$.

A. Simplifying assumptions

To simplify the presentation, and without loss of generality, we use a *single* SC to realize all the logic of the charging program. We will refer to such an SC as "the SC". For lack of space, we also use the following simplifications:

- We assume that the EV and the CSs involved in a charging program behave correctly. We have developed a solution which takes *faulty* behaviors into account, but which is not presented for lack of space.
- Payments include only charging costs, not the basic costs required to executing transactions in Ethereum.

B. Necessary features of CS and EV

1) Necessary features of CS: A CS must be equipped with an entity that can *enable* (resp. *disable*) energy supply, in order to permit (resp. prevent) an EV that is connected to the CS to be (resp. from being) supplied with energy by the CS. Such entity is typically controlled by an administrator of the CS. A CS must also be equipped with a *smart meter* that provides on-request information which has been recorded for a time interval specified in the request (e.g. the last hour). Examples of recorded information: the energy supplied over time, periods where the CS is connected to some EV, periods where the CS has enabled energy provision. A CS can order its smart meter to send a transaction to SC which contains the information recorded in a given time interval.

2) Necessary features of EV: Like for CSs, an EV must be equipped with an entity that can *enable* (resp. *disable*) energy reception, in order to permit (resp. prevent) a CS that is connected to the EV to provide (resp. from providing) energy to the EV. Such entity is typically controlled by the driver of the EV. An EV must also be equipped with a *smart meter* that provides on-request information which has been recorded for a time interval specified in the request. Examples of recorded information: the energy received over time, periods where the EV is connected to some CS, periods where the EV has enabled energy reception. An EV can order its smart meter to send a transaction to SC which contains the information recorded in a given time interval.

To ensure privacy, the information recorded by a smart meter should only be accessible to SC. This can be realized using a cryptographic solution, where the smart meter encrypts the recorded information using the public key of SC, and SC decrypts the information using its private key.

C. EVC-coin cryptocurrency

Payments are realized in the blockchin by transferring values of a cryptocurrency. Instead of using Ether which is Ethereum's default cryptocurrency, we propose EVC-coin, a cryptocurrency that should be created and used for payments related to EV charging (EVC) applications. In Ethereum, a cryptocurrency is usually created as a smart contract that implements an interface named ERC20. We assume that every EV and CS that wishes to participate in a charging program, has first obtained a sufficient EVC-coin balance.

D. Deployment of the SC and its initial operations

Initially, the SC is deployed in Ethereum and its deployer receives an event from the blockchain containing several information including $@_{\mathrm{SC}}$. The constructor of the SC is automatically executed when the SC is deployed. After some initializations, the constructor publishes the *specification* and *audit* of the SC and also other information addressed to CSs and EVs so that they can apply for registration. Simply speaking, the specification of the SC describes the rules it is supposed to enforce, while the audit certifies that the implementation of the SC conforms to its specification. SCs can be audited by specialized companies, such as ChainSecurity [21], Trail of Bits [22], and Certik [23].

E. CS and EV registration

1) CS registration: After consulting the information published by the SC, a CS_i can apply for registration by sending to the blockchain a specific transaction that contains its characteristics (l_i, π_i, T_i) (see Sect. II-A). As we will see, T_i will be updated by the blockchain when CS_i's availability period varies. The transaction implies the execution of a function of the SC that: sets CS_i as a listener of the SC (so that the SC can communicate with CS_i using events), and emits an event to require from CS_i the payment of a deposit $\mathcal{D}_{\mathrm{CS}}$. CS_i transfers in due time the amount $\mathcal{D}_{\mathrm{CS}}$ of EVC-coin from $@_{\mathrm{CS}_i}$ to $@_{\mathrm{SC}}$. Then the SC registers CS_i and emits an event to inform CS_i that its registration request is accepted.

2) EV registration: EV registration is similar as CS registration, except that an EV provides only its *constant* characteristics $(\gamma, \lambda, \psi, a_{\max})$ and makes a different deposit $\mathcal{D}_{\mathrm{EV}}$. As we will see, the *variable* characteristics $(a, l, l_{\mathrm{dest}}, \tau, \omega_{\max})$ are provided by the EV when it applies for a charging.

Remark 1: A deposit is only relevant in the presence of faulty behaviors, in order to financially penalize (resp. compensate) the perpetrator (resp. victim) of any faulty behavior. Here, we use deposits in anticipation of the general solution that takes faulty behaviors into account.

The interactions of a CS or an EV with the SC during registration are illustrated in Fig. 3, where: time evolves from top to bottom, a plain arrow represents a transaction sent by a CS or an EV to the blockchain which implies the execution of a function of the SC, and a dashed arrow represents an event emitted by the SC and detected by a CS or an EV.

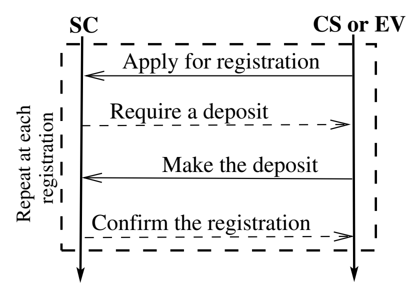

Fig. 3. CS and EV registration.

F. Charging protocol

Any registered EV (identified by $@_{\mathrm{EV}}$ and specified by its constant characteristics $(\gamma, \lambda, \psi, a_{\max})$) can apply for a charging by sending to the blockchain a specific transaction that contains its *variable* characteristics $(a, l, l_{\mathrm{dest}}, \tau, \omega_{\max})$. The transaction implies the execution by the SC of a function that executes the following operations:

- Compute a charging program $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$ and payments $(\mathrm{pay}_i)_{i=1\cdots q}$ as shown in Sects. III-A and III-B.

- Emit an event to: inform the EV of $(\mathrm{CS}_i, e_i, \nu_i)_{i=1\cdots q}$ and $(\mathrm{pay}_i)_{i=1\cdots q}$, and require from the EV the payment of $\mathrm{Pay} = \sum_{i=1\cdots q} \mathrm{pay}_i$.

The EV transfers in due time the amount Pay of EVC-coin from $@_{\mathrm{EV}}$ to $@_{\mathrm{SC}}$. Then, for each CS_i, $i = 1 \cdots q$, the SC updates the (variable) characteristic T_i to specify that CS_i will be occupied in the period $[\nu_i, \nu_i + \psi e_i]$ for the charging of an EV. Then the following will be executed iteratively for each charging of the EV by CS_i, $i = 1 \cdots q$:

- The SC emits an event to inform CS_i that at time ν_i it must start supplying the energy e_i to $@_{\mathrm{EV}}$ at the price pay_i.
- The SC emits an event to ask the EV that before time ν_i, it must be connected to CS_i and have sent a transaction $\mathrm{T_{wait}}$ to the SC indicating that it is connected to CS_i and waiting for the energy charging to begin.
- The SC receives $\mathrm{T_{wait}}$ from the EV in due time.
- The SC emits an event to ask CS_i to supply the EV connected to it with the amount e_i of energy.
- CS_i supplies the EV with the required amount of energy.
- CS_i and the EV order their respective smart meters SM_i and $\mathrm{SM_{EV}}$ to send a transaction for SC that contains the information recorded during the charging.
- SM_i and $\mathrm{SM_{EV}}$ send their transactions.
- The SC obtains the information of the transactions sent by SM_i and $\mathrm{SM_{EV}}$ and verifies that the charging was successful.
- The SC transfers the amount pay_i of EVC-coin from $@_{\mathrm{SC}}$ to $@_{\mathrm{CS}_i}$.

For simplicity and without loss of generality, processing and communication times of the three types of participants (i.e. SC, EV with its SM, and CSs with their SMs) are negligible compared to the time it takes to charge an EV. The sequence of interactions between the SC and the participants during the execution of a charging protocol is illustrated in Fig. 4. In addition to plain and dashed arrows (for transactions and events, respectively) already used ing Fig. 3, we also use a dotted arrow to represent an order given by a CS or the EV to its SM, and a selfloop arrow to specify local computer operations by the SC. Also, the three types of participants are visually distinguished by colors.

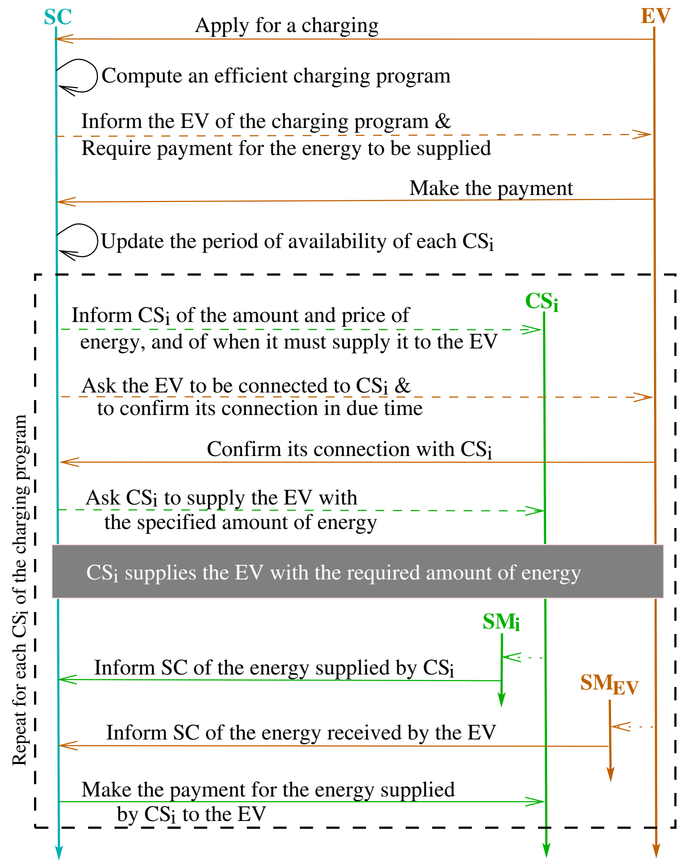

Fig. 4. Charging protocol.

G. *Unsubscriptions and SC lifetime*

A registered CS or EV should have the possibility to unsubscribe by sending a specific transaction to the SC. The SC may require some conditions before accepting an unsubscribe request, for example that the requester has used the services of the SC at least a given number of times. If the unsubscribe request is accepted, the SC returns the deposit ($\mathcal{D}_{\mathrm{CS}}$ or $\mathcal{D}_{\mathrm{EV}}$) to the requester and removes it from its lists of subscribers and listeners.

Solidity language provides the function “selfdestruct” which can be called by the SC to stop its existence. The SC may be designed so that it calls “selfdestruct” under given conditions, for example: if SC remains inactive during a given long time, if all registered CSs have unsubscribed. Before calling “selfdestruct”, the SC can for example transfer the balance of its account to an account specified in the deployment phase as an input argument of the constructor of the SC.

V. Conclusion

A. *Summary of contributions*

Several studies have been done on the use of blockchain for EV charging [10]–[18]. Our contributions compared to these references are summarized below.

1) First contribution: Given an infrastructure of public CSs, we study the problem of an EV (or rather its driver) that needs an efficient charging program that allows it to achieve a given objective which includes: arrive at destination and still have enough energy left to reach the CS closest to the destination; keep it as low as possible the overall cost and average waiting time of the EV near each CS before the charging starts; require from a CS to provide energy only in periods when it is available. We develop a model of charging program and a model of cost, and present a method to determine a charging program that allows the EV to reach its objective.

2) Second contribution: We show in great detail how blockchain and smart contract technology can be used to manage a charging program, enforce the corresponding payments, and record all operations in a secure, immutable, transparent and fully decentralized manner.

B. Ongoing work

1) Robustness to faulty behaviors: We have assumed that the EV and the CSs involved in a charging program behave *correctly* (i.e., according to the protocol, see Figs. 3 and 4). We have just finalized the development of a solution which takes *faulty* behaviors of EV and CS into account, but which cannot be presented for lack of space. The outline of our solution is to automatically perform the following operations: detect every faulty behavior of the EV or a CS during the execution of a charging program; identify the perpetrator(s) and victim(s) of the faulty behavior; determine the financial penalty (resp. indemnity) to be paid by the perpetrator(s) (resp. to the victim(s)) of the faulty behavior; and enforce the payments of penalties and indemnities (in addition to the payments for the energy supplied). The use of blockchain and SC is even more justified in the presence of faulty behaviors, as SC detects faulty behaviors and enforces the payments of the corresponding penalties and indemnities.

2) Reduce the computational complexity: All sequences of any number of CSs have been considered before determining an adequate charging program that allows an EV to meet its objective. The procedure might be computationally costly because the number of sequences of CSs can be very high. We are currently finalizing a solution that reduces the computational complexity, but at the price of possibly providing a less optimal solution. The principle of the solution is to seek first the least costly charging program involving a single CS. If no charging program is found, repeat the search among charging programs involving two CSs, and so on.

C. Future work

1) Formal study of the charging protocol: We plan to formally describe the developed charging protocol, in order to rigorously validate it and, if necessary, optimize and correct it.

2) Scalability evaluation: We plan to implement our proposal, probably on Ethereum, and evaluate its scalability in real life scenarios.

3) Integrate the waiting time in the minimization problem: In this paper, the waiting times of the EV near the CSs before the energy charging begins are only considered to eliminate charging programs where the average waiting time before energy charging begins exceeds a threshold. We plan to study how to introduce the charging program duration (including waiting times) in the cost model so that time is taken into account in the minimization of the overall cost.

4) Use a variable charging rate: We have assumed that the charging rate is constant for each vehicle. We plan to use a more realistic charging model, where the charging rate varies with the state of charge.

5) Consider the basic transaction costs: For simplicity, our cost model does not consider the basic costs required to executing transactions in Ethereum. We plan to remove this simplification.

6) Investigate a decentralized solution: The proposed solution uses a single SC. We plan to investigate the possibility of improving our solution by decentralizing it using multiple SCs instead of just one.

References

[1] O. Egbue and S. Long, "Barriers to widespread adoption of electric vehicles: An analysis of consumer attitudes and perceptions," *Energy Policy*, vol. 48, pp. 717–729, Sept. 2012.

[2] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2009. [Online]. Available: http://www.bitcoin.org/bitcoin.pdf

[3] Z. Zen, S. Xie, H.-N. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: architecture, consensus, and future trends," in *IEEE 6th Int. Congress on Big Data*, Honolulu, Hawaii, USA, June 2017.

[4] ——, "Blockchain challenges and oportunities: a survey," *Int. Journal of Web and Grid Services (IJWGS)*, vol. 14, no. 4, 2018.

[5] A. Ushmani, "Blockchain insight," *Int. Journal of Computer Science Trends and Technology (IJCST)*, vol. 7, no. 2, pp. 1–3, 2019.

[6] G. Dantzig and M. Thapa, *Linear Programming, 1:Introduction*. Springer, 1997.

[7] M. B. Mollah, J. Zhao, D. Niyato, K.-Y. Lam, X. Zhang, A. M. Ghias, L. H. Koh, and L. Yang, "Blockchain for future smart grid: a comprehensive survey," *IEEE Internet of Things journal*, vol. 8, no. 1, pp. 18–43, 2020.

[8] A. Hammoud, H. Sami, A. Mourad, H. Otrok, R. Mizouni, and J. Bentahar, "AI, blockchain, and vehicular edge computing for smart and secure IoV: challenges and directions," *IEEE Internet of Things Magazine*, vol. 3, no. 2, pp. 68–73, June 2018.

[9] L. Mendiboure, M. A. Chalouf, and F. Krief, "Survey on blockchain-based applications in internet of vehicles," *Computers and Electrical Engineering*, vol. 84, 2018.

[10] R. Su, "Supervisor synthesis to thwart cyber attack with bounded sensor reading alterations," *Automatica*, vol. 94, pp. 35–44, 2018.

[11] M. Baza, M. Nabil, M. Ismail, M. Mahmoud, E. Serpedin, and M. A. Rahman, "Blockchain-based charging coordination mechanism for smart grid energy storage units," in *IEEE Int. Conf. Blockchain*, 2019, pp. 504–509.

[12] X. Huang, C. Xu, P. Wang, and H. Liu, "LNSC: A security model for electric vehicle and charging pile management based on blockchain ecosystem," *IEEE Access*, vol. 6, pp. 13 565–13 574, 2018.

[13] F. Knirsch, A. Unterweger, and D. Engel, "Privacy-preserving blockchain-based electric vehicle charging with dynamic tariff decisions," *Computer Science-Research and Development*, vol. 33, no. 1–2, pp. 71–79, 2018.

[14] B. Kirpes and C. Becker, "Processing electric vehicle charging transactions in a blockchain-based information system," in *Twenty-fourth Americas Conf. Information Systems*, New Orleans, 2018, pp. 504–509.

[15] M. Kim, K. Park, S. Yu, J. Lee, Y. Park, S. Lee, and B. Chung, "A Secure charging system for electric vehicles based on blockchain," *Sensors*, vol. 33, no. 3028, 2019.

[16] U. Asfia, V. Kamuni, A. Sheikh, S. Wagh, , and D. Patel, "Energy trading of electric vehicles using blockchain and smart contracts," in *18th European Control Conf. (ECC)*, Napoli, Italy, June 2019, pp. 504–509.

[17] R. Jin, X. Zhang, Z. Wang, W. Sun, X. Yang, and Z. Shi, "Blockchain-enabled charging right trading among EV charging stations," *Energies*, vol. 12, no. 3922, 2019.

[18] F. C. Silva, M. A. Ahmed, J. M. Martinez, and Y.-C. Kim, "Design and implementation of a blockchain-based energy trading platform for electric vehicles in smart campus parking lots," *Energies*, vol. 12, no. 4814, 2019.

[19] Ethereum whitepaper. [Online]. Available: https://ethereum.org/en/whitepaper/

[20] Solidity. [Online]. Available: https://docs.soliditylang.org/

[21] Blockchain security - security audits and services. [Online]. Available: https://chainsecurity.com/

[22] Deepening the science of security. [Online]. Available: https://www.trailofbits.com/

[23] Certik foundation. [Online]. Available: https://certik.org/