A Lightweight User Interface for Smart Charging of Electric Vehicles: A Real-World Application

Stefan Meisenbacher[1], Karl Schwenk[1,2], Johannes Galenzowski[1], Simon Waczowicz[1], Ralf Mikut[1], and Veit Hagenmeyer[1]
[1] *Karlsruhe Institute of Technology, Institute for Automation and Applied Informatics, Karlsruhe, Germany*
[2] *Mercedes-Benz AG, Sindelfingen, Germany*
stefan.meisenbacher@kit.edu, {karl.schwenk, johannes.galenzowski, simon.waczowicz, ralf.mikut, veit.hagenmeyer}@kit.edu

***Abstract*—Intelligent power management of Electric Vehicles (EVs) integrates them into the power system to support grid reliability. However, uncoordinated EV charging, where the battery is fully charged at maximum available power after plugging in, is still the common practice. Introducing smart charging on a large scale necessitates a high acceptance by users of EVs. Making the underlying charging system tangible to the user is therefore mandatory. We present the User Interface (UI) of the *Smart Charging Wizard* web app to start, adjust and monitor the charging process. The web app is integrated into an Internet of Things (IoT) architecture to connect with the charging station. We demonstrate the operation of the UI and show how the charging system responds to dynamic adjustments by the EV user. The UI is customizable and enables future field studies to involve EV users and their individual preferences.**

Keywords—electric vehicles, smart charging, power distribution systems, mobile applications, internet of things, battery aging

I. INTRODUCTION

The electrification of the transport sector is receiving enormous attention due to social movements for climate and government paradigm shifts established by the 2015 Paris Agreement [1]. With the increasing penetration of Electric Vehicles (EVs), uncoordinated EV charging may cause a significant negative impact on power grid stability when many EVs are charged at peak load times [2,3]. In this situation, either the grid utilities have to be upgraded, which involves huge investments, or the EV charging load requires to be distributed intelligently. Smart charging aims to support the power grid in peak load times by power refeeding – Vehicle-to-Grid (V2G) – or by shifting the energy consumption to times of low demand and sufficient local power grid capacities [4,5]. To utilize the time and energy flexibility in EVs' charging processes, an integrated optimization scheme is mandatory [6]. However, the charging scheme needs to be tangible for the EV user to support a high user acceptance. Key to user acceptance are the awareness of grid stability and the capability of integrating renewable energy sources. Inhibitors to acceptance are the EV user's desire for individual mobility [7] and EV battery aging [8].

Various mobile app solutions have been proposed to involve EV users in the smart charging process. EV routing aims to predict the availability of charging stations nearby [9], and prioritize EV charging demand [10]. Charging control allows the user to monitor the process and set individual charging currents, either by manual intervention or by an optimization algorithm. Published optimization objectives are manifold. In [11], the system controls the power drawn by a household's charging station depending on other loads in the home. An open-source package with a smart charging algorithm is presented in [12]. The algorithm manages the charging of an EV fleet, minimizing peak load and electricity costs while taking into account infrastructure constraints.

In the broad research on smart charging, many approaches have been discussed theoretically; however, customizable solutions and real-world applications are still required. In this paper, we present the detailed realization of the *Smart Charging Wizard* web app, introduced in [13]:

- we briefly summarize the web app's Internet of Things (IoT) architecture in Section II,
- detail the User Interface (UI) in Section III, which was not shown in [13], and
- exemplify the UI visualization and EV user interaction with three scenarios in the Energy Lab 2.0 [14], a real-world environment for testing new approaches to stabilize power grids.

The proposed IoT architecture supports researchers to implement and realize new smart charging schemes, and the UI visualization makes the embedded processes transparent for EV users. The underlying optimization scheme [15] considers the user's wish for individual mobility, takes into account battery aging to compensate for the EV's value loss, and addresses the grid operator's stake through a dynamic electricity price.

II. ARCHITECTURE

Smart charging needs communication technology to integrate EV user interactions into the process. Fig. 1 shows an overview of the IoT architecture proposed in [13] and the participants. The EV user interacts with the *Smart Charging Wizard* with a smart device via HyperText Transfer Protocol (HTTP). The *Smart Charging Wizard* is a web app with an optimization module and a User Interface (UI), described in Section III. This web app runs together with the Charging Session Handler (CSH) and the Message Queuing Telemetry Transport (MQTT) server in a cloud environment. The MQTT server establishes communication between the web app, the CSH, and the Programmable Logic Controller (PLC) via JavaScript Object Notation (JSON) encoded messages. For communication between the PLC and the charging station, we choose the User Datagram Protocol (UDP), because it provides the largest number of parameters for our station compared to Modbus Transmission Control Protocol (TCP) and Open Charge Point Protocol (OCPP). The PLC sets the current at the charging station using the International Electrotechnical Commission (IEC) 61851 standard, receives the station's actual status and measurements, and forwards them via MQTT to the *Smart Charging Wizard*.

A. Charging Optimization

To consider the interests of the EV user, the EV manufacturer, and the grid operator, we define a multimodal optimization problem. Grid operators can create a decentralized incentive system for EV charging by dynamic electricity pricing. The profitability of V2G services based on bidirectional power flow depends on the electricity price spread and battery aging costs; the latter inhibits power refeeding from being profitable in present electricity tariffs [13,15]. Therefore, we amend the optimization problem [15] in the initial deployment of the *Smart Charging Wizard* to a unidirectional[1] charging scheme and obtain the following EV- and charging station-independent objective:[2]

$$\min_{\boldsymbol{p}\in\mathbb{R}^N} \sum_{\forall n\in N} J_{\mathrm{E},n}(p_n, \epsilon_{\mathrm{buy},n}) + J_{\mathrm{D},n}(\theta_n, e_n, H_0). \qquad (1)$$

We target to minimize the sum of electricity costs $J_{\mathrm{E},n}$ and aging costs $J_{\mathrm{D},n}$ in all time intervals $n \in N$ by controlling the decision variable, the charging power $\boldsymbol{p} = (p_0, p_1, \ldots, p_{N\text{-}1})^{\mathrm{T}}$. $J_{\mathrm{E},n}$ depend on the electricity price $\epsilon_{\mathrm{buy},n}$ and p_n, whereas $J_{\mathrm{D},n}$ is a function of the battery's temperature θ_n, its charge capacity e_n, and its state of health H_0. Thus, e_n, θ_n, and H_0 represent the battery state, which is affected by p_n, and requires adequate modeling. We apply an electrical, thermal, and aging model, described and validated in [15].[3] e_n, θ_n, and p_n are subject to constraints inherent with the battery's operating window and EV- and charging station-specific limits.

B. Charging Session Handler

The CSH runs independently of the *Smart Charging Wizard* and the PLC and manages access to a charging session from multiple devices [13]. More precisely, the charging process can be started from one device and edited or stopped later from another device using the session token set in the UI (Section III). The CSH creates the charging session and persists the corresponding data including the session token after receiving the optimization result from the *Smart Charging Wizard*. During the charging session, the CSH periodically sends the most recent optimal power setpoint to the PLC until the scheduled end of the session is reached or a stop is initiated by the user. Updating the session via UI overwrites the persisted session data. We deploy the CSH in a *Docker* [16] container on *Kubernetes* [17] to provide a continuously available service. The CSH subscribes to an MQTT topic, where the *Smart Charging Wizard* publishes initiated or updated charging sessions [13].

C. PLC Controlled Charging Station

The control of charging stations with a PLC enables implementing a generic Application Programming Interface (API) [13], independent of the stations' types and manufacturers. Apart from the *Smart Charging Wizard*, other devices and services may access a charging station by this API. Moreover, the PLC enables the integration of validation functions and safety limits. The PLC receives power setpoint messages from the CSH and verifies its format and reviews if the charging station's current limits are met. In case of an invalid message, the PLC publishes a response with an error code to report the failed request. If the message is valid, the PLC stores its values and sets the trigger for the next charging power update at the station. The PLC cyclically queries measured values and the station's state and transfers them to the *Smart Charging Wizard* for live information on the UI. Power setpoint requests are pooled in an MQTT topic, where the PLC's MQTT client subscribes. Thereby, the PLC concentrates communication between attached charging stations and multiple stakeholders via one IP address [13].

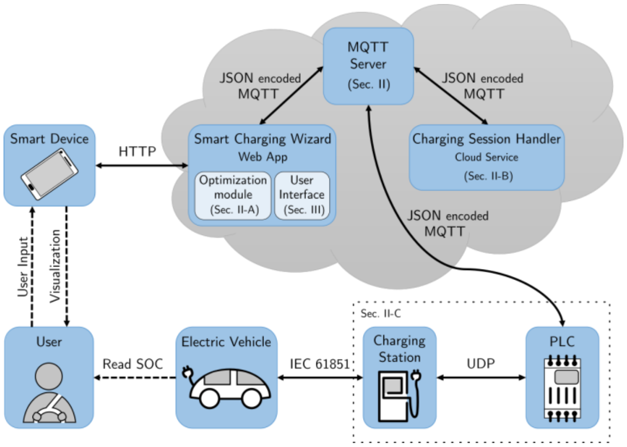


Fig. 1. An overview of the proposed IoT architecture and the communication protocols used [13].

III. USER INTERFACE

Adapting the charging power over time in an optimal way is nontrivial for humans. A simple user interface (UI) assists users to capture input variables and gives insights into the optimization scheme by visualizing the optimal charging power profile with the associated State of Charge (SOC) progression. Integrated live monitoring of the charging process provides a transparent view of the system's technical characteristics. A web-based UI offers a wide variety of possible components but typically requires a large amount of design effort. For an efficient UI design, we use *Streamlit* [18], a *Python*-driven [19] framework for building interactive web apps directly around the optimization module using given UI components, e.g., buttons, sliders and plots. Our user interface consists of three pages, accessible via a page selector in the

---

[1] The presented IoT architecture is ready for future integration of bidirectional charging.

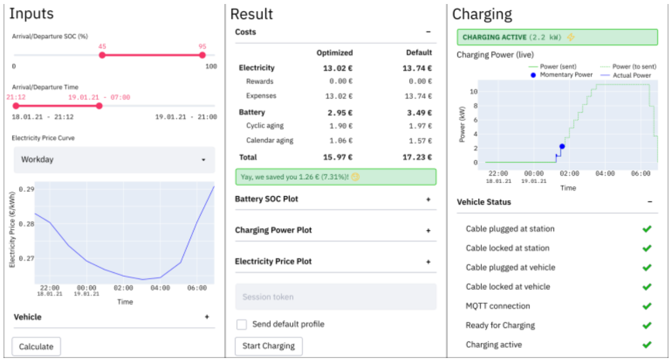
[2] The detailed formulation of the optimization problem including constraints can be found in [15].

[3] As the model parameters are EV-specific, the Smart Charging Wizard provides a default model, which allows demonstration but requires adaption to the actual EV.

sidebar. Fig. 2 shows the three pages with the sidebar hidden. The details for each page are explained in the following.

A. *Input Page*

This page allows the user to set up the charging optimization, see Fig. 2. After parking, the user sets the arrival SOC and selects the required departure SOC for the next usage of the EV.[5] The arrival time defaults to the current time, and the departure time reflects the EV's next usage. We provide five characteristic electricity price curves to demonstrate the smart charging scheme, selectable via drop-down and visualized below: workday (default), weekend, overproduction, underproduction, and static. The first four price curves are based on an analysis of 2019 European Power EXchange (EPEX) spot day-ahead prices, supplemented by taxes and subsidies of the Federal Republic of Germany [20].[6] The EV model and constraints associated with the EV, such as the SOC and charging power range, can be set in the "Vehicle"-expander[7] (shown as closed). After setting up the charging optimization, the user may start the problem solving with the "Calculate"-button. If a solution is found, a green banner appears to notify the user of the calculation success, and the app switches to the *Result Page*. Otherwise, a red banner provides troubleshooting hints, e.g., if the problem is infeasible, or the solver does not converge.



	Optimized	Default
Electricity	**13.02 €**	**13.74 €**
Rewards	0.00 €	0.00 €
Expenses	13.02 €	13.74 €
Battery	**2.95 €**	**3.49 €**
Cyclic aging	1.90 €	1.97 €
Calendar aging	1.06 €	1.57 €
Total	**15.97 €**	**17.23 €**

Yay, we saved you 1.26 € (7.31%)!

☐ Send default profile

CHARGING ACTIVE (2.2 kW)

Cable plugged at station	✓
Cable locked at station	✓
Cable plugged at vehicle	✓
Cable locked at vehicle	✓
MQTT connection	✓
Ready for Charging	✓
Charging active	✓

Fig. 2. The three page UI of the *Smart Charging Wizard* [4].

B. *Result Page*

This page displays the optimization result, see Fig. 2. The cost table compares the expenses of smart charging with non-optimized charging, broken down into their cost components: electricity expenses and V2G rewards, and battery aging costs due to cyclic and calendar degradation. Two separate graphs show the charging power and the corresponding evolution of the EV's SOC, allowing a comparison of the smart and non-optimized charging curves. In Fig. 2, these graphs are hidden in the "Plot" expanders, yet we show the predictive SOC progression in Section IV. The user can specify a session token to persist the charging session. We provide a choice between smart and non-optimized charging; the latter will be applied if checking "Send default profile". When the user clicks the "Start Charging" button, the app first verifies charging readiness by checking the connectivity, and switches to the *Charging Page* after starting the charging process.[8]

C. *Charging Page*

This page enables monitoring of the charging process, see Fig. 2. If charging is active, a green banner indicates the currently set power at the station.[9] The live power plot shows the active power measurement, records the realized power profile, and shows the planned profile. Information about the EV and the charging station is provided, e.g., plug/lock state of the cable and connectivity. The user may terminate the active charging session by clicking the "Stop Charging"-button.

IV. APPLICATION

The experimental deployment of the smart charging application enables the analysis of optimization schemes that engage the EV user. We integrate the charging application into the Energy Lab 2.0 real-world environment [14].

A. *Setup*

The deployed *Smart Charging Wizard* web app includes an optimization module, described in Section II-A. We demonstrate the realization of smart charging in three scenarios, which are based on characteristic electricity price curves (Section III); the scenario conditions are given in Table 1. We solve the constrained optimization problem (1) with the *CasADi* framework [21] using the nonlinear *Ipopt* solver [22]. As the electricity price has an hourly resolution, the temporal resolution for the optimization $\Delta t = t_{n+1} - t_n = 15$ min is sufficient. Compared to the scenarios in [13], the scenarios in this paper show the visualization of the expected SOC progression, displayed to the EV user on the UI of the *Smart Charging Wizard* web app.

B. *Results*

Scenario I Fig. 3-a shows the graph of the predicted SOC progression, provided in the *Results Page* of the UI (Section III). Electricity oversupply leads to low prices throughout the day, which increase towards the evening. The green line indicates the optimized profile, and the black line shows the curve of default charging. The cost savings compared to default charging are given in Table 1.

Scenario II The charging optimization starting from 7:30 to a planned departure on 12:30 results in the profile shown in Fig. 3-b. While default charging (black line) would ignore the price valley, smart charging shifts the optimized profile (green line) to the times of low electricity prices.

---

[4] https://energylabsmartcharging.github.io/Smart-Charging-Wizard/

[5] The hardware we installed does not support accessing the SOC via a communication interface, which will be provided with the CHAdeMO and ISO 15118 standards.

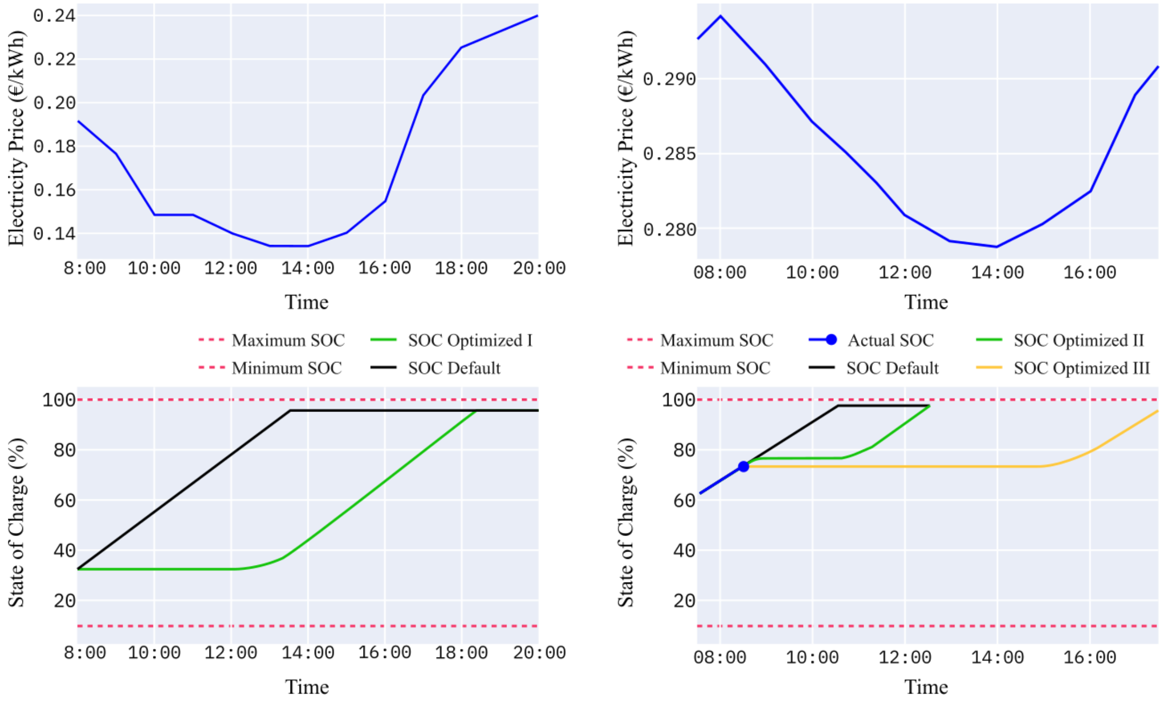
[6] In future applications, these price curves can be replaced by an actual price signal.

[7] A panel that can be expanded or collapsed with a click to show or hide UI components.

[8] If an active charging session exists, it can be updated by entering its session token.

[9] Without an active charging session, the page displays only the state. Accessing an active session from another device or later on requires entering the session token.

Scenario III At 8:30, the EV user in *Scenario II* decides not to use the EV until 17:30. The yellow line in Fig. 3-b shows the SOC progression after the user interaction. The blue line represents the already realized charging profile until this point in time. Rescheduling for a later EV usage frees up energy flexibility and optimized charging occurs between 15:00 and 19:00.



(a) (b)

Fig. 3. Charging optimization in a power overproduction scenario (a) and a workday scenario (b).

In all three scenarios, the consideration of calendar battery aging results in a shift of charging to a later time, instead of symmetrically filling electricity price valleys.[10] The operating costs summary in the UI makes the cost components transparent for the EV user, and the avoidance of excessive battery aging cost contributes to battery-friendly charging.

TABLE I. OPTIMIZATION SCENARIO CONDITIONS AND RESULTS. ABSOLUTE COST VALUES CORRESPOND TO THE OPTIMIZED PROFILE; RELATIVE COST VALUES REFER TO THE DIFFERENCE TO DEFAULT CHARGING

	Scenario I	Scenario II	Scenario III
Price curve	overproduction	workday	workday
t arr. – dep.	08:00 – 20:00	07:30 – 12:30	07:30 – 17:30
SOC arr. – dep.	35 % – 95 %	63 % – 95 %	63 % – 95 %
p min. – max.	0 kW – 11 kW	0 kW – 11 kW	0 kW – 11 kW
Operating costs	-19 % / 11.92 €	-6 % / 10.84 €	-30 % / 8.05 €
Aging costs	-15 % / 3.65 €	-9 % / 1.92 €	-2 % / 2.05 €
Electricity costs	-21 % / 8.28 €	-6 % / 8.93 €	-37 % / 6.01 €

arr.: arrival, dep.: departure, min.: minimum, max.: maximum

V. CONCLUSION

The transparent inclusion of the Electric Vehicle (EV) user's interests is a decisive factor for the acceptance of smart charging. In this paper, we introduce the User Interface (UI) of the *Smart Charging Wizard* web app, outline the app's integration into the Internet of Things (IoT) architecture, and exemplify the UI visualization and user interaction in three scenarios, realized in the Energy Lab 2.0 real-world environment. The optimization scheme considers charging-related battery aging and dynamic electricity prices as a stimulation measure from the power grid operator. Thereby, the charging system supports grid stability by EV power management. We provide a default EV battery model and representative electricity price curves for demonstration. Both modules allow the integration of custom models, e.g., a battery model of a specific EV or adaptive pricing. The exemplary scenarios show the UI visualizations, which are necessary for the transparent integration of the EV user into the charging scheme.

In future work, the optimization scheme of the *Smart Charging* Wizard will consider stochastic influences, e.g., uncertainties inherent to intermittent renewable power generation and random EV users' behavior. Furthermore, we plan to extend the IoT system by solar power generation and stationary battery storage.

---

[10] Minimizing idle time at high SOC reduces lithium-ion battery degradation [23].

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

SM developed the optimization scheme and wrote the paper, KS designed the User Interface and deployed the *Smart Charging Wizard* web app. JG established communication between the charging station and the PLC, VH acquired the funding for the Energy Lab 2.0, all authors had approved the final version.

ACKNOWLEDGMENT

The authors would like to thank Friedrich Wiegel, Tobias Moser, and Jan Lukas Wachter for hardware support. The authors thank Mercedes-Benz AG, Sindelfingen for the provided EVs. SM, SW, RM, and VH are funded by the Helmholtz Association under the Program Energy System Design (ESD) and the Joint Initiative Energy System 2050 – A Contribution of the Research Field Energy.

REFERENCES

[1] UNFCCC, “Paris Agreement,” 2015.

[2] A. Dubey and S. Santoso, “Electric Vehicle Charging on Residential Distribution Systems: Impacts and Mitigations,” IEEE Access, vol. 3 pp. 1871–1893, 2015, doi: 10.1109/ACCESS.2015. 2476996.

[3] A. Ramanujam et al., “Quantifying the Impact of Electric Vehicles on the Electric Grid: A Simulation-based Case-study,” in Proc. 8th Int. Conf. Future Energy Systems (e-Energy ’17), Association for Computing Machinery, New York, USA, 2017, pp. 228-233 doi: 10.1145/3077839.3077854.

[4] A. Al Zishan, M. M. Haji, and O. Ardakanian, “Adaptive Control of Plug-in Electric Vehicle Charging with Reinforcement Learning,” in Proc. 11th ACM Int. Conf. Future Energy Systems (e-Energy ’20), Association for Computing Machinery, New York, USA, 2020, pp. 116-120, doi:10.1145/3396851.3397706.

[5] P. Staudt, M. Schmidt, J. Gärttner, and C. Weinhardt, “Using Vehicle-to-grid Concepts to Balance Redispatch Needs: A case study in Germany,” in Proc. 9th Int. Conf. Future Energy Systems (e-Energy ’18), Association for Computing Machinery, New York, USA, 2018, pp. 80-84, doi:10.1145/3208903.3208926.

[6] K. Schwenk et al., “Data-driven Charging Strategies for Grid-beneficial, Customer-oriented and Battery-preserving Electric Mobility,” in Proc. 29th Workshop Computational Intelligence, Dortmund, Germany, 2019, pp. 73-93.

[7] C. Will and A. Schuller, “Understanding User Acceptance Factors of Electric Vehicle Smart Charging,” Transportation Research Part C: Emerging Technologies, vol. 71, pp. 198-214, 2016, doi: 10.1016/j.trc.2016.07.006.

[8] A. E. Trippe, R. Arunachala, T. Massier, A. Jossen, and T. Hamacher, “Charging Optimization of Battery Electric Vehicles Including Cycle Battery Aging,” in IEEE PES Innovative Smart Grid Technologies, Europe, IEEE, 2014, pp. 1-6, doi: 10.1109/ ISGTEurope.2014.7028735.

[9] G. Alface, J. C. Ferreira, and R. Pereira, “Electric Vehicle Charging Process and Parking Guidance App,” Energies, vol. 12, no. 11, pp. 2123-1-16, 2019, doi:10.3390/en12112123.

[10] A. Anil, S. Azeem, B. Panicker, and V. K. Saifudeen, “Electric Vehicle Charging Assistive System using Internet of Electric Vehicles,” in Proc. 2019 Int. Conf. Microelectronics, Signals and Systems, AIP Publishing, Melville, New York, USA, 2020, pp. 040016-1-7, doi: 10.1063/5.0003949.

[11] R. A. Sousa et al., “Development of an IoT System with Smart Charging Current Control for Electric Vehicles,” in Proc. 44th Annu. Conf. of the IEEE Industrial Electronics Society (IECON ’18), IEEE, Piscataway, USA, 2018, pp. 4662-4667, doi: 10.1109/IECON.2018.8591174.

[12] O. Frendo, N. Gaertner, and H. Stuckenschmidt, “Open Source Algorithm for Smart Charging of Electric Vehicle Fleets,” IEEE Trans. Industrial Informatics, 2020, doi: 10.1109/TII.2020. 3038144.

[13] S. Meisenbacher et al., “Poster: Smart Charging of Electric Vehicles with Cloud-based Optimization and a Lightweight User Interface: A Real-World Application in the Energy Lab 2.0,” in Proc. 12th ACM Int. Conf. Future Energy Systems (e-Energy ’21), Association for Computing Machinery, New York, USA, 2021.

[14] V. Hagenmeyer et al., “Information and communication technology in Energy Lab 2.0: Smart Energies System Simulation and Control Center with an Open-street-map-based Power Flow Simulation Example,” Energy Technology, vol. 4, no. 1, 2016, doi: 10.1002/ ente.201500304.

[15] K. Schwenk et al., "Integrating Battery Aging in the Optimization for Bidirectional Charging of Electric Vehicles," in IEEE Trans. on Smart Grid, 2021, doi: 10.1109/TSG.2021.3099206.

[16] D. Merkel, “Docker: Lightweight Linux Containers for Consistent Development and Deployment,” Linux J., vol. 239, no. 2, 2014.

[17] Cloud Native Computing Foundation, “Kubernetes,” https://www.kubernetes, 2020.

[18] Streamlit Inc., “Streamlit,” https://www.streamlit.io, 2020.

[19] Python Software Foundation, “Python,” https://www.python.org, 2020.

[20] E. Thalman and B. Wehrmann, “What German Households Pay for Power,” https://www.cleanenergywire.org/factsheets/what-german-households-pays-power, accessed: 2021-05-28, 2020.

[21] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, “CasADi,” https://web.casadi.org/docs, 2018.

[22] A. Wächter and L. T. Biegler, “On the Implementation of an Interior-point Filter Line-search Algorithm for Large-scale Nonlinear Programming,” Mathematical Programming, vol. 106, no. 1, pp. 25-57, 2006, doi:10.1007/s10107-004-0559-y.

[23] M. Woody, M. Arbabzadeh, G. M. Lewis, Keoleian G. A., and A. Stefanopoulou, “Strategies to Limit Degradation and Maximize Li-ion Battery Service Lifetime – Critical review and guidance for stakeholders,” J. Energy Storage, vol. 28, pp. 101231, 2020, doi: 10.1016/j.est.2020.101231.