Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Evolution of Policies for Fair and Efficient Public Electric Vehicle Charging

Steffen Limmer, Angus Kenny, Tapabrata Ray, Felix Lanfermann, Hemant Kumar Singh, Andrea Castellani

2026

Preprint:

This is an accepted article published in IEEE Congress on Evolutionary Computation (CEC) 2026. The final authenticated version is available online at: https://doi.org/10.1109/CEC69861.2026.11727159 Copyright 2026 IEEE

Evolution of Policies for Fair and Efficient Public Electric Vehicle Charging

Steffen Limmer*, Angus Kenny‡, Tapabrata Ray†, Felix Lanfermann*, Hemant Kumar Singh† and Andrea Castellani*

*Honda Research Institute Europe, 63073 Offenbach am Main, Germany

{steffen.limmer, felix.lanfermann, andrea.castellani}@honda-ri.de

†The University of New South Wales, Canberra ACT 2600, Australia

{t.ray, h.singh}@unsw.edu.au

‡Monash University, Clayton VIC 3168, Australia

gus.kenny@monash.edu

***Abstract*—The present paper addresses the problem of controlling the charging of electric vehicles (EVs) at a public charging site with limited total power. To achieve an efficient and fair distribution of the available power or energy among EVs, a recently proposed multi-objective genetic programming (GP) approach is adapted to the considered public charging scenario. The approach evolves rule-based charging control policies using historical data from the charging site. An evaluation in simulation experiments shows that the GP-approach not only outperforms various manually designed rule-based controllers, but is also highly competitive with mixed-integer linear programming-based charging control.**

***Index Terms*—electric vehicle charging, fairness, energy management, genetic programming.**

I. Introduction

Economic or technical factors, such as peak load charges or transformer constraints, might make it necessary to limit the total charging power at electric vehicle (EV) charging sites. Controlled charging can be used to enforce the power limit, but it may result in unmet charging requests. In this case, the charging site operator is typically interested in a control strategy that satisfies as many requests as possible. Additionally, fairness among users is of increasing importance [1]. A range of optimization-based [2], [3], [4] and rule-based approaches [5], [6] for efficient and fair EV charging under limited capacity have been proposed in the literature.

In a recent work, Limmer et al. [7] proposed an approach for learning charging policies from historical charging site data using multi-objective genetic programming (GP). They evaluated the approach in a residential charging scenario and showed that it achieves higher efficiency and fairness than commonly used manually designed rule-based controllers, as well as the approach from [8], which combines multiple rule-based controllers. In the considered residential charging scenario, energy requirements and departure times of users are assumed to be unavailable as inputs to the charging controller, and user dissatisfaction is expressed in terms of the time spent charging outside the residential garage. A drawback of this scenario is that learning a policy with the proposed approach requires detailed historical data on users' charging behavior, which is difficult to obtain in practice. Furthermore, the GP approach was not compared to a state-of-the-art mixed-integer linear programming (MILP) approach, since the latter cannot be applied if the users' departure times and energy requirements are not available as inputs.

The present work adapts the GP-based approach from [7] to a public charging scenario in which users specify their energy requirements and departure times at the beginning of their charging sessions. In this setting, it is reasonable to express user satisfaction solely in terms of the fulfillment of their energy demands. Under these assumptions, the GP-based approach does not require the collection of detailed data on charging behavior. Instead, it is sufficient to log charging requests over a historical period as training data for learning charging policies. Knowledge of users' energy requirements and departure times further enables the use of MILP to optimize charging plans. Hence, in simulation experiments, the GP-based approach is compared not only with different rule-based controllers but also with MILP-based controllers.

The rest of the paper is organized as follows. Section II details the assumed scenario and problem. Section III outlines the different charging control approaches evaluated on the assumed scenario, including the GP-based approach. Section IV describes the setup of the simulation experiments and presents and discusses their results. Finally, Section V provides a summary and conclusion.

II. Problem

Consider a public EV charging site at which a number N of users/EVs arrive for charging over a time horizon of T discrete timesteps, each of length Δt. Let t_n^{arr}, C_n, $P_n^{\max}$, and B_n^{arr} denote the arrival timestep, the battery capacity, the maximum charging power, and the arrival battery level of user n, respectively. Upon arrival, each user n specifies a departure timestep $t_n^{\mathrm{dep}} \geq t_n^{\mathrm{arr}}$ and a target battery level B_n^{target} with $C_n \geq B_n^{\mathrm{target}} > B_n^{\mathrm{arr}}$. It is assumed that a sufficient number of charging stations are available to serve all users, while the total charging power is limited by P^{lim}. A charging controller sets the charging powers $P_{n,t} \leq P_n^{\max}$ for each user n and

timestep t and ensures that the power limit is respected:

$$\sum_{n \in \mathcal{N}_t} P_{n,t} \leq P^{\lim} \quad \forall t = 1, \ldots, T, \tag{1}$$

where $\mathcal{N}_t$ denotes the set of users connected at timestep t. The departure battery level B_n^{dep} of user n is then given by

$$B_n^{\text{dep}} = \min\{C_n, B_n^{\text{arr}} + \sum_{t=t_n^{\text{arr}}}^{t_n^{\text{dep}}} P_{n,t} \cdot \Delta t\}. \tag{2}$$

The percentage satisfaction S_n of user n's charging request is computed as

$$S_n = 100 \cdot \frac{\min\{B_n^{\text{target}}, B_n^{\text{dep}}\} - B_n^{\text{arr}}}{B_n^{\text{target}} - B_n^{\text{arr}}}. \tag{3}$$

It is assumed that the charging site operator is interested in achieving a high number of users with 100% satisfaction, as such users can be considered more likely to return to the charging site in the future than users who are not satisfied. Furthermore, the operator is assumed to value a fair distribution of energy and wants to avoid situations in which users leave with very low satisfaction of their charging requests.

Ultimately, the problem consists of finding or constructing a charging control, which maximizes the user satisfaction rate and the minimum satisfaction over all users, expressed as the following two objectives:

$$\max f1 = 100 \cdot \frac{|\{1 \leq n \leq N | S_n = 100\}|}{N}, \tag{4}$$

$$\max f2 = \min\{S_1, \ldots, S_N\}. \tag{5}$$

For both objectives, the worst value is zero and the best value is 100.

III. Approaches

A. Rule-based Controllers

The following four manually designed rule-based controllers are considered:

Equal Distribution (ED): At each timestep, the available charging power (or energy) is distributed as evenly as possible among all connected EVs.

Earliest Deadline First (EDF): EVs with earlier departure times are prioritized over those with later departure times. Specifically, as much of the available power as possible is first allocated to the EV with the earliest departure time. Any remaining power is then allocated to the EV with the second-earliest departure time, and so on.

Least Laxity First (LLF): EVs with lower laxity values are preferred over EVs with higher laxities, analogous to the EDF policy. The laxity L_n of an EV n is defined as the maximum number of timesteps by which charging can be delayed while still allowing the EV's energy demand to be satisfied, and is computed as

$$L_n = (t^{\text{dep}} - t^{\text{cur}} + 1) - \frac{\max\{0, B_n^{\text{target}} - B_{t,n}\}}{P_n^{\max} \cdot \Delta t}, \tag{6}$$

where t^{cur} denotes the current timestep and $B_{t,n}$ is the battery level of EV n at the beginning of the current timestep. The first term represents the number of remaining timesteps during which the EV is connected, while the second term corresponds to the number of timesteps required to reach the target battery level when charging at maximum power. Note that the laxity of an EV may be negative, indicating that it is no longer possible to fully satisfy the EV's charging request.

Least Satisfaction First (LSF): EVs with lower percentage satisfaction values S_n are prioritized over those with higher satisfaction levels, analogous to the EDF policy.

B. Online MILP Controllers

Two online controllers that optimize charging plans for connected EVs using mixed-integer linear programming (MILP) are considered. The first controller, referred to as MILP_f1, optimizes plans with the objective of maximizing the number of satisfied users, while the second controller, MILP_f2, optimizes charging plans with the objective of maximizing the minimum satisfaction across all users. Both controllers operate online, meaning that they do not have information about EVs arriving in the future and that the charging plan is updated (i.e., reoptimized) whenever a new EV arrives. The MILP controllers optimize a charging plan for a horizon $\mathcal{T} = \{t^{\text{cur}}, \ldots, t^{\text{cur}} + H\}$ of $H+1$ timesteps, where t^{cur} is the current timestep. The value of H is set so that the end of the planning horizon corresponds to the latest departure time of all currently connected EVs. The MILP_f1 controller optimizes the charging plan by solving the following MILP problem:

$$\max \sum_{n \in \mathcal{N}} S_n \tag{7}$$

subject to:

$$0 \leq P_{n,t} \leq \begin{cases} P_n^{\max} & \text{if } t \leq t_n^{\text{dep}} \\ 0 & \text{else} \end{cases} \qquad \forall n \in \mathcal{N}, \forall t \in \mathcal{T}, \tag{8}$$

$$\sum_{n \in \mathcal{N}} P_{n,t} \leq P^{\lim} \qquad \forall t \in \mathcal{T}, \tag{9}$$

$$B_{n,t} = B_n^{\text{init}} + \sum_{s=t^{\text{cur}}}^{t} P_{n,s} \cdot \Delta t \qquad \forall n \in \mathcal{N}, \forall t \in \mathcal{T}, \tag{10}$$

$$B_{n,t} \leq C_n \qquad \forall n \in \mathcal{N}, \forall t \in \mathcal{T}, \tag{11}$$

$$B_n^{\text{target}} \leq B_{n,t^{\text{cur}}+H} + (1 - S_n) \cdot B_n^{\text{target}} \qquad \forall n \in \mathcal{N}, \tag{12}$$

$$S_n \in \{0, 1\} \qquad \forall n \in \mathcal{N}, \tag{13}$$

where $\mathcal{N}$ is the set of currently connected EVs, B_n^{init} is the battery level of EV n at the beginning of the current timestep, and S_n is a binary variable that is set to one if the charging request of EV n is fully satisfied. The objective is to maximize the number of satisfied EVs. Constraint (8) ensures that the EVs' maximum charging powers are respected and that they are not charged after departure. Constraint (9) limits the total charging power in each timestep. Constraints (10) and (11) set the battery levels of the EVs across all timesteps and ensure that they do not exceed the respective battery capacities. Constraint (12) enforces that the binary variable S_n is set to

zero if the battery level of EV n at the end of the planning horizon is below its target battery level B_n^{target}.

The MILP_f2 controller solves an analogous problem to maximize the minimum percentage satisfaction across all connected EVs:

$$\max S^{\min} \tag{14}$$

subject to:

$$S^{\min} \leq \frac{B_{n,t^{\text{cur}}+H} - B_n^{\text{arr}}}{B_n^{\text{target}} - B_n^{\text{arr}}} \qquad \forall n \in \mathcal{N}, \tag{15}$$

$(8) - (12)$.

C. Genetic Programming-based Control

Genetic programming (GP) [9] is a special type of evolutionary algorithm that evolves formulas. The idea of the GP-based approach is to evolve a formula based on features of an EV, such as its state of charge (SOC) or maximum charging power, which is then used for charging control, as illustrated in Figure 1.

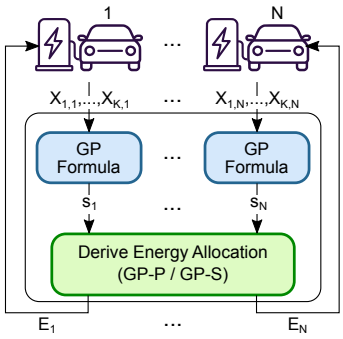


Fig. 1: Charging control based on formula evolved with GP.

In each timestep, a score s_n is computed for each connected EV n using the GP formula based on K EV features $X_{1,n}, \ldots, X_{K,n}$. The amounts of energy $E_1, \ldots, E_N$ allocated to the EVs are then derived from their respective scores. The GP formula is evolved using historical data from the charging site, assuming that records of the charging requests over a given historical time period are available. To evaluate a formula, the recorded data are used to simulate charging over the historical time period and to compute the corresponding values of the objective functions $f1$ and $f2$. Since these objectives are conflicting, multi-objective GP is employed to compute a Pareto set approximation consisting of mutually non-dominating solutions. In practice, the charging station operator would select a trade-off solution from this Pareto set, which is then implemented as the charging control for future operation of the charging site.

Two variants of the GP-based approach are evaluated, referred to as GP-P and GP-S in the following. They differ in how the scores computed by a GP formula are used to derive charging powers of connected EVs (green box in Fig. 1). In the GP-P variant, the scores are interpreted as priorities: the EV with the highest score is allocated as much of the available energy as possible. Any remaining energy, is then allocated, again as much as possible, to the EV with the second-highest score, and so on, until all available energy has been distributed or energy has been allocated to all connected EVs. In the GP-S variant, the scores are normalized and interpreted as shares of the available energy allocated to the corresponding EVs. An EV whose charging request is not yet fully satisfied is allocated a share of the available energy proportional to its normalized score. A more detailed description of the GP-based approach can be found in [7].

IV. Simulation Experiments

A. Simulation Setup

In the experiments, the approaches described in the previous section are evaluated in a simulation set up as follows. A time period of 400 days is simulated with a temporal resolution of $\Delta t = 0.25$ hours. The first 200 days are used as training data for learning the controllers with the GP-based approach, while the remaining 200 days are used as test data to evaluate whether the GP-based approach generalizes to unseen data. For each day d, the number N_d of EVs arriving on that day is sampled uniformly at random from $[80, 120]$. For each EV n, the maximum charging power $P_n^{\max}$, the battery capacity C_n, and the arrival battery level B_n^{arr} are uniformly sampled from the intervals specified in Table I. The power limit is set to 100 kW in the simulation.

TABLE I: Setting of parameters in the simulation.

Parameter	N_d	$P_n^{\max}$	C_n	B_n^{arr}	P^{lim}
Unit	-	kW	kWh	kWh	kW
Value	$[80, 120]$	$[7, 22]$	$[40, 64]$	$[0.1, 0.5] \cdot C_n$	100

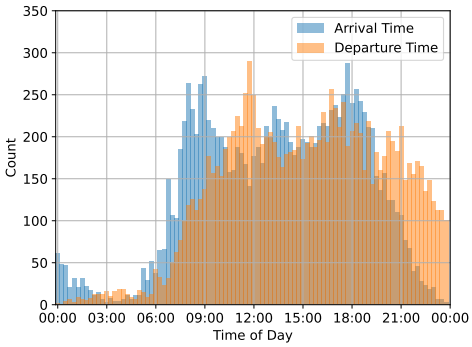
The setting of arrival and departure times is based on a real-world charging dataset [10]. The dataset contains information on 72,856 charging sessions across locations of different types (apartment, company, public area, etc.). For the simulation, all charging sessions at the public area location that start and end on the same day are used, resulting in 12,360 sessions. The start and end times of these sessions are converted into the corresponding arrival and departure timesteps within the day. The resulting distributions of arrival, departure and sojourn times are shown in Figure 2. For each EV n in the simulation, a charging session is selected uniformly at random from the dataset, and the EV's arrival and departure timesteps on the day of arrival are set according to the selected session. The target battery level B_n^{target} of EV n is set to

$$B_n^{\text{target}} = \min\{SOC^{\text{target}} \cdot C_n, B_n^{\text{arr}} + \sum_{t=t_n^{\text{arr}}}^{t_n^{\text{dep}}} P_n^{\max} \cdot \Delta t\}, \tag{16}$$

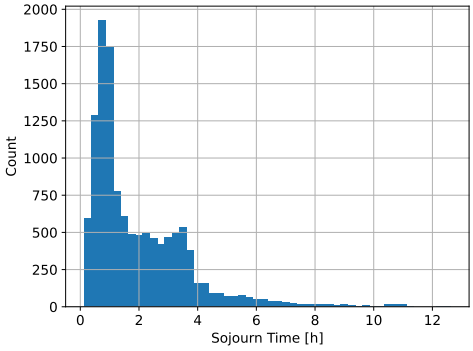
where SOC^{target} is sampled uniformly at random from $[0.6, 1.0]$. This ensures that satisfaction of the charging request is technically feasible.

B. Implementation and Parameters of the GP-based Approach

The GP algorithm is implemented using the Python libraries DEAP [11] and pymoo [12], employing selection operators



(a)



(b)

Fig. 2: Histograms of (a) daily arrival and departure timesteps and (b) sojourn times of the 12,360 charging sessions used for the simulation.

from the well-known Non-dominated Sorting Genetic Algorithm II (NSGA-II) [13] for multi-objective optimization. The population size is set to 100, and 1000 generations are executed in each optimization run. The mutation and crossover rates are set to 0.2 and 0.9, respectively. The depth and size of GP trees are limited to a maximum of 10 and 50, respectively. As outlined in Section III-C, the formulas evolved through GP are used to compute, per timestep, a score for each connected EV n in the charging control. The GP optimization can use the following seven features as input variables of a formula:

- X_1: The number of timesteps until departure of EV n, computed as $t_n^{\mathrm{dep}} - t^{\mathrm{cur}} + 1$, where t^{cur} denotes the current timestep.
- X_2: The energy missing to reach the target battery level, given by $B_n^{\mathrm{target}} - B_n$, where B_n is the current battery level of EV n.
- X_3: The maximum charging power $P_n^{\max}$ of EV n.
- X_4: The percentage satisfaction of EV n's charging request, computed as $(B_n^{\mathrm{target}} - B_n + 0.0001)/(B_n^{\mathrm{target}} - B_n^{\mathrm{arr}})$. A small constant is added to the numerator to avoid the variable becoming zero, which could lead to division by zero in a GP formula.
- X_5: The laxity of EV n as defined in (6).
- X_6: The current timestep within the current day.
- X_7: The number of currently connected EVs whose charging requirements are not yet fully satisfied.

These features were chosen based on results of preliminary experiments. In addition to input variables, GP formulas may contain constant values from the interval $[-5, 5]$ and arithmetic operators from the set $\{+, -, \times, /\}$.

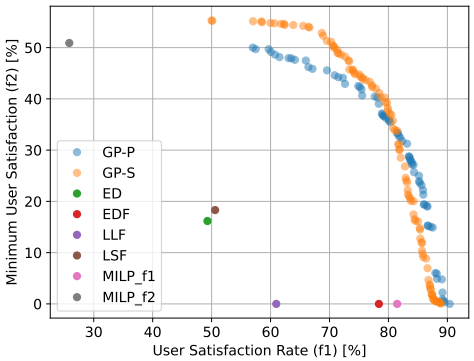
C. Further Experimental Settings

All experiments are run on an Intel Core i5-13400 with a clock speed of 4.6 GHz and 16 CPU cores. Version 12.0.2 of the Gurobi solver [14] is used for the MILP-based controllers. Since GP is a stochastic optimization algorithm, 21 trials are executed for each of the two GP-based variants, using 21 different random number generator seeds.

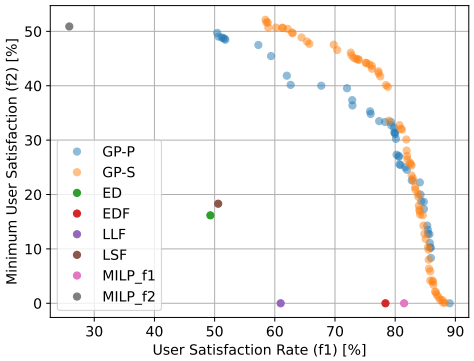
D. Experimental Results

One trial of the GP optimization took approximately 18 minutes. Figure 3 shows the objective values obtained by applying the different control approaches to the 200 days of training data. For the GP-based variants, the Pareto front approximations resulting from the optimization on the training data are shown. More precisely, Figure 3a presents the fronts combined over the 21 trials with each variant, i.e., the mutually non-dominating solutions in the union of the fronts obtained from the 21 trials. Figure 3b shows, for each GP-based variant, the front yielding the median hypervolume over the 21 trials. Among the manually designed rule-based controllers, the EDF controller achieves the highest user satisfaction rate, and the LSF controller yields the highest minimum user satisfaction. As expected, the two MILP-based controllers achieve better objective values than the rule-based controllers, but each of them is only good in one objective and poor in the other one. A combination of the two MILP-based controllers, which optimizes a weighted sum of the objectives (7) and (14), might yield better tradeoff solutions, but this requires deciding on a reasonable setting of the weights, which is not straightforward. The GP-based approaches yield even better objective values than the MILP-based controllers and provide a wide range of tradeoff solutions from which an operator can select. Note that although the MILP controllers solve all their subproblems to optimality, they have no information about the future and thus their results are not perfect over the whole time horizon. The GP-based approach is especially beneficial with respect to the second objective $f2$—the minimum user satisfaction. Compared to MILP_f2, it finds solutions that are not only superior in $f2$ but also notably better in the first objective $f1$. A comparison of the two GP-based variants shows that the GP-P variant tends to find solutions that perform better in $f1$, but that the GP-S variant finds solutions that perform better in $f2$. This is likely because the GP-S variant distributes the available energy more evenly among connected EVs than the GP-P variant, which prioritizes some EVs over others.

Further details on the results of the MILP-based controllers and the GP-based variants on the training data are provided in Table II. The table reports the objective values achieved by the MILP-based controllers as well as those corresponding to the extreme points of the combined fronts over the 21 trials

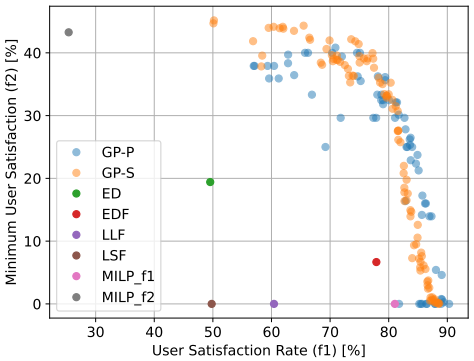


(a)

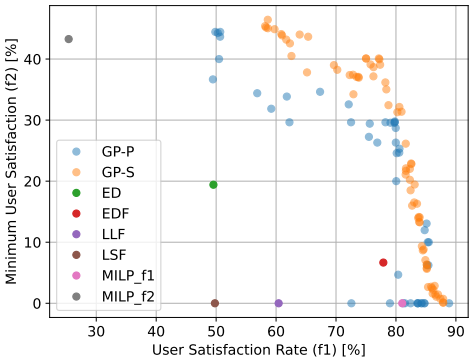


(b)

Fig. 3: Results with the different approaches on the 200 days training data. For the two variants of the GP-based approach, the Pareto front approximation combined over all 21 runs is shown in (a) and the Pareto front approximation resulting from the runs that yielded the median hypervolume is shown in (b).



(a)



(b)

Fig. 4: Results with the different approaches on the 200 days test data analogous Figure 3.

TABLE II: Objective values achieved with the two MILP-based controllers and the extreme points of the combined fronts of the two GP-based approaches on the training data. The best values are highlighted in bold.

Approach	User Satisfaction Rate ($f1$)	Minimum User Satisfaction ($f2$)
MILP_f1	81.48 %	0.00 %
MILP_f2	25.85 %	50.90 %
GP-P_f1	**90.41** %	0.00 %
GP-P_f2	56.99 %	50.00 %
GP-S_f1	88.87 %	0.06 %
GP-S_f2	49.99 %	**55.31** %

for the two GP-based variants. For example, GP-P_f1 denotes the solution in the combined front resulting from the GP-P variant that is best in the objective function $f1$.

Figure 4 shows the results obtained with the different approaches on the 200 days of test data, analogous to Figure 3. On the test data, the LSF controller no longer yields the highest minimum user satisfaction among the rule-based controllers; in fact, it results in a minimum user satisfaction of zero. This may appear surprising, as the LSF controller prioritizes EVs with low satisfaction over EVs with high satisfaction. However, when multiple EVs have a satisfaction of zero, the controller prioritizes them in a random order, without taking into account factors such as the remaining connection times of the EVs.

The fronts computed by the GP-based variants are no longer strictly non-dominated on the test data, i.e., they contain solutions that dominate others within the same set. This is not surprising, as the fronts are determined based on the training data. Compared to the training results, a drop in the performance of the GP-based variants can be observed, particularly for the GP-P variant in objective $f2$. Nevertheless, the GP-based approach still achieves better objective values than the MILP-based approaches and clearly outperforms the manually designed rule-based controllers. This is further supported by the detailed results shown in Table III.

Note that the results of the GP-based variants shown in Table III correspond to the extreme points of the fronts obtained on the training data, which are not necessarily extreme points on the test data. Overall, the results indicate that the GP-based approaches generalize reasonably well to the test data.

Table IV shows, for the two GP-based variants, how many formulas in the Pareto set approximations computed in the 21 trials contain different input variables. From this, it can be concluded that variables X_1 (remaining timesteps until departure), X_2 (missing energy), X_4 (percentage satisfaction), and X_5 (laxity) are the most important, whereas variables X_3

TABLE III: Objective values achieved with the two MILP-based controllers and the two GP-based approaches on the test data analogous Table II.

Approach	User Satisfaction Rate ($f1$)	Minimum User Satisfaction ($f2$)
MILP_f1	81.05 %	0.00 %
MILP_f2	25.40 %	43.27 %
GP-P_f1	**90.31** %	0.00 %
GP-P_f2	56.97 %	37.91 %
GP-S_f1	88.50 %	0.02 %
GP-S_f2	50.10 %	**44.70** %

TABLE IV: Statistics of unique formulas in the Pareto set approximations computed over 21 trials with GP-P and GP-S.

	GP-P	GP-S
No. of Formulas, Total	1113	1552
–, Containing X_1	1044 (93.80 %)	1529 (98.52 %)
–, Containing X_2	1036 (93.08 %)	1487 (95.81 %)
–, Containing X_3	826 (74.21 %)	1185 (76.35 %)
–, Containing X_4	1113 (100.00 %)	1531 (98.65 %)
–, Containing X_5	1020 (91.64 %)	1538 (99.10 %)
–, Containing X_6	807 (72.51 %)	1035 (66.69 %)
–, Containing X_7	782 (70.26 %)	1160 (74.74 %)

(maximum charging power), X_6 (current timestep), and X_7 (number of connected and not fully satisfied EVs) appear to be less important.

The following GP formula yielded the best value for the objective $f1$ on the training data (corresponding to GP-P_f1 in Tables II and III):

$$\frac{X_6(X_6 - X_2)(X_1 X_3^2 X_4 + 2X_5 - 1.459))}{X_1^2 X_3 (3.294 X_1 X_3 X_5 + X_4 - X_5)}. \tag{17}$$

The GP formula yielding the best value for $f2$ on the training data (corresponding to GP-S_f2 in Tables II and III) is

$$\frac{(X_2 - X_5)(X_4 - 0.404)}{X_1 \left(X_1 + \frac{X_6 + \frac{X_5 X_7 + X_6 / X_2}{X_3 X_4 - X_4}}{-X_1 + X_4 X_6 + X_4 X_7} \right)}. \tag{18}$$

The formulas are highly complex and hard to interpret. It is possible to reduce the complexity of the GP formulas by reducing the limit for the GP tree size (set to 50 in the experiments); however, this also leads to reduced performance.

V. Summary and Conclusion

The present work evaluated an approach for learning fair and efficient charging policies through multi-objective genetic programming (GP) and compared it to various rule-based and mixed-integer linear programming (MILP)-based controllers in a public charging scenario. Two objectives were considered: maximizing the user satisfaction rate and maximizing the minimum satisfaction across all users. Two variants of the GP-based approach, GP-P and GP-S, were evaluated, which differ in how the formulas evolved with GP are used for the actual control. The evaluation was performed on both the training data used for GP optimization and on unseen test data. The simulation experiments showed that the GP-based approach is able to learn policies that outperform all other considered control approaches on the training data. The learned policies also generalize reasonably well to the unseen test data and stay competitive with the MILP-controllers, although a slight performance drop compared to the training data can be observed. A comparison of the two GP-based variants showed that the GP-P variant is superior in terms of the user satisfaction rate, while the GP-S approach finds policies with a higher minimum satisfaction across the users. In practice, optimization could be done using both approaches, and the resulting Pareto front approximations could be combined to provide a set of options for a decision maker to select from. Compared to the MILP-based controllers, the GP-based approach has the additional advantage of yielding a relatively simple rule-based control, which is less computationally demanding than solving MILP optimizations.

As future work, it is planned to evaluate the approach on more complex scenarios, for example by considering a time-varying power limit or a limitation of the number of available charging stations. Furthermore, future work will investigate techniques for incorporating user feedback into the decision-making process for selecting a solution from the computed Pareto front approximation.

References

[1] J. Soares, F. Lezama, R. Faia, S. Limmer, M. Dietrich, T. Rodemann, S. Ramos, and Z. Vale, "Review on fairness in local energy systems," *Applied Energy*, vol. 374, p. 123933, 2024.

[2] S. Ø. Ottesen, "Techno-economic models in smart grids: Demand side flexibility optimization for bidding and scheduling problems," Ph.D. dissertation, Norwegian University of Science and Technology, 3 2017.

[3] O. Frendo, N. Gaertner, and H. Stuckenschmidt, "Real-time smart charging based on precomputed schedules," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 6921–6932, 2019.

[4] T. Panayiotou, M. Mavrovouniotis, and G. Ellinas, "On the fair-efficient charging scheduling of electric vehicles in parking structures," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 1627–1634.

[5] M. Zeballos, A. Ferragut, and F. Paganini, "Achieving fairness for EV charging in overload: A fluid approach," *ACM SIGMETRICS Performance Evaluation Review*, vol. 46, no. 3, pp. 62–67, 2019.

[6] D. Danner and H. de Meer, "Quality of service and fairness for electric vehicle charging as a service," *Energy Informatics*, vol. 4, pp. 1–20, 2021.

[7] S. Limmer, A. Kenny, T. Ray, F. Lanfermann, H. K. Singh, and A. Castellani, "Design of fair and interpretable electric vehicle charging policies through genetic programming," *Applied Energy*, vol. 404, p. 127176, 2026.

[8] S. Limmer and T. Rodemann, "Combination of charging policies for fair and efficient EV charging under limited capacity," in *2024 22nd International Conference on Intelligent Systems Applications to Power Systems (ISAP)*, 2024, pp. 1–6.

[9] J. R. Koza, "Genetic programming: On the programming of computers by means of natural selection," *MIT Press*, 1992.

[10] K. Baek, E. Lee, and J. Kim, "A dataset for multi-faceted analysis of electric vehicle charging transactions," *Scientific Data*, vol. 11, no. 1, p. 262, 2024.

[11] F.-A. Fortin, F.-M. De Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary algorithms made easy," *Journal of Machine Learning Research*, vol. 13, pp. 2171–2175, jul 2012.

[12] J. Blank and K. Deb, "pymoo: Multi-objective optimization in Python," *IEEE Access*, vol. 8, pp. 89 497–89 509, 2020.

[13] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[14] Gurobi Optimization, LLC, "Gurobi optimizer reference manual," 2025. [Online]. Available: https://www.gurobi.com