Online electric vehicle charging scheduling with commitment using surrogate-assisted optimization

1st Abdennour Azerine
IRIMAS UR 7499
Université de Haute-Alsace
F-68100 Mulhouse, France
abdennour.azerine@uha.fr

2nd Mahmoud Golabi
IRIMAS UR 7499
Université de Haute-Alsace
F-68100 Mulhouse, France
mahmoud.golabi@uha.fr

3rd Lhassane Idoumghar
IRIMAS UR 7499
Université de Haute-Alsace
F-68100 Mulhouse, France
lhassane.idoumghar@uha.fr

***Abstract*—The Electric Vehicle Charging Scheduling Problem (EVCSP) concerns the allocation of limited charging resources to heterogeneous vehicle requests under temporal and capacity constraints. Although advanced optimization models have been proposed for this problem, most assume an offline setting with complete knowledge of future arrivals, limiting their applicability in real charging stations. This paper introduces an online, preemptive EVCSP that preserves a high-fidelity problem formulation, including charger availability and grid capacity constraints, while enforcing a commitment policy under which admission decisions are irrevocable. We propose a hybrid solution approach that combines a genetic algorithm for online admission control and charger assignment with a mathematical optimization model for energy allocation. To meet real-time decision-making requirements, we embed a Gaussian-process surrogate within the evolutionary search to reduce the cost of repeatedly evaluating candidate schedules. Computational experiments across multiple instance sizes and booking modes show that surrogate assistance preserves objective values relative to exact evaluation while yielding statistically significant reductions in computational time, supporting practical online deployment of evolutionary optimization for real-time EV charging station operations.**

***Index Terms*—Online charging scheduling problem, Hybrid Genetic algorithm, Linear programming, Gaussian process, Surrogate modeling**

I. Introduction

The global transition to electric vehicles (EVs) is widely recognized as a key component of sustainable transportation and energy planning. This transition, driven by environmental imperatives and technological advances in batteries and powertrains, places increasing pressure on charging infrastructure [1]. A critical challenge has emerged as the deployment of public charging stations fails to keep pace with the rapid growth of the EV fleet, creating congestion and operational bottlenecks that threaten to slow widespread adoption [2]. In order to maximize the utilization of existing infrastructure and ensure reliable service, efficient operational management of charging stations is essential. This need is formalized as the Electric Vehicle Charging Scheduling Problem (EVCSP), which consists in optimizing the allocation of limited charging resources to a set of vehicles with heterogeneous energy demands, subject to power capacity limits, temporal constraints, and charger availability [3].

979-8-3315-3431-8/25/$31.00 ©2025 IEEE

Early research established the EVCSP as an $\mathcal{NP}$-hard problem [4], commonly modeled as a combinatorial problem with objectives such as cost minimization or peak shaving [5]. As a result, scalable metaheuristic approaches became the primary solution paradigm for practical problem sizes [1]. A significant modeling advancement in this line of work is the introduction of preemptive scheduling, which allows charging sessions to be interrupted and resumed over time [6]. Preemptive models provide increased flexibility for load management and have been shown to improve resource utilization when compared to non-preemptive formulations [7].

The evolution of optimization objectives further refined the scope of the EVCSP. Moving beyond aggregate economic metrics, researchers increasingly emphasized performance indicators directly related to service quality. Zaidi et al. [8] proposed a preemptive scheduling model aimed at minimizing the total shortfall between requested and delivered state of charge, thereby explicitly capturing user satisfaction. Subsequent studies focused on improving the scalability of this formulation through advanced metaheuristic designs [9] and surrogate modeling techniques [10]. A comprehensive and practically relevant formulation is presented in [4], where demand satisfaction is defined as the complete fulfillment of requested energy, focusing on maximizing the number of fully satisfied charging requests. This work integrates a linear programming formulation with heuristic and metaheuristic solution techniques and captures the essential trade-offs between temporal, energy, and capacity constraints encountered in real charging stations. The results underline both the robustness of the model and the substantial computational challenges associated with solving it at scale.

A core limitation of this body of work is its reliance on an offline paradigm that assumes complete *a priori* knowledge of all charging requests [11]. In operational environments, however, charging stations face dynamic conditions characterized by vehicle arrivals over time, uncertain dwell durations, and evolving system states [12]. While offline models provide valuable performance benchmarks and structural insights, their reliance on complete future information limits their direct applicability to real-time decision making in uncertain and dynamic environments. This observation has motivated extensive research on the online EVCSP, in which scheduling decisions must be taken sequentially without access to future information [13].

Research on online EVCSP has followed several methodological streams. One stream addresses uncertainty through stochastic or robust optimization, formulating charging decisions using horizon-based models that hedge against uncertain arrivals or demand [14]. While such approaches explicitly account for uncertainty, they typically rely on full-horizon planning or repeated rescheduling, which limits their suitability for strictly online settings with irrevocable decisions [15]. Another line of work applies classical online scheduling algorithms such as earliest-deadline-first (EDF), least-laxity-first (LLF), and optimal-available (OA) [16]. However, these policies generally perform poorly for EV charging applications because they ignore the nonlinear coupling between energy delivery and time constraints [13]. Reinforcement learning methods, including deep and multi-agent approaches, have also been proposed to learn real-time control policies for grid load management [17], charger coordination [18], and dynamic pricing [19]. Despite their flexibility, reinforcement learning approaches suffer from well-known limitations, including high sample complexity, lack of formal guarantees on constraint satisfaction, and limited interpretability [20], [21].

A systematic analysis of the literature reveals a persistent methodological divide. On the one hand, high fidelity offline formulations exemplified by [4] accurately capture the operational complexity of EV charging stations but do not account for real-time decision making. On the other hand, existing online approaches provide mechanisms for sequential decision making but are predominantly developed for simplified models. In order to ensure low computational latency or analytical tractability, many online studies rely on significant simplifications, such as idealized charging rates [22] or abstracted capacity constraints [23]. As a result, there is a lack of online scheduling algorithms that preserve the comprehensive problem definition of state-of-the-art optimization models while remaining computationally feasible in real time.

Our paper addresses this gap by proposing an online extension of the high-fidelity demand-maximizing, preemptive EVCSP introduced by Zaidi et al. [4]. The online setting considered here captures the sequential arrival of charging requests over time, where newly arriving demands must be scheduled in the presence of previously admitted but unfinished charging tasks. To represent this operational context, we additionally incorporate a commitment policy, meaning that once a charging request is admitted, the admission decision is not revised in subsequent decision epochs. This commitment requirement distinguishes the problem from rolling-horizon rescheduling schemes and reflects an important operational constraint faced by real charging stations.

To solve the resulting online decision problem, we develop a hybrid solution approach that combines metaheuristic search with mathematical optimization. A genetic algorithm is used to determine admission decisions and charger assignments for active charging requests, while the corresponding energy allocation is computed by a commitment-aware optimization model developed in this study. Since repeatedly solving this model can be computationally demanding in real-time operation, surrogate models are employed to accelerate the evaluation of candidate solutions within the genetic algorithm. This combination enables effective online decision making while retaining the key structural features of the underlying demand-maximizing EVCSP.

The remainder of the paper is structured as follows. Section II formally defines the studied problem. Section III details the proposed hybrid solution methodology. Section IV reports the computational setup and numerical results. Finally, Section V concludes the paper and outlines directions for future research.

II. Problem Definition

The problem addressed in this study is an online extension of the preemptive EVCSP originally introduced by Zaidi et al. [4]. The objective is demand-oriented and consists of maximizing the number of charging requests that are completely satisfied within their announced availability intervals, while respecting charger availability and grid capacity constraints. While the original formulation assumes that all charging demands are known *a priori*, the present study considers an online variant in which charging demands are revealed progressively over time. The underlying problem structure is retained, but decisions must be taken sequentially as new demands arrive, without revising prior admission commitments. As shown in [4], the offline version of this problem is NP-hard; the online setting preserves this inherent combinatorial difficulty and further complicates decision-making by requiring feasible and timely actions under partial information.

The charging station is equipped with a finite set of chargers denoted by $I = \{1, \ldots, m\}$. Each charger $i \in I$ delivers energy at a constant rate w_i (kW). Charging operations are further constrained by a global grid capacity limit W^g (kW), which bounds the total charging power that can be drawn by all chargers simultaneously. This system-level constraint induces coupling across charger assignments and prevents all chargers from operating at full capacity at the same time.

Let $J = \{1, \ldots, n\}$ denote the set of all charging demands over a discrete scheduling horizon $\mathcal{T} = \{1, \ldots, T\}$. Each demand $j \in J$ is characterized at submission by a submission time $st_j \in \mathcal{T}$, an arrival time $r_j \in \mathcal{T}$, a departure time $d_j \in \mathcal{T}$, and a required energy amount e_j (kWh). The uninterrupted charging time required for demand j on charger i, denoted as p_{ij}, is calculated as $p_{ij} = \frac{e_j}{w_i}$. Upon submission, an admission decision must be made. The binary decision variable x_j models the admission of demand j, taking the value 1 if the demand is accepted and 0 otherwise. Accordingly, the objective function can be formulated as:

$$\max \sum_{j \in J} x_j. \tag{1}$$

If demand j is accepted, it is assigned to exactly one charger, denoted by $c_j \in I$. In the setting considered here, charging begins exactly at the arrival time r_j, and the vehicle remains connected until its departure time d_j. Charging is preemptive, meaning that energy delivery may be interrupted and resumed during the interval $[r_j, d_j)$, provided that the full energy requirement e_j is delivered by time d_j. Charging demands cannot be partially fulfilled; each demand must be either entirely satisfied within its announced time interval or otherwise declined entirely.

Feasibility requires that charger-level and system-level capacity constraints are satisfied at all times. In particular, each charger can serve at most one demand at any time. In addition, the total charging power drawn by all demands receiving energy at time t must not exceed the grid capacity:

$$\sum_{j \in J | r_j \leq t < d_j} w_{c_j} \leq W^g, \quad \forall t \in \mathcal{T}, \tag{2}$$

where $|\mathcal{T}| = \max_{j \in J}\{d_j\}$. Demands that are rejected do not consume any charging resource.

The online problem enforces commitment through a state-based model for each demand. Upon submission, a demand is immediately either rejected or accepted. If accepted, the demand enters a `WAITING` state, representing a future reservation that has not yet arrived at the station. During the `WAITING` period, admission is irrevocable, but the charger assignment remains provisional and may be updated until the vehicle arrives. Once the demand reaches its arrival time r_j, it transitions to the `CHARGING` state. From this moment onward, both the admission and the final charger assignment become binding and cannot be altered. Charging proceeds, potentially with preemption, until the required energy is fully delivered. After the demand is completed by time d_j, it transitions to a final `COMPLETED` state and releases its assigned charger.

III. Solution Method

This section introduces the proposed hybrid solution approach for the online EVCSP with commitment, charger availability, and grid-capacity constraints. The approach is structured in two tightly coupled levels. At the first level, admission decisions are modeled implicitly through charger assignment: assigning a demand to a charger represents acceptance, whereas leaving it unassigned represents rejection. This level yields a candidate set of accepted demands, but the resulting assignment encoding does not directly guarantee satisfaction of commitment requirements or charging-station constraints.

Feasibility is addressed at the second level. Given a candidate admission and assignment decision, commitment-related requirements are first enforced through dedicated repair procedures. The resulting solution is then evaluated using an exact mathematical optimization model that determines whether the accepted demands can be fully served within their time windows while respecting charger availability and grid-capacity constraints, and computes the corresponding energy allocation.

The first-level problem is combinatorial in nature, and its feasibility depends on constraints that cannot be enforced directly within the assignment representation. For this reason, it is solved using a genetic algorithm (GA), which is well suited to constrained combinatorial problems where feasibility is handled through repair mechanisms and external evaluation rather than embedded in the encoding [24]. This choice is further supported by prior studies reporting strong empirical performance of GA-based approaches on EVCSP variants [25]. Since evaluating each candidate solution requires solving the associated optimization model, which is computationally expensive in an online setting, a surrogate-assisted evaluation is incorporated to reduce computational overhead and enable real-time decision making. The following subsections detail the representation, repair mechanisms, and the exact and surrogate-assisted evaluation components.

A. Chromosome Encoding

At a given submission epoch, let A denote the set of demands that have been accepted previously but have not yet arrived. These demands are subject to commitment and must remain assigned to a charger at the current epoch. Let B denote the set of newly submitted demands revealed at the same epoch. An ordered list is constructed by concatenating (A, B), and the decision at the current epoch is represented by an integer chromosome $H = (H_1, \ldots, H_{|A|+|B|})$.

Each gene $H_k \in H$ corresponds to the k-th demand in the ordered list and takes a value in $\{0, 1, \ldots, m\}$, where m denotes the number of available chargers. If $H_k = i$ for some $i \in \{1, \ldots, m\}$, then the k-th demand is assigned to charger i, whereas $H_k = 0$ indicates that the demand is not assigned to any charger and is deemed rejected. Since demands in A are committed, their associated genes are restricted to values in $\{1, \ldots, m\}$. In contrast, demands in B may either be rejected or assigned to any of the m chargers.

This chromosome representation encodes the epoch-level assignment decisions optimized by the GA. For a given chromosome H, an exact mathematical model evaluates the corresponding preemptive charging schedule and verifies feasibility with respect to charger-level and grid-capacity constraints.

B. GA Framework for Epoch-level Decision Making

At each decision epoch, a generational genetic algorithm is applied to explore the space of candidate assignment vectors. Multiple alternatives are considered for the main GA operators. Parent selection is performed using roulette-wheel, rank-based, or tournament selection. Recombination is implemented via one-point, two-point, or uniform crossover, while mutation is realized through random-reset or swap operations. In addition, elitist replacement strategies and alternative stopping criteria are examined. All operators follow standard implementations for genetic algorithms [26]. The final configuration of operators and parameter values is determined empirically through extensive preliminary experimentation, with the objective of balancing solution quality and computational efficiency under the constraints of online decision making.

To ensure commitment compliance and basic assignment consistency, each newly generated individual is processed by the repair procedure described in Section III-C before being

Algorithm 1 Online GA at a decision epoch (high-level)

Require: Epoch data (sets A, B; residual capacity $\bar{W}$), GA parameters

Ensure: Committed admissions/assignments and an exact feasible charging plan

1: Initialize population $\mathcal{P}$ of chromosomes H of length $|A| + |B|$
2: Ensure mandatory genes (for A) take values in $\{1, \ldots, m\}$
3: Repair each chromosome to preserve commitments and basic feasibility
4: Evaluate individuals (exactly or via surrogate-assisted policy)
5: **for** $g \leftarrow 1$ to G **do**
6: Select parents from $\mathcal{P}$
7: Generate offspring via crossover and mutation
8: Repair offspring
9: Evaluate offspring
10: Update $\mathcal{P}$ using replacement and elitism
11: **end for**
12: **return** Best epoch decision; validate by exact evaluation before commitment

evaluated by either the exact or surrogate-assisted evaluator presented in the subsequent subsections. Algorithm 1 outlines the resulting evolutionary loop executed at each decision epoch.

C. Feasibility repair and commitment handling

Chromosomes generated by the genetic algorithm may be infeasible, mainly due to commitment requirements for previously accepted demands and overlap conflicts created by assigning demands with intersecting availability windows to the same charger. To prevent such violations, we apply an epoch-level repair procedure. Repair is applied once after population initialization and again after each variation step, before performing exact feasibility evaluation.

For each committed demand $j \in A$, let $h_j^{\text{sch}} \in I$ denote its charger assignment in the previously committed schedule. Since demands in A cannot be rejected, their genes must take values in $\{1, \ldots, m\}$. Accordingly, if the gene of a committed demand takes the rejection value 0 or if its current assignment creates an overlap conflict, the gene is reassigned to $h_j^{\text{sch}} \in \{1, \ldots, m\}$. This rollback operation enforces commitment by ensuring that previously accepted demands retain a valid charger assignment across decision epochs.

In contrast, demands revealed at the current epoch, i.e., $j \in B$, are not yet committed and may be rejected. Repair for these demands starts by setting all corresponding genes to the rejection value 0. The demands in B are then processed in a randomized order and are tentatively assigned to their proposed chargers whenever this does not introduce an overlap conflict with demands that are already assigned. If a conflict occurs, the demand is left at 0 and is therefore rejected.

D. Exact evaluation via a commitment-aware charging model

Given a candidate chromosome H generated by the genetic algorithm, charger assignments are fixed for all demands that are not rejected. Let $h_j \in I$ denote the charger assigned to demand j under H. The purpose of the exact evaluation step is to determine whether these discrete assignment decisions admit a feasible preemptive charging schedule that satisfies commitment requirements and grid-capacity constraints at the current decision epoch.

The time horizon is discretized into slots of duration τ, and $\mathcal{T}$ denotes the resulting set of time slots. For each demand j, the corresponding charging window is defined as $\mathcal{T}_j = \{t \in \mathcal{T} \mid r_j \leq t < d_j\}$. The binary variable $y_{j,t}$ indicates whether demand j is charged during time slot $t \in \mathcal{T}_j$. As defined earlier, $x_j \in \{0, 1\}$ represents the acceptance decision for demand j. In the online setting, demands in A are already accepted and therefore have fixed acceptance status, whereas acceptance decisions remain to be determined for demands in B.

Let $\bar{W}_t$ denote the residual grid capacity available at time slot t after accounting for charging load that has already been committed at the current decision epoch. Exact evaluation is performed by solving the following binary linear program. The model enforces full service of all previously accepted demands and determines which newly arrived demands can be accepted, under the charger assignments specified by H, while respecting grid-capacity constraints:

$$\max \quad \sum_{j \in B} x_j \tag{3}$$

$$\text{s.t.} \quad \sum_{t \in \mathcal{T}_j} y_{j,t} = p_{h_j, j}, \qquad \forall j \in A, \tag{4}$$

$$\sum_{t \in \mathcal{T}_j} y_{j,t} = p_{h_j, j}\, x_j, \qquad \forall j \in B, \tag{5}$$

$$\sum_{j \in A \cup B} w_{h_j}\, y_{j,t} \leq \bar{W}_t, \quad \forall t \in \mathcal{T}, \tag{6}$$

$$x_j \in \{0, 1\}, \qquad \forall j \in B, \tag{7}$$

$$y_{j,t} \in \{0, 1\}, \qquad \forall j \in A \cup B,\ \forall t \in \mathcal{T}_j \tag{8}$$

Constraint (4) ensures that all demands in the committed set A are fully served. Constraint (5) links acceptance decisions to energy delivery, so that a newly arrived demand is fully charged if and only if it is accepted. Constraint (6) enforces the grid-capacity limit at each time slot based on the residual capacity available at the current epoch.

If the model is feasible, the resulting objective value defines the exact marginal contribution $f(H)$ of the candidate chromosome at the current decision epoch. If the model is infeasible, the corresponding assignment decision is discarded by the evaluator.

E. Surrogate-assisted evaluation

Exact evaluation of a candidate chromosome requires solving the commitment-aware mathematical model. Since this model is invoked repeatedly within the epoch-level genetic algorithm, exact evaluation can become the dominant contributor to computational cost. To mitigate this overhead, we adopt a surrogate-assisted evolutionary computation strategy, in which a learned regression model $\hat{f}(H)$ is used to approximate the exact fitness value $f(H)$ for most candidate chromosomes. Exact evaluations are still performed selectively to control approximation error and to update the surrogate in an online manner.

In the considered online setting, exact evaluations are generated incrementally during the evolutionary search and are therefore scarce, especially in early epochs. This motivates surrogate models that can be trained from limited data and that support principled model management to avoid misleading the search when approximation errors occur. In this work, we employ Gaussian process (GP) regression as the surrogate model. GP regression, often referred to as Kriging in traditional design optimization [27], defines a nonparametric Bayesian model over functions, where predictions are obtained by conditioning a prior distribution on the observed data [28]. GP-based surrogate models have been widely integrated into evolutionary algorithms for optimization problems involving expensive evaluations [29]. In addition to flexible nonlinear approximation, GP models provide predictive uncertainty estimates, which are commonly exploited in surrogate-assisted evolutionary computation to support model management and to control the use of approximate fitness evaluations during the search [27].

At each decision epoch, the genetic algorithm generates a population of candidate chromosomes H. Each chromosome is evaluated either exactly, by computing $f(H)$, or approximately, using the surrogate prediction $\hat{f}(H)$. To initialize surrogate learning, all individuals are evaluated exactly for a small number of initial generations, thereby constructing an initial training set. Subsequently, surrogate predictions are used for the majority of candidate solutions, while a limited subset is still evaluated exactly to provide new training data and to prevent error accumulation. In addition, whenever a new best-so-far solution is identified, its fitness is re-evaluated using the exact model. All exact evaluations collected during the search are incorporated into the training set, and the GP surrogate is retrained periodically to maintain accuracy as the population evolves.

IV. Computational Experiments

This section reports the computational experiments conducted to assess the performance of the proposed approach. We first describe the instance generation procedure as well as the operators and parameter settings employed, and then present the corresponding numerical results.

1) Instance generation: Computational experiments are conducted on synthetic benchmark instances adapted from [4] and extended to model advance booking and online decision making.

Each instance consists of n charging demands. Arrival times are sampled as $r_j \sim \mathcal{U}(0, 0.2n)$ (hours), and energy requirements as $e_j \sim \mathcal{U}(5.5, 66)$ (kWh). Let $p_j^{(11)} = e_j/11$ denote the uninterrupted charging time on a reference 11 kW charger. Departure times are generated as $d_j = r_j+(1+\alpha_j)p_j^{(11)}$, where $\alpha_j \geq 0$ controls temporal flexibility. As shown in Table I, the admissible range of α_j depends on $p_j^{(11)}$ to reflect reduced flexibility for longer charging sessions.

TABLE I

Flexibility factor intervals.

Range of $p_j^{(11)}$	Range of α_j	Range of $p_j^{(11)}$	Range of α_j
$[0.5, 1)$	$[0.1, 1.0]$	$[3, 4)$	$[0.1, 0.7]$
$[1, 2)$	$[0.1, 0.9]$	$[4, 5)$	$[0.1, 0.6]$
$[2, 3)$	$[0.1, 0.8]$	$[5, 6)$	$[0.1, 0.5]$

Each instance further specifies a charging-station configuration with m chargers of rated powers w_i and a station-level grid capacity W^g. A fixed mapping from n to (m, W^g) is used to ensure scalable yet realistic station sizing.

To model reservations and walk-in arrivals, each demand is assigned a submission time $st_j \leq r_j$. Three booking modes are considered: long-horizon (LH) booking with $st_j \sim \mathcal{U}(0, r_j)$, short-notice (SN) booking with $st_j \sim \mathcal{U}(\max\{0, r_j - 1\}, r_j)$, and a mixed mode (MD) in which each demand follows the long-horizon or short-notice booking rule with equal probability.

Experiments are performed for $n \in \{10, 20, 40, 50, 100\}$ under all booking modes. For each (n, mode) pair, 10 instances are generated, and each algorithm is run 30 times per instance with different random seeds. Results are averaged over instances and runs.

2) Algorithm configuration: Genetic operators and parameter values were selected through a preliminary tuning phase conducted prior to the main experiments. Alternative operator choices were first compared in a controlled setting, after which numerical parameters were tuned using Latin Hypercube Sampling (LHS) [30] to efficiently explore the parameter space under a limited computational budget. Robust configurations were identified based on aggregate performance over multiple instances and replications.

The final configuration employs tournament selection, one-point crossover, and elitist replacement preserving the two best individuals. The population size is set to 40, the maximum number of generations to 50, the crossover probability to 0.85, and the mutation probability to 0.05. The evolutionary process terminates when one of the following conditions is met: the maximum number of generations is reached, no improvement in the best fitness value is observed for 25 consecutive generations, or an upper bound on the objective value is attained. This upper bound corresponds to the case where all demands in B that are feasible under the current resource profile are successfully scheduled.

For surrogate-assisted evaluation, all individuals are evaluated exactly during the first three generations to initialize the training set. From the fourth generation onward, surrogate predictions are used for the population, while each individual is independently selected for exact evaluation with probability 0.1. In addition, the fitness of any new best-so-far solution is recomputed exactly. All exact evaluations are added to the training set, and the GP surrogate is retrained every 10 generations. GP hyperparameters are re-estimated periodically from the accumulating training data using standard GP model selection procedures [28].

A. Numerical Results

This subsection presents and analyzes the numerical results obtained from the computational experiments. All runs were performed on a desktop workstation with an Intel Xeon W-2295 CPU (3.0 GHz) and 128 GB RAM under Windows 11. The algorithms were implemented in `Python`, and the mathematical model was solved using CPLEX 22.1. *To support reproducibility, the benchmark instances and source code are publicly available at https://github.com/abdennourAzerine/cec-2026-ev-charging-benchmark.*

Figure 1 summarizes the performance of the proposed hybrid GA with exact evaluation and its surrogate-assisted variant across all tested instance sizes. The left panel reports the mean objective value over 30 independent runs per instance, averaged across the three booking modes. The right panel reports the corresponding mean *total computational time*, measured as the cumulative runtime over all submission epochs within each instance, averaged across booking modes.

Regarding solution quality, Figure 1 (left) shows that the surrogate-assisted variant achieves objective values comparable to those obtained with exact evaluation across all instance sizes. Figure 2 complements this view by reporting average objective values separately for each booking mode. These observations are supported by the Wilcoxon signed-rank results reported in *Test 1* of Table II, which compares objective values under each booking mode. At the 0.05 significance level, no statistically significant differences are observed, indicating that surrogate assistance preserves solution quality relative to exact evaluation.

Computational performance is reported in Figure 1 (right) and Figure 3. Figure 1 (right) shows consistent reductions in *total computational time* for the surrogate-assisted variant across instance sizes. Figure 3 further reports both the average total computational time and the average *per-decision computational time*, defined as the mean runtime required to process a single submission epoch, for each booking mode. Across all three booking modes, the surrogate-assisted variant exhibits lower runtimes under both metrics. Moreover, its average per-decision computational time remains below 18 seconds in all booking modes, supporting its practical suitability under online decision-time constraints. These differences are confirmed by the Wilcoxon signed-rank tests reported in *Test 2* (total computational time) and *Test 3* (per-decision computational time) of Table II, which indicate statistically significant runtime reductions for the surrogate-assisted variant across all booking modes at the 0.05 significance level.

Finally, Figure 4 reports the average per-decision computational time as a function of instance size, aggregated over all booking modes. For the largest instances, the surrogate-assisted variant requires approximately 32 seconds per decision on average, highlighting the impact of instance size on online runtime.

TABLE II
STATISTICAL SIGNIFICANCE TESTS COMPARING GA-EXACT WITH GA-GP BY BOOKING MODE. THREE METRICS EVALUATED: (1) OBJECTIVE FUNCTION QUALITY, (2) TOTAL EXECUTION TIME, (3) DECISION TIME PER EPOCH. WILCOXON SIGNED-RANK TESTS WITH SIGNIFICANT RESULTS ($p < 0.05$) IN **BOLD**.

Test 1: Objective Function Performance				
Mode	n	$\bar{x}_{Exact}$	$\bar{x}_{GP}$	p-value
SN	50	27.1	27.2	0.660
MD	170	21.8	21.8	0.974
LH	100	13.7	13.7	0.839
Test 2: Total Execution Time				
Mode	n	t_{Exact} (s)	t_{GP} (s)	p-value
SN	50	852.6	555.1	<**0.001**
MD	170	917.7	498.0	<**0.001**
LH	100	692.6	266.1	<**0.001**
Test 3: Decision Time per Epoch				
Mode	n	t_{Exact} (s)	t_{GP} (s)	p-value
SN	50	30.16	15.99	<**0.001**
MD	170	28.12	15.11	<**0.001**
LH	100	43.39	15.38	<**0.001**

V. CONCLUSION

This paper investigated an online variant of the Electric Vehicle Charging Scheduling Problem with admission commitment and preemptive charging under explicit charger-level and grid-capacity constraints. The problem requires sequential decisions to be computed under strict runtime limits. To address this challenge, we proposed a hybrid solution approach that combines an epoch-level genetic algorithm (GA) for admission and assignment decisions with a commitment-aware optimization model for exact feasibility certification and energy allocation. To further improve computational efficiency, a Gaussian process (GP) surrogate was integrated into the evolutionary loop to reduce the frequency of exact evaluations.

Computational results show that the GP-based surrogate preserves solution quality relative to the hybrid GA with exact evaluation. Across all tested instance sizes and booking modes, no statistically significant differences in objective values were observed between the exact and surrogate-assisted approaches, indicating that fitness approximation can be safely employed without degrading admission decisions. In contrast, surrogate assistance yields substantial and statistically significant reductions in both total computational time and per-decision computational time. These gains are particularly important in online charging settings, where decisions must be produced promptly as requests arrive.

Several directions for future research follow from this work. Adaptive surrogate management strategies could be explored to adjust retraining and infill policies based on online accuracy dynamically. Incorporating GP predictive uncertainty more explicitly into the decision process may further enhance evaluation efficiency. Finally, extending the framework to multi-station or networked charging systems and integrating stochastic arrival models would enable the study of larger-scale and more realistic online charging environments.

ACKNOWLEDGMENT

This work is supported by the CyMoVe project, funded under the AMI CMA France 2030 program, operated by the

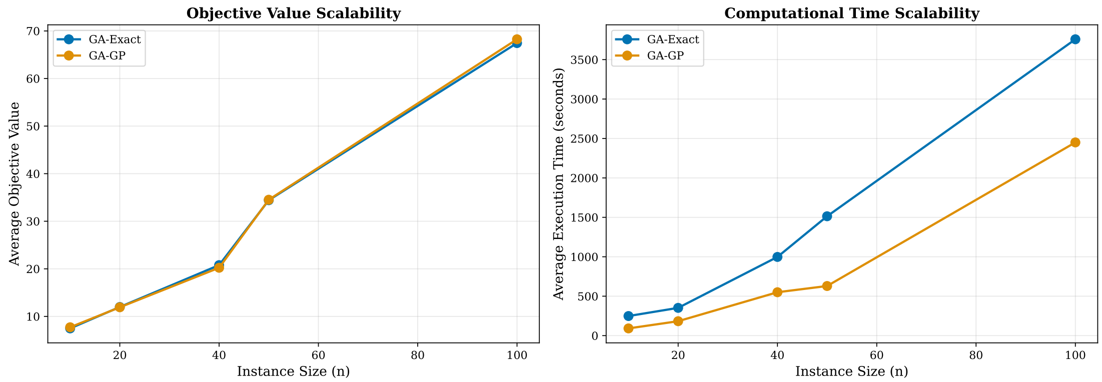

Fig. 1. Average objective values (left) and total computational time (right) across instances, grouped by number of demands.

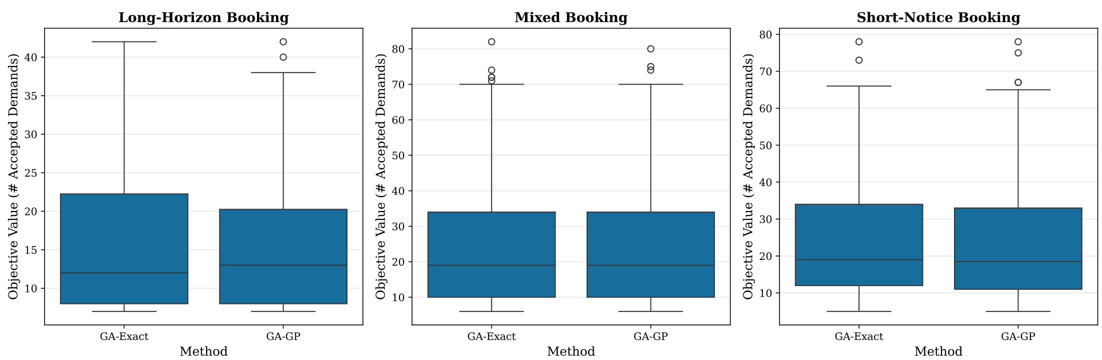

Fig. 2. Average objective function value by booking mode.

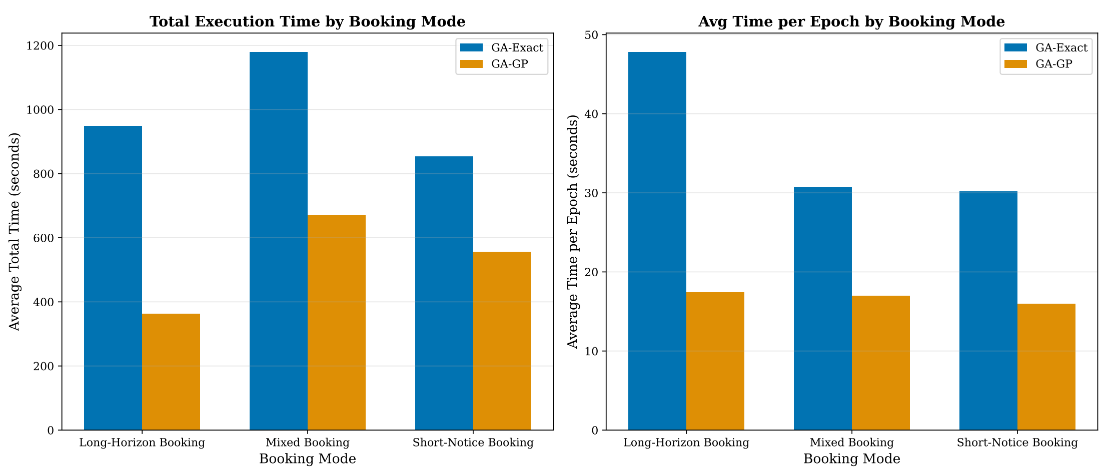

Fig. 3. Average execution time by booking mode.

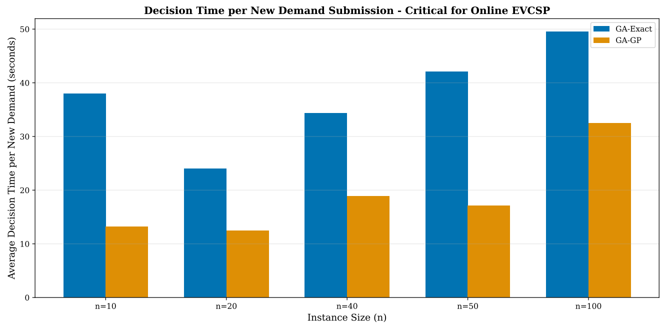

Fig. 4. Average execution time per decision epoch.

French *Agence Nationale de la Recherche* and *Groupe Caisse des Dépôts*.

References

[1] A. Azerine, M. Golabi, and L. Idoumghar, “Bi-level optimization of electric vehicle charging scheduling using hybrid genetic algorithm and reinforcement learning,” in *2025 IEEE Congress on Evolutionary Computation (CEC)*, 2025, pp. 1–8.

[2] IEA, “Trends in charging infrastructure,” 2023, accessed on 1 October 2024. [Online]. Available: https://www.iea.org/reports/global-ev-outlook-2023/trends-in-charging-infrastructure

[3] M. E. Kabir, C. Assi, M. H. K. Tushar, and J. Yan, “Optimal scheduling of ev charging at a solar power-based charging station,” *IEEE Systems Journal*, vol. 14, no. 3, pp. 4221–4231, 2020.

[4] I. Zaidi, A. Oulamara, L. Idoumghar, and M. Basset, “Maximizing the number of satisfied charging demands in electric vehicle charging scheduling problem,” in *International Conference on Artificial Evolution (Evolution Artificielle)*. Springer, 2022, pp. 89–102.

[5] Z. Yang, K. Li, A. Foley, and C. Zhang, “Optimal scheduling methods to integrate plug-in electric vehicles with the power system: a review,” *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 8594–8603, 2014.

[6] J. Han, J. Park, and K. Lee, “Optimal scheduling for electric vehicle charging under variable maximum charging power,” *Energies*, vol. 10, no. 7, p. 933, 2017.

[7] S. Shao, F. Harirchi, D. Dave, and A. Gupta, “Preemptive scheduling of ev charging for providing demand response services,” *Sustainable Energy, Grids and Networks*, vol. 33, p. 100986, 2023.

[8] I. Zaidi, A. Oulamara, L. Idoumghar, and M. Basset, “Hybrid heuristic and metaheuristic for solving electric vehicle charging scheduling problem,” in *Evolutionary Computation in Combinatorial Optimization: 21st European Conference, EvoCOP 2021, Held as Part of EvoStar 2021, Virtual Event, April 7–9, 2021, Proceedings 21*. Springer, 2021, pp. 219–235.

[9] A. Azerine, A. Oulamara, M. Basset, and L. Idoumghar, “Improved methods for solving the electric vehicle charging scheduling problem to maximize the delive,” in *2024 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2024, pp. 1–8.

[10] A. Azerine, M. Golabi, A. Oulamara, and L. Idoumghar, “Enhancing electric vehicle charging schedules: A surrogate-assisted approach,” in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 2024, pp. 183–186.

[11] X. Li, Z. Wang, L. Zhang, F. Sun, D. Cui, C. Hecht, J. Figgener, and D. U. Sauer, “Electric vehicle behavior modeling and applications in vehicle-grid integration: An overview,” *Energy*, vol. 268, p. 126647, 2023.

[12] A. S. Al-Ogaili, T. J. T. Hashim, N. A. Rahmat, A. K. Ramasamy, M. B. Marsadek, M. Faisal, and M. A. Hannan, “Review on scheduling, clustering, and forecasting strategies for controlling electric vehicle charging: Challenges and recommendations,” *IEEE Access*, vol. 7, pp. 128 353–128 371, 2019.

[13] W. Tang, S. Bi, and Y. J. Zhang, “Online charging scheduling algorithms of electric vehicles in smart grid: An overview,” *IEEE communications Magazine*, vol. 54, no. 12, pp. 76–83, 2016.

[14] H. Saber, H. Ranjbar, M. Ehsan, and A. Anvari-Moghaddam, “Transactive charging management of electric vehicles in office buildings: A distributionally robust chance-constrained approach,” *Sustainable Cities and Society*, vol. 87, p. 104171, 2022.

[15] H. Yu, C. Xu, W. Wang, G. Geng, and Q. Jiang, “Communication-free distributed charging control for electric vehicle group,” *IEEE Transactions on Smart Grid*, vol. 15, no. 3, pp. 3028–3039, 2023.

[16] Y. He, B. Venkatesh, and L. Guan, “Optimal scheduling for charging and discharging of electric vehicles,” *IEEE transactions on smart grid*, vol. 3, no. 3, pp. 1095–1105, 2012.

[17] F. Tuchnitz, N. Ebell, J. Schlund, and M. Pruckner, “Development and evaluation of a smart charging strategy for an electric vehicle fleet based on reinforcement learning,” *Applied Energy*, vol. 285, p. 116382, 2021.

[18] H. M. Abdullah, A. Gastli, and L. Ben-Brahim, “Smart management of electric vehicle chargers through reinforcement learning,” in *2024 4th International Conference on Smart Grid and Renewable Energy (SGRE)*. IEEE, 2024, pp. 1–8.

[19] V. Moghaddam, A. Yazdani, H. Wang, D. Parlevliet, and F. Shahnia, “An online reinforcement learning approach for dynamic pricing of electric vehicle charging stations,” *IEEE Access*, vol. 8, pp. 130 305–130 313, 2020.

[20] Z. Zhang, D. Zhang, and R. C. Qiu, “Deep reinforcement learning for power system applications: An overview,” *CSEE Journal of Power and Energy Systems*, vol. 6, no. 1, pp. 213–225, 2019.

[21] A. Farooq and K. Iqbal, “A survey of reinforcement learning for optimization in automation,” in *2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2024, pp. 2487–2494.

[22] M. H. S. Uiterkamp, M. E. Gerards, and J. L. Hurink, “Online electric vehicle charging with discrete charging rates,” *Sustainable Energy, Grids and Networks*, vol. 25, p. 100423, 2021.

[23] B. Alinia, M. H. Hajiesmaili, Z. J. Lee, N. Crespi, and E. Mallada, “Online ev scheduling algorithms for adaptive charging networks with global peak constraints,” *IEEE Transactions on Sustainable Computing*, vol. 7, no. 3, pp. 537–548, 2020.

[24] D. Orvosh and L. Davis, “Using a genetic algorithm to optimize problems with feasibility constraints,” in *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*. IEEE, 1994, pp. 548–553.

[25] A. Azerine, M. Golabi, and L. Idoumghar, “Bi-level optimization of electric vehicle charging scheduling using hybrid genetic algorithm and reinforcement learning,” in *2025 IEEE Congress on Evolutionary Computation (CEC)*, 2025, pp. 1–8.

[26] T. Bäck, *Evolutionary computation 1: Basic algorithms and operators*. CRC press, 2018.

[27] Y. Jin, “Surrogate-assisted evolutionary computation: Recent advances and future challenges,” *Swarm and Evolutionary Computation*, vol. 1, no. 2, pp. 61–70, 2011.

[28] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2, no. 3.

[29] A. I. Forrester and A. J. Keane, “Recent advances in surrogate-based optimization,” *Progress in aerospace sciences*, vol. 45, no. 1-3, pp. 50–79, 2009.

[30] M. D. McKay, R. J. Beckman, and W. J. Conover, “A comparison of three methods for selecting values of input variables in the analysis of output from a computer code,” *Technometrics*, vol. 42, no. 1, pp. 55–61, 2000.