

ORIGINAL RESEARCH

Open access

Real-Time Dynamic Pricing for EV Charging Stations with MCTS-Based Optimization

Minh Tran^{1*}, Duc Pham¹

Abstract

As electric vehicles (EVs) steadily gain popularity on public roads, commercial EV charging has become a routine business service. As this trend continues, charging station operators are actively exploring strategies to boost the profitability of their services or to distribute limited resources more effectively. Dynamic pricing has proven effective for increasing revenue in markets with varying customer demand. This paper introduces a Markov Decision Process (MDP)-based framework for dynamic pricing that maximizes either revenue or utilization for charging station operators. The proposed method is implemented through a Monte Carlo Tree Search (MCTS) algorithm. It is tested in simulation across multiple scenarios derived from a real-world dataset of EV charging sessions. Results demonstrate that the approach delivers near-optimal pricing decisions within milliseconds, even for large-scale instances, delivering substantial improvements in revenue or utilization compared to a conventional flat-rate baseline across various parameter settings.

Keywords Dynamic pricing, Reservations, Markov decision process, Monte carlo tree search, Electric mobility, EV charging

*Correspondence:

Minh Tran
minh.tran@gmail.com

¹ Department of Smart Power Systems, Faculty of Engineering, Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam

Introduction

Motivation

After more than a hundred years of limited adoption, electric vehicles (EVs) appear poised to replace internal combustion engine (ICE) vehicles as the dominant choice for personal transportation [1, 2]. Superior energy efficiency, reduced environmental impact, and lower costs give EVs a clear advantage over ICE vehicles. In this context, the European Environment Agency [3] noted that in 2020, the share of EVs in total new car registrations rose from 3.5% to 11%.

The growing adoption of EVs is attracting attention from a wide range of stakeholders, including governments, municipalities, automobile manufacturers, environmental organizations, and power utilities. All these parties are preparing for the anticipated surge in EV usage. For cities and electric utilities in particular, the broad uptake of EVs may necessitate major infrastructure investments, since a high volume of EVs could raise the grid's peak load by as much as threefold [4]. Consequently, demand-side management (DSM) techniques such as peak shaving and valley filling shift customer demand from high-demand periods (e.g., noon) to low-demand periods (e.g., early morning), thereby limiting the growth of infrastructure costs.

This issue of future infrastructure costs represents one of the main drivers behind the current focus on dynamic pricing. Accordingly, disciplines such as economics, revenue management, and supply chain management have extensively examined dynamic pricing as a tool for balancing demand across different sectors [5, 6]. In smart mobility research, which

avoids assumptions of centralized control, Gan *et al.* [7] suggest dynamic pricing to enhance taxi system efficiency. At the same time, Deilami *et al.* [8], Hayakawa *et al.* [9], and Versi and Allington [10] apply dynamic pricing to support power grid management in electric mobility by balancing demand, maintaining power quality, and addressing other grid-related objectives. These areas highlight dynamic pricing as a powerful instrument for shaping customer behavior. Therefore, this paper presents a dynamic pricing scheme to manage the rising demand for charging stations driven by the increasing prevalence of EVs.

Until recently, most studies on electric vehicle charging have focused on determining the optimal locations for charging stations [11–16]. However, such strategies offer only a temporary solution in an evolving landscape, as fixed placements quickly become suboptimal under new conditions. In contrast, dynamic pricing for EV charging and its use in load balancing remains effective despite ongoing changes in infrastructure, demand patterns, and energy prices. This path has been explored, for example, by Xiong *et al.* [17]. Their pricing model accounts for EV drivers' travel patterns and individually rational charging decisions. The authors treat the problem as a variant of sequential posted pricing [18] and formulate it as a mixed-integer, nonconvex optimization problem to maximize social welfare.

Dynamic pricing for EV charging can be a cost-effective, adaptable alternative to costly upgrades to existing grid infrastructure. Nevertheless, the approaches described earlier primarily focus on optimizing social welfare. In practice, however, many prospective charging station (CS) operators are privately owned businesses that lack strong motivation to prioritize social welfare. Instead, these private investors focus primarily on the expenses associated with deploying and operating charging services as well as on achieving satisfactory financial returns (From report “An Industry Study on Electric Vehicle Adoption in Hong Kong” by the Hong Kong Productivity Council (2014): www.hkpc.org/images/stories/corp_info/hkpc_pub/evstudyreport.pdf (accessed on 7 February 2022)).

Problem statement and contributions

This paper investigates how to assign available EV charging slots through variable pricing strategies. We concentrate on two primary goals: (1) increasing the income earned by the CS operator and (2) improving the total usage rate of the charging facility. The pricing challenge is modeled using the Markov Decision Process (MDP) approach [19].

Exact techniques such as value iteration (VI), policy iteration, or integer linear programming can deliver the best solution for smaller versions of this MDP. However, these approaches encounter severe difficulties with state-space explosion in realistic, large-scale settings. To overcome this obstacle, we employ a Monte Carlo Tree Search (MCTS) algorithm to find a near-optimal pricing rule. As far as we are aware, this is the initial application of MCTS to problems of this nature. Through this work, we extend current knowledge by applying theoretical tools to the practical task of setting dynamic EV charging prices in the domain of electric mobility.

Key contributions of our work are as follows:

- An original formulation of the EV charging dynamic pricing issue built on the Markov Decision Process (MDP) framework;
- A scalable heuristic pricing technique grounded in Monte Carlo Tree Search (MCTS) that works well for extensive real-world scenarios;
- Targeted optimization objectives focused on either boosting CS operator revenue or raising the usage of existing charging resources;
- A flexible collection of test cases developed using actual two-year operational data obtained from a German charging station operator;

- Simulation outcomes that reveal the suggested heuristic delivers performance close to that of precise solvers like Value Iteration, yet succeeds in producing solutions for very large problems where exact methods fail due to state-space explosion.

Related Work

Both price-driven and incentive-driven strategies offer strong potential for implementing demand-side management (DSM). Price-driven DSM motivates final consumers to adjust their electricity consumption patterns (e.g., load levels) in response to changing price signals. Incentive-driven DSM, by comparison, provides users with separate rewards or bonuses for altering their load, independent of or in addition to normal electricity tariffs. For the scenario examined here, we rely on the price-driven method.

Researchers in energy systems have developed various price-driven mechanisms over time, among them time-of-use (ToU) [20], real-time pricing (RTP) [21], and critical-peak pricing (CPP) [22]. These systems adjust consumer demand to better match supplier needs by raising charges when consumption peaks and lowering them during periods of abundant supply, such as when renewable output is high. More recently, an advanced technique called dynamic pricing [23] has emerged on the foundation of the earlier models.

In essence, dynamic pricing involves adjusting prices for offerings with strong or uneven demand, much like hotel rooms [24] or plane tickets [25]. Ticket and room prices shift depending on forecasted seasonal interest, current reservations, and traveler category (corporate versus vacation). Both airline seats and hotel nights come with hard deadlines — the flight takeoff or the reserved check-in day — after which the opportunity vanishes. EV charging capacity during a particular interval faces the same issue: it disappears if no vehicle connects. For resources with this perishable character, the aim is to generate maximum earnings while respecting time limits and shifting demand levels. Failure to use available capacity equals missed earnings for charging station operators. This challenge sits at the heart of revenue management [6].

Dynamic pricing has also been explored in studies seeking to better integrate renewable generation and electric vehicles into existing power networks. Earlier works in this area generally fall into two groups: price-profile methods and session-based methods. Price-profile approaches define separate rates for different fixed-time blocks, whereas session-based approaches apply a single uniform rate across the full length of a charging event. Our study follows the session-based direction. The next part reviews notable session-based proposals from previous research.

In Ban et al. [26], the researchers employ queuing theory to evaluate how charging stations perform when prices are adjusted in real time. They aim to boost overall system throughput and reduce customer wait times. Meanwhile, the work presented in Bhattacharya et al. [27] draws on game-theoretic principles, particularly the Vickrey–Clarke–Groves (VCG) auction mechanism, to set session prices that maximize social welfare. A notable feature of such auction-style methods is that EV users requesting identical session length, arrival moment, and energy needs may still end up paying different amounts.

When considering practical implementation, prior studies fall into two broad groups: offline and online strategies. Offline strategies establish fixed charging rates for extended horizons (e.g., a full day ahead) using forecasts of anticipated EV demand, such as the expected number of vehicles and their total energy requirements. Online strategies, however, calculate prices over short intervals and revise them regularly. This paper aligns with the online approach. Several relevant works exist along this line. Comparable to our method, Kim et al. [28] allow charging prices to fluctuate continuously and present EV users with individualized offers for each session. Users then choose whether to accept or decline the offer. They also introduce penalty payments that the CS operator must cover if any EV user's wait time exceeds an acceptable limit. The scheduling algorithm's primary target is to increase financial returns for the CS operators. Similarly, Limmer and Rodemann [29] address the problem of allocating limited charging-station capacity to

maximize operator profits. Their framework dynamically adjusts prices, giving EV users the option to accept or decline each offer, thereby helping lower the overall rejection rate.

This paper explores dynamic pricing for EV charging with an emphasis on online, session-based methods. Nevertheless, unlike earlier works, we cast the central problem as a Markov Decision Process (MDP). Our formulation extends the MDP pricing framework first described in Mrkos *et al.* [30], with important refinements in the representation of past charging demand patterns. Moreover, the introduction of the MCTS technique enables us to tackle substantially bigger problem sizes. As far as we know, this marks the first application of MCTS to dynamic EV charging pricing.

MDP Formulation of EV Dynamic Pricing Problem

This section presents the details of our dynamic pricing model, along with its formal definition within the Markov Decision Process (MDP) setting [19].

The model rests on two fundamental premises: (1) the presence of a single dominant provider — namely, the charging station (CS) operator — and (2) EV users who behave as non-strategic customers. At any given time, the CS operator manages a limited amount of charging capacity capable of serving several vehicles concurrently. The operator's purpose is to allocate this scarce resource to incoming requests while pursuing chosen performance targets (such as revenue generation or capacity utilization).

Over the course of the day, the operator receives a continuous stream of EV charging requests, presented as bookings for upcoming capacity slots [31, 32]. Each request receives a price determined by the active pricing policy. The EV user then decides whether to accept the offer (Figure 1) or reject it (Figure 1). Successful acceptance results in the operator reserving the requested capacity for that user. A rejection leaves the capacity open for later arrivals. This process constitutes a sequential, online, and session-oriented dynamic pricing task.

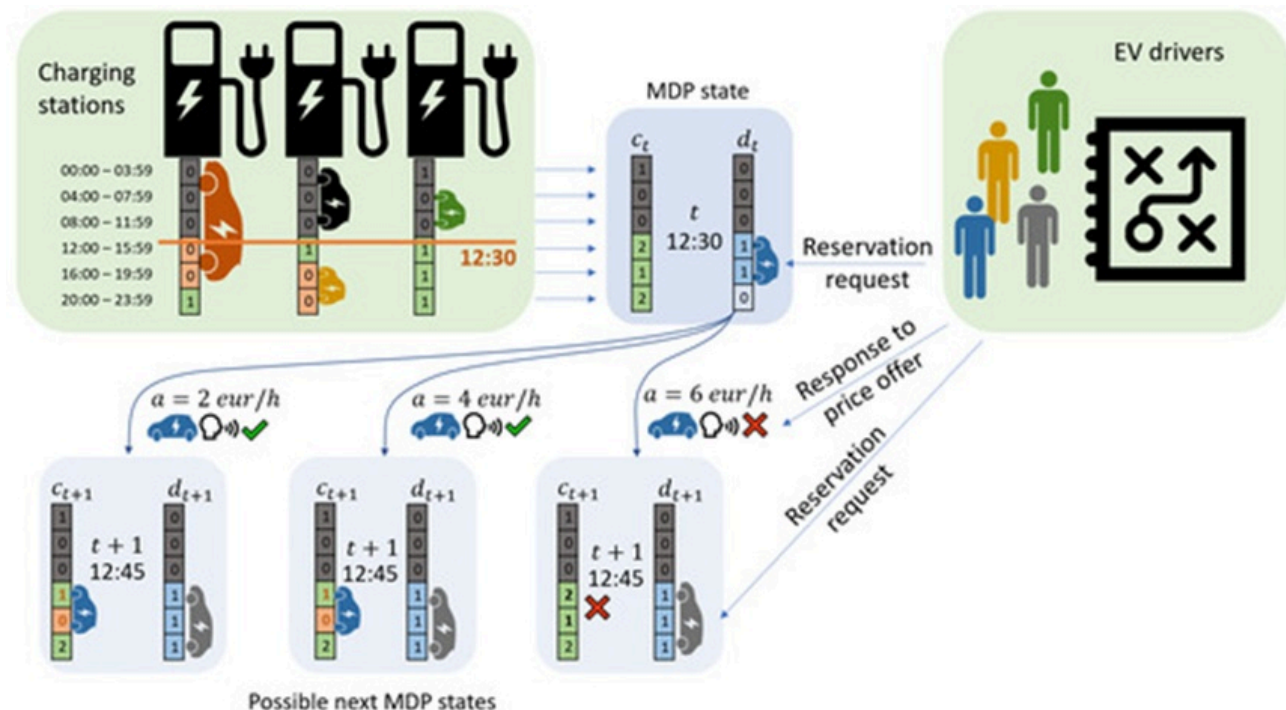


Figure 1. Depiction of MDP states. Blue squares denote the various MDP states. At any given timestep t , the charging station capacity is described via the capacity vector c_t . Individual elements within this vector indicate the remaining available charging capacity for each corresponding timeslot (time windows highlighted inside the green square). Incoming

charging session reservation requests received after the prior timestep appear in the vector d_t , where a value of one signals the specific timeslots being requested. Relying on the three state components c_t and d_t , the pricing policy generates an action a , which is the offered charging price. The user may accept this price (illustrated by the first two states at the bottom) or decline it (illustrated by the state on the right). Following this decision, the state progresses to the subsequent timestep (Figure 2). Acceptance of a charging request results in lowered capacity entries. The following reservation request is then added to the updated state. Importantly, timesteps operate at a significantly higher resolution compared to the charging timeslots themselves. Gray coloring highlights historical data about the charging capacity vector c_t and the session vector d_t , respectively.

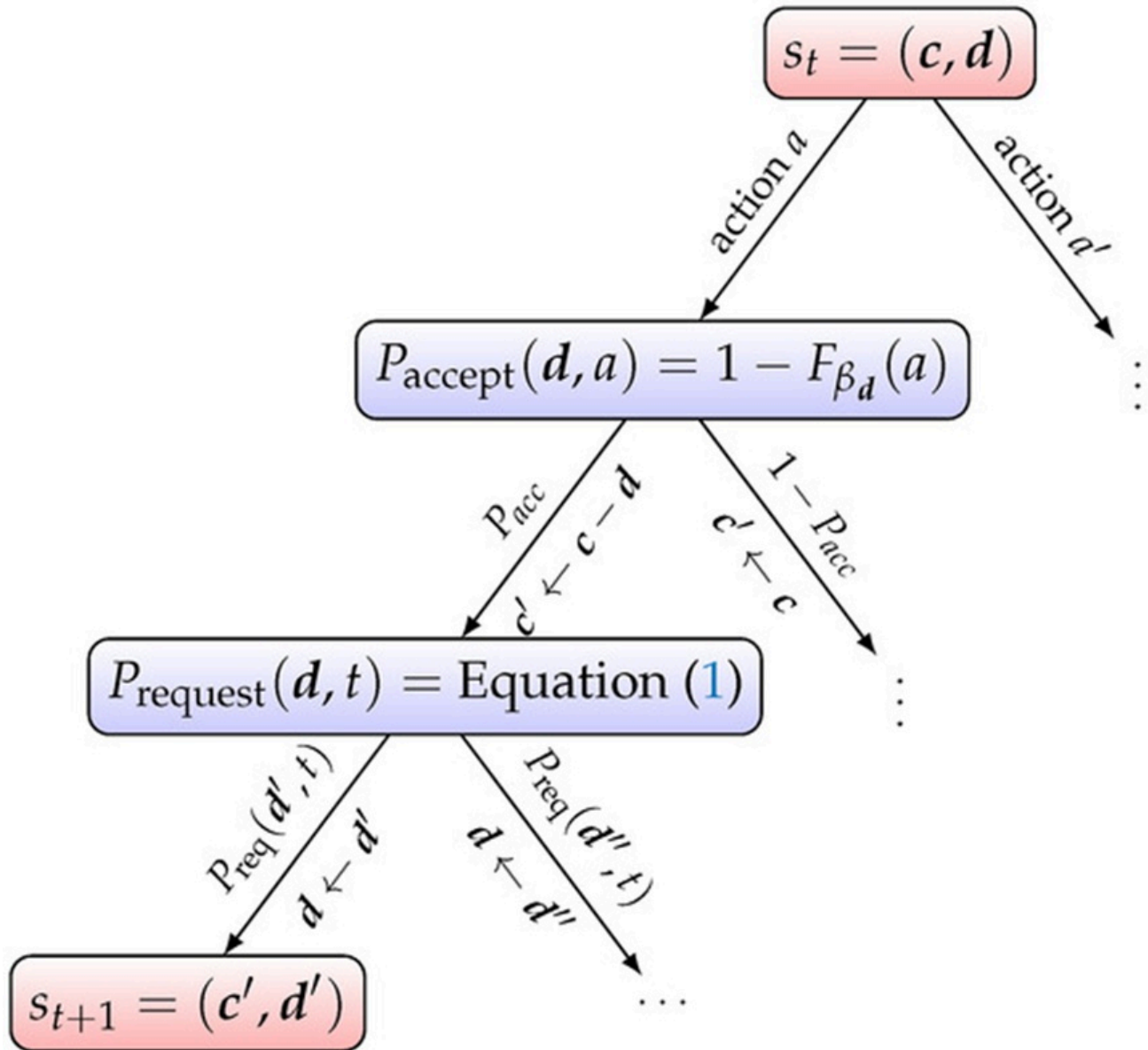


Figure 2. The structure of the transition function τ . Given state s_t , the probability of reaching the next state s_{t+1} is calculated by multiplying the probabilities shown along the edges. Decision nodes appear in red, while chance nodes are shown in blue and include the probability definitions applied to the edges.

The main difficulty facing the CS operator lies in the fact that optimal allocation of requested capacity would require perfect foresight of:

- Which future reservations will appear throughout the day
- How EV users will react to the prices being offered

However, the CS operator cannot access this information in advance. Instead, historical records allow the operator to build probabilistic models of reservation arrivals and user responses to different price levels. In this way, the operator can make optimal decisions in expectation.

MDP formalization

Markov Decision Processes supply a formal framework for sequential decision problems involving uncertainty. Solutions are either exactly optimal or approach optimality in expectation, relative to the underlying probability distributions. An MDP is formally defined by the tuple (S, A, τ, R) , where S denotes the set of states, τ is the transition function, R is the reward function, and A represents the set of available actions. For the EV charging dynamic pricing problem considered here, each of these four elements is described individually below.

State space

The state space S is composed of states of the form $s = (c, t, d)$ (we occasionally write $st = (c, d)$ to indicate the state at time t). In other words, a state is fully specified by the available capacity vector c at time t together with the particular charging session d requested by a customer at that moment. **Figure 1** depicts the MDP states under consideration. Among the three state variables, t corresponds to the current “wall-clock” time from the seller’s perspective (i.e., the CS operator). We discretize time into steps $t \in \mathbb{N}$ over a finite daily horizon, which means the total number of timesteps T is finite (for instance, a 15-minute timestep yields $T = 96$ steps per day).

The day is further divided into discrete EV charging timeslots. Each timeslot corresponds to a fixed unit of charging service (e.g., each timeslot spans 4 hours in **Figure 1**). The capacity vector $c \in \mathbb{N}^K$ indicates the remaining available charging capacity across each of the K timeslots for that day. Importantly, the number of timeslots K is considerably smaller than the number of timesteps T , satisfying $K \ll T$ (e.g., $K = 6 \ll T = 96$ in **Figure 1**).

Finally, d is a binary vector that encodes the charging session an EV user wishes to reserve. It has the same dimension as the capacity vector, so $d \in \{0,1\}^K$, where a value of 1 at a given position indicates that the user requests charging in that timeslot, and 0 indicates no request.

The total size of the state space is then $T c_0^K 2^K$, where c_0 stands for the initial capacity available in every timeslot. Even with a moderate number of EV users and charging stations, the number of possible MDP states grows exponentially as the number of time slots increases. This exponential growth creates a state-space explosion, rendering exact solution techniques such as value iteration (VI) or integer linear programming (ILP) impractical for realistic cases.

Action space

The action space A consists of all possible price values (expressed per hour of charging) that the CS operator may quote to arriving EV users. We consider a discrete, finite set of prices distributed uniformly across the range of users’ budgets, with $|A| = 2K$, the total number of price levels, set to twice the number of charging timeslots.

Reward function

The reward function $R(st, a, st_{+1})$ specifies the immediate payoff received when moving from state st to st_{+1} after selecting action a (i.e., offering a particular price). When the objective is revenue maximization for the CS operator, the reward equals the price accepted by the EV user, so $R(st, a, st_{+1}) = a$, or 0 in the case of rejection. When the objective shifts to utilization maximization (a value ranging from 0 for no sessions served to 1 for full usage of all daily timeslots), the reward upon acceptance becomes the fraction of capacity that is occupied: $R(st, a, st_{+1}) = |d|_1 / |c_0|_1$, where $|d|_1$ denotes the sum of

the elements in vector $\mathbf{d}$. Different objective functions naturally yield distinct optimal pricing policies. Any utility measure that can be defined solely in terms of MDP state transitions may be employed in this framework.

The transition function τ establishes the MDP reward by tracing the tree edges shown in **Figure 2** from the root node down to a leaf. Acceptance or rejection by the EV user is detected inside the reward function through a direct comparison of the capacity vector $\mathbf{c}$ between two successive states at times t and $t+1$. A reduction in capacity from s_t to s_{t+1} signals that the EV user has accepted the price proposed by the CS operator; otherwise, the offer is considered rejected. The formal expression below focuses on the revenue-maximization case only:

$$R(s_t, a, s_{t+1}) = \begin{cases} a, & \text{if } \mathbf{c}_{t+1} = \mathbf{c}_t - \mathbf{d}_t \\ 0, & \text{otherwise} \end{cases}$$

Transition function

The transition function $\tau(s_t, a, s_{t+1})$ expresses the probability of moving from state s_t to state s_{t+1} after the CS operator chooses action a . In this setting, it arises from the interplay between the arrival process of charging reservations and the budget distributions of EV users. Reservation arrivals at the charging station are modeled as Poisson processes ([33, 34]), with a separate process defined for each possible charging session d and its own arrival rate λ_d . In short, a Poisson process generates arrivals that are independent and identically distributed (IID), with the time gaps between consecutive arrivals following an exponential distribution.

The likelihood that a particular charging session d is requested at timestep t is derived from these discretized Poisson arrival processes. When time is divided into discrete steps, the total count of requests N_d for session d roughly follows a binomial distribution with expected value $E(N_d) = \text{Prequest}(d) \times T_d$. Here, T_d denotes the total number of timesteps during which session d remains available for sale. In contrast, $\text{Prequest}(d)$ denotes the probability that session d is requested at any timestep between 0 and T_d . Accordingly, the probability of observing any charging session request (or none) at timestep t is defined as:

$$P_{\text{request}}(d, t) = \begin{cases} P_{\text{request}}(d), & d \in D \text{ s.t. } t < T_d \\ 1 - \sum_{d \in D \text{ s.t. } t < T_d} P_{\text{request}}(d), & d = \emptyset \end{cases} \quad (1)$$

Where the second term accounts for the possibility that no charging session is requested at timestep t .

Once the observed demand N_d for session d is known, we estimate $\text{Prequest}(d)$ as $\text{Prequest}(d) = N_d/T_d$. The timestep granularity must be sufficiently fine to ensure that $\text{Prequest}(d)$ remains much smaller than 1, thereby limiting the discretization-induced approximation error.

In the proposed model, the sales window for each possible charging session opens at timestep 0 (the beginning of the day) and closes at timestep T_d , no later than the final timestep T , which marks the end of the day. Whenever a reservation request for session d arrives at timestep t , the probability that the EV user will accept the quoted price a is obtained from the cumulative distribution function of the associated budget distribution β_d as follows:

$$P_{\text{accept}}(d, a) = 1 - F_{\beta_d}(a) \quad (2)$$

The two building blocks of the transition function — $P_{\text{accept}}(d, a)$ and $P_{\text{request}}(d, t)$ — are combined by multiplication following the structure illustrated in **Figure 2**, yielding the overall transition probability $\tau(s_t, a, s_{t+1})$.

Referring to **Figure 2**, the root node offers $|A|$ possible actions, each leading to a binary accept/reject outcome at the second level. This outcome decides whether the capacity vector in $st+1$ is reduced from its previous value or stays unchanged. At the third level, one of the $[[0,1]^K]$ possible charging sessions (including the null session) may arrive, which sets the value of d in the subsequent state $st+1$. As a result, the largest branching factor at any state st equals $2^K + 1$, where K is the number of charging timeslots.

It is important to observe in the decision tree of **Figure 2** that the chance node involving $P_{accept}(d, a)$ appears before the node for $Prequest(d, t)$. The ordering is deliberate: the model first evaluates whether the chosen price a would be accepted for the currently requested session d , and only afterward determines which new session d' will appear in the next state.

MDP solution

Solving the MDP described above yields a pricing policy that maps every state in the state space S to an action in the action space A . An optimal policy maximizes expected utility given the transition function. In this work, the utility is defined as either the revenue earned by the CS operator or the utilization rate of the available charging capacity.

Optimal and Heuristic Solutions

This section explains the techniques employed to generate dynamic pricing policies. We first outline Value Iteration (VI), an offline technique for solving MDPs that converges to an optimal policy. We then present the Monte Carlo Tree Search (MCTS) solver, an online randomized heuristic that estimates high-quality actions from any given state.

VI offers a straightforward yet effective way to solve MDPs and reliably converges to an optimal policy regardless of the starting point. Its main strength is the ability to rapidly produce a full, near-optimal pricing policy, albeit at the expense of storing the entire state space in memory. Given that the state space size is $T \times K^2$, VI exhibits exponential space complexity relative to the number of timeslots. Consequently, it scales poorly for larger problem sizes. We therefore apply VI exclusively to smaller instances to obtain optimal policies and thereby benchmark the performance of the MCTS heuristic. It should be noted that alternative exact solution techniques for MDPs exist, including policy iteration and linear programming. These methods can also deliver the same optimal pricing policy as VI. However, like VI, they all require full enumeration of the state space. The selection of VI in this study is thus purely a matter of convenience.

For tackling large-scale problems, we prefer MCTS. In contrast to VI, MCTS avoids exhaustive enumeration of the state space. Instead, it focuses on identifying the best action from the current state and selectively expands only those branches that appear most promising. While MCTS provides only approximate best actions for each state, the quality of these approximations improves steadily as the number of iterations increases.

Importantly, the algorithm can be terminated at any moment and still return the current best available approximation. These characteristics make MCTS particularly well-suited for dynamic pricing applications. It allows rapid adaptation to changes in the operating environment, and even in sizable systems, it can compute price offers quickly enough to maintain acceptable response times for customers. To the best of our knowledge, this work represents the first effort to address the EV charging dynamic pricing problem by combining MDP modeling with MCTS.

VI operates as an offline method, with most of the computation performed before policy execution. Once the policy is ready, applying it during operation involves retrieving the best action from a stored data structure. MCTS, by comparison, does not construct the full policy in advance. Rather, it estimates the best action for a given state on the fly during execution. As a result, MCTS generally requires more runtime computational effort than VI.

Broadly speaking, MCTS [35] refers to a class of algorithms that rely on four repeated phases to select the best action from a given state s . Starting from a decision tree rooted at s , a tree policy guides the traversal until a node s_0 is chosen

for expansion, at which point a new leaf is appended to the tree. The value of this new node is then evaluated through a rollout—a forward simulation from s_0 to a terminal state using a random policy. The total reward collected during the rollout is subsequently propagated backward through the tree to update the relevant nodes.

This cycle of selection, expansion, rollout, and backpropagation is repeated until a preset time budget or iteration limit is reached. In general, more iterations lead to actions closer to the optimal choice. The overall structure of the MCTS procedure is summarized in **Algorithm 1**. The specific MCTS implementation used in this paper builds upon a Julia-based MCTS library [36].

```

input : Some MDP state  $s$ 
output: Action  $a$  to be taken in  $s$ 
Initialize root of the tree
for  $i = 0$  to 800 do
    Apply tree policy (traversing tree and expanding a new leaf node);
    Apply rollout policy (getting approximate value of leaf node);
    Backup values up the tree;
end
return Most used action in the root

```

Algorithm 1. General MCTS structure.

Tree policy

A central element of the algorithm is the tree policy, which controls navigation through the search tree and decides which branches to grow. Our MCTS version relies on the well-known Upper Confidence Bound for Trees (UCT) [37-39]. This variant treats each node as a separate bandit problem. It applies the upper confidence bound equation to carefully balance exploration of already-visited parts of the tree with opening new branches from the current state. Further details on the tree policy appear in Browne et al. [35]. In the experiments conducted here, trees are grown only to a maximum depth of 3, and use an exploration constant of 1. Values at leaf nodes are assessed via random rollouts.

Rollout policy

Another essential stage in MCTS is the rollout policy. It starts from the newly expanded leaf nodes generated by the tree policy and produces an estimate of the expected reward for those leaves. Our experimental setup employs a purely random rollout policy that repeatedly selects an arbitrary action from the leaf until the MDP reaches a terminal state.

Given that customer arrivals are modeled as a Poisson process (which becomes a Bernoulli process after discretization), we can accelerate the rollout procedure by directly drawing the interval until the next arrival from the corresponding inter-arrival distribution (exponential in continuous time or geometric after discretization). This shortcut enables the simulation to reach the end state with far fewer steps. In the closing phase of each MCTS cycle, the total reward gathered during the rollouts is propagated back through the tree, refreshing the statistics of all affected nodes. Across our experiments, previously built decision trees are retained and reused at successive decision points, noticeably accelerating convergence. The total count of tree traversals and rollouts is fixed at 800 per decision.

Experiments and Results

This section reports on computational experiments conducted using the MDP-based dynamic pricing framework and the accompanying MCTS solver outlined in earlier sections. The performance of the proposed approach is benchmarked against several reference strategies using synthetic test cases built to mirror the statistics of an actual charging-station dataset.

Pricing strategies are tested on full-day sequences of incoming EV charging requests covering the period from 00:00 until 23:59. Each sequence is composed of ordered tuples:

$$B_i = ((d_1, t_1, b_1), (d_2, t_2, b_2), \dots), i \in 1 \dots N \quad (3)$$

Where d_i , t_i , b_i denote the successive EV user charging requests.

Although the requested session d_i and its arrival time t_i are generated according to the demand process specified in [Eq. 1](#), the associated user budget b_i is drawn from the budget distribution β_d specific to the charging session.

Every pricing method processes the requests one after another, beginning with the first in the sequence. For each request, the method outputs a price, records the simulated user reaction, and decreases available capacity only when acceptance occurs. Once the full sequence has been handled, final values for capacity utilization and total revenue are recorded.

Evaluation methodology

Ideally, dynamic pricing techniques should be assessed by deploying them in actual operating conditions and comparing their outcomes with those of conventional fixed-price approaches. Such real-world trials are seldom possible in research contexts, however, since they require running controlled tests with genuine customers and live services.

In addition, side-by-side comparisons with previously published dynamic pricing techniques prove complicated. Most accessible methods impose narrow constraints on the problem formulation or rely on models that cannot be directly aligned with our setup, and they typically offer less adaptability than the MCTS-MDP framework. For example, the approximate dynamic programming method introduced in [Ke et al. \[40\]](#) delivers optimal prices for sizable instances, yet only after imposing strong simplifying assumptions, such as linear demand relationships. A further complication arises from the current lack of widely accepted standard benchmark collections for EV charging dynamic pricing. Nevertheless, useful conclusions about the effectiveness of our Monte Carlo Tree Search heuristic can still be drawn by pitting it against clearly specified baseline approaches.

Baseline methods

Owing to the challenges in assessing dynamic pricing rules, we test the proposed MCTS method against three reference strategies: flat-rate, MDP-optimal VI, and an oracle. The flat-rate approach provides a conservative lower bound for revenue performance. The VI baseline delivers an optimal policy given the MDP parameters and represents the strongest achievable result within the modeled environment. Lastly, the oracle strategy reflects the theoretical best-case allocation that would be attainable if the CS operator possessed complete advance knowledge of every upcoming request and every user's exact budget — something unattainable under normal operating conditions.

Flat-rate

The flat-rate approach maintains a constant price for every charging session regardless of when it occurs. It applies a single, unchanging charging rate per minute to all incoming requests. This fixed rate is calculated using a collection of training sequences sampled from the given problem instance. The chosen rate is the one that delivers the highest average utility value across those training sequences. Once determined, this same fixed rate is applied uniformly during

all testing simulation runs. Training is performed on 25 randomly selected sequences from the 100 sequences used for evaluation.

Value iteration

The optimal policy obtained through the Value Iteration algorithm acts as our second reference pricing method. This policy achieves optimality in expectation given the transition probabilities specified within the MDP formulation. Nevertheless, Value Iteration cannot handle larger problem sizes because it requires storing a value for each state. As a result, its use is limited to assessing only small-scale problem instances. To produce the VI baseline, we apply a standard Value Iteration solver (<https://github.com/JuliaPOMDP/DiscreteValueIteration.jl> (accessed on 7 February 2022)). The policy generated by this solver is then used in the experiments to select the pricing actions.

Oracle

Lastly, we evaluate our MCTS-based pricing technique against the oracle reference strategy. In contrast to the other methods, the oracle cannot be implemented in practice. It assumes full prior knowledge of the entire request sequence and of each EV user's budget when setting prices. Using this perfect foresight, the oracle optimizes the selected performance metric and thereby establishes a theoretical ceiling on the attainable revenue and capacity utilization for any pricing-based allocation approach.

For the k th request sequence B_k , where i indexes individual requests, the maximum possible revenue is found by solving a simple binary integer program:

$$\begin{aligned} & \text{maximize} \quad \sum_{i \in \{1 \dots |B_k|\}} x_i [b_i]_A \quad (4) \\ & \text{subject to} \quad \sum_{i \in \{1 \dots |B_k|\}} x_i d_i^j \leq c_0^j \quad j = 1, \dots, |R| \\ & \quad \quad \quad x_i \in \{0, 1\} \quad i = 1, \dots, |B_k|. \quad (5) \end{aligned}$$

In the objective Eq. 4, the term $b_i = \max_{a \in A, a \leq b_i} a$ represents the mapping of each budget value in sequence B_k to the nearest lower price level in the action space A . Constraints (5) guarantee that the total resources consumed by accepted sessions stay within the initial available capacity c_0 . We solve the optimization problem above using a standard off-the-shelf mathematical programming solver [41].

Problem instances and EV charging dataset

The experiments in this study examine several parametric problem instances constructed from a real-world dataset of EV charging sessions collected at a location in Germany. After data cleaning and preparation, the dataset consists of 1513 charging sessions collected over two years. The preparation steps included eliminating mostly invalid sessions lasting under 10 minutes and combining successive sessions performed by the same EV user, which were typically the result of faulty cable connections.

Figure 3 shows the histograms of start times and durations for the charging sessions present in the dataset. For generating the problem instances, we model these histograms using a normal distribution for start times and an exponential distribution for durations. Within the dataset, start times and durations exhibit virtually no correlation (Pearson's correlation coefficient = 0.008), allowing us to treat the two distributions as independent for further information on computing Pearson's correlation coefficient [42]. In addition, all charging sessions in the dataset finish by 23:59, and this same cutoff is enforced in our problem instances.

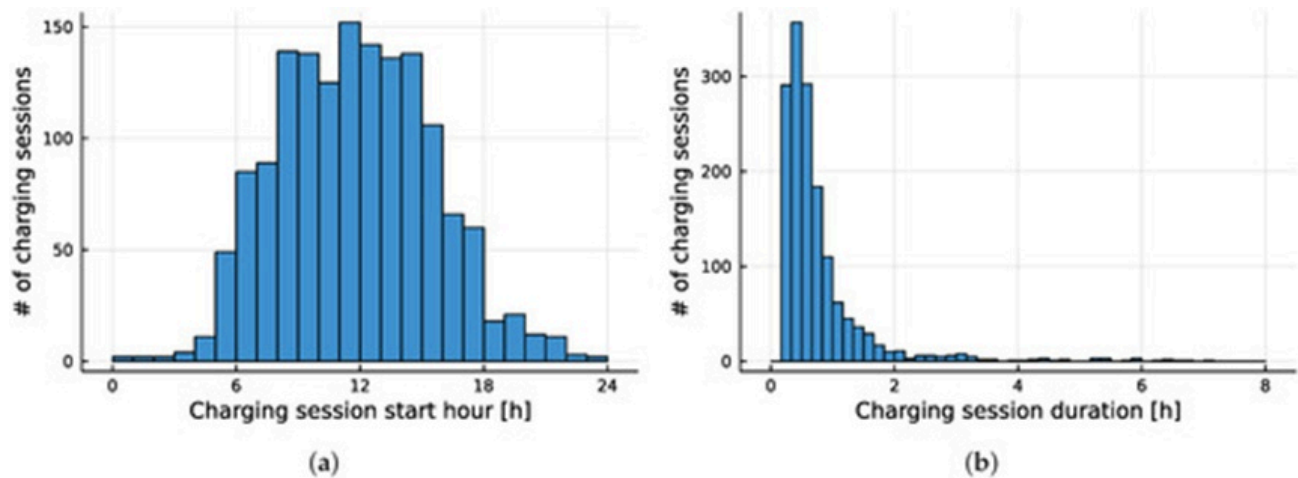


Figure 3. Histograms show the start of charging sessions (a) and duration (b) contained in the considered dataset.

The distributions of start time and duration represent fundamental elements of the demand model inside the MDP. However, the dataset captures only sessions that actually occurred, providing no information on the interval between the time an EV user submits a reservation request and the intended charging start time (the lead-time distribution). For simplicity, we assume a uniform lead-time distribution that is independent of the start-time and duration distributions.

These three distributions — start time, duration, and lead time — together fully specify the demand model. To create the EV charging request sequences (Eq. 3), the remaining piece is the customer budget distribution. Lacking any real data on budgets (expressed as a price per minute of charging), we adopt a normal distribution, a conventional choice when no other information is available and supported by the central limit theorem [43]. It is worth noting that the model itself does not depend on any unique characteristics of the normal distribution; any suitable probability distribution could be used instead.

Once we have defined the three key demand distributions—covering start time, duration, and lead time—along with the user budget distribution, we can generate sequences of EV charging requests that constitute the individual pricing problem instances. Given that these instances operate on discretized timesteps and timeslots, the distributions are first converted into discrete forms, after which sampling is performed directly from the discretized versions.

In the end, the single remaining tunable parameter for constructing the problem instances is the overall number of requested timeslots. This parameter is drawn from a binomial distribution, and its expected value serves as the demand scaling factor applied uniformly across all problem instances.

Results

Our experimental evaluation relies on a collection of pricing problem instances to illustrate both the scalability and the competitive strength of the MCTS-based method. The first experiment assesses how pricing techniques perform when the expected duration of each charging request (session) remains constant while the total number of charging timeslots and charging requests is progressively increased. The second experiment investigates pricing behavior under steadily rising overall demand, achieved by simultaneously increasing the count of requests and the combined duration of all requests. Across both experiments, all pricing policies and baseline methods are evaluated using the same parameter configurations.

Fixed-demand experiment

In the opening experiment, a set of 15 parametric problem instances was produced. These instances feature charging time slot durations ranging from 8 h to 1 h, along with varying numbers of timesteps (each instance defines the total

timesteps as exactly 8 times its number of time slots). The demand scaling parameter—specifically the expected number of requested timeslots—was chosen so that the total requested charging duration (calculated from the expected value according to the MDP distributions, not by adding up the actual durations of sampled requests) across all charging sessions reaches precisely 2/3 of the station's total charging capacity. The facility is equipped with three charging points that can support up to three charging sessions running simultaneously.

All charging requests are generated by sampling from the discretized distributions introduced previously. Importantly, this design leads to a higher request count as the number of timeslots grows, accompanied by a corresponding drop in their average duration (**Figure 4**). In addition, charging demand exhibits a strong peak in the middle of the day (**Figure 3**). As a result, the scenario qualifies as high-demand, with a realistic chance that not every request can be accommodated given the limited available capacity.

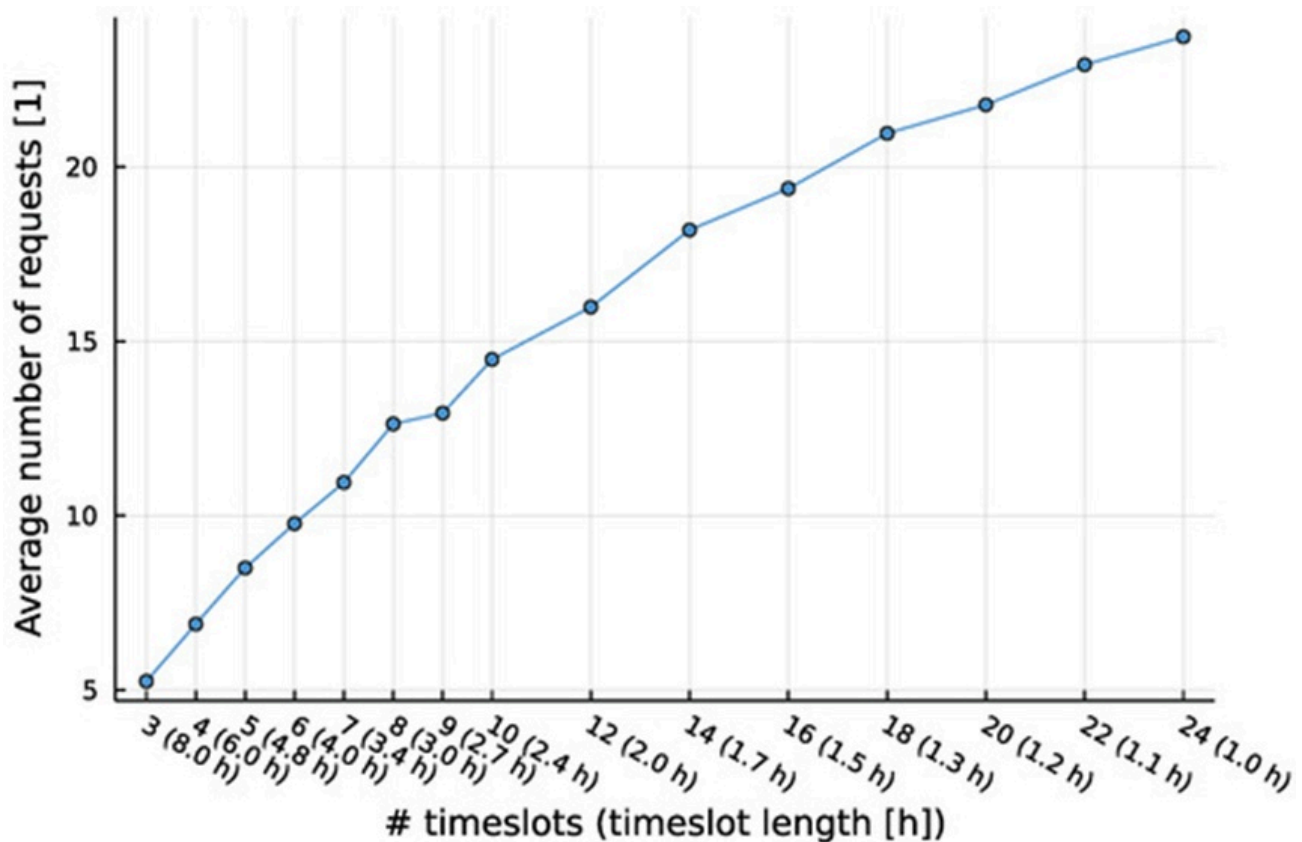


Figure 4. The number of requests in the experiment is constant, with a constant expected duration of all charging requests, but a varying number of timeslots.

Under these conditions, optimization is first carried out for both revenue and utilization using each of the tested methods. For every problem instance, 100 separate charging request sequences are sampled, and the resulting revenue and utilization figures are computed for all methods. The average performance when optimizing for revenue is shown in **Figure 5**, while the corresponding utilization-focused outcomes are shown in **Figure 6**. In both charts, it is evident that the oracle establishes the theoretical performance ceiling for any pricing technique, thanks to its perfect-information assumption.

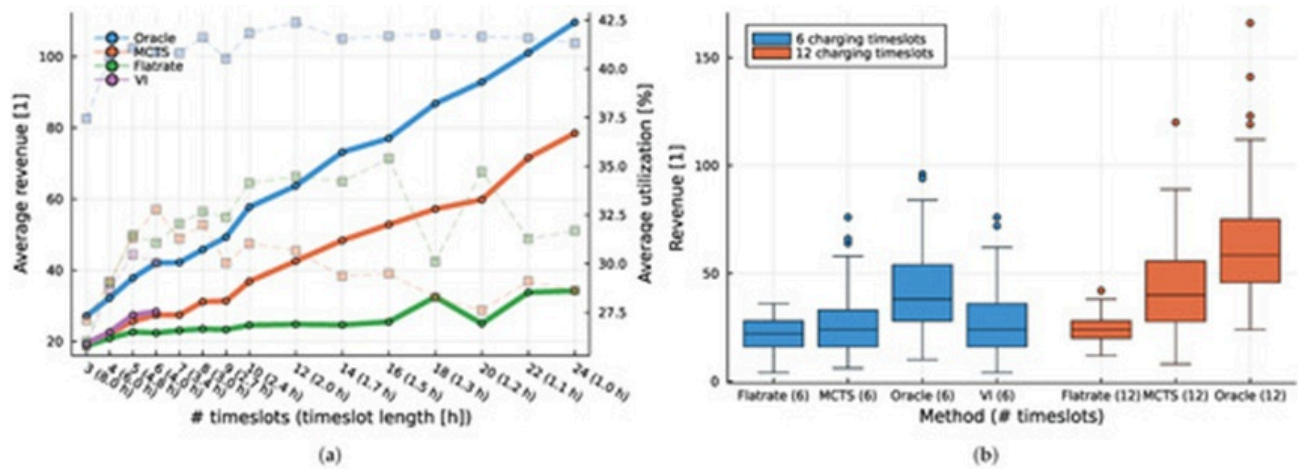


Figure 5. Revenue achieved in the constant-demand experiment under revenue maximization. Panel (a) presents the average values computed over every instance, whereas panel (b) uses boxplots to depict only the scenarios involving 6 and 12 timeslots. The secondary y-axis on the right, along with the faint lines in (a), represents the utilization levels attained by each approach (even though utilization was not the targeted optimization goal). In this experiment, the anticipated total charging hours remain fixed, but the value of K — the number of timeslots available in the 24-hour sales horizon — varies across problem instances.

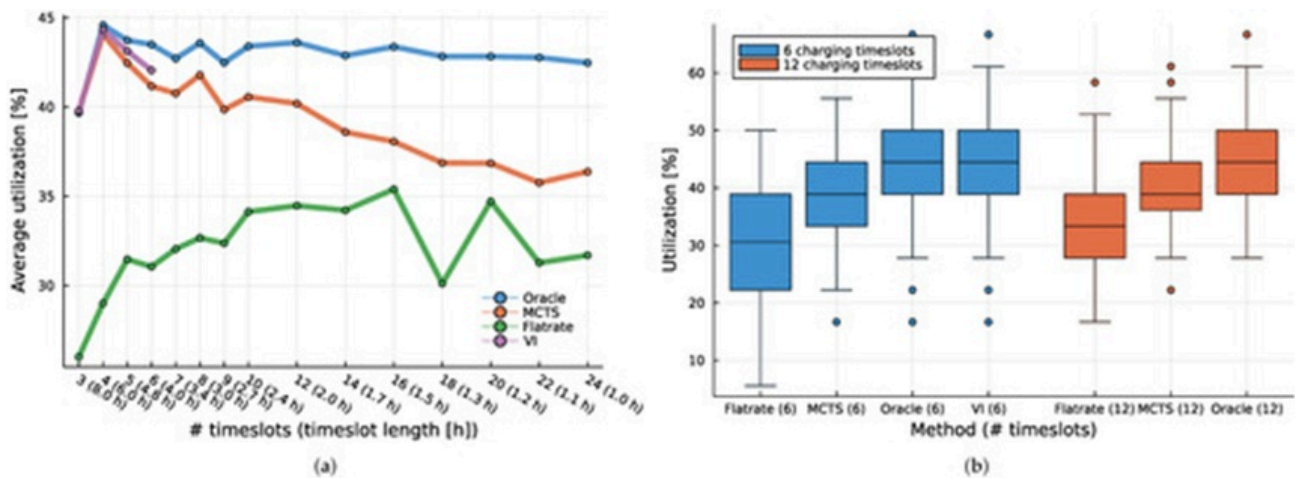


Figure 6. Utilization achieved in the constant-demand experiment under utilization maximization. Panel (a) presents the average values computed over every instance, whereas panel (b) uses boxplots to depict only the scenarios involving 6 and 12 timeslots. The anticipated total charging hours remain fixed, but the value of K — the number of timeslots available over the 24-hour sales horizon — varies across problem instances.

Turning to revenue optimization (Figure 5a), the VI baseline, which yields realistic expected-value-optimal pricing decisions, can handle instances with up to 6 time slots. Once the number exceeds 6, memory constraints prevent storing the entire state space. In comparison, the MCTS solution performs well against the VI baseline. It clearly exceeds the flat-rate approach by a considerable margin and comes close to matching the oracle baseline. That said, when the objective is revenue maximization, the flat rate still delivers noticeably better utilization than MCTS.

Interestingly, as shown in Figure 5a, both the oracle baseline and MCTS generate progressively higher revenue as the number of timeslots increases. The main driver of this pattern is that the overall requested charging duration remains fixed across all problem instances. More time slots naturally translate into more separate charging requests. Thanks to the finer resolution of the timeslots, the same total expected charging time can now be distributed across a wider range of

session lengths and combinations. This expanded pool of requests enables the oracle and MCTS to selectively accept those coming from EV users willing to pay more. Because the flat rate lacks a mechanism to make such selections, its revenue output remains essentially unchanged regardless of the number of timeslots.

Shifting the focus to utilization optimization in **Figure 6a** changes the picture. Here, the dynamic pricing strategies lose their ability to benefit from the higher budgets of certain users. The flat-rate and oracle methods continue to produce fairly consistent utilization levels in most cases. In contrast, MCTS utilization gradually drops as both the request count and timeslot count increase (while remaining close to the VI baseline in the range where VI remains computable). Once again, the root cause lies in the larger volume of smaller, more fragmented requests. Smaller fragments increase the probability that multiple requests will compete for the same time window. The oracle can perfectly sidestep these conflicts due to its complete foresight, but MCTS cannot. As a result, the scheduling task grows considerably harder with each additional request and timeslot. Keeping the aggregate requested charging duration constant while raising the number of shorter individual requests opens a clear path for dynamic pricing to lift revenue by directing capacity toward customers who value it most. The experiments confirm that MCTS effectively captures this advantage, closely tracking the ideal VI solution whenever comparisons are possible. It is worth noting that the spread of high-speed, long-distance charging infrastructure—such as Tesla Superchargers—tends to create precisely this kind of environment, with many brief sessions concentrated at high-traffic spots like highways. Such sites would therefore be especially suitable for applying the dynamic pricing framework presented here. At the same time, increased fragmentation also creates more potential scheduling conflicts, in which MCTS can still outperform the flat rate. However, its relative advantage diminishes as the total number of requests continues to climb.

Variable-demand experiment

For the second experiment, the timeslot count is locked at 48, so each timeslot corresponds to a 30-minute interval. The total requested charging duration is then swept from 12 h up to 84 h across the various problem instances. The bottom end of this range reflects a low-demand setting, where the expected number of requests is only 17% of the station's full capacity of 72 charging hours. The top end simulates an extremely high-demand situation in which satisfying every request becomes impossible.

As before, 100 charging request sequences are drawn from each instance, and the performance metrics are averaged. Revenue optimization outcomes are presented in **Figure 7a**, while utilization optimization outcomes appear in **Figure 7b**.

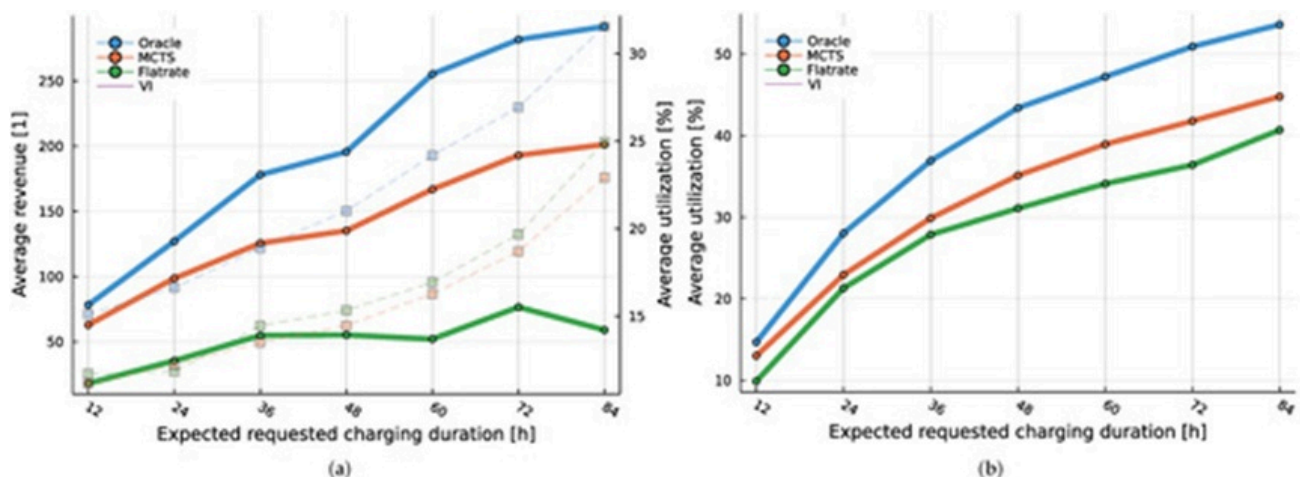


Figure 7. Revenue (a) and utilization (b) from the experiment that employs distributions recorded at an actual charging site. The right-hand y-axis, together with the thin plot lines in (b), illustrates the utilization achieved by each method (which was not the optimization objective). In this setup, the number of timeslots remains fixed at 48, corresponding to 30-

minute charging slots. In contrast, the expected demand rises from 1/6th of the total available charging capacity (3 chargers with 48 charging timeslots each) to 7/6.

Across both experiments, MCTS consistently outperforms the flat-rate approach, and the performance gap widens as demand increases. Under revenue maximization, MCTS achieves markedly superior revenue compared with the flat rate, all while maintaining roughly equivalent utilization (Figure 7a).

For the specific problem instance where the total expected demand equals 48 charging hours, Figure 8 shows how MCTS adjusts its pricing decisions based on the current state. In particular, the plot tracks the price quoted for a one-hour charging session scheduled to begin at 16:00. The price starts high but declines steadily as the scheduled start time approaches, eventually approaching zero right at the session's start time.

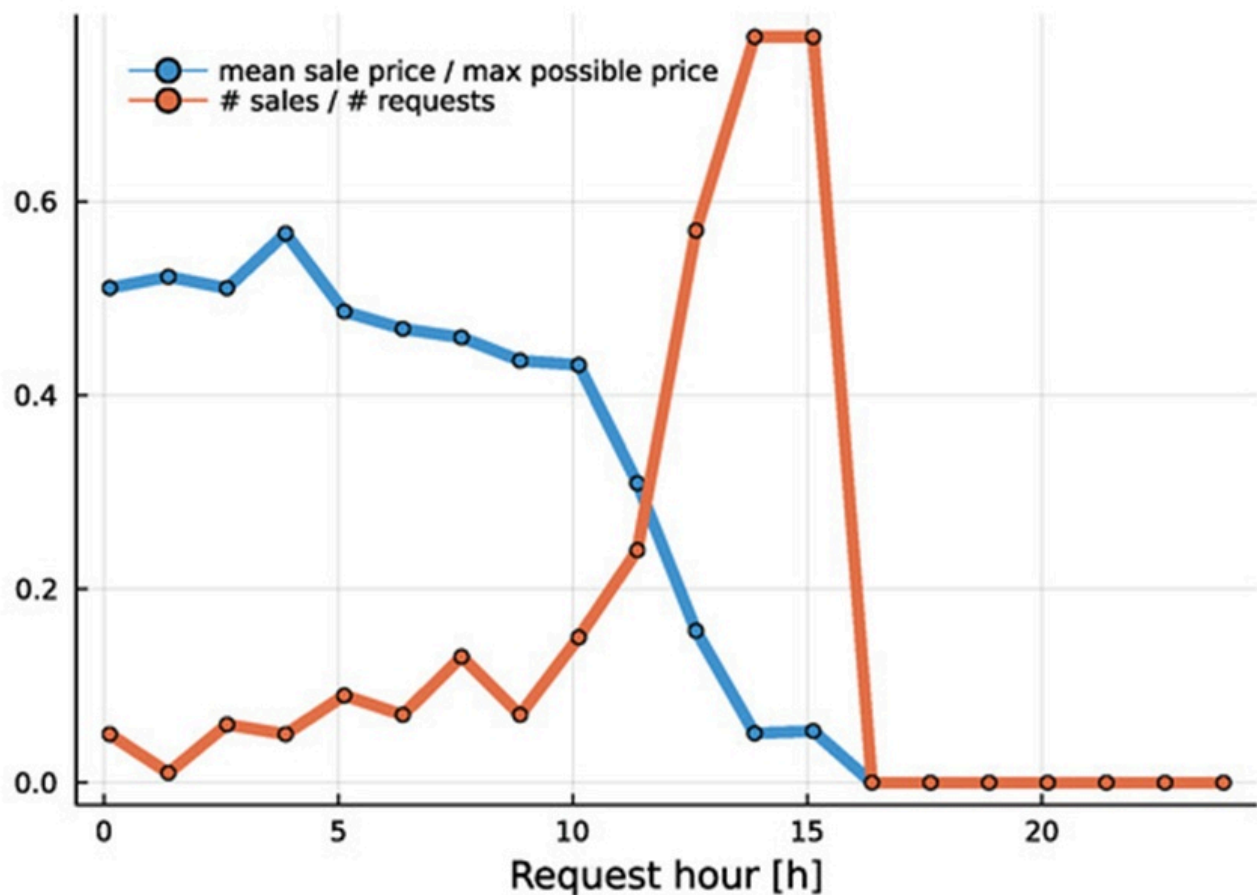


Figure 8. MCTS pricing reaction for the state $s=(c0,1,d)$ (that is, complete initial capacity at the first time step) in response to a request d involving a one-hour charging session that begins at 16:00. The graph displays the average MCTS prices normalized by the highest feasible price ((A)), along with the average count of accepted price offers across 100 randomly selected user budgets.

In summary, the MCTS approach boosts both revenue and utilization when demand rises, closely tracking the results of the oracle baseline. Regarding computational performance, generating the best-action estimate with MCTS requires no more than 9 ms per state across all problem instances examined in this study (on a single core of an Intel(R) Core(TM) i7-3930K CPU @ 3.20 GHz).

Conclusion

Dynamic pricing strategies have existed in the marketplace for over two decades. Thanks to their proven effectiveness in sectors such as airline and hotel reservations, the approach has also gained traction within energy systems. It represents a valuable tool for demand-side management, enabling grid operators to influence end-user consumption patterns, particularly during periods of limited supply.

In recent years, dynamic pricing has begun to be applied to electric vehicle (EV) charging. This paper investigates dynamic pricing for EV charging to maximize either (1) the revenue earned by the charging station (CS) operator or (2) the overall utilization of the installed capacity. We model the problem using a Markov Decision Process framework and introduce a heuristic solution grounded in Monte Carlo Tree Search (MCTS). The effective deployment of MCTS to address dynamic pricing for EV charging constitutes a new contribution of this work. Experiments were conducted on a real-world dataset provided by a German charging station operator. We benchmarked the conventional flat-rate approach, value iteration, and an oracle scenario against the proposed MCTS-based pricing mechanism.

The outcomes of the first experiment (Figures 5 and 6) demonstrate that the MCTS method attains at least 93.6% of the revenue achieved by the optimal policy from value iteration, while maintaining comparable result variance. Moreover, MCTS successfully scales to problem instances up to ten orders of magnitude larger than those solvable by value iteration when measured by state-space size.

Additionally, the variable-demand experiments (Figure 7) indicate that MCTS delivers substantially higher revenue to CS operators than the flat-rate baseline, up to 3.5 times higher. At the same time, pursuing revenue maximization results in utilization levels that are up to 8% lower than those under the flat rate. Conversely, when the objective is switched to utilization maximization, MCTS can improve utilization by up to 32% compared to the flat-rate scheme.

Because Markov Decision Processes support the integration of multiple objectives within a single reward function, a promising avenue for future research is to explore alternative optimization targets and their combinations—for instance, approaches that enhance revenue while avoiding significant drops in utilization. Further directions could include extending the framework to multiple geographically dispersed charging sites or refining the user behavior model to allow customers to modify their requests to meet their charging requirements.

The current MDP formulation also offers opportunities for refinement. The state space might be compressed by adapting the dimensionality of the capacity and product vectors once certain time slots become occupied. Within the MCTS algorithm, performance could be enhanced by incorporating additional domain-specific knowledge, such as specially designed rollout policies informed by domain expertise. In addition, the MCTS framework could, in theory, be adapted to accommodate continuous timesteps and continuous action spaces. On the evaluation side, realism could be increased by integrating actual user budget distributions and lead-time patterns drawn from other real-world domains.

Abbreviations

CPP	Critical peak pricing
CS	Charging station
DSM	Demand-side management
EV	Electric vehicle
ICE	Internal combustion engine
ILP	Integer linear programming
MCTS	Monte Carlo tree search
MDP	Markov decision process

RTP	Real-time pricing
ToU	Time of use
VI	value iteration

Acknowledgements

The research leading to these results was supported by the European Union Horizon 2020 ELECTRIFIC project (Grant No. 713864).

Conflict of interest

None

Financial support

This research was supported by the OP VVV MEYS-funded project CZ.02.1.01/0.0/0.0/16_019/0000765 “Research Center for Informatics”.

Ethics statement

None

Received: 11 Mar 2025
Published online: 20 September 2025

Revised: 09 May 2025

Accepted: 18 Jun 2025

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

References

Eider M, Sellner D, Berl A, Basmadjian R, de Meer H, Klingert S. Seamless Electromobility. In: Proceedings of the Eighth International Conference on Future Energy Systems; 2017 May 16-19; Hong Kong, China. New York (NY): ACM; 2017. p. 316-21.

Kirpes B, Klingert S, Basmadjian R, de Meer H, Eider M, Perez Ortega M. EV Charging Coordination to secure Power Grid Stability. In: Proceedings of the 1st E-Mobility Power System Integration Symposium; 2017 Oct 23; Berlin, Germany. 2017.

European Environment Agency. New Registrations of Electric Vehicles in Europe, 2021 [Internet]. 2021 [cited 2022 Feb 7]. Available from: <https://www.eea.europa.eu/ims/new-registrations-of-electric-vehicles#:~:text=The%20uptake%20of%20electric%20cars,registrations%20in%20just%201%20year>

Fairley P. Speed Bumps Ahead for Electric-Vehicle Charging [Internet]. IEEE Spectrum. 2021 [cited 2022 Feb 7]. Available from: <https://spectrum.ieee.org/speed-bumps-ahead-for-electricvehicle-charging>

Albadi M, El-Saadany E. A summary of demand response in electricity markets. *Electr Power Syst Res*. 2008;78(11):1989-96.

McGill JI, van Ryzin GJ. Revenue Management: Research Overview and Prospects. *Transp Sci*. 1999;33(2):233-56.

Gan J, An B, Miao C. Optimizing Efficiency of Taxi Systems: Scaling-up and Handling Arbitrary Constraints. In: *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*; 2015 May 4-8; Istanbul, Turkey. Richland (SC): IFAAMAS; 2015. p. 523-31.

Deilami S, Masoum AS, Moses PS, Masoum MAS. Real-Time Coordination of Plug-In Electric Vehicle Charging in Smart Grids to Minimize Power Losses and Improve Voltage Profile. *IEEE Trans Smart Grid*. 2011;2(3):456-67.

Hayakawa K, Gerding EH, Stein S, Shiga T. Online mechanisms for charging electric vehicles in settings with varying marginal electricity costs. In: *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*; 2015 Jul 25-31; Buenos Aires, Argentina. 2015.

Versi T, Allington M. Overview of the Electric Vehicle Market and the Potential of Charge Points for Demand Response [Technical Report]. London (UK): ICF Consulting Services; 2016 Mar 10.

Frade I, Ribeiro A, Gonçalves G, Antunes AP. Optimal location of charging stations for electric vehicles in a neighborhood in Lisbon, Portugal. *Transp Res Rec*. 2011;2252(1):91-8.

Wang H, Huang Q, Zhang C, Xia A. A novel approach for the layout of electric vehicle charging station. In: *Proceedings of the 2010 International Conference on Apperceiving Computing and Intelligence Analysis Proceeding*; 2010 Dec 17-19; Chengdu, China. 2010. p. 64-70.

Chen D, Khan M, Kockelman K. The Electric Vehicle Charging Station Location Problem: A Parking-Based Assignment Method for Seattle. In: *Proceedings of the 92nd Annual Meeting of the Transportation Research Board*; 2013 Jan 9-13; Washington, DC, USA. 2013. Under review.

He F, Wu D, Yin Y, Guan Y. Optimal deployment of public charging stations for plug-in hybrid electric vehicles. *Transp Res Part B Methodol*. 2013;47:87-101.

Xiong Y, Gan J, An B, Miao C, Bazzan ALC. Optimal Electric Vehicle Charging Station Placement. In: *Proceedings of the 24th International Conference on Artificial Intelligence*; 2015 Jul 25-31; Buenos Aires, Argentina. Palo Alto (CA): AAAI Press; 2015. p. 2662-8.

Xiong Y, Gan J, An B, Miao C, Bazzan ALC. Optimal Electric Vehicle Fast Charging Station Placement Based on Game Theoretical Framework. *IEEE Trans Intell Transp Syst*. 2018;19(8):2493-504.

Xiong Y, Gan J, An B, Miao C, Soh YC. Optimal pricing for efficient electric vehicle charging station management. In: *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*; 2016 May 9-13; Singapore. 2016. p. 749-57.

Chawla S, Hartline JD, Malec DL, Sivan B. Multi-parameter mechanism design and sequential posted pricing. In: *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*; 2010 Jun 5-8; Cambridge, MA, USA. 2010. p. 311-20.

Garcia F, Rachelson E. Markov Decision Processes. In: *Markov Decision Processes in Artificial Intelligence*. Hoboken (NJ): John Wiley & Sons, Ltd.; 2013. Chapter 1, p. 1-38.

Celebi E, Fuller JD. Time-of-Use Pricing in Electricity Markets Under Different Market Structures. *IEEE Trans Power Syst*. 2012;27(3):1170-81.

Basmadjian R. Optimized Charging of PV-Batteries for Households Using Real-Time Pricing Scheme: A Model and Heuristics-Based Implementation. *Electronics*. 2020;9(1):113.

Park SC, Jin YG, Yoon YT. Designing a Profit-Maximizing Critical Peak Pricing Scheme Considering the Payback Phenomenon. *Energies*. 2015;8(10):11363-79.

Lee S, Choi DH. Dynamic pricing and energy management for profit maximization in multiple smart electric vehicle charging stations: A privacy-preserving deep reinforcement learning approach. *Appl Energy*. 2021;304:117754.

Gu Z. Proposing a room pricing model for optimizing profitability. *Int J Hosp Manag*. 1997;16(3):273-7.

Subramanian J, Stidham S, Lautenbacher CJ. Airline Yield Management with Overbooking, Cancellations, and No-Shows. *Transp Sci*. 1999;33(2):147-67.

Ban D, Michailidis G, Devetsikiotis M. Demand response control for PHEV charging stations by dynamic price adjustments. In: *Proceedings of the 2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*; 2012 Jan 16-20; Washington, DC, USA. 2012. p. 1-8.

Bhattacharya S, Kar K, Chow JH, Gupta A. Extended Second Price Auctions With Elastic Supply for PEV Charging in the Smart Grid. *IEEE Trans Smart Grid*. 2016;7(4):2082-93.

Kim Y, Kwak J, Chong S. Dynamic Pricing, Scheduling, and Energy Management for Profit Maximization in PHEV Charging Stations. *IEEE Trans Veh Technol*. 2017;66(2):1011-26.

Limmer S, Rodemann T. Multi-objective optimization of plug-in electric vehicle charging prices. In: *Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI)*; 2017 Nov 27-Dec 1; Honolulu, HI, USA. 2017. p. 1-8.

Mrkos J, Komenda A, Jakob M. Revenue Maximization for Electric Vehicle Charging Service Providers Using Sequential Dynamic Pricing. In: *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*; 2018 Jul 10-15; Stockholm, Sweden. IFAAMAS; 2018. p. 9.

Basmadjian R, Kirpes B, Mrkos J, Cuchý M. A Reference Architecture for Interoperable Reservation Systems in Electric Vehicle Charging. *Smart Cities*. 2020;3(4):1405-27.

Basmadjian R, Kirpes B, Mrkos J, Cuchý M, Rastegar S. An Interoperable Reservation System for Public Electric Vehicle Charging Stations: A Case Study in Germany. In: *Proceedings of the 1st ACM International Workshop on Technology Enablers and Innovative Applications for Smart Cities and Communities*; 2019 Nov 13-14; New York, NY, USA. New York (NY): ACM; 2019. p. 22-9.

Basmadjian R, Niedermeier F, de Meer H. Modelling Performance and Power Consumption of Utilisation-Based DVFS Using M/M/1 Queues. In: *Proceedings of the Seventh International Conference on Future Energy Systems*; 2016 Jun 21-24; Waterloo, ON, Canada. New York (NY): ACM; 2016.

Basmadjian R, de Meer H. Modelling and Analysing Conservative Governor of DVFS-Enabled Processors. In: *Proceedings of the Ninth International Conference on Future Energy Systems*; 2018 Jun 12-15; Karlsruhe, Germany. New York (NY): ACM; 2018. p. 519-25.

Browne CB, Powley E, Whitehouse D, Lucas SM, Cowling PI, Rohlfshagen P, et al. A Survey of Monte Carlo Tree Search Methods. *IEEE Trans Comput Intell AI Games*. 2012;4(1):1-43.

MCTS.jl [Internet]. JuliaPOMDP; 2022 [cited 2022 Feb 7]. Available from: <https://github.com/JuliaPOMDP/MCTS.jl>

Auer P, Cesa-Bianchi N, Fischer P. Finite-time analysis of the multiarmed bandit problem. *Mach Learn*. 2002;47(2-3):235-56.

Coulom R. Efficient selectivity and backup operators in Monte-Carlo tree search. In: Proceedings of the International Conference on Computers and Games; 2006 May 29-31; Turin, Italy. 2006. p. 72-83.

Mausam, Kolobov A. Planning with Markov Decision Processes: An AI Perspective. San Rafael (CA): Morgan & Claypool Publishers; 2012. Vol. 6, p. 1-210.

Ke J, Zhang D, Zheng H. An Approximate Dynamic Programming Approach to Dynamic Pricing for Network Revenue Management. *Prod Oper Manag*. 2019;28(11):2719-37.

Gurobi Optimization LLC. Gurobi Optimizer Reference Manual [Internet]. 2020 [cited 2022 Feb 7]. Available from: <https://www.gurobi.com/>

Basmdjian R, Shaafieyoun A, Julka S. Day-Ahead Forecasting of the Percentage of Renewables Based on Time-Series Statistical Methods. *Energies*. 2021;14(21):7443.

Wittman MD, Belobaba PP. Customized dynamic pricing of airline fare products. *J Revenue Pricing Manag*. 2018;17(2):78-90.