On Efficient Operation of a V2G-Enabled Virtual Power Plant

When Solar Power Meets Bidirectional Electric Vehicle Charging

Saidur Rahman
saidur@ualberta.ca
University of Alberta
Canada

Linda Punt
punt@rsm.nl
Erasmus University
Netherlands

Omid Ardakanian
ardakanian@ualberta.ca
University of Alberta
Canada

Yashar Ghiassi
y.ghiassi@rsm.nl
Erasmus University
Netherlands

Xiaoqi Tan
xiaoqi.tan@ualberta.ca
University of Alberta
Canada

Abstract

Virtual power plants (VPP) can increase reliability and efficiency of power systems with a high share of renewables. However, their adoption largely depends on their profitability, which is difficult to maximize due to the heterogeneity of their components, different sources of uncertainty and potential profit streams. This paper proposes two profit-maximizing operating strategies for a VPP that aggregates solar systems and electric vehicle (EV) chargers with vehicle-to-grid (V2G) support, and generates profit by trading energy in day-ahead and imbalance electricity markets. Both strategies solve a two-stage stochastic optimization problem. In the first stage, energy bids are placed by solving a sequence of linear programs, each formulated for a specific forecast scenario. In the second stage, given the day-ahead commitments and real-time measurements, the decisions with respect to charging or discharging EVs are made sequentially for every hour and adjustments to the day-ahead commitments are settled in the imbalance market. The two strategies differ in how they solve the sequential decision making problem in the second stage. But, they both foresee the effect of their current (dis)charge decisions on the feasibility of fulfilling the EV charging demands using a one-step lookahead technique. The first strategy employs a heuristic algorithm to find a feasible charging schedule for every EV that is connected to a charger. The second one utilizes a soft actor-critic reinforcement learning method with a differentiable projection layer that enforces constraint satisfaction. We empirically evaluate the proposed operating strategies using real market prices, solar traces, and EV charging sessions obtained from a network of chargers in the Netherlands, and analyze how the uptake of V2G could affect profitability of this VPP.

CCS Concepts

• **Mathematics of computing** → *Mathematical optimization*; • **Computing methodologies** → *Reinforcement learning*.

ACM Reference Format:
Saidur Rahman, Linda Punt, Omid Ardakanian, Yashar Ghiassi, and Xiaoqi Tan. 2022. On Efficient Operation of a V2G-Enabled Virtual Power Plant: When Solar Power Meets Bidirectional Electric Vehicle Charging. In *The 9th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys '22), November 9–10, 2022, Boston, MA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3563357.3564067

1 Introduction

Future power systems will heavily rely on distributed energy resources (DERs) as they provide energy at lower cost than the electric grid, and enable greater resilience during adverse grid events. These resources, which generate, store, or reshape energy profiles, can be classified into five types: distributed generation units (*e.g.*, solar systems), battery storage, electric vehicle (EV) charging stations, grid-interactive appliances, and power-to-heat resources (*e.g.*, heat pumps and thermal storage) [25]. Massive DER growth is expected in the next several years. The cumulative DER capacity in the United States will reach 387GW by 2025 [7]. In Europe, DERs will provide 100GW of demand response, and in Australia, they will supply 30% to 45% of the total electricity demand by 2050 [19].

With the growing adoption of DERs, the concept of a virtual power plant (VPP) has become increasingly popular [9]. A VPP aggregates and orchestrates disparate DERs through sensing, communication, and control technology, to provide various services to the grid and increase the value of the DERs. For example, Tesla [6] and Swell Energy [4], in partnership with local utility companies, have implemented VPPs to support the grid by aggregating energy storage and solar systems in residential buildings. A novel VPP implementation that we study in this paper consists of solar systems, a fleet of EVs, and charging stations with vehicle-to-grid (V2G) support. This VPP combines renewable generation with mobile energy storage that can be charged and discharged subject to various constraints, such as fulfilling the EV charging demand by some deadline. Such a combination has been shown to enhance the mutual benefits of solar generation and flexibility of the EV fleet [34, 35], yet the uncertainty in solar production and EV mobility together with the large number of decision variables have hindered the progress towards an optimal operating strategy. There are a few pilot projects of this type of VPP which are still in early stages (*e.g.,* the project in Utrecht [8] will combine 2,000 solar panels, 250 bidirectional chargers, and a car-sharing fleet).

The VPPs can trade their aggregate energy in different stages of electricity markets, such as the wholesale market (day-ahead and intra-day), ancillary service, and capacity markets [25]. Participation in the wholesale electricity market, in particular, has attracted more attention [37] due to its simple form of bidding, higher predictability, manageable types of (energy) commitments, and the fact that it is by far the largest electricity market today. The participation of VPPs in this market could reduce wholesale prices considerably and cut end users' electricity bills. For example, Tesla's VPP in South Australia [3] can reduce the annual electricity bill of a typical customer by 30% by trading in the wholesale market. Inspired by

BuildSys '22, November 9–10, 2022, Boston, MA, USA

ACM ISBN 978-1-4503-9890-9/22/11…$15.00
https://doi.org/10.1145/3563357.3564067

this, we consider a setting in which the VPP trades (buys/sells) energy in the day-ahead (DA) market. Due to the uncertainty in solar power and EV charging demand, the amount of energy provided by the VPP in real time may not match its DA commitment. In this case, real-time deviations from the DA commitments are adjusted by the system operator, and the resulting financial cost or profit in the imbalance (IM) market is transferred to the VPP operator. Figure 1 illustrates the VPP's participation in these two markets. In the DA market, the (price-taker) VPP operator places an *energy bid* for every hour of the next day according to its predictions of the available solar energy, DA and IM market prices, and EV charging demands. On the operation day (*i.e.*, the next day), deviations from the DA bids are adjusted in the IM market according to imbalance prices. The price volatility in these markets further complicates the design of an optimal operating strategy for this VPP.

The VPP operator can take advantage of V2G to shape the EV charging demand to increase its profit. We study the setting in which the VPP operator owns and operates the EV fleet, thus it does not collect payment from the EVs for charging their battery, nor does it compensate them for participating in V2G. Its profit solely depends on the amount of energy traded in the two markets in each hour. An example of this VPP is a car-sharing or taxi company that owns a fleet of EVs in addition to bidirectional chargers and solar systems installed across the city (as depicted in Figure 1). The optimal operation of this VPP is a complex, two-stage stochastic optimization problem due to time-varying constraints and high uncertainty that can be attributed to intermittent solar generation, mobility and energy demand of EVs, and electricity market prices. We develop practical operating strategies for this VPP that involve (a) bidding in the DA market according to the average solution of a sequence of linear programs defined for different *forecast scenarios*, *i.e.*, realizations of random variables; (b) charging or discharging EVs in the next day via an online algorithm to maximize the VPP's profit, while ensuring the demand of every EV can be met before it disconnects from the charger. Our contribution is threefold:

(1) **Developing efficient operating strategies:** We propose two profit-maximizing strategies for the VPP described above. Both strategies place energy bids in the DA market by solving a number of linear programs for different forecast scenarios and taking the average of the solutions. This wait-and-see (WS) approach [33] is computationally efficient and yields a good approximation to the solution of the original stochastic program. To hedge against the uncertainty of solar generation and satisfy the charging constraints, a decision making problem is solved in an online fashion to (dis)charge the connected EVs and trade energy in the IM market. One strategy solves this problem using a heuristic algorithm, while the other adopts a policy learned via reinforcement learning. Nevertheless, they both perform a *laxity lookahead* (defined in Section 5) to ensure that the problem remains feasible if they take a specific action at the present time.

(2) **Evaluating the VPP operating strategies using real data:** Using real EV charging sessions from a network of chargers in the Netherlands along with market prices and solar traces from the same region, we show that the proposed strategies fulfill the energy demand of each EV by its deadline and outperform today's best practice of EV charging, which charges EVs at the maximum power as soon as they arrive, without taking advantage of V2G. Despite considerable uncertainty, using the reinforcement-learning-based strategy, the VPP can earn up to 59.6% of the profit that could be earned with prefect information (ignoring all sources of uncertainty).

(3) **Analyzing the economic implication of V2G:** We evaluate the efficacy of the proposed strategies as the number of EVs that participate in V2G increases. We show that V2G can increase the VPP's profit by more than 42% compared to the case where V2G is not supported at all.

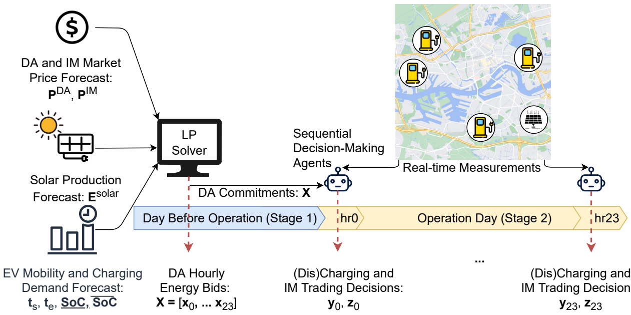

DA and IM Market
Price Forecast:
$\mathbf{P}^{DA}$, $\mathbf{P}^{IM}$
Solar Production
Forecast: $\mathbf{E}^{solar}$
EV Mobility and Charging
Demand Forecast:
t_s, t_e, $\underline{SoC}$, $\overline{SoC}$
LP
Solver
DA Commitments: $\mathbf{X}$
Sequential
Decision-Making
Agents
Real-time Measurements
Day Before Operation (Stage 1)
hr0
Operation Day (Stage 2)
hr23
DA Hourly
Energy Bids:
$\mathbf{X} = [\mathbf{x}_0, \dots \mathbf{x}_{23}]$
(Dis)Charging and
IM Trading Decisions:
$\mathbf{y}_0$, $\mathbf{z}_0$
...
(Dis)Charging and
IM Trading Decisions:
$\mathbf{y}_{23}$, $\mathbf{z}_{23}$

Figure 1: An illustration of the two-stage optimization problem. Solid lines show information flow. Dashed lines show decisions regarding market operation and EV charging.

The rest of the paper is organized as follows. Section 2 summarizes the related work on VPP implementation and operation. Section 3 introduces actor-critic methods and differentiable projection layers. Section 4 states our assumptions and formulates the optimization problem, Section 5 outlines the two operating strategies that we propose, and Section 6 introduces our datasets and baselines. We compare the proposed operating strategies with the baselines in Section 7. We conclude the paper in Section 8.

2 Related Work

2.1 Types and trading platforms of VPPs

Extensive research has been conducted on a VPP that incorporates some form of energy storage [22, 32, 36]. Bagchi et al. [12] quantify the additional gain of adding a stationary energy storage system to a VPP. A VPP that integrates EVs without V2G has been the subject of many studies too (see for instance [16, 27, 42]). Fewer studies consider a VPP that integrates renewable generation and EV chargers with V2G capability [26, 41]; these are the closest work to ours. We study the same type of VPP in a more practical setting and propose efficient operating strategies (for bidding and smart charging) that honor day-ahead commitments and guarantee the fulfilment of the EV charging demands, an important requirement, especially for V2G, that was not considered in previous work.

The VPP operator is typically assumed to be profit seeker, with a few exceptions such as [12] where the VPP aims to become an energy independent entity. Most related work considers the wholesale electricity market (typically the DA market) as the primary trading platform for the VPP operator. However, a simplified version of the wholesale market with a single stage and exogenous hourly prices is commonly considered [16, 22, 41]. Only a small number of papers envisage a two-stage model of trading in the electricity markets by accounting for the bidding in the DA market and the energy adjustments made in the IM and/or reserve market [12, 26, 32, 42]. We also adopt the two-stage trading model where the VPP operator trades energy in the DA market and settles its adjustments in the IM market. Despite this similarity, the VPP that we study has a unique configuration that includes EVs, chargers with V2G support, and solar systems. The interactions among these DERs make the design of the operating strategies more complicated. Besides, we quantify the additional gain that V2G provides in this type of VPP.

2.2 The VPP operation strategy

The optimal control of DERs in a VPP can be viewed as a decision making problem under uncertainty and risk. Thus, a wide range of techniques, from stochastic dynamic programming to robust optimization, can be applied to optimize the VPP operation. For example, a distributionally robust chance-constrained model is proposed in [43] to control several HVAC systems to absorb as much solar generation as possible. In another work, a chance-constrained energy management model is proposed in [39] to optimally control renewable generation and battery energy storage systems in a microgrid. Despite providing theoretical guarantees, robust optimization methods usually find overly conservative strategies as they do not take full advantage of the underlying data distribution.

Model Predictive Control (MPC) is another approach that has been used to control DERs in an online fashion. In this approach, a model is utilized to predict the system dynamics and changes in the environment. Vasirani et al. [41] adopt MPC to decide on the operation of a VPP that integrates wind turbines and EVs (with V2G) in an intra-day market. Distributed MPC is used in [13] to coordinate renewable generation in one control area with storage in another area. To lower the computational cost of MPC, a neural network is trained in [29] to approximate the control policy of an MPC. This neural network is then used in an online fashion to control a solar-plus-battery system. In addition to having high computational cost, a major drawback of the MPC-based approaches is the need for an accurate predictive model. In our problem, the MPC-based approach performs poorly, because market prices, and in particular imbalance prices, are highly variable and depend on various factors that cannot be accurately modelled. Moreover, mispredictions could result in violation of the charging constraints.

Different types of Reinforcement Learning (RL) have been used in recent years to solve control tasks in the energy domain that have continuous and high dimensional action spaces. In particular, policy gradient and actor-critic methods are used to control the charging of EVs and stationary batteries [11, 38]. Model-free RL methods are advantageous for DER control because they do not require a separately trained model of the system dynamics. Instead, the RL agent continuously interacts with the environment to learn an *optimal* policy that governs the operation of DERs. Traditional RL algorithms suffer from one major issue during training and when deployed in the real world: they may take actions that violate operational or physical constraints [21]. In the context of controlling DER, there are usually several *hard constraints* that must be satisfied at all times. It is critical to ensure that they are satisfied in the deployment phase, and during training if it takes place in the real environment rather than a simulator. To address this shortcoming, various *safe-RL* techniques have emerged in the literature [24, 40]. They enforce bounds on the agent's actions to satisfy the hard constraints. Bingqing et al. [15] show that a deep reinforcement learning agent trained with a differentiable projection layer embedded in the neural network can safely control inverters and building energy systems. To our knowledge, the application of safe-RL techniques to smart EV charging has not been explored in the literature yet. In this context, the charging deadlines and operational constraints of the battery can be viewed as hard constraints.

3 Reinforcement Learning

In RL, interactions between a decision-making agent and its surrounding environment are modelled as a Markov Decision Process (MDP) $(\mathcal{S}, \mathcal{A}, R, P, \gamma, \mu)$, where $\mathcal{S}$ is the set of states, $\mathcal{A}$ is the set of actions, R is the reward function, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ describes the next state distribution as a function of the current state and action, $\gamma \in [0, 1]$ is the discount factor, and $\mu : \Delta(\mathcal{S})$ is the distribution of starting states. Here, $\Delta(\mathcal{S})$ denotes the probability simplex over states. The interaction between the RL agent and the environment is as follows: at step t, the agent observes a state vector s_t and selects an action a_t according to a policy $\pi(\cdot|s_t)$, which is used to interact with the environment. The environment returns a reward r_{t+1} and the next state s_{t+1} to the agent. The goal of the RL agent is to learn a policy that maximizes its expected discounted cumulative reward (or return). We define the return at time t as $G_t \doteq \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$, and the value function $v^{\pi}(s)$ as the expected return when starting at s and following the policy π, given by: $v^{\pi}(s) \doteq \mathbb{E}_{\pi} [G_t \mid S_t = s]$. When the state space is continuous or large, the state value function is approximated as $v^{\pi}(s) \approx v_w(s)$, where $w \in \mathbb{R}^d$ collects the parameters of this approximation. When deep neural networks are used for function approximation, we refer to it as *deep reinforcement learning*.

Actor Critic Methods They are RL methods that learn the approximation to the policy in addition to the approximation to the value function. The *actor* refers to the policy and the *critic* refers to the value function learned by the agent. The main advantage of actor critic methods is their ability to tackle continuous action spaces. In our work, we use the Soft-Actor Critic (SAC) algorithm [23], which is an off-policy, maximum-entropy policy gradient method, using a stochastic actor. We explain the SAC algorithm later in Section 5.

Safe-RL via Differentiable Projection The RL algorithms could violate the hard constraints of the actuator or the environment during training (as they explore the space) and/or after deployment. Clipping or hand-crafted projection of the action taken by the RL agent to a *safe* region (aka the feasible set), can prevent the violation of hard constraints when the policy is used in practice. However, this does not guarantee that the resulting safe action is *optimal* too. Furthermore, the RL agent does not actually *learn* the hard constraints, hence the actions must be mapped to the safe region even after deployment.

To make the RL agent effectively *learn* the hard constraints without sacrificing optimality, we embed an optimization problem [10] that projects a point onto the safe region, in the neural network[1] used for approximating the policy function in the actor-critic method. The optimization problem (described below) is a

[1] The optimization problem can be implemented as a neural network layer using a Python library called cvxpylayers: https://github.com/cvxgrp/cvxpylayers.

differentiable layer within the actor network, allowing the RL agent to learn the hard constraints through backpropagation. Consider the ℓ_2-norm projection $\mathcal{P}_\mathcal{S} : \mathbb{R}^n \to \mathcal{S}$ which maps a point in $\hat{a} \in \mathbb{R}^n$ to the point closest to it in a constraint set $\mathcal{S} \subseteq \mathbb{R}^n$, as shown below:

$$\mathcal{P}_\mathcal{S}(\hat{a}) = \underset{a \in \mathcal{S}}{\operatorname{argmin}} \|a - \hat{a}\|_2^2 \, . \qquad (1)$$

This is a convex optimization problem if $\mathcal{S}$ is a convex set. Thus, it can be solved using a standard solver, which constitutes the forward procedure in the neural network. The backward procedure is constructed using implicit function theorem [31] where the gradients of the solution variables of (1) are calculated by differentiating the Karush-Kuhn Tucker (KKT) conditions at the solution. The respective gradients are then propagated back to the neural network. As a result, the agent learns the hard constraints in $\mathcal{S}$ through the parameter updates done when training the neural network.

4 Problem Statement

We consider a VPP that consists of a fleet of EVs, solar systems and bidirectional chargers distributed across the city, all of which are owned by the VPP operator. The EVs visit charging stations, each containing multiple chargers, at random times to replenish their battery. They stay there for a certain amount of time before they start their next trip. This determines their charging deadline. We assume an EV's utility remains the same as long as it is charged to the desired level by this deadline, and that bidirectional chargers and solar inverters do not cause overloading or voltage violation problems in the distribution grid. We consequently ignore the grid constraints when optimizing the VPP operation. Moreover, we assume that the VPP operator is not certain about the arrival time, charging deadline, and energy demand of an EV before it arrives at a charging station. However, once an EV arrives, it communicates its energy demand and departure time (or deadline) to the VPP. It is also assumed that the VPP operator allows a subset of the EV fleet to participate in V2G, for example they can be the EVs whose battery has a high remaining cycle life.

Because of the DERs that comprise the VPP, its demand and supply are both variable and flexible. The VPP operator trades (*i.e.*, buys or sells) energy in the wholesale day-ahead (DA) market, which is a pool-based energy market. The players in this market, place separate energy-price bids for every hour of the next day, and each hour has an independent auction. DA markets are typically cleared based on the uniform pricing mechanism, hence the same (clearing) price applies to any market player (seller or buyer) whose bid is accepted. We further assume that the VPP operator is a price taker, which is reasonable given the size of a typical DA market today. Since the marginal cost of supplying power by the VPP is much lower than conventional generators that typically govern the clearing prices, the VPP's energy-price bid reduces to an energy (quantity) bid as the corresponding price does not affect the price it receives. This is a practical assumption for battery operators and aggregators in electricity markets [14, 28]. With this assumption, we can treat the DA prices as exogenous random variables, denoted by $\mathbf{P}^{DA} = [P_0^{DA}, \cdots, P_{23}^{DA}]$, and postulate that the VPP operator only bids for quantity (and not for price). At the time of operation, the DA market players might deviate from their DA commitments. Any such deviation must be financially settled through the imbalance (IM) market. The IM market prices reflect the additional costs incurred to serve the unexpected demand beyond wholesale commitments. As such, these prices are treated as exogenous random variables too. We assume the IM market adopts the single-pricing (aka one-pricing) model [20] – the most prevalent pricing scheme in the imbalance market today. In this model, selling and buying prices in each hour t are identical, denoted by P_t^{IM}. The vector $\mathbf{P}^{IM} = [P_0^{IM}, \cdots, P_{23}^{IM}]$ represents IM prices for one day.

In the DA bidding stage, the VPP operator does not deterministically know the market prices, available solar energy in every hour of the next day, and the EV arrival and demand patterns in the next day. It must submit a vector of energy bids $\mathbf{X} = [x_0, \cdots, x_{23}]$ to the market, where x_t is positive when the VPP commits to sell energy in hour t of the next day and is negative when the VPP commits to buy energy in that hour. On the operation day, the VPP must schedule the charge and discharge of EVs that arrive at the charging stations. We denote this schedule by $\mathbf{Y}^n = [y_0^n, \cdots, y_{23}^n]$ with n being the index into the set of EVs that arrive in the next day ($n \in \mathcal{N} = \{1, 2, ..., N\}$). Hence, every element y_t^n represents the amount of energy stored in (positive sign) or withdrawn from (negative sign) the battery of the n^{th} EV in hour t. When EV n is not in a charging station, the respective elements of $\mathbf{Y}^n$ are set to zero.[2] Note that for a given vector $\mathbf{X}$ and a set of vectors $\mathbf{Y}^n$ ($\forall n \in \mathcal{N}$), the amount of energy that must be traded in the IM market, denoted by $\mathbf{Z} = [z_0, \cdots, z_{23}]$, can be computed from the energy balance equation, which states that the VPP's demand and supply must be equal in each hour. The VPP operator will buy enough energy from the two markets to satisfy the energy demand of all EVs by their deadlines. Additionally, the operator can sell energy (solar production and/or energy discharged from the batteries) in the markets. These decisions must be made so as to maximize the expected profit of the VPP operator. Energy bids are submitted to the DA market all at once on the day before the operation day, whereas (dis)charging and trading decisions in the IM market are made for every hour in an online fashion.

Optimal VPP Operation Strategy: The optimal VPP operation is the solution of a two-stage stochastic optimization problem. We decompose this problem into two subproblems, namely stage 1 and stage 2. The stage 1 problem determines the optimal bidding strategy in the DA market. This is not a sequential decision making problem as hourly bids are submitted all at once. The stage 2 problem entails finding feasible schedules for charging or discharging the EVs as they arrive at the charging stations, while trading in the IM market to close the gap between day-ahead bids and the realized solar generation minus the total realized EV charging demand. This can be cast as a sequential decision making problem. We formally define the two problems below:

Stage 1 Problem: Given different forecasts for (hourly) DA and IM market prices, hourly solar production, EV arrivals, stay times, and energy demands for every hour t of the next day, where $t \in \mathcal{T} = \{0, 1, , ..., 23\}$, the goal is to compute day-ahead energy bids, *i.e.*, $\mathbf{X} = [x_0, \cdots, x_{23}]$, that maximize the expected profit of the VPP as a result of participating in both markets. This problem is solved

[2] Without loss of generality, we assume EV arrivals and departures occur in the beginning of a 1-hour time slot as it is the timescale of trading in the IM market. In practice, the VPP can round the arrival or departure time to the nearest hour.

in an offline fashion, typically in the beginning of the day before the operation day.

Stage 2 Problem: Given the energy bids submitted to the DA market the day before (*i.e.*, the stage 1 solution), the current price of the IM market and IM market price forecast for every hour until the end of the day, the deadline and unmet energy demand of the EVs that are currently in the charging stations, and forecast data for EV arrivals, stay (sojourn) times, and energy demands in future time slots, this problem concerns determining the amount of energy that must be charged/discharged in/from the EV batteries in the current time slot to maximize the expected profit of the VPP, such that the energy demand of every EV is guaranteed to be satisfied by their deadline. In other words, the stage 2 problem concerns determining y_t^n for every EV that is currently present in the charging stations while ensuring that it is possible to fulfill their demand before their departure. Once these values are fixed in hour t, the net difference between demand and supply (z_t) will be traded in the IM market according to its current price.

5 Methodology

We explain how to solve the two problems introduced in the previous section. Specifically, we show that the stage 1 problem can be formulated as a stochastic Linear Program (LP) and its solution can be approximated by solving a sequence of deterministic LPs formulated for different forecast scenarios (*i.e.*, different hourly market prices, solar production levels, EV arrival times, and charging demands). Given the solution of stage 1, *i.e.*, the DA energy bids, we propose two algorithms to solve the stage 2 problem. Both utilize observations up to the current time slot of the operation day and make real-time decisions for (dis)charging EVs, and subsequently trading in the IM market. The decisions are made while ensuring that the charging deadlines can be met. Note stage 1 and stage 2 problems are solved in the VPP's in-house or cloud server.

5.1 Stage 1: Linear Programming

Let $\mathbf{E}^{solar} = [E_0^{solar}, \cdots, E_{23}^{solar}]$ be the available solar energy in every hour of the operation day, E_{max}^{solar} be the total peak generation capacity of the solar systems, $\mathcal{N}_t$ be the set of EVs that are connected to a charger owned by the VPP in hour t of the operation day, and $\mathcal{N} = \cup_{t\in\mathcal{T}}\{N_t\}$ be the set of all EVs that visit the charging stations on the operation day. Let $\mathcal{N}^D \subseteq \mathcal{N}$ denote the set of EVs that participate in V2G; hence, the operator can discharge their battery as long as it is possible to meet their charging demand by the deadline. We denote the EVs in $\mathcal{N}^D$ that are connected to chargers in hour t of the operation day by $\mathcal{N}_t^D \subseteq \mathcal{N}_t$. For the EV indexed by n, we denote the energy capacity of its battery by b_n, its arrival time by t_s^n, and the length of its charging session by τ^n. Hence, its departure time will be $t_e^n = t_s^n + \tau^n$. The maximum charge and discharge power supported by a charger are denoted by $\alpha_c, \alpha_d > 0$, respectively. We assume these maximum rates are the same for all chargers. Since the length of each time slot is one hour, we use the same notation (α_c, α_d) to represent the maximum amount of energy that can be stored or withdrawn from a battery in one time slot. The state-of-charge (SoC) of the battery of EV n at time $t \in \mathcal{T}$ is denoted by SoC_t^n, which is bounded by δ_{min} and δ_{max}. The charge and discharge efficiencies are denoted by η_c and η_d, respectively. Given the energy demand of the EV and its SoC upon arrival, denoted by $\underline{SoC}^n$, we calculate the target SoC, which is denoted by $\overline{SoC}^n$.

Let $\mathbf{P}^{DA}, \mathbf{P}^{IM}, \mathbf{E}^{solar}, \mathbf{t}_s, \mathbf{t}_e, \underline{\mathbf{SoC}}, \overline{\mathbf{SoC}}$ be random vectors that collect the random variables and Ω be the set of possible realizations of these random variables. The stage 1 problem maximizes the expected profit of the VPP operator subject to a set of constraints:

$$\underset{\mathbf{X},\mathbf{Y},\mathbf{Z},AC,AD}{\text{maximize}} \quad \mathbb{E}_{<\mathbf{P}^{DA},\mathbf{P}^{IM},\mathbf{E}^{solar},\mathbf{t}_s,\mathbf{t}_e,\underline{\mathbf{SoC}},\overline{\mathbf{SoC}}>\sim\Omega}\ \mathbf{X}^\top\mathbf{P}^{DA}+\mathbf{Z}^\top\mathbf{P}^{IM} \tag{2a}$$

$$\text{subject to} \quad -|\mathcal{N}_t|\cdot\alpha_c \le x_t \le E_{max}^{solar} + |\mathcal{N}_t^D|\cdot\alpha_d, \quad \forall t\in\mathcal{T} \tag{2b}$$

$$x_t + z_t + y_t = E_t^{solar}, \quad \forall t\in\mathcal{T} \tag{2c}$$

$$y_t = \sum_{n\in\mathcal{N}} y_t^n, \quad \forall t\in\mathcal{T} \tag{2d}$$

$$y_t^n = AC_t^n + AD_t^n, \quad \forall n\in\mathcal{N}_t, \forall t\in\mathcal{T} \tag{2e}$$

$$AD_t^n = 0, \quad \forall n\in\mathcal{N}_t\setminus\mathcal{N}_t^D, \forall t\in\mathcal{T} \tag{2f}$$

$$-\alpha_d \le AD_t^n \le 0, \quad \forall n\in\mathcal{N}_t^D, \forall t\in\mathcal{T} \tag{2g}$$

$$0 \le AC_t^n \le \alpha_c, \quad \forall n\in\mathcal{N}_t, \forall t\in\mathcal{T} \tag{2h}$$

$$SoC_{t+1}^n = SoC_t^n + \frac{AC_t^n\eta_c}{b_n} + \frac{AD_t^n}{\eta_d b_n}, \forall n\in\mathcal{N}_t, \forall t\in\mathcal{T} \tag{2i}$$

$$\delta_{min} \le SoC_t^n \le \delta_{max}, \quad \forall n\in\mathcal{N}_t, \forall t\in\mathcal{T} \tag{2j}$$

$$SoC_{t_s^n}^n = \underline{SoC}^n, \quad \forall n\in\mathcal{N}_t \tag{2k}$$

$$SoC_{t_e^n}^n = \overline{SoC}^n, \quad \forall n\in\mathcal{N}_t \tag{2l}$$

Recall that x_t is the DA energy bid placed for hour t of the next day and z_t is the amount of energy that would be traded in the IM market in that hour. The sign of z_t determines whether the VPP operator buys or sells in the IM market: positive implies selling and negative implies buying. Constraint (2b) is an operational constraint that defines the DA hourly selling and buying bid caps for the VPP operator. We assume that the selling bid cap in any hour is the sum of the peak solar generation capacity and the maximum amount of energy that can be discharged from the connected EVs that participate in V2G in that hour. Similarly, the buying bid cap in any hour is the maximum amount of energy that can be stored in the connected EVs in that hour. Constraint (2c) is the energy balance equation and constraint (2d) expresses the total charging demand as the sum of the demands of individual chargers within charging stations. Constraint (2e) splits the contribution of each EV n to the total charging demand in time slot t into two parts: energy stored in its battery AC_t^n, and energy withdrawn from its battery AD_t^n. This is necessary as charge and discharge efficiencies can be less than 1 in Constraint (2i). Constraints (2f)-(2h) set bounds for the amount of energy that can be stored or withdrawn from the battery of an EV. These bounds depend on the maximum charge and discharge rates supported by the chargers. For the EVs that do not participate in V2G, the amount of energy that can be withdrawn from their battery is set to zero. Constraint (2i) updates the SoC of each EV according to the amount of energy stored or withdrawn from its battery in the previous time slot and the respective inefficiency parameter. Finally, constraints (2j)-(2l) define bounds for the SoC of each EV, and assign a value to it at arrival and departure time. Observe that all constraints are affine in Problem (2).

Remark We do not need to introduce binary variables to ensure that AC_t^n and AD_t^n are not nonzero at the same time. This is because,

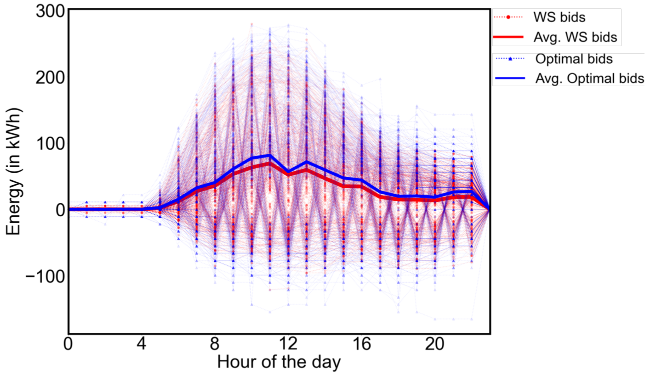


Figure 2: Comparison between the optimal DA bids (from the ORACLE baseline defined in Section 6.2.1) and DA bids computed using the WS approach over one year.

due to battery imperfections, a strategy that simultaneously charges and discharges the battery of one or multiple EVs will either increase the deficit energy that must be purchased from the IM market or decrease the surplus energy that could be sold in the IM market. Either way, assuming that IM prices are nonnegative, this strategy reduces the profit of the VPP and is therefore not optimal.

Solution of Stochastic LP Solving Problem (2) via sample-average approximation (SAA) [30] is overly costly due to the large number of random variables that appear in the objective function and some of the constraints, leading to a huge optimization problem even for a small number of samples. Thus, we opt for a practical wait-and-see (WS) approach [33] that approximates the solution of this stochastic linear program.[3] In this approach, we consider a large number of forecast scenarios and formulate a deterministic linear program (written below) for each forecast scenario $\omega \in \Omega$:

$$\begin{aligned} \underset{\mathbf{X},\,\mathbf{Y},\,\mathbf{Z},AC,AD}{\text{maximize}} \quad & \mathbf{X}^\top \mathbf{P}^{DA} + \mathbf{Z}^\top \mathbf{P}^{IM} \\ \text{subject to} \quad & \text{Constraints (2b) to (2l)} \end{aligned} \tag{3}$$

We solve these deterministic linear programs independently and take the average of the respective solutions to efficiently compute near-optimal DA bids. We discard the tentative smart charging and IM market trading schedules because they will be recalculated in an online fashion (in stage 2), using more accurate data.

To create different forecast scenarios, we add white Gaussian noise with standard deviation being 10% of the realized value to the realized value of each random variable, namely $\mathbf{P}^{DA}$, $\mathbf{P}^{IM}$, $\mathbf{E}^{solar}$, $\mathbf{t}_s$, $\mathbf{t}_e$, $\underline{\mathbf{SoC}}$, $\overline{\mathbf{SoC}}$. We have found empirically that considering 1,000 forecast scenarios $(\omega_1, \cdots, \omega_{1000})$ provides a good-quality solution[4] and the total running time of solving 1,000 deterministic LPs is less than 15 minutes on an Intel core-i9 server with 128GB of memory. Once the deterministic LPs are solved, we take the average of the respective solutions and treat this as the energy bids that will be submitted to the DA market. Figure 2 compares the optimal energy bids (assuming perfect information) with the energy bids computed efficiently using the WS approach. It can be seen that the difference between the average hourly energy bids is generally insignificant.

Algorithm 1: LLA for EV charging

```
1  $\mathcal{S}_1 \leftarrow$ FindEVsWithNegativeLaxity($\mathcal{N}_t$) ;    // lookahead
2  $e_t \leftarrow$ ChargeEVs($\mathcal{S}_1$);
3  $x_t \leftarrow x_t + e_t$;
4  if $x_t > E_t^{solar}$ then
5      $\mathcal{S}_2 \leftarrow$ FindEVsToDischarge($x_t - E_t^{solar}$);
6      $e_t \leftarrow$ DischargeEVs($\mathcal{S}_2$);
7      $x_t \leftarrow x_t$ - $e_t$;
8      BuyFromImbalanceMarket($x_t - E_t^{solar}$);
9  end
10 else if $x_t \leq E_t^{solar}$ then
11     $\mathcal{S}_3 \leftarrow$ FindEVsToCharge($E_t^{solar} - x_t$);
12     $e_t \leftarrow$ ChargeEVs($\mathcal{S}_3$);
13     $x_t \leftarrow x_t + e_t$;
14     SellToImbalanceMarket($E_t^{solar} - x_t$);
15 end
```

Note that the objective function of Problem (3) is linear and its constraints are affine. We model this deterministic LP in CVXPY [17] and solve it using Gurobi. Next, we propose two algorithms for solving the stage 2 problem given the DA energy bids, which is the solution of the stage 1 problem.

5.2 Stage 2: Laxity-Lookahead (LLA) Algorithm

The first algorithm we propose to solve the stage 2 problem is a heuristic algorithm, called Laxity-LookAhead (LLA). When we run LLA for hour t of the operation day, it computes the *laxity* of every EV that is currently in a charging station and uses this value to find out if the charging demand of this EV will be satisfied before its departure. This is essential for constraint enforcement. The laxity of an EV is defined as the maximum amount of time we can delay its charging, while still being able to charge its battery to the desired SoC by the deadline. Specifically, the laxity of EV n, with departure time t_e^n, battery size b_n, and target SoC, $\overline{SoC}^n$, in time slot t is:

$$lax_t^n = t_e^n - t - \frac{(\overline{SoC}^n - SoC_t^n) \cdot b_n}{\alpha_c \eta_c}. \tag{4}$$

Note that the laxity of an EV can be calculated deterministically in any time slot after its arrival, because its deadline and energy demand are communicated to the VPP upon arrival. Nevertheless, the EV's laxity is unknown before it arrives. The basic idea of this algorithm is to identify all EVs that will have a *negative* laxity in the next time slot if they are not charged in the current time slot. These EVs must be charged at the maximum charge power supported by the charger, otherwise the problem becomes infeasible. Next, depending on whether there is surplus solar energy in this time slot, other EVs are charged or discharged.

The LLA algorithm utilizes three main functions (see Algorithm 1). In Line 1, the FindEVsWithNegativeLaxity function returns the set of EVs that are presently connected to a charger and will have negative laxity in the next time slot if they are not charged in the current time slot. This set is denoted by $\mathcal{S}_1$ and is determined via a one-step laxity lookahead. Concretely, to calculate the laxity of an EV in the next time slot, we substitute t with $t + 1$

[3] It can be proved that WS yields a bound on SAA [30] as it interchanges the order of summation and maximization.

[4] Our experiment shows that the resulting DA bids do not vary noticeably if we consider more forecast scenarios. We omit the convergence analysis to save space.

in (4) and let SoC_{t+1}^n be equal to SoC_t^n. The set $\mathcal{S}_1$ is then passed to the ChargeEVs function (Line 2), which is responsible for charging these EVs at the maximum power supported by the charger. Once these EVs are charged, we add the energy delivered to these EVs to the day ahead commitment for this time slot, x_t, to update the amount of energy required in this time slot (Line 3). This value is then compared with the available solar energy in this time slot, E_t^{solar}. If the available solar energy is not enough to supply the demand (Line 4), we must discharge a subset of EVs or buy the deficit from the IM market. Otherwise (Line 10), we can charge a subset of EVs or sell the surplus in the IM market. LLA uses a simple heuristic to specify the order in which we use smart charging and trade in the IM market in both cases.

In the case that $x_t > E_t^{solar}$, the FindEVsToDischarge function gets the amount of deficit and returns the set of EVs, denoted by $\mathcal{S}_2$, that (a) participate in V2G, (b) have the highest laxity, and (c) their laxity will not become negative in the next time slot if we discharge them at the maximum power supported by the charger in this time slot by calling DischargeEVs. If there is not enough EVs in $\mathcal{S}_2$ to cover the deficit, we buy the remainder from the IM market according to the current market price (Line 8). This gives z_t. In the case that $x_t \leq E_t^{solar}$, the FindEVsToCharge function gets the amount of surplus and returns the set of EVs, denoted by $\mathcal{S}_3$ not intersecting with $\mathcal{S}_1$, that have the lowest laxity. These EVs are charged at the maximum power in this time slot by calling ChargeEVs. If there is not enough EVs in $\mathcal{S}_3$ to absorb the surplus energy, we sell the remainder to the IM market according to the current market price (Line 14). This gives z_t.

Since LLA is an online algorithm, it does not assume the knowledge of the available solar energy and IM market prices in the next time slots of the day, and future EV arrival times, stay times, and energy demands. Moreover, it does not find the optimal EV charging strategy because regardless of future charging demands and market prices it always prioritizes (a) discharging EVs with highest laxity that will not have negative laxity in the next time slot over buying the deficit energy from the IM market; (b) charging EVs with lowest laxity over selling the surplus energy to the IM market.

5.3 Stage 2: Laxity-Aware Soft Actor Critic (LA-SAC) Algorithm

The second algorithm we propose to solve the stage 2 problem is a model-free RL algorithm that respects the EV charging deadlines. This algorithm is designed based on SAC [23] and is called Laxity-Aware Soft Actor Critic (LA-SAC). It borrows the notion of laxity from LLA and incorporates one-step laxity lookahead in the differentiable projection layer embedded in the actor network. We now describe the MDP and explain how the projection layer is used to ensure safe exploration and convergence to the optimal policy.

5.3.1 MDP Formulation

State The state at time t, is a tuple s_t, where the first state variable is the moving average of solar generation for the particular time slot t in the past 3 days, denoted $\bar{E}_t^{solar}$. We use the moving average of the previous 3 days as one of the state variables to capture the diurnal pattern of solar generation and help the (model-free) RL agent to implicitly learn this temporal relationship. The second and third variables are the moving average of DA and IM market prices for the particular time slot t in the past 7 days, denoted $\bar{p}_t^{DA}$ and $\bar{p}_t^{IM}$ respectively. Although market prices change drastically over the course of the day, incorporating the average of previous-day market prices (in the same hour) could help the agent learn temporal patterns that might exist in the two markets. The rest of the state variables are the laxity and SoC level of each of the EVs present in the current time slot in a charging station, before the RL agent's (dis)charging action is implemented.

$$s_t = (\bar{E}_t^{solar}, \bar{p}_t^{DA}, \bar{p}_t^{IM}, \{lax_{t-1}^n\}_{n \in \mathcal{N}_t}, \{SoC_{t-1}^n\}_{n \in \mathcal{N}_t}) \tag{5}$$

Action The action space in continuous and multidimensional. Specifically, the action taken by the RL agent for every hour t forms a vector $a_t = [y_t^0, \cdots, y_t^{|\mathcal{N}_t|-1}]$ where y_t^n is the charge or discharge decision taken for the n^{th} EV that is present in one of the charging stations in this hour and satisfies the following conditions:

$$-\alpha_d \leq y_t^n \leq \alpha_c \qquad \forall n \in \mathcal{N}_t^D, \tag{6}$$

$$0 \leq y_t^n \leq \alpha_c \qquad \forall n \in \mathcal{N}_t \setminus \mathcal{N}_t^D. \tag{7}$$

Reward The reward obtained by the RL agent in hour t, denoted by r_t, is a scalar value calculated based on the profit that will be generated by taking actions that charge or discharge the EVs feasibly and trade the surplus or deficit in the IM market according to its current price. To help the agent in taking actions that result in the maximum return, we reward the agent as follows

$$r_t = z_t \cdot p_t^{IM} + \zeta \cdot \bar{lax}_t. \tag{8}$$

The first term is the immediate reward received by the agent. It depends on how much energy is traded in the IM market (*i.e.*, $z_t = E_t^{solar} - x_t - \mathbf{1}^\intercal a_t$) and the current market price. The second term is the average laxity of all the EVs that are present at the charging stations at time t, denoted as $\bar{lax}_t$, and a positive scaling factor ζ ($\zeta = 2$ in our experiments). As discussed earlier, the laxity of an EV characterizes the charging flexibility. Thus, the second term is included to incentivize the agent to take actions that do not significantly lower this flexibility on average.

5.3.2 Soft Actor Critic RL Agent

The SAC algorithm [23] is a policy gradient algorithm which is more suitable for tackling continuous action space problems. In this case, a parameterized function (*i.e.*, a neural network) denoted π_θ, represents the policy of the RL agent. The policy parameter is updated towards the gradient direction of a performance function $J(\theta)$, as follows: $\theta \leftarrow \theta + \gamma \nabla_\theta J(\theta)$. The SAC algorithm encourages further exploration of the agent by incorporating an entropy measure in the reward function. The objective is to maximize both the entropy and expected return. The performance function is given by: $J(\theta) = \sum_{t=1}^{T} \mathbb{E}_{(s_t,a_t)\sim\rho_\pi}[r_t + \nu\mathcal{H}(\pi(.|s_t))]$. Here, ρ_π denotes the marginal distribution for the state-action pairs (s_t, a_t) sampled from the policy π. The entropy measure and entropy importance are denoted by $\mathcal{H}$ and ν, respectively. We refer the readers to [23] for further details about SAC, including the policy and value functions along with their gradients. To enable the RL agent to do safe exploration and learn the hard constraints, we redefine the loss function as explained in the next section. The Adam optimizer is used with a learning rate of 0.0001, the discount factor is set to be 0.99, and the batch size is set to 72. We allow automatic entropy tuning, which balances exploitation and exploration for the agent.

5.3.3 Safe Projection Layer

The action selected according to $\hat{\pi_\theta}$, which is the neural network that approximates the policy and is parameterized by θ, may not satisfy the hard constraints in our problem (*i.e.*, meeting charging demands by the respective deadlines). To get a *safe* policy, the output of this neural network is passed to a differentiable projection layer $\mathcal{P}$ that maps the action to the safe region by solving an optimization problem. Hence, we write the safe policy as $\pi_\theta(s_t) = \mathcal{P}_\mathcal{S}\big(\hat{\pi_\theta}(s_t)\big)$ where $\mathcal{S}$ is the safe region, *i.e.*, the feasible set of Problem (10) described below. The SAC agent is trained to minimize the following loss function:

$$\mathcal{L}(\theta, s_t) = -J(\theta) + \xi \|\pi_\theta(s_t) - \hat{\pi_\theta}(s_t)\|_2^2\,, \tag{9}$$

where ξ is a non-negative hyper-parameter. We add $\mathcal{L}(\theta, s_t)$ to the policy function equation and to the function used for calculating the loss of automatic entropy tuning [23]. As a result, the hard constraints of our problem formulation are learned by the neural network of the SAC agent via backpropagation.

The optimization problem that becomes the differentiable projection layer within the actor network of SAC is written below:

$$\begin{aligned}
\underset{a_t, AC, AD}{\text{minimize}} \quad & \|a_t - \tilde{a_t}\|_2^2 && \text{(10a)}\\
\text{subject to} \quad & AD_t^n = 0, \qquad \forall n \in \mathcal{N}_t \setminus \mathcal{N}_t^D && \text{(10b)}\\
& -\alpha_d \le AD_t^n \le 0, \qquad \forall n \in \mathcal{N}_t^D && \text{(10c)}\\
& 0 \le AC_t^n \le \alpha_c, \qquad \forall n \in \mathcal{N}_t && \text{(10d)}\\
& y_t^n = AC_t^n + AD_t^n, \qquad \forall n \in \mathcal{N}_t && \text{(10e)}\\
& SoC_{t+1}^n = SoC_t^n + \frac{AC_t^n \eta_c}{b_n} + \frac{AD_t^n}{\eta_d b_n}, \quad \forall n \in \mathcal{N}_t && \text{(10f)}\\
& \delta_{min} \le SoC_{t+1}^n \le \delta_{max}, \qquad \forall n \in \mathcal{N}_t && \text{(10g)}\\
& lax_{t+1}^n \ge 0, \qquad \forall n \in \mathcal{N}_t && \text{(10h)}
\end{aligned}$$

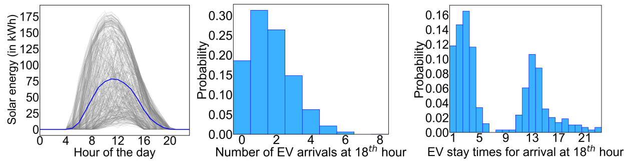
In this formulation, $\tilde{a_t}$ is the pre-projection action vector, *i.e.*, the set of actions taken by the RL agent concerning the connected EVs, before it is passed to the projection layer in the neural network. This optimization problem finds the post-projection action vector a_t that ensures the charging problem remains feasible for each EV and has the minimum Euclidean distance from the pre-projection action vector. The constraints define bounds for charge and discharge rates, and the SoC of batteries. The last constraint is to force the laxity of every EV to remain non-negative in the next hour if we implement a_t in this hour. Here lax_{t+1}^n can be defined by plugging in SoC_{t+1}^n in Equation (4) and replacing t with $t+1$. This ensures the problem remains feasible and all charging deadlines can be met.

6 Evaluation

We describe the datasets used for training and evaluation of the proposed VPP operating strategies, and explain our two baselines.

6.1 Dataset Preparation

We combine four datasets that contain real solar traces, DA market prices, IM market prices, and EV charging sessions between January 1, 2020 and December 31, 2020, to create a test dataset that is used to evaluate the proposed VPP operating strategies and baselines (described in the next section). All these datasets pertain to the same region in Rotterdam, Netherlands. Specifically, we pull hourly solar irradiance data via the Solcast API,[5] using latitude and longitude of an arbitrary location in Rotterdam. This irradiance data is fed to the PVWatts model [18] to compute the power generated by a PV system located at these coordinates. The tilt angle of the panels is set to 51 degrees and their orientation angle to 270 degrees. The size of the PV system is defined according to the maximum EV charging demand, which is 200 kW-peak in our study. Figure 3a shows the daily solar production curves of this PV system. We obtain the hourly DA market price data for the Dutch market from the European Network of Transmission System Operators [2], and the IM market price data from the regional Transmission System Operator, called TENNET [5]. We use the dataset released by ElaadNL, a large charging infrastructure in the Netherlands [1], as our EV dataset. This dataset contains 10,000 charging sessions that occurred in several public EV chargers operated by EVnetNL.[6] Note that since the initial and target SoC levels are not reported in this dataset, we assume the target SoC of every EV is 1 and use the total energy charged into the battery in the respective charging session to calculate its initial SoC. We evaluate the proposed operating strategies (LLA and LA-SAC) and baselines on this test dataset.



(a) (b) (c)

Figure 3: Empirical distribution of solar generation and EV mobility data: (a) daily solar generation where each gray curve represents the PV system output on a specific day and the blue curve represents the hourly mean solar generation over one year; (b) the probability mass function (pmf) of the number of EV arrivals in a specific hour of the day; (c) the conditional pmf of stay times for EVs arrived in that hour.

We create a new dataset, separate from the test dataset described above, and use it only to train the LA-SAC agent in the second operating strategy. This is necessary because the amount of historical data (charging sessions) in the ElaadNL dataset is not enough to learn a near-optimal policy via reinforcement learning. To obtain a sufficiently large number of episodes, we synthesize realistic charging sessions. Specifically, we fit distributions to EV arrival times (depicted in Figure 3b), EV stay times (depicted in Figure 3c), and EV charging demands in the ElaadNL dataset. We assume that the number of arrivals in each hour of the day follows Poisson distribution with a parameter that depends on that particular hour. Since the stay time correlates with the arrival time, we fit a Gaussian mixture model, using Kernel Density Estimation (KDE), to the empirical distribution of stay times for EVs that arrived in a certain hour of the day. For the initial SoC levels, we use a truncated Gaussian distribution with mean 0.49 and standard deviation of 0.25, with the minimum and maximum SoC being 0.03 and 0.97 respectively; the two moments of the Gaussian distribution are defined according to the distribution of initial SoC in the test dataset. Furthermore, we assume that all EVs must be fully charged before they leave the charging station and set $\overline{SoC}^n$ accordingly. We set the size of all EV batteries to 80kWh (similar to Tesla Model 3), the

[5] https://solcast.com/

[6] https://evnet.nl/

maximum charge and discharge rates to 11kW, and the charge and discharge efficiencies to 0.98. We generate almost 3 years worth of EV charging sessions by sampling from these distributions. To create the training dataset, this data is combined with real market prices and solar traces that are collected from the same sources we used to create the test dataset, this time for a period that ends on December 31, 2019. This allows us to train the LA-SAC agent for up to 1,000 episodes, where each episode consists of 24 one-hour time steps, representing 1 day.

6.2 Baselines

6.2.1 Offline Deterministic Baseline The first baseline solves Problem (3) using the actual values of the hourly market prices ($\mathbf{P}^{DA}$, $\mathbf{P}^{IM}$), solar generation ($\mathbf{E}^{solar}$), and EV mobility and energy demand ($t_s^n, \tau^n, \underline{SoC}^n, \overline{SoC}^n$). The solution would give the maximum profit the VPP can make by operating in the two-stage electricity market, *i.e.*, $\mathbf{X}^*$, $\mathbf{Y}^{*n}$, and $\mathbf{Z}^*$. Note that we do not need to solve the stage 2 problem separately, because we obtain the optimal EV charging schedule as the LP is solved with perfect information. This baseline, which we call **ORACLE**, gives an upper bound on the VPP's profit. In practice, the VPP operates under significant uncertainty, hence it is impossible to generate this much profit.

6.2.2 Current Practice in EV Charging The current practice in EV charging, referred to as **CHRG_ASAP**, is charging an EV at the maximum power as soon as it gets connected to a charger. This strategy minimizes the length of the charging session. It combines the solution of stage 1 and stage 2 problems, when they are solved without taking advantage of V2G. More specifically, for solving stage 1 under this baseline, Problem (2) is modified by dropping constraint (2g) and writing constraint (2f) for all $n \in \mathcal{N}_t$, and then Problem (2) is solved using the approach described in Section 5.1.

In stage 2, EVs are charged at the maximum power when they arrive at a charging station. The main drawback of this strategy is the reduced flexibility, limiting the VPP options when it comes to addressing potential deviations from day-ahead commitments.

7 Results

We now evaluate the performance of the proposed VPP operating strategies on the dataset described in Section 6. We investigate the profitability of the VPP in different scenarios considering five different V2G participation rates (0%, 25%, 50%, 75%, 100%), where the participation rate is defined as the ratio of EVs whose battery can be discharged to the total number of EVs that visit the charging stations in one day. Hence, 0% participation implies that *none* of the EVs that arrive at the charging stations can be discharged, and 100% participation implies that *all* EVs might be discharged as long as their charging demand can be met by their deadline. For a fixed participation rate, we randomly sample the required number of EVs to participate in V2G from the set of EVs that will visit the charging stations during the day.

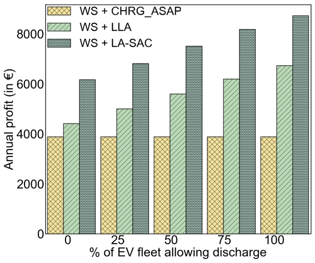


Figure 4: Annual profit earned by the VPP for various V2G participation rates.

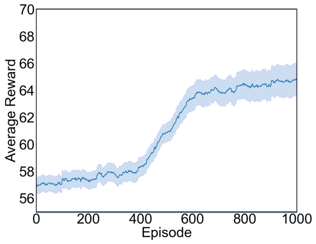


Figure 5: The reward obtained per episode by LA-SAC for 100% V2G participation.

Figure 4 compares the annual profit earned by the VPP when it adopts CHRG_ASAP, LLA, and LA-SAC. [7] We see that both LLA and LA-SAC greatly increase the VPP's profit for all V2G participation rates, while the profit earned under CHRG_ASAP is much lower and does not vary with the V2G participation rate because it does not take advantage of bidirectional charging. The increased profitability of the VPP when all EVs participate in V2G compared to when V2G is not supported (52% and 42% increase under LLA and LA-SAC, respectively) highlights the importance of bidirectional charging in making this kind of VPP viable. Table 1 compares the annual profit earned by the VPP using each operating strategy as a percentage of the annual profit generated by ORACLE for the same V2G participation rate. The result indicates that for 100% V2G participation case, LA-SAC is the best performing algorithm, achieving 51.4% of the profit that could be possibly earned if there was no uncertainty, followed by LLA which achieves 39.6% of the profit earned by ORACLE. The gap between the performance of the proposed operating strategies and ORACLE widens slightly as the V2G participation rate increases. We attribute this to the increased complexity of the problem when more storage capacity becomes available. It is important to note that LLA achieves around 40% of the profit that could be possibly earned if there was no uncertainty using a simple heuristic. However, should the VPP be able to afford the training cost of LA-SAC, its annual profit can be increased by up to €2,002 compared to when it adopts LLA. Recall that LLA and LA-SAC do not violate the charging deadlines because they guarantee the non-negativity of laxity at all times.

Lastly, we briefly discuss the number of episodes required to train a policy using the LA-SAC algorithm. We use 3 random seeds (independent trials) to train a policy using this algorithm. In each trial, the policy is trained for 1,000 episodes. Figure 5 depicts the learning curve of the RL agent assuming 100% participation in V2G. The solid curve corresponds to the mean over 3 trials and the shaded region shows one standard error from the mean. It can be seen that the RL agent demonstrates improved performance after around 600 episodes (days). We witnessed stable performance when the learned policy (after 1,000 episodes) was evaluated on the test dataset.

8 Conclusion and Future Directions

Operating the DERs in a VPP is a challenging task owing to the large number of stochastic processes that govern demand, supply, and market prices. In this paper, we considered an emerging type of VPP that integrates a fleet of EVs with bidirectional chargers and solar systems. We proposed efficient and practical operating strategies for this VPP when participating in a two-stage electricity market. The VPP places energy bids in the DA market according to the average solution of a sequence of deterministic linear programs solved for

[7] Recall that both strategies solve the stage 1 problem using a WS approach. But, for brevity, we just use the name of the algorithm used in stage 2 to refer to each strategy.

	V2G participation (%)				
Algorithm	0	25	50	75	100
CHRG_ASAP	37.5%	32.4%	28.3%	25.2%	22.8%
LLA	42.7%	41.7%	40.8%	40.3%	39.6%
LA-SAC	59.6%	56.8%	54.8%	53.2%	51.4%

Table 1: The profit earned under different strategies as a percentage of the profit earned by ORACLE (assuming perfect information) for the same V2G participation rate.

different realizations of random variables (forecast scenarios), and trades in the IM market to honor its day-ahead commitments and satisfy the EV charging demands. We proposed one heuristic and one RL-based algorithm with a differentiable projection layer to maximize the profit of this VPP on the operation day, given the day-ahead commitments. We evaluated profitability of this VPP under these operating strategies using real data pertaining to a specific region in the Netherlands, and compared it with the offline optimal and current EV charging baselines. Our result shows that the proposed operating strategies exhibit strong performance and outperform the prevalent practice in EV charging, and that enabling V2G can substantially increase the profit of this kind of VPP.

In future work, we aim to explore a related but structurally different case, in which the EV owners and the owner of the charging stations (*i.e.*, the aggregator) are different agents. In that case, the aggregator needs to design a billing mechanism to charge the EV owners for the service and another mechanism to incentivize them to participate in V2G (given the battery degradation cost), while making sure that the VPP remains profitable and deadlines are met.

Acknowledgments

This research was supported by funding from the Canada First Research Excellence Fund as part of the University of Alberta's Future Energy Systems research initiative.

References

[1] [n. d.]. ElaadNL Open Data. https://platform.elaad.io/download-data/

[2] [n. d.]. ENTSO-E Transparency Platform. https://transparency.entsoe.eu

[3] [n. d.]. South Australia's Virtual Power Plant. https://www.energymining.sa.gov.au/growth_and_low_carbon/virtual_power_plant

[4] [n. d.]. Swell Energy's Virtual Power Plants. https://www.swellenergy.com/

[5] [n. d.]. Tennet Export data. https://www.tennet.org/english/operational_management/export_data.aspx

[6] [n. d.]. Tesla Virtual Power Plant. https://www.tesla.com/support/energy/tesla-virtual-power-plant-pge-2022

[7] [n. d.]. United States distributed energy resources outlook. https://www.woodmac.com/news/editorial/der-growth-united-states/

[8] [n. d.]. Utrecht wants to be the first city to use its electric car fleet as a giant battery. https://www.fastcompany.com/90705832/utrecht-wants-to-be-the-first-city-to-use-its-electric-car-fleet-as-a-giant-battery

[9] Kankam Adu-Kankam et al. 2018. Towards collaborative Virtual Power Plants: Trends and convergence. *Sustainable Energy, Grids and Networks* 16 (2018), 217–230.

[10] Akshay Agrawal et al. 2019. Differentiable Convex Optimization Layers. *Advances in neural information processing systems* 32 (2019).

[11] Abdullah Al Zishan et al. 2020. Adaptive control of plug-in electric vehicle charging with reinforcement learning. In *Proceedings of the Eleventh ACM International Conference on Future Energy Systems.* ACM, 116–120.

[12] Arijit Bagchi et al. 2019. Adequacy Assessment of Generating Systems Incorporating Storage Integrated Virtual Power Plants. *IEEE Transactions on Smart Grid* 10, 3 (2019), 3440–3451.

[13] Kyri Baker et al. 2016. Distributed MPC for Efficient Coordination of Storage and Renewable Energy Sources Across Control Areas. *IEEE Transactions on Smart Grid* 7, 2 (2016), 992–1001.

[14] Kyle Bradbury et al. 2014. Economic viability of energy storage systems based on price arbitrage potential in real-time US electricity markets. *Applied Energy* 114 (2014), 512–519.

[15] Bingqing Chen et al. 2021. Enforcing Policy Feasibility Constraints through Differentiable Projection for Energy Optimization. In *Proceedings of the Twelfth ACM International Conference on Future Energy Systems.* ACM, 199–210.

[16] Sayed Yaser Derakhshandeh et al. 2013. Coordination of Generation Scheduling with PEVs Charging in Industrial Microgrids. *IEEE Transactions on Power Systems* 28, 3 (2013), 3451–3461.

[17] Steven Diamond et al. 2016. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research* 17, 83 (2016), 1–5.

[18] A. P. Dobos. 2014. PVWatts Manual. (2014). https://www.osti.gov/biblio/1158421

[19] Energy Networks Australia. 2017. *Electricty Network Transformation Roadmap: Final Report.* https://www.energynetworks.com.au/resources/reports/electricity-network-transformation-roadmap-final-report/

[20] EU. 2020. *Decisions of the Agency for the Cooperation of Energy Cooperation of Energy Regulators No 18-2020.* https://nordicbalancingmodel.net/roadmap-and-projects/single-price-model/

[21] Javier García et al. 2015. A Comprehensive Survey on Safe Reinforcement Learning. *Journal of Machine Learning Research* 16, 42 (2015), 1437–1480.

[22] Marco Giuntoli and Davide Poli. 2013. Optimized thermal and electrical scheduling of a large scale virtual power plant in the presence of energy storages. *IEEE Transactions on Smart Grid* 4, 2 (2013), 942–955.

[23] Tuomas Haarnoja et al. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning.* PMLR, 1861–1870.

[24] Nathan Hunt et al. 2021. Verifiably safe exploration for end-to-end reinforcement learning. In *Proceedings of the 24th International Conference on Hybrid Systems: Computation and Control.* 1–11.

[25] IRENA. 2019. *Innovation landscape brief: Market integration of distributed energy resources.* https://www.irena.org/-/media/Files/IRENA/Agency/Publication/2019/Feb/IRENA_Market_integration_distributed_system_2019.pdf

[26] José Iria et al. 2019. Optimal bidding strategy for an aggregator of prosumers in energy and secondary reserve markets. *Applied Energy* 238 (2019), 1361–1372.

[27] Chenrui Jin et al. 2013. Optimizing electric vehicle charging with energy storage in the electricity market. *IEEE Transactions on Smart Grid* 4, 1 (2013), 311–320.

[28] Mostafa Kazemi et al. 2017. Operation scheduling of battery storage systems in joint energy and ancillary services markets. *IEEE Transactions on Sustainable Energy* 8, 4 (2017), 1726–1735.

[29] Fiodar Kazhamiaka et al. 2019. Adaptive Battery Control with Neural Networks. In *Proceedings of the Tenth ACM International Conference on Future Energy Systems.* ACM, 536–543.

[30] Sujin Kim et al. 2015. A Guide to Sample Average Approximation. In *Handbook of Simulation Optimization.* Springer, Chapter 8, 207–243.

[31] Steven George Krantz and Harold R Parks. 2002. *The implicit function theorem: history, theory, and applications.* Springer Science & Business Media.

[32] Zheming Liang et al. 2019. Risk-Constrained Optimal Energy Management for Virtual Power Plants Considering Correlated Demand Response. *IEEE Transactions on Smart Grid* 10, 2 (2019), 1577–1587.

[33] Albert Madansky. 1960. Inequalities for stochastic linear programming problems. *Management Science* 6, 2 (1960), 197–204.

[34] Francis Mwasilu et al. 2014. Electric vehicles and smart grid interaction: A review on vehicle to grid and renewable energy sources integration. *Renewable and Sustainable Energy Reviews* 34 (2014), 501–516.

[35] Pacific Northwest National Library. 2012. *Utilizing Electric Vehicles to Assist Integration of Large Penetrations of Distributed Photovoltaic Generation.*

[36] Pandžić et al. 2013. Offering model for a virtual power plant based on stochastic programming. *Applied Energy* 105 (2013), 282–292.

[37] Bala Suraj Pedasingu et al. 2020. Bidding Strategy for Two-Sided Electricity Markets: A Reinforcement Learning Based Framework. In *Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation.* ACM, 110–119.

[38] Baihong Qi et al. 2019. Energyboost: Learning-based control of home batteries. In *Proceedings of the Tenth ACM International Conference on Future Energy Systems.* ACM, 239–250.

[39] Zhichao Shi et al. 2018. Distributionally robust chance-constrained energy management for islanded microgrids. *IEEE Transactions on Smart Grid* 10, 2 (2018), 2234–2244.

[40] Thiago D. Simão et al. 2021. AlwaysSafe: Reinforcement Learning without Safety Constraint Violations during Training. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems.* IFAAMAS, 1226–1235.

[41] Matteo Vasirani et al. 2013. An agent-based approach to virtual power plants of wind power generators and electric vehicles. *IEEE Transactions on Smart Grid* 4, 3 (2013), 1314–1322.

[42] Yao Wang et al. 2016. Interactive Dispatch Modes and Bidding Strategy of Multiple Virtual Power Plants Based on Demand Response and Game Theory. *IEEE Transactions on Smart Grid* 7, 1 (2016), 510–519.

[43] Yiling Zhang et al. 2019. Distributionally Robust Building Load Control to Compensate Fluctuations in Solar Power Generation. In *2019 American Control Conference (ACC).* 5857–5863.