Dynamic Spatio-Temporal Planning Strategy of EV Charging Stations and DGs Using GCNN-Based Predicted Power Demand

Shahriar Rahman Fahim, *Member, IEEE*, Rachad Atat, *Senior Member, IEEE*, Cihat Kececi, Abdulrahman Takiddin, *Member, IEEE*, Muhammad Ismail, *Senior Member, IEEE*, Katherine R. Davis, *Senior Member, IEEE*, and Erchin Serpedin, *Fellow, IEEE*

***Abstract*—As a sustainable participant in the modernization of transportation systems, electric vehicles (EVs) call for a well-planned charging infrastructure. To meet the ever-increasing charging demands of EVs, an efficient dynamic spatio-temporal allocation strategy of charging stations (CSs) is necessary. With newly allocated CSs, additional distributed generators (DGs) are required to compensate for the load increase. Given a budget to be allocated over a certain time horizon, we formulate the joint spatio-temporal CSs and DGs planning problem as a multi-objective optimization problem. During each planning period, the allocation strategy aims at minimizing the total power generation costs and CSs/DGs installation costs while satisfying budgetary and power constraints and ensuring a minimum level for the charging requests satisfaction rate. In this regard, we first predict the future power demand of EVs using a graph convolutional neural network (GCNN). Then, using the power demand forecast, we obtain the optimal number and locations of CSs and DGs at each time stage using reinforcement learning. A case study of the proposed allocation strategy over 6 time stages for the 2000-bus power grid of Texas coupled with 720 initially existing CSs is presented to illustrate the performance of the planning strategy.**

***Index Terms*—Charging stations allocation, dynamic program, electric vehicles, prediction, and reinforcement learning.**

NOMENCLATURE

Acronyms

CS	Charging station
DD	Demand difference
DG	Distributed generator
EV	Electric vehicle
GCNN	Graph convolutional neural network
LSTM	Long short-term memory
MDP	Markov decision process
RE	Reinforcement learning

Parameters

$\mathcal{S}$	Total planning period.
μ_{tr}, μ_{gr}	Total cost for transportation system and power grid, resp.
$\Omega_{b,s}^{\text{CS}}$	Total number of CSs at bus b at time stage s.
$\Omega_{b,s}^{\text{DG}}$	Total number of DGs at bus b at time stage s.
ζ_b	Unit traffic flow charging demand rate at CS b.
B_{tot}	Total budget.
c_{cs}	Installation cost of a CS.
c_{DG}	Installation cost of a DG.
c_{pen}	Penalty cost for unsatisfied charging requests.
c_{pg}	Power generation costs.
C_{tr}, C_{gr}	Weight for transportation and power systems, resp.
$N_{\text{CS},b,s}$	Number of already available CSs at bus b at stage s.
$N'_{\text{CS},b,s}$	Number of CSs to be installed at bus b at stage s.
$N^{\text{th}}_{\text{CS},b,s}$	Maximum number of allowable CSs at bus b at time stage s.
$N'_{\text{DG},b,s}$	Number of DGs to be installed at bus b at stage s.
$N_{\text{EV},b,s}$	Total number of EVs at CSs connected to bus b at time stage s.
$P_{b,\tau}^{max}, P_{b,\tau}^{min}$	Maximum and minimum active power at bus b at time τ, resp.
$Q_{b,\tau}^{max}, Q_{b,\tau}^{min}$	Maximum and minimum reactive power at bus b at time τ, resp.
$V^{\text{max}}, V^{\text{min}}$	Maximum and minimum voltage limits, resp.
z_i^d	Decision variable for choosing CS d by i^{th} EV.
B	Number of all buses in the network.
b	Bus index.

Variables

$\delta_{b,\tau}$	Angle magnitude.
$\mathcal{Q}_b$	Set of all paths connecting bus b.
ϕ_s	EVs satisfaction rate at time stage s.
$\psi_{d,\tau}\left(z^d\right)$	Total expected demand for EVs at CS d at time τ.
E_i	Energy demand of i^{th} EV.
$F_{q,s}$	Volume of traffic flow on path q at time stage s.
$I_{q,s}$	Fraction of traffic flow on path q at time stage s whose charging demands are not met.
$P_{b,\tau}, Q_{b,\tau}$	Active and reactive power at bus b at time τ, resp.
$P_{b,\tau}^{\text{D}}, Q_{b,\tau}^{\text{D}}$	Active and reactive power demand at bus b at time τ, resp.
$P_{b,\tau}^{\text{G}}, Q_{b,\tau}^{\text{G}}$	Active and reactive power generated at bus b at time τ, resp.
$P_{b,\tau}^{\text{R}}$	Regular load at bus b at time τ.
$P_{h,\tau}^{\text{DG}}, Q_{h,\tau}^{\text{DG}}$	Active and reactive power of DG unit h at bus b at time τ, resp.
$P_{i,b}^{\text{CS}}$	Power demand of CS i connected to bus b.
$V_{b,\tau}$	Voltage magnitude at time τ.
$\text{R}_{b,b'}$	Resistance between bus b and b'.
$\text{X}_{b,b'}$	Reactance between bus b and b'.

S. R. Fahim, C. Kececi, K. R. Davis, and E. Serpedin are with the Electrical & Computer Engineering Department, Texas A&M University, College Station, TX 77843, USA (e-mail: {sr-fahim, kececi, katedavis, eserpedin}@tamu.edu).

A. Takiddin is with the Department of Electrical and Computer Engineering, FAMU-FSU College of Engineering, Florida State University, Tallahassee, FL 32310 USA (e-mail: a.takiddin@fsu.edu).

R. Atat is with the Department of Computer Science and Mathematics, Lebanese American University, Beirut, Lebanon (email: rachad.atat@lau.edu.lb).

M. Ismail is with the Department of Computer Science, Tennessee Tech University, Cookeville, TN 38505 USA (e-mail: mismail@tntech.edu).

This work is supported by NSF EPCN Awards 2220346 and 2220347.

I. INTRODUCTION AND MOTIVATION

AT present, petroleum-based vehicles contribute to 25% of global energy consumption with carbon dioxide emission of 26% [1]. China's National Petroleum Corporation stated in

the "Energy Outlook 2050" report that transportation energy consumption would rise by 23% in 2050 compared to 2015 [2]. These numbers explain the growth of electric vehicles (EVs) for a greener environment. The electrification and integration of the transportation system into the power grid result in high energy efficiency and decreased emissions of greenhouse gases. Additionally, it enables the adoption of intelligent charging techniques and the optimization of energy use, creating a more robust and sustainable transportation system.

The worldwide increase in EVs traffic translates into a significant increase in the number of charging requests. This necessitates that additional charging stations (CSs) be deployed over multiple time phases to satisfy the charging demands. If too many CSs are installed, many might end up underutilized leading to wastage of resources, increase in costs, and loss in profits. Conversely, if too few CSs are installed, many charging requests will be left unaddressed leading to a loss in quality-of-service. Moreover, the installation of new CSs is accompanied by an increase in the load demands, which must be balanced by increasing the power generation. This can be achieved by installing distributed generator (DG) units. In this paper, we consider dispatchable DG units for simplicity. Therefore, given a budget limit at each time stage, a dynamic spatio-temporal planning strategy for EV CSs and DGs is required. At each time stage, the allocation strategy should minimize the power generation and CSs installation costs, while satisfying the power balance equations, the budgetary constraint, as well as the charging requests satisfaction rate. Thus, the optimal joint allocation planning problem of EV CSs and DGs involves two main steps: i) forecasting the required future charging demand and ii) solving the multi-objective optimization problem in order to obtain the number and locations of CSs and DGs to install at each time stage [3].

A. Related Work

1) Predicting EVs Charging Demand: So far, a few approaches have been reported to predict the EVs charging demand growth. Some approaches considered EVs battery usage and estimated the demand under varying traffic scenarios. For instance, the aggregated behavior of EV drivers was used in [4] to predict the EVs charging demand. In [5], the authors used stochastic modeling and analysis based on EV charging measurements and load data to predict future charging demand. However, these approaches are deemed computationally expensive and might be challenging for large-scale power and transportation systems applications.

Other approaches estimated the EVs demand growth using travel chains and Markov decision process (MDP) [6], mixed-integer programming models [7], and the geographical features of EVs points of interests [8]. All these models failed to capture the higher-order patterns of the dependencies between power and transportation systems. Thus, these models create biases and errors in prediction, which can lead to a non-accurate allocation plan.

Recently, computational intelligence-based approaches have gained popularity in demand prediction due to their strong generalization abilities [9]. To predict future EV charging loads the authors of [10] investigated the performance of artificial neural networks (ANN), rough ANN (RANN), and recurrent RANN (RRANN). Results pointed out that the RRANN model outperformed the other models in terms of accuracy. As conventional RANNs may suffer due to the vanishing and exploding gradients condition, a separate study addressed this concern by adopting a long short-term memory (LSTM)-based approach [11]. Different approaches in this direction have ventured into various modifications of basic LSTM, encompassing variations such as LSTM equipped with an attention mechanism [12], Gaussian process regression [13], or integration with convolutional neural networks (CNNs) [14]. Each of these modifications enhanced and fine-tuned the effectiveness of the LSTM model. Unfortunately, these works still fall short in effectively capturing the extended temporal dependencies present within the dataset.

The effectiveness of a charging load prediction strategy relies on the abilities of deep learning models to manage nonlinearities (e.g., capturing nonlinear dependencies) and generalize effectively. In this regard, reference [15] considered a CNN equipped with attention mechanism. This CNN-based strategy extracted the features of complex coupling relationships while simultaneously optimized the computation time. An autoencoder-based model was presented in [16] with the aim of generating EV load profiles. In the pursuit of developing deep learning-based load prediction models, researchers have proposed hybrid methodologies that combine two or more models to forecast future charging demands. For instance, the study [17] combined a stacked autoencoder with an LSTM-based model. Autoencoders were also combined with Restricted Boltzmann Machines to enhance feature extraction, as illustrated in [18]. Nevertheless, these works failed to capture the inherent temporal dependencies and association relationships present in the coupled datasets. Other instances of hybrid strategies include CNN with LSTM [19], RE with LSTM [20], and multi-channel CNN with temporal CNN to capture both spatial and temporal features [21]. The major limitation of the aforementioned works was that the training model was based on one-dimensional sequential data in the Euclidean space, and therefore, they fail to extract the localized spatial features. Therefore, these models are not suitable for dynamic spatio-temporal data structures.

The subsequent advancements have embraced a range of proposals. For instance, Bayesian deep learning techniques were adopted to account for uncertainties in forecasting [22], while probabilistic queuing models combined with CNN were employed to capture driver behaviors and the limitations of charging services [23]. The remarkable increase in computational capacity has enabled the utilization of more sophisticated deep learning algorithms, including gated recurrent units [24] and recurrent neural networks [25], resulting in further improved accuracy in load forecasting. Nonetheless, the aforementioned algorithms often encounter challenges in effectively capturing the complex coupling relationships as well as dependencies present in the combined dataset.

2) CSs Allocation: In the context of the EV CS allocation problem, different optimization approaches have been reported. A multi-objective uncertainty-aware allocation strategy

was proposed in [26], where the uncertainties of traffic flows and EVs state of charge during the day were considered. Particle swarm optimization (PSO) technique was adopted in [27] to optimally allocate CSs. The proposed allocation strategy was shown to reduce the cost of EVs charging infrastructure by 75%. Other works optimized the CSs allocation by considering the power loss costs and CSs installation costs [26], [27], voltage deviation costs [28], penalty costs [29], waiting time costs [30], EVs population and land costs [31], and charging willingness of EV drivers [32]. Integrating all these costs into an optimization framework requires careful balancing to achieve the most efficient and cost-effective solution. If the technical and user-oriented criteria are not defined carefully, these approaches may produce erroneous planning results and may fail to account for the full scope of uncertainties.

Additional approaches were proposed. Mixed integer linear programming (MILP)-based optimization tools were proposed to address allocation problem, see e.g., [33]–[35]. However, the problem formulations in the aforementioned works rely on linear relationships between variables, which may lead to infeasible solutions. Other notable works employed heuristic or meta-heuristic algorithms such as Genetic Algorithm (GA) [36], PSO [37], multi-population GA [38], and multi-objective PSO [39]. An approximate dynamic programming optimization method is proposed in [40] that uses continuous state and action spaces that ensures consistent feedback actions regardless of initial operating conditions. This method automatically adjusts for variations in electricity prices and available solar energy during optimization. However, its major limitation lies in its focus on immediate optimization rather than long-term strategic planning. This method does not incorporate predicted demand, which is crucial for anticipating future requirements and ensuring the sustainable growth of the EV charging infrastructure. A data-driven method that employs a finite-discrete MDP in conjunction with the RL framework to address the challenge of high dimensional features space was reported in [41]. The RL-based allocation strategy aims to minimize the total investment budget while satisfying the coupling network constraints [41]. Another study modeled the historical EVs charging demands using a Markovian process and then formulated a scenario-based stochastic expansion planning from the observations [35]. From a different perspective, reference [42] adopted a deterministic policy gradient-based RL approach to regulate the energy storage system in an EV CS network to achieve maximum operating profits. It is worth mentioning that the aforementioned approaches considered the interaction of EVs with policy makers, which is useful for operational rather than allocation purposes.

The planning and installation of CSs lead to an increase in power demand, which is usually balanced with installing DGs. In [43], the authors considered a coupled transportation-smart grid system and performed load balancing through EV charging networks. However, most of the previous works solved the joint allocation strategy for CSs and DGs without considering the predicted EVs charging demand, see e.g., [44]–[46]. In [44], we performed the joint allocation of CSs and DGs via a constrained MDP model. Reference [45] proposed a GA-based approach to implement the joint allocation strategy of CSs and DGs. In [46], a hybrid scheme that adopts Grey Wolf Optimizer (GWO) and PSO approach for the joint allocation of CSs and DGs is proposed. It is important to mention at this stage that the aforementioned works did not present a framework that reflects the transportation system dependencies on the power grid. The power and transportation networks are tightly coupled and the interactions between these systems must be considered in the problem formulation. This is important because the objective in one system may affect the objective in the other system.

To address the limitations of traditional approaches, herein we propose to solve the multi-objective CSs-DGs allocation strategy by employing a combination of a Q-learning method with a graph convolutional neural network (GCNN). Our primary objective is to develop a strategic framework for optimizing the placement and number of CSs and DGs over an extended planning period. This long-term perspective is crucial for ensuring that the power system can sustainably meet the growing demand for EV charging.Thus, we considered non-dispatchable DGs (solar) which have low carbon emissions.

B. Contributions and Organization

The contributions of this paper are summarized as follows:

- First, we predict the EVs demand growth using a fusion model that combines GCNN and LSTM, where GCNN allows to capture the graph-structured features and LSTM helps to catch the temporal correlations in data.
- Second, given the predicted EVs demand growth and a budget limit that can be spent over a fixed time horizon, we formulate the spatio-temporal allocation strategy for EV CSs and DGs that minimizes the power and CSs/DGs installation costs, while satisfying the budget, power balance, and charging satisfaction constraints.
- Third, we solve the problem using a model-free Q-learning algorithm that outputs the minimum number and locations of CSs/DGs to allocate at each time stage.

The remainder of this paper is organized as follows. Initially, Section II presents the power system model with integrated CSs. Section III predicts the EVs charging demand growth using deep learning. Section IV formulates the multi-objective CSs and DGs allocation problem. Section V presents the solution approach. Section VI presents the simulation results. Finally, conclusions are drawn in Section VIII.

II. System modeling

In this section, we describe the modeling of considered power-dependent transportation systems.

A. Power system modeling

The coupled power-transportation system is a graph $G = (V_{\mathrm{P}}, E_{\mathrm{P}})$, where $V_{\mathrm{P}} = \{1, 2, .., B\}$ stands for the nodes set, B denotes the total number of buses and E_{P} identifies the edge set. Let $\mathbf{W} \in \{0, 1\}^{B \times B}$ be the binary adjacency matrix of graph G whose element w_{bv} denotes its $(b, v)^{\text{th}}$ entry. The buses represent load or generator substations, characterized by unique active/reactive power values and voltage magnitudes

and angles. The power lines are subject to active/reactive power flows dependent on the respective line impedance.

The operational constraints of the considered power system is presented in Eq.1a. The active and reactive power balance equations ensure that the power injected at each bus (b) and at each time period (τ) is correctly balanced. The active power balance equation ($P_{b,\tau}$) considers the power flows between buses, incorporating the effects of voltage differences ($V_{b,\tau} - V_{b',\tau}$) and phase angle differences ($\delta_{b,\tau} - \delta_{b',\tau}$), modulated by the line admittance ($\alpha_{b,b'}$). Similarly, the reactive power balance equation ($Q_{b,\tau}$) ensures that the reactive power is balanced by accounting for the same voltage differences. The system also includes the power generated ($P^G_{b,\tau}$) and consumed ($P^D_{b,\tau}$) at each bus, where the demand includes both regular loads and contributions from electric vehicle charging stations ($P_{CS_{i,b}}$). Additionally, the system accounts for distributed generators ($P_{DG_{h,\tau}}$ and $Q_{DG_{h,\tau}}$), which contribute to the overall power balance. Operational constraints are also considered to ensure system stability and efficiency. These constraints include limits on power generation ($P^{min}_{b,\tau}$ and $P^{max}_{b,\tau}$) and voltage levels ($V^{min}_{b,\tau}$ and $V^{max}_{b,\tau}$), ensuring that the power system operates within safe and reliable bounds. The inclusion of distributed generation and charging stations reflects the modern grid's complexity, where multiple energy sources and consumption points must be managed dynamically. The equations also integrate the constraints related to the capacities and operational limits of distributed generators and charging stations, ensuring that their contributions to the power system are feasible and within specified limits.

Power system operational constraints:

$$\begin{cases} P_{b,\tau} = \sum_{b' \in B, b' \neq b} \alpha_{b,b'} \left(V_{b,\tau} - V_{b',\tau}\right) \\ +\alpha'_{b,b'} \left(\delta_{b,\tau} - \delta_{b',\tau}\right), \forall b \in B, \forall \tau \in T \\ Q_{b,\tau} = \sum_{b' \in B, b' \neq b} \alpha_{b,b'} \left(V_{b,\tau} - V_{b',\tau}\right) \\ -\alpha'_{b,b'} \left(\delta_{b,\tau} - \delta_{b',\tau}\right), \forall b \in B, \forall \tau \in T \\ P^{\mathrm{D}}_{b,\tau} = P^{\mathrm{R}}_{b,\tau} + \sum_{i \in \{\Omega^{\mathrm{CS}}_{b,s}\}} P^{\mathrm{CS}}_{i,b}, \forall b \in B, \forall \tau \in T \\ \sum_b \left(P_{bb',\tau} - P_{b'b,\tau}\right) = \sum_{h \in \{\Omega^{\mathrm{DG}}_{b,s}\}} P^{\mathrm{DG}}_{h,\tau} - P^{\mathrm{D}}_{b,\tau} \\ - \sum_{d \in \{\Omega^{\mathrm{CS}}_{b,s}\}} \psi_{d,\tau}\left(z^d\right) \\ \sum_b \left(Q_{bb',\tau} - Q_{b'b,\tau}\right) = \sum_{h \in \{\Omega^{\mathrm{DG}}_{b,s}\}} Q^{\mathrm{DG}}_{h,\tau} - Q^{\mathrm{D}}_{b,\tau}, \forall b, \forall \tau \\ P^{\min}_{b,\tau} \leq P^{\mathrm{D}}_{b,\tau} \leq P^{\max}_{b,\tau}, \quad \forall b \in B, \forall \tau \in T \\ Q^{\min}_{b,\tau} \leq Q^{\mathrm{D}}_{b,\tau} \leq Q^{\max}_{b,\tau}, \quad \forall b \in B, \forall \tau \in T \\ V^{\min}_{\tau} \leq V_{b,\tau} \leq V^{\max}_{\tau} \\ \sum_{b=1}^{N_{\mathrm{d}}} P^{\mathrm{D}}_{b,\tau} < \sum_{b=1}^{N_{\mathrm{g}}} P^{\mathrm{G}}_{b,\tau} + \sum_{h \in \{\Omega^{\mathrm{DG}}_{b,s}\}} P^{\mathrm{DG}}_{h,\tau}, \forall b, \forall \tau \\ \sum_{b=1}^{N_{\mathrm{d}}} Q^{\mathrm{D}}_{b,\tau} < \sum_{b=1}^{N_{\mathrm{g}}} Q^{\mathrm{G}}_{b,\tau} + \sum_{h \in \{\Omega^{\mathrm{DG}}_{b,s}\}} Q^{\mathrm{DG}}_{h,\tau}, \forall b, \forall \tau \end{cases} \tag{1a}$$

Since EV CSs impose significant additional loads on the power grid, efficient integration of these loads requires careful coordination to prevent overloading within the power system. Moreover, the spatial distribution of CSs must align with the existing power grid to optimize its use. Finally, understanding traffic flow patterns is essential to place CSs in locations that maximize the use of CSs while balancing the load on the power grid. To initially integrate existing CSs within the power system, we used the geographical coordinates of power substations and CSs to establish a connection within the networks. For this purpose, we consider a circular boundary of a specific radius for all load substations, and we assign all the CSs that fall within the said boundary (determined by Haversine distance) of the load substations. If a single CS falls within the boundary of multiple load substations, then we calculate the Haversine distances between power substations and CSs, and we assign CSs to load substations based on the shortest Haversine distance. A power-dependent transportation system is presented in Fig.1

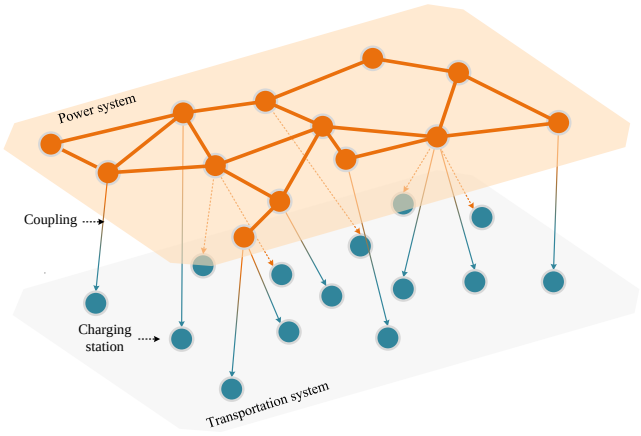

Fig. 1: An illustration of power-dependent transportation network.

B. Power-Transportation system modeling

The coupling relationship between the power and transportation system is illustrated in Eq. 2a as:

Power-transportation system Constraints:

$$\psi_{d,\tau,s}\left(z^d\right) = \sum_{i \in N_{\mathrm{EV},b,s}} E_i z^d_i, \forall d \in \{\Omega^{\mathrm{CS}}_{b,s}\}, \forall b \in B, \forall \tau \in T \tag{2a}$$

Considering the impacts of traffic conditions on transportation networks [47], we specifically considered $F_{q,s}$: the volume of traffic flow on path q at time stage s. The expected demand $\psi_{d,\tau,s}(z^d)$ must be balanced with the available power supply at each bus, ensuring that the energy requirements for EV charging are met without compromising the stability and efficiency of the power grid. This balancing act involves several key components: the power generated by distributed generators (DGs), the power supplied by the central grid, and the power flow between buses. Distributed generators at each bus produce active power ($P_{DG_{h,\tau}}$) and reactive power ($Q_{DG_{h,\tau}}$), which directly supply the local demand, including the CSs. This localized generation reduces the burden on the central power grid and minimizes transmission losses. However, the central grid also plays a crucial role in meeting the excess demand that local DGs cannot satisfy. The power supplied by the grid must be managed to maintain voltage levels and system frequency within acceptable limits, ensuring overall grid stability.

Additionally, the interconnected nature of the power system allows power to flow between buses, enabling buses with surplus power to support those with deficits. Together, these elements ensure that both the transportation and power systems are effectively managed, meeting the dynamic demands of EV charging while maintaining grid reliability and efficiency.

III. EVs Demand Growth Prediction Model

This section describes the dataset generation and the deep learning architecture that predicts the future EVs charging demand. The accuracy of such a prediction greatly depends on the multi-dimensional feature extraction of the coupled power-transportation system. Therefore, in order to capture the spatio-temporal dependencies of the power and transportation data, we propose to use a fusion architecture of GCNN and LSTM. The predicted future demand is then utilized in Section IV to optimize the joint allocation of CSs and DGs.

A. Dataset Generation

The proposed CSs and DGs allocation strategy is dependent upon the accurate prediction of future EVs charging demand. However, when considering a specific planning horizon, the prediction model should be trained adequately with historical data for accurate demand prediction. Thus, it is crucial to generate a dataset over a number of years to serve as the ground truth. Part of this dataset will be used for training and the rest for testing and validation of the proposed GCNN-LSTM-based prediction model. Since the historical data available for prediction is limited, we generate the temporal features of the graph in the form of time-series data that simulate the power flow within the system. To execute the power flow analysis, we employ Newton's method to determine the active and reactive power flows using MATLAB's MATPOWER toolbox.

The first step involves normalizing the load data obtained from the Electric Reliability Council of Texas (ERCOT) [48] into a scalar vector, F. The active and reactive power values from the previous timestamp are then multiplied by a scaling factor S_f derived from the growth of EV traffic data over the years. This scaling factor is computed based on the hourly traffic data, reflecting the traffic growth rate, which is then applied to the ERCOT power data. This process introduces dynamic variations in the time-series data, contributing to a comprehensive representation of charging load values. The equation for the scaling factor can be represented as:

$$S_f = 1 + \alpha \Delta T + \mathcal{N}(0, 0.01) \tag{3}$$

where $\mathcal{N}(0, 0.01)$ represents a normal distribution with mean 0 and standard deviation 0.01. ΔT represents the growth rate of the traffic data over the time horizon, and α is a proportionality constant derived from the historical traffic data analysis. This approach allows us to obtain a time-series power dataset that combines traffic growth information with raw power data, ensuring a comprehensive dataset. The hourly traffic volume at each CS informs the dataset. The hourly charging demand of EVs at a specific CS is then calculated using the charging power of each EV. This comprehensive dataset forms the basis for training and validating the GCNN-LSTM-based prediction model, ensuring accurate and robust predictions for long-term infrastructure planning.

B. Graph Data Representation and Spectral Graph Filtering

As the power system with initially integrated CSs is modeled as a graph as described in Section II, GCNN represents a very useful method to predict EVs demand growth by fusing (i) the temporal features from the power grid and transportation system (e.g., power injections, flows, EVs traffic flow) and (ii) the topological features (i.e., the spatial distribution of buses and CSs and their connectivity).

Prior to deploying GCNN, the graph data structure needs to be properly represented so that convolutional filters can be applied on it [49]. Specifically, we aim to represent the data in the spectral domain to enable the graph filtering and feature extraction operations of GCNN. Towards this objective, we start by deriving the graph Laplacian matrix to capture the features of the graph data structure with different levels of connectivity among its nodes, which in turn allows to calculate the spectral graph filters. The spectral filters enable efficient analysis of complex graph-structured data via GCNN.

The unnormalized Laplacian matrix $\mathbf{\Delta}_{\mathrm{u}}$ of graph $\mathcal{G}$ is defined as $\mathbf{\Delta}_{\mathrm{u}} = \mathbf{L} - \mathbf{W}$, where $\mathbf{L} \in \mathbb{R}^{B \times B}$ is the diagonal matrix with entries $\mathbf{L}_{i,i} = \sum_j \mathbf{W}_{ij}$. The normalized graph Laplacian matrix is given by

$$\mathbf{\Delta} = \mathbf{L}^{-1/2} \mathbf{\Delta}_u \mathbf{L}^{-1/2} = \mathbf{I} - \mathbf{L}^{-1/2} \mathbf{W} \mathbf{L}^{-1/2}, \tag{4}$$

where $\mathbf{I}$ denotes the identity matrix.

Next, the graph Fourier transform (GFT) is performed over the considered sequential data in order to represent it in frequency domain by decomposing it along a set of orthogonal basis functions. These directions constitute the eigenvectors of the Laplacian matrix. The GFT takes as input the signal $f \in \mathbb{R}^n$, where n denotes the number of features. Let $\boldsymbol{\psi}$ represent the orthonormal eigenvectors and $\mathbf{\Lambda}$ denote the associated set of ordered non-negative eigenvalues $\mu_n \geq ... \geq \mu_2 \geq \mu_1 = 0$, then the singular value decomposition of $\mathbf{\Delta}$ yields $\mathbf{\Delta} = \boldsymbol{\psi} \Lambda \boldsymbol{\psi}^\top$. The eigenvector corresponding to the smallest eigenvalue, μ_1, is $\psi_1 = (1/\sqrt{n}, \cdots, 1/\sqrt{n})$. The inverse GFT is $f = \boldsymbol{\psi} \hat{f}$. Given g, the filter response, we perform spectral convolution operation on the signal f as [50]

$$g * f = \boldsymbol{\psi} \left(\left(\boldsymbol{\psi}^\top g \right) \odot \left(\boldsymbol{\psi}^\top f \right) \right) = \boldsymbol{\psi} \, \mathrm{diag} \left(\hat{g}_1, \ldots, \hat{g}_n \right) \boldsymbol{\psi}^\top f, \tag{5}$$

where $\odot$ indicates the Hadamard product. Thereafter, the signal f is filtered by the spectral filter $\boldsymbol{H}(\mathbf{\Delta})$ and can therefore be expressed as $\boldsymbol{\psi} \boldsymbol{H} \boldsymbol{\psi} f$. However, one major limitation of this type of filter is that it only extracts features from a specific spatial location leading to a less efficient feature extraction. To overcome this issue, we consider the linear combination of the kth-order Chebyshev polynomials $C_k(\tilde{\mathbf{\Lambda}})$, with $k = 0, \ldots, K$, as in [51]

$$\boldsymbol{H}(\mathbf{\Lambda}) = \sum_{k=0}^{K} \upsilon_k C_k(\tilde{\mathbf{\Lambda}}), \tag{6}$$

where K denotes the polynomial degree, $\upsilon = (\upsilon_1, \upsilon_2, ..., \upsilon_k)$ are the coefficient vectors to be learned for the filters, and $\tilde{\Lambda} =$

$2\mathbf{\Lambda}/\mu_n - \mathbf{I}$. The Chebyshev polynomials' recursive generation formulation is $C_0 = 1$, $C_1 = x$, and $C_k(x) = 2xC_{k-1}(x) - C_{k-2}(x)$, and the filtering process is formulated as

$$\boldsymbol{\psi}\boldsymbol{H}(\mathbf{\Lambda})\boldsymbol{\psi}^\top \mathrm{f} = \boldsymbol{H}(\mathbf{\Lambda})\mathrm{f} = \sum_{k=0}^{K} \upsilon_k C_k(\tilde{\Delta})\mathrm{f}, \tag{7}$$

where $\tilde{\mathbf{\Delta}} = 2\mathbf{\Delta}/\mu_n - \mathbf{I}$. Concretely, $\mathbf{p}_k = 2\tilde{\mathbf{\Delta}}\mathbf{p}_{k-1} - \mathbf{p}_{k-2}$ is calculated recursively from the initial values $p_0 = f$ and $p_1 = \tilde{\mathbf{\Delta}}f$. Considering the sparsity of $\mathbf{\Delta}$, the computational complexity of the filtering operation $\boldsymbol{H}(\mathbf{\Lambda})\mathrm{f}$ is $\mathcal{O}(K|E_P|)$.

Due to the truncation of the Chebyshev polynomials to the Kth order, the filter can only be K-hop localized with respect to the connections in $\mathbf{\Delta}$. The aforementioned formulation of spectral convolution on the graph is thereafter used in the architecture of GCNN, which is discussed next.

C. GCNN Architecture

The GCNN architecture for predicting the EVs charging demand is presented in Fig. 2. The input information $X \in [\boldsymbol{P}_t, \boldsymbol{Q}_t] \in \mathbb{R}^{|\mathcal{N}|\times 2}$ is passed through the graph convolution layers L, then to the fully-connected layer L_f, where the softmax activation function is applied to the input X. Particularly, the jth feature map is obtained as

$$\mathbf{y}_j = \sum_{i=1}^{H_{\text{in}}} \boldsymbol{H}(\mathbf{\Delta})\mathbf{d}_i, \tag{8}$$

where $\mathbf{d}_i \in \mathbb{R}^n$ signifies the ith feature map; H_{in} stand for the number of input filters. If H_{out} denotes the number of output filters, the total number of trainable parameters present in this layer is $H_{\text{in}} \times H_{\text{out}} \times K$. Increasing H_{in} results in more efficient feature extraction from input data. Meanwhile, raising H_{out} generates a greater diversity of extracted features. In this paper, $H_{\text{in}} = 12$ is specifically chosen as it yields the optimal performance. After being transformed into a one-dimensional array, the output of the final layer is passed to the fully-connected layers.

At start, we obtain the adjacency matrix $\mathbf{W}$, the distance matrix $\mathbf{D} \in \mathbb{R}^{n\times n}$, and $\mathbf{\Delta}$ using (4). In this approach, each node is connected to its k nearest neighbors based on a distance function. The distance matrix elements, $d_{i,j}$, are calculated using Gaussian kernel function as

$$d_{i,j} = \begin{cases} e^{\left(-\frac{||x_i - x_j||^2}{2\sigma^2}\right)}, & ||x_i - x_j||^2 \leqslant \omega \\ 0, & otherwise, \end{cases} \tag{9}$$

where x_i and x_j denote the feature vectors of nodes i and j, respectively; σ represents the width of the Gaussian kernel; and ω stands for the distance threshold.

Then the lowest K_n value is measured from each row of $\mathbf{D}$ so as to get $\tilde{\mathbf{D}} \in \mathbb{R}^{n\times K_n}$, and thereafter $\sigma_D = \sum_i \tilde{\mathbf{D}}_{iK_n}/n$. After sorting, we retain the smallest K_n values in each row of W to get $\tilde{\mathbf{W}} \in \mathbb{R}^{n\times K_n}$ and calculate it as $\tilde{\mathbf{W}} = e^{-\tilde{\mathbf{D}}_{ij}^2/\sigma_D^2}$.

D. LSTM Cells Fusion

The prediction model takes the historical sequence of the GCNN features as input. In particular, the time series prediction model performs a nonlinear mapping analyzing the history-driven time-series sequence features $X = (X_1, X_2, ..., X_T)$ and their target values $y = (y_1, y_2, ..., y_T)$ to obtain the predicted value $\tilde{y_T}$, where $\tilde{y_T} = f(X, y)$. The objective is to learn the nonlinear mapping function $f(\cdot)$.

In this context, we use LSTM cells to capture the long-term dependencies between data values. The main component of LSTM is the memory cell that is capable of memorizing information over an extended period of time. Specifically, LSTM assumes a gated architecture consisting of an input gate, a forget gate, and an output gate. The input gate specifies how much new information should be inserted into the cell. It takes the current input and the prior hidden state as inputs and passes them through a sigmoid activation function. As for the output gate, it determines which components of the cell state should be given as output to the next layer based on the current input and hidden state. On the other hand, the forget gate plays a key role in determining which data from the previous time period should be discarded using a sigmoid activation function. Once the output from the GCNN layer is fed into the LSTM, the nonlinear mapping function can be learned by the gated architecture of LSTM as

$$i^t = \sigma\left(W_i \cdot [h_{t-1}, y_t] + b_i\right) \tag{10}$$

$$f^t = \sigma\left(W_f \cdot [h_{t-1}, y_t] + b_f\right) \tag{11}$$

$$o^t = \sigma\left(W_o \cdot [h_{t-1}, y_t] + b_o\right), \tag{12}$$

where $\sigma(\cdot)$ denotes the activation function; m_{t-1} and h_{t-1} indicate the state and output of LSTM cell at $t-1$, respectively; and b_i, b_f, b_o stand for the biases for input, forget, and output gate, respectively. The cell's current memory is

$$\tilde{m^t} = \sigma\left(W_c \cdot [h_{t-1}, y_t] + b_c\right), \tag{13}$$

where W_c and b_c are the weight matrix and bias, respectively. Moreover, the cell state can be represented as $m^t = f^t \odot m_{t-1} + i^t \odot \tilde{m^t}$ and cell output as $h^t = o^t \odot \tanh(m^t)$.

E. Loss Function

The loss function of the fused GCNN-LSTM architecture is formulated as:

$$L_{\text{GCNN-LSTM}} = \text{CE}(p, l) + \vartheta\|\eta\|_2, \tag{14}$$

where p is the predicted value of the model; l denotes the label; η represents all of the model's parameters; ϑ is the regularization coefficient; and $\text{CE}(p, l)$ is the cross-entropy function that determines the difference between the actual and predicted label. The last term, $\vartheta\|\eta\|_2$, reduces the overfitting of the model's learning parameters.

Next, we define the update rule of the graph convolution parameters in each iteration as

$$\beta^* = \beta^* + \gamma\frac{\partial L_{\text{GCNN-LSTM}}}{\partial \beta^*}, \tag{15}$$

where γ denotes the learning rate and $\beta^* \in \mathbb{R}^{K\times T}$ is the Chebyshev polynomial coefficient for GCNN.

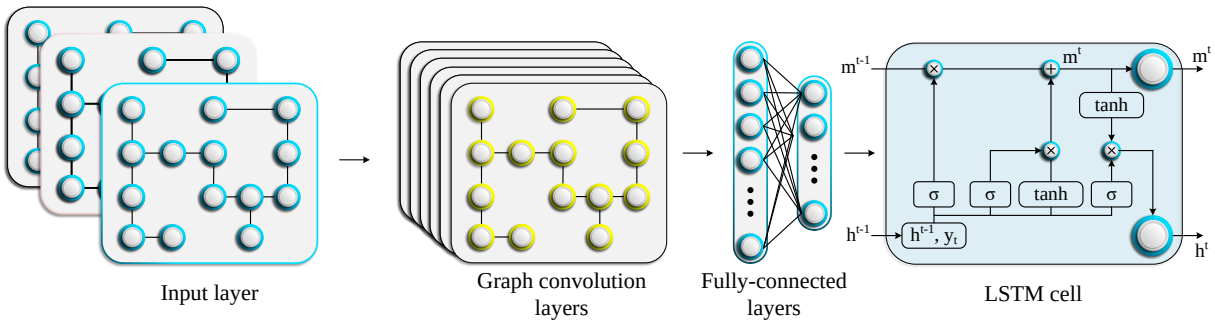

Fig. 2: Architecture of GCNN-LSTM network.

Algorithm 1 GCNN-LSTM demand prediction model

Input: Sample collection $FS \in \mathbb{R}^{n \times n_T \times CN \times n_f}$, Data label l, Chebyshev order K, Error threshold e, Maximum number of iteration MAX, Model learning rate λ, Regularization weight α, Count of LSTM hidden layer cells n_{cell}, Count of graph convolution structures n_T.
Output: The targeted parameters of GCNN-LSTM model

Initiate $\theta^* \in \mathbb{R}^{K \times n_T}$ and other parameters for learning the GCNN-LSTM model
▷ Calculate $\left(C_0(\tilde{\Delta})x, C_1(\tilde{\Delta})x, \ldots, C_{K-1}(\tilde{\Delta})x\right)$ for each sample.
for $p=0; p<n; p=p+1$ **do**
  for $q = 0; q < n_T; q=q+1$ **do**
    $x = FS_{p,q,:,:}$
    Calculate W of x according to k-NN.
    Obtain the Laplace matrix Δ of x using formula (1).
    $\tilde{\Delta} = \frac{\Delta}{\lambda_{MAX}} - I$
    Calculate $C_K(\tilde{\Delta})$ of x
    Set Chebyshev polynomial coefficients as the convolution kernel and perform the convolution.
    $temp_{p,q} = \left(C_0(\tilde{\Delta})x, C_1(\tilde{\Delta})x, \ldots, C_{K-1}(\tilde{\Delta})x\right)$
  end for
end for
$count_{step} = 0$
while $Loss > e$ || $count_{step} < MAX$ **do**
  $y_j = sigmoid(batch\ normalization(temp_{p,q} * \theta^*_j))$
  ▷ (*) indicates a convolution operation.
  Convert y_j to column vector y^*_j
  $y^* = \left(y^*_1, y^*_2, \ldots, y^*_{n_T}\right)$
  Transfer y^* to the LSTM receiving cell
  Monitor loss according to formula (13)
  if $loss < e$ **then**
    Break
  end if
  Obtain LSTM parameters according to loss and BP algorithm
  Obtain GC parameters according to formula (14)
  $count_{step}$ ++
end while

F. Integration of transformer model

In addition to the LSTM model, we explored the use of a Transformer model after the graph convolution layer to investigate its effectiveness separately. Transformer models leverages self-attention mechanisms to capture dependencies in the data, allowing for effective modeling of time-series data. The input sequence to the transformer model is represented as $\mathbf{X} = (X_1, X_2, ..., X_T)$. In this context, The encoder-decoder based Transformer model consists of l_T identical layers. Each layer comprises two primary components: the multi-head self-attention mechanism and the feed-forward neural network. The formulation of the transformer model is discussed next.

1) Embedding Layer: Each time step X_T is mapped to a high-dimensional space using an embedding layer, $E_T = \text{Embedding}(X_T)$. To incorporate temporal information, a position vector $\mathbf{P}_T$ is added to the embedding as

$$\mathbf{Z}_T = \mathbf{E}_T + \mathbf{P}_T. \tag{16}$$

The position encoding $\mathbf{P}_t$ is defined as:

$$\mathbf{P}_{T,2i} = \sin\left(\frac{T}{10000^{2i/d}}\right), \quad \mathbf{P}_{T,2i+1} = \cos\left(\frac{T}{10000^{2i/d}}\right)$$

where d is the number of features in each position encoding vector and i is the feature dimension index.

2) Self-Attention Mechanism: Self-attention is used to compute a weighted sum of input features, allowing the model to focus on different parts of the input sequence. For each time step T, the self-attention mechanism, A_s computes

$$A_{self}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}. \tag{17}$$

where:

- $\mathbf{Q} = \mathbf{W}_Q \mathbf{Z}_T$ is the query matrix,
- $\mathbf{K} = \mathbf{W}_K \mathbf{Z}_T$ is the key matrix,
- $\mathbf{V} = \mathbf{W}_V \mathbf{Z}_T$ is the value matrix,
- $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V$ are learnable weight matrices,

To allow the model to attend to information from different representation subspaces, multi-head attention, A_m is used as

$$A_m(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = [\text{head}_1, \text{head}_2, \ldots, \text{head}_h]\mathbf{W}_O \tag{18}$$

where each head_i is an attention mechanism with its own learnable parameters: $\text{head}_i = A_s(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ and $\mathbf{W}_O$ is the output weight matrix.

3) Feed-Forward Neural Network: Each position's output from the multi-head attention layer is passed through a feed-forward neural network:

$$\mathbf{F}(\mathbf{Z}_t) = \max(0, \mathbf{Z}_t\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2$$

where $\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}_1, \mathbf{b}_2$ are learnable parameters.

4) Output Layer: Finally, the output layer predicts the next value in the sequence:

$$\hat{x}_{t+k} = \mathbf{O}_{t+k}\mathbf{W}_p + \mathbf{b}_p$$

where k represents the forecast horizon, which is the number of time steps ahead for which the prediction is made.$\mathbf{W}_p$ and $\mathbf{b}_p$ are the parameters of the output layer.

G. Prediction Algorithm

The GCNN-LSTM prediction algorithm is depicted in Algorithm 1. It takes the data sample, learning rate, regularization weight, the number of LSTM hidden layers, as well as the number of graph convolution structures as input, and outputs the desired LSTM parameters. After initializing the learning parameters, Algorithm 1 calculates the Chebyshev polynomial coefficient. For each sample, it calculates the adjacency and Laplacian matrices to obtain the Chebyshev polynomial coefficient (lines 3-13). Afterwards, Algorithm 1 creates feature maps and performs convolution operations on them. Then, the extracted features are sent to the receiving cells of LSTM. At line 21, Algorithm 1 calculates the loss function according to Eq. (14). This process keeps iterating until a minimum error threshold is achieved. At the final stage (line 25), Algorithm 1 updates the LSTM parameters based on the loss function and the backpropagation algorithm. Additionally, with an NVIDIA GeForce RTX 3080 hardware accelerator, and the training time for the optimal model is recorded approximately as 2 hours and 30 minutes. Moreover, the training and validation curve of the proposed prediction model is presented in Fig. 3.

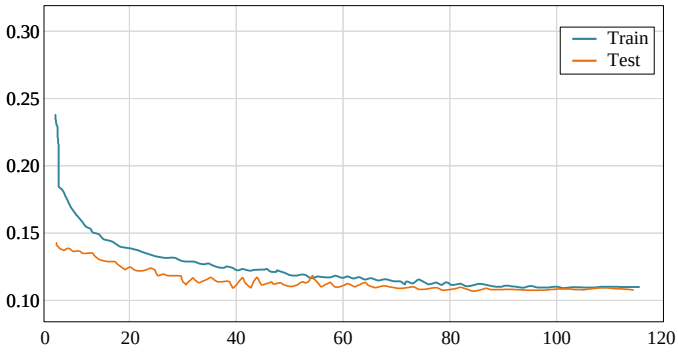

Fig. 3: Training and testing curve of the proposed model.

IV. Problem Formulation

In this section, the optimization problem of the CSs-DGs allocation problem over a time horizon $\mathcal{S}$ is formulated as

$$\min_{N_{\text{CS}},N_{\text{DG}}} \sum_{s\in\mathcal{S}} \Big(\mu_{\text{tr}} C_{\text{tr},s} + \mu_{\text{gr}} C_{\text{gr},s}\Big), \tag{19}$$

$$\text{where } C_{\text{tr},s} = c_{\text{CS}} \sum_{b\in B} |\Omega^{\text{CS}}_{b,s}| + \sum_{b\in B}\sum_{q\in\mathcal{Q}_b} c_{\text{pen}} I_{q,s} F_{q,s} \zeta_b$$

$$\text{and } C_{\text{gr},s} = c_{\text{pg}} \sum_{b\in B}\sum_{i\in\{\Omega^{\text{CS}}_{b,s}\}} P^{\text{CS}}_{i,b} + c_{\text{DG}} \sum_{b\in B} |\Omega^{\text{DG}}_{b,s}|$$

s.t. *Planning Constraints:*

$$\begin{cases} L'_{b,s} > 0, \text{ if } N'_{\text{CS},b,s} \geq 1, \forall b \in B, \forall s \in \mathcal{S} \\ N_{\text{CS},b,s} + N'_{\text{CS},b,s} \leq N^{\text{th}}_{\text{CS},b,s}, \forall b \in B, \forall s \in \mathcal{S} \\ 0 < \sum_{s\in\mathcal{S}} B_{\text{tot},s} \leq B_{\text{tot}}, \forall s \in \mathcal{S} \end{cases} \tag{20a}$$

$$N_{\text{CS}}, N_{\text{DG}} \geq 0. \tag{20b}$$

The first term in the objective function in Eq.(19) corresponds to the cost of the transportation system, which includes the cost of installing CSs and the cost of unsatisfied EVs charging demand. The second term in Eq.(19) corresponds to the cost of the power system, which includes the cost of the power drawn by the CSs and the cost of installing new DGs.

The constraints of the optimization problem include power system operational constraints as described in Eq. (1a); the power-transportation system constraints as illustrated in Eq. 2a; and the budget constraints in Eq. 20a. As for the budget constraints in Eq. (20a), the first constraint indicates that an additional load $L'_{b,s}$ is required by bus b if new CSs are to be installed; the second constraint depicts that the number of total CSs at bus b remains within the maximum allowable limit; and the third constraint ensures that the cumulative budget at stage s should not exceed the total budget. Finally, Eq. (20b) ensures the non-negativity of CSs and DGs.

These constraints of the optimization problem involve nonlinear relationships and interactions between different power variables, which introduce non-convexity into the problem. Specifically, the power balance equations relate the active and reactive power injections ($P_{b,\tau}$ and $Q_{b,\tau}$) to the power generated, consumed, and exchanged between different buses. These relationships involve nonlinear terms such as the product of voltage magnitudes and phase angle differences $\cos(\theta_i - \theta_j)$ and $\sin(\theta_i - \theta_j)$, which are inherently non-convex. These non-linearities make the feasible region of the optimization problem non-convex. The optimization problem also involves integer variables including the upper bound of the number of CSs, and DG units. Thus the formulated optimization problem is a mixed-integer non-liner problem

This non-convexity of the functions do not guarantee that a local optimum is also the global optimum, making it complex to find the absolute best solution. This complexity arises from the interaction between continuous variables such as power flows and voltages limits and discrete variables such as the number of CSs and DGs. Searching for global optimum within the large search space is computationally intensive and often impractical for large-scale problems like ours where we considered a 2000-bus power system topology. However, to achieve a near-optimal solution we employ a multi-start strategy where the optimization algorithm is executed multiple times, each time with a randomly generated initial starting point. By exploring diverse regions of the solution space, this technique ensures that the search space is covered more comprehensively. In each run of the optimizer, the solution converges to a local optimum based on the starting point. The best solution among all runs is selected as the final solution. It is worth mentioning that, the multi-start strategy allows for a thorough exploration of the solution space, ensuring that high-quality solutions are identified. These solutions are typically very close to the global optimum and provide valuable insights and practical recommendations for the problem at hand. In the context of our power and transportation system optimization, these solutions remain valid and reliable and can offer actionable guidance for real-world implementation.

V. Solution Approach

In this section, we present an efficient approach based on finite-horizon constrained MDP to solve the multi-stage allocation problem of CSs and DGs.

We start by defining state and action space, and the reward function corresponding to each stage. Then, we formulate the dynamic program based on the finite-horizon episodic model.

$\mathcal{U}$-State Space:

In the joint allocation problem of EV CSs and DG units, the state space consists of variables describing the predicted demand at a bus, the allocated budget at a time stage, the generation capacity of a bus, and the time periods over which the CSs and DG units will be allocated. Therefore, the state-space ($\mathcal{U}$) is expressed as

$$\mathcal{U} = \{u | u_s = (s_1, s_2, s_3, \cdots)\}, \tag{21}$$

where s_i denotes the i^{th} state variable and u_s indicates the state of the MDP at s^{th} time stage. Since all the constraints are known in advance, and the traffic flow is dependent on the locations of EVs, it is evident that u includes all the necessary information for taking an action.

The states are causally and sequentially linked, i.e., $u_s - u_{s-1} - u^{s-2}$ forms a Markov chain for all time stages $s \in \mathcal{S}$ and u^{s-2} represents the aggregation of the preceding $s-2$ states [52]. Hence, the optimization problem formulated in this paper presents a finite-horizon-constrained MDP.

$\mathcal{A}$-Action Space:

We define the action space, $\mathcal{A}$, as

$$\mathcal{A} = \{a | a_s\}, \tag{22}$$

where a_s indicates the action executed at time stage s. The possible actions that the model can take are allocating a CS(s) at a particular location; allocating DG(s) at a location where additional capacity is required; or simultaneously allocating both DGs and CSs at a particular location. The specific actions taken depend on the objective function and the constraints of the EV CSs and DGs joint allocation problem.

$\mathcal{R}$-Reward Function:

In this paper, we use a potential-based reward shaping function that defines the reward for a state-action pair, (u_s, a_s), at a particular time stage s as

$$R(u_s, a_s) = R_{s-1} - R_s, \tag{23}$$

where R_s is defined as in Eq.(19) and for the first stage R_{s-1} is considered as 0.

$$R_s = \mu_{\text{tr}} C_{\text{tr}}(s) + \mu_{\text{gr}} C_{\text{gr}}(s). \tag{24}$$

Hence, we can decompose the reward function calculation into two parts as

$$R_s^1 = \sum_{i \in B} C_{\text{gr}} m_s^i, \tag{25}$$

$$R_s^2 = \mu_{\text{tr}} C_{\text{tr}}(s) + \mu_{\text{gr}} C_{\text{gr}}(s), \tag{26}$$

where m_s^i is a binary variable that is 1 when a CS is allocated to bus i, and 0 otherwise. If no feasible solutions are found, then the cost is set to a high value. The rewards assigned to each action indicate whether they improve or worsen the solution. At each time stage, a policy, π_s, is defined based on the action space $(a_s|u_s)$. The optimal policy is defined as

$$\pi_s^*(a_s \mid u_s) = \text{argmax}_\pi \, \mathbb{E}\left[\sum_{s=1}^{\mathcal{S}} \rho^{(s-1)} R(u_s, a_s)\right], \tag{27}$$

where ρ denotes the discounting factor that balances the short-term and long-term rewards. The short-term reward is associated with the actions and outcomes of the nearest future stage, and the long-term reward is associated with the actions and outcomes of the further future stage.

$\mathcal{H}$-Horizon:

Based on the definitions of the state space, action space, and reward function, we interpret the constrained finite-horizon MDP problem as an episodic model, where an agent interacts at each time stage $s \in \mathcal{S}$. A full episode is defined as a sequence $\{(u_0, a_0), (u_1, a_1), ..., (u_{|\mathcal{S}|}, a_{|\mathcal{S}|})\}$. Here $u_{|\mathcal{S}|}$ is the terminating state which is activated if the change in the reward over the time stage, $s = |\mathcal{S}|$, is less than a pre-specified value ϵ. This hyper-parameter is used to determine the convergence of a single episode solution. The iterative computation process for a single episode of the finite-horizon constrained MDP is given by Algorithm 2.

RL allows for modeling the dependencies of power-transportation networks by learning an optimal policy that adapts to changing states over multiple stages. Moreover, RL's ability to balance exploration and exploitation ensures that the model can discover and optimize complex action spaces, such as deciding where and how many CSs and DGs to allocate. The traditional solvers might not achieve this efficiently due to the combinatorial explosion of possibilities, where the number of potential combinations of actions becomes too large to handle effectively. The state space of the RL model includes four state variables as stated earlier in this section. After the RL model selects an action, which might involve allocating a certain number of CSs and/or DGs at specific locations, the model minimizes the cost function in Eq.19 to ensure that all relevant constraints are met. Once these constraints are checked and satisfied, the reward function is updated based on the effectiveness of the allocation. This iterative process continues until an optimal or near-optimal allocation strategy is developed.

Algorithm 2 describes the functions of CSs allocation, 'allocateCS', and DGs allocation 'calcDGs'. The first function allocates all CSs, $N'_{\text{CS},s}(s)$, which is obtained from the proposed solution approach. The total number of CSs is allocated at each candidate bus during a time stage s. 'allocateCS' first calculates the demand difference (DD) for each bus (line 5), and then allocates a single CS at the bus with the highest DD (line 6). Afterward, the function allocates a CS at the bus with the second-largest DD and the process continues until all CSs at the deployment stage s have been allocated (lines 9 to 11) or all candidate buses have had their initial CSs allocations (line 4). In the case when all buses have been allocated a single CS and there are CSs left to allocate, the DDs of buses are calculated again (line 14), and a single CS is allocated to the bus with the largest DD (line 15). The process repeats until all CSs have been deployed (lines 19 to 21).

To account for the anticipated increase in demand brought on by the dynamic allocation of CSs, the second function, 'calcDGs', is introduced. This function calculates the number of DG units to install at each bus at each time stage s. The buses that have additional CSs allocated by function 'allocateCS' may require an extra load capacity. The extra load capacity is calculated at line 27, and it satisfies the power constraints in Eq. (1a).

Algorithm 2 Defining Functions for CSs and DG Units Allocation

1: **function** ALLOCATECS($(B, DD(s), N'_{\mathrm{CS},s}(s)) : N'_{\mathrm{CS},b}$)
2: ▷ This function allocates $N'_{\mathrm{CS},b}(s)$ CSs at bus $b \in B$
3: $count \leftarrow 0$
4: **while** $B(s) \neq \emptyset$ **do**
5: Find a bus $b \in B$ with largest DD_b
6: Allocate a single CS on $b : N'_{\mathrm{CS},b}(s) \leftarrow 1$
7: count $\leftarrow$ count +1
8: $\{B(s)\} \setminus b$
9: **if** count $== N'_{\mathrm{CS},s}(s)$ **then**
10: Break
11: **end if**
12: **end while**
13: **while** count $< N'_{\mathrm{CS},s}(s)$ **do**
14: Find a bus $b \in B$ with largest DD_b
15: Allocate a single CS on $b : N'_{\mathrm{CS},b}(s) \leftarrow$
16: $N'_{\mathrm{CS},b}(s) + 1$
17: count $\leftarrow$ count +1
18: $\{B(s)\} \setminus b$
19: **if** count $== N'_{\mathrm{CS},s}(s)$ **then**
20: Break
21: **end if**
22: **end while**
23: **end function**
24: **function** CALDGS($(B, N'_{\mathrm{CS},b}, L_b(s), P^R_{b,\tau})$):$N'_{\mathrm{DG},b}$
25: ▷ This function allocates DG units at candidate bus
26: **while** $\{B(s)\} \neq \emptyset$ **do**
27: Additional load capacity required at bus $b \in B$: $L_b(s)' \leftarrow \left| L_b(s) - P^R_{b,\tau} \right| N'_{\mathrm{DG},b}$
28: $\{B(s)\} \setminus b$
29: **end while**
30: **end function**

A. Computational Complexity

The computational complexity of the Q-learning algorithm is associated with the size of the state-action space, training episodes, and the size table, which represent the Q-value function. For the state space $\mathcal{U}$, action space $\mathcal{A}$, and the number of time stages $\mathcal{S}$, the time complexity of the proposed Q-learning algorithm is given by $\mathcal{O}(\mathcal{UAS})$ as the algorithm iterates through each state-action pair for each episode. Moreover, the number of feasible state-action pairs is equal to the product of the number of considered states and actions. Therefore, the time complexity is related linearly to the number of episodes. Moreover, the space complexity of Q-learning is given by $\mathcal{O}(\mathcal{UA})$, as the Q-value function must be stored for each state-action pair. Therefore, the algorithm's space complexity increases linearly with the sizes of the state and action space.

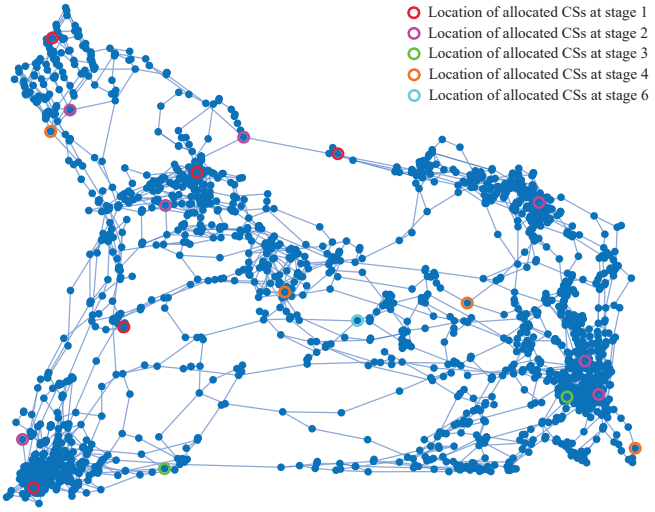

Fig. 4: Graph structure of Texas 2000-bus test case and its interconnection with transportation nodes.

VI. Experimental Results

In this section, we present a case study for the proposed dynamic multi-stage (six time stages) CSs-DGs allocation strategy on the synthetic 2000-bus power grid of Texas [53] with initially 720 integrated CSs. The geographical locations of the existing CSs are obtained from [54]. The power system graph is shown in Fig. 4. We present and compare four different allocation strategies in terms of EVs charging demand satisfaction ratio when no prediction of EVs traffic demand is considered, and when the EVs traffic demand is predicted using i) power system data only, ii) transportation data only, and iii) a combination of power and transportation data. The simulation parameters considered are depicted in Table I.

TABLE I: Simulation parameters.

Total number of power bus, B = 2000
Total number of already available charging stations = 318
Planning period, $\mathcal{S}$ = 6
DG unit cost, c_{DG} = $ 1104.032/kWh [44]
Number of GCNNs layer = 6
Number of Chebyshev coefficients = 4
Number of LSTM hidden layer cells = 30
Prediction model maximum iteration, MAX = 100000
Prediction model Error threshold, e = 0.1
Prediction model learning rate, λ = 0.003
Optimization model learning rate = 0.1
Optimization model discount factor = 0.99
Optimization model exploration rate = 0.1

A. Effect of budget on user satisfaction

We start by obtaining the predicted charging demand using GCNN-LSTM approach as described by Section III. The power dataset consists of five-year values for the active and reactive power demands generated using Monte Carlo simulations. As for the transportation data, the traffic flow information at the CSs of Texas state from 2016 to 2020 is obtained from [54]. To generate the combined power-transportation dataset, we start by calculating the difference

of EVs charging demand at two consecutive time stages, and then the power data is scaled proportionally to that difference. The user satisfaction rate is calculated as follows:

$$\phi_s = \frac{\sum_{b\in B}\sum_{i\in\{\Omega^{\text{CS}}_{b,s}\}} P^{\text{CS}}_{i,b}}{\sum_{b\in B}\sum_{d\in\{\Omega^{\text{CS}}_{b,s}\}} \psi_{d,\tau}\left(z^d\right) P_{\text{EV}}} \times 100\%, \tag{28}$$

where $\Omega^{\text{CS}}_{b,s} = N_{\text{CS},b,s} + N'_{\text{CS},b,s}$, which is the sum of existing CSs and the new CSs that are installed at time stage s; $P^{\text{CS}}_{i,b}$ denotes the power demand of CS i connected to bus b; and P_{EV} represents the power of each EV. To obtain the budget required to satisfy at least 80% of EVs charging requests, we plot in Fig. 5 the EVs average charging requests satisfaction rate for different budget values. As can be seen from Fig. 5, the user satisfaction rate increases with an increase in the budget, indicating the necessity of allocating enough resources to provide a better-charging demand satisfaction rate.

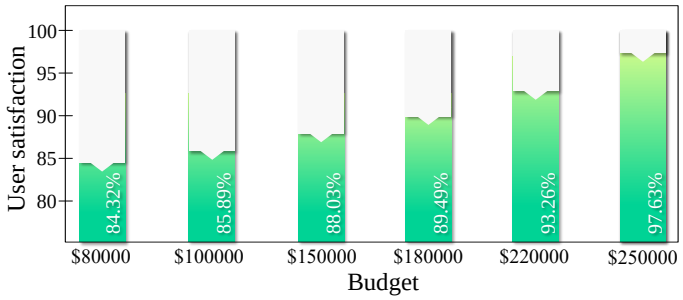

Fig. 5: User satisfaction rate for different budget values.

B. *Effect of planning strategy on user satisfaction rate*

Using the predicted demand and a budget value of $\$250,000$ as an example, we solve the joint allocation problem over $\mathcal{S}$ stages using Q-learning, which yields the minimum number of CSs and DGs required at each time stage to satisfy 80% of EVs charging requests. As for the locations of the allocated CSs and DGs, they are specified according to Algorithm 2.

Fig. 6 shows the demand satisfaction rate achieved by the four different aforementioned allocation strategies over six deployment stages. From Fig. 6, we can see that the strategy that combines both power and transportation data to predict future demand outperforms the other three strategies. This is explained by the fact that the fusion of power and transportation data helps to recognize the patterns in the usage of EVs and the power grid. This, in turn, provides a more comprehensive understanding of the spatial and temporal distribution of charging demand and ensures a more efficient usage of budgets, while maximizing the satisfaction rate.

On the other hand, when transportation data is used solely to predict the future charging demand, the model ignores information about the power capacity expansion and the availability of the power grid. Moreover, overlooking the power grid data greatly impacts the estimation of DGs, as this strategy ignores the load growth information over the time stages. Thus, this strategy leads to a less accurate estimation of the charging demand and EVs satisfaction rate. Similarly, the estimation of future demand using power dataset only ignores the traffic flow information over the time stages. Hence, the power data alone fails to obtain a holistic view of the charging demand,

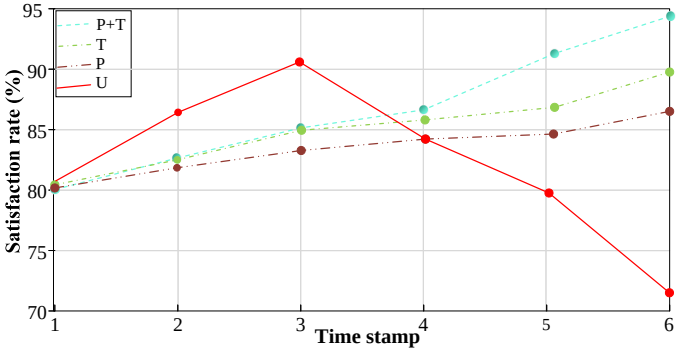

Fig. 6: EVs charging demand satisfaction rate over 6 stages.

leading to a decline in EVs satisfaction rate. Furthermore, the no-prediction strategy seems to outperform the predicted ones but only for the initial time stages, where it achieves more than 90% satisfaction rate at the third deployment stage. However, this strategy is not feasible for long-term planning of EVs charging infrastructure. As can be seen from Fig. 6, the satisfaction rate for the non-predicted strategy drops significantly after the third stage. This is explained by the fact that the budget has ran out for future subsequent stages. Conversely, the strategy that uses a combination of power and transportation data for prediction achieves nearly 95% satisfaction rate at the sixth deployment stage. This proves the superiority of the proposed GCNN-LSTM based structure.

C. *CSs and DGs allocation over the planning periods*

Table II presents the optimal number and locations of CSs and DGs capacity and their corresponding installation costs over the deployment horizon $\mathcal{S}$ when the demand prediction is based on the combined power and transportation data. We see from Table II that to achieve at least 90% EVs demand satisfaction rate, 12 CSs need to be installed in the first stage, additional 18 CSs are installed in the second stage, 3 additional CSs are installed in the third stage, 9 additional CSs are installed in the fourth stage, no CSs need to be installed in the fifth stage, and 2 additional CSs are installed in the last stage. Moreover, Table II also shows the location and the capacity required to fulfill the EVs charging demand at each time stage.

VII. Performance Comparison

A. *Benchmark Detectors*

In this section, a comprehensive evaluation is conducted to justify the prediction performance of the proposed GCNN-LSTM model with established benchmark counterparts. Table III presents the predictive performance of various benchmark methodologies across different allocation stages in terms of Normalized Mean Absolute Error (NMAE) and Normalized Root Mean Squared Error (NRMSE) metrics [55]. The prediction model assessed in the table includes CNN, feed-forward neural neywork (FNN), support vector machine (SVM), and autoregressive integrated moving average (ARIMA) models. The adopted benchmark models represent diverse attributes, encompassing structure (shallow/deep/graph) and training methodology (unsupervised/supervised).

TABLE II: Optimal number and locations of CSs and DGs with the corresponding costs over the six stages of deployment.

Stage 1			**Stage 2**			**Stage 3**			**Stage 4**			**Stage 5**			**Stage 6**		
Bus #	# CSs	DGs (kWh)	Bus #	# CSs	DGs (kWh)	Bus #	# CSs	DGs (kWh)	Bus #	# CSs	DGs (kWh)	Bus #	# CSs	DGs (kWh)	Bus #	# CSs	DGs (kWh)
298	3	3.8	197	3	8.6	539	2	4.8	76	3	8.6				18	2	4.8
490	3	4.8	365	3	8.6	756	1	-	259	2	4.8						
961	2	-	770	3	8.6				313	2	4.8						
1230	2	-	1480	3	8.6				1467	2	-						
1296	2	-	1876	3	8.6												
			1921	2	4.8												
			1930	2	4.8												
Cost($)	78857.14	9494.67		28212.12	58072.08		6021.60	5299.35		17550.32	20093.38					3716.98	5299.35
Total cost ($)									232616.99								

B. Hyperparameter Optimization

For hyparameter optimization, we use a grid-search hyperparameter selection strategy, where each hyperparameter is chosen within a specific stage. For instance, in the case of ARIMA, the parameters for differencing and moving averages are fixed at 1 and 0, respectively. On the other hand, SVM utilizes the scale and sigmoid kernels for the kernel and gamma parameters. CNN's design incorporates four layers with 32 units, a neighborhood order of 5, Rmsprop optimizer, and a ReLU activation function. Notably, for the proposed model, there were 4 layers in each stage with 32 units, 3 neighborhood orders, Adam optimizer, and a ReLU activation.

C. Comparison Results

The outcomes depicted in Table III exhibits the GCNN-LSTM model's superior predictive capability across all evaluated time spans. In addition, the table depicts a consistent trend: as the prediction horizon progresses, errors invariably escalate. Specifically, the ARIMA model emerges as the least accurate prediction scheme, struggling to adapt to evolving data patterns and trends. Though the SVM and FNN based scheme surpass the performance of ARIMA model, their performance still falls short of expectations, with error rates exceeding 20%. In comparison, the CNN model displays better accuracy than the ARIMA, SVM, and FNN counterparts, yet it remains outperformed by the proposed model. This outcome could stem from CNN's primary emphasis on extracting local features, potentially inhibiting its ability to effectively capture temporal dependencies embedded within the data. Remarkably, the proposed model outperforms the compared models by reducing errors by 15-20%. The results of our experiments demonstrated that the Transformer model consistently outperformed the LSTM model across all evaluation metrics. The Transformer model achieved lower MAE and RMSE values, indicating higher accuracy in predicting future EV charging demand. However, since both the LSTM and Transformer models achieved comparable performance, the allocation results remain unaffected. One possible explanation for this is that both models, when combined with GCNN, effectively capture the temporal and spatial dependencies in the data, leading to accurate predictions of future time-series data This comparison highlights the superiority of the proposed model in capturing intricate patterns and dependencies within the interconnected data, resulting in improved precision and demand prediction accuracy.

TABLE III: Comparison of prediction performance with established benchmark counterparts.

Forecasting model	Time stage	NMAE [%]	NRMSE [%]
ARIMA	4	29.12	33.41
	12	31.86	34.79
	24	34.56	36.01
SVM	4	19.53	20.33
	12	20.19	21.43
	24	21.08	22.97
FNN	4	18.65	20.65
	12	19.21	20.99
	24	19.93	21.56
CNN	4	12.33	11.00
	12	12.94	11.61
	24	13.84	13.74
GCNN-LSTM	**4**	**4.41**	**6.27**
	12	**4.50**	**6.31**
	24	**4.69**	**6.97**
GCNN-Transformer	**4**	**4.29**	**5.98**
	12	**4.38**	**6.20**
	24	**4.63**	**6.49**

VIII. Conclusions

In this paper, we have proposed a dynamic spatio-temporal joint allocation framework to address the optimal CSs and DGs deployment problem using the predicted future EVs power demand. The proposed framework incorporated two main tasks: i) a prediction task and ii) an allocation task. First, the prediction task predicted the future EV charging demand based on data from a power grid coupled with the transportation system. For this purpose, we have used a GNN-LSTM fusion architecture which captures the dynamic and complex relationships of the coupled network. Second, the predicted future demand was used as input to the multi-stage dynamic CSs and DGs allocation optimization problem. The optimization problem minimized the power and CSs/DGs costs

subject to power balance constraints, CSs/DGs constraints, and budget constraints. A Q-learning approach was adopted to determine the optimal number of CSs and DGs to install at each time stage. Simulations were conducted on Texas power grid with 720 initially integrated CSs. Our investigations revealed that using the predicted demand from the combined power and transportation data achieves the maximum satisfaction rate over the deployment stages. Thus, exploiting such a coupled power-transportation data enables a more efficient and sustainable planning of CSs and DGs, while simultaneously enhancing the power grid's stability and dependability. For future work, we recommend investigating the use of advanced energy storage solutions and their impact on balancing the intermittency of non-dispatchable DGs, which can provide deeper insights of this work.

REFERENCES

[1] P. Xu *et al.*, "Real-time fast charging station recommendation for electric vehicles in coupled power-transportation networks: A graph reinforcement learning method," *International Journal of Electrical Power & Energy Systems*, vol. 141, p. 108030, 2022.

[2] N. Zhou *et al.*, "China's energy and emissions outlook to 2050: Perspectives from bottom-up energy end-use model," *Energy Policy*, vol. 53, pp. 51–62, 2013.

[3] Z. Yi, X. C. Liu, and R. Wei, "Electric vehicle demand estimation and charging station allocation using urban informatics," *Transportation Research Part D: Transport and Environment*, vol. 106, p. 103264, 2022.

[4] K. Chaudhari *et al.*, "Agent-based aggregated behavior modeling for electric vehicle charging load," *IEEE Trans. Ind. Informat.*, vol. 15, no. 2, pp. 856–868, 2019.

[5] R.-C. Leou, C.-L. Su, and C.-N. Lu, "Stochastic analyses of electric vehicle charging impacts on distribution network," *IEEE Trans. Power Syst.*, vol. 29, no. 3, pp. 1055–1063, 2014.

[6] T. Shun, L. Kunyu, X. Xiangning, W. Jianfeng, Y. Yang, and Z. Jian, "Charging demand for electric vehicle based on stochastic analysis of trip chain," *IET Generation, Transmission & Distribution*, vol. 10, no. 11, pp. 2689–2698, 2016.

[7] T. Chen, B. Zhang, H. Pourbabak, A. Kavousi-Fard, and W. Su, "Optimal routing and charging of an electric vehicle fleet for high-efficiency dynamic transit systems," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3563–3572, 2016.

[8] G. Dong, J. Ma, R. Wei, and J. Haycox, "Electric vehicle charging point placement optimisation by exploiting spatial statistics and maximal coverage location models," *Transportation Research Part D: Transport and Environment*, vol. 67, pp. 77–88, 2019.

[9] J. Zhu, Z. Yang, M. Mourshed, Y. Guo, Y. Zhou, Y. Chang, Y. Wei, and S. Feng, "Electric vehicle charging load forecasting: A comparative study of deep learning approaches," *Energies*, vol. 12, no. 14, p. 2692, 2019.

[10] H. Jahangir, H. Tayarani, A. Ahmadian, M. A. Golkar, J. Miret, M. Tayarani, and H. O. Gao, "Charging demand of plug-in electric vehicles: Forecasting travel behavior based on a novel rough artificial neural network approach," *Journal of Cleaner Production*, vol. 229, pp. 1029–1044, 2019.

[11] T. Unterluggauer, K. Rauma, P. Järventausta, and C. Rehtanz, "Short-term load forecasting at electric vehicle charging sites using a multivariate multi-step long short-term memory: A case study from finland," *IET Electrical Systems in Transportation*, vol. 11, no. 4, pp. 405–419, 2021.

[12] Y. Dai, Q. Zhou, M. Leng, X. Yang, and Y. Wang, "Improving the bi-lstm model with xgboost and attention mechanism: A combined approach for short-term power load prediction," *Applied Soft Computing*, vol. 130, p. 109632, 2022.

[13] W. Wang, C. Zhou, H. He, W. Wu, W. Zhuang, and X. Shen, "Cellular traffic load prediction with lstm and gaussian process regression," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–6.

[14] M. J. Gul, G. M. Urfa, A. Paul, J. Moon, S. Rho, and E. Hwang, "Mid-term electricity load prediction using cnn and bi-lstm," *The Journal of Supercomputing*, vol. 77, pp. 10 942–10 958, 2021.

[15] G. Zhang, X. Bai, and Y. Wang, "Short-time multi-energy load forecasting method based on cnn-seq2seq model with attention mechanism," *Machine Learning with Applications*, vol. 5, p. 100064, 2021.

[16] Z. Pan, J. Wang, W. Liao, H. Chen, D. Yuan, W. Zhu, X. Fang, and Z. Zhu, "Data-driven ev load profiles generation using a variational auto-encoder," *Energies*, vol. 12, no. 5, p. 849, 2019.

[17] A. Zaboli, V.-N. Tuyet-Doan, Y.-H. Kim, J. Hong, and W. Su, "An lstm-sae-based behind-the-meter load forecasting method," *IEEE Access*, 2023.

[18] G. Ramesh, J. Logeshwaran, T. Kiruthiga, and J. Lloret, "Prediction of energy production level in large pv plants through auto-encoder based neural-network (auto-nn) with restricted boltzmann feature extraction," *Future Internet*, vol. 15, no. 2, p. 46, 2023.

[19] M. P Sasidharan, S. Kinattingal, and S. Simon, "Comparative analysis of deep learning models for electric vehicle charging load forecasting," *Journal of The Institution of Engineers (India): Series B*, vol. 104, no. 1, pp. 105–113, 2023.

[20] Y. Li, S. He, Y. Li, L. Ge, S. Lou, and Z. Zeng, "Probabilistic charging power forecast of evcs: Reinforcement learning assisted deep learning approach," *IEEE Trans. Intell. Veh.*, vol. 8, no. 1, pp. 344–357, 2023.

[21] J. Zhang, C. Liu, and L. Ge, "Short-term load forecasting model of electric vehicle charging load based on mccnn-tcn," *Energies*, vol. 15, no. 7, p. 2633, 2022.

[22] D. Zhou, Z. Guo, Y. Xie, Y. Hu, D. Jiang, Y. Feng, and D. Liu, "Using bayesian deep learning for electric vehicle charging station load forecasting," *Energies*, vol. 15, no. 17, p. 6195, 2022.

[23] X. Zhang, K. W. Chan, H. Li, H. Wang, J. Qiu, and G. Wang, "Deep-learning-based probabilistic forecasting of electric vehicle charging load with a novel queuing model," *IEEE Transactions on Cybernetics*, vol. 51, no. 6, pp. 3157–3170, 2020.

[24] L. Kuan, Z. Yan, W. Xin, C. Yan, P. Xiangkun, S. Wenxue, J. Zhe, Z. Yong, X. Nan, and Z. Xin, "Short-term electricity load forecasting method based on multilayered self-normalizing gru network," in *2017 IEEE Conference on Energy Internet and Energy System Integration (EI2)*. IEEE, 2017, pp. 1–5.

[25] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *The 41st international ACM SIGIR Conference on Research & Development in Information Retrieval*, 2018, pp. 95–104.

[26] P. Tadayon-Roody, M. Ramezani, and H. Falaghi, "Multi-objective locating of electric vehicle charging stations considering travel comfort in urban transportation system," *IET Generation, Transmission & Distribution*, vol. 15, no. 5, pp. 960–971, 2021.

[27] M. Z. Zeb, K. Imran, A. Khattak, A. K. Janjua, A. Pal, M. Nadeem, J. Zhang, and S. Khan, "Optimal placement of electric vehicle charging stations in the active distribution network," *IEEE Access*, vol. 8, pp. 68 124–68 134, 2020.

[28] L. Chen, C. Xu, H. Song, and K. Jermsittiparsert, "Optimal sizing and sitting of evcs in the distribution system using metaheuristics: A case study," *Energy Reports*, vol. 7, pp. 208–217, 2021.

[29] S. Deb, X.-Z. Gao, K. Tammi, K. Kalita, and P. Mahanta, "A novel chicken swarm and teaching learning based algorithm for electric vehicle charging station placement problem," *Energy*, vol. 220, p. 119645, 2021.

[30] S. N. Hashemian, M. A. Latify, and G. R. Yousefi, "Pev fast-charging station sizing and placement in coupled transportation-distribution networks considering power line conditioning capability," *IEEE Trans. Smart Grid*, vol. 11, no. 6, pp. 4773–4783, 2020.

[31] A. Pal, A. Bhattacharya, and A. K. Chakraborty, "Allocation of electric vehicle charging station considering uncertainties," *Sustainable Energy, Grids and Networks*, vol. 25, p. 100422, 2021.

[32] Y. Zhang *et al.*, "Efficient deployment of electric vehicle charging infrastructure: Simultaneous optimization of charging station placement and charging pile assignment," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 10, pp. 6654–6659, 2020.

[33] X. Wang *et al.*, "Coordinated planning strategy for electric vehicle charging stations and coupled traffic-electric networks," *IEEE Trans. Power Syst.*, vol. 34, no. 1, pp. 268–279, 2018.

[34] M. A. Mejia, L. H. Macedo, G. Muñoz-Delgado, J. Contreras, and A. Padilha-Feltrin, "Multistage planning model for active distribution systems and electric vehicle charging stations considering voltage-dependent load behavior," *IEEE Trans. Smart Grid*, vol. 13, no. 2, pp. 1383–1397, 2021.

[35] A. Ehsan and Q. Yang, "Active distribution system reinforcement planning with ev charging stations—part i: Uncertainty modeling and problem formulation," *IEEE Transactions on Sustainable Energy*, vol. 11, no. 2, pp. 970–978, 2020.

[36] J. Jordán, J. Palanca, P. Martí, and V. Julian, "Electric vehicle charging stations emplacement using genetic algorithms and agent-based simulation," *Expert Systems with Applications*, vol. 197, p. 116739, 2022.

[37] M. S. K. Reddy and K. Selvajyothi, "Optimal placement of electric vehicle charging station for unbalanced radial distribution systems," *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, pp. 1–15, 2020.

[38] J. Li, Z. Liu, and X. Wang, "Public charging station location determination for electric ride-hailing vehicles based on an improved genetic algorithm," *Sustainable Cities and Society*, vol. 74, p. 103181, 2021.

[39] E. Hadian, H. Akbari, M. Farzinfar, and S. Saeed, "Optimal allocation of electric vehicle charging stations with adopted smart charging/discharging schedule," *IEEE Access*, vol. 8, pp. 196 908–196 919, 2020.

[40] C. D. Korkas, S. Baldi, S. Yuan, and E. B. Kosmatopoulos, "An adaptive learning-based approach for nearly optimal dynamic charging of electric vehicle fleets," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 7, pp. 2066–2075, 2017.

[41] Z. Zhao, C. K. Lee, and J. Huo, "Ev charging station deployment on coupled transportation and power distribution networks via reinforcement learning," *Energy*, vol. 267, p. 126555, 2023.

[42] C. S. Lai, D. Chen, J. Zhang, X. Zhang, X. Xu, G. A. Taylor, and L. L. Lai, "Profit maximization for large-scale energy storage systems to enable fast ev charging infrastructure in distribution networks," *Energy*, vol. 259, p. 124852, 2022.

[43] X. Chen, H. Wang, F. Wu, Y. Wu, M. C. González, and J. Zhang, "Multimicrogrid load balancing through ev charging networks," *IEEE Internet of Things Journal*, vol. 9, no. 7, pp. 5019–5026, 2022.

[44] R. Atat, M. Ismail, E. Serpedin, and T. Overbye, "Dynamic joint allocation of ev charging stations and dgs in spatio-temporal expanding grids," *IEEE Access*, vol. 8, pp. 7280–7294, 2020.

[45] M. F. Shaaban *et al.*, "Joint planning of smart ev charging stations and dgs in eco-friendly remote hybrid microgrids," *IEEE Trans. Smart Grid*, vol. 10, no. 5, pp. 5819–5830, 2019.

[46] M. Bilal, M. Rizwan, I. Alsaidan, and F. M. Almasoudi, "Ai-based approach for optimal placement of evcs and dg with reliability analysis," *IEEE Access*, vol. 9, pp. 154 204–154 224, 2021.

[47] T. Zhang, X. Chen, B. Wu, M. Dedeoglu, J. Zhang, and L. Trajkovic, "Stochastic modeling and analysis of public electric vehicle fleet charging station operations," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9252–9265, 2022.

[48] "ercot 2020," Nov 2020. [Online]. Available: https://www.ercot.com/mktinfo/loadprofile/alp

[49] A. Takiddin, R. Atat, M. Ismail, O. Boyaci, K. R. Davis, and E. Serpedin, "Generalized graph neural network-based detection of false data injection attacks in smart grids," *IEEE Transactions on Emerging Topics in Computational Intelligence*, pp. 1–13, Jan. 2023, (Early Access).

[50] K. Chen *et al.*, "Fault location in power distribution systems via deep graph convolutional networks," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 1, pp. 119–131, 2019.

[51] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[52] L. Sankar, S. R. Rajagopalan, S. Mohajer, and H. V. Poor, "Smart meter privacy: A theoretical framework," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 837–846, 2012.

[53] A. B. Birchfield, T. Xu, K. M. Gegner, K. S. Shetye, and T. J. Overbye, "Grid structural characteristics as validation criteria for synthetic networks," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 3258–3265, 2017.

[54] "Tmas 2020," Nov 2021. [Online]. Available: https://catalog.data.gov/dataset/tmas-2020

[55] M. Chang, S. Bae, G. Cha, and J. Yoo, "Aggregated electric vehicle fast-charging power demand analysis and forecast based on lstm neural network," *Sustainability*, vol. 13, no. 24, p. 13783, 2021.