TOWARDS CARBON-INTELLIGENT COMPUTING: CARBON FORECASTING AND ACCOUNTING MODELS FOR COMPUTER AND LARGE AI SYSTEMS

XIAOYANG ZHANG

PhD

The Hong Kong Polytechnic University

2026

The Hong Kong Polytechnic University
Department of Computing

Towards Carbon-Intelligent Computing: Carbon Forecasting and Accounting Models for Computer and Large AI Systems

XIAOYANG ZHANG

A thesis submitted in partial fulfillment of the requirements for
the degree of Doctor of Philosophy
August 2025

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgment has been made in the text.

Signature: ____________________

Name of Student: XIAOYANG ZHANG

Abstract

The rapid expansion of artificial intelligence (AI) and large-scale computing systems has led to a significant increase in their carbon footprint, which includes both operational carbon (from electricity consumption) and embodied carbon (from hardware manufacturing). To enable the carbon reduction of carbon-aware systems, three critical challenges remain understudied: (1) accurate carbon intensity forecasting of electricity in complex cross-border power grids, (2) generalizable carbon intensity forecasting in carbon data-scarce regions, and (3) granular embodied carbon accounting of computer systems that captures spatial-temporal dynamics and carbon attribution methods.

First, existing carbon intensity forecasting models are designed for regional or one-hop neighboring grids. They fail to capture the complex spatial and temporal dependencies in cross-border power grids, where electricity exchanges involve multiple countries and dynamic carbon flows. This leads to inaccurate forecasts, which in turn limit the effectiveness of carbon-aware scheduling in distributed computing systems. To solve this problem, we propose cross-border carbon flows, and such flows form a carbon network. The challenge is to learn the complex spatial and temporal dependencies that are involved. We propose a new CFCG model based on Graph Neural Networks (GNN). Our model has three features: (1) the carbon intensity variation shows periodic patterns in mul-tiple levels of granularity, e.g., hourly, daily, and weekly. To capture the patterns across various granularities, we develop a multi-periodic pattern

encoding scheme to transform the original carbon intensity sequence data into data with different periodic granularity settings (e.g., hourly, daily, weekly); (2) Cross-border power grids have carbon flows with spatial dependencies. Thus, in our CFCG model, we develop and integrate a GNN-based submodel (i.e., GNN layers) to learn the spatial dependencies; (3) Carbon flows have time dependencies. Thus, in our CFCG model, we develop and integrate an LSTM-based submodel (i.e., LSTM layers) to uncover the temporal dependencies. We further evaluate our model through two case studies where we use the model to support Large Language Model training and inferences to reduce their carbon footprint and show the potential carbon footprint reduction of up to 90.89%.

Second, existing carbon intensity forecasting methods focus on data-rich regions, yet most areas lack sufficient carbon data due to carbon intensity's inherent measurement limitations (e.g., reliance on indirect estimation rather than direct sensor metering) and dependence on upstream entities like grid operators for data disclosure. We propose DGCFM, a Dual Graph empowered Carbon-domain Foundation Model, enabling cross-regional general carbon intensity forecasting, especially for data-scarce regions. DGCFM integrates a pre-trained Time Series Foundation Model (TSFM) to capture temporal patterns under data constraints, empowered by metadata-driven carbon hypergraph fine-tuning and a heterogeneous spatiotemporal graph to capture spatial dependency in carbon networks. Evaluated on 102 real-world datasets, DGCFM achieves 20.04% average accuracy improvement in low-data scenarios.

Third, the embodied carbon of computer systems is the carbon emissions in the manufacturing process of hardware, which dominate the overall carbon footprint. Existing studies derive the embodied carbon through life cycle analysis (LCA) reports. Current LCA reports only provide the carbon emission of a product class, e.g. 28nm CPU, whereas a product instance can be made in various regions and time periods. Carbon emissions depend on the electricity generation process, which has spatial-temporal dynamics. Therefore, the embodied carbon of a product instance can differ from its

product class. Additionally, different carbon attribution methods (e.g., location-based and market-based) can affect the carbon emissions of electricity, thus further affecting the embodied carbon of products. In this paper, we present new Spatial-Temporal Embodied Carbon (STEC) accounting models with dual attribution methods. We observe significant differences between STEC and current models, e.g., for 7nm CPU the difference is 13.69%. We further examine the impact of STEC models on existing embodied carbon accounting schemes on computer applications, such as Large Language Model (LLM) training and LLM inference. We observe that using STEC results in much greater differences in the embodied carbon of certain applications as compared to others (e.g., 32.26% vs. 6.35%).

This thesis provides comprehensive models to support carbon-intelligent computing, facilitating both operational and embodied carbon optimization across the lifecycle of computers and AI systems.

Publications Arising from the Thesis

1. Zhang, Xiaoyang, Yijie Yang, and Dan Wang, "Spatial-Temporal Embodied Carbon Models with Dual Carbon Attribution for Embodied Carbon Accounting of Computer Systems", Accepted by *IEEE Transactions on Computers* (2025)

2. Zhang, Xiaoyang, Fang He, Yang Deng, and Dan Wang, "Unveiling the Uncertainty in Embodied and Operational Carbon of Large AI Models through a Probabilistic Carbon Accounting Model", accepted by *Advances in neural information processing systems (NeurIPS)* (2025)

3. Zhang, Xiaoyang, Yijie Yang, and Dan Wang, "Spatial-temporal embodied carbon models for the embodied carbon accounting of computer systems" in *Proceedings of the 15th ACM International Conference on Future and Sustainable Energy Systems (e-Energy)* (2024)

4. Zhang, Xiaoyang, and Dan Wang, "A GNN-based day ahead carbon intensity forecasting model for cross-border power grids", in *Proceedings of the 14th ACM International Conference on Future Energy Systems (e-Energy)* (2023)

5. Zhang, Xiaoyang, Yucheng Bao, Taiqi Zhou and Dan Wang, "Demo: CarbonReveal: Embodied Carbon Accounting with Retrieval-Augmented LLM for Com-

puter Systems” in *Proceedings of the 11th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys)* (2024)

6. Zhang, Xiaoyang, Taiqi Zhou, Fang He, Yang Deng, and Dan Wang, “Carbon Intensity Forecasting via Time Series Foundation Model”, Under Review* by *The Acm Web Conference* (2026)

   * For those papers under review, titles have been modified to maintain anonymity.

Acknowledgments

I am immensely grateful to my supervisor, Dr. Dan Wang, for his exceptional guidance and unwavering support throughout my doctoral journey. His profound knowledge and vision were instrumental in shaping the direction of this research. Mr. Wang consistently encouraged my curiosity for cutting-edge technologies and provided invaluable advice at every step of the process. His critical comments and persistent encouragement have greatly enhanced the quality of this work.

My sincere thanks go to Dr. He Fang, Dr. Deng Yang, Dr. Lu Rui, and Dr. Shi Siping for their insightful suggestions and generous assistance. Their expertise not only enriched this dissertation but also broadened my perspective toward academic research and beyond.

I am also deeply thankful to my group mates, Mr. Bihai Zhang, Mr. Tao Ling, Mr. Shuntao Zhu, Mr. Yanting Liu, Mr. Jiaqi Fan, Mr. Boan Liu, Miss Yijie Yang, and Mr. Rui Liang, for the stimulating discussions and collaborative spirit that characterized our daily interactions. Their companionship and intellectual camaraderie made my Ph.D. experience both productive and enjoyable.

To all my friends, thank you for your endless support and encouragement. Your presence has been a constant source of joy and motivation throughout these years.

Lastly, I owe my deepest gratitude to my parents and my family. Their unconditional love, understanding, and belief in me have been the foundation upon which this

accomplishment was built. Without them, none of this would have been possible.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview and Thesis Questions

The rapid growth of artificial intelligence (AI) and large-scale computing systems has led to a significant increase in their carbon footprint, which comprises both operational carbon (from electricity consumption) and embodied carbon (from hardware manufacturing). While there is growing awareness and effort to reduce operational carbon through carbon-aware scheduling and renewable energy usage, three critical challenges remain understudied:

Carbon intensity forecasting of electricity in complex grids: Existing carbon intensity forecasting methods are designed for regional or one-hop neighboring grids. They fail to capture the complex spatial and temporal dependencies in cross-border power grids, where electricity exchanges involve multiple countries and dynamic carbon flows. This leads to inaccurate forecasts, which in turn limit the effectiveness of carbon-aware scheduling in distributed computing systems.

Carbon intensity forecasting of electricity in carbon data-scarce regions: Existing carbon intensity forecasting methods focus on data-rich regions, yet most areas lack sufficient carbon data due to carbon intensity's inherent measurement limitations (e.g., reliance on indirect estimation rather than direct sensor metering) and dependence on upstream entities like grid operators for data disclosure. This data scarcity creates a critical performance gap: existing models exhibit significantly degraded accuracy when applied to regions with insufficient historical data, thereby hindering global decarbonization efforts through computing solutions.

Granular embodied carbon accounting: Current embodied carbon accounting models, such as ACT, rely on static, averaged emission factors and do not account for spatial-temporal variations in carbon intensity or different carbon attribution methods (location-based vs. market-based). This results in inaccurate carbon footprints accounting for computer hardware, especially for products manufactured in different regions and times.

To address these gaps, this thesis poses the following overarching research question:

How can we develop accurate and generalizable models for carbon intensity forecasting and embodied carbon accounting that incorporate spatial-temporal dynamics and multi-regional carbon flow dependencies to support carbon-aware AI and computing systems?

This question is further decomposed into three sub-questions:

- How can we accurately forecast the carbon intensity of electricity in cross-border power grids by modeling multi-hop spatial dependencies and multi-periodic temporal patterns? (Addressed in Chapter 3)
- How can we generalize carbon intensity forecasting to data-scarce regions using time series foundation models? (Addressed in Chapter 4)
- How can we account for embodied carbon in computer systems with spatial-temporal granularity and dual carbon attribution methods? (Addressed in Chapter 5)

By answering these questions, this thesis aims to provide a comprehensive and practical toolkit for carbon-aware computing, enabling both operational and embodied carbon optimization across the lifecycle of AI and computing systems

1.2 Background

1.2.1 Carbon Emission and Carbon Intensity Forecasting

In electricity generation, *carbon emissions* result from burning fuels in power plants. There are two types of carbon emissions [72]: (1) direct emissions, i.e., the operational carbon emissions that occur when the fuel is converted into electricity and (2) life-cycle emissions, which can be determined using methods of assessing life cycles [113, 8, 47].

The *carbon emission factor* (in g/kWh) is the quantity of carbon emitted per unit of electricity produced by a specific energy source (e.g., coal, wind, solar). Calculating the carbon emission factor is complex work and is beyond the scope of this paper. We take carbon emission factors as inputs. Table 1.1 shows the carbon emission factors for both direct emissions and life-cycle emissions [72]. The *carbon intensity* of a power grid is the carbon emission rate (in g/kWh) of this power grid, i.e., the total amount of carbon emitted (Gram) as against the electricity generated (Kilowatt-Hour). This can be calculated by the energy sources used in this power grid and the carbon emission factor of the energy sources. As compared to the sheer amount of carbon emissions, carbon intensity is more commonly used as an index in making optimization decisions. Table 1.2 provides the annual average carbon intensity of electricity for six major IC production regions in 2022.

Carbon intensity forecasting aims to predict the carbon intensity of the electricity supplied by a power grid for a period of time in the future. It is useful for demand-side optimization, e.g., demand-side systems can switch their workloads to use electricity

Table 1.1: Carbon emission factors (g/kWh) for different energy source

Emission factors	Oil	Coal	Natural gas	Nuclear	Wind	Solar	Hydro	Geothermal	Biomass	Other
Direct emissions	406	760	370	0	0	0	0	0	0	575
Life-cycle emissions	650	820	490	12	11	45	24	38	230	700

Table 1.2: Carbon intensity of electricity in six major IC production regions in 2022.

Countriy/Region	Carbon intensity of electricity (g/kWh) (Date Sources: Our World in Data [50])
Taiwan	561
China	531
South Korea	436
United States	367
Italy	372
Ireland	346

at a time that is "greener." For example, algorithms have been developed for smart home systems to enable the workloads of household appliances to be scheduled based on carbon intensity forecasting to reduce their carbon emissions [90]. Cloud computing schedules have been developed to distribute the workloads according to the carbon intensity at different times and places [121, 89]. In [7], deep learning model training schedules are based on carbon intensity forecasting.

1.2.2 Cross-boarder Power Grids

Cross-border power grids [40, 123, 85] link two or more power grid systems, and allow electricity to be transmitted over larger areas across borders. There are three prominent advantages to cross-border power girds.

First, cross-border power grids help market participants to benefit from economies of scale on both the supply and demand sides. Larger generators can be operated to serve more consumers.

Second, cross-border power grids are diverse in terms of both supply and demand. This improves the security of the grids. For example, cross-border power grids can meet the peak demands with relatively fewer resources [100, 84].

Third, and increasingly important, is the environmental benefits of cross-border power grids. Cross-border power grids can integrate more variable renewables. On the one hand, they allow operators to leverage weather patterns across larger spaces. On the other hand, cross-border grids make it easier to balance the local variable renewable energy, as it can access greater supplies as well as additional pools of demands [11, 17].

Cross-border power grids are operated by joint operators. For example, the European Network of Transmission System Operators for Energy (ENTSO-E) operates the European cross-border power grids [117]. It was created to enhance cooperation between national power grid operators in Europe. The European cross-border power grids have expanded dramatically during over past years. According to the latest ENTSO-E statistic report [30], the volume of electricity that was exchanged was approximately 467TWh, comprising 13% of the net electricity that was generated. In 2022, 12 new borders received their first electric connectivity, joining 80 European borders [31]. Nine of the ENTSO-E member countries imported more than 50% of their total yearly electricity from their neighbors. Nearly 50% percent of the ENTSO-E member states imports and exports more than 20% of their domestic electricity generation from the cross-border power grid [45].

The intrinsic difference of a cross-border grid brings about as compared to a regional grid is that the carbon intensity borne by the grids involved a cross-border grid is calculated to reflect their state-level carbon intensity. Therefore, carbon intensity forecasting should carry this information.

1.2.3 Embodied Carbon

The carbon footprint of computer systems primarily comes from two sources: embodied carbon and operational carbon. Embodied carbon refers to the emissions generated during the manufacturing process, while operational carbon pertains to emissions produced during the system's use, primarily due to electricity consumption. To decarbonize a system, it is crucial to reduce both its embodied emissions and operational emissions, which together are known as lifecycle emissions. With advancements in the efficiency of ICT, the primary source of ICT's carbon emissions is shifting from operational to embodied carbon. For instance, in 2019, Apple reported that hardware manufacturing constituted 74% of its overall carbon footprint, whereas the operational use of all its devices accounted for 19%. Similarly, at the individual system level, for an iPhone 11, manufacturing and operational use contribute to 79% and 17% of the carbon emissions, respectively, with the remaining 4% attributed to product transport and recycling [39].

1.2.4 Carbon Attribution Methods

Companies are increasingly setting their targets to become carbon-neutral by reducing indirect greenhouse gas (GHG) emissions resulting from the purchase of electricity (scope 2 emissions in GHG protocol [3]). To achieve the targets, companies are investing in green energy (e.g., solar, wind, etc) through Power Purchase Agreements (PPAs). PPAs are long-term contracts between consumers and electricity producers, allowing consumers to claim renewable energy credits and reduce their electricity-related emissions [70]. According to GHG protocol [3], emissions can be attributed to consumers based on how renewable energy and electricity source mixes are allocated.

Location-based Method assumes all electricity consumers within a specific geographical area receive the same electricity mix in the location-based method. Renewable energy is attributed to the power grid. This method does not account for individual green energy investments by specific consumers; instead, all consumers benefit from any renewable energy investments made by any single consumer.

Market-based Method allows consumers who invest in green energy to claim carbon credits for their purchased electricity, thereby reducing their emissions, regardless of whether the green energy invested is physically delivered through transmission lines. On the other hand, consumers without PPAs account for their emissions based on the residual electricity mix in power grids.

1.3 Thesis Organization

The rest of this thesis is organized as follows.

- **Chapter 2** This chapter reviews the related work about carbon intensity forecasting and carbon footprint accounting for computer systems;
- **Chapter 3** This chapter presents CFCG, which is a GNN-based **C**arbon intensity **F**orecasting model for **C**ross-border power **G**rids;
- **Chapter 4** This chapter proposes DGCFM, which is a **D**ual **G**raph (i.e., a heterogeneous spatial-temporal carbon graph and a metadata-assisted carbon hypergraph) empowered **C**arbon-domain time series **F**oundation **M**odel;
- **Chapter 5** This chapter proposes STEC, which is **S**patial-**T**emporal **E**mbodied **C**arbon Models with Dual Carbon Attribution for Embodied Carbon Accounting of Computer Systems;
- **Chapter 6** This chapter concludes the thesis.

Chapter 2

Literature Review

2.1 Carbon Intensity Forecasting

Estimating the Carbon intensity of a corporate entity has become an important problem with the increasing concerns over carbon emissions. Carbon intensity cannot be directly measured, but there are many ways to estimate carbon intensity. Below is a categorization of the related work. A summary is given in Table 2.1.

The related work can be grouped under the headings of *generation-based* carbon intensity estimation and *consumption-based* carbon intensity estimation. This follows the Greenhouse Gas (GHG) protocol [103] where the carbon emissions of a corporate entity are distinguished into (1) Scope 1, the emissions associated with the electricity

Table 2.1: Research summary on Carbon Intensity Estimation

Generation-based Carbon Intensity Estimation	**Consumption-based Carbon Intensity Estimation**		
[5] [4] [27]	Carbon Intensity Accounting	[87][93] [110]	
	Carbon Intensity Forecasting	Regional grids	[15] [68][71][72]
		One-hop neighbor grids	[58][90]
		Cross-border grids	CFCG

generation of a corporate entity, e.g., burning fuels and (2) Scope 2, the emissions associated with the electricity consumed by a corporate entity [44]. [1] Generation-based estimation focuses on the calibration of carbon emission factors [82]. In particular, [113] calibrates the factors at an individual country level.

Our paper falls under the heading of consumption-based carbon intensity estimation. Consumption-based estimation can differ from generation-based estimation since a corporate entity, as an electricity consumer, can consume electricity generated from different power grids, e.g., through delivery by cross-border grids.

Research into consumption-based estimation can be further divided into *carbon intensity accounting* [110], i.e., to estimate the carbon intensity at a particular point in history or at the current time; and *carbon intensity forecasting*, i.e., to estimate the carbon intensity at a future time. Intrinsically, carbon intensity accounting is about trying to accurately estimate the ground truth; thus, it is useful for guiding financial investments, informing policy-making decisions, and measuring compliance with regulations [75]. The carbon intensity accounting of an individual regional grid is equivalent to generation-based carbon intensity estimation. To measure both the carbon generated in a regional grid and the carbon injected from networked grids, three recent schemes were developed [87, 110, 93]. All of these schemes calculate the injected carbon based on a proportional sharing principle, i.e., an equal distribution of the inflows. The difference is that (1) two schemes (direct coupling schemes) [87, 110] assume that each grid in the grid network is coupled with all other grids, with a difference in a weighting factor; and (2) one scheme (an aggregate coupling scheme) [93] assumes that the grid will firstly serve its own country, with the residual electricity available for use by other countries.

Our paper is about carbon intensity forecasting. As discussed, carbon intensity fore-

[1] In GHG, there is a Scope 3, which includes all other emissions that occur in the upstream and downstream activities of a corporate entity. It is less related to this paper, and we do not discuss this category.

casting involves trying to accurately estimate carbon intensity at a future time; thus is useful for making decisions about demand-side optimization. Carbon intensity forecasting requires historical data on carbon intensity data and takes carbon intensity accounting as a building block for the ground truths. The technical difference between carbon intensity accounting and carbon intensity forecasting is that accounting is designed to calculate the ground truth at a point in history (or current) time; thus, the calculation uses the data in the *same* time slot. For forecasting, learning methods should be developed to learn the correlation within historical data across various time granularities.

There are carbon intensity forecasting schemes. Lowry, G. [68] proposed an algorithm to forecast the day-ahead carbon intensity of the power grids of the UK. Additional information was used to improve the forecasting results. For example, weather data were leveraged in [58] to forecast the carbon intensity of the power grids in Denmark. Recently, a scheme [72] developed a deep neural network (DNN) model for each energy source, e.g., coal, oil, solar, etc., and the carbon intensity forecasting of a regional grid is calculated by the average carbon intensity of all energy sources. Another study [71] focuses on multi-day carbon intensity forecasting, developing a hierarchical DNN model. As exchanges of electricity take place among cross-border grids, the carbon intensity of one-hop neighboring grids was considered in [90, 58] to improve the forecasting results. More specifically, the carbon intensity accounting takes one-hop neighbor grids into account when constructing historical and current carbon intensity data. Then, an LSTM model [90] or a hybrid model with linear regression, splines, and ARIMA [58] is used for forecasting. To the best of our knowledge, no existing study takes into consideration the electricity that is exchanged at the level of cross-border power grids.

Besides, all existing methods share a crippling dependency on abundant local training data, an assumption fundamentally disconnected from reality. Most regions, especially developing regions, lack sufficient historical carbon data due to (1) non-uniform measurement protocols across grid operators, (2) proprietary restrictions on commercial energy data, and (3) delayed reporting. Models trained on data-rich regions may have great performance degradation when applied to other regions, due to differing energy infrastructures and market mechanisms. This paper fills in these gaps.

We note that several commercial companies provide carbon intensity estimation services, e.g., ElectricityMap [109], Watttime [119]. Unfortunately, their models and data are not publically available.

2.2 Carbon Footprint Accounting

The carbon footprint consists of operational carbon and embodied carbon. Operational carbon refers to the carbon emissions caused by the consumption of electricity during system operation; embodied carbon refers to the carbon emissions caused by the manufacturing process of the system. Embodied carbon accounting in computing systems is an emerging and crucial topic as global carbon reduction efforts intensify. Most current methods for embodied carbon accounting are not specifically designed for computing systems. For example, LCA tools, developed by environmental scientists, aim to quantify the carbon of various products. EIO-LCA [120] estimates carbon emissions based on the economic cost of electronics, typically converting component costs to carbon emissions using industry-wide scale factors. Besides economic cost-based LCAs, another category of LCA tools employs database-driven approaches. Sphera [105] uses a system's bill of materials to estimate its carbon footprint. However, the coarse granularity of these databases limits the accuracy of accounting. The Architectural CO2 Tool (ACT) Model [39] is the first and the state-of-the-art embodied carbon accounting model for computer systems. ACT estimates the embodied carbon of three kinds of fundamental computer hardware: processors, memory, and storage through more detailed manufacturing process modeling and more granular data adoption. However, ACT neglects the effect of different carbon attribution methods and the spatial-temporal factors. This paper fills in this gap.

There are also many works that focus on estimating or optimizing the carbon footprint of computer applications, such as LLM [33, 23], edge computing [106, 69], federated learning [14, 86], and cloud computing [42, 61], which will benefit from more accurate embodied carbon accounting in this paper.

The current methodologies for carbon footprint estimation of AI models predominantly concentrate on operational carbon, which is calculated as the product of electricity consumed and carbon intensity. Most studies predominantly focus on tracking or estimating the electricity consumed. A subset of studies has developed software-integrated instrumentation to monitor real-time CPU/GPU power utilization during inference or training phases [43, 7, 18, 67], while alternative approaches derive energy estimates through the parameters of and hardware (e.g., thermal design power) [56, 57]. However, these methodologies systematically neglect the embodied carbon accounting associated with AI hardware infrastructure, which is non-negligible in the carbon footprint of large AI models, as power grids decarbonize and data centers increasingly adopt zero-carbon energy sources.

Regarding embodied carbon modeling, existing frameworks such as SustainableAI [125] estimated the embodied carbon of AI models through manufacturer-reported emission factors for hardware components, which is an average value. LLMCarbon [33] adopt a deterministic parametric model to estimate the embodied carbon for processors and average values for storage and memory in ESG reports. However, these approaches lead to oversimplified modeling constrained by two critical limitations: (1) fail to capture the effect of temporal and spatial specificity in hardware instance (e.g., 28nm CPU made in winter in China vs. 28nm CPU made in summer in the US); (2) fail to capture the effect of heterogeneous manufacturing configurations within device class (e.g., yield and fabrication efficiency improvement for certain processors). These characteristics lead to uncertainty in carbon estimates in AI models, but current AI carbon models fail to capture this uncertainty. This paper fills in this gap.

Chapter 3

CFCG: A GNN-based Day Ahead Carbon Intensity Forecasting Model for Cross-Border Power Grids

3.1 Introduction

Global warming and carbon emissions have gained more and more attention lately in many research communities, e.g., photovoltaic power [80, 26], economics [55, 127], sustainable industry [1, 81], carbon capture [135], and green building [129, 19]. Notably, carbon emissions by the power sector increased by 700 Mt in 2021, accounting for 46% of the global rise [49]. Consequently, there is an increasing imperative for electricity producers and consumers to reduce their carbon footprint.

Forecasting the carbon intensity of the electricity supplied by power grids is critical in the optimization of demand-side consumers. For example, the computing community's interest in reducing the carbon footprint of information systems is rising (e.g., the carbon footprint of Large Language Models (LLMs) [60], Federated Learning [122], Cloud Computing [121], Edge Computing [106], etc.).

Recently, *cross-border power grids* have emerged, which enable the transmission of electricity across the transmission systems of different countries [88]. Notably, the total volume of cross-border electricity exchanged accounted for 13% of the annual electricity generated in Europe.

Current studies on carbon intensity forecasting primarily focus on individual regional power grids. Forecasting models, e.g., Long Short Term Memory (LSTM), are trained using the historical carbon intensity data of a local regional grid, or in some recent studies, the carbon intensity of the one-hop neighboring grids are considered. In the context of cross-border power grids, when a consumer uses the electricity of a local grid, the carbon intensity of a local grid depends not only on the electricity generated by the local grid but also on the electricity exchanged with cross-border grids. For instance, Switzerland, despite having low carbon intensity in its local grids, imports electricity from neighboring countries such as Germany, Austria, Italy, and France, thereby incorporating carbon emissions from the supplied electricity to consumers.

Clearly, neglecting the cross-border electricity exchanges can lead to significant forecasting errors. We also observe that even if we estimate the carbon intensity using cross-border grids and subsequently conduct forecasting based on LSTM, non-trivial errors can still occur. Intrinsically, the historical trends in cross-border grids and the electricity that is exchanged in accordance with geographical topologies are dynamic and correlated across diverse granularities. It is, therefore, challenging to capture such correlations. Additionally, the electricity data resolution in the cross-border power grids is hourly, which may match the optimization operation frequency of downstream applications or not. High-resolution carbon data can enable precise optimization for electricity consumers but is challenging to obtain [65].

In this paper, we present a holistic study by formulating a new Carbon Intensity Forecasting for Cross-border Power Grids (CFCG) problem. In this problem, we propose cross-border carbon flows, and such flows form a carbon network. The challenge is to learn the complex spatial and temporal dependencies that are involved. We propose a new CFCG model based on Graph Neural Networks (GNN) and LSTM. Our model has three features: (1) the carbon intensity variation shows periodic patterns in multiple levels of granularity, e.g., hourly, daily, and weekly. To capture the patterns across various granularities, we develop a multi-periodic pattern encoding scheme to transform the original carbon intensity sequence data into data with different periodic granularity settings (e.g., hourly, daily, weekly); (2) Cross-border power grids have carbon flows with spatial dependencies. Thus, in our CFCG model, we develop and integrate a GNN-based submodel (i.e., GNN layers) to learn the spatial dependencies; (3) Carbon flows have time dependencies. Thus, in our CFCG model, we develop and integrate an LSTM-based submodel (i.e., LSTM layers) to uncover the temporal dependencies. Finally, to fuse different granularities, we generate the forecasting results using a Hadamard product. (4) The resolution of forecasting carbon intensity may match the optimization operation frequency of downstream applications or not. We develop the pluggable carbon intensity super-resolution neutral network layers (CISPNN) to achieve higher carbon intensity output.

We evaluate the CFCG model using the real-world dataset encompassing the cross-border power grids of 28 European countries. Our evaluations demonstrate that CFCG can achieve a notable improvement of 26.46% and 20.34% compared to two state-of-the-art specialized carbon intensity forecasting schemes using regional grids or one-hop neighbor grids, respectively. Additionally, we present two case studies where CCFG supports applications (e.g., LLMs training and LLMs inferences) to reduce their carbon footprint. The maximum carbon reduction potential can be 90.89% (LLMs training) and 89.03% (LLMs inferences).

The contributions of the paper can be summarized as follows:

- We present a new study of carbon intensity forecasting in the context of cross-border power grids. We carefully analyze the literature and formulate a new carbon intensity forecasting problem, CFCG.

- We propose a CFCG model based on GNN and LSTM, where CFCG can uncover both spatial and temporal dependencies. We design a node-aware embedding mechanism to embed physical rules of a carbon network and develop the CISPNN to enable the selective resolution forecasting data output for better matching the operation frequency of downstream applications.

- We present an evaluation of CFCG using real-world data from the cross-border power grids of Europe, involving 28 countries. Our evaluation shows that CFCG outperforms state-of-the-art schemes and that for certain countries, the improvement can be significant.

- We present two case studies on Large Language Models (LLMs) training and LLMs inference to show their maximum potential for reducing carbon footprint supported by the CFCG model.

3.2 Problem Statement

3.2.1 Carbon Network Modeling

Let i and j present two power grids. The *electricity flow* $f^e_{ij}(t)$ denotes the total amount of electricity transmitted from grid i to j during a specific period of time starting at t. In practice, this time period is typically one hour. Let $E_i(t)$ denote the electricity generated by grid i at time t. Let $\mathcal{S}$ be the set of energy sources (e.g., Solar, Coal, etc) and $S = |\mathcal{S}|$. Let $E^k_i(t)$ represent the electricity generated by energy source k at grid i. It follows that $E_i(t) = \sum_{k \in \mathcal{S}} E^k_i(t)$.

As previously mentioned, different sources are utilized for electricity generation, and each type of source emits a distinct amount of carbon. Let ef^k denote the *carbon emission factor* of a source k. In this paper, we take this carbon emission factor as a constant for a specific source [72]. For instance, the carbon emission factor of Coal is 760g/kWh, while the carbon emission factor of Wind is 0. The most common energy sources used for electricity generation and their respective carbon emission factors are presented in Table 1.1.

At time t, a grid i generates electricity using various sources, as the quantity of certain sources fluctuates at different times (e.g., wind, hydro, etc.). This variability leads to dynamic carbon emissions at different times. Let *carbon intensity* $ci(t)$ be the ratio of the total carbon emissions to the total electricity generated. Specifically,

$$ci(t) = \frac{\sum_{k \in \mathcal{S}} ef^k \times E^k(t)}{E(t)} \tag{3.1}$$

It is important to note that carbon intensity cannot be directly measured and is calculated using Eq. 3.1.

In practice, two power grids can exchange electricity when they are interconnected by transmission lines. These interconnected grids are referred to as *neighboring grids*. Note that when a grid i imports electricity flow f^e_{ij} from grid j, grid i also bears the carbon released by this flow. We call this *carbon flow*. Intuitively, we can think that the carbon emission "flows" from grid i to grid j. Let carbon flow $f^c_{ij}(t)$ represent the total amount of carbon released by the electricity flow $f^e_{ij}(t)$ at the time t. Specifically, this denotes the amount of carbon associated with the electricity generated by power grid i and transmitted to power grid j at time t. Mathematically, $f^c_{ij}(t) = f^e_{ij}(t) \times ci_i(t)$.

We formally present a carbon network modeling as follows. For clarity, we summarize key notations as Table 3.1 shows.

Table 3.1: Notations and Descriptions.

Notations	Descriptions
$\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$	Directed graph representing carbon network
$ci_i(t)$	Carbon intensity of power grid i at time t
$\boldsymbol{ci}(t) = \{ci_0(t), ..., ci_i(t)\}$	Carbon intensity of all power grids at time t
$\boldsymbol{ci}_i = \{ci_i(0), ..., ci_i(t)\}$	Carbon intensity of power grid i from time 0 to t
$\boldsymbol{CI} = \{\boldsymbol{ci}(0), ..., \boldsymbol{ci}(t)\}$	Carbon intensity sequence of all power grids
$f^e_{ij}(t)$	Electricity flow from grid i to grid j at time t
$\boldsymbol{f^e}(t) = \{f^e_{ij}(t)\} \in \mathbb{R}^{\mathcal{E}}$	Electricity flows of all power grids at time t
$\boldsymbol{F^e} = \{\boldsymbol{f^e}(0), ..., \boldsymbol{f^e}(t)\}$	Electricity flow sequence of all power grids

Definition 1. ***Carbon network.*** *We use a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ to model a carbon network where $\mathcal{V} = v_1, ..., v_n$ represents a set of nodes, with v_i denoting a regional grid, and $|\mathcal{V}| = n$. The set $\mathcal{E}$ represents a collection of links, where a directed link (v_i, v_j) signifies that grid v_j directly imports electricity from grid v_i. To present the connectivity among grids, we employ a binary adjacency matrix $\mathcal{A} \in \mathbb{R}^{n \times n}$*

We now introduce two features that are essential for our learning model: (1) *carbon intensity sequence*, which is a feature associated with nodes; and (2) *electricity flow sequence*, which is a feature related to links. The history carbon intensity data and electricity flow data serve as the inputs for our model.

Definition 2. ***Carbon intensity sequence.*** *We represent the carbon intensity sequence of grid i from time 0 to t as a vector $\boldsymbol{ci}_i = \{ci_i(0), ..., ci_i(t)\}$ and the carbon intensity of all grids at time t as a vector $\boldsymbol{ci}(t) = \{ci_0(t), ..., ci_i(t)\}$. The vector $\boldsymbol{ci}_i$ is a time series data. Furthermore, the carbon intensity sequence of all grids from time 0 to t is denoted as a matrix $\boldsymbol{CI} = \{\boldsymbol{ci_i}\}$.*

Definition 3. ***Electricity flow sequence.*** *Let all electricity flows of a carbon network at time t be a matrix $\boldsymbol{f^e}(t) = \{\forall i, j, f^e_{ij}(t)\} \in \mathbb{R}^{\mathcal{E}}$. The electricity flow sequence of a carbon network over a period of time t is represented as a three-dimensional matrix $\boldsymbol{F^e} = \{\boldsymbol{f^e}(0), ..., \boldsymbol{f^e}(t)\}$.*

The carbon intensity sequence is represented as a two-dimensional matrix, as it is on a node i, whereas the electricity flow sequence is denoted as a three-dimensional matrix, as it operates on a link (i, j).

3.2.2 Carbon Intensity Super Resolution Modeling

Let $\boldsymbol{ci}^l$ denote the carbon intensity sequence in a power grid with frequency f_l and length d. Let $\boldsymbol{ci}^h$ denote another carbon intensity sequence of the same grid in the same time period with a higher frequency $f_h = SRF \times f_l$, where SRF represents the super-resolution factor (positive integer).

Consequently, the high-frequency carbon intensity $\boldsymbol{ci}^h$ is a vector of length $SRF \times d$ with a denser time index. The relationship between $\boldsymbol{ci}^h$ and $\boldsymbol{ci}^l$ can be described by a downsampling model: $\boldsymbol{ci}^l = A \times \boldsymbol{ci}^l + n$, where $A \in \mathbb{R}^{d\times(SRF*d)}$ is a down-sampling matrix and n represents the additive noise. The high-frequency carbon intensity $\boldsymbol{ci}^h$ contains more detailed information about the monitored physical quantity, making it easier to analyze. However, obtaining this high-frequency data is more challenging.

The carbon intensity super resolution presents a challenge due to its under-determined nature, which means there are infinite possible high frequency carbon intensity sequences that satisfy the down-sampling model. In order to identify the most probable solution, it is necessary to introduce an additional constraint. Within the framework of Maximum a Posteriori (MAP) estimation, the ultimate estimated high frequency carbon intensity $\boldsymbol{ci}^h$ is determined by maximizing the posterior probability $p(\boldsymbol{ci}^h|\boldsymbol{ci}^l)$. Utilizing the Bayesian formula, we can get the posterior probability as follows:

$$p(\boldsymbol{c}^h|\boldsymbol{ci}^l) = \frac{p(\boldsymbol{ci}^l|\boldsymbol{ci}^h)p(\boldsymbol{ci}^h)}{p(\boldsymbol{ci}^l)} \tag{3.2}$$

where $p(\boldsymbol{ci}^l|\boldsymbol{ci}^h)$ is the likelihood according to the down-sampling model, $p(\boldsymbol{ci}^h)$ represents the prior on $\boldsymbol{ci}^h$ and $p(\boldsymbol{ci}^l)$ is a constant when $\boldsymbol{ci}^l$ is given. The estimation of the corresponding $\boldsymbol{ci}^h$ for a specific $\boldsymbol{ci}^l$ is achieved through the solution of the MAP problem:

$$\boldsymbol{ci}^{h'} = \max_h p(\boldsymbol{ci}^l|\boldsymbol{ci}^h)p(\boldsymbol{ci}^h) \tag{3.3}$$

which is equivalent as follows:

$$\boldsymbol{ci}^{h'} = \max_h logp(\boldsymbol{ci}^l|\boldsymbol{ci}^h) + logp(\boldsymbol{ci}^h) \tag{3.4}$$

The $p(\boldsymbol{ci}^l|\boldsymbol{ci}^h)$ can be solved by modelling the degradation process and $p(\boldsymbol{ci}^h)$ is obtained by solving the prior model. This prior model serves as a regularization term, imposing constraints on the solution such that the estimated $\boldsymbol{ci}^{h'}$ aligns with the prior distribution of $\boldsymbol{ci}^h$. Therefore, the effective modeling of the prior distribution of high-frequency carbon intensity data holds significant theoretical implications for the carbon intensity super resolution. Through the effective modeling of the prior distribution of high-frequency carbon intensity and the degradation process, a well-informed estimation of $\boldsymbol{ci}^h$ can be attained.

3.2.3 Problem Formulation

Problem CFCG (**C**arbon intensity **F**orecasting for **C**ross-border power **G**rids): Given a carbon network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, hourly historical carbon intensity data $\boldsymbol{CI} = \{(\boldsymbol{ci}(0), ..., \boldsymbol{ci}(t)\}$ with frequency f_l, hourly historical electricity flow data $\boldsymbol{F^e} = \{\boldsymbol{f^e}(0), ..., \boldsymbol{f^e}(t)\}$ with frequency f_l, learn a predictive function $y = f(\mathcal{G}, \boldsymbol{CI}, \boldsymbol{F^e})$, which infers the day-ahead carbon intensity sequence in cross-border power grids with a selectable frequency (f_l or higher frequency f_h).

3.3 A GNN-based CFCG Model

3.3.1 Overview

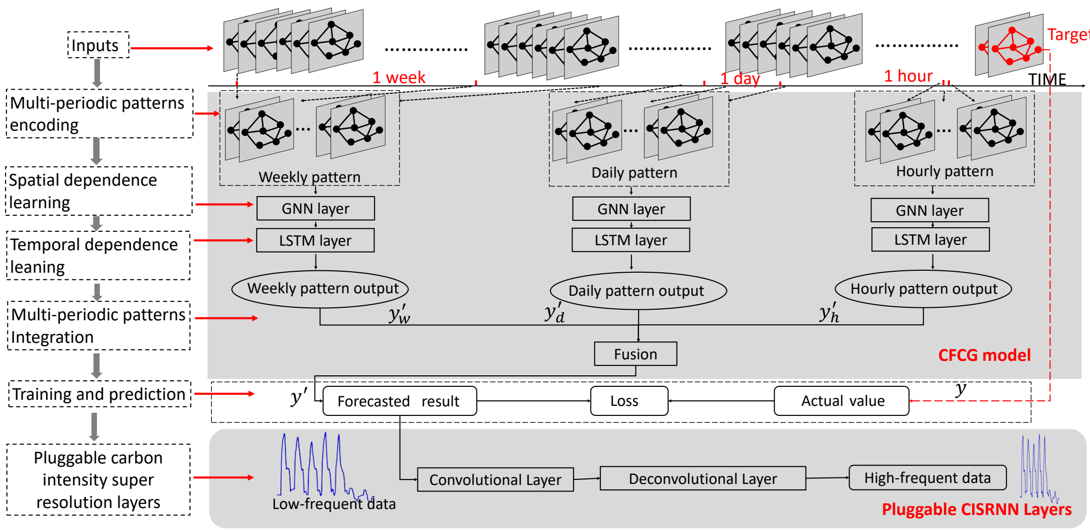


Figure 3.1: Overview. We take the history carbon intensity and electricity flow sequence as input and obtain the final prediction result through our CFCG model. The Pluggable CISPNN layers can generate high-resolution carbon intensity sequences if downstream applications need.

We have developed a new CFCG model as shown in Fig. 3.1. CFCG comprises five key components. Firstly, to capture the dynamics across various time periods, CFCG divides the sequence data into multiple periods, such as hourly, daily, and weekly. In our experiments, the granularity of these periods will be evaluated. CFCG has a multi-periodic pattern encoding component to conduct a granularity-aware encoding for the input data. Secondly, to capture spatial dependencies, CFCG employs a GNN layer with a specifically designed embedding mechanism. Thirdly, to capture temporal dependencies, CFCG leverages an LSTM layer to take the embedding results generated by the GNN layer as inputs and outputs a high-level representation. Fourthly, CFCG uses a Hadamard product to combine the outputs of the LSTM layer and then generates the forecasting result with the same frequency as input data. Finally, if the high-frequency carbon data is required by downstream applications, the pluggable CISPNN layers can take the forecasting results with the low resolution as inputs and output the carbon intensity sequence data with high resolution. To clearly demonstrate the workflow of CFCG, we also present a systemic flowchart as Figure 3.2 shows.

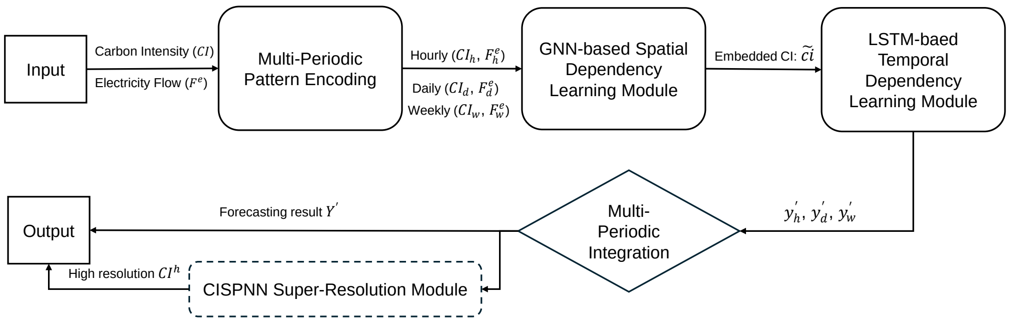


Figure 3.2: The workflow of CFCG.

3.3.2 Multi-periodic Pattern Encoding

The variation in carbon intensity exhibits multi-periodic patterns [72]. To address this, we propose to leverage multiple time granularities (e.g., hourly, daily, and weekly) to capture the temporal patterns. Similar techniques have been employed in predicting city traffic [76]. Specifically, each time granularity is considered as a period for sampling the carbon intensity data points, thereby generating a set of granularity-specific data series. We define the terms *granularity-aware carbon intensity sequence* and *granularity-aware electricity flow sequence.*

Definition 4. ***Granularity-aware Carbon Intensity Sequence.*** *Let* $\boldsymbol{CI^p} \in \mathbb{R}^{N \times T_p}$ *denote a granularity-specific carbon intensity sequence with a granularity of* p *(e.g., day), where* N *is the number of grids, and* T_p *is the length of the carbon intensity sequence under* p*.*

For instance, when $p = day$, T_p is 24 times that of $p = hour$.

$$\boldsymbol{CI^p} = \{\boldsymbol{ci}^p(0), ..., \boldsymbol{ci}^p(t-1), \boldsymbol{ci}^p(t), \boldsymbol{ci}^p(t+1), ..., \boldsymbol{ci}^p(T_p)\}(p = hour)$$

$$\boldsymbol{CI^p} = \{\boldsymbol{ci}^p(0), ..., \boldsymbol{ci}^p(t-24), \boldsymbol{ci}^p(t), \boldsymbol{ci}^p(t+24), ..., \boldsymbol{ci}^p(T_p)\}(p = day)$$

Similarly, we can define the granularity-aware electricity flow sequence as follows:

Definition 5. ***Granularity-aware Electricity Flow Sequence.*** *Let* $\boldsymbol{F^{e,p}} \in \mathbb{R}^{M \times T_p}$ *denote a granularity-specific electricity flow sequence with the time granularity of* p *(e.g., a day), where* M *is the total number of links, and* T_p *is the length of the carbon intensity sequence under* p.

In this component, the multi-periodic pattern encoding takes the granularity period p, the historical carbon intensity data, and the electricity flow data as inputs and outputs granularity-aware carbon intensity and electricity flow sequences.

3.3.3 Spatial Dependency Learning

The two main characteristics of the CFCG problem are the spatial dependency of the carbon intensity of nodes and the spatial dependency between the carbon intensity of nodes and the electricity flows of the links. To address this, We develop a GNN layer with a specifically designed embedding mechanism based on graph attention networks [114] to capture such spatial dependencies.

Algorithm 1 Node-Aware Embedding Mechanism

Input: Carbon Network $\mathcal{G}$, node features $\mathbf{ci}^p$, edge features $\mathbf{f}^{e,p}$

Output: Node embeddings $\tilde{ci}_i^p$

for each node $i \in \mathcal{G}$ **do**
  Determine the neighboring node set $\mathcal{N}_i$ for node i based on physical rule type:
  if Rule 1 (Exporter only) **then**
    $\mathcal{N}_i \leftarrow \{i\}$, $\mathcal{E}_i \leftarrow \emptyset$
  else if Rule 2 (Importer only) **then**
    $\mathcal{N}_i \leftarrow \mathcal{N}_i^{all}$, $\mathcal{E}_i \leftarrow \mathcal{E}_i^{all}$
  else if Rule 3 (Importer & Exporter) **then**
    $\mathcal{N}_i \leftarrow \{j | j \text{ exports to } i\}$
    $\mathcal{E}_i \leftarrow \{e_{jk} | j \text{ exports to } i\}$
  end if
  Compute node attention weights:
  for each $j \in \mathcal{N}_i$ **do**
    $\alpha_{ij}^n \leftarrow \frac{\exp\left(\sigma\left(\mathbf{a}_n^\top [\mathbf{W}_n \mathbf{ci}_i^p \| \mathbf{W}_n \mathbf{ci}_j^p]\right)\right)}{\sum_{k \in \mathcal{N}_i} \exp\left(\sigma\left(\mathbf{a}_n^\top [\mathbf{W}_n \mathbf{ci}_i^p \| \mathbf{W}_n \mathbf{ci}_k^p]\right)\right)}$
  end for
  Compute edge attention weights:
  for each $k \in \mathcal{E}_i$ **do**
    $\alpha_{ik}^e \leftarrow \frac{\exp\left(\sigma\left(\mathbf{a}_e^\top [\mathbf{W}_n \mathbf{ci}_i^p \| \mathbf{W}_e \mathbf{f}_k^{e,p}]\right)\right)}{\sum_{j \in \mathcal{E}_i} \exp\left(\sigma\left(\mathbf{a}_e^\top [\mathbf{W}_n \mathbf{ci}_i^p \| \mathbf{W}_e \mathbf{f}_j^{e,p}]\right)\right)}$
  end for
  Aggregate node features:
  $\tilde{ci}_{\mathcal{N}_i}^p \leftarrow \sigma\left(\mathbf{W}_n \sum_{j \in \mathcal{N}_i} \alpha_{ij}^n \mathbf{ci}_j^p\right)$
  Aggregate edge features:
  $\tilde{ci}_{\mathcal{E}_i}^p \leftarrow \sigma\left(\mathbf{W}_e \sum_{k \in \mathcal{E}_i} \alpha_{ik}^e \mathbf{f}_k^{e,p}\right)$
  Generate final embedding:
  $\tilde{ci}_i^p \leftarrow \mathrm{CONCAT}(\tilde{ci}_{\mathcal{N}_i}^p, \tilde{ci}_{\mathcal{E}_i}^p)$
end for

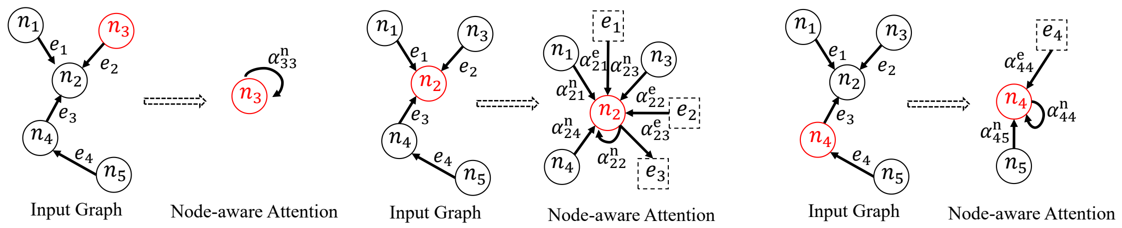
In the carbon network, nodes are subject to physical constraints. For instance, certain nodes exclusively function as exporters. When all nodes are treated equally in the GNN layer, we observe that it can lead to the introduction of noise. Therefore, we develop a node-aware embedding mechanism with a set of rules to effectively eliminate such noise.

More specifically, we summarize three physical rules:

- The carbon intensity of a grid that solely serves as an exporter will remain unaffected by neighboring grids.
- The carbon intensity of a grid that solely serves as an importer will be affected by all neighboring grids.
- The carbon intensity of a grid that serves as both importer and exporter will be impacted by the neighboring grids exporting electricity to it.

Note that a node embedding will be operated in a GNN iteration. Specifically, each node aggregates the features (i.e., electricity flow sequence and carbon intensity sequence) of neighboring nodes using learnable attention weights. In this context, the following rules apply: (Related to rule 1) For a node that solely serves as an exporter, its embedding is based solely on its own features; (Related to rule 2) For a node that solely serves as an importer, its embedding is generated by aggregating the features of all neighboring nodes, links, and their corresponding weight; and (Related to rule 3) For a node that serves as both an importer and exporter, its embedding is generated based on the features of neighboring nodes, links, and their corresponding weights except for the neighboring nodes that import from it and related flows. An example of the node embedding process is shown in Fig. 3.3, where α represents a learnable weight coefficient.

We would like to present the designed node-aware embedding mechanism to meet the physical rules. The attention mechanism [114] is leveraged to learn the node embedding. Specifically, the node embedding for each node i is generated by aggregating features from neighboring nodes and edges with corresponding weights. Let $\mathcal{N}_i^{all}$ and $\mathcal{E}_i^{all}$ be the set of nodes and edges connected to node i, respectively; Note that, node i is also included in $\mathcal{N}_i^{all}$. The neighboring nodes set $\mathcal{N}_i \subseteq \mathcal{N}_i^{all}$ and neighboring edges set $\mathcal{E}_i \subseteq \mathcal{E}_i^{all}$ are different for each node i depending on the specific rules. We will first define the neighboring nodes and edges for each physical rule, and then demonstrate the node-aware embedding calculation based on the defined neighboring nodes and edges.



(a) Node serves only as an exporter (b) Node serves only as importer (c) Node serving as both exporter and importer

Figure 3.3: (a) Illustration of the node-aware attention embedding for node n_3. The generation of the embedding for a node that only serves as an exporter is based on its own feature. (b) Illustration of the node-aware attention embedding of node n_2. The generation of the embedding for a node that only serves as an importer is based on the features from all neighboring nodes and edges. (c) Illustration of the node-aware attention embedding of node n_4. The generation of the embedding for a node that serves as both an importer and exporter is based on the features from neighboring nodes and edges that export to it.

For a node i that conforms to Physical Rule 1, the node-aware embedding is calculated based on the exporter itself, with only the exporter's feature contributing to the embedding. This results in neighboring nodes set $\mathcal{N}_i = \{i\}$ and the neighboring edges set $\mathcal{E}_i = \emptyset$.

For a node i that conforms to Physical Rule 2, the node-aware embedding is calculated considering all the nodes and edges that are connected to it. This leads to the neighboring nodes set $\mathcal{N}_i = \mathcal{N}_i^{all}$ and the neighboring edges set $\mathcal{E}_i = \mathcal{E}_i^{all}$.

For a node i that conforms to Physical Rule 3, the node-aware embedding is calculated based solely on the nodes and edges exporting to it. Consequently, the neighboring nodes set is $\mathcal{N}_i \subset \mathcal{N}_i^{all}$ and the neighboring edges set is $\mathcal{E}_i \subset \mathcal{E}_i^{all}$.

Having established neighboring nodes $\mathcal{N}_i$ and neighboring edges E_i, we can proceed to derive the node-aware embedding using the embedding weights commonly employed in graph attention networks [114]. Specifically, assuming there are N_i neighboring nodes and E_i neighboring edges for each node i, let $\mathbf{ci}_i^p(t) = \{ci_1^p(t), \ldots, ci_{N_i}^p(t)\}$ and $\mathbf{f}_i^{e,p}(t) = \{f_1^{e,p}(t), \ldots, f_{E_i}^{e,p}(t)\}$ be the sets of neighboring node features and edge features, respectively. The embedding weight α_{ij}^n(t) for node i can be defined as follows:

$$\alpha_{ij}^n(t) = \frac{exp(\sigma(a_n^T[W_n ci_i^p(t) \| W_n ci_j^p(t))]))}{\sum_{k \in \mathcal{N}_i} exp(\sigma(a_n^T[W_n ci_i^p(t) \| W_n ci_k^p(t))]))}, \tag{3.5}$$

where W_n denotes a learnable weight matrix linearly transforming the node features into high-level features, a_n represents the parameter vector of a feed-forward network with a single layer, and σ is the activation function. Similarly, the embedding weight of edge $k \in \mathcal{E}_i$ to node i can be defined as $\alpha_{ik}^e(t)$, and is given by:

$$\alpha_{ik}^e(t) = \frac{exp(\sigma(a_e^T[W_n ci_i^p(t) \| W_e f_j^{e,p}(t))]))}{\sum_{j \in \mathcal{E}_i} exp(\sigma(a_e^T[W_n ci_i^p(t) \| W_e f_j^{e,p}(t))]))}, \tag{3.6}$$

where W_e denotes a learnable weight matrix linearly transforming the edge features into high-level features, and a_e represents the parameter vector of a feed-forward network with a single layer.

With the embedding weights $\alpha_{ij}^n(t)$ and $\alpha_{ik}^e(t)$ learned, we proceed to derive the node-aware embedding for node i, which concatenates the weighted features of neighboring nodes and edges. Let $\tilde{ci}_{\mathcal{N}_i}^p(t)$ and $\tilde{ci}_{\mathcal{E}_i}^p(t)$ denote the weighted sum of neighboring nodes and edge features, respectively. We obtain:

$$\tilde{ci}_{\mathcal{N}_i}^p(t) = \sigma(W_n \cdot \sum_{j \in \mathcal{N}_i} \alpha_{ij}^n(t) ci_j^p(t)), \tag{3.7}$$

and

$$\tilde{ci}^{p}_{\mathcal{E}_i}(t) = \sigma(W_e \cdot \sum_{k \in \mathcal{E}_i} \alpha^{e}_{ik}(t) f^{e,p}_{k}(t)). \tag{3.8}$$

Finally, the node-aware embedding $\tilde{ci}^{p}_{i}(t)$ for node i is combined by a concatenation operation $\|$. We summary the node-aware Embedding Mechanism in Algorithm 1.

$$\tilde{ci}^{p}_{i}(t) = (\tilde{ci}^{p}_{\mathcal{N}_i}(t) \| \tilde{ci}^{p}_{\mathcal{E}_i}(t)) \tag{3.9}$$

The single-hop model only considers direct neighboring grids (e.g., Germany for Switzerland) in carbon intensity forecasting. However, in cross-border grids, electricity flows often traverse multi-hop paths (e.g., Switzerland imports electricity from Germany, which itself imports electricity from France). Single-hop models fail to account for these indirect dependencies, leading to significant forecasting errors. The CFCG model addresses this via a node-aware GNN embedding mechanism that implicitly captures multi-hop dependencies by: (1) Global Neighbor Aggregation: The GNN aggregates features from both directly and indirectly neighboring nodes and edges based on carbon flow physical rules; (2) Attention Weights: The learned attention coefficients prioritize influential nodes and edges across the network, enabling the model to implicitly propagate information beyond immediate neighbors. This approach effectively captures spatial dependencies across the entire carbon network, not just immediate neighbors, by modeling the flow of carbon through interconnected grids.

3.3.4 Temporal Dependence Learning

The spatial dependency learning outputs the representation of each node integrated with the representation of the related nodes and links in a carbon network. Subsequently, We proceed to learn the temporal dependency. To achieve this, we leverage LSTM [46], a type of neural network capable of effectively processing time series data. Specifically, a memory cell $\mathbf{c}_t$ is used to store the observation at time step t, and three gates, namely forget gate $\mathbf{f}_t$, input gate $\mathbf{i}_t$ and output gate $\mathbf{o}_t$, are designed to control the state of the memory cell. Denoting $\mathbf{x}_t$ as the input, the LSTM function $LSTM(\mathbf{x}_t)$ is defined as:

$$LSTM(\mathbf{x}_t) = \mathbf{o}_t \odot \phi(\mathbf{c}_t), \tag{3.10}$$

where $\phi(\cdot)$ is a tangent function, $\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{c}_t$, and $\odot$ denotes the operation of product.

The input of the LSTM network for the hourly carbon intensity can be defined as $\widetilde{\mathbf{CI}}_h = \{\widetilde{\mathbf{ci}}^p(0), \ldots, \widetilde{\mathbf{ci}}^p(t), \widetilde{\mathbf{ci}}^p(t+1), \ldots, \widetilde{\mathbf{ci}}^p(T_p)\}$. Then, the output of the LSTM network can be defined as

$$y'_h = LSTM(\widetilde{\mathbf{CI}}_h) \tag{3.11}$$

The input of the LSTM network for the daily carbon intensity can be defined as $\widetilde{\mathbf{CI}}_d = \{\widetilde{\mathbf{ci}}^p(0), \ldots, \widetilde{\mathbf{ci}}^p(t), \widetilde{\mathbf{ci}}^p(t+24), \ldots, \widetilde{\mathbf{ci}}^p(T_p)\}$. Then, the output of the LSTM network can be defined as

$$y'_d = LSTM(\widetilde{\mathbf{CI}}_d) \tag{3.12}$$

The input of the LSTM network for the weekly carbon intensity can be defined as $\widetilde{\mathbf{CI}}_w = \{\widetilde{\mathbf{ci}}^p(0), \ldots, \widetilde{\mathbf{ci}}^p(t), \widetilde{\mathbf{ci}}^p(t+168), \ldots, \widetilde{\mathbf{ci}}^p(T_p)\}$. The, the output of the LSTM network can be defined as

$$y'_w = LSTM(\widetilde{\mathbf{CI}}_w) \tag{3.13}$$

3.3.5 Multi-periodic Patterns Integration

We have developed a gating mechanism [37] which can fuse representations with different granularities. Specifically, the prediction result Y' is generated by a Hadamard product $\odot$. Following the learning of temporal dependence of carbon intensity with LSTM, we obtain the predicted future carbon intensity in three granularities: hour representation y'_h, day representation y'_d, and week representation y'_w. These representations are then integrated to obtain the prediction result as follows:

$$Y' = W_h \odot y'_h + W_d \odot y'_d + W_w \odot y'_w \tag{3.14}$$

where W_h, W_d, and W_w represent learnable parameters for different granularity-aware representations, signifying the weight of the three components on the forecasting result; Y' is the forecasting carbon intensity sequence with the same frequency as the input data.

3.3.6 Carbon Intensity Super Resolution (CISR)

CISR Analysis

The objective of CISR problem is to determine a reconstruction mapping $F : \mathbb{R}^d \rightarrow \mathbb{R}^{\alpha \times d}$ to reconstruct high-frequency carbon intensity data. F can be implemented by a deep neural network. The loss function with Mean Squared Error (MSE) can be written as follows:

$$L(\boldsymbol{ci}^h, \boldsymbol{ci}^{h'}) = ||\boldsymbol{ci}^h - \boldsymbol{ci}^{h'}||_2^2 \tag{3.15}$$

Let θ be the parameter set of the deep neural network F. Then, the network is optimized by minimizing the loss function:

$$\theta' = \min_{\theta} L(\boldsymbol{ci}^h, F(\boldsymbol{ci}^l, \theta)) \tag{3.16}$$

Given the ill-posed nature of the CISR problem, regularization is necessary to constrain the solution. Following MAP estimation, the estimation of the corresponding high-frequency carbon intensity $\boldsymbol{ci}^h$ given a low-frequency carbon intensity $\boldsymbol{ci}^l$ can be formulated as the following optimization problem:

$$y\prime = \min_{\boldsymbol{ci}^h} ||Aci_h - \boldsymbol{ci}^l||_2^2 + \Omega(\boldsymbol{ci}^h) \tag{3.17}$$

The term $||A\boldsymbol{ci}^h - \boldsymbol{ci}^l||_2^2$ represents the distortion measurement under the Gaussian noise assumption, while $\Omega(\boldsymbol{ci}^h)$ denotes the regularization term containing prior information. This equation illustrates that $\boldsymbol{ci}_{h\prime}$ is a function of the input $\boldsymbol{ci}^h$ and the down-sampling matrix A. The MAP solution of CISR is equivalent to the following:

$$y' = \mathcal{F}(ci_l, A; \theta) \tag{3.18}$$

When the down-sampling matrix A is fixed, it is equivalent to the CISR mapping we previously established. The prior information is actually embedded in the network parameter set θ. Although the prior information of $\boldsymbol{ci}^h$ is not explicitly modeled, the deep neural network, trained on a large amount of data, inherently contains this prior knowledge. Additionally, the proposed deep neural network avoids the direct modeling of the prior distribution of $\boldsymbol{ci}^h$ by transforming the optimization into an inference problem.

Carbon Intensity Super Resolution Neural Network layers

In this section, we present the design of our carbon intensity super-resolution neural network layers (CISPNN). Currently, the most widely used area for super-resolution is images. A series of related technologies related to image super-resolution have been developed [64][38]. However, these methods can not be directly applied to the CISR problem, as the carbon intensity sequence is a one-dimensional signal and has a relationship in the temporal dimension. Currrently, there is still no related network developed for carbon intensity series. The most relevant works are smart meter data

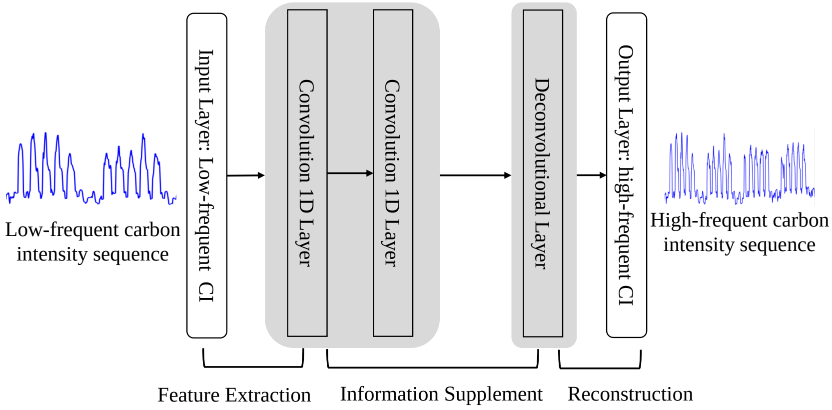


Figure 3.4: The proposed carbon intensity super-resolution layers.

super-resolution in the electricity sector. The neural networks-based method of super-resolution for electricity can be categorized as CNN-based [65][63] and GAN-based [102]. GAN-based super-resolution methods need a substantial amount of data due to the nature of the GAN network, which is not suitable for our CISR problem. Therefore, we develop a lightweight CISRNN inspired by [65], which is suitable for the small high-resolution carbon intensity dataset.

The network architecture of CISRNN is shown in Fig.3.4. Specifically, CISRNN takes the low-frequent carbon intensity sequence as the input and outputs the estimated high-frequency data directly. The network consists of three parts: feature extraction, information supplement, and reconstruction. The first two parts contain 1D convolutional layers and the last one contains a transposed convolutional layer.

3.4 Evaluation

3.4.1 Evaluation Setup

In this section, we will present the evaluation of our CFCG model and address the following research questions.

RQ1: How does CFCG compare to state-of-the-art day-ahead carbon intensity forecasting techniques?

RQ2: How do the designs of the key components contribute to the performance of CFCG?

RQ3: What explainable patterns does CFCG capture in carbon intensity forecasting?

Data Description. We use the real-world electricity dataset obtained from ENTSO-E [111] as Table 3.2 shown to evaluate the performance of the CFCG model. The data encompasses of hourly electricity production data and cross-border electricity flow data from 2019 to 2021, covering a total of 28 countries. As only Germany, Netherlands, and Hungry have high-frequency data (15 minutes), we use the data in the three countries to evaluate CFCG-CISPNN.

Evaluation Metrics. We adopt three commonly used metrics: (1) *Mean absolute error (MAE)*: $MAE = \frac{1}{m}\sum_i^m |y_i - y_i'|$. (2) *Mean Absolute Percentage Error (MAPE)*: $MAPE = \frac{1}{m}\sum_i^m \left|\frac{y_i - y_i'}{y_i}\right|$. (3) *Root Mean Squared Error (RMSE)*: $RMSE = \sqrt{\frac{1}{m}\sum_i^m |y_i - y_i'|}$.

Carbon Intensity Ground Truth. As discussed, the direct measurement of carbon intensity is not feasible. Various carbon accounting methods are utilized to estimate the carbon intensity ground truth. In this paper, the baseline ground truth is derived from [87].

Training and Testing. We use the first two years of data as the training set, and the final year of data as the test set. This ensures the model is evaluated on future unseen data, avoiding leakage of future information into training. We also adopt the sliding window mechanism. The window slides across the entire training set to generate multiple input-output pairs, enabling the model to learn temporal patterns at various time steps. We employ the RMSprop optimizer [91] with a learning rate of 0.0001. A batch size of 64 is selected based on GPU memory constraints. We adopt

Table 3.2: The description of the dataset.

Data	Resolution	Periods	Regions
Electricity generation mix	Hourly, 15 minute	2019-2021	AT BE BG CH CY CZ DE DK EE ES FI FR GR HR
Electricity flow	Hourly, 15 minute	2019-2021	HU IE IT LT LV NL NO PL PT RS RO SE SI SK

early-stopping and checkpointing techniques to prevent overfitting. Additionally, to keep the forecasting accuracy, CFCG is re-trained every three months.

Baselines. We compare our CFCG model with advanced baselines in (1) specialized carbon intensity forecasting methods: regional model, e.g, *DACF* [72]; one-hop model, e.g., *TSBP* [90] and *HCFM* [58]; and (2) general time forecasting models: transformer-based model, e.g., Autoformer [126], informer [134], PatchTST [78], itransformer [66]; GNN-based model, e.g., DCRNN[62], STGCN [130], STSGCN [101].

3.4.2 Performance Results (RQ1)

Performance Results of CFCG

To evaluate the effectiveness of CFCG, we conducted comprehensive comparisons with three categories of baseline methods. Table 3.4.2 presents the aggregated performance metrics (e.g., MAE, RMSE, MAPE) with the best-performing method highlighted in bold font, while Figure 3.5 reveals country-level error patterns.

Comparison with specialized carbon intensity forecasting Models: CFCG achieves significant improvements over state-of-the-art carbon-specific models: 24.27% MAE (28.33 vs. 37.41) and 26.46% MAPE (12.92 vs. 17.57) reduction compared to DACF (regional model): Demonstrates the critical importance of modeling cross-border electricity flows, as DACF's single-region assumption fails to capture imported

Table 3.3: Comparison of the performance of all methods in terms of MAE, MAPE, and RMSE (based on lifecycle emission factors).

Methods		Metrics		
		MAE	RMSE	MAPE
General TS. Forecasting	Autoformer	35.65	45.87	16.91
	Informer	35.52	44.75	16.67
	PatchTST	35.40	44.50	16.53
	iTransformer	35.30	43.81	16.51
GNN-based TS. Forecasting	DCRNN	32.52	41.57	14.92
	STGCN	32.32	41.28	14.89
	STSGCN	32.25	41.12	14.85
Specialized carbon TS. Forecasting	DACF	37.41	48.39	17.57
	TSBP	35.38	45.15	15.78
	HCMF	37.44	47.34	16.22
	CFCG	**28.33**	**37.72**	**12.92**

carbon emissions. 24.33% MAE (28.33 vs. 37.44) and 20.34% MAPE (12.92 vs. 16.22) improvement over HCMF (one-hop neighbor model): Highlights the limitation of local neighborhood approaches in handling multi-hop carbon flows through interconnected grids. **Comparison with General Time Series Models:** CFCG outperforms advanced temporal models by 19.75% in MAE (28.33 vs. 35.30) and 21.74% (12.92 vs. 16.51) in MAPE compared to the best transformer variant (iTransformer), also confirming that ignoring the electricity exchange in the spatial dimension will lead to significant errors. The 12.16% MAE (28.33 vs. 32.25) and 12.99% MAPE (12.92 vs. 14.85) improvement over the strongest GNN-based baseline (STSGCN) emphasizes the value of our node-aware embedding mechanism and multi-periodic pattern encoding in handling complex grid interaction patterns.

Geographical Performance Analysis: Fig.3.5 shows predicting errors as a func-

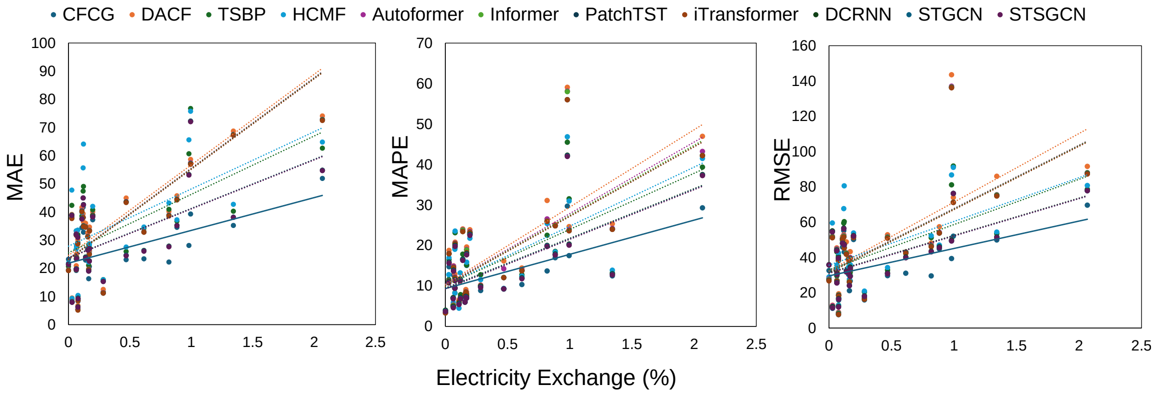


Figure 3.5: The scatter plot represents the performance of each model across different countries.

tion of the ratio of electricity imports (ranging from 0 to over 200%). Each data point represents a specific country, with varying import ratios across different countries. Traditional carbon forecasting models (DACF, TSBP, HCMF) and general time series models (e.g., Autoformer, etc.) perform comparably in regions with less 30% electricity exchanges. It is observed that forecasting errors tend to increase with higher levels of electricity exchanges, indicating that the complexity of forecasting rises with increased dynamics. Notably, our CFCG model not only surpasses all other approaches in performance but also exhibits a slower increase in errors relative to the rise in electricity exchanges ratios.

3.4.3 Model Ablation Study (RQ2)

We proceed to evaluate our designs in three key components of CFCG through an ablation study: the granularity-aware encoding component, the node-aware embedding mechanism, and the carbon intensity super-resolution neural network layers.

The Granularity-Aware Encoding Component: We investigate the influence of the granularity in capturing the temporal patterns on the results of carbon intensity forecasting. We analyze four configurations based on the granularity period g of the CFCG model:

- $CFCG_h : g \in \{hour\}$
- $CFCG_{h,d} : g \in \{hour, day\}$
- $CFCG_{h,w} : g \in \{hour, week\}$
- $CFCG_{h,d,w} : g \in \{hour, day, week\}$

The results are shown in Fig. 3.6. It is evident that $CFCGh, d, w$ consistently exhibits superior performance compared to the other configurations. Specifically, $CFCG_{h,d,w}$ outperforms $CFCG_h$ by 11.07% (from 12.92 % to 14.35 %) in terms of MAPE. This highlights the beneficial impact of our design in integrating multiple time granularities into the CFCG model, enhancing the learning process. Additionally, the slightly better performance of $CFCGh, d$ compared to $CFCG_{h,w}$ may be attributed to the finer granularity. In summary, capturing temporal patterns in multiple granularities contributes to the overall improvement in model performance.

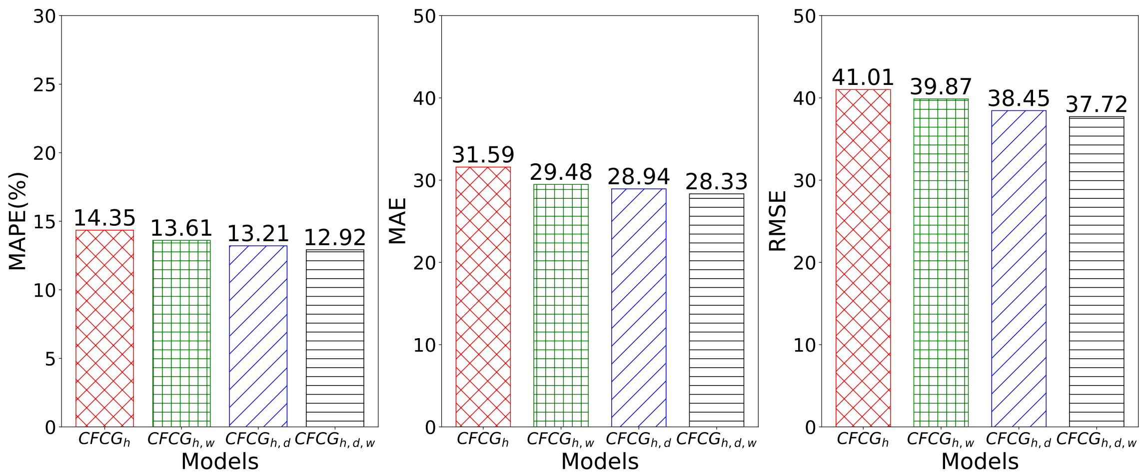


Figure 3.6: Effect of multi-granularity dynamics in terms of MAPE, MAE, and RMSE.

The Node-aware Embedding Mechanism: We examine the impact of our node embedding mechanism, which involves embedding different types of nodes according to different physical rules) on the results of carbon intensity forecasting. We compare the performance of *CFCG* with a simplified CFCG model denoted as $CFCG_{wn}$, where the node-aware embedding mechanism is removed. Fig. 3.7 (a) illustrates the errors. It is observed that the average forecasting error with and without the node-aware embedding mechanism is comparable, indicating that, on average, the level of noise is not substantial. However, it is noted that in specific countries, the presence of the node-aware embedding mechanism can lead to a more significant impact. Fig. 3.7 (b) demonstrates that the node-aware embedding mechanism improved predicted results in Sweden and the Czech Republic by 12.76% and 13.07%, respectively. When we investigate the data in detail, we find that throughout the 2019–2021 period, Sweden and the Czech Republic exported 4185.95 MW and 2682.14 MW of power hourly, respectively. This accounted for 23.21% and 29.29% of their respective electricity generation. As a result, the node-aware mechanism is more significant.

The Carbon Intensity Super-Resolution Neural Network Layers:

We examine the impact of CISPNN layers based on five configurations of the different methods to achieve data super-resolution:

- CFCG-Linear: CFCG is enhanced by a linear interpolation.
- CFCG-Cubic: CFCG is enhanced by a cubic interpolation.
- CFCG-Lagrange: CFCG is enhanced by a Lagrange interpolation.
- CFCG-GAN: CFCG is enhanced by a GAN-based model [102] to achieve carbon data super-resolution.
- CFCG-CISPNN: our original design.

The results are displayed in Table 3.4. Overall, CFCG-CISPNN shows consistently

Table 3.4: The performance of CISPNN.

Model \ MAPE(%) \ Country	DE	NL	HU	Average
CFCG-Linear	11.93	7.47	8.4	9.27
CFCG-Cubic	11.89	7.46	8.38	9.24
CFCG-Lagrange	12.4	7.78	8.78	9.65
CFCG-GAN	11.55	7.43	8.27	9.08
CFCG-CISPNN	11.26	7.11	8.49	8.95

better performance than the other configurations at most 7.86%. More specifically, CFCG achieves an average improvement of 3.54 % (from 9.27% to 8.95%), 3.28% (from 9.24% to 8.95%), 7.86% (from 9.65% to 8.95%), and 1.49% (from 9.08% to 8.95%) in MAPE. The result shows that our CISPNN layers are capable of effectively reconstructing high-resolution data and performing good data reconstruction and information supplement on low-resolution carbon intensity time series for higher accuracy.

3.4.4 Explainable Patterns Learned (RQ3)

We attribute the performance improvements of the CFCG model to its ability to capture spatial and temporal dependencies. We now visualize that the CFCG models has indeed learned these dependencies.

Spatial dependency learned. Fig. 3.8 shows the carbon intensity ground truth of the countries in the cross-border grids and the predicted results in the spatial dimension. We present the first two days of the testing data. The color represents the ground truth, where darker shades indicate higher carbon intensity, while the size of a circle represents the forecasting results, with larger circles indicating greater carbon intensity. It is evident that the pattern of our forecasting results aligns with

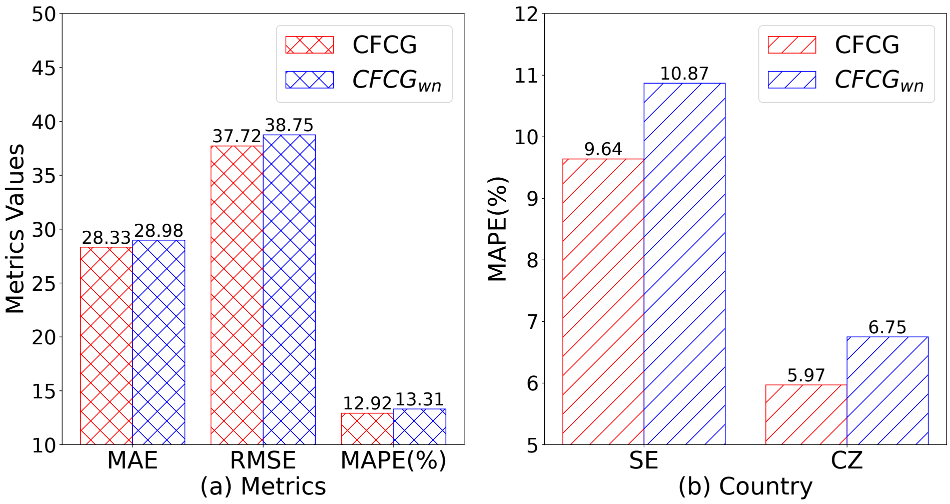


Figure 3.7: (a) The forecasting error with and w/o node-aware embedding mechanism (NEM). (b) The forecasting error with and w/o node-aware embedding mechanism (NEM) on selected countries.

the ground truth in the spatial dimension.

Temporal dependence learned. Fig. 3.9 shows the carbon intensity ground truth of two randomly selected countries (Sweden and Hungary) in the cross-border grids and the predicted results in the temporal dimension. We can find that the pattern of our predicted results aligns with the ground truth in the temporal dimension.

3.5 Case Study

We present the case studies where we use our CFCG model to support applications (e.g., LLM inference [23] and LLM training [33]) to reduce their operational carbon emissions. The operational carbon emission of a LLM comes from hardware electricity consumption. The amount of operational carbon released from LLMs depends heavily on where and when it happens, as the carbon intensity of electricity is highly variable

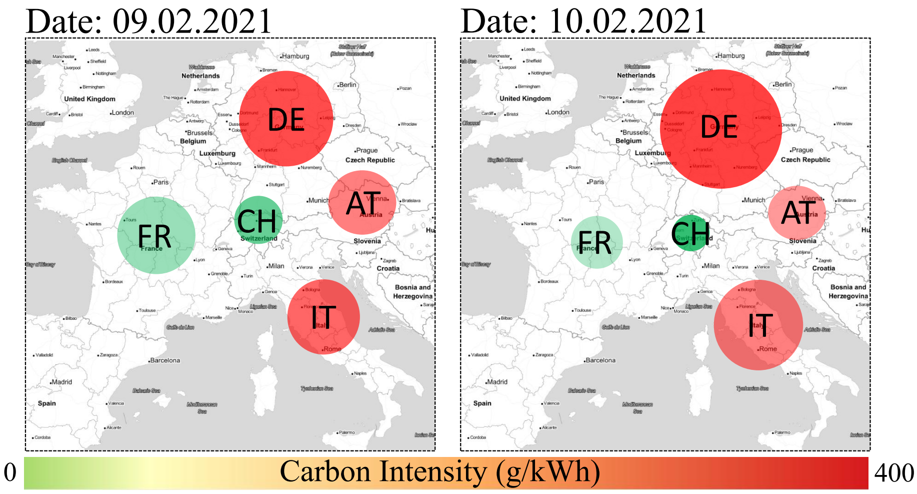


Figure 3.8: The spatial dependency learned by CFCG.

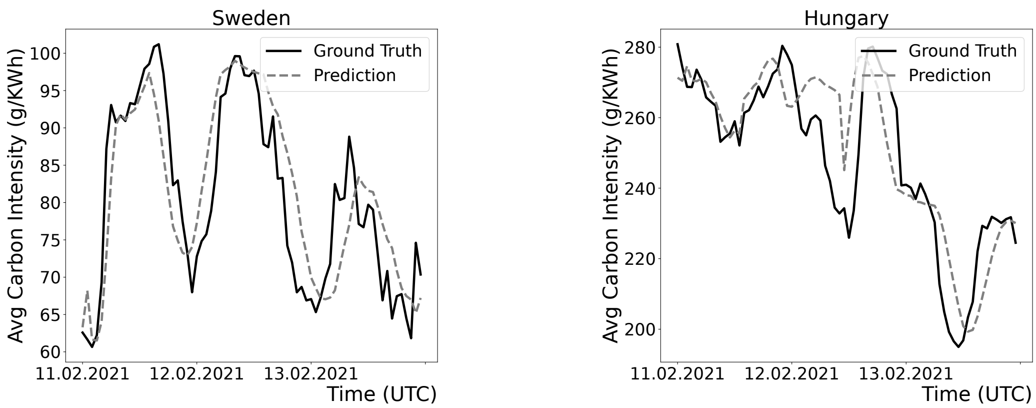


Figure 3.9: The temporal dependency learned by CFCG.

across power grids. Supported by the future carbon intensity forecasted by CFCG, applications can have chances to reduce their own carbon emissions. Note that the carbon intensity of electricity is not published in real time in many countries.

3.5.1 LLM Training

LLMcarbon[33] models the carbon emissions in the training process of LLMs. We extend their work to evaluate the maximum potential of carbon reduction in LLMs training process supported by CFCG. The amount of operational carbon released from LLMs training can be calculated as follows:

$$CO_2 = Energy_{all} \times CI \tag{3.19}$$

where $Energy_{all}$ represents the electricity consumed in LLMs training, and CI denotes the carbon intensity of the electricity.

The total electricity consumed in LLMs training $Energy_all$ can be calculated as follows:

$$Energy_{all} = \sum_{i \in Hardware_set} (P_i \times n_i \times t_i) \times PUE \tag{3.20}$$

where P_i is the power of hardware unit i; n_i presents the number of hardware i; t_i is the total execution time of hardware i; and PUE represents the power usage effectiveness of the data center where the LLMs training conducts. Eq. 3.19 uncovers that the amount of carbon released from LLM training is affected by the amount of electricity consumed $Energy_{all}$ and the carbon intensity of the electricity CI. Eq. 3.20 shows the amount of electricity consumed is relatively fixed in a training workload using the same hardware set. While CI is highly variable across temporal dimensions. Fig. 3.10 shows the carbon intensity of electricity during the three days in winter and summer in Germany. We observe that electricity is usually the greenest during mid-day in summer, when most solar energy is available, while electricity is greenest at night in winter when electricity demand is generally low, and fossil fuel power plants are throttled back. We also observe that there is a high variation of values in the dataset of Germany in 2021, reaching from 87.4 to 608.63 gCO2/kWh. The reasons behind these phenomena are that one-third of German electricity production comes from highly variable, renewable sources (e.g., 24.7% wind and 8.3% solar power). The high variation indicates the potential to reduce carbon emissions of the latency-tolerant LLMs training task by scheduling the workload to low carbon periods in the future, as Fig. 3.10 shows.

We evaluate the maximum potential of carbon reduction in the latency-tolerant LLMs training process supported by CFCG through two basic scheduling criteria: (i) WorkFirst: workloads are processed immediately when arriving. (ii) CarbonFirst: workloads are shifted to the period when the carbon intensity is lowest according to the CFCG forecasted results. To evaluate the maximum of carbon reduction, the computing capacity is assumed to be elastic and unlimited.

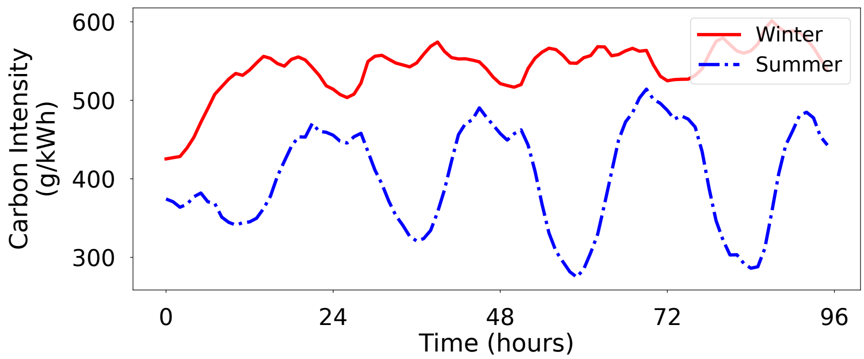


Figure 3.10: The carbon intensity of electricity in winter (06.01.2021-09.01.2021) and summer (01.07.2021-04.07.2021) in Germany.

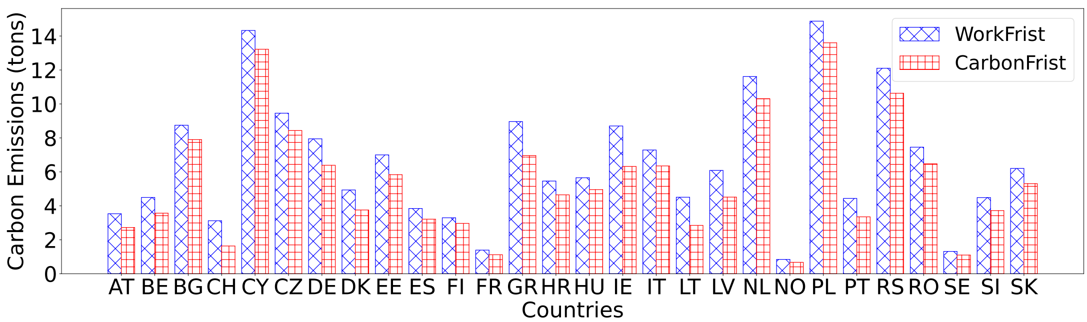


Figure 3.11: The carbon emissions of the LLM training across different countries using two basic scheduling criteria.

Fig.3.11 shows the amount of carbon released (in tons) during the LLM (i.g., GShard [83]) training process across different countries using the two different scheduling criteria. Overall, the average performance of carbon reduction can improve by 24.29%. More specifically, the performance of carbon reduction can improve at most 90.89%

(3.12 vs. 1.63) in Switzerland and at least 8.35% (14.33 vs. 13.22) in Cyprus.

3.5.2 LLM Inference

Carbonmin [23] models the carbon emissions in the inferences process of LLMs and geographically shifts workloads to low-carbon regions for reducing the amount of carbon emissions from LLMs inferences. We extend their work and evaluate the carbon reduction in 28 countries supported by CFCG.

Inferences requests are directed based on varied optimization criteria: (i) Local: requests are processed at the region that they are received at; and (ii) CarbonMin: requests are directed to the region with the lowest carbon intensity, minimizing carbon emissions. These two criteria are subject to the computing capacity constraint. We assume that the amount of workload in one time period at each region is the same (denoted as *Local*). The computing capacity is also the same in each region and greater than *Local*. To evaluate the maximum carbon reduction, we also consider (iii) CarbonMin (unlimited) which eliminates the computing capacity constraint.

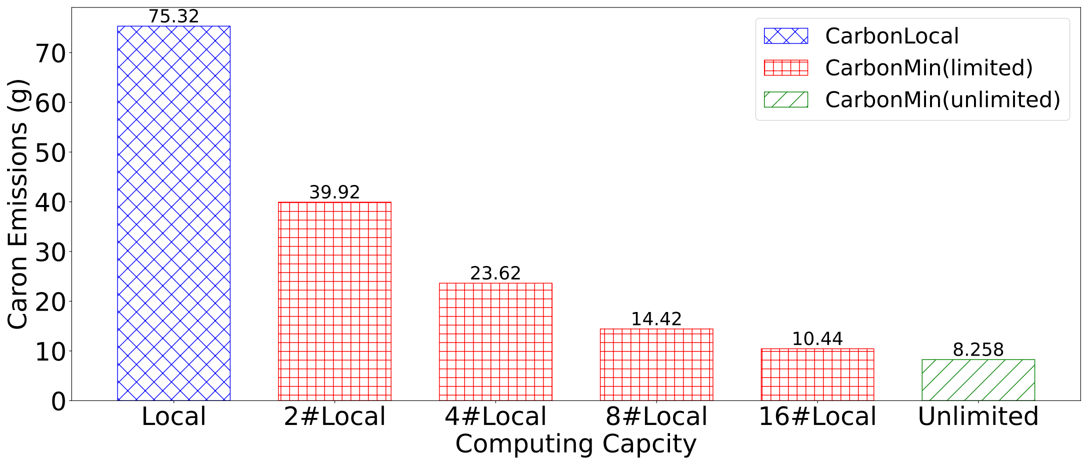


Figure 3.12: The carbon emissions of the LLM inferences based on different request direction criteria.

Fig.3.12 shows the amount of carbon released (in grams per 1k inferences) during the

GPT3 inferences using the three different optimization criteria supported by CFCG in the 28 different countries. Overall, the average performance of carbon reduction can improve by 74.33%. More specifically, the amount of carbon emissions can be reduced by at most 89.03% (8.26 vs. 75.32) using CarbonMin (unlimited) and at least 46.99% (39.92 vs. 75.32) using CarbonMin (limited) when the computing capacity is two times *Local*.

3.6 Chapter Summary

In this paper, we studied day-ahead carbon intensity forecasting in the context of cross-border power grids. In cross-border power grids, the total carbon emissions of a local grid will not only carry the carbon emissions of its own grids (or its adjacent neighbor grids) but also carry the carbon emissions of the member grids in the cross-border grid. Current forecasting studies have yet to take this factor into their learning model. We showed that ignorance can lead to forecasting errors. We also showed that even we first estimate the carbon intensity of cross-border grids and use such history data to perform a learning using existing methods, e.g., LSTM, there can still be non-trivial errors. From the perspective of carbon intensity forecasting, cross-border power grids have both spatial and temporal dependencies in the carbon flow among member grids, and we need new designs to appropriately capture such dependencies. We developed a new learning model with embedded layers based on GNN and LSTM, learning the spatial and temporal dependencies, as well as designs of multi-periodic pattern encoding and node-aware embedding. We evaluated our model through the real-world data of the cross-border power grids in Europe with 28 member countries and the results showed that our model is effective in improving forecasting errors.

Chapter 4

General Carbon Intensity Forecasting via Dual Graph Empowered Time Series Foundation Model

4.1 Introduction

As part of the United Nations' sustainable development goals, there is increasing interest in the Artificial Intelligence (AI) research community in developing techniques to address societal problems [108, 21, 79, 132, 116, 53, 28], especially to facilitate the decarbonization of AI computing [92] and society and achieve carbon neutrality by 2050 [77].

Recently, demand-side techniques have been proposed to reduce the operational carbon footprint of carbon-aware systems resulting from consuming electricity, i.e., shift electricity demand from periods when the carbon intensity of electricity is high to periods when it is low. For example, cloud computing providers leverage the variations in carbon intensity to schedule workloads since some computing loads exhibit temporal elasticity [2, 104] and LLM inference can reduce its carbon footprint by guiding the autoregressive generation process according to carbon intensity [59]. However, the effectiveness of such carbon-aware systems fundamentally depends on accurate electricity carbon intensity forecasting, a critical yet underexplored component in current implementations.

The carbon intensity of electricity (in $grams/kWh$) is defined as the average amount of carbon released per unit of electricity generated. While commercial services like ElectricityMaps [73] offer regional forecasts through proprietary models, their closed architecture and high operational costs limit broad accessibility. Academic efforts have proposed various forecasting models [71, 131], but these typically require substantial historical data from target regions, a requirement that proves impractical given the global scarcity of reliable carbon measurement infrastructure. This data scarcity creates a critical performance gap: existing models exhibit significantly degraded accuracy when applied to regions with insufficient historical data, thereby hindering global decarbonization efforts through computing solutions.

In this paper, we present a holistic study by formulating a new General Carbon Intensity Forecasting problem (GCIF). To solve GCIF, we propose DGCFM, a new **D**ual **G**raph (i.e., a heterogeneous spatial-temporal carbon graph and a metadata-assisted carbon hypergraph) empowered **C**arbon-domain time series **F**oundation **M**odel. First, we introduce carbon flows, which collectively establish a carbon network. Then, we model the carbon network as a heterogeneous spatial-temporal carbon graph. To uncover complex temporal patterns, especially under insufficient carbon data, DGCFM leverages a pre-trained Time Series Foundation Model (TSFM). The TSFM is a large model pre-trained on vast quantities of time series data and has strong generalization capability in time series tasks. Furthermore, we adapt the TSFM to carbon intensity forecasting through a metadata-assisted carbon hypergraph enhanced fine-tuning on carbon datasets. More specifically, we propose a two-phase clustering based on the hypergraph to avoid the negative transfer in the fine-tuning process. To uncover the complex spatial dependency, we propose a Physics-constrained Heterogeneous Garph Neural Network (PHGNN) to empower the ability of the TSFM to capture the spatial dependency.

We evaluate DGCFM on 102 real-world datasets, including carbon intensity and partial electricity transmission data for 102 regions. We use (81 datasets to fine-tune the TSFM backbone in DGCFM, and 21 datasets to construct a carbon network to evaluate the model. Our evaluations show that our model can achieve an improvement of 20.04% as compared to the state-of-the-art carbon forecasting model in the carbon data-limited scenario.

The contributions of this paper can be summarized as follows:

- We present a new study of carbon intensity forecasting across diverse regions and formulate it as a new carbon intensity forecasting problem, GCIF. We model GCIF as a heterogeneous spatial-temporal carbon graph by introducing carbon flows.

- We propose DGCFM, a general carbon domain foundation model for carbon intensity forecasting across diverse regions. DGCFM leverages the pre-trained TSFM to achieve the general capability of uncovering temporal dependency under insufficient carbon data. Furthermore, DGCFM leverages the dual graph to empower TSFM to capture spatial dependency.

- We conduct an evaluation of DGCFM on 102 real-world carbon datasets. The results demonstrate that DGCFM steadily outperforms the state-of-the-art baseline.

4.2 Problem Statement

4.2.1 Spatial-Temporal Carbon Network Modeling

In this section, we model the carbon intensity and electricity exchanges into a spatial-temporal carbon network. Let $E_i(t)$ represent the electricity generation of grid i at time t. Recall that electricity is a mix of the electricity generated by each energy source (e.g., gas, wind, etc). Let $\mathcal{W}$ represent the set of energy sources and $W = |\mathcal{W}|$. Let $E_i^k(t)$ represent the electricity generated by energy source k in grid i. Then we have $E_i(t) = \sum_{k\in\mathcal{W}} E_i^k(t)$. Let ef^k denote the carbon emission factor associated with the energy source k. The carbon emission factors for the predominant energy sources utilized in electricity generation are detailed in Table 1.1.

Since some renewable energy sources(e.g., wind, solar, etc) change with time, the composition of electricity generated by a power grid i fluctuates over time. These fluctuations result in dynamic carbon emissions. At timestamp t, a power grid i generates electricity from various energy sources. This variability is due to fluctuations in certain energy sources, such as solar and wind, at different times. Consequently, carbon emissions are dynamic and vary over time.

Carbon intensity $ci(t)$ is defined as the ratio of the total carbon emitted against the total electricity generation. Specifically, $ci(t) = \sum_{k\in\mathcal{W}} ef^k \times E^k(t)/E(t)$.

When two power grids have transmission links, they can trade electricity in practice. They are referred to as *neighboring grids.* Let i and j represent two regional power grids. The electricity flow $f_{ij}^e(t)$ represents the total amount of electricity exchange from grid i to grid j in a period of time starting at timestamp t. Note that the grid j bears the carbon emitted by the electricity flow f_{ij}^e from power grid i. Intuitively, we can think the carbon "flows" from the power grid i to j. Then, we call this *carbon flow.* Let carbon flow $f_{ij}^c(t)$ denote the amount of carbon emitted by the electricity flow $f_{ij}^e(t)$ at time t, which is the total amount of carbon associated with the corresponding electricity in power grid i and transmitted to grid j at time t. Therefore, $f^c_{ij}(t) = f^e_{ij}(t) \times ci_i(t)$. We formally present the spatial-temporal carbon network modeling formally in the following.

- **Spatial-temporal carbon network.** At each timestamp t, we represent a carbon network using a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$. The set of nodes $\mathcal{V} = v_1, ..., v_n$ represents regional power grid, with $|\mathcal{V}| = n$. Let $\mathcal{E}$ denote a set of links, where directed link (v_i, v_j) represents that power grid v_j can directly import electricity from power grid v_i. The connectivity among the power grids is represented by a binary adjacency matrix $\mathcal{A} \in \mathbb{R}^{n\times n}$.

- **Carbon intensity time series.** This is a feature on nodes and also the input of our learning model. Recall that $ci_i(t)$ represents the carbon intensity of a grid i at timestamp t. We use a vector $\boldsymbol{ci}_i = \{ci_i(0), ..., ci_i(t)\}$ to represent the carbon intensity time series of grid i from time 0 to t, and a vector $\boldsymbol{ci}(t) = \{ci_0(t), ..., ci_i(t)\}$ to denote the carbon intensity of all grids at time t. We denote the carbon intensity time series of all grids from time 0 to t as a two-dimensional matrix $\boldsymbol{CI} = \{\boldsymbol{ci_i}\}$.

- **Electricity flow time series.** This is a feature on edges and also the input of our learning model. Recall that $f^e_{ij}(t)$ represents the electricity flow from grid i to j at timestamp t. We use the matrix $\boldsymbol{f^e}(t) = \{\forall i, j, f^e_{ij}(t)\} \in \mathbb{R}^{\mathcal{E}}$ to represent all the electricity flows of a carbon network at times t. We denote the electricity flow time series of a carbon network at time t as a three-dimensional matrix $\boldsymbol{F^e} = \{\boldsymbol{f^e}(0), ..., \boldsymbol{f^e}(t)\}$.

4.2.2 Problem Formulation

Problem GCIF (**G**eneral **C**arbon **I**ntensity **F**orecasting): Given the carbon network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, historical carbon intensity time series data $\boldsymbol{CI} = \{(\boldsymbol{ci}(0), ..., \boldsymbol{ci}(t)\}$,

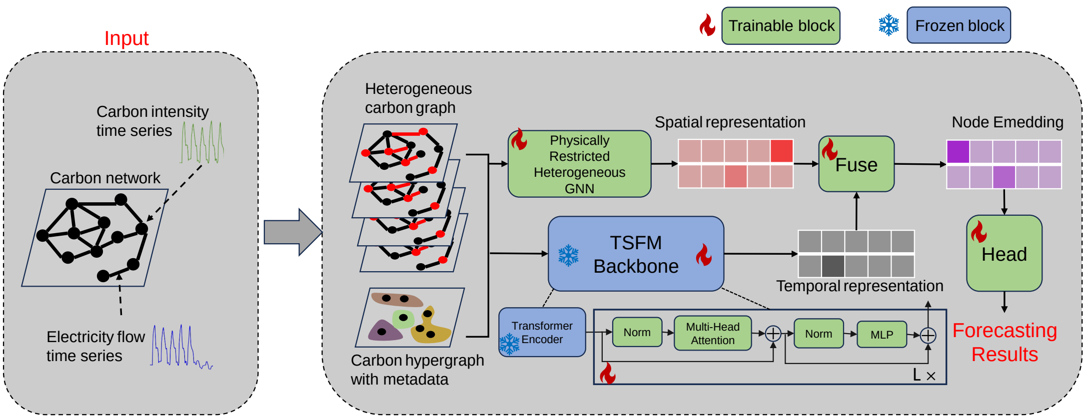


Figure 4.1: The overall architecture of DGCFM.

historical electricity flow time series data $\boldsymbol{F^e} = \{\boldsymbol{f^e}(0), ..., \boldsymbol{f^e}(t)\}$, learn an general predictive model $y = f(\mathcal{G}, \boldsymbol{CI}, \boldsymbol{F^e})$, which can infer the day-ahead carbon intensity of electricity across diverse regions (i.e., regions with insufficient/sufficient carbon data).

4.3 Methods

The overview architecture of DGCFM is exhibited in Figure 4.1. The input of DGCFM is historical carbon intensity and electricity. Initially, we map the spatial-temporal carbon network into a heterogeneous spatial-temporal carbon graph. Subsequently, DGTSM leverages a Physics-constrained Heterogeneous Graph Neural Network (PHGNN) to capture the spatial dependency in the heterogeneous carbon graph with physical characteristics. Then, DGCFM leverages a fine-tuned TSFM enhanced by a metadata-assisted carbon hypergraph to capture the complex temporal dependency, especially under insufficient data. Finally, a fusion block fuses the spatial representation output by PHGNN and the temporal representation output by the TSFM to generate forecasting results.

4.3.1 Physics-constrained Heterogeneous GNN

In the carbon network, nodes are subject to physical constraints. For instance, certain nodes exclusively function as exporters or importers, while others can serve as both importers and exports. Additionally, inflow and outflow have different accounting methods (e.g., direct coupling and aggregated coupling [93]) for carbon intensity. When all nodes are treated equally in the GNN layer, it can lead to the introduction of noise. Inherently, the carbon network is a heterogeneous graph since such physical constraints exist. Therefore, we develop a Physics-constrained Heterogeneous GNN (PHGNN) based on the message-passing framework [34] and attention mechanism[115] to capture the complex spatial dependency. The key of PHGNN is to extend the original attention mechanism by allocating learnable node-type and edge-type weights according to the physical constraints in the embedding process.

First, we model the carbon network into a heterogeneous carbon graph with three different types of nodes and two types of edges. Here, we summarize the types of nodes with different physical constraints:

- Exporter: Its carbon intensity remains unaffected by neighboring grids or electricity flows.
- Importer: Its carbon intensity will be affected by all neighboring grids and electricity inflows and outflows.
- Both importer and exporter: Its carbon intensity will be impacted by the neighboring grids exporting electricity to it and electricity inflows and outflows.

Here, we summarize the types of edges:

- Outcoming electricity arc: It will not affect the carbon intensity of the node.
- Incoming electricity arc: It can affect the carbon intensity of the node.

We would like to present the designed PHGNN with learnable node-types and edge-types embedding by including the type information of nodes and edges in the embedding process to capture the physical constraints and heterogeneity in the carbon network.

$\mathcal{N}_i^{all}$ and $\mathcal{E}_i^{all}$ denote the set of nodes and edges connected to node i, respectively. Note that, node i is also included in $\mathcal{N}_i^{all}$. The neighboring nodes set $\mathcal{N}_i \subseteq \mathcal{N}_i^{all}$ and neighboring edges set $\mathcal{E}_i \subseteq \mathcal{E}_i^{all}$ are different for each node i.

Having established neighboring nodes $\mathcal{N}_i$ and neighboring edges $\mathcal{E}_i$, we can proceed to derive the node embedding. Specifically, assuming there are N_i neighboring nodes and E_i neighboring edges for each node i, let $\mathbf{ci}_i^p(t) = \{ci_1^p(t), \ldots, ci_{N_i}^p(t)\}$ and $\mathbf{f}_i^{e,p}(t) = \{f_1^{e,p}(t), \ldots, f_{E_i}^{e,p}(t)\}$ denote the sets of features on neighboring node features and edge, respectively. We allocate embedding weights $r^e_{\theta_e(i,j)}$, $r^n_{\theta_n(i)}$ for each edge type $\theta_e(i,j)$, and node type $\theta_n(i)$, respectively. We calculate the attention score with the edge type embeddings and node embeddings as follows:

The embedding weight of edge $k \in \mathcal{E}_i$ to node i is defined as $\alpha_{ik}^e(t)$, and we have:

$$\alpha_{ik}^e(t) = \frac{exp(\sigma(a_e^T[W_n ci_i^p(t) \| W_e f_j^{e,p}(t)) \| W_r^e r^e_{\theta_e(i,j)})]))}{\sum_{j \in \mathcal{E}_i} exp(\sigma(a_e^T[W_n ci_i^p(t) \| W_e f_j^{e,p}(t)) \| W_r^e r^e_{\theta_e(i,j)}]))}, \tag{4.1}$$

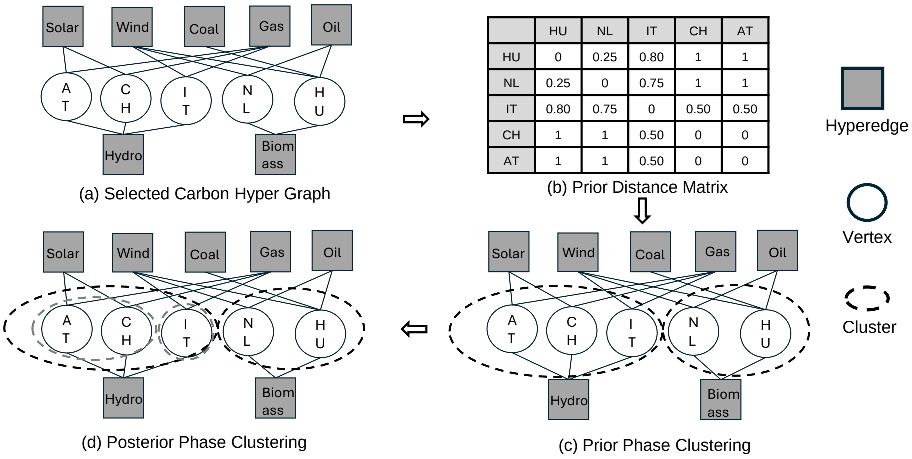
where W_n denotes a learnable weight matrix linearly transforming the node features into high-level features, a_e represents the parameter vector of a feed-forward network with a single layer, $r^e_{\theta_e(i,j)}$ represents the type of edge from node i to j, and σ is the activation function.

Similarly, the embedding weight of node j to node i can be defined as follows:

$$\alpha_{ij}^n(t) = \frac{exp(\sigma(a_n^T[W_n ci_i^p(t) \| W_n ci_j^p(t) \| W_r^n r^n_{\theta_n(i)}]))}{\sum_{k \in \mathcal{N}_i} exp(\sigma(a_n^T[W_n ci_i^p(t) \| W_n ci_k^p(t)) \| W_r^n r^n_{\theta_n(i)}]))}, \tag{4.2}$$

where $r^n_{\theta_n(i)}$ represents the type of node i.

With the embedding weights $\alpha_{ij}^n(t)$ and $\alpha_{ik}^e(t)$ learned, we can derive the embedding result for node i. Let $\tilde{ci}^p_{\mathcal{N}_i}(t)$ and $\tilde{ci}^p_{\mathcal{E}_i}(t)$ denote the weighted sum of neighboring nodes



	HU	NL	IT	CH	AT
HU	0	0.25	0.80	1	1
NL	0.25	0	0.75	1	1
IT	0.80	0.75	0	0.50	0.50
CH	1	1	0.50	0	0
AT	1	1	0.50	0	0

Figure 4.2: Example of metadata-assisted carbon hypergraph and two-phase clustering.

and edge features, respectively. We obtain:

$$\tilde{ci}^{p}_{\mathcal{N}_i}(t) = \sigma(W_n \cdot \sum_{j\in\mathcal{N}_i} \alpha^{n}_{ij}(t) ci^{p}_{j}(t)), \tag{4.3}$$

and

$$\tilde{ci}^{p}_{\mathcal{E}_i}(t) = \sigma(W_e \cdot \sum_{k\in\mathcal{E}_i} \alpha^{e}_{ij}(t) f^{e,p}_{k}(t)). \tag{4.4}$$

Finally, the embedding result $\tilde{ci}^{p}_{i}(t)$ for node i is the concatenation of the weighted features of neighboring nodes and edges:

$$\tilde{ci}^{p}_{i}(t) = (\tilde{ci}^{p}_{\mathcal{N}_i}(t) \| \tilde{ci}^{p}_{\mathcal{E}_i}(t)) \tag{4.5}$$

4.3.2 Carbon Hypergraph Enhanced TFSM Fine-tuning.

To solve the challenge of learning the complex temporal patterns, especially under insufficient carbon intensity data, we leverage the capability of the TSFM [6], which is a state-of-the-art TSFM with strong zero-shot and few-shot capability developed by AWS. Note that the pre-trained TSFM is a building block for DGCFM and the evolution of TSFM will benefit DGCFM. The existing TSFM is not trained on carbon intensity data, which can lead to suboptimum performance in carbon intensity forecasting. Therefore, we adapt the TSFM to carbon intensity forecasting by fine-tuning it on the carbon intensity dataset. The adaption is not easy since the availability of carbon data has distribution bias and the differences in the energy structure of each region lead to different temporal patterns in each region. Improper fine-tuning can lead to negative transfer.

We propose a metadata-assisted carbon hypergraph to help the fine-tuning process by a two-phased clustering. The function of this carbon hypergraph is to select appropriate carbon data in relevant regions and avoid negative transfer in the fine-tuning process for the target regions, especially the regions with insufficient data.

Metadata-assisted carbon hypergraph generation. Previous studies have shown the effectiveness of metadata to avoid negative transfer [128, 133]. Metadata refers to the descriptive information that characterizes various aspects of a dataset, essentially serving as data about the data itself. Here, the energy structure (e.g., solar, wind, coal, etc.) can be considered as a type of metadata for the carbon dataset of each region. We generate the carbon hypergraph assisted by the metadata as Figure 4.2 shows. The metadata is utilized to infer the edges in the hypergraph, where an edge between a group of source regions and a target region represents a feasible transfer case, and its weight indicates the predicted performance of that transfer.

We use these edges to estimate the globally optimal transfer policy in TSFM fine-tuning. To get the weight of edges, i.e. the correlation of different regions, our idea is to leverage the number of common neighbors in the carbon hypergraph for distance measurement. The hypothesis is that the carbon data of a region should be more similar to another region, if it has higher numbers of shared metadata. We employ the Jaccard Distance[52] to compare the number of shared neighbors between nodes i and j as follows,

$$d_{jd}(i,j) = 1 - |N_i \cap N_j|/|N_i \cup N_j| \tag{4.6}$$

where N_i, N_j represent metadata and neighbors of node i, j. For all possible region pair nodes (i,j), the output is named prior distance matrix as Figure 4.2 shows, which is computed using metadata, i.e., $d_{meta} = [d_{jd}(i,j)]$.

The final step involves inferring correlation mapping of different regions based on an affinity matrix derived from weighted metadata and training samples as follows,

$$d = w_0 d_{meta} + w_1 d_{samp} \tag{4.7}$$

where d_{samp} denotes the distance matrix computed with samples using cosine similarity.

Two-phased clustering. To avoid negative sample transfer in the fine-tuning process, we propose a two-phased clustering, i.e., prior clustering with metadata and posterior clustering with samples. The prior clustering phase establishes clusters based on metadata, which is particularly useful for regions with insufficient carbon data. Subsequently, the posterior clustering phase performs further clustering using historical training samples, building upon the prior clusters, i.e

$$\Psi = c_b(s, d_{samp}|c_a(d_{meta})); \tag{4.8}$$

where Ψ represents the allocation of sample index for n_c clusters, which is defined as the matrix $C_j^k = \{c|c \in \text{cluster } k \text{ for region } j\}$, $k \in [0, n_c], j \in [1, J]$; $c_a(\cdot)$ and $c_b(\cdot)$ denote the function of prior and posterior clustering, respectively. Since page limitations, we concentrate on discussing the clustering with metadata and omit the traditional clustering with samples in the following. For the prior clustering, we propose to employ Normalized Minimum Cut [99] for clustering based on the distance matrix, which can naturally adopt the scenarios when the graph is unconnected.

When we get Ψ, we can fine-tune the TSFM for the target region using the data in the corresponding cluster. We follow the common fine-tuning method, i.e., freeze the TFSM encoder and fine-tune other layers using cross entropy loss. In the inference phase, the TSFM ($TSFM(\cdot)$) takes the historical carbon intensity $\boldsymbol{ci}_i$ as input and outputs the temporal embedding representation $\boldsymbol{ci}_i^f$. Mathematically, we have

$$\boldsymbol{ci}_i^f = TSFM(\boldsymbol{ci}_i) \tag{4.9}$$

4.3.3 Fusion Block

Fusion block next fuse spatial representation and temporal representation into context hidden-states, which is formalized as:

$$Y = Fusion(\tilde{\boldsymbol{ci}}_i^p, \boldsymbol{ci}_i^f | \Theta_{fuse}) \tag{4.10}$$

The $Fusion(\cdot)$ function is trainable with parameters Θ_{fuse}. MLPs are utilized as the fusion structure in this work. The process involves concatenating $\tilde{\boldsymbol{ci}}_i^p$ and $\boldsymbol{ci}_i^f$, and then feeding the resulting vector into MLPs.

4.4 Evaluation

4.4.1 Experimental Setup

Dateset

We evaluate the performance of DGCFM on real-world datasets from 2021 to 2023 containing hourly carbon intensity in 102 regions and partial hourly electricity transmission data. They are public and collected from the European Network of Transmission System Operators for Electricity's (ENTSOE) transparency platform [32] and

ElectricityMaps [73]. The carbon intensity record is formatted as (timestamp, zones, carbon intensity), and each electricity transmission record is formatted as (timestamp, sources, destination, electricity). We use the data in 81 regions to fine-tune the TSFM backbone in DGCFM and the data in 21 regions to construct a carbon network to evaluate the models. The data to fine-tune is mostly estimated data or from third parties. The data to evaluate models is directly from the grid operators.

Training and Testing.

To evaluate the model performance in the scenario where there is no carbon data in the target region (zero-data setting), we train the model on the dataset except the target region to ensure that the model has never seen any data from the target dataset. To evaluate the model performance in the scenario where there is limited carbon data in the target region (low-data setting), we train the model on a small portion (20%) of the target dataset to train the models and test on the remaining (80%) target dataset. To evaluate the accuracy of forecasting carbon intensity, we utilize three common metrics: Mean absolute error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Squared Error (RMSE).

Baselines

We compare our DGCFM model with state-of-the-art baselines in specialized carbon emission forecasting methods: CFCG [131], CarbonCast [71], DACF [72], SFCI [58], TSGP [90]; and state-of-the-art general time forecasting models: TimeXer [118], MOMENT [35], PatchTST [78].

4.4.2 Performance Results

Table 4.1 shows the model performances on 21 real-world testing datasets in the scenario with insufficient data under each setting, where the best-performing model is highlighted in bold font. Note that we do not present the result of CFCG in the zero-data setting as it does not support the settings.

The performance under zero-data setting represents the forecasting capability of models in the regions with no carbon data, which is common in most regions in the world. Overall, DGCFM outperforms existing models by at least 15.23%. More specifically, DGCFM achieves an average improvement of 15.23% compared to PatchTST (from 15.50% to 13.14%), TimeXer 16.15% (from 15.67% to 13.14%), MOMENT 14.79%(from 15.42% to 13.14%), CarbonCast 15.98% (from 15.64% to 13.14%), DACF 25.21% (from 17.57% to 13.14%), TSCP 40.16% (from 21.96% to 13.14%), and SFCI 45.68% (from 24.19% to 13.14%) in MAPE. On average, our DGCFM not only outperforms previous models, but we also observe that DGCFM steadily outperforms other baselines in all datasets, which presents that DGCFM has a powerful generalization capability. The evaluation results present the effectiveness of DGCFM for carbon intensity forecasting in regions with insufficient carbon data.

In operational deployments, if limited carbon data is available in the target region, users can typically leverage it to adapt the model in the target region and enhance the model performance. The low-data setting in Table 4.1 shows the performances of each model in the regions with limited carbon data. Overall, DGCFM outperforms all existing models by at least 20.04%. More specifically, DGCFM achieves an average improvement of 22.04% compared to PatchTST (from 13.93% to 10.86%), TimeXer 21.30% (from 13.80% to 10.86%), MOMENT 20.96% compared to CFCG (from 13.74% to 10.86%), CFCG 21.65% (from 13.86% to 10.86%), CarbonCast 22.70% (from 14.05% to 10.86%), DACF 30.92% (from 15.72% to 10.86%), TSGP 45.26% (from 19.84% to 10.86%), and SFCI 49.65% (from 21.57% to 10.86%) in MAPE. We

Settings	Methods	Metric	Datasets																					
			AT	BE	BG	CH	CZ	ES	FI	FR	GR	HR	HU	IT	LT	NL	NO	PL	PT	RO	RS	SI	SK	AVE
Zero-data	DGCFM	MAE	**12.58**	**5.49**	**22.41**	**7.32**	**37.24**	**12.44**	**9.06**	**1.97**	**39.07**	**9.66**	**15.74**	**39.31**	**58.83**	**5.21**	**1.79**	**68.17**	**19.40**	**18.16**	**16.42**	**12.16**	**29.94**	**21.07**
		RMSE	**16.62**	**6.35**	**26.33**	**9.22**	**45.54**	**17.39**	**10.34**	**2.09**	**45.54**	**12.48**	**18.62**	**43.25**	**61.78**	**6.31**	**2.08**	**80.69**	**27.58**	**20.96**	**21.10**	**15.15**	**35.30**	**24.99**
		MAPE	**11.99**	**12.95**	**6.95**	**32.66**	**8.88**	**14.97**	**14.41**	**16.07**	**11.54**	**5.71**	**9.11**	**14.53**	**21.01**	**15.69**	**7.69**	**10.11**	**27.22**	**6.45**	**3.97**	**8.62**	**15.39**	**13.14**
	PatchTST	MAE	14.16	6.25	26.38	8.70	44.23	15.45	10.28	2.28	47.20	11.06	18.95	44.96	66.20	6.32	2.19	74.58	21.75	21.03	18.74	14.48	36.85	24.38
		RMSE	19.06	7.51	32.60	10.54	51.59	20.06	12.11	2.60	55.38	13.59	20.90	50.83	75.88	7.41	2.39	103.20	32.13	25.74	24.51	16.22	42.09	29.82
		MAPE	14.45	16.13	8.07	37.51	10.49	17.34	16.73	20.06	13.63	6.55	9.88	15.82	25.07	18.01	8.99	12.64	33.23	7.57	4.94	9.22	19.15	15.50
	TimeXer	MAE	14.18	6.26	26.44	8.72	44.32	15.49	10.30	2.28	47.31	11.08	18.99	45.04	66.30	6.34	2.20	74.67	21.78	21.07	18.78	14.52	36.95	24.43
		RMSE	19.00	7.49	32.46	10.51	51.46	20.01	12.07	2.59	55.16	13.56	20.85	50.66	75.57	7.38	2.38	102.70	32.03	25.63	24.44	16.19	41.94	29.72
		MAPE	14.62	16.36	8.14	37.85	10.60	17.51	16.90	20.34	13.78	6.61	9.94	15.91	25.36	18.17	9.09	12.82	33.66	7.65	5.01	9.27	19.42	15.67
	MOMENT	MAE	14.44	6.38	27.10	8.95	45.49	15.99	10.50	2.34	48.66	11.32	19.53	45.98	67.53	6.52	2.26	75.74	22.17	21.55	19.16	14.90	38.10	24.98
		RMSE	18.86	7.42	32.10	10.44	51.11	19.85	11.97	2.56	54.59	13.50	20.72	50.22	74.76	7.32	2.36	101.41	31.77	25.36	24.24	16.13	41.55	29.44
		MAPE	14.37	16.03	8.03	37.35	10.43	17.26	16.66	19.93	13.56	6.52	9.86	15.77	24.94	17.93	8.95	12.56	33.03	7.54	4.91	9.20	19.03	15.42
	CarbonCast	MAE	14.31	6.32	26.77	8.84	44.91	15.74	10.4	2.31	47.99	11.2	19.26	45.51	66.92	6.43	2.23	75.21	21.98	21.31	18.97	14.71	37.53	24.71
		RMSE	19.09	7.53	32.68	10.56	51.67	20.1	12.13	2.61	55.51	13.6	20.93	50.93	76.07	7.42	2.39	103.5	32.19	25.8	24.56	16.23	42.18	29.89
		MAPE	14.59	16.32	8.13	37.79	10.58	17.48	16.87	20.29	13.75	6.6	9.93	15.89	25.31	18.14	9.07	12.79	33.58	7.64	5	9.26	19.37	15.64
	DACF	MAE	16.68	6.45	30.86	9.76	48.85	16.23	11.84	2.44	52.54	13.99	20.33	49.5	75.07	6.77	2.34	89.67	26.56	23.1	20.96	15.46	37.63	27.48
		RMSE	22.75	7.92	33.43	12.15	57.36	23.08	14.19	2.69	59.78	16.02	26.38	60.74	77.47	8.33	2.67	104.55	34.71	26.77	25.77	20.28	46.14	32.53
		MAPE	15.17	17.08	10.05	46.35	11.33	18.54	19.17	20.94	14.84	7.27	12.41	18.39	30.93	20.69	10.34	14.32	34.67	8.72	5.35	11.17	21.16	17.57
	TSGP	MAE	20.65	8.28	35.05	12.43	66.06	21.67	16.34	3.26	67.69	17.65	27.26	71.05	97.68	9.27	2.8	117.39	33.23	32.13	28.13	22.49	52.35	36.33
		RMSE	28.58	10.17	45.12	15.74	87.15	30.79	16.34	3.78	74.18	20	31.39	76.4	104.08	10.99	3.4	135.89	45.26	35.87	35.62	29.03	56.62	42.69
		MAPE	21.77	19.07	11.11	56.03	14.97	25.32	24.51	27.42	19.47	9.53	15.29	24.97	36.12	25.4	11.96	18.69	43.35	10.51	6.99	13.36	25.42	21.96
	SFCI	MAE	24.36	10.11	45.82	13.96	67	22.43	17.16	3.93	77.79	18	30.22	73.22	109.67	9.61	3.41	121.32	35.99	33.83	31.23	23.36	58	39.54
		RMSE	30.96	11.52	46.97	16.68	89.76	33.41	19.68	3.93	86.73	23.98	34.6	81.62	112.82	11.6	3.58	152.66	48.26	41.17	39.75	29.94	66.08	46.94
		MAPE	22.92	23.96	13.61	58.23	17.15	27.23	25.53	28.36	20.01	10.45	16.12	29.93	37.91	31.32	13.82	19.42	48.99	11.81	7.26	16.19	27.87	24.19
Low-data	DGCFM	MAE	**10.49**	**4.57**	**18.67**	**6.10**	**31.03**	**10.37**	**7.55**	**1.64**	**32.56**	**8.05**	**13.11**	**32.76**	**49.02**	**4.34**	**1.50**	**56.81**	**16.17**	**15.14**	**13.68**	**10.13**	**24.95**	**17.18**
		RMSE	**13.85**	**5.29**	**21.94**	**7.68**	**37.95**	**14.49**	**8.62**	**1.74**	**37.95**	**10.40**	**15.52**	**36.04**	**51.48**	**5.26**	**1.74**	**67.24**	**22.99**	**17.47**	**17.58**	**12.62**	**29.41**	**20.39**
		MAPE	**9.99**	**10.79**	**5.79**	**27.22**	**7.40**	**12.48**	**12.01**	**13.39**	**9.62**	**4.76**	**7.59**	**12.11**	**17.50**	**13.08**	**6.41**	**8.43**	**22.68**	**5.38**	**3.31**	**7.19**	**12.83**	**10.86**
	PatchTST	MAE	12.37	5.66	22.34	7.38	37.21	12.36	9.09	1.96	39.57	9.77	15.53	38.87	58.20	5.30	1.82	66.71	18.78	18.60	15.98	12.24	30.39	20.96
		RMSE	16.47	6.38	26.62	9.28	47.08	17.29	10.06	2.09	44.69	12.29	18.61	44.98	62.46	6.25	2.13	79.64	28.26	21.09	20.77	15.03	35.94	25.11
		MAPE	12.97	13.42	7.24	33.93	9.57	15.98	15.34	16.97	12.31	6.05	9.31	15.68	22.17	16.71	8.17	10.51	28.19	6.95	4.23	9.22	16.23	13.86
	TimeXer	MAE	12.62	5.80	22.84	7.56	38.06	12.63	9.30	2.01	40.53	10.00	15.86	39.70	59.46	5.43	1.87	68.06	19.14	19.08	16.30	12.53	31.13	21.42
		RMSE	16.40	6.35	26.49	9.24	46.83	17.21	10.02	2.08	44.51	12.24	18.52	44.73	62.16	6.22	2.12	79.29	28.11	20.99	20.68	14.96	35.75	25.00
		MAPE	13.03	13.48	7.27	34.07	9.62	16.05	15.41	17.05	12.36	6.07	9.35	15.76	22.27	16.78	8.20	10.55	28.30	6.98	4.25	9.26	16.30	13.93
	MOMENT	MAE	12.59	5.79	22.79	7.54	37.97	12.60	9.28	2.00	40.43	9.98	15.83	39.62	59.33	5.41	1.86	67.92	19.10	19.03	16.26	12.50	31.06	21.38
		RMSE	16.32	6.32	26.35	9.19	46.55	17.13	9.97	2.07	44.30	12.18	18.43	44.46	61.82	6.19	2.10	78.92	27.95	20.88	20.58	14.89	35.56	24.86
		MAPE	12.90	13.37	7.21	33.78	9.53	15.90	15.27	16.90	12.25	6.02	9.28	15.61	22.07	16.63	8.13	10.46	28.07	6.91	4.21	9.17	16.15	13.80
	CFCG	MAE	12.51	5.74	22.62	7.48	37.69	12.51	9.21	1.99	40.11	9.9	15.72	39.34	58.91	5.37	1.85	67.47	18.98	18.87	16.16	12.4	30.81	20.74
		RMSE	16.5	6.39	26.67	9.3	47.17	17.32	10.07	2.09	44.76	12.31	18.64	45.07	62.57	6.26	2.13	79.76	28.31	21.13	20.8	15.05	36	24.62
		MAPE	12.84	13.31	7.18	33.64	9.48	15.83	15.2	16.82	12.19	5.99	9.24	15.53	21.97	16.55	8.09	10.42	27.95	6.88	4.19	9.13	16.08	13.74
	CarbonCast	MAE	12.56	5.46	23.46	8.43	37.54	13.25	9.98	2.03	42.19	9.7	15.95	39.92	57.4	5.52	1.86	72.23	20.06	19.04	17.46	14.49	31.28	21.43
		RMSE	17.31	6.4	28.63	9.81	48.99	17.93	10.74	2.22	46.55	13.5	20.45	44.71	61.69	6.63	2.22	80.79	27.8	22.2	21.04	15.37	33.18	25.25
		MAPE	13.44	12.97	7.34	36.24	9.12	15.25	16.28	16.66	11.97	6.24	9.62	15.84	23.64	18.01	8.38	10.61	29.71	6.9	3.93	8.87	15.66	14.05
	DACF	MAE	14.15	6.43	26.46	8.64	45.27	14.58	10.74	2.38	45.38	11.17	17.63	43.25	65.93	5.69	2.27	77.84	23.23	20.65	20.25	14.56	36.15	23.83
		RMSE	20.66	7.79	29.92	10.87	54.54	19.18	11.55	2.38	54.49	14.53	22.54	51.84	71.91	7.07	2.35	91.46	32.23	26.7	25.16	18.2	42.06	28.77
		MAPE	14.85	14.66	7.7	43.43	10.31	16.64	17.05	20.75	12.98	6.56	10.77	18.19	24.68	19.1	9.09	11.56	33.31	7.85	4.28	10.62	18.08	15.72
	TSGP	MAE	20.12	8.21	33.45	10.94	55.05	17.96	13.72	2.9	61.76	14.61	21.79	58.83	94.77	7.83	2.55	111.13	30.33	26.78	24.69	16.86	46.24	31.71
		RMSE	26.44	9.82	38.45	14.26	70.78	27.88	15.45	3.07	70.22	17.61	29.69	67.8	99.08	9.37	3.08	124.06	42.9	33.21	29.45	25.7	51.16	37.92
		MAPE	17.51	18.16	10.75	51.66	13.33	22.75	22.13	26	17.2	8.42	13.84	22.29	33.97	21.85	11.47	15.1	42.08	9.6	5.82	12.9	23.42	19.84
	SFCI	MAE	21.21	9.5	37.49	11.75	66.53	19.78	14.28	2.99	63.64	15.32	28.2	62.41	97.82	8.37	3.28	111.39	32.2	30.92	25.81	20.53	52.21	34.17
		RMSE	26.91	9.91	45.91	15.57	74.2	29.44	18.03	3.49	73.82	21.09	33.81	70.78	107.56	10.73	3.48	129.72	46.86	34.8	37.25	28.22	59.45	41.08
		MAPE	20.95	22.8	11.27	54.24	14.42	24.02	24.3	26.55	18.72	9.54	15.48	24.03	35.19	25.36	11.84	16.76	43.95	10.95	6.4	14.56	27.43	21.57

Table 4.1: Performance comparison of all models in the scenario with insufficient data in terms of MAE, RMSE, and MAPE (%).

Models	DGCFM	PatchTST	TimeXer	MOMENT	CFCG	CarbonCast	DACF	TSGP	SFCI
MAE	**13.75**	15.31	15.33	15.35	15.13	16.89	19.24	24.63	26.87
RMSE	**16.31**	18.39	18.45	18.24	17.95	20.29	23.28	28.82	31.84
MAPE	**8.68**	9.77	9.80	9.62	9.62	10.99	12.09	15.71	17.41

Table 4.2: The full-shot performance.

also observe that DGCFM steadily outperforms other baselines in all datasets. The result demonstrates the effectiveness of DGCFM in the regions with limited carbon data.

To comprehensively evaluate the performance, we also conduct the experiment under a full-shot setting, i.e., 70% data in the target dataset is used to train the models, and 30% data is used for the test. Table 4.2 shows the performances under the full-shot setting. Overall, DGCFM outperforms existing models by at least 9.77%. More specifically, CFCG achieves an average improvement of 11.16% compared to PatchTST (from 9.77% to 8.68%), TimeXer 11.43% (from 9.80% to 8.68%), MOMENT 9.77% (from 9.62% to 8.68%), CFCG 9.78% (from 9.62% to 8.68%), CarbonCast 21.02% (from 10.99% to 8.68%), DACF 28.21% (from 12.09% to 8.68%), TSGP 44.75% (from 15.71% to 8.68%), and SFCI 50.14% (from 17.41% to 8.68%) in MAPE. We also observe that DGCFM can steadily outperform other baselines in all datasets under the full-shot setting.

4.4.3 Explainable Capabilities Learned

The superior performance of DGCFM stems from its dual capacity to capture both spatial and temporal dependencies in carbon data. We demonstrate these learned capabilities through systematic analysis:

Temporal Dependency Analysis. Figure 4.3 illustrates the relationship between forecasting accuracy (MAPE) and renewable energy penetration rates across different regions. Each data point represents a geographical area with distinct renewable

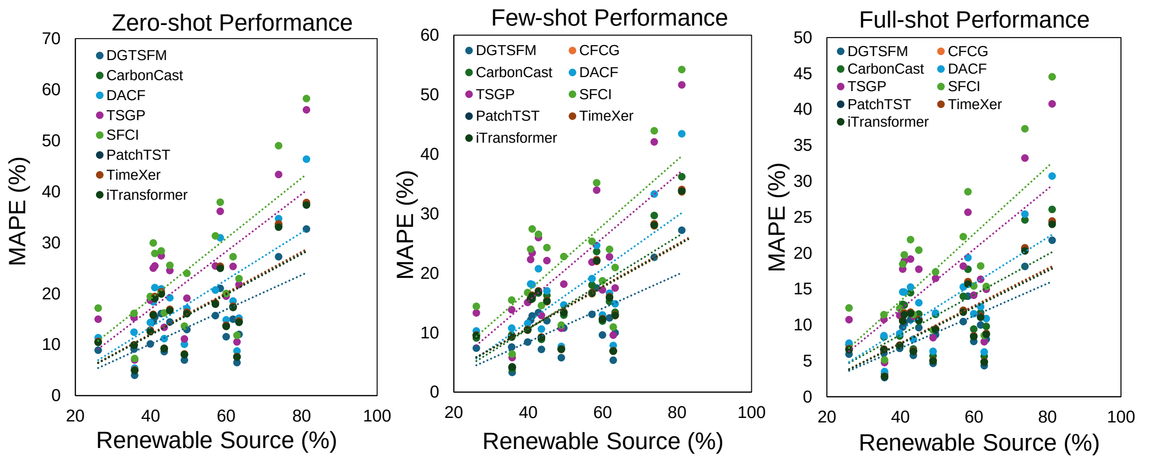


Figure 4.3: The scatter plot of model performance as a function of the ratio of renewable sources.

source compositions (e.g., solar, wind, hydro) in their energy mix. Higher renewable ratios induce greater temporal volatility in carbon intensity due to the intermittent nature of renewable generation. Our analysis reveals a positive correlation between renewable penetration and forecasting error, highlighting the inherent challenges in forecasting carbon patterns with increasing temporal dynamics. Notably, while all models exhibit performance degradation, DGCFM demonstrates (1) absolute superiority over baseline methods and (2)superior resistance to error escalation (slower error growth with renewable ratios increasing). This resilience confirms DGCFM's enhanced temporal modeling capabilities through its TSFM blocks.

Spatial Dependency Analysis. Figure 4.4 examines forecasting performance against electricity exchange ratios, where higher values indicate stronger grid interdependencies between regions. We observe an increase in forecasting errors with rising exchange ratios, reflecting the compounded complexity of spatial interactions in interconnected power systems. We can also observe that our DGCFM model has a slower error increase. This demonstrates the capability of DGCFM to uncover the complex spatial dependency.

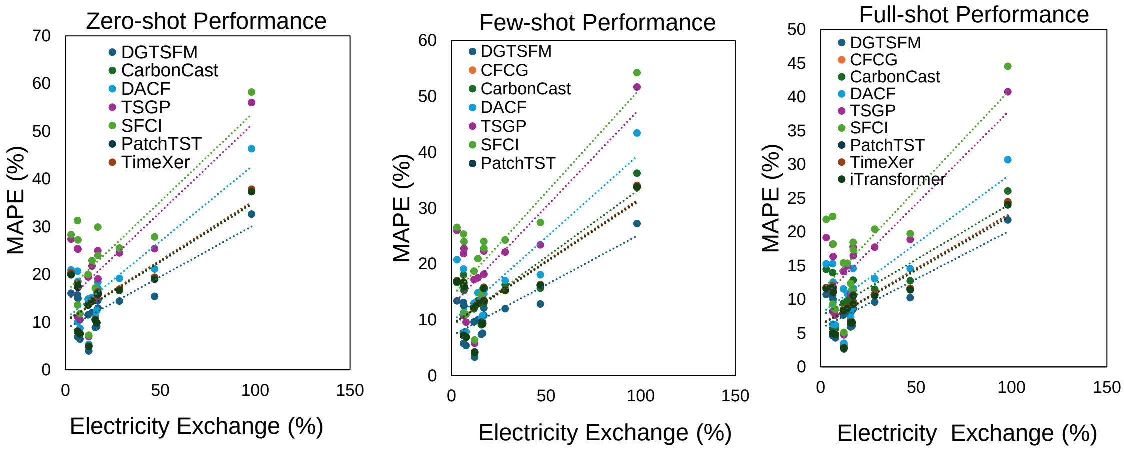


Figure 4.4: The scatter plot of model performance as a function of the ratio of electricity exchange.

4.4.4 Model Ablation Study

To systematically evaluate the contribution of core components in DGCFM, we conduct ablation studies as Table 4.3 shows. We compare the performance of degraded variants against the complete model through three key modifications:

(1) Architecture Simplification: Removing the Physics-guided Heterogeneous Graph Neural Network (PHGNN) component and replacing it with a generic GNN degrades MAPE by 7.18% (from 10.86% to 11.64%), indicating PHGNN's critical role in encoding spatial dependencies through physical constraints.

(2) Temporal Modeling Ablation: Substituting the Time Series Foundation Model (TSFM) backbone with conventional LSTM results in the most significant performance drop (11.70% MAE increase and 19.89% MAPE), confirming the superiority of our TSFM block for capturing complex temporal patterns.

(3) Fine-tuning Strategy Removal: Disabling the carbon hypergraph-enhanced fine-tuning mechanism leads to 11.42% MAPE degradation, demonstrating the importance of our domain-specific adaptation strategy for handling data scarcity in carbon

Models	MAE	RMSE	MAPE
DGCFM	17.18	20.39	10.86
w/o PHGNN	18.11	21.45	11.64
w/o TSFM Backbone	20.19	23.44	13.02
w/o Carbon Hypergraph Fine-tuning	19.07	22.81	12.10

Table 4.3: Results of ablation study under the few-shot setting

intensity forecasting.

The progressive performance deterioration across all metrics (MAE: 17.18 to 20.19, RMSE: 20.39 to 23.44, MAPE: 10.86% to 13.02%) reveals the complementary nature of spatial-temporal modeling and domain adaptation in our framework. Particularly, the TSFM backbone contributes the largest performance gain (19.89% MAPE improvement over the LSTM baseline), suggesting that effective temporal dependency modeling forms the foundation for accurate carbon intensity forecasting. The synergistic combination of physical constraints and hypergraph-based fine-tuning further enhances model robustness in data-constrained scenarios.

4.5 Chapter summary

We present DGCFM, a dual-graph enhanced carbon-domain foundation model that systematically addresses the critical challenge of general carbon intensity forecasting across data-scarce regions. Extensive evaluations across 102 global regions demonstrate 20.04% accuracy improvement over SOTA in low-data scenarios, representing the practical viability for supporting worldwide decarbonization.

Chapter 5

STEC: Spatial-Temporal Embodied Carbon Models with Dual Carbon Attribution for Embodied Carbon Accounting of Computer Systems

5.1 Introduction

In recent years, there has been a growing awareness of the critical importance of sustainability and carbon reduction. Estimates show If current trends continue, information and computing technologies (ICT) could account for up to 8% of global emissions within the next decade [39]. *Embodied carbon* refers to the total amount of carbon emissions generated from the processes associated with a product from cradle to gate [41]. In many industry sectors, embodied carbon constitutes the majority of a product's total carbon footprint, surpassing its *operational carbon* [13]. For instance, the embodied carbon of an iPhone 11 accounts for 79% of its total carbon footprint [39].

Embodied carbon accounting, which involves estimating the embodied carbon of a product, has become an important topic [22]. There are studies on embodied carbon accounting for the computer hardware components such as processors, memory, and storage [107, 39]. The methodology typically utilizes life cycle analysis (LCA) reports [74]. For instance, the Environmental, Social, and Governance (ESG) report of SK Hynix reports the embodied carbon of LPDDR4 memory is 48 g/GB [48].

The embodied carbon of a product is significantly affected by the *carbon intensity* [131] of the electricity consumed during its manufacturing process. Specifically, carbon intensity refers to the amount of carbon emissions per unit of electricity generated; and different energy sources (e.g., oil or wind) can result in different carbon released when a unit of electricity is generated. The carbon intensity of electricity exhibits both spatial and temporal dynamics. The spatial dynamics come from the energy policies of the geographic locations. For instance, the electricity generated in the power grid of Taiwan has a higher carbon intensity than that in Ireland, since Taiwan's energy policy is to rely on brown energy sources due to Taiwan's lack of green energy sources. The temporal dynamics are driven by the environmental dynamics that affect the availability of renewable energy sources. For example, in Ireland, electricity generated during the winter typically has a lower carbon intensity than in the summer, due to the higher availability of wind energy during the winter months.

There are two main approaches both recognized by the Greenhouse Gas (GHG) Protocol [3] to attribute green energy in power grids when a consumer uses electricity: the location-based method and the market-based method. These approaches result in different carbon intensity metrics, referred to as location-based and market-based carbon intensity. The location-based method assumes that the electricity consumed by each user is derived from the overall mix of generation sources available at that location. In the context of location-based attribution, it is not feasible to segregate the exact energy source that generated the electricity consumed by each user. This leads to the assumption that all users in a power grid consume energy in the same proportion as the grid's generation mix. In the context of Market-based attribution, it assumes a consumer can select their electricity generation source (e.g., only use renewable energy) despite the technical impossibility of directing specific electricity generation to specific end-users on the same grid. This attribution is facilitated through market-based accounting mechanisms, enabling organizations to claim renewable energy for their exclusive use through Power Purchase Agreements (PPAs) [70]. As a result, consumers with PPAs will have a carbon intensity reflecting their green energy purchases in addition to any grid usage.

None of the existing studies on embodied carbon accounting has considered the spatial-temporal dynamics and different energy source attribution methods. Existing LCA reports on the embodied carbon of a product represent a *product class* (e.g., 28nm CPU) with the same manufacturing process. However, a *product instance* can be produced in various regions, at different times, and by different manufacturers with/o PPAs, e.g., in the summer in Ireland by Intel or in the winter in Taiwan by TSMC.

In this paper, we present new embodied carbon accounting models that can capture the spatial-temporal dynamics in different granularities and the difference in attribution methods. Specifically, we observe that (1) different carbon attribution (e.g., location-based and market-based) can affect the carbon intensity and thus further affect the embodied carbon accounting.(2) energy policies have granularity at the country-level; at the treaty-zone-level of multiple countries with energy treaties, (e.g., the European Union (EU), Association of Southeast Asian Nations (ASEAN), etc.); and at the global-level and (3) environmental dynamics have granularity at the day-level, season-level, and year-level. Embodied carbon models with varying granularities are valuable for different applications. For instance, manufacturers who regularly purchase computer hardware for their assembly lines may be concerned about embodied carbon at the year-level. In contrast, consumers with more flexible needs may prioritize embodied carbon information at finer granularities, such as the season-level or day-level, particularly those who are environmentally conscious [82].

Existing models can be considered spatial-temporal embodied carbon with location-based attribution (STEC-L) models of a global level and a year level (STEC-L-GY). In this paper, we also study (1)an STEC model of a country-level and a week-level with both location and market-based attribution (STEC-L/M-CW), (2) an STEC model of a country-level and a season-level with both location and market-based attribution (STEC-L/M-CS), and (3) an STEC model of a zone-treaty-level and a year-level with location attribution (STEC-L-ZY). The reason we study these models is the availability of data, and the IC chip manufacturing typically takes weeks. Note that one challenge in developing our models is the need to collect public data, which is dispersed across various reports (e.g., ESG reports, product reports, etc). We have made an effort to collect and organize data.

We compare our models with ACT [39], a state-of-the-art embodied carbon model. We observe significant differences in embodied carbon accounting when the location/market-based attributions and the spatial-temporal factors are taken into consideration and when they are not. First, when location-based attribution is adopted, the differences are significant at any granularity (e.g., 13.69% for 7nm CPU at the country-level and season-level). Second, we find greater dynamics on finer granularity. For instance, the average maximum difference in STEC-L-CW is higher than in STEC-L-ZY (36.34% vs. 19.29%). Additionally, we observe that the embodied carbon in some countries exhibits seasonal patterns and can be significantly influenced by specific weather conditions. When market-based attribution is adopted, the difference may be greater than through location-based attribution (e.g., 37.39% for 7nm Intel CPU by STEC-M-CW vs. 10.38% for 7nm CPU by STEC-L-CW). Last, as time goes by, PPAs gradually increase, and the embodied carbon estimated has a decreasing trend, which uncovers that hardware manufacturers are increasingly purchasing green electricity through PPAs to reduce their carbon footprint.

There are an increasing number of studies on embodied carbon accounting for key computer applications, such as Large Language Model (LLM) training (e.g., LLM-Carbon [33]) and LLM inference (e.g., CarbonMin [23]). We further examine the impact of our STEC models on these embodied carbon accounting schemes. Note that, different applications require different types of supporting hardware. We observe that STEC models affect applications differently compared to other models. For instance, the embodied carbon of LLM inference accounted for by STEC-L-CS ranges from 0.21g to 0.35g, compared to 0.31g by CarbonMin; a difference of 32.26%. This is because CarbonMin requires ordinary hardware with a wide range of manufacturing options across various locations and time periods. In contrast, LLMCarbon has restrictions on its hardware options (e.g., V100 TSMC 12nm), and the difference is only 6.35%.

The core contributions of this paper are summarized as follows:

- We observe that the carbon attribution method and spatial-temporal factors can affect the embodied carbon accounting for computer systems. Existing embodied carbon models have yet to consider these factors; thus, they can have inaccuracies.
- We develop new STEC models based on the ACT model for the embodied carbon of computer components and the electricity carbon intensity model for the spatial-temporal location/market-based carbon intensity in electricity.
- We evaluate STEC at both the hardware level and application levels. Our evaluations demonstrate that there can be non-trivial inaccuracies if the attribution method and spatial-temporal factors are not considered.

The remaining part of the paper is organized as follows. In Section 5.2, we present a framework for STEC models and the details of Location-based STEC models (STEC-L). Section 5.3 presents the spatial-temporal market-based carbon intensity and the details of Market-based STEC models (STEC-M). In Section 5.4, we evaluate our

STEC models, and we use STEC accounting for applications in Section 5.5. Finally, Section 5.7 concludes this paper. Some frequently used notations and definitions are listed in Table 5.2.

5.2 Spatial-Temporal Location-based Embodied Carbon (STEC-L) Models

5.2.1 A Framework for STEC Models

Embodied carbon models can be categorized based on their spatial and temporal granularities. Spatially, they can be classified at the country-level, treaty-zone-level, and global-level. Temporally, they can be classified at the week-level, season-level, and year-level, as the wafer manufacturing typically takes weeks. Further, we can classify embodied carbon models based on location-based (STEC-L) and market-based (STEC-M) carbon attribution methods (refer to Table 5.1).

The key difficulty is to collect data on hardware-related parameters, the PPAs of manufacturers, and renewable energy sources at different granularities. Our efforts have been focused on collecting data for the study of STEC-L/M-CW/CS and STEC-L-ZY; STEC-L/M-CY can be derived from STEC-L/M-CW. The current data cannot support us in studying other, and the operational scope of STEC is confined to regions with carbon intensity data records. Specifically, daily granular carbon data can be averaged to obtain weekly and monthly data. But for areas with only coarse-grained data (e.g., yearly), STEC can not estimate to obtain fine-grained data (e.g., monthly, weekly). For clarity, we summarize key acronyms in this paper in Table 5.2.

Table 5.1: Classification of embodied carbon models with different carbon attribution methods based on different spatial and temporal granularity

Spatial \ Temporal	Week-level	Season-level	Year-level
Country-level	STEC-L/M-CW	STEC-L/M-CS	-
Treaty-zone-level	×	×	STEC-L-ZY
Global-level	×	×	[39][54][107]

5.2.2 Spatial-Temporal Location-based Carbon Intensity

Firstly, we present the state-of-the-art embodied carbon accounting model for computer systems, the Architectural CO_2 Tool (ACT) Model [39], which is designed for three kinds of fundamental computer hardware: memory (e.g., DRAM), and storage (e.g., SSD, HDD), and processors (e.g., CPU, GPU). For memory and storage, ACT estimates the embodied carbon by directly applying the embodied carbon values from a corporation's annual ESG or LCA report as Eq. 5.1 and 5.2 shows.

$$EC_M^{ACT} = LCA\_ESG\_Report(Memory) \tag{5.1}$$

$$EC_S^{ACT} = LCA\_ESG\_Report(Storage) \tag{5.2}$$

To estimate the embodied carbon of processors, ACT models the carbon emitted in the manufacturing process as three components as Eq. 5.3 shows: (1) the carbon emissions per unit die size from chemicals gas burned (e.g., fluorinated compounds) in the manufacturing process, denoted as GPS (gas per size); (2) the carbon released per unit die size from up-stream raw materials, denoted as MPS (material per size); and (3) the carbon released per unit die size by the electricity consumption during manufacturing. The latter is calculated using the electricity consumption per unit of size, denoted as EPS, and the carbon intensity of electricity, denoted by CI. CI

Table 5.2: The summary of acronyms.

Acronyms	Descriptions
GPS	Gas per size (in $g/cm^2, g/GB$)
EPS	Energy per size (in $kWh/cm^2, kWh/GB$)
MPS	Material per size (in kWh/cm^2)
$E(s,t)$	Electricity generated by a power grid in location s at time t
$E^k(s,t)$	Electricity generated by source k in location s at time t
E^k_{ppa}	Electricity generated by source k contracted through PPAs
E_{ppa}	Electricity contracted through PPAs
CI_l	Location-based carbon intensity (in g/kWh)
CI_m	Market-based carbon intensity (in g/kWh)
CI_{res}	Residual carbon intensity
BD	Bit intensity (in GB/cm^2)
EPG	Energy per GB (in kWh/GB)
ef	Emission factor (in g/kWh)
ef^k	Emission factor of energy source k (in g/kWh)
f_{ppa}	Fraction of electricity consumption met by PPAs
α_M	Carbon emissions independent of electricity in memory manufacturing (in g/GB)
α_s	Carbon emissions independent of electricity in storage manufacturing (in g/GB)
$EC_P^{l/m}(s,t)$	Embodied carbon of processors with location/market-based attribution
$EC_M^{l/m}$	Embodied carbon of memory with location/market-based attribution
$EC_S^{l/m}$	Embodied carbon of storage with location/market-based attribution
STEC	Spatial-Temporal Embodied Carbon
STEC-L-CW	Spatial-Temporal Location-based Embodied Carbon in Country and Week level
STEC-L-CS	Spatial-Temporal Location-based Embodied Carbon in Country and Season level
STEC-L-GY	Spatial-Temporal Location-based Embodied Carbon in Gobal and Year level
STEC-L-ZY	Spatial-Temporal Location-based Embodied Carbon in Zone and Year level
STEC-M-CW	Spatial-Temporal Market-based Embodied Carbon in Country and Week level
STEC-M-CS	Spatial-Temporal Market-based Embodied Carbon in Country and Season level

represents the amount of carbon emissions when generating a unit of electricity, and it can be obtained or calculated based on the public reports on power grids [73, 111]. The parameters GPA and EPA are available from a related research paper [12], while MPA data can be obtained from industrial reports [16].

$$EC_P^{ACT} = EPS \times CI + GPS + MPS \tag{5.3}$$

The key to the development of an STEC-L model is to study the three carbon emission components during manufacturing across the temporal-spatial dimensions. For GPS and MPS, since they are less influenced by renewable energy, we conjecture that they are less dynamic across the spatial and temporal dimensions, although we admit that further investigation should be carried out.

The electricity component significantly contributes to carbon emissions. For instance, for a 7nm CPU, the carbon emitted contributions are EPS 61%, GPS 11%, and MPS 28%. The carbon emissions from electricity depend on the energy sources used for electricity generation (e.g., solar or coal) and the availability of renewable energy sources (e.g., solar). The former varies across different locations, while the latter varies with weather and seasons. Previous studies have explored spatial and temporal variations in carbon emissions from electricity, and we have adopted the model presented in [71].

The carbon intensity $CI(s,t)$ in STEC models is calculated based on a specific hardware manufacturing time t and location s. Let $\mathcal{E}$ represent the set of energy sources, and $E^k(s,t)$ denotes the electricity generated by energy source k at time t and location s. It follows that $E(s,t) = \sum_{k \in \mathcal{E}} E^k(s,t)$. Each type of energy source k in electricity generation has a distinct carbon emission factor, denoted as ef^k. Table **??** shows the carbon emission factors. Finally, the carbon intensity is the ratio of the total amount of carbon emissions against the total amount of electricity generation ($E(s,t)$). Here, any renewable investments (e.g., PPAs) are not considered; hence, the carbon intensity of electricity in this method is a location-based attribution, which is denoted as location-based carbon intensity ($CI_l(s,t)$).

$$CI_l(s,t) = \frac{\sum_{k \in \mathcal{E}} ef^k \times E^k(s,t)}{E(s,t)} \tag{5.4}$$

5.2.3 Spatial-Temporal Location-based Embodied Carbon Models

We now present our models, which aim to capture the spatial and temporal dynamics in the location-based carbon intensity $CI_l(s,t)$. The embodied carbon can be estimated using location-based $CI_l(s,t)$ with the electricity consumption in manufacturing. To develop our spatial and temporal embodied carbon models, we need electricity consumption data; which is difficult to obtain in these annual ESG reports or product reports on memory and storage. Therefore, we first reversely calculate the electricity consumption based on the information on the average embodied carbon and carbon intensity.

We first establish basic models for the memory ($EC^l_M(s,t)$, storage ($EC^l_S(s,t)$), and processor ($EC^l_P(s,t)$). The embodied carbon is calculated based on the per-unit size of the respective computer hardware components; specifically, the per-unit die size (in cm^2) for processors and the per-unit storage size (in Gigabytes/GB) for storage and memory.

The following illustrates the three basic models:

Processor: The spatial-temporal location-based embodied carbon of processor, denoted as $EM^l_P(s,t)$, comprises three components as presented in Eq. 5.12: (1) the carbon emissions from the electricity consumption in manufacturing, calculated by multiplying (EPS) by $CI_l(s,t)$; (2) the carbon emissions from raw materials (MPS); and (3) the carbon emissions from chemical gas (GPS). The parameters data we collected are shown in Table 5.5.

$$EC_P^l(s,t) = CI_l(s,t) \times EPS + MPS + GPS \tag{5.5}$$

Memory: The spatial-temporal location-based embodied carbon of the memory ($EC_M^l(s,t)$) comprises two components: (1) the carbon emissions from the electricity consumption in the process of manufacturing the memory. This is calculated by multiplying the electricity consumed per unit of size (EPS) by the location-based carbon intensity of electricity ($CI_l(s,t)$); and (2) the carbon emissions independent of electricity (e.g., raw materials and distribution), denoted as α_M. The calculation of α_M involves annual EC_M^l, EPS, and BD (bit density), along with the yearly location-based carbon intensity CI_l, all of which can be found from the reports. This relationship is illustrated in Eq. 5.6. The parameters data we collected are shown in Table 5.4.

$$\alpha_M = EC_M^l - CI_l \times EPS \div BD \tag{5.6}$$

As such, $EC_M^l(s,t)$ is calculated using Eq. 5.13.

$$EC_M^l(s,t) = CI_l(s,t) \times EPS \div BD + \alpha_M \tag{5.7}$$

Storage: The spatial-temporal location-based embodied carbon of the storage, denoted as $EC_S^l(s,t)$, also comprises two components: (1) the carbon emitted in the process of manufacturing the storage, determined by multiplying the electricity consumption in the manufacturing process (EPG) by the location-based carbon intensity $CI_l(s,t)$; and (2) the carbon emissions independent of electricity (e.g., raw materials and distribution), denoted as α_S. α_S can be found in the industry reports. EPG can be calculated through the average EC_S^l in the reports and the average location-based carbon intensity CI_l, which is shown in Eq. 5.8. The parameters data we collected

Table 5.3: The summary of the parameters of storage from product reports [95]

Types	Technology	Embodied Carbon (g/GB)	Manufacturing Energy Carbon (g/GB)	Other Carbon (g/GB)
Enterprise SSD	Nytro 3530	6.27	4.25	2.02
	Nytro 1551	3.91	1.53	2.38
	Nytro 3331	5.48	0.92	4.56
	Nytro 3332	2.42	0.78	1.64
Consumer SSD	BarraCuda 120	26.28	23.85	2.43
Enterprise HDD	EXOS X20	0.88	0.36	0.52
	EXOS X18	0.88	0.39	0.49
	Exos 2X14	1.28	0.51	0.78
	Exos 7E8	5.28	2.34	2.94
	Exos 5E8	2.54	1.14	1.40
	Exos 10E2400	10.75	6.94	3.81
	EXOS 15E900	21.62	10.65	10.97
	Exos X16	1.46	0.77	0.69
	Exos X12	1.32	0.53	0.79
Consumer HDD	BarraCuda 3.5	9.40	4.84	4.56
	BarraCuda	4.25	2.08	2.17
	BarraCuda Pro	2.62	1.22	1.40
	FireCuda	5.16	3.81	1.35
	IronWolf	5.28	2.22	3.06
	IronWolf Pro	3.80	1.33	2.47
	Skyhawk 3 TB	9.85	2.17	7.68
	Skyhawk Surveillance HDD	4.37	1.54	2.83
	Skyhawk 6 TB	4.18	1.09	3.09
	Video 3.5 HDD	8.20	3.22	4.98
	Video 3.5 HDD (Pipeline HDD)	9.54	3.23	6.31
External HDD	ULTRA TOUCH	5.54	3.40	2.13
	Rugged Mini	4.22	2.98	1.25

are shown in Table 5.3.

$$EPG = (EC_S^l - \alpha_S)/CI_l \tag{5.8}$$

As such, $EC_S^l(s,t)$ can be estimated using Eq. 5.14.

$$EC_S^l(s,t) = CI_l(s,t) \times EPG + \alpha_S \tag{5.9}$$

5.3 Spatial-Temporal Market-based Embodied Carbon (STEC-M) Models

5.3.1 Spatial-Temporal Market-based Carbon Intensity

If some renewable energy sources are contracted out through power purchase agreements (PPAs), the remaining source mix in the grid is referred to as the residual grid mix, and the associated carbon intensity is termed residual carbon intensity (CI_{res}) [70]. In a grid, let $E^k(s,t)$ be the total electricity generated by an energy source k (e.g., solar), out of which E^k_{ppa} denotes the electricity contracted through PPAs. Let $E(s,t)$ be the total electricity generated in that grid, and E_{ppa} be the total electricity under PPAs, then, $\sum E^k_{ppa} = E_{ppa}$. In the market-based method, entities with PPAs claim the environmental friendliness of their invested electricity. Consequently, all the contracted renewable energy is excluded before calculating CI_{res}. Then,

$$CI_{res}(s,t) = \frac{\sum_{k \in \mathcal{E}} ef^k \times (E^k(s,t) - E^k_{ppa})}{E(s,t) - E_{ppa}} \tag{5.10}$$

It follows that the carbon intensity of a manufacturer using the market-based method (CI_m) in that power grid can be calculated using

$$CI_m(s,t) = (1 - f_{ppa}) \times CI_{res}(s,t) \tag{5.11}$$

where $f_{ppa} \in [0,1]$ is the fraction of electricity consumption of a consumer met by PPAs. In contrary to the location-based approach, the market-based carbon intensity fluctuates among consumers within the same grid. For consumers without PPAs ($f_{ppa} = 0$), $CI_m = CI_{res}$, while for consumers capable of fulfilling their total electricity demand through PPAs, $CI_m = 0$. The market-based approach consistently favors the investors of renewable energy over non-investors.

5.3.2 Spatial-Temporal Market-based Embodied Carbon Models

Hardware manufacturers can buy renewable energy to reduce their carbon emissions through PPAs (e.g., 9.2% electricity consumption of TSWC is met by PPAs in 2021 [112]). Here, we develop the spatial-temporal market-based embodied carbon (STEC-M) models to capture the effect of PPAs on carbon accounting. Subsequently, we develop STEC-L/M-CW, STEC-L/M-CS, and STEC-L-ZY, in which we differentiate between different granularities on location s (e.g., country-level, treaty-zone-level of multiple countries) and time t (e.g., day, season). The embodied carbon of the processor ($EC_P^m(s,t)$), memory ($EC_M^m(s,t)$), and storage ($EC_S^m(s,t)$) manufactured by the manufacturers with PPAs can be estimated as follows:

Processor:

$$EC_P^m(s,t) = CI_m(s,t) \times EPS + GPS + MPS \tag{5.12}$$

where $CI_m(s,t)$ is the spatial-temporal market-based carbon intensity; EPS is the electricity consumption per unit of die size of chips in manufacturing; GPS is the carbon emissions from chemical gas in manufacturing; and MPS is the carbon emissions from raw material.

Memory:

$$EC_M^m(s,t) = CI_m(s,t) \times EPS \div BD + \alpha_M \tag{5.13}$$

where BD is the bit intensity (in GB/cm^2); and α_M is the carbon emissions independent of electricity in the memory manufacturing (e.g., raw materials and distribution).

Storage:

$$EC_S^m(s,t) = CI_m(s,t) \times EPG + \alpha_S \tag{5.14}$$

where EPG is the electricity consumption per unit of storage in manufacturing (in

Table 5.4: The embodied carbon parameters about energy per size (EPS), gas per size (GPS), and material per size (MPS) for the processors manufacturing process of 28nm to 3nm process nodes.

Process Node	EPS (kWh/cm2) (Data Source: Research Paper [12])	GPS (g/cm2) (Data Source: Research Paper [12])	MPS (g/cm2) (Data Source: Environmental Report [16])
28	0.9	100	500
20	1.2	110	500
14	1.2	125	500
10	1.475	150	500
7	1.52	200	500
7-EUV	2.15	200	500
7-EUV-DP	2.15	200	500
5nm	2.75	225	500
3nm	2.75	275	500

kWh/GB); and α_S is the carbon emissions independent of electricity in the storage manufacturing.

Tables 5.4, 5.5, and 5.3 summarize the related embodied carbon data for processors, memory, and storage, respectively.

5.3.3 STEC-L/M-CW, STEC-L/M-CS, and STEC-L-ZY

In the STEC-L/M-CW model, time t is defined as $t \in \{week_i, \forall i \in [1, 52]\ \}$, location $s \in \{country\}$. $\{country\}$ is the set of all of the hardware production places, e.g., Ireland, China, USA, etc.

In the STEC-L/M-CS model, time t is defined as $t \in \{spring, summer, fall, winter\}$, location $s \in \{country\}$. $\{country\}$ is the set of all of the hardware production places.

In the STEC-L-ZY model, time t is defined as $t \in \{year\}$, location $s \in \{zone\}$. $\{year\}$ is the set of production years. $\{zone\}$ is the set of the treaty-zone-level of multiple

Table 5.5: The embodied carbon parameters about total embodied carbon, bit density, and carbon emissions from electricity consumption for the memory manufacturing process, from 30nm LPDDR3 to LPDDR4 technology. [24, 39].

Technology	Embodied Carbon (g/GB)	Bit Density (G/mm2)	Carbon Emissions from Electricity Consumption (g/GB)
30nm LPDDR3	230	0.06	67.50
20nm LPDDR3	184	0.11	51.43
10nm DDR4	65	0.19	35.74
LPDDR4	48	0.17	39.04

countries with energy treaties, such as the EU, ASEAN, and others. Intrinsically, ACT can be considered as an STEC-L-GY model.

5.4 Evaluation

In this section, we compare STEC models with the state-of-the-art model (the ACT model), and we compare them in terms of three types of hardware: CPU, Memory, and Storage.

Table 5.6 provides the data sources we collected for the STEC models. The data sources are publicly available. Specifically, Table 5.6(a) outlines the sources for electricity data, including the carbon intensity (yearly and monthly) and energy sources (hourly); on the other hand, Table 5.6(b) details the sources for the hardware-related parameters. From a temporal perspective, the data is available at yearly-level, monthly-level, and daily-level. Spatially, the data covers two zones (EU and ASEAN) and individual regions, with a focus on regions with a semiconductor industry. The selected regions are determined based on the scale of annual semiconductor production, considering the total production of leading semiconductor companies in each region.

Table 5.6: The Data Sources for the STEC models

(a) Data Sources for the Carbon Emissions of Electricity Energy

	EU	ASEAN	China	Taiwan	South Korea	USA	Ireland	Italy	Source
Yearly [2019-2022]	✓	✓	✓	✓	✓	✓	✓	✓	Our World in Data [50]
Monthly [2019-2022]	-	-	✓	✓	✓	✓	✓	✓	EMBIR [29]
Daily [2021]	-	-	-	✓	✓	✓	✓	✓	ENTSOE [?] ElectricityMaps [73]

(b) Data Sources for Other Types of Carbon Emission

Parameter	Description	Unit	Source
EPS	Electricity consumed per die Size	kWh/cm2	Research paper [12]
GPS	Carbon emission from Gas used per die Size	g/cm2	Research paper [12]
MPS	Carbon emission from Material used per die Size	g/cm2	Industrial research reports [16]
BD	Bit density for memory	GB/cm2	Industrial research reports [24]
EPG	Electricity consumed per GB	kWh/GB	Industrial environmental reports of manufacturers [48, 98]
PPAs	Power purchase agreements	%	ESG reports [112, 51]

According to the rankings in [124], the top companies are Samsung, Intel, TSMC, SK Hynix, Micron, and Seagate, leading to the selection of Taiwan, the USA, China, South Korea, Ireland, Italy as the regions of interest. Data for China were obtained from EMBIR, while ElectricityMaps provided data for Taiwan, the USA, South Korea, Ireland, and Italy. The granularity of data in ElectricityMaps is daily, whereas in EMBIR, it is monthly. The evaluation of our STEC-L/M-CW model is conducted using five regions Taiwan, USA, South Korea, Ireland, and Italy, and STEC-L/M-CS is evaluated using six regions Taiwan, USA, China, South Korea, Ireland, Italy.

5.4.1 Evaluation of STEC-L

Evaluation of STEC-L-CW

Table 5.7 provides a comparison between STEC-L-CW and ACT (baseline), revealing the following findings. (1) The difference in performance between STEC-L-CW and ACT is significant. Specifically, the average difference is 10.38%, and the average maximum difference can reach up to 36.34%. (2) The SSD [97] exhibits a greater difference than the HDD [96] (12.61% vs. 8.13%). This is attributed to the more

Table 5.7: The comparison between STEC-L-CW and ACT on CPU (7nm), memory (10nm DDR4), SSD [97], and HDD [96].

Hardware	Ave. Difference (%)	Max. Difference (%)
CPU	10.77	37.69
SSD	12.61	44.14
HDD	8.13	28.46
Memory	10.01	35.06
Average	10.38	36.34

advanced manufacturing process of SSD compared to HDD, resulting in higher electricity consumption and greater dynamics. Consequently, advanced processes amplify the differences in embodied carbon. To investigate the impact of advanced processes on embodied carbon, Fig. 5.2 illustrates the embodied carbon of CPUs across technology nodes from 28nm to 3nm, considering different spatial information. The top and middle figures demonstrate that the electricity consumption per unit size (EPS in kWh/cm^2) and carbon emissions from chemical gas during manufacturing increase but do not vary according to location, as previously discussed. The bottom figure shows the embodied carbon of CPUs across technology nodes, assuming manufacturing in five major IC-producing regions. It is evident that more advanced technology nodes further exacerbate the disparities in embodied carbon. (3) The dynamics of embodied carbon can be affected by the proportion of variable renewable energy sources (e.g., solar, wind, etc) in electricity. For instance, there is a higher percentage of variable renewable energy sources in the power grids of Ireland (31.45%) compared to the power grids of Taiwan (6.44%). This results in greater dynamics for the embodied carbon of memory manufactured in Ireland (with a variance of 32.99) than that in Taiwan (with a variance of 2.57), as shown in Fig. 5.1.

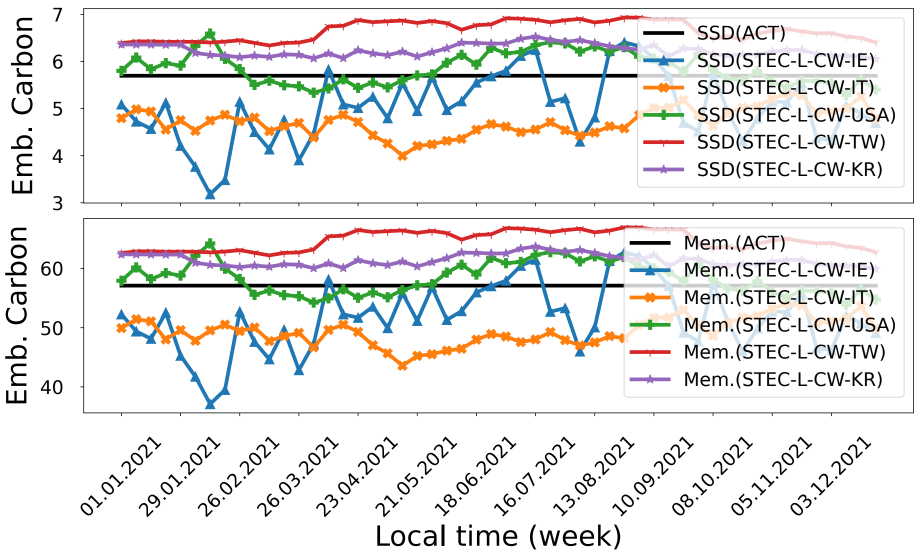


Figure 5.1: The comparison between STEC-L-CW and ACT on SSD [97] and memory (10nm DDR4).

Evaluation of STEC-L-CS

Table 5.8 presents a comparison between STEC-L-CS and ACT (baseline). Additionally, Fig. 5.3 visually shows the embodied carbon (g/cm^2) of the CPU estimated by the STEC-L-CS and ACT in Ireland and Italy from 2019 to 2022. Our findings are as follows: (1) The difference is significant, with an average difference of 13.33% and an average maximum difference of 27.50%. For each type of hardware component, the average difference is 13.69% (CPU), 16.51% (SSD), 10.36% (HDD), and 12.74% (Memory). (2) The embodied carbon exhibits seasonal patterns in some regions. As illustrated in Figure 5.3, the embodied carbon is higher in summer and lower in winter in Ireland, while, the opposite trend is observed in Italy. This pattern is influenced by the climate and on the composition of energy sources in the power grids of a region. Specifically, wind power dominates in the green energy sources in Ireland, while, solar power dominates in the green energy sources in Italy. Furthermore, solar is abundant in the summer in Italy, while wind is abundant in the winter in Ireland. These factors contribute to the differing patterns observed in the two countries.

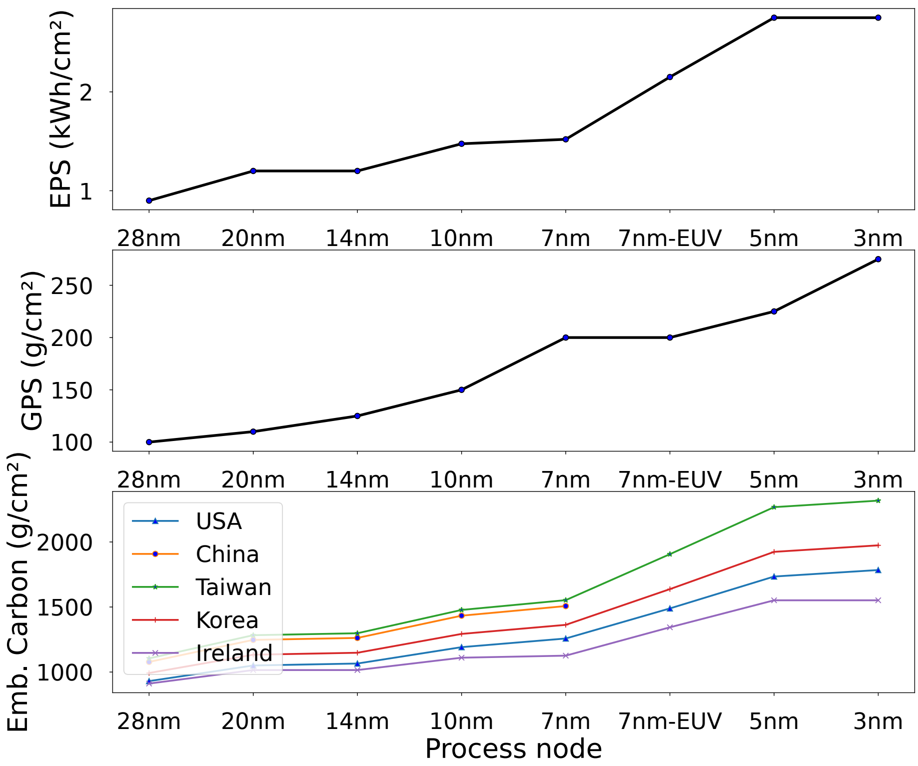


Figure 5.2: Embodied carbon of CPUs varying from IC manufacturing technology nodes and production locations. Carbon footprint breakdown into electricity consumed in manufacturing (top), emissions from chemical gases (middle), and aggregate carbon per size in varying manufacturing locations (bottom).

Table 5.8: The comparison between STEC-L-CS and ACT on CPU (7nm), memory (10nm DDR4), SSD [97], and HDD [96].

Hardware	Ave. Difference (%)	Max. Difference (%)
CPU	13.69	27.32
SSD	16.51	34.46
HDD	10.36	21.63
Memory	12.74	26.60
Average	13.33	27.50

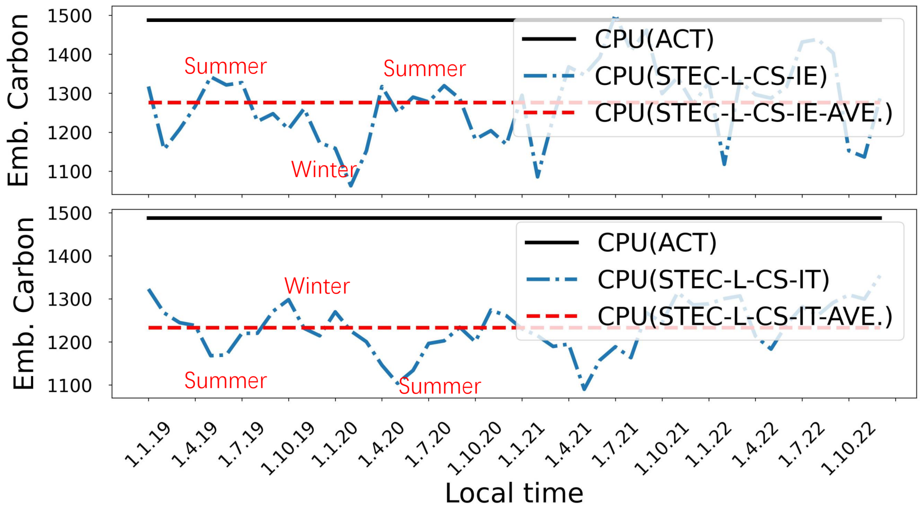


Figure 5.3: The comparison between STEC-L-CS and ACT. The embodied carbon is higher in summer and lower in winter in Ireland, while, Italy shows the opposite trend. The seasonal pattern captured by STEC-L-CS is influenced by the climate and the composition of energy sources in the power grids of a region.

Evaluation of STEC-L-ZY

Table 5.9 compares STEC-L-ZY and ACT (baseline) in the two treaty zones (ASEAN and EU). Overall, the difference is 18.00%, with an average maximum difference of 19.29%. Our findings are as follows: (1) the STEC-L-CW/CS/ZY models all have significant differences compared to the ACT model. (2) Finer granularity can introduce greater dynamics. For instance, the average maximum difference between STEC-L-CW and ACT exceeds that between STEC-L-CS and ACT, STEC-L-ZY and ACT (36.34% compared to 27.50%, 36.34% compared to 19.29%).

Summary

Overall, the difference between STEC-L and ACT comes from the spatial-temporal characteristics of carbon intensity. The difference is affected by (1) Manufacturing processes: advanced technology nodes (e.g., SSD vs. HDD, CPUs from 28nm to

Table 5.9: The comparison between STEC-L-ZY and ACT on CPU (7nm), memory (10nm DDR4), SSD [97], and HDD [96].

Hardware	STEC-L-ZY		ACT	Ave.	Max.
	Emb.carbon in EU	Emb.carbon in ASEAN	Emb.carbon	difference (%)	difference (%)
CPU	1266.88	1848.10	1557.49	18.65	19.99
SSD	4.77	7.41	6.09	21.64	23.19
HDD	4.36	5.81	5.08	14.28	15.3
Memory	70.72	49.72	60.22	17.42	18.67
Average	/	/	/	18.00	19.29

3nm) amplify embodied carbon differences due to higher electricity consumption; (2) Regional energy mix dynamics: regions with higher variable renewable energy (e.g., Ireland: 31.45% wind/solar vs. Taiwan: 6.44%) exhibit greater temporal variance in embodied carbon; (3) Climate-driven seasonal trends: higher embodied carbon in summer (wind scarcity) and lower in winter (wind abundance) in Ireland, while opposite pattern due to solar energy dominance in Italy (higher in summer, lower in winter); (4) Spatial-temporal granularity: finer spatial-temporal granularity (e.g., STEC-L-CW) increases dynamics, leading to larger gaps (max difference up to 36.34%).

5.4.2 Evaluation of STEC-M

Currently, the data about the residual mix of electricity in power grids typically may not be available in each region, which is potentially happening in practice. Here, we assume the PPAs in the power grids are small and set CI_{res} equal to CI_l [70] for the evaluation in the following Section 5.4.2 and 5.4.2 and then further discuss the effect of CI_{res} by simulation in Section 5.4.2.

Evaluation of STEC-M-CW

Table 5.10 presents a comparison between STEC-M-CW and ACT on 7nm CPU, 10nm DDR4 memory, SSD[97], and HDD [96] in 2021. We can find the following: (1) The difference in performance between STEC-M-CW and ACT is significant. Specifically, the average difference is 9.59% (TSMC), 37.39% (Intel), and the average maximum difference can reach up to 37.64% (TSMC), 47.70% (Intel). (2) Intel exhibits a greater difference than the TSMC (37.39% vs. 9.59%). This is attributed to the proportion of electricity consumption met by PPAs of Intel is greater than that of TSMC (80% vs. 9.2% in 2021). It shows that our STEC-M-CW can both capture the difference between carbon attribution methods and spatial-temporal information.

Evaluation of STEC-M-CS

Fig 5.4 presents a comparison between STEC-M-CS and ACT on 7nm CPUs manufactured by TSMC, and the corresponding PPA change over time (from 2019 to 2021). Our findings are as follows: (1) As time goes by, PPAs gradually increase, and the embodied carbon estimated by STEC-M-CS has a decreasing trend. It uncovers that hardware manufacturers are increasingly purchasing green electricity through PPAs, which help their product's embodied carbon reduction. (2) ACT does not have the ability to capture the effect of PPAs on embodied carbon accounting, while STEC-M-CS captures both PPAs and spatial-temporal information.

The Effect of Residual Carbon Intensity

The difference between location-based carbon intensity CI_l and residual carbon intensity CI_{res} can be significant in some regions. For example, Norway's electricity mix consists of 99% renewable energy sources. There, location-based carbon intensity CI_l in Norway is nearly zero. However, since renewable energy producers in Norway

Table 5.10: The comparison between STEC-M-CW and ACT on CPU (7nm), memory (10nm DDR4), SSD [97], and HDD [96].

Hardware	Ave. Difference (%)		Max. Difference (%)	
	TSMC	Intel	TSMC	Intel
CPU	9.93	41.94	39.05	49.48
SSD	11.70	49.10	45.72	57.94
HDD	7.50	31.65	29.48	37.35
Memory	9.24	39.01	36.32	46.03
Average	9.59	37.39	37.64	47.70

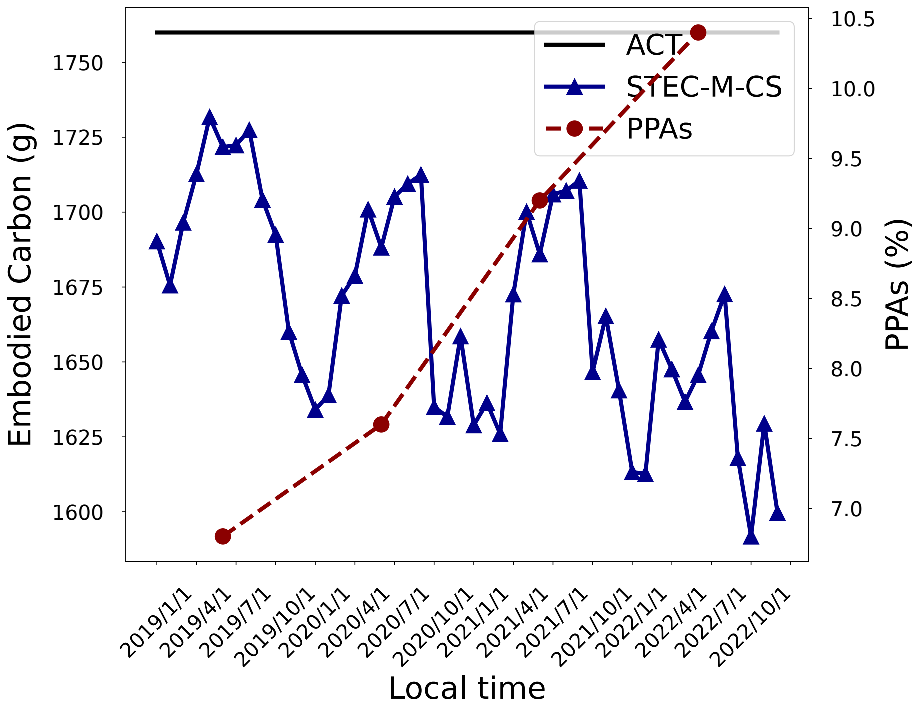


Figure 5.4: The comparison between STEC-M-CS and ACT on the CPUs. STEC-M-CS captures both PPAs and spatial-temporal information. PPAs increase over time, leading to embodied carbon showing a decreasing trend.

export electricity to consumers in other countries, the residual mix within Norway drops to just 15% renewable sources [70]. This leads to a significantly higher residual carbon intensity CI_{res}. Currently, information about the residual mix of electricity in power grids typically may not be available in each region, which is potentially happening in practice. In this section, we show the case for different proportions of renewable energy contracted out, resulting in the discrepancy between location-based and market-based evaluations.

Fig. 5.5 shows the residual carbon intensity CI_{res} increases with the proportion of contracted renewable energy through PPAs. The low-carbon regions can become browner as more and more renewable energy is contracted out. Fig. 5.6 and Fig. 5.7 present the dynamics of total carbon intensity CI_l and the residual carbon intensity CI_{res} with all renewables excluded through PPAs (100% PPAs) , respectively. We can find as follows: (1) overall, the difference between CI_l and CI_{res} is significant; (2) the dynamics of CI_{res} become less as all renewables are contracted through PPAs.

The changes of residual carbon intensity CI_{res} can further affect the embodied carbon accounting for the products that are made by the manufacturers that consume the electricity in the power grid. Fig. 5.8 shows the embodied carbon of 7nm CPUs that are made using electricity under different proportions of PPAs(%) contracted. More renewables excluded through PPAs can result in higher embodied carbon (the green line vs. the black line). Within the same power grid, i.g. the same proportions of renewables excluded through PPAs in CI_{res}, the embodied carbon of 7nm CPUs that are made by the manufacturers with their own PPAs (e.g., Intel with 80% PPAs) in CI_m are smaller (the blue line vs. the red line).

Summary

The difference between STEC-M and ACT is caused by (1) Power Purchase Agreements (PPAs): STEC-M reveals substantial differences due to varying PPA adoption

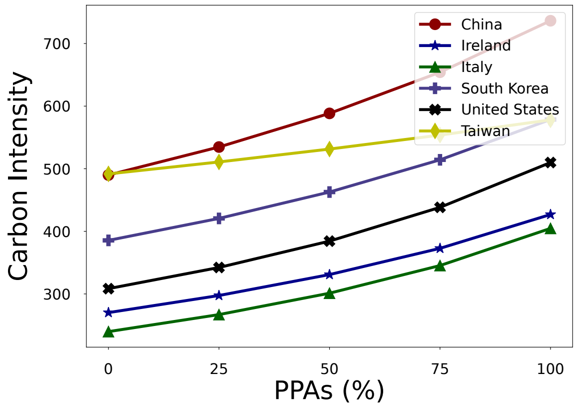


Figure 5.5: Residual carbon intensity over PPAs (%) in 2021

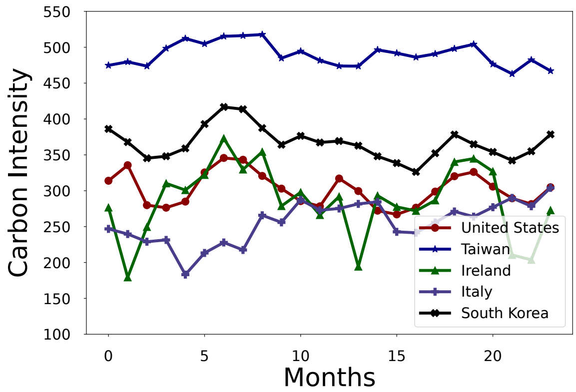


Figure 5.6: Total carbon intensity from 2021 to 2022

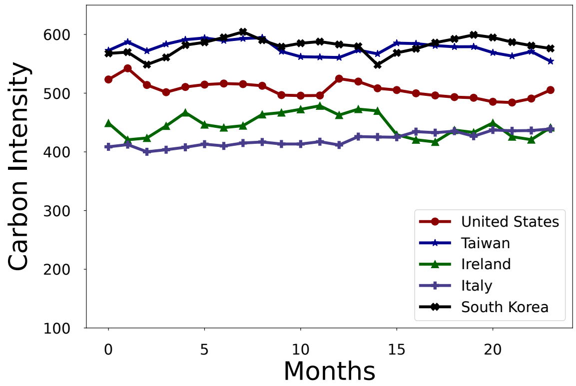


Figure 5.7: Residual carbon intensity from 2021 to 2022

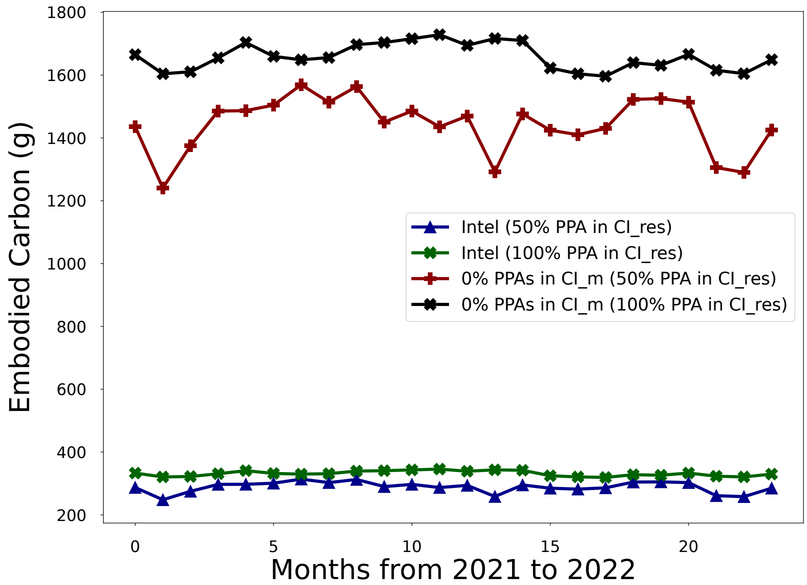


Figure 5.8: The monthly embodied carbon of 7nm CPUs in from 2021 to 2022 in Ireland made by Intel and manufacturers without PPAs.

(Intel vs. TSMC). PPAs amplify discrepancies as STEC-M captures carbon attribution methods, whereas ACT does not; (2) Temporal Trends: STEC-M highlights decreasing embodied carbon over time as manufacturers (e.g., TSMC) increase PPA use. ACT fails to reflect this dynamic. (3) Residual Carbon Intensity: higher renewable excluded through PPAs can result in higher embodied carbon.

5.5 STEC Accounting for Applications

Nowadays, there are studies on embodied carbon accounting and optimization on computer applications, such as CarbonMin for LLM inference [23], LLMCarbon for LLM training [33], Carbond for operating systems [94], and others. The embodied carbon of these applications is accounted for by the hardware that they require, which differs across applications to meet their performance requirements. For example, LLMCarbon performs carbon accounting on LLM training and requires more high-end memory and storage (e.g., 64×256GB Mircron 18nm, 64×32TB Samsung 20nm), whereas CarbonMin performs LLM inference and requires ordinary memory and storage (e.g., 900GB memory, 6000GB SSD).

Existing studies have estimated the carbon of specific hardware from ESG reports, which have been selected an ad hoc manner. Due to space limitations, in this paper we examine the embodied carbon accounting of two applications, LLM inference (CarbonMin) and LLM training (LLMCarbon) under the STEC-L-CS model, i.e., we recalculate CarbonMin and LLMCarbon using the STEC-L-CS model.

5.5.1 The Embodied Carbon Accounting of Applications

The embodied carbon of applications (EC_{app}) running on computing systems can be computed using the formula $EC_{app} = EC_{sys} \times TS \times RS$, where EC_{sys} represents the total embodied carbon of the system on which the application operates, TS denotes the time-share (the fraction of the total lifetime of the system reserved by the application), and RS is the resource-share (the fraction of the total available hardware resources used by the application) [94]. The difference between STEC models of applications and the previous model is that EC_{sys} is not an average value estimated by ACT. Instead, it is a range determined by the hardware range (HR) that is dynamic in the time and place of manufacture. Computing system suppliers, such as Microsoft [10] and AWS [9], typically provide the basic HR. Consequently,EC_{sys} can be computed as follows:

$$EC_{sys} = STEC(HR_{sys}) = \sum_{i \in HR_{sys}} EC_i(s_i, t_i) + N_{sys} \times P \tag{5.15}$$

where $EC_i(s_i, t_i)$ is the embodied carbon of hardware i with manufacturing time t_i and location s_i, P is the packaging carbon footprint (0.15kg [39]), and N_{sys} denotes the number of hardware components.

The Carbon Footprin of LLM Inference

CarbonMin evaluates the embodied carbon footprint of ChatGPT inference on Azure ND A100 v4-series instances [10]. We reestimate the embodied carbon using data from CarbonMin. For this analysis, we assume a hardware lifetime of 3 years and set the RS to 1. Based on CarbonMin's experimental results, the average computation time per the GPT-3 inference request is 2.07 GPU-sec/request. Therefore, the TS for a single request is 2.07 sec/3 years. Then, the embodied carbon of one ChatGPT-3 inference request is determined by the formula $EC_{infer} = EC_{sys} \times TS \times RS = EC_{sys} \times 2.07sec/3years$. Table 5.11 (a) details the HR for the Azure ND A100 v4-series as used in CarbonMin. The CPU and GPU in Azure ND A100 v4-series are AMD EPYC 7V12 (Rome) and Nvidia A100, which is manufactured by TSMC. Therefore, the spatial location for CPU and GPU is limited in Taiwan. Azure ND A100 v4-series only provides information on capacity for storage and memory and does not give detailed information about the manufacturer. One potential reason is that the memory and storage that can meet the performance requirements can be supplied by multiple manufacturers in the Azure could computing. Therefore the HR for storage and memory components contains six major IC-producing regions (refer to Table 1.2).

The Carbon Footprint of LLM Training

We reestimate the embodied carbon for training the LLM [25] using STEC models, drawing on data from LLMCarbon. Specifically, the training period of the LLM is 20.4 days, the hardware lifespan is 5 years, and RS is set to 1. Thus, the embodied carbon is calculated using the formula $EC_{train} = EC_{sys} \times TS \times RS = EC_{sys} \times 20.4days/5years$. Table 5.12 (a) outlines the HR in LLMCarbon. Here, the components are from certain manufacturers; therefore, the spatial location is limited to Taiwan for CPU and GPU, Korea for SSD, and USA for DRAM.

5.5.2 The Results of Embodied Carbon Accounting of Applications

The primary criterion for CarbonMin is the embodied carbon measured in grams per 1000 inference requests , specifically for ChatGPT-3 inference. The main criterion of LLMCarbon is the embodied carbon measured in tons for training a large language model (specifically, the XLM model [25] with 570M parameters).

Table 5.11 (a) and Table 5.12 (a) list the hardware of CarbonMin and LLMCarbon. The corresponding embodied carbon values are presented in Table 5.11 (b) and Table table: llm (b), respectively. We develop algorithms to find the location and time period with the minimum amount of embodied carbon to manufacture the hardware, as well as with the 20th percentile, median, 80 percentile, and maximum amount (refer to Table 5.11 (b) and Table 5.12 (b)). We find that STEC-CS leads to great differences for the LLM inference. Specifically, CarbonMin reports an embodied carbon of 0.31g per 1000 ChatGPT-3 inferences. Under STEC-CS, this value can be the 0.21g per 1K inference (or the 0.25g per 1K inference at the 20th percentile), a difference of 32.26% (or 19.35%). As a comparison, STEC-L-CS results in small differences for LLM training, e.g., the difference between STEC-L-CS and LLMCarbon is 3.17% (or 1.59% at the 20th percentile). This is attributed to the fact that LLM inference utilizes ordinary hardware with a wide range of spatial-temporal manufacturing options, whereas the hardware for LLM training, to the best of our knowledge, is more restricted.

5.5.3 STEC Accounting for Real-world Processors

To comprehensively characterize the environmental impacts of modern computing systems, we conduct a systematic location-based Spatial-Temporal Embedded Carbon (STEC) analysis across six real-world representative processors from leading manufac-

Table 5.11: The comparison between CarbonMin and STEC on the embodied carbon accounting of the LLM inference on Azure ND A100 v4-series instances.

Table (a)	Hardware	Number/Size	Technology Nodes	Die Size
	CPU	1	TSMC,7nm	74 mm2
	GPU	8	TSMC,7nm	826 mm2
	Memory	900GB	/	/
	Storage	6000GB	/	/

Table (b)	Embodied carbon (g)/1k inference requests				
	Min.	20th percentile	Median	80th percentile	Max.
CarbonMin	0.31				
STEC-CS	0.21	0.25	0.28	0.33	0.35
Dif. (%)	32.26%	19.35%	9.68%	6.45%	12.90%

Table 5.12: The comparison between LLMCarbon and STEC on the embodied carbon accounting of the LLM training.

Table (a)	Hardware	Description	Unit	Number
	GPU	V100 TSMC 12nm	815 mm2	512
	CPU	TSMC 16nm	147 mm2	64
	SSD	Samsung 20nm	32TB	64
	DRAM	Micron 18nm	256GB	64
	Other	/	/	64

Table (b)	Embodied carbon (t)				
	Min.	20th percentile	Median	80th percentile	Max.
LLMCarbon	0.63				
STEC-CS	0.61	0.62	0.64	0.66	0.67
Dif. (%)	3.17%	1.59%	1.59%	4.76%	6.35%

turers. The investigation encompasses both high-performance computing GPUs and consumer-grade processors, revealing significant variations in embodied carbon footprints influenced by manufacturing locations, temporal energy dynamics, and technological characteristics. Table 5.13 presents our quantified findings with extended statistical analysis.

Manufacturing Geography Impacts. The analysis reveals distinct carbon footprint patterns between TSMC (Taiwan)-manufactured and Samsung (South Korea)-fabricated devices. TSMC's advanced nodes (12nm-4nm from V100, A100, and H100) demonstrate relatively stable embodied carbon ranges (5.56%-7.93%), attributable to Taiwan's coal-dominated energy mix with limited temporal variation. In contrast, Samsung's 8nm process (Nvidia 3090) shows a 16.13% embodied carbon range fluctuation, reflecting South Korea has more renewable energy than Taiwan.

Multi-Foundry Production Dynamics. Processors manufactured across multiple foundries exhibit amplified embodied carbon variability:

- Ascend 910 (TSMC in Taiwan/SMIC in China): 16.07% range spanning 7.15-8.30 kg CO_2
- Apple A9 (TSMC in Taiwan/Samsung in South Korea): 22.64% variation (1.06-1.30 kg CO_2)

The spatial and temporal characteristics of carbon intensity amplify the fluctuations of the embodied carbon.

Technology Node Correlation. Our analysis reveals the critical amplification effect between technology node advancement and spatial-temporal embodied carbon variation. The progression to advanced nodes (from 12nm to 4nm for V100, A100, and H100) not only increases absolute embodied carbon but significantly magnifies its temporal fluctuation range by 42.6% (from 5.56% to 7.93%). Each node transition can introduce additional manufacturing steps, increasing total energy consumption

Table 5.13: The STEC accounting result for real-world processors.

Processors	Technology Nodes	Die Size	Embodied Carbon (kg)			
			Min.	Median	Max.	Relative Range (%)
V100 GPU	TSMC, 12nm	815 mm^2	9.62	9.93	10.16	5.56%
A100 GPU	TSMC, 7nm	826 mm^2	13.21	13.76	14.16	7.26%
H100 GPU	TSMC, 4nm	814 mm^2	15.47	16.18	16.70	7.93%
Nvidia 3090	Samsung, 8nm	628 mm^2	6.95	7.38	8.07	16.13%
Ascend 910	TSMC/SMIC, 7nm	456.25 mm^2	7.15	7.70	8.30	16.07%
Apple A9	TSMC/Samsung, 14	104.5 mm^2	1.06	1.22	1.3	22.64%

per die. This energy intensification amplifies sensitivity to temporal grid carbon fluctuations.

These findings necessitate a paradigm shift in sustainable semiconductor design and manufacturing. System designers must evaluate embodied carbon through both temporal and spatial dimensions to avoid greenwashing risks. Blind reliance on annual regional carbon factors may miscalculate embodied carbon. Systematically incorporating STEC metrics into procurement contracts creates market incentives for manufacturers to adopt low-carbon energy sources and optimize production schedules to align with renewable energy availability.

5.6 Discussions

It should be acknowledged that STEC models carry inherent limitations, which are caused by partially using averaged/aggregated and outdated data in various aspects. There is an industry value-chain, and our calculation of the carbon released in the upstream industry value-chain uses industry averages. For example, In the absence of granular, manufacturer-specific GPS/MPS data across regions and years, our model adopts averaged values derived from industry-level reports (e.g., $500g/cm^2$ for MPS). This assumption may overlook temporal or spatial heterogeneities. For instance, advancements in production technologies or logistics could reduce material/energy intensities over time. However, obtaining dynamic, facility-level data remains a significant challenge due to limited transparency in industrial carbon accounting and inconsistent reporting standards globally. Another limitation is caused by the residual mix of electricity in grids and PPAs. In market-based carbon attribution approaches, precise quantification of power purchase agreement (PPA) volumes serves as a critical determinant for computing residual grid mix to get CI_{res} and CI_m. The current landscape presents significant transparency challenges, as PPA transactions typically remain obscured from grid operators and the common public. While specialized commercial services [20] track the amount of electricity under PPA, such market data is incomplete and constitutes paid subscription content rather than public infrastructure. Existing open-access repositories of residual grid composition [36] are only available in certain regions with yearly data. Addressing this information asymmetry requires coordinated action between PPA contracting parties and energy market regulators. For example, establish standardized reporting protocols for PPA volumes through grid-connected generation assets and implement regulatory frameworks mandating near-real-time disclosure of contractual generation data. Such infrastructure enhancements would enable dynamic carbon accounting mechanisms. The computational sustainability community must advocate for these systemic changes through interdisciplinary collaborations with energy economists and policy architects.

5.7 Chapter summary

This paper observes that for the same product class of computer hardware, e.g., 28nm CPUs, the embodied carbon of its product instances (e.g., manufactured in the winter in Taiwan or in the summer in Ireland) can differ significantly. Additionally, this difference can also exist in different renewable energy attribution (e.g., location-based vs. market-based). We present new spatial-temporal embodied carbon (STEC) models with two attribution methods (STEC-L/M). We evaluate STEC with a state-of-the-art embodied carbon accounting model and also apply it to embodied carbon accounting for LLM training and inference. The results showed that significant differences can exist; in other words, neglecting the attribution methods and spatial-temporal factors can lead to non-trivial inaccuracies in embodied carbon accounting.

Chapter 6

Conclusion and Future Direction

6.1 Conclusion

This thesis addresses the critical challenge of carbon awareness in AI and computing systems through two complementary pillars: carbon intensity forecasting and embodied carbon accounting. The increasing carbon footprint of the ICT sector, coupled with the global urgency to decarbonize, necessitates accurate, granular, and actionable carbon intelligence to guide sustainable computing practices.

In the first part of the thesis, we focus on carbon intensity forecasting of electricity. We proposed CFCG, a GNN-LSTM hybrid model that captures both spatial and temporal dependencies in cross-border power grids, significantly outperforming existing regional and one-hop models. Building upon this, we introduce DGCFM in the second part, a dual-graph empowered time series foundation model for carbon intensity forecasting that generalizes across diverse regions with limited data, demonstrating strong few-shot and zero-shot capabilities. These models enable downstream applications, such as LLM training and inference, to optimize their operational carbon emissions by shifting workloads to times and locations with lower carbon intensity.

In the third part, we turn to embodied carbon accounting for computer systems. We develop the STEC framework, which incorporates spatial-temporal dynamics and dual carbon attribution methods (location-based and market-based) to provide a more accurate and granular accounting of embodied carbon. Our evaluation reveals significant variations in embodied carbon due to manufacturing locations, temporal factors, and corporate renewable energy investments (PPAs), which are overlooked by existing models like ACT. We further demonstrate how STEC can refine the embodied carbon accounting of real-world applications such as LLM training and inference.

Together, these contributions provide a comprehensive toolkit for carbon-aware computing, enabling both operational and embodied carbon optimization across the life-cycle of computer and AI systems.

6.2 Future Directions

This thesis concludes with a discussion of potential avenues for future research and recommendations for advancing the field. First, extend STEC to reflect technological evolution and regional supply chain changes. Incorporate uncertainty quantification to provide confidence intervals for embodied carbon accounting. Second, explore joint optimization of hardware design, manufacturing scheduling, and operational policies to minimize end-to-end carbon footprint. Develop carbon-aware schedulers for cloud and edge systems that consider both operational and embodied carbon. Third, advocate for standardized reporting of carbon intensity data, residual mixes, and PPA information across regions and manufacturers. Contribute to open-source carbon databases and tools to foster transparency and reproducibility. Lastly, studying the suitable carbon signal (average carbon intensity vs. marginal carbon intensity) for carbon-aware systems is also a promising direction.

References

[1] Farah Abdoune, Lorenzo Ragazzini, Maroua Nouiri, Elisa Negri, and Olivier Cardin. Toward digital twin for sustainable manufacturing: A data-driven approach for energy consumption behavior model generation. *Computers in Industry*, 150:103949, 2023.

[2] Bilge Acun, Benjamin Lee, Fiodar Kazhamiaka, Kiwan Maeng, Udit Gupta, Manoj Chakkaravarthy, David Brooks, and Carole-Jean Wu. Carbon explorer: A holistic framework for designing carbon aware datacenters. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, pages 118–132, 2023.

[3] United Satets Environmental Protection Agency. Greenhouse gas protocol: Scope 2 guidance. *https://csrreportbuilder.intel.com/pdfbuilder/pdfs/CSR-2022-23-Full-Report.pdf*, 2021.

[4] Beng Wah Ang and Bin Su. Carbon emission intensity in electricity production: A global analysis. *Energy Policy*, 94:56–63, 2016.

[5] BW Ang, Peng Zhou, and LP Tay. Potential for reducing global carbon emissions from electricity production—a benchmarking analysis. *Energy Policy*, 39(5):2482–2489, 2011.

[6] Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebas-

tian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

[7] Lasse F Wolff Anthony, Benjamin Kanding, and Raghavendra Selvan. Carbontracker: Tracking and predicting the carbon footprint of training deep learning models. *arXiv preprint arXiv:2007.03051*, 2020.

[8] Francesco Asdrubali, Giorgio Baldinelli, Francesco D'Alessandro, and Flavio Scrucca. Life cycle assessment of electricity production from renewable energies: Review and results harmonization. *Renewable and Sustainable Energy Reviews*, 42:1113–1122, 2015.

[9] AWS. Amazon ec2. `https://aws.amazon.com/ec2/instance-types/?nc1=h_ls`, 2023.

[10] Microsoft Azure. Nd a100 v4-series. `https://learn.microsoft.com/en-us/azure/virtual-machines/nda100-v4-series/`, 2023.

[11] Heymi Bahar and Jehan Sauvage. Cross-border trade in electricity and the development of renewables-based electric power: lessons from europe. *OECD trade and environment working papers*, 2013.

[12] M Garcia Bardon, P Wuytens, L-Å Ragnarsson, G Mirabelli, D Jang, G Willems, A Mallik, A Spessot, J Ryckaert, and B Parvais. Dtco including sustainability: Power-performance-area-cost-environmental score (ppace) analysis for logic technologies. In *2020 IEEE International Electron Devices Meeting (IEDM)*, pages 41.4.1–41.4.4. IEEE, 2020.

[13] Noman Bashir, David Irwin, and Prashant Shenoy. On the promise and pitfalls of optimizing embodied carbon. In *Proceedings of the 2nd Workshop on Sustainable Computer Systems*, pages 1–6, 2023.

[14] Jieming Bian, Lei Wang, Shaolei Ren, and Jie Xu. Cafe: Carbon-aware federated learning in geographically distributed data centers. In *Proceedings of the*

15th ACM International Conference on Future and Sustainable Energy Systems, pages 347–360, 2024.

[15] Neeraj Dhanraj Bokde, Bo Tranberg, and Gorm Bruun Andresen. Short-term co2 emissions forecasting based on decomposition approaches and its impact on electricity market scheduling. *Applied Energy*, 281:116061, 2021.

[16] Sarah B Boyd. *Life-cycle assessment of semiconductors.* Springer Science & Business Media, 2011.

[17] Hans G Brauch. Environmental and energy security: Conceptual evolution and potential applications to european cross-border energy supply infrastructure. In *Environmental Security of the European Cross-Border Energy Supply Infrastructure*, pages 155–185. Springer, 2015.

[18] Semen Andreevich Budennyy, Vladimir Dmitrievich Lazarev, Nikita Nikolaevich Zakharenko, Aleksei N Korovin, OA Plosskaya, Denis Valer'evich Dimitrov, VS Akhripkin, IV Pavlov, Ivan Valer'evich Oseledets, Ivan Segundovich Barsola, et al. Eco2ai: carbon emissions tracking of machine learning models as the first step towards sustainable ai. In *Doklady Mathematics*, volume 106, pages S118–S128. Springer, 2022.

[19] José Pedro Carvalho, Luís Bragança, and Ricardo Mateus. Sustainable building design: Analysing the feasibility of bim platforms to support practical building sustainability assessment. *Computers in Industry*, 127:103400, 2021.

[20] CEBA. CEBA Deal Tracker. `https://cebuyers.org/deal-tracker/`, 2025.

[21] Bowen Chen, Gillian Dobbie, Neelesh Rampal, and Yun Sing Koh. Unveiling climate drivers via feature importance shift analysis in new zealand. In *Proceedings of the ACM on Web Conference 2024*, pages 4595–4606, 2024.

[22] Andrew A Chien. Genai: Giga, terawatt-hours, and gigatons of co2. *Communications of the ACM*, 66(8):5–5, 2023.

[23] Andrew A Chien, Liuzixuan Lin, Hai Nguyen, Varsha Rao, Tristan Sharma, and Rajini Wijayawardana. Reducing the carbon impact of generative ai inference (today and in 2035). In *The 2nd Workshop on Sustainable Computer Systems (HotCarbon'23)*, pages 1–7, 2023.

[24] Jeongdong Choe. XPoint Memory Comparison Process Architecture. https://www.flashmemorysummit.com/English/Collaterals/Proceedings/2017/20170808_FR12_Choe.pdf, 2017.

[25] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, 2020.

[26] Lingyun Deng and Sanyang Liu. Advancing photovoltaic system design: An enhanced social learning swarm optimizer with guaranteed stability. *Computers in Industry*, 164:104209, 2025.

[27] Art Diem, Cristina Quiroz, and TH Pechan. How to use egrid for carbon footprinting electricity purchases in greenhouse gas emission inventories. *US Environmental Protection Agency*, pages 1–24, 2012.

[28] Arka Dutta, Adel Khorramrouz, Sujan Dutta, and Ashiqur R KhudaBukhsh. Down the toxicity rabbit hole: A framework to bias audit large language models with key emphasis on racism, antisemitism, and misogyny. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence AI for Good*, pages 7242–50, 2024.

[29] EMBIR. Electricity sector. https://ember-climate.org/countries-and-regions/, 2023.

[30] ENTSO-E. Electricity in europe 2017, statistics and data. *https://www.entsoe.eu/publications/statistics-and-data/#electricity-in-europe*, 2017.

[31] ENTSO-E. High-level report tyndp 2022. *https://2022.entsos-tyndp-scenarios.eu/*, 2022.

[32] Entsoe. Entsoe transparency platform. `https://transparency.entsoe.eu/dashboard/show`, 2024.

[33] Ahmad Faiz, Sotaro Kaneda, Ruhan Wang, Rita Chukwunyere Osi, Prateek Sharma, Fan Chen, and Lei Jiang. LLMCarbon: Modeling the end-to-end carbon footprint of large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[34] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[35] Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. In *Forty-first International Conference on Machine Learning*, 2024.

[36] Green-e. Residual Mix Emissions Rates (2017 Data). `https://www.green-e.org/2019-residual-mix`, 2025.

[37] Albert Gu, Caglar Gulcehre, Thomas Paine, Matt Hoffman, and Razvan Pascanu. Improving the gating mechanism of recurrent neural networks. In *International Conference on Machine Learning*, pages 3800–3809. PMLR, 2020.

[38] Jie Gui, Zhenan Sun, Yonggang Wen, Dacheng Tao, and Jieping Ye. A review on generative adversarial networks: Algorithms, theory, and applications. *IEEE transactions on knowledge and data engineering*, 35(4):3313–3332, 2021.

[39] Udit Gupta, Mariam Elgamal, Gage Hills, Gu-Yeon Wei, Hsien-Hsin S Lee, David Brooks, and Carole-Jean Wu. Act: Designing sustainable computer systems with an architectural carbon modeling tool. In *Proceedings of the 49th Annual International Symposium on Computer Architecture*, pages 784–799, 2022.

[40] Edward Halawa, Geoffrey James, Xunpeng Shi, Novieta H Sari, and Rabindra Nepal. The prospect for an australian–asian power grid: A critical appraisal. *Energies*, 11(1):200, 2018.

[41] Geoffrey Hammond, Craig Jones, Ed Fiona Lowrie, and Peter Tse. Embodied carbon. *The inventory of carbon and energy (ICE). Version (2.0)*, 2011.

[42] Walid A Hanafy, Qianlin Liang, Noman Bashir, David Irwin, and Prashant Shenoy. Carbonscaler: Leveraging cloud workload elasticity for optimizing carbon-efficiency. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 7(3):1–28, 2023.

[43] Peter Henderson, Jieru Hu, Joshua Romoff, Emma Brunskill, Dan Jurafsky, and Joelle Pineau. Towards the systematic reporting of the energy and carbon footprints of machine learning. *Journal of Machine Learning Research*, 21(248):1–43, 2020.

[44] Edgar G Hertwich and Richard Wood. The growing importance of scope 3 greenhouse gas emissions from industry. *Environmental Research Letters*, 13(10):104013, 2018.

[45] Lion Hirth, Jonathan Mühlenpfordt, and Marisa Bulkeley. The entso-e transparency platform–a review of europe's most ambitious electricity data platform. *Applied energy*, 225:1054–1067, 2018.

[46] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[47] Hiroki Hondo. Life cycle ghg emission analysis of power generation systems: Japanese case. *Energy*, 30(11-12):2042–2056, 2005.

[48] SK hynix. Sustainability report. `https://www.skhynix.com/sustainability/UI-FR-SA1601/`, 2021.

[49] Paris IEA. Global energy review: Co2 emissions in 2021. *https://www.iea.org/reports/global-energy-review-co2-emissions-in-2021-2*, 2022.

[50] World in Data. Carbon intensity of electricity. `https://ourworldindata.org/grapher/carbon-intensity-electricity?tab=chart`, 2023.

[51] Intel. 2022-23 corporate responsibility report. *https://csrreportbuilder.intel.com/pdfbuilder/pdfs/CSR-2022-23-Full-Report.pdf*, 2022.

[52] Paul Jaccard. Nouvelles recherches sur la distribution florale. *Bull. Soc. Vaud. Sci. Nat.*, 44:223–270, 1908.

[53] Zhuangzhuang Jia, Grani A Hanasusanto, Phebe Vayanos, and Weijun Xie. Learning fair policies for multi-stage selection problems from observational data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 21188–21196, 2024.

[54] Sven Köhler, Benedict Herzog, Henriette Hofmeier, Manuel Vögele, Lukas Wenzel, Andreas Polze, and Timo Hönig. Carbon-aware memory placement. In *Proceedings of the 2nd Workshop on Sustainable Computer Systems*, pages 1–7, 2023.

[55] Sandeep Kumar, Amit K Shukla, and Pranab K Muhuri. Anomaly based novel multi-source unsupervised transfer learning approach for carbon emission centric gdp prediction. *Computers in Industry*, 126:103396, 2021.

[56] Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, 2019.

[57] Loïc Lannelongue, Jason Grealey, and Michael Inouye. Green algorithms: quantifying the carbon footprint of computation. *Advanced science*, 8(12):2100707, 2021.

[58] Kenneth Leerbeck, Peder Bacher, Rune Grønborg Junker, Goran Goranović, Olivier Corradi, Razgar Ebrahimy, Anna Tveit, and Henrik Madsen. Short-term forecasting of co2 emission intensity in power grids by machine learning. *Applied Energy*, 277:115527, 2020.

[59] Baolin Li, Yankai Jiang, Vijay Gadepally, and Devesh Tiwari. Sprout: Green generative ai with carbon-efficient llm inference. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21799–21813, 2024.

[60] Baolin Li, Yankai Jiang, Vijay Gadepally, and Devesh Tiwari. Toward sustainable genai using generation directives for carbon-friendly large language model inference. *arXiv preprint arXiv:2403.12900*, 2024.

[61] Jie Li, Yuhui Deng, Yi Zhou, Zhen Zhang, Geyong Min, and Xiao Qin. Towards thermal-aware workload distribution in cloud data centers based on failure models. *IEEE Transactions on Computers*, 72(2):586–599, 2022.

[62] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations*, 2018.

[63] Gaoqi Liang, Guolong Liu, Junhua Zhao, Yanli Liu, Jinjin Gu, Guangzhong Sun, and Zhaoyang Dong. Super resolution perception for improving data completeness in smart grid state estimation. *Engineering*, 6(7):789–800, 2020.

[64] Anran Liu, Yihao Liu, Jinjin Gu, Yu Qiao, and Chao Dong. Blind image super-resolution: A survey and beyond. *IEEE transactions on pattern analysis and machine intelligence*, 45(5):5461–5480, 2022.

[65] Guolong Liu, Jinjin Gu, Junhua Zhao, Fushuan Wen, and Gaoqi Liang. Super resolution perception for smart meter data. *Information Sciences*, 526:263–273, 2020.

[66] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. *arXiv preprint arXiv:2310.06625*, 2023.

[67] Kadan Lottick, Silvia Susai, Sorelle A Friedler, and Jonathan P Wilson. Energy usage reports: Environmental awareness as part of algorithmic accountability. *arXiv preprint arXiv:1911.08354*, 2019.

[68] Gordon Lowry. Day-ahead forecasting of grid carbon intensity in support of heating, ventilation and air-conditioning plant demand response decision-making to reduce carbon emissions. *Building Services Engineering Research and Technology*, 39(6):749–760, 2018.

[69] Huirong Ma, Zhi Zhou, Xiaoxi Zhang, and Xu Chen. Toward carbon-neutral edge computing: Greening edge ai by harnessing spot and future carbon markets. *IEEE Internet of Things Journal*, 10(18):16637–16649, 2023.

[70] Diptyaroop Maji, Noman Bashir, David Irwin, Prashant Shenoy, and Ramesh K Sitaraman. The green mirage: Impact of location-and market-based carbon intensity estimation on carbon optimization efficacy. In *Proceedings of the 15th ACM International Conference on Future and Sustainable Energy Systems*, pages 256–267, 2024.

[71] Diptyaroop Maji, Prashant Shenoy, and Ramesh K Sitaraman. Carboncast: multi-day forecasting of grid carbon intensity. In *Proceedings of the 9th ACM*

International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, pages 198–207, 2022.

[72] Diptyaroop Maji, Ramesh K Sitaraman, and Prashant Shenoy. Dacf: day-ahead carbon intensity forecasting of power grids using machine learning. In *Proceedings of the Thirteenth ACM International Conference on Future Energy Systems*, pages 188–192, 2022.

[73] Electricity Maps. electricity data. `https://app.electricitymaps.com/map`, 2024.

[74] Karim Ali Ibrahim Menoufi et al. Life cycle analysis and life cyle impact assessment methodologies: a state of the art. 2011.

[75] Gregory J Miller, Kevin Novan, and Alan Jenn. Hourly accounting of carbon emissions from electricity consumption. *Environmental Research Letters*, 17(4):044073, 2022.

[76] Jiqian Mo and Zhiguo Gong. Cross-city multi-granular adaptive transfer learning for traffic flow prediction. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[77] United Nations. Carbon neutrality by 2050: the world's most urgent mission. `https://www.un.org/sg/en/content/sg/articles/2020-12-11/carbon-neutrality-2050-the-worlds-most-urgent-mission`, 2020.

[78] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

[79] Ding Ning, Varvara Vetrova, Karin R Bryan, Yun Sing Koh, Andreas Voskou, N Kouagou, and Arnab Sharma. Diving deep: Forecasting sea surface temperatures and anomalies. *arXiv preprint arXiv:2501.05731*, 2025.

[80] Simona-Vasilica Oprea and Adela Bâra. Ultra-short-term forecasting for photovoltaic power plants and real-time key performance indicators analysis with big data solutions. two case studies-pv agigea and pv giurgiu located in romania. *Computers in Industry*, 120:103230, 2020.

[81] C Anna Palagan, S Sebastin Antony Joe, SJ Jereesha Mary, and E Edwin Jijo. Predictive analysis-based sustainable waste management in smart cities using iot edge computing and blockchain technology. *Computers in Industry*, 166:104234, 2025.

[82] Divya Pandey, Madhoolika Agrawal, and Jai Shanker Pandey. Carbon footprint: current methods of estimation. *Environmental monitoring and assessment*, 178:135–160, 2011.

[83] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. Carbon emissions and large neural network training. *arXiv preprint arXiv:2104.10350*, 2021.

[84] Lidia Puka and Kacper Szulecki. The politics and economics of cross-border electricity infrastructure: A framework for analysis. *Energy Research & Social Science*, 4:124–134, 2014.

[85] O Pupo-Roncallo, J Campillo, D Ingham, L Ma, and M Pourkashanian. The role of energy storage and cross-border interconnections for increasing the flexibility of future power systems: The case of colombia. *Smart Energy*, 2:100016, 2021.

[86] Xinchi Qiu, Titouan Parcollet, Javier Fernandez-Marques, Pedro PB Gusmao, Yan Gao, Daniel J Beutel, Taner Topal, Akhil Mathur, and Nicholas D Lane. A first look into the carbon footprint of federated learning. *Journal of Machine Learning Research*, 24(129):1–23, 2023.

[87] Shen Qu, Hongxia Wang, Sai Liang, Avi M Shapiro, Sanwong Suh, Seth Sheldon, Ory Zik, Hong Fang, and Ming Xu. A quasi-input-output model to improve the estimation of emission factors for purchased electricity from interconnected grids. *Applied energy*, 200:249–259, 2017.

[88] Miloš Radenković, Jelena Lukić, Marijana Despotović-Zrakić, Aleksandra Labus, and Zorica Bogdanović. Harnessing business intelligence in smart grids: A case of the electricity market. *Computers in industry*, 96:40–53, 2018.

[89] Ana Radovanovic, Ross Koningstein, Ian Schneider, Bokan Chen, Alexandre Duarte, Binz Roy, Diyue Xiao, Maya Haridasan, Patrick Hung, Nick Care, et al. Carbon-aware computing for datacenters. *arXiv preprint arXiv:2106.11750*, 2021.

[90] Ana Carolina Riekstin, Antoine Langevin, Thomas Dandres, Ghyslain Gagnon, and Mohamed Cheriet. Time series-based ghg emissions prediction for smart homes. *IEEE Transactions on Sustainable Computing*, 5(1):134–146, 2018.

[91] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[92] Soumyendu Sarkar, Avisek Naug, Ricardo Luna, Antonio Guillen, Vineet Gundecha, Sahand Ghorbanpour, Sajad Mousavi, Dejan Markovikj, and Ashwin Ramesh Babu. Carbon footprint reduction for sustainable data centers in real-time. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22322–22330, 2024.

[93] Mirko Schäfer, Bo Tranberg, Dave Jones, and Anke Weidlich. Tracing carbon dioxide emissions in the european electricity markets. In *2020 17th International Conference on the European Energy Market (EEM)*, pages 1–6. IEEE, 2020.

[94] Andreas Schmidt, Gregory Stock, Robin Ohs, Luis Gerhorst, Benedict Herzog, and Timo Hönig. carbond: An operating-system daemon for carbon awareness.

In *Proceedings of the 2nd Workshop on Sustainable Computer Systems*, pages 1–6, 2023.

[95] Seagate. Esg. https://www.seagate.com/esg/planet/product-sustainability/, 2019.

[96] Seagate. Exos 7e8. *https://www.seagate.com/gb/en/esg/planet/product-sustainability/exos-7e8-sustainability-report/*, 2019.

[97] Seagate. Nytro 3530. *https://www.seagate.com/gb/en/esg/planet/product-sustainability/nytro-3530-sustainability-report/*, 2019.

[98] Seagate. Product life cycle assessments. *https://www.seagate.com/gb/en/esg/planet/product-sustainability/*, 2020.

[99] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000.

[100] Bhupendra Kumar Singh. South asia energy security: Challenges and opportunities. *Energy policy*, 63:458–468, 2013.

[101] Chao Song, Youfang Lin, Shengnan Guo, and Huaiyu Wan. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 914–921, 2020.

[102] Lidong Song, Yiyan Li, and Ning Lu. Profilesr-gan: A gan based super-resolution method for generating high-resolution load profiles. *IEEE Transactions on Smart Grid*, 13(4):3278–3289, 2022.

[103] Mary Elizabeth Sotos. Ghg protocol scope 2 guidance. 2015.

[104] Abel Souza, Noman Bashir, Jorge Murillo, Walid Hanafy, Qianlin Liang, David Irwin, and Prashant Shenoy. Ecovisor: A virtual energy system for carbon-efficient applications. In *Proceedings of the 28th ACM International Conference*

on Architectural Support for Programming Languages and Operating Systems, Volume 2, pages 252–265, 2023.

[105] Sphera. Life cycle assessment software. *https://sphera.com/solutions/product-stewardship/life-cycle-assessment-software-and-data/lca-for-experts/*, 2023.

[106] Shuomiao Su, Zhi Zhou, Tao Ouyang, Ruiting Zhou, and Xu Chen. Learning to be green: Carbon-aware online control for edge intelligence with colocated learning and inference. In *2023 IEEE 43rd International Conference on Distributed Computing Systems (ICDCS)*, pages 567–578. IEEE, 2023.

[107] Swamit Tannu and Prashant J Nair. The dirty secret of ssds: Embodied carbon. *arXiv preprint arXiv:2207.10793*, 2022.

[108] Nguyen Thach, Patrick Habecker, Bergen Johnston, Lillianna Cervantes, Anika Eisenbraun, Alex Mason, Kimberly Tyler, Bilal Khan, and Hau Chan. Analyzing and predicting short-term substance use behaviors of persons who use drugs in the great plains of the us. *Plos one*, 19(11):e0312046, 2024.

[109] R Tomorrow. electricitymap. *https://www.electricitymap.org*, 2019.

[110] Bo Tranberg, Olivier Corradi, Bruno Lajoie, Thomas Gibon, Iain Staffell, and Gorm Bruun Andresen. Real-time carbon accounting method for the european electricity markets. *Energy Strategy Reviews*, 26:100367, 2019.

[111] ENTSOE transparency platform. European association for the cooperation of transmission system operators. *Retrieved February 1, 2023 from https://transparency.entsoe.eu/*, 2022.

[112] TWSC. Tsmc 2022 sustainability report. *https://esg.tsmc.com/en-US/file/public/e-all$_2$022.pdf*, 2022.

[113] Jan Frederick Unnewehr, Anke Weidlich, Leonhard Gfüllner, and Mirko Schäfer. Open-data based carbon emission intensity signals for electricity generation in

european countries–top down vs. bottom up approach. *Cleaner Energy Systems*, 3:100018, 2022.

[114] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[115] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph Attention Networks. *International Conference on Learning Representations*, 2018. accepted as poster.

[116] Shresth Verma, Arshika Lalan, Paula Rodriguez Diaz, Panayiotis Danassis, Amrita Mahale, Kumar Madhu Sudan, Aparna Hegde, Milind Tambe, and Aparna Taneja. Leveraging ai to improve health information access in the world's largest maternal mobile health program. *AI Magazine*, 45(4):526–536, 2024.

[117] Jean Verseille and Konstantin Staschus. The mesh-up: Entso-e and european tso cooperation in operations, planning, and r&d. *IEEE Power and Energy Magazine*, 13(1):20–29, 2014.

[118] Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. Timexer: Empowering transformers for time series forecasting with exogenous variables. *arXiv preprint arXiv:2402.19072*, 2024.

[119] Watttime. Watttime. *https://www.watttime.org/*, 2022.

[120] Christopher Weber, Deanna Matthews, Aranya Venkatesh, Christine Costello, and H Matthews. The 2002 us benchmark version of the economic input-output life cycle assessment (eio-lca) model. *Green Design Institute, Carnegie Mellon University*, 2009.

[121] Philipp Wiesner, Ilja Behnke, Dominik Scheinert, Kordian Gontarska, and Lauritz Thamsen. Let's wait awhile: how temporal workload shifting can reduce

carbon emissions in the cloud. In *Proceedings of the 22nd International Middleware Conference*, pages 260–272, 2021.

[122] Philipp Wiesner, Ramin Khalili, Dennis Grinwald, Pratik Agrawal, Lauritz Thamsen, and Odej Kao. Fedzero: Leveraging renewable excess energy in federated learning. In *The 15th ACM International Conference on Future and Sustainable Energy Systems*, pages 37–49, 2024.

[123] Priyantha Wijayatunga, Deb Chattopadhyay, and Prem N Fernando. Cross-border power trading in south asia: A techno economic rationale. 2015.

[124] Wiki. Semiconductor industry. `https://en.wikipedia.org/wiki/Semiconductor_industry`, 2019.

[125] Carole-Jean Wu, Ramya Raghavendra, Udit Gupta, Bilge Acun, Newsha Ardalani, Kiwan Maeng, Gloria Chang, Fiona Aga, Jinshi Huang, Charles Bai, et al. Sustainable ai: Environmental implications, challenges and opportunities. *Proceedings of Machine Learning and Systems*, 4:795–813, 2022.

[126] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.

[127] Wei Wu, Wei Chen, Yelin Fu, Yishuo Jiang, and George Q Huang. Unsupervised neural network-enabled spatial-temporal analytics for data authenticity under environmental smart reporting system. *Computers in Industry*, 141:103700, 2022.

[128] Zhuoyan Xu, Zhenmei Shi, Junyi Wei, Fangzhou Mu, Yin Li, and Yingyu Liang. Towards few-shot adaptation of foundation models via multitask finetuning. In *The Twelfth International Conference on Learning Representations*, 2024.

[129] Xiaoyue Yi, Llewellyn Tang, Reynold Cheng, Mengtian Yin, and Yu Zheng. Domain ontology to integrate building-integrated photovoltaic, battery energy

storage, and building energy flexibility information for explicable operation and maintenance. *Computers in Industry*, 166:104250, 2025.

[130] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017.

[131] Xiaoyang Zhang and Dan Wang. A gnn-based day ahead carbon intensity forecasting model for cross-border power grids. In *Proceedings of the 14th ACM International Conference on Future Energy Systems*, pages 361–373, 2023.

[132] Yunfan Zhao, Niclas Boehmer, Aparna Taneja, and Milind Tambe. Towards foundation-model-based multiagent system to accelerate ai for social impact. *arXiv preprint arXiv:2412.07880*, 2024.

[133] Zimu Zheng, Yuqi Wang, Quanyu Dai, Huadi Zheng, and Dan Wang. Metadata-driven task relation discovery for multi-task learning. In *IJCAI*, pages 4426–4432, 2019.

[134] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11106–11115, 2021.

[135] Yilin Zhuang, Yixuan Liu, Akhil Ahmed, Zhengang Zhong, Ehecatl A del Rio Chanona, Colin P Hale, and Mehmet Mercangöz. A hybrid data-driven and mechanistic model soft sensor for estimating co2 concentrations for a carbon capture pilot plant. *Computers in Industry*, 143:103747, 2022.