

3-15-2024

State of health and remaining useful life prediction of lithium-ion batteries with conditional graph convolutional network

Yupeng Wei

San Jose State University, yupeng.wei@sjsu.edu

Dazhong Wu

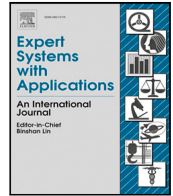
University of Central Florida

Follow this and additional works at: https://scholarworks.sjsu.edu/faculty_rsca

Recommended Citation

Yupeng Wei and Dazhong Wu. "State of health and remaining useful life prediction of lithium-ion batteries with conditional graph convolutional network" *Expert Systems with Applications* (2024). <https://doi.org/10.1016/j.eswa.2023.122041>

This Article is brought to you for free and open access by SJSU ScholarWorks. It has been accepted for inclusion in Faculty Research, Scholarly, and Creative Activity by an authorized administrator of SJSU ScholarWorks. For more information, please contact scholarworks@sjsu.edu.



State of health and remaining useful life prediction of lithium-ion batteries with conditional graph convolutional network

Yupeng Wei ^{a,*}, Dazhong Wu ^b

^a Department of Industrial and Systems Engineering, San Jose State University, San Jose, 95192, CA, USA

^b Department of Mechanical and Aerospace Engineering, University of Central Florida, Orlando, 32816, FL, USA

ARTICLE INFO

Keywords:

Lithium-ion battery
State of health
Remaining useful life
Graph convolutional network
Temporal convolutional network

ABSTRACT

Graph convolutional networks (GCNs) have been increasingly used to predict the state of health (SOH) and remaining useful life (RUL) of batteries. However, conventional GCNs have limitations. Firstly, the correlation between features and the SOH or RUL is not considered. Secondly, temporal relationships among features are not considered when projecting aggregated temporal features into another dimensional space. To address these issues, two types of undirected graphs are introduced to simultaneously consider the correlation among features and the correlation between features and the SOH or RUL. A conditional GCN is built to analyze these graphs. A dual spectral graph convolutional operation is introduced to analyze the topological structures of these graphs. Additionally, a dilated convolutional operation is integrated with the conditional GCN to consider the temporal correlation among the aggregated features. Two battery datasets are used to evaluate the effectiveness of the proposed method. Experimental results show that the proposed method outperforms other machine learning methods reported in the literature.

1. Introduction

1.1. Background

Lithium-ion batteries have been increasingly adopted as an energy resource for electric vehicles (EVs), drones, and portable electronics due to their high electricity density, lightweight, long lifetime, and low self-discharge rate (Cui, Gao, Mao, & Wang, 2022; Liu, Wu, Zhang, & Chen, 2014; Xi, Wang, Fu, & Mi, 2022). Over the past decades, many efforts have been made in the design of lithium-ion batteries to improve energy efficiency (Guo, Song, & Chen, 2009; Wang & Cao, 2008). However, similar to other engineered systems, the performance of lithium-ion batteries will deteriorate over time, also known as battery aging, due to physical and chemical changes as a result of daily usage and operations (He, Williard, Chen, & Pecht, 2014; Lee, Kim, & Lee, 2022). The battery aging problem may result in catastrophic failures such as fire hazards and outbursts. As a result, it is critical to estimate the state of health (SOH) and predict the remaining useful life (RUL) of batteries (Chao & Chen, 2011).

Over the past few years, data-driven methods have shown superior performance for SOH and RUL predictions of lithium-ion batteries (Khaleghi et al., 2022; Shen, Sadoughi, Li, Wang, & Hu, 2020; Wang, Zhao, Yang, & Tsui, 2017). Data-driven SOH and RUL prediction methods can be classified into two categories: filter-based approaches and

machine-learning approaches. The filter-based approaches incorporate the extended Kalman filter (EKF) (Nian, Shuzhi, & Xiongwen, 2021), unscented Kalman filter (UKF) (Zhang, Tu, et al., 2022), Lebesgue-sampling-based EKF (Niu, Wang, Liu, & Zhang, 2021), and particle filter (PF) (Zhang et al., 2021). For instance, Chen, Hu, Gui, and Zou (2014) used an unscented Kalman filter to predict the SOH and RUL of batteries. The proposed filter-based method included the internal resistance in the model parameters, the internal resistance was employed to infer the future degradation trajectory. The simulation results have demonstrated that the UKF achieved an accurate prediction of SOH, and its relative error rate is less than 4%. Yan et al. (2018) introduced the Lebesgue-sampling Kalman filter to reduce the computational cost in SOH predictions. The proposed filter dramatically reduced the computation time and cost. The experimental results indicated that the proposed filter reduced computational cost and delivered a precise estimation of the SOH. Dong, Chen, Wei, and Ling (2018) introduced a particle filter model to predict the SOH of batteries. The particle filter was combined with Brownian motion to simulate the dynamic changes during the degradation process. The accuracy and efficiency of the method were demonstrated by testing it on several lithium-ion batteries that were in a defective stage. The results of the experiments have shown that the particle filter achieved accurate RUL and SOH predictions.

* Corresponding author.

E-mail addresses: yupeng.wei@sjsu.edu (Y. Wei), dazhong.wu@ucf.edu (D. Wu).

One of the advantages of filter-based methods is their self-correction capabilities, however, these methods have limitations in dealing with large volumes of data (Lee, Kwon, & Lee, 2023; Park, Lee, Kim, Park, & Kim, 2020; Wei, Dong, & Chen, 2017). To address this issue, machine learning methods (Greenbank & Howey, 2023; Luo, Fang, Deng, & Tian, 2022) especially deep learning methods (Chang, Wang, Jiang, & Wu, 2021; Xu, Yang, Fei, Huang, & Tsui, 2021), such as convolutional neural network (CNN) (Al-Dulaimi, Zabihi, Asif, & Mohammadi, 2019; Ma et al., 2023), recurrent neural network (RNN) (Lu, Xiong, Tian, Wang, Hsu, Tsou, Sun, & Li, 2022), gated recurrent network (GRU) (Ungureanu, Micea, & Carstoiu, 2020), long short term memory (LSTM) (Wei, 2023; Xia, Song, Zheng, Pan, & Xi, 2020; Zhang, Jiang, et al., 2022), bidirectional LSTM (Guo, Wang, Yao, Fu, & Ning, 2023), and bidirectional GRU (Zhang et al., 2023), have been utilized to predict SOH and RUL, where the parameters of these methods can be trained and tuned with backpropagation (Ma, Yao, Liu, & Tang, 2022) or metaheuristics (Raskar & Nema, 2022; Zamfirache, Precup, Roman, & Petriu, 2022). For instance, Li et al. (2022) presented a hybrid deep learning approach, where one-dimensional CNN was integrated with LSTM to identify features related to battery degradation phenomenon, and Kolmogorov–Smirnov test was implemented to infer the prior distribution of hyperparameters used in the presented hybrid deep learning approach. Eddahech, Briat, Bertrand, Deléage, and Vinassa (2012) proposed an RNN-based model to estimate the SOH of a high electricity density battery cell. The RNN was implemented to track the degradation trajectory of several batteries in hybrid EV and EV usage. Cheng, Wang, and He (2021) integrated the empirical mode decomposition (EMD) approach with a LSTM network for accurate SOH and RUL predictions of batteries. The voltage and current measurements were fed into a LSTM model for SOH predictions, and the predicted SOH was fed into the EMD approach to eliminate the randomness brought by the capacity regeneration phenomenon so that the RUL of a battery can be predicted precisely. Duong and Raghavan (2018) combined a metaheuristic optimization approach with particle filtering methods to address the problem of sample degeneracy, thus enhancing the RUL prediction performance for lithium-ion batteries. Experimental results have demonstrated that the proposed method outperforms traditional metaheuristic approaches, such as the optimized particle filtering method.

One of the issues with the aforementioned deep learning methods is that they are not effective in revealing feature correlations. Such a correlation can be used to identify and aggregate features with high affinity and similarity to enhance the precision and robustness of a predictive model (Li, Zhao, Sun, Yan, & Chen, 2020; Wei & Wu, 2023). To reveal this correlation, undirected graphs have been increasingly used in current literature (Wei, Wu, & Terpenney, 2023), where the graph nodes represent feature vectors and the graph edges denote the similarity or affinity between features. To handle these undirected graphs more effectively, the Graph Convolutional Network (GCN) is increasingly used to predict the RUL of complex systems, due to its ability to leverage the topology of undirected graphs and provide better insights into data correlation (Wang, Cao, Xu, & Liu, 2022). For example, Wei and Wu (2022b) proposed an optimization model to build an undirected graph by simultaneously minimizing the graph density and maximizing the graph entropy. The GCN was adopted to handle the constructed graph and predict the SOH and RUL of lithium-ion batteries. The experimental results have shown that the GCN enables accurate predictions of SOH and RUL for batteries. Similarly, Li, Zhao, Sun, Yan, and Chen (2021) constructed multiple undirected graphs to represent the sensor and feature correlations in condition monitoring data collected from aircraft engines. The GCN was employed to handle these constructed undirected graphs and predict RUL for the engines. Numerical studies have demonstrated the capabilities of GCNs in dealing with these undirected graphs.

1.2. Research gap

While the effectiveness of GCNs in predicting RUL has been demonstrated, two issues remain to be addressed to enhance their robustness and precision. Firstly, the undirected graphs are usually constructed to represent the correlation among features. However, these graphs are not able to reveal the correlation between features and SOH or RUL. Revealing the correlation between features and SOH or RUL can help identify the most significant features that directly impact SOH or RUL. By understanding this correlation, predictive models can prioritize the important features during the prediction process, resulting in improved accuracy (Wei, Wu, & Terpenney, 2021). Most of the current methods are unable to consider this correlation because true SOH and RUL data are only available during training instead of testing. Secondly, traditional GCNs typically stack several spectral graph convolutional layers, where each layer performs two operations: aggregation and projection. The features with high affinity or similarity are first aggregated based on a pre-constructed undirected graph, and then the aggregated features are projected onto another higher-dimensional space. While the feature correlation can be effectively considered through the repeated aggregation of similar features, traditional GCNs do not take into account the temporal correlation of these aggregated features. To address these issues, this work introduced a conditional GCN with a dilated convolution operation. To address the first issue, this work constructed two types of undirected graphs. The first type of graphs (denoted as G_1) was used to consider the correlation among features, and the second type of graphs (denoted as G_2) was used to consider the correlation between features and SOH/RUL. Two feature spaces were extracted from the two types of graphs, respectively. Then, a KL-divergence was used to minimize the distance between two feature spaces so that the feature space extracted from G_1 can approximate the feature space extracted from G_2 . Therefore, even without the SOH/RUL, the correlation between the features and SOH/RUL can be taken into account when the feature space extracted from G_1 is used for testing. To address the second issue, this work implemented the dilated convolutional operation to consider temporal correlations after aggregating similar features in GCNs. The dilated convolutional operation was implemented for two primary reasons. First, the condition monitoring data collected for battery health management is a time series. The dilated convolutional operation in the temporal convolutional network (TCN) is designed to deal with time series data, making it capable of capturing long-term dependencies in time series data compared to GRU or LSTM (Zhen, Fang, Zhao, Ge, & Xiao, 2022). Second, the dilated convolutional operation can expand the receptive field of convolutional layers without significantly increasing the number of parameters (Bai, Kolter, & Koltun, 2018). This capability ensures that the proposed method is less susceptible to overfitting, which is particularly crucial when dealing with limited battery health data. The constraint of having limited battery health data is a common challenge in real-world applications due to the high cost and time-consuming nature of battery testing and monitoring. The major contributions of this work are outlined below:

- Two types of undirected graphs, denoted as G_1 and G_2 , were constructed. G_1 was used to capture correlations among features, while G_2 was used to capture correlations between features and SOH/RUL.
- The KL-divergence was introduced to minimize the distance between two feature spaces extracted from the two types of graphs so that the feature space extracted from G_1 can approximate the feature space extracted from G_2 .
- The dilated convolutional operation was implemented after aggregating similar features in GCNs to increase the receptive field of convolutional layers, thereby allowing for considering the temporal correlation among the aggregated features.

The remaining sections of this paper are organized in the following manner. Section 2 introduces the proposed conditional graph convolutional network with dilated convolution operations. Section 3 utilizes the NASA battery dataset to demonstrate the effectiveness of the proposed method, and Section 4 uses the Oxford battery degradation dataset to further demonstrate the efficiency of the proposed method. Section 5 concludes with a summary of this work and an examination of future work.

2. Conditional graph convolutional network with dilated convolutional operations

In this section, the conditional GCN with the dilated convolutional operations is introduced. Specifically, two types of undirected graphs are constructed in Section 2.1: standard undirected graphs and conditional undirected graphs. Next, in Section 2.2, the dual spectral graph convolution operation is presented, which is designed to deal with the topological structures of these graphs. Then, in Section 2.3, the dilated convolution operation is introduced. Finally, in Section 2.4, the training procedure for the conditional GCN with the dilated convolutional operation is outlined.

2.1. Constructing undirected graphs

In the field of battery health management, the most commonly used condition monitoring data include voltage, current, and temperature. Although numerous studies utilize both charge and discharge cycle data to estimate SOH and predict RUL, the availability of condition monitoring data during charge cycles is limited. Therefore, in this work, only condition monitoring data from discharge cycles was used for SOH estimations and RUL predictions. To capture the degradation trajectory of a battery, this work extracts several temporal features from each discharge cycle. These features include time to reach the minimum voltage, time discharged under constant or variable current modes, time to reach the maximum temperature, voltage decrease rate, and temperature increase rate. These features have been proven to be successful in estimating SOH and predicting RUL (Audin et al., 2021; Wei & Wu, 2022b).

Constructing undirected graphs involves initializing two types of graphs: standard undirected graphs and conditional undirected graphs. Standard undirected graphs are used to consider the correlation among features, while conditional undirected graphs are used to consider the correlation between features and SOH or RUL. The standard undirected graphs is the first graph set, denoted as \mathcal{G}_1 , consist of two graphs: the positively connected graph $\mathbb{G}_{1,+}$ and the negatively connected graph $\mathbb{G}_{1,-}$. Mathematically, \mathcal{G}_1 can be represented as $\mathcal{G}_1 = \{\mathbb{G}_{1,+}, \mathbb{G}_{1,-}\}$. To construct the graph $\mathbb{G}_{1,+}$ and the graph $\mathbb{G}_{1,-}$, this work first selects two different temporal feature vectors, $\mathbf{f}_{i,k} \in \mathbb{R}^{1 \times T_i}$ and $\mathbf{f}_{i,k'} \in \mathbb{R}^{1 \times T_i}$, from the extracted temporal feature matrix $\mathbf{F}_i \in \mathbb{R}^{K \times T_i}$. Then their covariance is calculated, denoted as $c_{k,k'}^1$, which can be defined using Eq. (1). In the equation, K represents the number of extracted temporal features, T_i denotes the number of discharge cycles of battery cell i , $f_{i,j,k}$ and $f_{i,j,k'}$ refers to the k th and k' th feature of the feature matrix \mathbf{F}_i for battery cell i in discharge cycle j , N represents the number of battery cells, and $\bar{f}_{..k}$ and $\bar{f}_{..k'}$ represents the expectation of the k th and k' th feature of the feature matrix. It is worth noting that covariance is utilized to determine edges because it can assist in identifying positively and negatively correlated features. Positively correlated features refer to those with the same monotonicity, while negatively correlated features represent those with different monotonicity. By identifying features with the same or different monotonicity and aggregating them in spectral graph convolutional operations, it can be guaranteed that features with the same monotonicity are summed together, and features with different monotonicity are subtracted from each other. In this way, the

monotonicity of the aggregated features can be maximized, potentially improving prediction performance.

$$c_{k,k'}^1 = \frac{1}{\sum_{i=1}^N T_i} \sum_{i=1}^N \sum_{j=1}^{T_i} (f_{i,j,k} - \bar{f}_{..k})(f_{i,j,k'} - \bar{f}_{..k'}) \quad (1)$$

Then, the edge in the graph $\mathbb{G}_{1,+}$ and the graph $\mathbb{G}_{1,-}$ can be determined by using Eq. (2), where ϵ_1 denotes a non-negative threshold for determining the edge. If the covariance $c_{k,k'}^1$ is positive and greater than ϵ_1 , a positive edge between the feature node k and the feature node k' in the graph $\mathbb{G}_{1,+}$ is added, and the corresponding edge $e_{k,k'}^{1,+}$ in the graph $\mathbb{G}_{1,+}$ between these two nodes is assigned as 1. If the covariance $c_{k,k'}^1$ is negative and less than $-\epsilon_1$, a negative edge between the feature node k and the feature node k' in the graph $\mathbb{G}_{1,-}$ is added, and the corresponding edge $e_{k,k'}^{1,-}$ between these two nodes is assigned as -1 .

$$e_{k,k'}^{1,+} = \begin{cases} 1 & \text{if } c_{k,k'}^1 > \epsilon_1, \\ 0 & \text{otherwise.} \end{cases} \quad \text{and} \quad e_{k,k'}^{1,-} = \begin{cases} 0 & \text{otherwise,} \\ -1 & \text{if } c_{k,k'}^1 < -\epsilon_1. \end{cases} \quad (2)$$

The graph construction process will be repeated $\binom{K}{2}$ times until all combinations of two features in the feature matrix have been examined. The constructed graphs $\mathbb{G}_{1,+}$ and $\mathbb{G}_{1,-}$ are respectively represented as $\mathbb{G}_{1,+} = \{\mathbb{V}_1, \mathbb{E}_{1,+}, \mathbf{F}_i\}$ and $\mathbb{G}_{1,-} = \{\mathbb{V}_1, \mathbb{E}_{1,-}, \mathbf{F}_i\}$, where \mathbb{V}_1 refers to a set of feature nodes and \mathbf{F}_i represents the extracted feature matrix. Moreover, $\mathbb{E}_{1,+}$ and $\mathbb{E}_{1,-}$ refer to sets of edges for the graph $\mathbb{G}_{1,+}$ and the graph $\mathbb{G}_{1,-}$, respectively.

The conditional undirected graphs \mathcal{G}_2 also consist of two graphs: the positively connected graph $\mathbb{G}_{2,+}$ and the negatively connected graph $\mathbb{G}_{2,-}$. Mathematically, \mathcal{G}_2 is the second graph set and can be represented as $\mathcal{G}_2 = \{\mathbb{G}_{2,+}, \mathbb{G}_{2,-}\}$. To initialize the graph $\mathbb{G}_{2,+}$ and the graph $\mathbb{G}_{2,-}$, a conditional feature matrix $\mathbf{C}_i \in \mathbb{R}^{(K+1) \times T_i}$ is firstly constructed, where $\mathbf{C}_i = [\mathbf{F}_i, \mathbf{y}_i]$ and \mathbf{y}_i refers to a time series vector of SOH or RUL. Similar to the step used to build the first graph set \mathcal{G}_1 , two different vectors $\mathbf{c}_{i,k}$ and $\mathbf{c}_{i,k'}$ are selected from the conditional feature matrix \mathbf{C}_i and examine their covariance $c_{k,k'}^2$. Then, the edges in the graph $\mathbb{G}_{2,+}$ and the graph $\mathbb{G}_{2,-}$ can be determined using Eq. (3), where ϵ_2 represents a non-negative threshold for edge determination. If the covariance $c_{k,k'}^2$ is positive and greater than ϵ_2 , a positive edge between node k and node k' in the graph $\mathbb{G}_{2,+}$ is added, and the corresponding edge $e_{k,k'}^{2,+}$ in $\mathbb{G}_{2,+}$ is assigned a value of 1. If the covariance $c_{k,k'}^2$ is negative and less than $-\epsilon_2$, a negative edge between node k and node k' in the graph $\mathbb{G}_{2,-}$ is added, the corresponding edge $e_{k,k'}^{2,-}$ between these two nodes is assigned a value of -1 .

$$e_{k,k'}^{2,+} = \begin{cases} 1 & \text{if } c_{k,k'}^2 > \epsilon_2, \\ 0 & \text{otherwise.} \end{cases} \quad \text{and} \quad e_{k,k'}^{2,-} = \begin{cases} 0 & \text{otherwise,} \\ -1 & \text{if } c_{k,k'}^2 < -\epsilon_2. \end{cases} \quad (3)$$

The graph construction process will be repeated $\binom{K+1}{2}$ times until all combinations of two vectors in the conditional feature matrix \mathbf{C}_i have been examined. The constructed graphs $\mathbb{G}_{2,+}$ and $\mathbb{G}_{2,-}$ are respectively represented as $\mathbb{G}_{2,+} = \{\mathbb{V}_2, \mathbb{E}_{2,+}, \mathbf{C}_i\}$ and $\mathbb{G}_{2,-} = \{\mathbb{V}_2, \mathbb{E}_{2,-}, \mathbf{C}_i\}$, where \mathbb{V}_2 refers to a set of nodes, and \mathbf{C}_i represents the conditional feature matrix. Moreover, $\mathbb{E}_{2,+}$ and $\mathbb{E}_{2,-}$ refer to sets of edges for the graph $\mathbb{G}_{2,+}$ and the graph $\mathbb{G}_{2,-}$, respectively.

2.2. Dual spectral graph convolutional operation

To utilize the topological structures of the first graph set \mathcal{G}_1 and the second graph set \mathcal{G}_2 , the dual spectral graph convolutional operation is introduced to handle \mathcal{G}_1 and \mathcal{G}_2 , respectively. The dual spectral graph convolutional operation performed on the graph set \mathcal{G}_n can be written as Eq. (4), where $n = 1$ refers to the first graph set \mathcal{G}_1 and $n = 2$ refers to the second graph set \mathcal{G}_2 . In this equation, $\mathbf{F}_i^{(\omega)} \in \mathbb{R}^{K \times \tau}$ refers to the ω th sampled temporal feature matrix using a sliding window with a step length of one and a window length of τ ; $\mathbf{C}_i^{(\omega)} \in \mathbb{R}^{(K+1) \times \tau}$ refers to the ω th sampled conditional feature matrix using a sliding window with a step length of one and a window length of τ ; \mathcal{F} and \mathcal{F}^{-1} denote

the Fourier transform and its inverse transform; \mathbf{g}_n represents the graph filter from the graph set \mathcal{G}_n ; S is a set that refers to $+$ or $-$, therefore, $\mathbf{g}_{n,+}$ and $\mathbf{g}_{n,-}$ respectively denote the graph filter from the graph $\mathbb{G}_{n,+}$ and the graph $\mathbb{G}_{n,-}$, and \parallel represents the concatenation operator.

$$\mathbf{g}_n \left(\circ_{\mathcal{G}_n} \right) \mathbf{X}_i = \parallel_s^S \mathcal{F}^{-1} \left(\mathcal{F} \left(\mathbf{g}_{n,s} \right) \odot \mathcal{F} \left(\mathbf{X}_i \right) \right) \quad (4)$$

$$\mathbf{X}_i = \mathbf{F}_i^{(\omega)}, \text{ if } n = 1; \mathbf{X}_i = \mathbf{C}_i^{(\omega)}, \text{ if } n = 2; s = + \text{ and } -$$

The graph Fourier transform for the graph filter $\mathbf{g}_{n,s}$ can be represented as $U_{n,s}^T \mathbf{g}_{n,s}$, where $U_{n,s}$ refers to the eigenvectors of the Laplacian matrices. The graph Fourier transform for the sampled matrix can be written as $U_{n,s}^T \mathbf{X}_i$. By substituting the eigenvector $U_{n,s}$ into Eq. (4), Eq. (5) can be obtained, where $\Lambda_{n,s}$ is the vector that stores the eigenvalues of the Laplacian matrix, and $\psi_{n,s}$ represents a collection of parameters provided by the graph filter $\mathbf{g}_{n,s}$.

$$\parallel_s^S U_{n,s} \left(U_{n,s}^T \mathbf{g}_{n,s} \odot U_{n,s}^T \mathbf{X}_i \right) = \parallel_s^S U_{n,s} \mathbf{g}_{n,s}^{\psi_{n,s}} (\Lambda_{n,s}) U_{n,s}^T \mathbf{X}_i \quad (5)$$

For the graph $\mathbb{G}_{n,s}$, the normalized Laplacian matrix $\mathcal{L}_{n,s}$ is given by Eq. (6), where \mathbf{I} is the identity matrix; $\mathbb{A}_{n,s}$ denotes the adjacency matrix of the graph $\mathbb{G}_{n,s}$, and the elements of this adjacency matrix are derived from the edge set $\mathbb{E}_{n,s}$.

$$\mathcal{L}_{n,s} = \mathbf{I} - D_{n,s}^{-1/2} \mathbb{A}_{n,s} D_{n,s}^{-1/2} \quad (6)$$

Moreover, $D_{n,s}$ refers to the degree matrix of $\mathbb{A}_{n,s}$; the diagonal entry of the degree matrix can be written as Eq. (7), where $e_{k,k'}^{1,+}$ and $e_{k,k'}^{1,-}$ are obtained from Eq. (2); and $e_{k,k'}^{2,+}$ and $e_{k,k'}^{2,-}$ are obtained from Eq. (3).

$$\begin{cases} d_{k,k'}^{1,+} = \sum_{k'=1}^K e_{k,k'}^{1,-} & \text{and} \quad d_{k,k'}^{1,-} = \sum_{k'=1}^K |e_{k,k'}^{1,-}| \\ d_{k,k'}^{2,+} = \sum_{k'=1}^{K+1} e_{k,k'}^{2,+} & \text{and} \quad d_{k,k'}^{2,-} = \sum_{k'=1}^{K+1} |e_{k,k'}^{2,-}| \end{cases} \quad (7)$$

Because the Laplacian matrix $\mathcal{L}_{n,s}$ is a real and symmetric in nature, the eigendecomposition of this Laplacian matrix can be expressed as Eq. (8).

$$\mathcal{L}_{n,s} = U_{n,s} \Lambda_{n,s} U_{n,s}^{-1} = U_{n,s} \Lambda_{n,s} U_{n,s}^T \quad (8)$$

By inserting Eq. (8) into Eq. (5), Eq. (9) can be derived.

$$\mathbf{g}_n \left(\circ_{\mathcal{G}_n} \right) \mathbf{X}_i = \parallel_s^S \mathbf{g}_{n,s}^{\psi_{n,s}} (\mathcal{L}_{n,s}) \mathbf{X}_i \quad (9)$$

Previous research indicated that the first-order Chebyshev polynomials is capable of reducing the computational cost of the spectral graph convolution (Hammond, Vandergheynst, & Gribonval, 2011). By using the Chebyshev polynomials, Eq. (9) can be rewritten as Eq. (10), where \mathcal{C}_p is the p th order Chebyshev polynomial; $\tilde{\mathcal{L}}_{n,s}$ represents the scaled Laplacian matrix.

$$\parallel_s^S \left(\sum_{p=0}^1 \psi_{n,s}^p \mathcal{C}_p (\tilde{\mathcal{L}}_{n,s}) \mathbf{X}_i \right) = \parallel_s^S \left(\left(\psi_{n,s} \left(D_{n,s}^{-1/2} \mathbb{A}_{n,s} D_{n,s}^{-1/2} + \mathbf{I} \right) \right) \mathbf{X}_i \right) \quad (10)$$

To enable a feature to aggregate its adjacent features along with itself, this work incorporates a self-connection in the positively connected graphs $\mathbb{G}_{n,+}$ for all n . The updated adjacency matrix can be denoted as $\tilde{\mathbb{A}}_{n,s} = \mathbb{A}_{n,s} + \mathbf{I}$ when $s = +$, and $\tilde{\mathbb{A}}_{n,s} = \mathbb{A}_{n,s}$ when $s = -$. In this work, a self-connection is not included in the negatively connected graphs $\mathbb{G}_{n,-}$ for all n because the self-aggregation process is already completed in $\mathbb{G}_{n,+}$. Next, $\psi_{n,s}$ is rewritten in a matrix format, and the dual graph convolutional operations can be expressed as Eq. (11).

$$\mathbf{g}_n \left(\circ_{\mathcal{G}_n} \right) \mathbf{X}_i = \parallel_s^S \tilde{\mathbb{A}}_{n,s} \mathbf{X}_i \psi_{n,s} \quad (11)$$

In Eq. (10), $\tilde{\mathbb{A}}_{n,s}$ can be written as $\tilde{D}_{n,s}^{-1/2} \tilde{\mathbb{A}}_{n,s} \tilde{D}_{n,s}^{-1/2}$, where $\tilde{D}_{n,s}$ denotes the degree matrix of the updated adjacency matrix $\tilde{\mathbb{A}}_{n,s}$. Additionally, $\psi_{n,s} \in \mathbb{R}^{\tau \times \tau'}$ represents the graph filter parameters in matrix format. The dual spectral graph convolutional operation is introduced based on the dual spectral convolutional operation in Eq. (11). To enhance the robustness of this layer, a bias weight vector $\mathbf{b}_{n,s}$ and an activation function σ is incorporated. The output of a single dual spectral graph

convolutional layer is expressed as Eq. (12), where σ represents the activation function and $\mathbf{b}_{n,s}$ denotes the bias weight vector.

$$\parallel_s^S \mathbf{H}_i^{n,s} = \parallel_s^S \sigma \left(\tilde{\mathbb{A}}_{n,s} \mathbf{X}_i \psi_{n,s} + \mathbf{b}_{n,s} \right) \quad (12)$$

In summary, there are two graph sets (\mathcal{G}_1 and \mathcal{G}_2) are initialized, each consisting of two graphs when $n = 1, 2$ and $s = +, -$. The dual spectral graph convolutional operation is then performed on each graph set. As a result, there will be four outputs generated by the proposed dual spectral graph convolutional operation. These outputs can be mathematically represented as $\mathbf{H}_i^{1,+}$, $\mathbf{H}_i^{1,-}$, $\mathbf{H}_i^{2,+}$, and $\mathbf{H}_i^{2,-}$.

Fig. 1 shows an example of the introduced dual spectral graph convolutional operation for the graph sets \mathcal{G}_1 and \mathcal{G}_2 with one single layer for the purpose of illustration. First of all, two different vectors are selected from the feature matrix \mathbf{F}_i to examine their covariance. The edges of the graph $\mathbb{G}_{1,+}$ and the graph $\mathbb{G}_{1,-}$ in the graph set \mathcal{G}_1 are determined based on the covariance and the threshold ϵ_1 . Likewise, two different vectors are selected from the conditional feature matrix \mathbf{C}_i to examine their covariance. The edges of the graph $\mathbb{G}_{2,+}$ and the graph $\mathbb{G}_{2,-}$ in the graph set \mathcal{G}_2 are determined based on the covariance and the threshold ϵ_2 . Next, the dual spectral graph convolution operation is utilized twice to respectively handle the topological structure of the graphs in the graph set \mathcal{G}_1 and the graph set \mathcal{G}_2 . For the positively connected graph $\mathbb{G}_{n,+}$, $\forall n$, a positive self-connected adjacency matrix $\tilde{\mathbb{A}}_{n,+}$, $\forall n$, is generated, and all the vectors in the sampled feature matrix or the sampled conditional feature matrix are aggregated and projected based on the constructed matrix $\tilde{\mathbb{A}}_{n,+}$, $\forall n$. With respect to the negatively correlated graph $\mathbb{G}_{n,-}$, $\forall n$, a negative adjacency matrix $\tilde{\mathbb{A}}_{n,-}$, $\forall n$, is generated, and all the vectors in the sampled feature matrix or the sampled conditional feature matrix are aggregated and projected based on the matrix $\tilde{\mathbb{A}}_{n,-}$, $\forall n$. The outputs after performing the proposed dual spectral graph convolutional operation are denoted as $\mathbf{H}_i^{1,+}$, $\mathbf{H}_i^{1,-}$, $\mathbf{H}_i^{2,+}$, and $\mathbf{H}_i^{2,-}$.

2.3. Dilated convolutional operation

The dilated convolutional operation is the most critical component of the temporal convolutional network, which has been demonstrated to outperform canonical recurrent neural networks such as RNN and LSTM in revealing temporal correlations in time series data (Bai et al., 2018). Therefore, the dilated convolutional operation is employed to more effectively consider the temporal correlation of the aggregated and projected features $\mathbf{H}_i^{1,+}$, $\mathbf{H}_i^{1,-}$, $\mathbf{H}_i^{2,+}$, and $\mathbf{H}_i^{2,-}$ after performing the proposed dual spectral graph convolutional operation. The dilated convolutional operation is similar to the typical convolutional operation that uses filter matrices sweeping over the entire input matrix. The output of the dilated convolutional operation \mathbb{D} can be mathematically represented as Eq. (13), where $\mathbf{H}_{i,\gamma}^{n,s}$ is the γ -th part of the resulting tensors $\mathbf{H}_i^{n,s}$ from the dual spectral graph convolutional operation. If $n = 1$, $\mathbf{H}_{i,\gamma}^{n,s} \in \mathbb{R}^{F \times K}$; if $n = 2$, $\mathbf{H}_{i,\gamma}^{n,s} \in \mathbb{R}^{F \times (K+1)}$. Here, K is the total number of columns in the extracted feature matrix, F refers to the filter size of the dilated convolutional operation, $h_{a,b}^{(\gamma)}$ represents one element in the matrix $\mathbf{H}_{i,\gamma}^{n,s}$, $w_{a,b,\delta}$ is one of the weight elements in the δ -th filter matrix, and D represents the dilation factor. This work sets $\delta = 1, \dots, \Delta$, where Δ refers to the number of filters provided in the dilated convolutional operation, and sets $\gamma = 1, \dots, \Gamma$, where $\Gamma = \tau' + F - 1$ refers to the total number of input matrices multiplied by the filter matrices. Therefore, the output of the dilated convolutional operation is a Γ -by- Δ matrix.

$$\left(\mathbf{H}_{i,\gamma}^{n,s} \otimes_D \mathbb{D} \right) := o_{\gamma,\delta} = \sum_a \sum_b h_{a,b}^{(\gamma)} \cdot w_{a,b,\delta} \quad (13)$$

Moreover, the dilated convolutional filter matrix is a matrix with sparsity, which can be mathematically represented as Eq. (14), where the weight is trainable if $b = 1 + \pi \cdot D$ and $\pi = 1, \dots, \Pi$, otherwise the weight is not trainable and equals zero constantly. In addition,

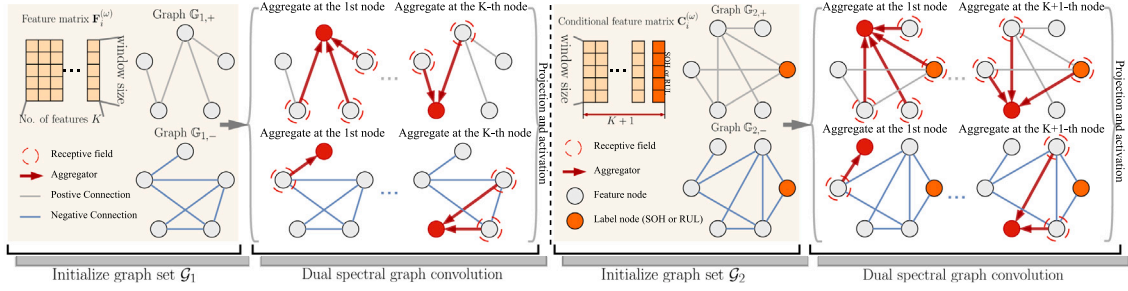


Fig. 1. The dual spectral graph convolutional operation for the graph set \mathcal{G}_1 and \mathcal{G}_2 with one single layer.

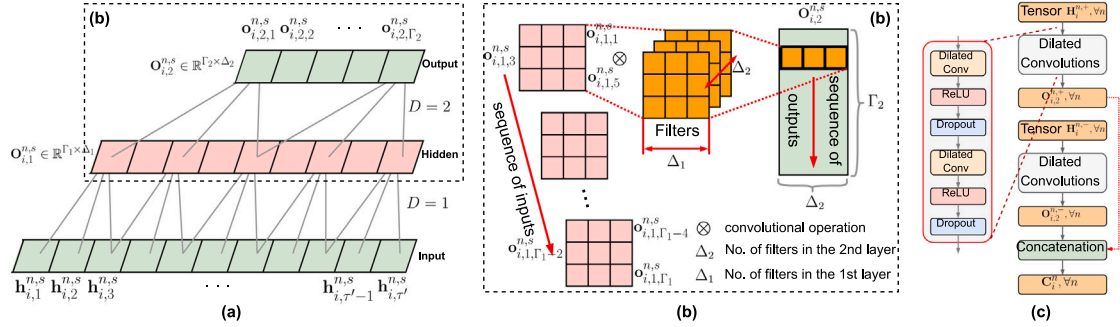


Fig. 2. (a) Two dilated convolutional operations with dilation factors $D = 1, 2$ and filter size $F = 3, 5$; (b) Details about the second dilated convolutional operation; (c) The architecture of multiple dilated convolutional operations.

the relation between the dilation factor D and the filter size F can be mathematically represented as $1 + \Pi \cdot D = F$;

$$w_{a,b,\delta} = \begin{cases} w_{a,b,\delta} & \text{if } b = 1 + \pi \cdot D, \\ 0 & \text{otherwise.} \end{cases} \quad (14)$$

Fig. 2 shows an example of two continuous dilated convolutional operations for illustration purposes. The first dilated convolutional operation has the dilation factor $D_1 = 1$ and the filter size $F_1 = 3$; and the second dilated convolutional operation has the dilation factor $D_2 = 3$ and the filter size $F_2 = 5$. The first dilated convolutional operation is performed on the tensors $\mathbf{H}_i^{n,s}, \forall n, s$ generated by the dual spectral graph convolutional operations. After performing the first dilated convolutional operations, the ReLU activation function and the dropout function are adopted, then the resulting tensor can be represented as $\mathbf{O}_{i,1}^{n,s} \in \mathbb{R}^{\Gamma_1 \times \Delta_1}$, where Δ_1 refers to the amount of filters in the first dilated convolutional layer and Γ_1 is the reduced time length after performing the first dilated convolutional operation and Γ_1 equals $\tau' + F_1 - 1$. Next, the second dilated convolutional operation is performed on the tensor $\mathbf{O}_{i,1}^{n,s}, \forall n, s$ generated by the first dilated convolutional operation. After performing the second dilated convolutional operations, the ReLU activation function and the dropout function are implemented, then the resulting tensor can be represented as $\mathbf{O}_{i,2}^{n,s} \in \mathbb{R}^{\Gamma_2 \times \Delta_2}$. Here, Δ_2 refers to the amount of filters in the second dilated convolutional operation and Γ_2 is the reduced time length after performing the second dilated convolutional operation and Γ_2 equals $\Gamma_1 + F_2 - 1$. Next, $\mathbf{O}_{i,1}^{n,+}$ and $\mathbf{O}_{i,1}^{n,-}$ are concatenated as a vector \mathbf{O}_i^n for all n , which can be mathematically written as Eq. (15), where \mathcal{C} represents the number of dilated convolutional operations has been used.

$$\mathbf{O}_i^n = \parallel_s^S \mathbf{O}_{i,\mathcal{C}}^{n,s}, s = + \text{ and } - \quad (15)$$

2.4. Training the condition graph convolutional network with dilated convolutional operations

In summary, the proposed method initializes two sets of graphs: the undirected graph set \mathcal{G}_1 and the conditional undirected graph set \mathcal{G}_2 .

Each graph set consists of two undirected graphs representing positively and negatively connected graphs. The purpose of \mathcal{G}_1 is to consider the correlation among features, while \mathcal{G}_2 considers the correlation between features and SOH or RUL. To handle the topological structures of these two graph sets, the method introduces the dual spectral graph convolutional operation and adopts the dilated convolutional operation. The dual spectral graph convolutional operation is applied to \mathcal{G}_1 and \mathcal{G}_2 , and the resulting tensors for the battery unit i are denoted as \mathbf{O}_i^1 and \mathbf{O}_i^2 , respectively. The dilated convolutional operation allows effective consideration of the temporal correlation among the aggregated features provided by the dual spectral graph convolutional operation. To train the proposed conditional GCN with dilated convolutional operation, the resulting tensors \mathbf{O}_i^n for all n are flattened and passed through a fully connected (FC) layer for SOH estimations and RUL predictions. This process can be mathematically represented as Eq. (16),

$$\hat{y}_{i,n}^{(\omega)} = \sigma(\mathbf{W}_n \mathbf{O}_i^n + \mathbf{b}_n), n = 1, 2 \quad (16)$$

where $\hat{y}_{i,n}^{(\omega)}$ refers to the estimated SOH or predicted RUL provided by the n th graph set for battery unit i in the ω th sample, σ is the activation function, \mathbf{W}_n denotes the weight matrix of the FC layer for the n th graph set, and \mathbf{b}_n denotes the bias vector of the FC layer for the n th graph set. Therefore, the training loss \mathcal{L}_1 for the first graph set \mathcal{G}_1 and the training loss \mathcal{L}_2 for the second graph set \mathcal{G}_2 can be written as shown in Eq. (17), where I refers to the total number of battery units used for training, and Ω denotes the total number of discharge cycles used for training.

$$\mathcal{L}_1 = \sum_{i=1}^I \sum_{\omega=1}^{\Omega} \frac{1}{I \times \Omega} \left(\hat{y}_{i,1}^{(\omega)} - y_{i,1}^{(\omega)} \right)^2, \mathcal{L}_2 = \sum_{i=1}^I \sum_{\omega=1}^{\Omega} \frac{1}{I \times \Omega} \left(\hat{y}_{i,2}^{(\omega)} - y_{i,2}^{(\omega)} \right)^2 \quad (17)$$

The dual spectral graph convolutional operation performed on the second graph set \mathcal{G}_2 involves the SOH and RUL in the conditional feature matrix. While the second graph set \mathcal{G}_2 can be used for training, it cannot be used for predicting because the true SOH and RUL are not available during prediction. Therefore, only the first graph set \mathcal{G}_1 can be used for prediction. To leverage the correlation between features and

SOH or RUL from the second graph set \mathcal{G}_2 during prediction, this work introduces a KL-divergence loss to minimize the divergence between \mathbf{O}_i^1 and \mathbf{O}_i^2 , which can be written as Eq. (18).

$$\mathcal{L}_{KL} = \sum_{i=1}^I \sum_{\omega=1}^{\Omega} \mathbb{KL} \left(p_{\Psi_1}(\mathbf{O}_i^1 | \mathbf{F}_i^{(\omega)}) || p_{\Psi_2}(\mathbf{O}_i^2 | \mathbf{C}_i^{(\omega)}) \right) \quad (18)$$

In this equation, $p_{\Psi_1}(\mathbf{O}_i^1 | \mathbf{F}_i^{(\omega)})$ represents the probability distribution of \mathbf{O}_i^1 given the sampled feature matrix $\mathbf{F}_i^{(\omega)}$. Here, Ψ_1 denotes the collection of parameters in the first dual spectral graph convolutional operation with dilated convolutional operation performed on the first graph set \mathcal{G}_1 . Similarly, $p_{\Psi_2}(\mathbf{O}_i^2 | \mathbf{C}_i^{(\omega)})$ denotes the probability distribution of \mathbf{O}_i^2 given the sampled conditional feature matrix $\mathbf{C}_i^{(\omega)}$, where Ψ_2 represents the collection of parameters in the second dual spectral graph convolutional operation with dilated convolutional operation performed on the second graph set \mathcal{G}_2 . Since it is infeasible to directly determine the conditional distribution of \mathbf{O}_i^1 and \mathbf{O}_i^2 , it is commonly assumed that these two distributions in the KL-divergence follow normal distributions (Kusner, Paige, & Hernández-Lobato, 2017; Wei & Wu, 2022a). This assumption can be written as Eq. (19), where $\mu_{i,\omega}^1$ and $\Sigma_{i,\omega}^1$ represent the resulting tensors from the first dilated convolutional operation and are used to sample \mathbf{O}_i^1 , referring to the mean and variance of \mathbf{O}_i^1 . Similarly, $\mu_{i,\omega}^2$ and $\Sigma_{i,\omega}^2$ represent the resulting tensors from the second dilated convolutional operation and are used to sample \mathbf{O}_i^2 , referring to the mean and variance of \mathbf{O}_i^2 .

$$p_{\Psi_1}(\mathbf{O}_i^1 | \mathbf{F}_i^{(\omega)}) \sim \mathcal{N}(\mathbf{O}_i^1; \mu_{i,\omega}^1, \Sigma_{i,\omega}^1) \quad p_{\Psi_2}(\mathbf{O}_i^2 | \mathbf{C}_i^{(\omega)}) \sim \mathcal{N}(\mathbf{O}_i^2; \mu_{i,\omega}^2, \Sigma_{i,\omega}^2) \quad (19)$$

By utilizing the reparameterization trick (Huang, Wu, Wang, & Tan, 2013; Kingma & Welling, 2013), the KL-divergence can be expressed as Eq. (20), where d represents the dimensionality of the learned deep-level representations, and $\text{tr}(\cdot)$ indicates the trace of a matrix.

$$\mathcal{L}_{KL} = \sum_{i=1}^I \sum_{\omega=1}^{\Omega} \frac{1}{2} \left(\text{tr} \left[\left(\Sigma_{i,\omega}^2 \right)^{-1} \Sigma_{i,\omega}^1 \right] + \left(\mu_{i,\omega}^2 - \mu_{i,\omega}^1 \right)^T \left(\Sigma_{i,\omega}^2 \right)^{-1} \left(\mu_{i,\omega}^2 - \mu_{i,\omega}^1 \right) - d - \log \left(|\Sigma_{i,\omega}^2| / |\Sigma_{i,\omega}^1| \right) \right) \quad (20)$$

Then, the overall training loss consists of a triplet loss, which includes a KL-divergence loss and two prediction losses. The triplet loss can be represented as Eq. (21).

$$\mathcal{L}_{triplet} = \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_{KL} \quad (21)$$

Next, the obtained training loss is utilized to train the proposed method. The training process of the proposed method involves two training steps. In the first training step, the collection of parameters Ψ_1 and Ψ_2 in the first and second spectral graph convolutional operations with dilated convolutional operation, as well as the parameters in the FC layers, can be updated using the gradient descent method. The updating process can be described as Eq. (22), where α represents the learning rate.

$$(\Psi_n, \mathbf{W}_n, \mathbf{b}_n) \leftarrow (\Psi_n, \mathbf{W}_n, \mathbf{b}_n) - \alpha \cdot \left(\frac{\nabla \mathcal{L}_{triplet}}{\nabla \Psi_n}, \frac{\nabla \mathcal{L}_{triplet}}{\nabla \mathbf{W}_n}, \frac{\nabla \mathcal{L}_{triplet}}{\nabla \mathbf{b}_n} \right), \forall n \quad (22)$$

In the second training step, the single training loss \mathcal{L}_1 is used and the learned Ψ_1 from the first training step is used to retrain the parameters \mathbf{W}_n and \mathbf{b}_n in the corresponding FC layer. This process can be mathematically represented as Eq. (23).

$$(\mathbf{W}_n, \mathbf{b}_n) \leftarrow (\mathbf{W}_n, \mathbf{b}_n) - \alpha \cdot \left(\frac{\nabla \mathcal{L}_{triplet}}{\nabla \mathbf{W}_n}, \frac{\nabla \mathcal{L}_{triplet}}{\nabla \mathbf{b}_n} \right), n = 1 \quad (23)$$

After completing the training process, the trained first spectral graph convolutional operation with dilated convolutional operations and the retrained FC layer are used to estimate the SOH and predict the RUL of batteries.

In summary, the proposed method employs two dual spectral graph convolutional networks with dilated convolutional operations. The inputs of the first dual spectral graph convolutional network include

temporal features extracted from the condition monitoring data and the initialized graph set \mathcal{G}_1 , while its output is a predicted SOH/RUL. The inputs of the second dual spectral graph convolutional network are the extracted temporal features, SOH/RUL, and the initialized graph set \mathcal{G}_2 , yielding another predicted SOH/RUL as the output. During the training process, both dual spectral graph convolutional networks with dilated convolutional operations are used to minimize prediction errors and reduce the distance between the two feature spaces extracted from these networks using the triplet training loss. In contrast, during the testing phase, only the first dual spectral graph convolutional network with dilated convolutional operations is used for SOH/RUL predictions.

Fig. 3 illustrates the two steps used to train the conditional GCN. In the first training step, condition monitoring data is used to initialize two graph sets, \mathcal{G}_1 and \mathcal{G}_2 . \mathcal{G}_1 captures the correlation among features, while \mathcal{G}_2 reflects the correlation between features and the SOH or RUL. For each graph set, the proposed dual spectral graph convolutional operation is employed to aggregate data with high similarity, resulting in four tensors: $\mathbf{H}_i^{1,+}$, $\mathbf{H}_i^{1,-}$, $\mathbf{H}_i^{2,+}$, and $\mathbf{H}_i^{2,-}$. These four tensors are subsequently fed into the dilated convolutional operation to consider the temporal correlation of the aggregated data, resulting in another four tensors: $\mathbf{O}_{i,\mathcal{G}}^{1,+}$, $\mathbf{O}_{i,\mathcal{G}}^{1,-}$, $\mathbf{O}_{i,\mathcal{G}}^{2,+}$, and $\mathbf{O}_{i,\mathcal{G}}^{2,-}$. Next, $\mathbf{O}_{i,\mathcal{G}}^{1,+}$ and $\mathbf{O}_{i,\mathcal{G}}^{1,-}$ are concatenated into a single vector, and $\mathbf{O}_{i,\mathcal{G}}^{2,+}$ and $\mathbf{O}_{i,\mathcal{G}}^{2,-}$ are also concatenated into another single vector to aggregate the information from both positive and negative graphs in each graph set. Next, these two single vectors are used to generate $\mu_{i,\omega}^1$, $\Sigma_{i,\omega}^1$ to sample \mathbf{O}_i^1 , and generate $\mu_{i,\omega}^2$, $\Sigma_{i,\omega}^2$ to sample \mathbf{O}_i^2 . The sampled \mathbf{O}_i^1 and \mathbf{O}_i^2 are then fed into two different FC layers to obtain the predicted SOH or RUL, which are $\hat{y}_{i,1}^{(w)}$ and $\hat{y}_{i,2}^{(w)}$. The triplet loss is employed to train the entire framework and update all trainable parameters, including two prediction losses and one KL-divergence loss. In the second training step, all parameters in the dual spectral graph convolutional operation with the dilated convolutional operation for \mathcal{G}_1 are frozen and transferred. Only the first prediction loss, \mathcal{L}_1 , is used to retrain the parameters in the corresponding FC layer. During prediction, only the trained dual spectral graph convolutional operation with the dilated convolutional operation for \mathcal{G}_1 and the retrained FC layer are used to make SOH and RUL predictions.

3. Case study I

3.1. Data description

The battery dataset released by the NASA Ames Prognostics Center of Excellence (PCoE) (Saha & Goebel, 2007) was used to demonstrate the effectiveness of the conditional GCN. This dataset includes three subsets. Subset 1 includes condition monitoring data collected from four lithium-ion batteries (Battery No. 5, No. 6, No. 7, and No. 18), subset 2 includes data obtained from four batteries (Battery No. 29, No. 30, No. 31, No. 32), and subset 3 includes data obtained from three batteries (Battery No. 25, No. 26, and No. 27). These lithium-ion batteries underwent three distinct operational conditions: charging, discharging, and impedance. In the charging and discharging cycles, current, voltage, and temperature data were collected. For all three subsets, the charging process was executed using a Constant Current (CC) mode at a rate of 1.5 A until the voltage reached 4.2 V, and then switched to a Constant Voltage (CV) mode until the current dropped below 20 mA.

For subset 1, the discharge process was performed with a CC mode of 2 A until the voltage of the batteries reached 2.7 V for Battery No. 5, 2.5 V for Battery No. 6, 2.2 V for Battery No. 7, and 2.5 V for Battery No. 18. For subset 2, the discharge procedure was performed at a CC level of 4 A until the voltage reached 2.0 V, 2.2 V, 2.5 V, and 2.7 V for Batteries No. 29, No. 30, No. 31, and No. 32, respectively. This work used the data collected during the discharge cycles only. Fig. 4 shows voltage, current, and temperature readings during the discharge cycles of Battery No. 5.

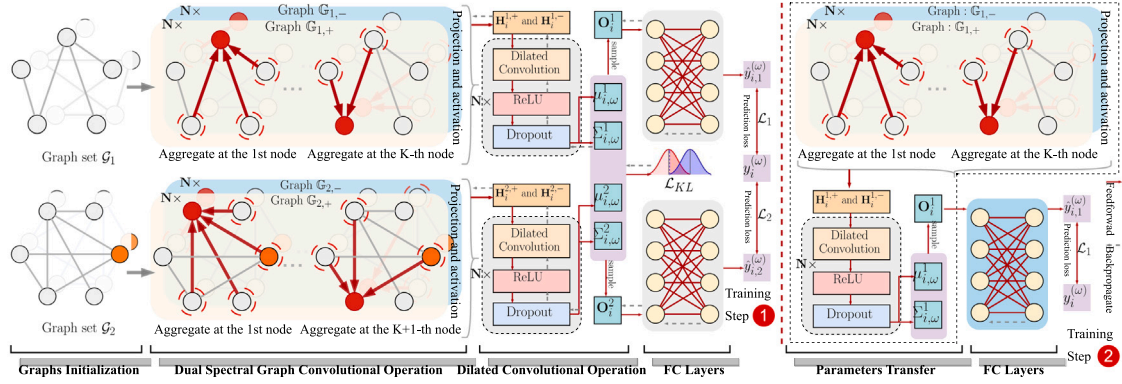


Fig. 3. Two training steps used to train the proposed conditional graph convolutional network with dilated convolutional operations.

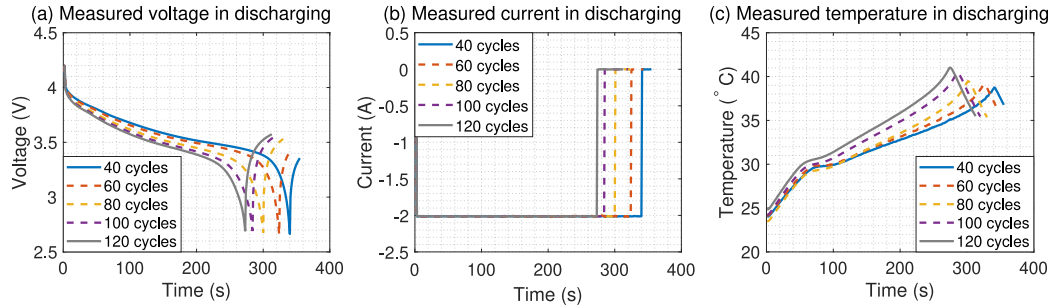


Fig. 4. The voltage, current, and temperature measurements over time in discharge cycles for battery No. 5.

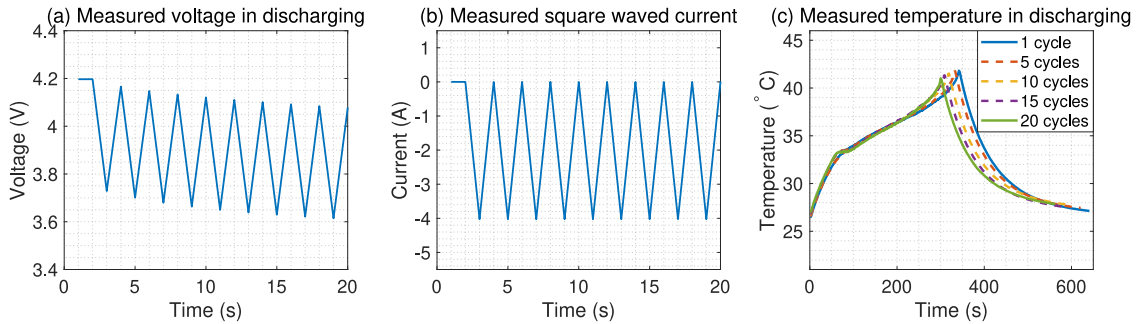


Fig. 5. (a) shows a part of the measured voltage in the first discharge cycle for battery No. 25; (b) shows a part of the measured squared waved current in the first discharge cycle for battery No. 25; (c) shows the measured temperature with respect to different discharge cycles for battery No. 25.

For subset 3, the discharge process was performed under a square wave loading profile with a frequency of 0.05 Hz and an amplitude of 4 A, with a duty cycle of 50%. The process continued until the voltage reduced to 2.0 V, 2.2 V, 2.5 V, and 2.7 V for batteries No. 25, No. 26, and No. 27, respectively. Fig. 5(a) shows a part of the measured voltage in the first discharge cycle for battery No. 25; Fig. 5(b) shows a part of the measured squared waved current in the first discharge cycle for battery No. 25; Fig. 5(c) shows the measured temperature with respect to different discharge cycles for battery No. 25. From this figure, it can be observed that the voltage does not monotonically decrease due to the 0.05 Hz square wave loading profile, and such a loading profile may bring some randomness and difficulty in SOH estimations. In addition, it can also be observed that the trajectory of the measured data alternates with the increase in the number of discharge cycles. For example, as depicted in Fig. 5(c), the time required to reach the maximum temperature in the initial discharge cycles is longer than the time needed to reach the maximum temperature in 20 discharge cycles. Therefore, the extracted temporal features can still be utilized when predicting.

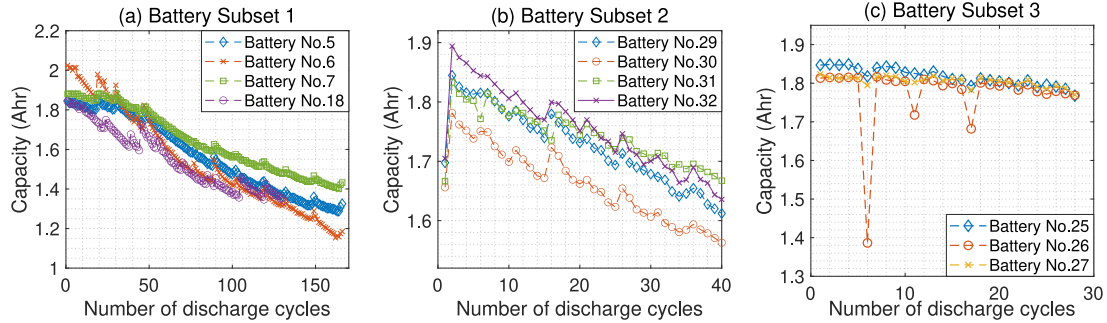
Table 1 shows the operating conditions for three different subsets in charge and discharge cycles and the corresponding battery indices. In summary, subset 1 and subset 2 performed the discharging process under a constant loading profile, and subset 3 performed the discharging process under a square wave loading profile. In addition, with respect to all the battery datasets, the charging process was conducted under a CC mode.

The rate of degradation of lithium-ion batteries increased through repeated cycles of charging and discharging, leading to capacity loss or capacity fading. Fig. 6 shows the degradation trajectories of capacity for batteries in three battery subsets. With respect to batteries listed in subset 1, run-to-failure tests were conducted, where the experiments ended under the condition that the battery capacity decreased by 30%. The capacity of these batteries was 2 Ahr at its peak, with an end-of-life (EOL) capacity of 1.4 Ahr. With respect to batteries listed in subset 2, run-to-failure tests were conducted, where the experiments were terminated under the condition that the battery capacity decreased by 15%. The capacity of these batteries was 2 Ahr at its peak, with an end-of-life capacity of 1.7 Ahr. In this case study, the SOH is estimated

Table 1

The operating conditions for three different datasets in charge and discharge cycles and the corresponding battery indices.

Subset no.	Charging	Discharging	Battery index
1	CC mode at 1.5 A	Constant load at 2 A	Battery No. 5, No. 6, No. 7, No. 18
2	CC mode at 1.5 A	Constant load at 4 A	Battery No. 29, No. 30, No. 31, No. 32
3	CC mode at 1.5 A	Square wave load of 4 A amplitude	Battery No. 25, No. 26, No. 27

**Fig. 6.** The degradation trajectories of capacity for batteries in three battery subsets.

and the RUL is predicted for batteries in subset 1 and subset 2, and the SOH is estimated only for batteries in subset 3 because run-to-failure tests were not conducted on batteries in subset 3. Moreover, four-fold cross-validation was performed for the batteries in subset 1 and subset 2, while three-fold cross-validation was performed for the batteries in subset 3 to thoroughly evaluate the effectiveness of the proposed method on these batteries.

3.2. Feature extraction and network structure

In real-world applications, condition monitoring data in charge cycles are often not available. Therefore, we extracted features from voltage, current, and temperature measurements collected during discharge cycles only. These features include the time to reach minimum voltage, discharge duration under constant or variable current conditions, time to reach maximum temperature, voltage decrease rate, and temperature increase rate. These features have been demonstrated to be effective in tracking the capacity trajectory of a battery (Audin et al., 2021; Wei & Wu, 2022b). For example, the voltage decrease rate is calculated as the voltage drop divided by the discharge time, while the temperature increase rate is determined by dividing the temperature increment by the discharge time.

Next, the extracted features were fed into the conditional GCN with dilated convolutional operations to estimate the SOH and RUL. The details on the network structure and hyperparameters used in the case study for all three battery subsets are provided in Table 2. These hyperparameters were determined using the grid search method. As shown in this table, Batch refers to the batch size of 100. $K = 5$ represents the number of extracted features. $\tau = 20$ represents the window size of the sampling window. $\tau' = 100$ refers to the dimensionality after projection in the dual spectral graph convolutional operation. Γ_i and Γ_j respectively refer to the reduced time length after performing the i th and the j th dilated convolutional operations. In addition to these aforementioned parameters, the filter sizes in the dilated convolutional layers are $F_1 = F_2 = F_3 = 10$. The amount of filters in the dilated convolutional layers are $\Delta_1 = \Delta_2 = \Delta_3 = 100$, and the dilation factors are $D_1 = 1$, $D_2 = 2$, and $D_3 = 4$. Moreover, the learning rate is set as 5×10^{-3} , the threshold level ϵ is set as zero for simplification, and the Adam optimizer is adopted to train the proposed conditional graph convolutional network with dilated convolutional operation.

3.3. SOH estimation

Fig. 7 shows the SOH estimations for Battery No. 5, No. 6, No. 7, and No. 18 in subset 1, and Battery No. 29, No. 30, No. 31, and No. 32 in subset 2. The SOH estimations for batteries in subset 1 start from the 20th discharge cycle since subset 1 has more discharge cycles. The SOH estimations for batteries in subset 2 start from the 5th discharge cycle since subset 2 has fewer discharge cycles. From Fig. 7, it can be observed that the proposed method is capable of estimating the SOH of lithium-ion batteries with high accuracy. For example, for Battery No. 5, the estimated SOH matches the true SOH of 0.915 when 31 discharge cycles have been observed. Likewise, during the 20th discharge cycle of Battery No. 29, the estimated SOH is 0.906, matching closely with the true SOH of 0.907.

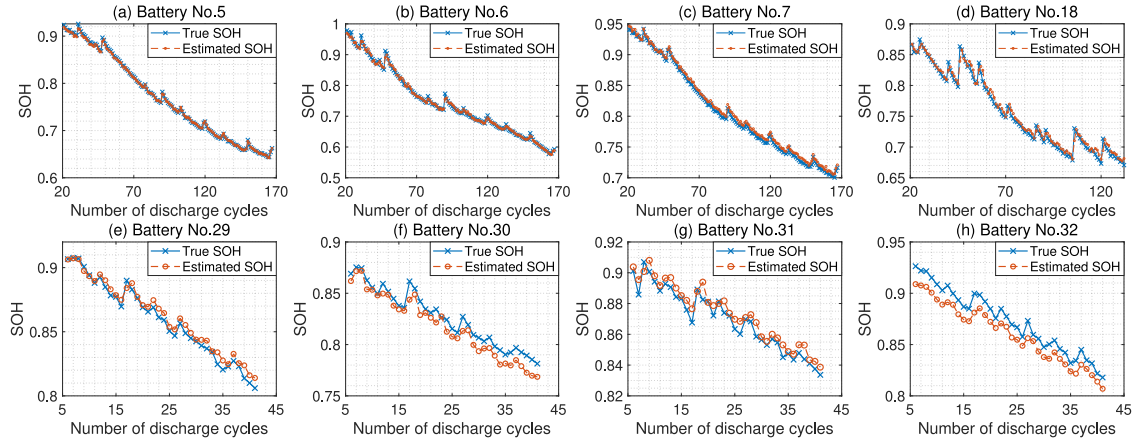
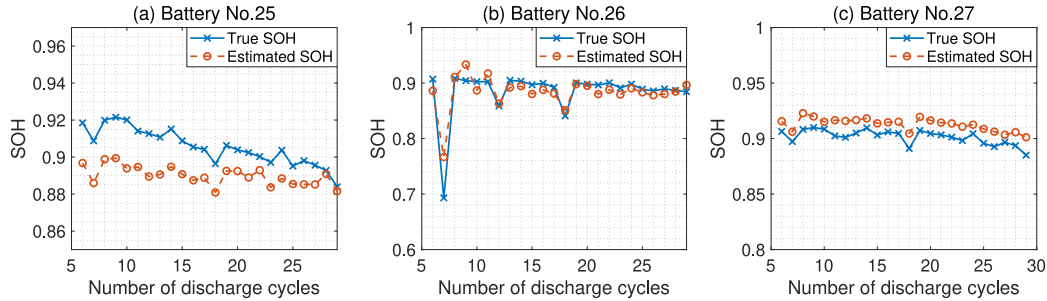
Fig. 8 shows the SOH estimations for Battery No. 25, No. 26, and No. 27 in subset 3. The SOH estimations for batteries in subset 3 start from the 5th discharge cycles since these batteries have fewer discharge cycles. Fig. 8 shows that the proposed method accurately estimates the SOH of batteries subjected to the square wave load in discharge cycles. For example, when 10 discharge cycles have been observed, the estimated SOH for Battery No. 25 is 0.895, while the actual SOH is 0.914. Although there is a gap between the estimated SOH trajectory and the true SOH trajectory for Battery No. 25 and Battery No. 27, the proposed method is still capable of tracking the fluctuation of the trajectories for these two batteries. There are two reasons for the estimation error. The first reason is that the square wave load in discharge cycles introduces randomness in SOH estimations. The second reason is that the experiments on these batteries terminated at the very early degradation stage, as a result, this subset provides limited training data.

To further illustrate the performance of the conditional GCN with dilated convolutional operations, an ablation study was conducted. Table 3 lists the methods used in this ablation study: CGCN-DCO refers to the proposed method, GCN-DCO represents the graph convolutional network with dilated convolutional operations, and CGCN denotes the conditional graph convolutional network. The comparison between CGCN-DCO and GCN-DCO aims to demonstrate the effectiveness of the proposed conditional graphs, while the comparison between CGCN-DCO and CGCN aims to showcase the effectiveness of the dilated convolutional operation. It should be noted that the conditional graphs should not be used in the testing phase because they include SOH/RUL information, which is not available during testing. In addition to this ablation study, a comparative study was also conducted to show that the proposed method outperforms other deep learning methods, such

Table 2

Hyperparameters and network structure utilized in this case study for SOH estimations and RUL predictions.

Layer order	Description	Connected to	Output dimensions
1	1st Input layer	–	Batch $\times K \times \tau$
2	2nd Input layer	–	Batch $\times (K+1) \times \tau$
3	1st Dual graph convolutional layer	1st Input layer	Batch $\times 2 \times K \times \tau'$
4	2nd Dual graph convolutional layer	2nd Input layer	Batch $\times 2 \times (K+1) \times \tau'$
5–7	1st Dilated convolutional layer	1st Dual graph convolutional layer	Batch $\times 2 \times \Delta_i \times F_i$
8–10	2nd Dilated convolutional layer	2nd Dual graph convolutional layer	Batch $\times 2 \times \Delta_i \times F_i$
11	1st Flatten layer	1st Dilated convolutional layer	Batch $\times 2\Delta_i F_i$ ($i = 3$)
12	2nd Flatten layer	2nd Dilated convolutional layer	Batch $\times 2\Delta_i F_i$ ($j = 3$)
13	1st Fully connected layer	1st Flatten layer	Batch $\times 1$
14	2nd Fully connected layer	2nd Flatten layer	Batch $\times 1$

**Fig. 7.** The SOH estimation results for Battery No. 5, No. 6, No. 7, No. 18 in subset 1 and Battery No. 29, No. 30, No. 31, No. 32 in subset 2.**Fig. 8.** The SOH estimation results for Battery No. 25, No. 26, and No. 27 in Dataset 3.**Table 3**

Methods used in this ablation study for SOH estimations.

Method symbol	Method description
CGCN-DCO	Conditional graph convolutional network with dilated convolutional operations (Proposed)
GCN-DCO	Graph convolutional network with dilated convolutional operations
CGCN	Conditional graph convolutional network

as Transformer, MGCN (multi-receptive field GCN), and CNN+LSTM (convolutional neural network with long-short term memory).

Table 4 lists the RMSE (Root Mean Squared Error), MAE (Mean Absolute Error), MSE (Mean Squared Error), MedAE (Median Absolute Error), and R2-score (R-squared score) for SOH estimations across all batteries in three subsets, utilizing the methods presented in Table 3 and other deep learning methods. It can be concluded from this table that the proposed conditional graphs and the dilated convolutional operation can enhance the SOH estimation performance of the graph convolutional network. For instance, the average MAE of the CGCN-DCO method for all batteries is 0.0078. In contrast, the average MAE for CGCN and GCN-DCO for all batteries is 0.02228 and 0.01926,

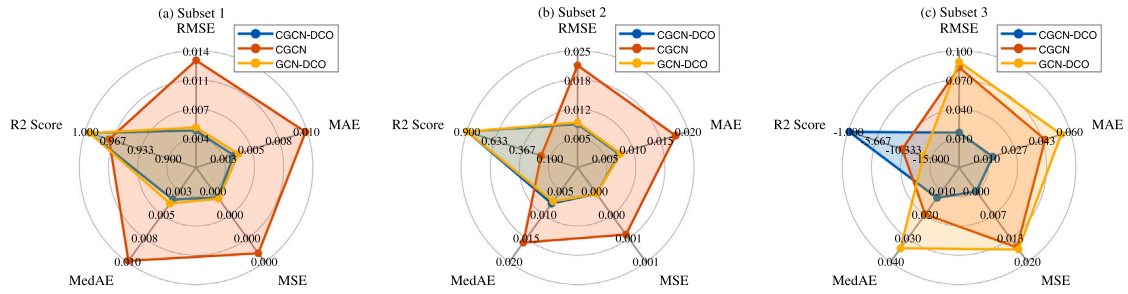
respectively. By employing the proposed conditional graphs and the dilated convolutional operation, the average prediction RMSE can be reduced by up to 73.5%, and the average R2-score can increase by up to 29.4%. Furthermore, as indicated in Table 4, the proposed method also outperforms other deep learning methods. For instance, the average RMSE for the proposed method across all batteries is 0.00913, while the average RMSE for Transformer, MGCN, and CNN+LSTM are 0.08039, 0.01254, and 0.127, respectively.

Fig. 9 shows a spider plot of five evaluation metrics used to assess the SOH estimation performance of the methods employed in the ablation study. These evaluation metrics include RMSE, MAE, MSE, MedAE, and R2-score. Based on this figure, it can also be concluded that both

Table 4

The RMSE, MAE, MedAE, MSE, and R2-score of SOH estimations for all batteries in three subsets with using methods in Table 3 and other deep learning methods.

Methods	Metrics	Subset 1				Subset 2				Subset 3			AVG
		No. 5	No. 6	No. 7	No. 18	No. 29	No. 30	No. 31	No. 32	No. 25	No. 26	No. 27	
CGCN-DCO	RMSE(10^{-2})	0.228	0.513	0.501	0.711	0.480	0.980	0.567	1.274	1.686	1.941	1.157	0.913
	MAE(10^{-2})	0.164	0.355	0.460	0.553	0.406	0.864	0.489	1.243	1.566	1.346	1.129	0.780
	MedAE(10^{-2})	0.120	0.220	0.475	0.477	0.433	0.881	0.466	1.259	1.542	1.146	1.166	0.744
	MSE(10^{-2})	0.001	0.003	0.003	0.005	0.002	0.010	0.003	0.016	0.028	0.038	0.013	0.011
	R2-score	1.000	0.999	1.000	0.995	0.996	0.990	0.983	0.997	0.817	0.922	0.918	0.965
CGCN	RMSE(10^{-2})	0.857	1.420	1.724	1.152	2.090	1.162	2.592	2.803	2.208	21.68	0.923	3.510
	MAE(10^{-2})	0.630	1.054	1.339	0.924	1.665	1.011	2.152	2.251	1.741	10.95	0.795	2.228
	MedAE(10^{-2})	0.435	0.747	1.155	0.764	1.607	1.029	1.853	1.855	1.412	3.835	0.684	1.398
	MSE(10^{-2})	0.007	0.020	0.030	0.013	0.044	0.013	0.067	0.079	0.049	4.701	0.009	0.457
	R2-score	0.997	0.992	0.973	0.985	0.761	0.966	0.484	0.795	0.634	0.188	0.436	0.746
GCN-DCO	RMSE(10^{-2})	0.279	0.572	0.532	0.714	0.806	1.033	0.562	1.051	2.545	21.96	2.047	2.918
	MAE(10^{-2})	0.227	0.408	0.478	0.576	0.650	0.927	0.495	1.031	2.146	12.25	2.000	1.926
	MedAE(10^{-2})	0.200	0.326	0.466	0.483	0.605	0.842	0.417	1.000	2.312	5.928	2.009	1.326
	MSE(10^{-2})	0.001	0.003	0.003	0.005	0.006	0.011	0.003	0.011	0.065	4.823	0.042	0.452
	R2-score	1.000	0.999	0.999	0.995	0.970	0.989	0.973	0.999	-0.02	-0.16	0.838	0.781
Transformer	RMSE(10^{-2})	4.520	4.410	25.03	4.856	7.857	4.049	2.069	1.931	2.618	23.05	8.035	8.039
	MAE(10^{-2})	3.618	3.496	16.61	3.877	4.603	3.491	1.826	1.590	2.244	18.24	7.348	6.086
	MedAE(10^{-2})	2.751	2.933	13.26	3.224	2.064	3.176	1.723	1.272	2.202	11.58	6.390	4.598
	MSE(10^{-2})	0.204	0.194	6.266	0.236	0.617	0.164	0.043	0.037	0.069	5.311	0.646	1.253
	R2-score	0.888	0.918	0.668	0.908	0.360	0.312	0.803	0.881	-0.46	0.269	-0.20	0.486
MGCN	RMSE(10^{-2})	0.659	1.395	1.689	1.268	0.491	1.159	0.472	1.049	1.675	2.292	1.645	1.254
	MAE(10^{-2})	0.538	1.034	1.343	1.018	0.416	1.118	0.416	1.034	1.552	1.544	1.627	1.058
	MedAE(10^{-2})	0.478	0.701	1.096	0.934	0.384	1.139	0.418	1.012	1.517	1.080	1.639	0.945
	MSE(10^{-2})	0.004	0.019	0.029	0.016	0.002	0.013	0.002	0.011	0.028	0.053	0.027	0.019
	R2-score	0.998	0.992	0.988	0.981	0.997	0.997	0.992	0.999	0.820	0.871	0.926	0.960
CNN+LSTM	RMSE(10^{-2})	7.249	4.558	30.62	13.28	12.62	2.371	2.181	5.478	9.216	13.70	31.49	12.07
	MAE(10^{-2})	6.047	3.554	30.24	11.85	12.45	1.939	1.556	5.324	8.975	10.02	31.44	11.22
	MedAE(10^{-2})	7.420	3.085	31.42	11.17	12.77	1.634	1.266	5.663	8.899	7.505	31.23	11.10
	MSE(10^{-2})	0.526	0.208	9.374	1.762	1.592	0.056	0.048	0.300	0.849	1.878	9.918	2.410
	R2-score	0.880	0.943	0.816	0.726	0.973	0.844	0.665	0.953	-0.61	0.261	-0.04	0.583

**Fig. 9.** The spider plot of five evaluation metrics are used to evaluate the SOH estimation performance of the methods used in this ablation study.

the proposed conditional graph and the use of dilated convolutional operations can improve prediction performance across all evaluation metrics. For example, in the first subset of batteries, the proposed CGCN-DCO achieves a MedAE of 0.00383, while the MedAE for GCN-DCO and CGCN is 0.00422 and 0.00987, respectively. Furthermore, in the second subset of batteries, the R2 Score of the proposed CGCN-DCO is 0.89677, whereas the R2-scores of GCN-DCO and CGCN are 0.89572 and 0.18335, respectively. In addition, a two-sample t-test was conducted to demonstrate that the proposed conditional graph and dilated convolutional operations can statistically improve prediction performance. The two-sample t-test indicates that the p -value for the hypothesis test, with the null hypothesis stating that the average RMSE of the proposed CGCN-DCO is equal to or greater than the average RMSE of CGCN, is 0.086 at a significance level of 0.1. Similarly, the p -value for the hypothesis test, with the null hypothesis stating that the average RMSE of the proposed CGCN-DCO is equal to or greater than the average RMSE of GCN-DCO, is 0.095 at a significance level of 0.1. This implies that the confidence level for rejecting the null hypothesis is greater than 0.9, indicating that the proposed method can significantly reduce prediction errors.

In addition, a comparison of the proposed CGCN-DCO with other methods reported in the literature was also conducted. Table 5 shows the RMSE of the SOH estimation of the proposed method (CGCN-DCO), graph convolutional network with dual attention mechanism (GCN-DA), multiple Gaussian regression models (MGP), logic regression with Gaussian process regression (LRGP), gradient boosting decision tree (GBDT), and Gaussian process (GP), and health index informed attention model (HIIA). Based on the RMSE of the SOH estimation presented in this table, it can be concluded that the proposed method (CGCN-DCO) exhibits superior performance compared to previously established methods. For instance, the average RMSE of the proposed method is 0.0049, whereas the average RMSE of other methods documented in the literature ranges between 0.0135 and 0.0493, further highlighting the effectiveness of the proposed method.

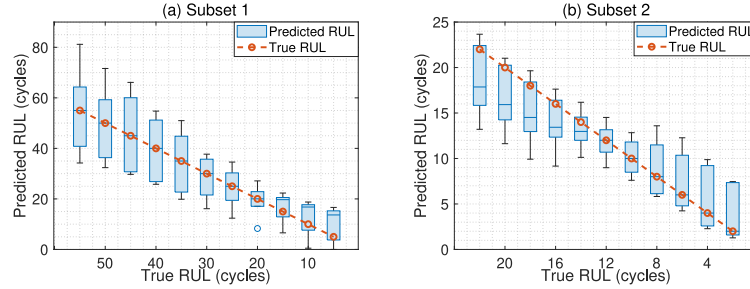
3.4. RUL prediction

In this section, the effectiveness of the proposed method in predicting RUL for subsets 1 and 2 is demonstrated. Because the number of cycles for each battery in subset 3 is very small, subset 3 does not have

Table 5

The RMSE of the SOH estimation of the proposed method and other methods reported in the literature.

Battery	CGCN-DCO	CGCN	GCN-DCO	GCN-DA (Wei & Wu, 2022b)	MGP (Zheng & Deng, 2019)	LRGP (Yu, 2018)	GBT (Qin, Zhao, & Liu, 2022)	GP (Yu, 2018)	HIIA (Wei, 2023)
No. 5	0.0023	0.0086	0.0028	0.0139	0.0096	0.0168	0.0192	0.0751	0.0165
No. 6	0.0051	0.0142	0.0057	0.0136	0.0167	0.0292	0.0281	0.0406	0.0153
No. 7	0.0050	0.0172	0.0053	0.0104	0.0129	–	0.0157	–	0.0089
No. 18	0.0071	0.0115	0.0071	0.0162	0.0228	0.0169	–	0.0323	0.0139
Average	0.0049	0.0129	0.0052	0.0135	0.0155	0.0210	0.0210	0.0493	0.0136

**Fig. 10.** The box plot of RUL prediction results for batteries in subset 1 and subset 2.

sufficient data to perform RUL predictions. For example, the number of discharge cycles for subset 3 is only 28, compared to over 100 for subset 1 and 40 for subset 2. Fig. 10 shows the box plot of RUL prediction results for Battery No. 5, No. 6, No. 7, No. 18 in subset 1 and Battery No. 29, No. 30, No. 31, No. 32 in subset 2. Based on this figure, it can be observed that the proposed method achieved high prediction accuracy because the predicted RUL aligns with the true RUL, and the range of the box plot includes the true RUL. It should be noted that the RUL prediction accuracy is not very high in comparison with the SOH prediction accuracy. This is because RUL predictions involve predicting the future end-of-life (EOL) of a battery, which can be very challenging. In addition, in this case study, the dataset is relatively small since each subset includes data collected from only four batteries.

Moreover, an ablation study was conducted to demonstrate the effectiveness of the proposed conditional graphs and the use of the dilated convolutional operation. In addition to this ablation study, a comparative study was also conducted to show that the proposed method outperforms other deep learning methods. Table 6 shows the RMSE, MAE, MSE, MedAE, and R2-score for the RUL prediction using methods listed in Table 3 and other deep learning methods. From this table, it can be concluded that both the proposed conditional graphs and the dilated convolutional operation improve the RUL prediction performance. For example, the average prediction RMSE of the proposed method is 10.45, while the average RMSE of the CGCN and GCN-DCO is 16.32 and 11.18, respectively. In addition, based on Table 6, it can also be observed that the proposed method outperforms other deep learning methods. For instance, the average MAE of the proposed method is 9.15, whereas the average MAE of Transformer, MGCN, and CNN+LSTM is 10.06, 9.65, and 11.13, respectively.

Fig. 11 shows the spider plot of five evaluation metrics used to assess the RUL prediction performance of the methods employed in this ablation study. These evaluation metrics include RMSE, MAE, MSE, MedAE, and R2-score. Based on this figure, it can also be concluded that both the proposed conditional graph and the use of dilated convolutional operations can improve RUL prediction performance across all evaluation metrics. For example, in the first subset of batteries, the proposed CGCN-DCO achieves a R2-score of 0.51, whereas the R2 Score for CGCN and GCN-DCO are -0.19 and 0.46 , respectively. Furthermore, in the second subset of batteries, the MedAE of the proposed CGCN-DCO is 3.48, while the MedAE of CGCN and GCN-DCO are 4.35 and 3.84, respectively.

To further demonstrate the effectiveness of the proposed method, a comparison was conducted between the proposed CGCN-DCO and other methods reported in the literature. Table 7 provides the RMSEs in the RUL predictions for the proposed method CGCN-DCO, GCN-DCO, CGCN, logic regression with Gaussian process (LRGP), Gaussian process (GP), LSTM, and LSTM with dual attention mechanisms (LSTM-DA). Based on the table, it can be concluded that the proposed CGCN-DCO outperforms other methods. For instance, the average RMSE of the proposed method is 16.84, while the average RMSE of the other methods ranges from 17.07 to 30.27.

4. Case study II

4.1. Data description

The proposed method is also demonstrated on the Oxford battery degradation dataset (Birkel, 2017). The Oxford battery dataset includes eight lithium-ion battery cells, each with a maximum capacity of 740 mAh. These battery cells underwent repeated charging and discharging operations, during which current, voltage, and temperature data were collected. The charging and discharging cycles exposed these battery cells to a CC and CV charging profile, followed by a drive cycle discharging profile. More details about this dataset can be found in Birkel, Roberts, McTurk, Bruce, and Howey (2017). Similar to the batteries in the NASA dataset, the capacity of these battery cells decreased as the number of charging and discharging cycles increased. Fig. 12 displays the degradation trajectories of capacity and capacity fade for all the battery cells. For all battery cells in the Oxford battery dataset, run-to-failure tests were conducted, and End-of-Life (EOL) was reached when the capacity decreased by 15%. Additionally, an eight-fold cross-validation was performed to thoroughly evaluate the efficacy of the proposed method for all eight battery cells. In both SOH estimations and RUL predictions, the extracted features and network structure are the same as those used in the first case study. The only difference is that the number of filters in the dilated convolutional layers is set to 10, and the learning rate is set to 10^{-4} .

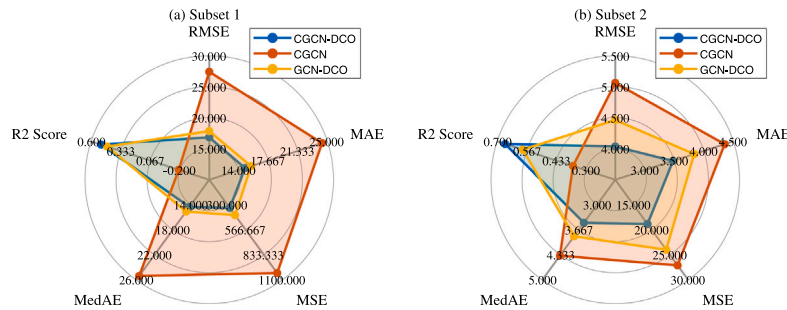
4.2. SOH estimation

Fig. 13 displays the SOH estimations for all battery cells in the Oxford battery dataset. The initial estimation point for these battery cells is 10 cycles. From Fig. 13, it can be concluded that the proposed method is capable of estimating the SOH of lithium-ion batteries with

Table 6

The RMSE, MAE, MedAE, MSE, and R2-score of RUL predictions for batteries in subset 1 and subset 2 with using methods in Table 3 and other deep learning methods.

Methods	Metrics	Subset 1				Subset 2				AVG
		No. 5	No. 6	No. 7	No. 18	No. 29	No. 30	No. 31	No. 32	
CGCN-DCO	RMSE	12.31	16.94	28.40	9.85	3.84	1.49	4.95	5.78	10.45
	MAE	11.89	14.01	24.21	9.18	3.32	1.37	4.39	4.81	9.15
	MedAE	12.38	11.13	24.57	8.98	2.72	1.61	4.33	3.88	8.70
	MSE	151.5	286.9	806.7	96.94	14.74	2.23	24.54	33.43	177.1
	R2-score	0.994	0.982	0.989	0.976	0.988	0.975	0.980	0.970	0.982
CGCN	RMSE	21.41	21.07	55.58	12.07	3.53	4.44	5.55	6.90	16.32
	MAE	20.19	17.21	50.30	10.35	2.95	4.14	4.66	6.01	14.48
	MedAE	21.00	14.89	57.50	8.36	2.18	4.27	4.60	5.37	14.77
	MSE	458.4	444.0	3089	145.6	12.48	19.68	30.80	47.57	531.0
	R2-score	0.974	0.967	0.935	0.693	0.988	0.918	0.980	0.974	0.929
GCN-DCO	RMSE	12.37	18.56	33.16	7.49	3.85	1.56	5.30	7.14	11.18
	MAE	11.93	15.06	28.56	6.83	3.25	1.43	4.58	6.14	9.72
	MedAE	12.27	10.19	30.60	6.77	2.50	1.34	4.69	5.39	9.22
	MSE	153.1	344.3	1099	56.03	14.82	2.43	28.11	50.96	218.7
	R2-score	0.994	0.985	0.989	0.980	0.983	0.979	0.972	0.960	0.980
Transformer	RMSE	18.47	17.56	28.43	7.29	8.86	5.03	5.53	4.03	11.90
	MAE	15.72	13.60	25.08	6.56	7.22	4.53	4.72	3.06	10.06
	MedAE	14.89	8.39	18.86	6.09	6.17	4.91	4.41	2.00	8.22
	MSE	341.1	308.2	808.5	53.17	78.53	25.27	30.62	16.27	207.7
	R2-score	0.993	0.968	0.984	0.975	0.443	0.780	0.835	0.962	0.867
MGCN	RMSE	12.94	19.25	30.02	8.92	3.81	1.80	5.31	6.72	11.10
	MAE	12.66	15.98	25.57	7.76	3.19	1.70	4.56	5.81	9.65
	MedAE	12.98	13.31	27.34	8.92	2.47	1.87	4.73	4.95	9.57
	MSE	167.3	370.6	901.0	79.59	14.53	3.25	28.21	45.14	201.2
	R2-score	0.996	0.986	0.990	0.967	0.985	0.987	0.979	0.967	0.982
CNN+LSTM	RMSE	15.13	14.93	35.93	12.06	6.52	5.17	8.02	5.61	12.92
	MAE	14.26	11.16	31.78	11.32	5.39	5.02	6.28	3.87	11.13
	MedAE	14.07	5.36	34.09	10.29	5.75	4.73	4.89	2.02	10.15
	MSE	228.8	222.9	1291	145.4	42.45	26.74	64.26	31.48	256.7
	R2-score	0.982	0.990	0.998	0.973	0.978	0.945	0.953	0.928	0.968

**Fig. 11.** The spider plot of five evaluation metrics is used to evaluate the RUL prediction performance of the methods used in this ablation study.**Table 7**

The RMSE of the RUL prediction of the proposed method and other methods reported in the literature.

	CGCN-DCO	CGCN	GCN-DCO	LRGP (Yu, 2018)	GP (Yu, 2018)	LSTM (Wei & Wu, 2022b)	LSTM-DA (Wei & Wu, 2022b)
Battery No. 5	12.23	21.28	12.30	16.60	40.70	27.52	32.56
Battery No. 6	16.88	21.02	18.47	21.10	29.50	40.20	42.00
Battery No. 7	28.31	55.36	33.03	–	–	33.62	33.01
Battery No. 18	9.92	11.98	7.56	13.50	20.60	6.46	8.97
Average	16.84	27.41	17.84	17.07	30.27	26.950	29.135

high precision, as the estimated SOH aligns closely with the true SOH. For example, regarding battery Cell 1, both the true SOH and the estimated SOH are 0.915 when 12 discharge cycles have been observed.

Similar to the first case study, an ablation study and a comparative study were also conducted to demonstrate the effectiveness of the proposed method. Table 8 shows the RMSE, MAE, MSE, MedAE, and R2-score of SOH estimations for all battery cells in the Oxford battery

dataset. From this table, it can be observed that the proposed method outperforms the methods listed in Table 3 and other deep learning methods. For example, the average SOH estimation RMSE of the proposed method is 0.0150. In comparison, the average SOH estimation RMSE of Transformer and MGCN is 0.0235 and 0.0294, respectively. Moreover, the average R2-score of the proposed method is 0.9672. However, the average R2-score of other deep learning methods in this table ranges from 0.8093 to 0.9658.

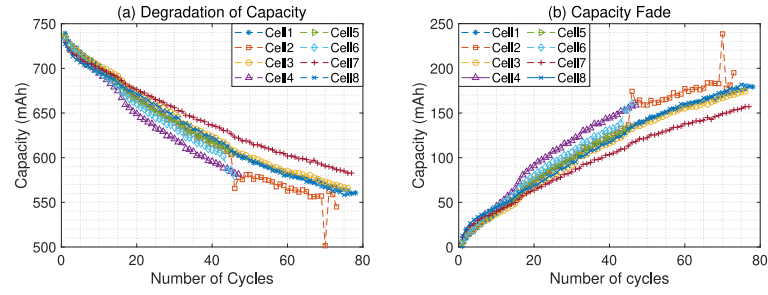


Fig. 12. The degradation trajectories of capacity and capacity fade for battery cells in the Oxford battery degradation dataset..

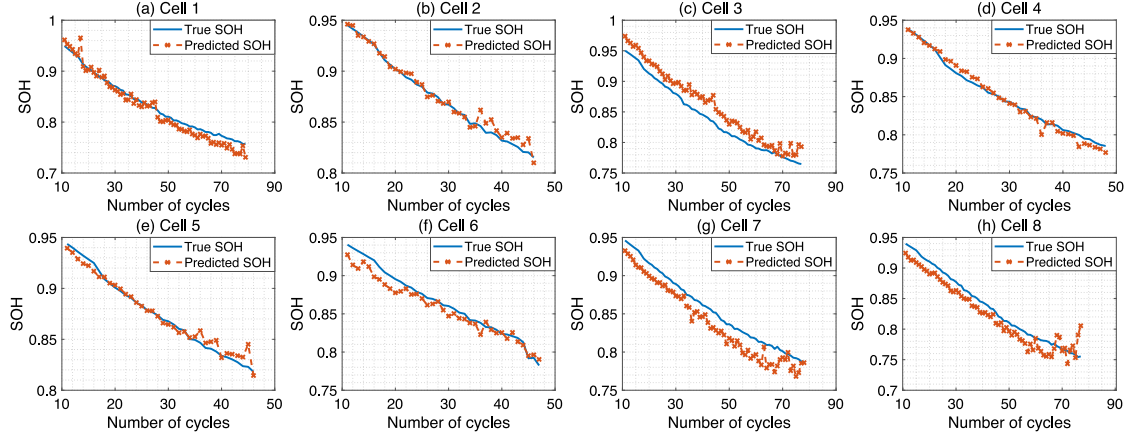


Fig. 13. The SOH estimations for all battery cells in the Oxford battery dataset.

Table 8

The RMSE, MAE, MSE, MedAE, and R2-score of SOH estimations for all battery cells in the Oxford battery dataset.

Methods	Metrics	Battery index								AVG
		Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8	
CGCN-DCO	RMSE	0.0097	0.0335	0.0156	0.0049	0.0213	0.0080	0.0140	0.0131	0.0150
	MAE	0.0079	0.0232	0.0145	0.0037	0.0067	0.0067	0.0131	0.0120	0.0110
	MedAE	0.0071	0.0069	0.0146	0.0030	0.0026	0.0049	0.0127	0.0120	0.0080
	MSE	0.0001	0.0011	0.0002	0.0000	0.0005	0.0001	0.0002	0.0002	0.0003
	R2-score	0.9928	0.9075	0.9908	0.9933	0.8967	0.9921	0.9898	0.9746	0.9672
CGCN	RMSE	0.0135	0.0339	0.0167	0.0063	0.0205	0.0062	0.0138	0.0131	0.0155
	MAE	0.0113	0.0235	0.0154	0.0050	0.0077	0.0048	0.0129	0.0120	0.0116
	MedAE	0.0102	0.0077	0.0174	0.0042	0.0035	0.0039	0.0131	0.0119	0.0090
	MSE	0.0002	0.0012	0.0003	0.0000	0.0004	0.0000	0.0002	0.0002	0.0003
	R2-score	0.9937	0.9063	0.9914	0.9860	0.9010	0.9883	0.9858	0.9739	0.9658
GCN-DCO	RMSE	0.0102	0.0334	0.0162	0.0055	0.0216	0.0065	0.0140	0.0133	0.0151
	MAE	0.0087	0.0231	0.0151	0.0039	0.0061	0.0060	0.0131	0.0122	0.0110
	MedAE	0.0087	0.0071	0.0155	0.0031	0.0019	0.0060	0.0128	0.0123	0.0084
	MSE	0.0001	0.0011	0.0003	0.0000	0.0005	0.0000	0.0002	0.0002	0.0003
	R2-score	0.9906	0.9077	0.9907	0.9907	0.8828	0.9957	0.9900	0.9743	0.9653
Transformer	RMSE	0.0150	0.0374	0.0179	0.0298	0.0307	0.0258	0.0150	0.0167	0.0235
	MAE	0.0106	0.0242	0.0135	0.0248	0.0221	0.0239	0.0128	0.0132	0.0181
	MedAE	0.0069	0.0128	0.0111	0.0214	0.0171	0.0247	0.0116	0.0106	0.0145
	MSE	0.0002	0.0014	0.0003	0.0009	0.0009	0.0007	0.0002	0.0003	0.0006
	R2-score	0.9431	0.7276	0.9340	0.9168	0.7257	0.9587	0.9549	0.9763	0.8921
MGCN	RMSE	0.0356	0.0316	0.0104	0.0163	0.0323	0.0130	0.0378	0.0583	0.0294
	MAE	0.0243	0.0249	0.0079	0.0113	0.0132	0.0110	0.0301	0.0454	0.0210
	MedAE	0.0173	0.0202	0.0059	0.0068	0.0070	0.0108	0.0262	0.0403	0.0168
	MSE	0.0013	0.0010	0.0001	0.0003	0.0010	0.0002	0.0014	0.0034	0.0011
	R2-score	0.8260	0.7894	0.9687	0.9234	0.6432	0.9451	0.7430	0.6356	0.8093
CNN+LSTM	RMSE	0.0211	0.0353	0.0396	0.0269	0.0492	0.0204	0.0285	0.0211	0.0303
	MAE	0.0176	0.0271	0.0360	0.0200	0.0208	0.0152	0.0264	0.0160	0.0224
	MedAE	0.0158	0.0207	0.0378	0.0142	0.0137	0.0107	0.0253	0.0124	0.0188
	MSE	0.0004	0.0012	0.0016	0.0007	0.0024	0.0004	0.0008	0.0004	0.0010
	R2-score	0.9348	0.8568	0.9606	0.9284	0.2772	0.8978	0.9461	0.8945	0.8370

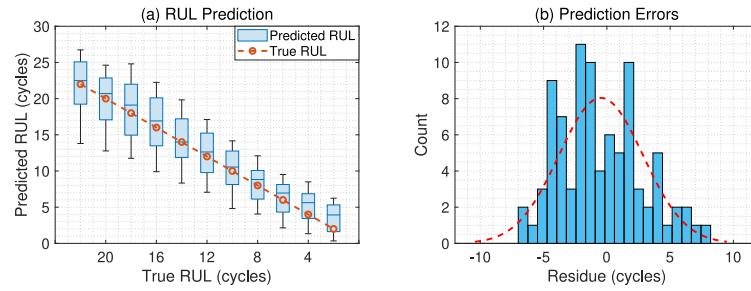


Fig. 14. The box plot of RUL prediction results and prediction errors for battery cells in the Oxford battery dataset.

Table 9

The RMSE, MAE, MSE, MedAE, and R2-score of RUL predictions for all battery cells in the Oxford battery dataset.

Methods	Metrics	Battery index								AVG
		Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8	
CGCN-DCO	RMSE	1.846	2.896	3.054	4.342	4.330	1.752	7.828	1.822	3.484
	MAE	1.674	2.783	2.875	3.928	4.303	1.700	7.079	1.412	3.219
	MedAE	1.672	2.474	2.971	3.890	4.321	1.629	6.244	1.215	3.052
	MSE	3.408	8.384	9.326	18.85	18.75	3.068	61.28	3.319	15.80
	R2-score	0.988	0.998	0.960	0.999	0.999	0.998	0.998	0.999	0.992
CGCN	RMSE	2.049	2.973	2.867	4.414	4.398	1.728	7.788	1.933	3.519
	MAE	1.864	2.791	2.683	3.879	4.355	1.682	7.080	1.552	3.236
	MedAE	2.000	2.446	2.742	3.796	4.267	1.649	6.395	1.473	3.096
	MSE	4.200	8.839	8.220	19.49	19.34	2.987	60.66	3.736	15.93
	R2-score	0.990	0.998	0.963	0.999	0.999	0.999	0.998	0.999	0.993
GCN-DCO	RMSE	1.660	3.062	3.827	4.564	4.383	1.715	7.807	2.046	3.633
	MAE	1.422	2.951	3.691	4.151	4.343	1.674	7.121	1.564	3.365
	MedAE	1.392	2.815	3.769	4.236	4.285	1.552	6.198	1.509	3.219
	MSE	2.756	9.376	14.65	20.83	19.21	2.940	60.95	4.188	16.86
	R2-score	0.993	0.999	0.960	0.999	0.999	0.999	0.998	0.999	0.993
Transformer	RMSE	5.119	3.597	8.511	4.850	4.211	3.846	7.706	6.402	5.530
	MAE	4.161	3.265	7.072	3.446	3.904	2.803	6.877	5.103	4.579
	MedAE	2.354	3.268	6.360	1.368	3.394	2.514	6.323	4.227	3.726
	MSE	26.20	12.94	72.44	23.52	17.73	14.79	59.38	40.99	33.50
	R2-score	0.918	0.978	0.931	0.981	0.988	0.987	0.998	0.996	0.972
MGCN	RMSE	1.282	1.047	4.460	3.645	1.247	3.310	10.30	7.693	4.123
	MAE	0.940	0.899	4.172	3.325	1.137	3.255	9.781	7.473	3.873
	MedAE	0.508	0.702	3.622	3.319	1.035	3.261	9.453	7.436	3.667
	MSE	1.642	1.096	19.89	13.29	1.554	10.95	106.2	59.18	26.72
	R2-score	0.996	0.999	0.988	1.000	0.998	0.998	0.999	0.999	0.997
CNN+LSTM	RMSE	3.476	4.006	5.146	4.423	4.155	2.883	8.923	4.870	4.735
	MAE	2.743	3.627	4.260	4.012	3.752	2.595	7.088	3.436	3.939
	MedAE	2.313	4.057	3.367	4.515	4.279	2.884	4.928	1.430	3.472
	MSE	12.08	16.05	26.48	19.56	17.27	8.311	79.63	23.72	25.39
	R2-score	0.934	0.938	0.968	0.935	0.949	0.947	0.937	0.936	0.943

4.3. RUL prediction

Fig. 14 shows the box plot of RUL prediction results and prediction errors for battery cells in the Oxford battery dataset. From this figure, it can be concluded that the proposed method has a relatively high prediction accuracy, as the mean of the predicted RUL aligns with the true RUL, and the majority of prediction errors range from -5 to 5 cycles. It is worth noting that the RUL prediction performance is better than the RUL prediction performance in the first case study for two primary reasons. First, there are more battery cells included in this case study, leading to increased prediction performance. Second, the degradation trajectories of these battery cells are close to each other, therefore reducing the difficulties in RUL predictions.

Similar to the first case study, an ablation study was conducted to demonstrate the effectiveness of the proposed conditional graphs and the use of the dilated convolutional operation. A comparative study was also conducted to show that the proposed method outperforms other deep learning methods. Table 9 displays the RMSE, MAE, MSE, MedAE, and R2-score for RUL prediction using methods listed in Table 3 and other deep learning methods. Based on this table, it can be observed

that both the proposed conditional graphs and the dilated convolutional operation improve RUL prediction performance. For example, the average RUL prediction RMSE of the proposed method is 3.484. In contrast, the average RUL prediction RMSE of the CGCN and GCN-DCO is 3.519 and 3.633, respectively. From Table 9, it can be concluded that the proposed method outperforms other deep learning methods. As an example, the average prediction MAE of the proposed method is 3.219. In comparison, the average prediction MAE of Transformer, MGCN, and CNN+LSTM ranges from 3.873 to 4.579.

5. Conclusions and future work

In this work, two types of undirected graphs were introduced. The first type of graphs (G_1) was used to consider the correlation among features, while the second type of graphs (G_2) was used to consider the correlation between features and SOH/RUL. Two feature spaces were extracted from the two types of graphs, respectively. KL-divergence was then adopted to minimize the distance between the two feature spaces, allowing the feature space extracted from G_1 to approximate the feature space extracted from G_2 . Even without SOH/RUL, the correlation between the features and SOH/RUL can be taken into

account when the feature space extracted from \mathcal{G}_1 is used for testing. Additionally, the dilated convolutional operations were implemented after aggregating similar features in GCN, allowing one to consider the temporal correlation among the aggregated features. To evaluate the effectiveness of the proposed method, two battery datasets (i.e., NASA and Oxford battery datasets) were used, where the current, voltage, and temperature data in discharge cycles were used to predict the SOH and RUL. Experimental results have demonstrated that the proposed method outperforms other methods, such as the Transformer encoder, multi-receptive field GCN, and convolutional neural network with long short-term memory, in terms of RMSE, MAE, MedAE, MSE, and R2-score. Furthermore, the experimental results have shown that the proposed method outperforms other machine learning methods reported in the literature. In the future, we will take into account the underlying physics of battery aging in the proposed method.

CRedit authorship contribution statement

Yupeng Wei: Conceptualization, Methodology, Software, Investigation, Validation, Visualization, Writing – original draft. **Dazhong Wu:** Conceptualization, Methodology, Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This research was in part supported by the National Science Foundation, USA under Grant No. 2131619.

References

- Al-Dulaimi, Ali, Zabihi, Soheil, Asif, Amir, & Mohammadi, Arash (2019). A multimodal and hybrid deep neural network model for remaining useful life estimation. *Computers in Industry*, 108, 186–196.
- Audin, Paul, Jorge, Inès, Mesbahi, Tedjani, Samet, Ahmed, De Beuvron, François De Bertrand, & Boné, Romuald (2021). Auto-encoder LSTM for Li-ion SOH prediction: a comparative study on various benchmark datasets. In *2021 20th IEEE international conference on machine learning and applications* (pp. 1529–1536). IEEE.
- Bai, Shaojie, Kolter, J. Zico, & Koltun, Vladlen (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271.
- Birkel, Christoph (2017). *Oxford battery degradation dataset 1*. University of Oxford.
- Birkel, Christoph R., Roberts, Matthew R., McTurk, Euan, Bruce, Peter G., & Howey, David A. (2017). Degradation diagnostics for lithium ion cells. *Journal of Power Sources*, 341, 373–386.
- Chang, Chun, Wang, Qiyue, Jiang, Jiuchun, & Wu, Tiezhou (2021). Lithium-ion battery state of health estimation using the incremental capacity and wavelet neural networks with genetic algorithm. *Journal of Energy Storage*, 38, Article 102570.
- Chao, Kuei-Hsiang, & Chen, Jing-Wei (2011). State-of-health estimator based-on extension theory with a learning mechanism for lead-acid batteries. *Expert Systems with Applications*, 38(12), 15183–15193.
- Chen, Ning, Hu, Xiaojun, Gui, Weihua, & Zou, Jiachi (2014). Estimation of Li-ion battery state of charging and state of healthy based on unscented Kalman filtering. In *The 26th Chinese control and decision conference* (pp. 4725–4729). IEEE.
- Cheng, Gong, Wang, Xinzhi, & He, Yurong (2021). Remaining useful life and state of health prediction for lithium batteries based on empirical mode decomposition and a long and short memory neural network. *Energy*, 232, Article 121022.
- Cui, Zhiqian, Gao, Xuhong, Mao, Jiawei, & Wang, Chunhui (2022). Remaining capacity prediction of lithium-ion battery based on the feature transformation process neural network. *Expert Systems with Applications*, 190, Article 116075.
- Dong, Guangzhong, Chen, Zonghai, Wei, Jingwen, & Ling, Qiang (2018). Battery health prognosis using Brownian motion modeling and particle filtering. *IEEE Transactions on Industrial Electronics*, 65(11), 8646–8655.
- Duong, Pham Luu Trung, & Raghavan, Nagarajan (2018). Heuristic Kalman optimized particle filter for remaining useful life prediction of lithium-ion battery. *Microelectronics Reliability*, 81, 232–243.
- Eddahech, Akram, Briat, Olivier, Bertrand, Nicolas, Delétage, Jean-Yves, & Vinassa, Jean-Michel (2012). Behavior and state-of-health monitoring of Li-ion batteries using impedance spectroscopy and recurrent neural networks. *International Journal of Electrical Power & Energy Systems*, 42(1), 487–494.
- Greenbank, Samuel, & Howey, David A. (2023). Piecewise-linear modelling with automated feature selection for Li-ion battery end-of-life prognosis. *Mechanical Systems and Signal Processing*, 184, Article 109612.
- Guo, Peng, Song, Huaihe, & Chen, Xiaohong (2009). Electrochemical performance of graphene nanosheets as anode material for lithium-ion batteries. *Electrochemistry Communications*, 11(6), 1320–1324.
- Guo, Xifeng, Wang, Kaize, Yao, Shu, Fu, Guojiang, & Ning, Yi (2023). RUL prediction of lithium ion battery based on CEEMDAN-CNN BiLSTM model. *Energy Reports*, 9, 1299–1306.
- Hammond, David K., Vandergheynst, Pierre, & Gribonval, Rémi (2011). Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30(2), 129–150.
- He, Wei, Williard, Nicholas, Chen, Chaochao, & Pecht, Michael (2014). State of charge estimation for Li-ion batteries using neural network modeling and unscented Kalman filter-based error cancellation. *International Journal of Electrical Power & Energy Systems*, 62, 783–791.
- Huang, Yongzhen, Wu, Zifeng, Wang, Liang, & Tan, Tieniu (2013). Feature coding in image classification: A comprehensive study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3), 493–506.
- Khaleghi, Sahar, Hosen, Md Sazzad, Karimi, Danial, Behi, Hamidreza, Beheshti, S. Hamidreza, Van Mierlo, Joeri, et al. (2022). Developing an online data-driven approach for prognostics and health management of lithium-ion batteries. *Applied Energy*, 308, Article 118348.
- Kingma, Diederik P., & Welling, Max (2013). Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.
- Kusner, Matt J., Paige, Brooks, & Hernández-Lobato, José Miguel (2017). Grammar variational autoencoder. In *International conference on machine learning* (pp. 1945–1954). PMLR.
- Lee, Gyumin, Kim, Juram, & Lee, Changyong (2022). State-of-health estimation of Li-ion batteries in the early phases of qualification tests: An interpretable machine learning approach. *Expert Systems with Applications*, 197, Article 116817.
- Lee, Gyumin, Kwon, Daeil, & Lee, Changyong (2023). A convolutional neural network model for SOH estimation of Li-ion batteries with physical interpretability. *Mechanical Systems and Signal Processing*, 188, Article 110004.
- Li, Penghua, Zhang, Zijian, Grosu, Radu, Deng, Zhongwei, Hou, Jie, Rong, Yujun, et al. (2022). An end-to-end neural network framework for state-of-health estimation and remaining useful life prediction of electric vehicle lithium batteries. *Renewable and Sustainable Energy Reviews*, 156, Article 111843.
- Li, Tianfu, Zhao, Zhibin, Sun, Chuang, Yan, Ruqiang, & Chen, Xuefeng (2020). Multireceptive field graph convolutional networks for machine fault diagnosis. *IEEE Transactions on Industrial Electronics*, 68(12), 12739–12749.
- Li, Tianfu, Zhao, Zhibin, Sun, Chuang, Yan, Ruqiang, & Chen, Xuefeng (2021). Hierarchical attention graph convolutional network to fuse multi-sensor signals for remaining useful life prediction. *Reliability Engineering & System Safety*, 215, Article 107878.
- Liu, Xingtao, Wu, Ji, Zhang, Chenbin, & Chen, Zonghai (2014). A method for state of energy estimation of lithium-ion batteries at dynamic currents and temperatures. *Journal of Power Sources*, 270, 151–157.
- Lu, Jiahuan, Xiong, Rui, Tian, Jinpeng, Wang, Chenxu, Hsu, Chia-Wei, Tsou, Nien-Ti, et al. (2022). Battery degradation prediction against uncertain future conditions with recurrent neural network enabled deep learning. *Energy Storage Materials*.
- Luo, Jian, Fang, Shu-Cherng, Deng, Zhibin, & Tian, Ye (2022). Robust kernel-free support vector regression based on optimal margin distribution. *Knowledge-Based Systems*, 253, Article 109477.
- Ma, Guijun, Wang, Zidong, Liu, Weibo, Fang, Jingzhong, Zhang, Yong, Ding, Han, et al. (2023). A two-stage integrated method for early prediction of remaining useful life of lithium-ion batteries. *Knowledge-Based Systems*, 259, Article 110012.
- Ma, Yan, Yao, Meihao, Liu, Hongcheng, & Tang, Zhiguo (2022). State of health estimation and remaining useful life prediction for lithium-ion batteries by improved particle swarm optimization-back propagation neural network. *Journal of Energy Storage*, 52, Article 104750.
- Nian, Peng, Shuzhi, Zhang, & Xiongwen, Zhang (2021). Co-estimation for capacity and state of charge for lithium-ion batteries using improved adaptive extended Kalman filter. *Journal of Energy Storage*, 40, Article 102559.
- Niu, Guangxing, Wang, Xuan, Liu, Enhui, & Zhang, Bin (2021). Lebesgue sampling based deep belief network for lithium-ion battery diagnosis and prognosis. *IEEE Transactions on Industrial Electronics*, 69(8), 8481–8490.
- Park, Jinhyeong, Lee, Munsu, Kim, Gunwoo, Park, Seongyun, & Kim, Jonghoon (2020). Integrated approach based on dual extended Kalman filter and multivariate autoregressive model for predicting battery capacity using health indicator and SOC/SOH. *Energies*, 13(9), 2138.
- Qin, Pengliang, Zhao, Linhui, & Liu, Zhiyuan (2022). State of health prediction for lithium-ion battery using a gradient boosting-based data-driven method. *Journal of Energy Storage*, 47, Article 103644.

- Raskar, Charushila, & Nema, Shikha (2022). Metaheuristic enabled modified hidden Markov model for traffic flow prediction. *Computer Networks*, 206, Article 108780.
- Saha, B., & Goebel, K. (2007). Battery data set. *NASA AMES Prognostics Data Repository*.
- Shen, Sheng, Sadoughi, Mohammadkazem, Li, Meng, Wang, Zhengdao, & Hu, Chao (2020). Deep convolutional neural networks with ensemble learning and transfer learning for capacity estimation of lithium-ion batteries. *Applied Energy*, 260, Article 114296.
- Ungurean, Lucian, Micea, Mihai V., & Carstoiu, Gabriel (2020). Online state of health prediction method for lithium-ion batteries, based on gated recurrent unit neural networks. *International Journal of Energy Research*, 44(8), 6767–6777.
- Wang, Ying, & Cao, Guozhong (2008). Developments in nanostructured cathode materials for high-performance lithium-ion batteries. *Advanced Materials*, 20(12), 2251–2269.
- Wang, Lei, Cao, Hongrui, Xu, Hao, & Liu, Haichen (2022). A gated graph convolutional network with multi-sensor signals for remaining useful life prediction. *Knowledge-Based Systems*, 252, Article 109340.
- Wang, Dong, Zhao, Yang, Yang, Fangfang, & Tsui, Kwok-Leung (2017). Nonlinear-drifted Brownian motion with multiple hidden states for remaining useful life prediction of rechargeable batteries. *Mechanical Systems and Signal Processing*, 93, 531–544.
- Wei, Yupeng (2023). Prediction of state of health of lithium-ion battery using health index informed attention model. *Sensors*, 23(5), 2587.
- Wei, Jingwen, Dong, Guangzhong, & Chen, Zonghai (2017). Remaining useful life prediction and state of health diagnosis for lithium-ion batteries using particle filter and support vector regression. *IEEE Transactions on Industrial Electronics*, 65(7), 5634–5643.
- Wei, Yupeng, & Wu, Dazhong (2022a). Material removal rate prediction in chemical mechanical planarization with conditional probabilistic autoencoder and stacking ensemble learning. *Journal of Intelligent Manufacturing*, 1–13.
- Wei, Yupeng, & Wu, Dazhong (2022b). Prediction of state of health and remaining useful life of lithium-ion battery using graph convolutional network with dual attention mechanisms. *Reliability Engineering & System Safety*, Article 108947.
- Wei, Yupeng, & Wu, Dazhong (2023). Remaining useful life prediction of bearings with attention-awared graph convolutional network. *Advanced Engineering Informatics*, 58, Article 102143.
- Wei, Yupeng, Wu, Dazhong, & Terpenney, Janis (2021). Learning the health index of complex systems using dynamic conditional variational autoencoders. *Reliability Engineering & System Safety*, 216, Article 108004.
- Wei, Yupeng, Wu, Dazhong, & Terpenney, Janis (2023). Bearing remaining useful life prediction using self-adaptive graph convolutional networks with self-attention mechanism. *Mechanical Systems and Signal Processing*, 188, Article 110010.
- Xi, Zhimin, Wang, Rui, Fu, Yuhong, & Mi, Chris (2022). Accurate and reliable state of charge estimation of lithium ion batteries using time-delayed recurrent neural networks through the identification of overexcited neurons. *Applied Energy*, 305, Article 117962.
- Xia, Tangbin, Song, Ya, Zheng, Yu, Pan, Ershun, & Xi, Lifeng (2020). An ensemble framework based on convolutional bi-directional LSTM with multiple time windows for remaining useful life estimation. *Computers in Industry*, 115, Article 103182.
- Xu, Fan, Yang, Fangfang, Fei, Zicheng, Huang, Zhelin, & Tsui, Kwok-Leung (2021). Life prediction of lithium-ion batteries based on stacked denoising autoencoders. *Reliability Engineering & System Safety*, 208, Article 107396.
- Yan, Wuzhao, Zhang, Bin, Zhao, Guangquan, Tang, Shijie, Niu, Guangxing, & Wang, Xiaofeng (2018). A battery management system with a lebesgue-sampling-based extended Kalman filter. *IEEE Transactions on Industrial Electronics*, 66(4), 3227–3236.
- Yu, Jianbo (2018). State of health prediction of lithium-ion batteries: Multiscale logic regression and Gaussian process regression ensemble. *Reliability Engineering & System Safety*, 174, 82–95.
- Zamfirache, Iuliu Alexandru, Precup, Radu-Emil, Roman, Raul-Cristian, & Petriu, Emil M. (2022). Policy iteration reinforcement learning-based control using a grey wolf optimizer algorithm. *Information Sciences*, 585, 162–175.
- Zhang, Yong, Chen, Liaogehao, Li, Yi, Zheng, Xiujuan, Chen, Jianliang, & Jin, Junyang (2021). A hybrid approach for remaining useful life prediction of lithium-ion battery with adaptive levy flight optimized particle filter and long short-term memory network. *Journal of Energy Storage*, 44, Article 103245.
- Zhang, Jiushi, Huang, Congsheng, Chow, Mo-Yuen, Li, Xiang, Tian, Jilun, Luo, Hao, et al. (2023). A data-model interactive remaining useful life prediction approach of lithium-ion batteries based on PF-BiGRU-TSAM. *IEEE Transactions on Industrial Informatics*.
- Zhang, Jiushi, Jiang, Yuchen, Li, Xiang, Huo, Mingyi, Luo, Hao, & Yin, Shen (2022). An adaptive remaining useful life prediction approach for single battery with unlabeled small sample data and parameter uncertainty. *Reliability Engineering & System Safety*, 222, Article 108357.
- Zhang, Yong, Tu, Lei, Xue, Zhiwei, Li, Sai, Tian, Lulu, & Zheng, Xiujuan (2022). Weight optimized unscented Kalman filter for degradation trend prediction of lithium-ion battery with error compensation strategy. *Energy*, 251, Article 123890.
- Zhen, Yan, Fang, Junyi, Zhao, Xiaoming, Ge, Jiawang, & Xiao, Yifei (2022). Temporal convolution network based on attention mechanism for well production prediction. *Journal of Petroleum Science and Engineering*, 218, Article 111043.
- Zheng, Xueying, & Deng, Xiaogang (2019). State-of-health prediction for lithium-ion batteries with multiple gaussian process regression model. *IEEE Access*, 7, 150383–150394.