Journal of Intelligent Manufacturing (2022) 33:2295–2306
https://doi.org/10.1007/s10845-021-01784-1

A novel hypergraph convolution network-based approach for predicting the material removal rate in chemical mechanical planarization

Liqiao Xia[1] · Pai Zheng[1] · Xiao Huang[2] · Chao Liu[1]

Received: 16 February 2021 / Accepted: 12 May 2021 / Published online: 24 May 2021

Abstract
The material removal rate (MRR) plays a critical role in the chemical mechanical planarization (CMP) process in the semiconductor industry. Many physics-based and data-driven approaches have been proposed to-date to predict the MRR. Nevertheless, most of them neglect the underlying equipment structure containing essential interaction mechanisms among different components. To fill the gap, this paper proposes a novel hypergraph convolution network (HGCN) based approach for predicting MRR in the CMP process. The main contributions include: (1) a generic hypergraph model to represent the interrelationships of complex equipment; and (2) a temporal-based prediction approach to learn the complex data correlation and high-order representation based on the hypergraph. To validate the effectiveness of the proposed approach, a case study is conducted by comparing with other cutting-edge models, of which it outperforms in several metrics. It is envisioned that this research can also bring insightful knowledge to similar scenarios in the manufacturing process.

Keywords Material removal rate · Graph convolutional network · Gate recurrent unit · Hypergraph · Chemical mechanical planarization

Introduction

Chemical mechanical planarization (CMP) is a critical process widely adopted in the semiconductor industry, since the surface flatness largely influences the manufacturing quality. The CMP process can be used to planarize numerous materials, such as: dielectrics, semiconductors, metals, and composites. The contact area and pressure of the wafer play an essential role for the polishing speed. Meanwhile, the synergistic mechanism between chemical reaction and mechanical abrasion has an extensive effect on the contact area, which in turn affects the wafer surface removal rate (Ludwig & Kuna, 2012). Excessive material removal rate (MRR) leads to the defect and depression of wafers material, which increases the fault rate of CMP (Hong et al., 2020). On the contrary, low MRR represents that the wafer is not polished sufficiently, which affects its final quality. Therefore, MRR serves as one of the important indicators to measure its final quality of the polished surface.

Despite its significance, wafer is normally wrapped in the CMP tool between the pad and the wafer carrier, resulting a difficulty to estimate the MRR until it finishes the whole process. Therefore, it is necessary to predict the MRR during the CMP process for prognostics and health management. Conventionally, research studies focus on investigating the components (Evans et al., 2003) and manufacturing environment (Xu et al., 2020) of CMP that affect the MRR. Meanwhile, various physics-based mathematical models have been established to fit a curve to predict the MRR (Lee et al., 2013) or simulate the manufacturing process (Lee & Jeong, 2011). Furthermore, empowered by the capability to collect multimodal CMP data, and the high computation power, machine learning, and deep learning approaches have been ever-increasingly implemented to predict the MRR.

Most CMP equipment owns a pre-defined and clear operation mechanism that indicates its corresponding connection among the inner components and parts (Jia et al.,

✉ Pai Zheng
pai.zheng@polyu.edu.hk

[1] Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, HKSAR, China

[2] Department of Computing, The Hong Kong Polytechnic University, Hung Hom, HKSAR, China

2018). Nevertheless, the structural knowledge contained in the equipment is often neglected in the existing MRR prediction models, which can play a significant role. On one hand, it can reflect the dependency between various components/parts, which serves as the fundamental basis for determining the sources of data to be considered. On the other hand, although recent work started to establish the knowledge graph-based model, it only considers the interrelationships such as 'is part of', 'lead to', 'has a function' (Yan et al., 2020), while ignoring the impact propagation among component/parts.

To address this issue, a proper industrial graph representing the structural knowledge of CMP equipment and its interrelationship mechanisms should be first established. Meanwhile, advanced graph convolution network (GCN) approaches (Wu et al., 2021), as the potential solution for solution recommendation and prediction, can be further leveraged and enhanced to support the MRR prediction process. Motivated by this, this paper proposes a novel temporal hypergraph convolutional network-based approach for MRR prediction in CMP. The rest of this paper is organized as follows. In "Related work" section reviews the related work of the MRR prediction, industrial graph applications, and state-of-the-art methods of graph-based reasoning. In "CMP hypergraph construction" section introduces the proposed methodology of constructing an equipment hypergraph model of CMP. Meanwhile, in "HGCN-based model" section presents the proposed combined HGCN with GRU model for MRR prediction. To validate its effectiveness, in "Case study" section undertakes a comparative study based on an open-source MRR dataset, and the experimental results are further discussed in "Discussion" section. At last, in "Conclusion" section outlines the contributions of this work and highlights the future directions.

Related work

This section summarizes the related work about MRR prediction and provides a comprehensive review of the development and categories of industrial graph and the graph-based reasoning approaches.

MRR prediction

The existing MRR prediction approaches can be divided into physics-based and data-driven ones. One of the most popular physical-based approaches is the Preston equation (Evans et al., 2003), which indicates $MRR = K_p P^{\alpha} V^{\beta}$, where P represents the downward pressure push to a wafer, V represents the rotating speed, K_p is the Preston coefficient. Following this model, many efforts have been done by adding contact stress, relative velocity, and chemical reaction rate into the Preston coefficient (Lee & Jeong, 2011). Also, other research takes the size, concentration, distribution of particles, slurry flow rate, polishing pad surface topography into consideration (Lee et al., 2013). However, the major limitation of physical-based approaches lies in the prior assumptions of the model, which often may not be correct in practice.

For the data-driven based approaches, machine learning and statistical methods have been widely adopted. For instance, the nonlinear Bayesian model (Kong et al., 2010) and the decision tree-based model were introduced for MRR prediction (Li et al., 2019). Recently, with the rapid development of deep learning, a deep belief network was proposed (Jia et al., 2018; Wang et al., 2017). Furthermore, some derived deep learning approaches have been adopted to MRR prediction, such as a feature-incorporated approach combined with a recurrent neural network and a convolutional neural network (Lee & Kim, 2020) and least squares generative adversarial network (Kim et al., 2020). Similar to MRR prediction in the industrial scenario, when facing the prediction problem (e.g. RUL estimation), there have been already mature solutions based on deep learning and machine learning (Ushakov & Zhang, 2019). Nevertheless, they often neglect the structural knowledge and underlined interactive mechanism of the equipment itself.

Industrial graph

Recent work on industrial graph can be mainly categorized into twofold: knowledge management and operation simulation.

The objective of the former one is to organize the data and knowledge from various resources in graph form systematically. It normally includes four steps: (1) schema design, (2) knowledge extraction, (3) knowledge fusion, and (4) reasoning. Firstly, a schema design is performed to define the node and edge in such domain-specific knowledge graph, since the node/edge types vary much (Wang et al., 2019). Next, knowledge extraction aims to collect triples (i.e., head entity, edge, and tail entity) from semi-/un-/constructed data by leveraging the natural language processing (Yan et al., 2020) and disassembly analysis (Weise et al., 2019). Then, it is essential to fuse the similar entities of extracted knowledge by creating an ontology link and building a concept graph (Li et al., 2020). Finally, after constructing the industrial knowledge graph, the querying process can be conducted by navigating potential key entities for making intelligent decisions (Wang et al., 2019) and recommendations (Li et al., 2021).

The latter one aims to digitize parts of the equipment information, the system working process, or even the entire production process, and connects the data in different vertical fields to construct a corresponding industrial graph. The most straightforward manner is to transform the working process

into a graph (Alsafi & Vyatkin, 2010) or disassembling the components as nodes in the graph (Hedberg et al., 2020). Furthermore, an event graph is generated to simulate and understand the manufacturing process and represent the event logic by setting events as entities in graph form (Tiacci, 2020).

However, both methods fail to represent the synergistic impact relation among components and parts in the equipment.

Graph-based reasoning

Graph neural network (GNN) is a prevailing methodology utilized to reflect the impact of interactions of graph-based structural data (Wu et al., 2021). GNN propagates the node attributes until convergence and generates embedding vectors for each node. Encouraged by the success of CNNs in computer vision, a graph convolution network (GCN) was proposed, which utilizes convolution for the spectral graph. After that, numerous researchers had developed improved and extensive versions of GCN by re-defining the convolution in the graph, such as lighten (He et al., 2020), and localize (Wang et al., 2018). Besides, some approaches are based on the spatial graph which convolutions on the graph directly (Yan et al., 2018). Among spatial theories, GraphSAGE (Hamilton et al., 2017) has achieved impressive performance, which inductively generates node embedding vectors. Furthermore, the attention mechanism had used to adjust the weight of the node base on their neighbor node (Velicković et al., 2017). In industrial applications, GCN has been utilized in manufacturing optimization (Hu et al., 2020), and modeling the equipment structure by determined the dependencies of sense data (Narwariya et al., 2018) or based on the Pearson Correlation Coefficient among their feature (Zhang et al., 2020).

Although some previous efforts attempt to establish the connection between pairwise sense data to form a graph. However, one interaction or synergistic mechanism in complex equipment may be related to more than two components and parts in a 'one-to-many' or 'many-to-many' relationship, which is out of the capability and expression of the conventional graph.

To address the abovementioned research gaps, this paper aims to propose a novel hypergraph convolution network (HGCN) based approach for MRR prediction in CMP, considering both the impact relationships between inherent component/parts and temporal features of data collected.

CMP hypergraph construction

To represent the complex impact relationships of multiple nodes in the CMP tool, this paper adopts the concept of hypergraph (Feng et al., 2019), of which an edge can join any number of nodes. This paper further introduces a CMP hypergraph model including three steps: (1) CMP graph data modelling; (2) hypergraph construction; (3) heterogeneous data correlation by the proposed HGCN-based model.

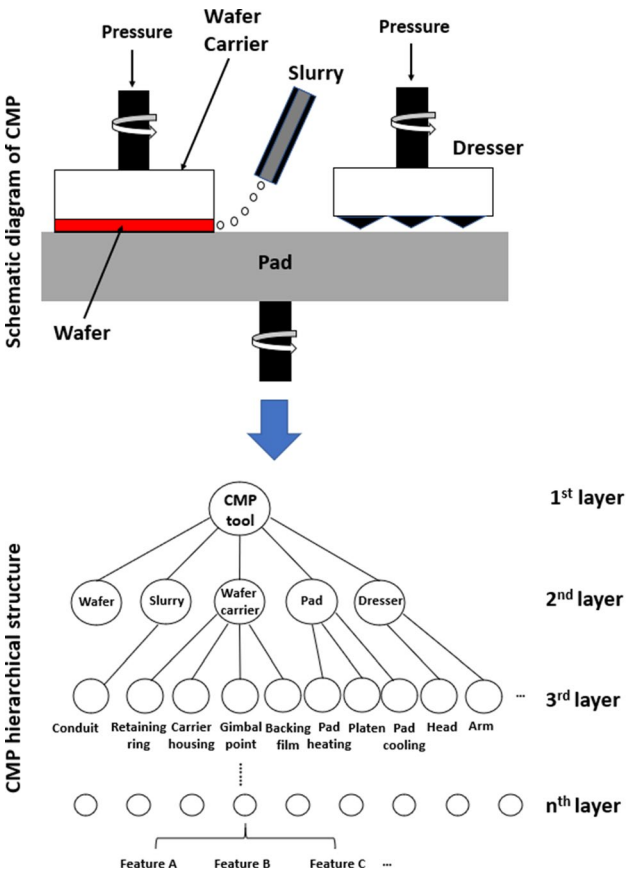

Fig. 1 CMP schematic diagram and corresponding hierarchical structure

CMP graph data model

Different from the existing industrial graph, the CMP graph data model, aims to reflect the impact among various components or parts, and to manage and represent the impact relationship and store their features in a graph form.

In the initial stage, it is essential to determine the components or parts involved as nodes in the graph, which are based on the physical structure and the operating mechanism. However, they are normally constructed in a hierarchical structural relationship. Hence, it is necessary to classify the hierarchical affiliation of all the nodes of the CMP graph data model into the following three levels, as shown in Fig. 1: (1) *product-level node*, as the top-level node in the hierarchical structural relationship, representing the product itself; (2) *part-level node*, denoting the individual product module in the second layer; and (3) *component-level node*,

referring to the ones decomposed by the product modules in the third layer to *n*th layer, of which the nodes in the *n*th layer contains its corresponding features.

In the CMP hierarchical structure (see Fig. 1), the top-level node is a product-level node representing the CMP tool product entity. Meanwhile, the CMP equipment modules are regarded as the part-level nodes (i.e., the wafer, slurry, wafer carrier, pad, and dresser), and the components of each module (e.g., conduit of slurry) are depicted in the component-levels, of which the *n*th layer are linked to the features.

Apart from the hierarchical structure, there also lies the impact relationship among various nodes of the same layer horizontally based on the equipment mechanism, such as the downward force of the wafer carrier to wafer. To better describe and summarize their impact, edges between the nodes can be utilized to represent their relationship in a graph-based form. Due to the complex relationship lying in the physical or chemical reactions between nodes, the types of edges should also be categorized as: (1) *undirected edge*, representing the two nodes that have hidden or fuzzy interaction; (2) *directed edge*, denoting that one node has a certain effect/action to the other, while not the other way around; and (3) *bi-directed edge*, referring to the certain effect/action on the nodes of each other. Based on the hierarchical and horizontal structure, the CMP graph data model can be established, as shown in Fig. 2.

According to the mechanism of CMP (Evans et al., 2003), for the part-level nodes (hollow circle in Fig. 2), for instances a downward physical force applies to the wafer carrier to push the wafer toward the pad, and therefore, a directed edge connects from the wafer carrier node to the wafer node. Meanwhile, the wafer material is passivated and etched by the slurry chemicals, which represents the slurry node has an impact on the wafer node. Also, the chemical interaction effect leads to an undirected edge connecting from the slurry node to the wafer node. Moreover, a downward force applies to the wafer to against the pad, and therefore, a directed edge connects from the wafer node to the pad node. Furthermore, the dresser is used to roughen the pad surface while the pad does not have effect/action to it reversely, leading to a directed edge from the dresser to the pad.

For the component-level node (filled circle in Fig. 2), first, in the Dresser node, the arm uses to fix the position of the head, so an undirected edge is connected from the arm to the head. Besides, in the pad node, the pad cooling device and the pad heating device heat conduct to the platen, so there are two directed edges from the pad cooling and the pad heating to the platen respectively. Meanwhile, in the wafer carrier node, as shown in Fig. 3, the backing film lay in the bottom, and due to the physical downward force to the wafer carrier, directed edges connect from the rest of the component-level nodes to the backing film. Moreover, the retaining ring and gimbal point lean on the carrier house without force, which have undirected edges among them. Furthermore, the last component layer nodes are connected with their corresponding data features (dashed line in Fig. 2).

Hypergraph construction

The CMP graph data model has clarified the relationship among different level nodes, while it is still difficult to determine the exact mathematical expression or weight of each edge due to the limitation of data and prior knowledge available.

To fill this gap, this paper proposes a hypergraph to represent their complex relationship in the CMP equipment. The main characteristic of the hypergraph is using a hyperedge to connect with multiple nodes which indicates the impact interaction among the connected nodes. There are three types of hyperedge and summarizes in Table 1.

After constructing the CMP graph data model in Fig. 2, it needs to consider which edge can be merged as a hyperedge based on their operation mechanism. For the part-level nodes, firstly the wafer node is influenced by both the wafer carrier node and the slurry node. Because the downward

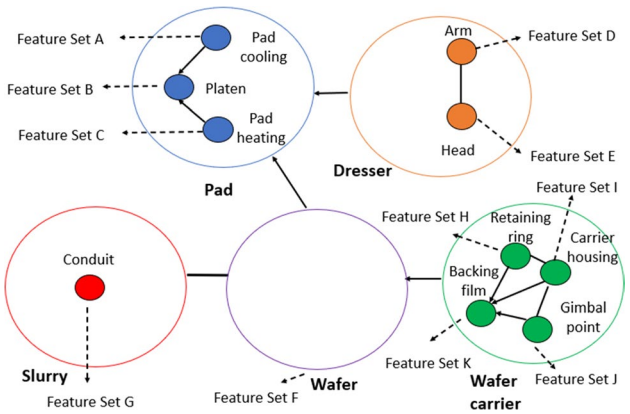

Fig. 2 CMP graph data modelling

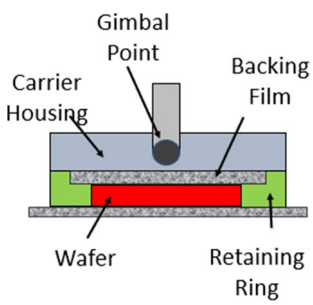

Fig. 3 Assembly of the wafer carrier

Table 1 Different type of hyperedge and its vector

Edge type	Head entity	Tail entity	Example	Vector
Undirected	1	1	n_1 e n_2	n_1 [1, 1] n_2
Directed	1	− 1	n_1 e n_2	n_1 [1, − 1] n_2
Bi-directed	1	1	n_1 e n_2	n_1 [1, 1] n_2

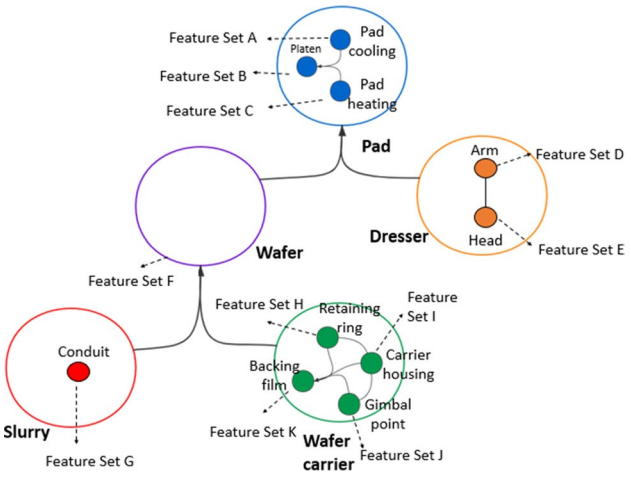

Fig. 4 CMP hypergraph

force is applied on the wafer, leading to its contact area is changed in the chemical reaction with slurry. Simultaneously, the wafer is removed by the chemical reaction of the slurry which also influent the effect of original downward pressure on the wafer node. Therefore, it is difficult to distinguish how the slurry and the wafer carrier influence the wafer module respectively, it needs to merge these two edges as a hyperedge to represent the associated impact relationship. Secondly, the wafer and the dresser are setting up on the pad vertically, both of which are applied a downward force pushing the pad node indirectly. Both the wafer node and the dresser node have certain actions on the same pad node, thereby it is difficult to divide them separately. Accordingly, a hyperedge connecting the wafer node and the dresser node to the pad node should be used to represent this associated impact relationship. After analyzing the relationship between different part-level nodes, a hypergraph is generated and each of the part-level nodes contains one or more associated component-level nodes, as shown in Fig. 4.

Furthermore, the hypergraph construction of component-level nodes also follows the same analysis logic. In the Pad module, the heat conduct transfers from the pad cooling device and the pad heating device to the platen, because the heat conduct is discrete and hard to calculate separately, a directed hyperedge connects from pad cooling and pad heating to the platen. Meanwhile, in the wafer carrier node, the retaining ring, the carrier housing, and the gimbal point set up on the backing film with a downward physical force. Hence there is a hyperedge connects from the former three component-level nodes to the backing film node. Additionally, the retaining ring and the gimbal point place nearby the carrier housing horizontally, therefore there are undirected edges connects from the retaining ring and the gimbal point to carrier housing separately. After the analysis of the CMP mechanism, the visual hypergraph can be seen in Fig. 4 which contains directed hyperedges and undirected hyperedges, and its corresponding hypergraph matrix can be applied according to Table 1.

HGCN-based model

This paper introduces the HGCN-based model to predict the wafer removal rate in the CMP. The input data in the proposed model have samples across different time dimensions and each feature in the sample belongs to a corresponding part-level node or component-level node. The schematic diagram of the HGCN-based model is shown in Fig. 5 and the main notations is shown in Table 2. This paper focuses on modeling the interrelationships among the part-level nodes, and the component-level nodes follow the same modeling process.

Embedding layer

The different part-level nodes contain different number of features which are uneven and difficult for the subsequent modules to use. Therefore, this paper proposes the embedding layers to transfer the different dimensions vector into the same fixed dimension (128 dimensions). The embedding equation is as follows:

$$z_j = z_j^{'} w_z + b_z, \tag{1}$$

where $z_j^{'}$ denotes the part-level node with original features, z_j denotes the embedding vector of part-level node, $w_z \in \mathbb{R}^{od \times ed}$ denotes the embedding matrix, *od* is the original dimension and *ed* is the embedding dimension. For instance, part-level node *wafer* has 3 features, therefore for each timestamp t, its vector is $z_{pad,t}^{'} \in \mathbb{R}^{1\times 3}$. After embedding layer, $z_{pad,t}^{'}$ transfers to $z_{pad,t} \in \mathbb{R}^{1\times 128}$, which contains larger representation spaces.

Piecewise aggregate approximation

The length of timestamps of each wafer sample is different. Therefore, it is necessary to reduce to the same timestamp length for training efficiently. This paper introduces Piecewise aggregate approximation (PAA) to convert

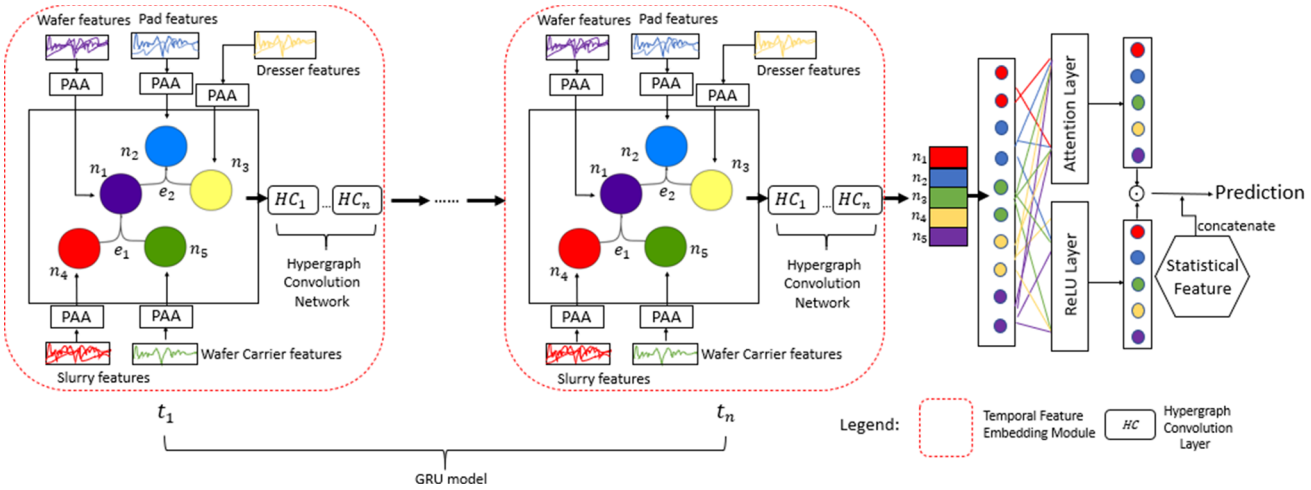

Fig. 5 The schematic diagram of the HGCN-based model

Table 2 The main notations and definitions in this paper

Notations	Space	Definitions
n	$\mathbb{R}$	THE total number of timestamps
$z_t^{'}$	$\mathbb{R}^{1\times k}$	Part-level node with original k-features
z_t	$\mathbb{R}^{1\times 128}$	Part-level node embedding vector before Piecewise aggregate approximation
$x_t^{'}$	$\mathbb{R}^{1\times 128}$	Part-level node embedding vector after Piecewise aggregate approximation
x_t	$\mathbb{R}^{5\times 128}$	Embedding representations of five part-level nodes at timestamp t
A	$\mathbb{R}^{5\times 2}$	Hypergraph matrix
x_t^l	$\mathbb{R}^{5\times 128}$	The node embedding vectors at lth layer at timestamp t
h_t	$\mathbb{R}^{5\times 128}$	Output vector of Gated Recurrent Unit at timestamp t
h_{tk}	$\mathbb{R}^{1\times 128}$	The kth node embedding vector at timestamp t
$h_{ti}^{'}$	$\mathbb{R}^{1\times 128}$	The ith node embedding vector after graph attention mechanism at timestamp t
H'	$\mathbb{R}^{1\times 640}$	The hypergraph embedding vector

different wafer samples into the same length, and the targeted length sets as the minimal time length among all the wafer samples. The mathematical algorithm of the PAA can be written as:

$$x_i^{'} = \frac{n}{m} \sum_{j=\frac{m}{n}(i-1)+1}^{\frac{m}{n}i} z_j, \tag{2}$$

where $z_1, \ldots, z_m$ denote a wafer sample with m timestamps, and n is the minimal time length. $x_1^{'}, \ldots, x_n^{'}$ are the n number of 128-dimensional vectors. Each $x_t^{'} \in \mathbb{R}^{1\times 128}$ represents the specific node embedding vector in timestamp t. Because the CMP tool has five part-level nodes, we concatenate them into $x_t \in \mathbb{R}^{5\times 128}$, and represent the equipment structure. Hence, we employ $x_1, \ldots, x_n$ to denote embedding representations of five part-level nodes in n timestamps.

Hypergraph convolution network

The hypergraph convolution network (HGCN) (Feng et al., 2019) is introduced to learn the data correlation and output refined embedding vectors with the same dimensions. By applying the Fourier transform to the spectral convolution and inverse Fourier transform, the HGCN can be iterated as the following function:

$$x_t^l = \sigma\left(D_v^{-\frac{1}{2}} A W D_e^{-1} A^T D_v^{-\frac{1}{2}} x_t^{l-1} \Theta^{l-1}\right), \tag{3}$$

where $x_t^0 = x_t$, σ denotes the sigmoid function, W denotes the trainable diagonal matrix, D_v and D_e denote the diagonal matrices of edges degrees and the nodes degrees, A denotes hypergraph matrix which calculates from Table 1 and CMP

hypergraph (Fig. 4), and $\Theta \in \mathbb{R}^{C_1 * C_2}$, denotes the convolution filter to inverse transform to the spatial domain, C_1 and C_2 are the feature dimensions before and after convolution. This hypergraph iteration equation utilizes the core idea of graph convolutional networks. As shown in Fig. 6, the HGCN can achieve node-edge-node transformation so that it can extract the high order features base on the hypergraph structure. Initially, x_t^l multiplies of A^T can transform the node level embedding vectors into hyperedge embedding vectors, representing gather information to the hyperedges. Subsequently, by multiplying matrix A, it can generate the refined node embedding vectors which means aggregated their related hyperedge embedding vectors (the lower part of Fig. 6). Therefore, by utilizing this node-hyperedge-node mechanism, the HGCN can extract the high-order feature efficiently.

For the hypergraph of part-level nodes in Fig. 4, it contains two hyperedges and five part-level nodes, hence $\mathrm{A} \in \mathbb{R}^{5\times 2}$. The $x_t^l \in \mathbb{R}^{5\times 128}$ in Eq. (3) is one timestamp unit of the whole temporal data, and its dimensions remain the same through the HGCN layer.

Gated recurrent unit

After applying HGCN in each timestamp, it generates sequence data $x_1^l, \ldots, x_n^l$. Establishing a Gated recurrent unit (GRU) model for the sequence data to obtain the prediction. The main idea of GRU is to use a gate mechanism (i.e., update gate and reset gate). The mathematical algorithm of GRU is as follows:

$$z_t = \sigma\left(W_z x_t^l + U_z h_{t-1}\right), \tag{4}$$

$$r_t = \sigma\left(W_r x_t^l + U_r h_{t-1}\right), \tag{5}$$

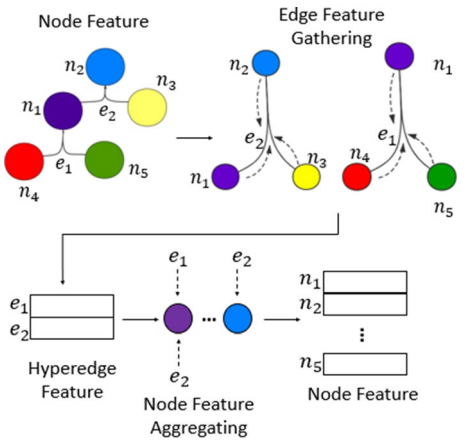

Fig. 6 The illustration of HGCN

$$\widehat{h}_t = \tanh\left(W_h x_t^l + U_h\left(r_t \odot h_{t-1}\right) + b_h\right), \tag{6}$$

$$h_t = \left(1 - z_t\right) \odot h_{t-1} + z_t \odot \widehat{h}_t, \tag{7}$$

where h_t is the output vector, and $W_z, W_r, W_h, U_z, U_r, U_h, b_h$ are the trainable parameters. Setting the hidden layer number as same as the input dimensions, hence $h_t \in \mathbb{R}^{5\times 128}$.

Hypergraph attention mechanism

GRU module's output h_t represents the vertical concatenation of the nodes' embedding vectors, denotes $h_{tk} \in \mathbb{R}^{1\times 128}$ as the kth node in the graph. Also, h_{tk} can be refined by applying graph attention mechanism. In this hypergraph attention mechanism, it considers its first order neighbor to calculate its attention coefficient a_{ij}. Also, the nodes will treat as neighbors if they connect with a hyperedge in the hypergraph. The updated $h_{ti}^{'} \in \mathbb{R}^{1\times 128}$ can be iterated by:

$$h_{ti}^{'} = \sigma\left(\sum_{j\in N_i} a_{ij} W_a h_{tj}\right), \tag{8}$$

where W_a is the trainable weight matrix, and a_{ij} is the impact factor, which can be calculated as follows:

$$a_{ij} = \frac{\exp\left(LeakyReLU\left(\lambda^T\left[W_a h_{ti} || W_a h_{tj}\right]\right)\right)}{\sum_{k\in N_i} \exp\left(LeakyReLU\left(\lambda^T\left[W_a h_{ti} || W_a h_{tj}\right]\right)\right)} \tag{9}$$

where N_i denotes the neighbor of the ith node, λ is the weight vector applies in the LeakyReLU function, and || is the concatenation process. The refined output from the hypergraph attention mechanism can be readout as a graph embedding vector by concatenating them horizontally, denotes the graph embedding vector as $H' = \left[h_{t1}^{'}, \ldots, h_{t5}^{'}\right]$ and $H' \in \mathbb{R}^{1\times 640}$.

Comprehensive representation

Overall, the architecture of the HGCN-based model is shown in Fig. 7. Although it can handle the heterogeneous vectors of the equipment structure, statistical features also benefit to the prediction result. Therefore, this proposed algorithm concatenates three statistical metrics of each feature: *standard deviation, skewness,* and *kurtosis* with the graph embedding vector as the comprehensive representation and denotes it as $H'' = \left[H', X_{extra}\right]$, where X_{extra} is the feature set of statistical metrics. Hence, the final estimated value can be calculated through a fully connected layer as:

$$x_{hidden} = ReLU\left(W_d * H'' + b_{h1}\right), \tag{10}$$

$$y_{output} = W_o * x_{hidden} + b_{h2}, \tag{11}$$

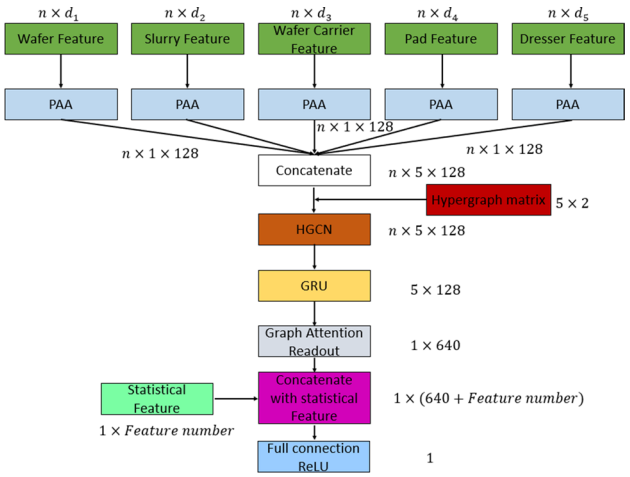

Fig. 7 The detailed structure of HGCN-based model for part-level nodes

where *ReLU* is the non-linear active function, W_d, W_o, b_{h1}, b_{h2} are the trainable parameters. Finally, the model trains through backpropagation with the mean square error as the loss function:

$$L = \frac{1}{n}\sum_{i}^{n}\left(y_{output} - y_{true}\right)^2 \tag{12}$$

Case study

To demonstrate the effectiveness of the proposed approach in a generic manner, one open dataset obtained from the competition of PHMS 2016 of the wafer CMP (Wang et al., 2017) is adopted to predict the average material removal rate.

Data description

The dataset contains multiple sensory signals collecting from a CMP that removes the material from wafers. This paper selects 14 features out of 25 total features, which are relevant to the parts and components in the CMP tool. They mainly include the usage of the polish-pad backing film, dresser, polishing table, dresser table, wafer carrier sheet, the flow rate of slurry, and the pressure of different components. Besides, the time length ranges from 199 to 5492, but they all correspond to one MRR (target). The dataset includes two stages: A and B. The number of the total dataset of stage A is 376,859 and corresponding to 1166 wafers records (i.e., a distinct wafer id has many timestamps but one corresponding MRR) and the total dataset of stage B is 295,885 and corresponding to 815 wafers records this experiment split 80% of the dataset as the training dataset and the rest as the test dataset. Table 3 provides numerical details on the training and test dataset.

Average removal rate prediction

Hypergraph matrix

Due to the limited features, it fails to construct the complete CMP graph data model as shown in Fig. 5. Nevertheless, since all the features in the open dataset are related to the part-level nodes, this paper considers only involves those ones holistically. Following the same analysis described in "Hypergraph construction" section, its hypergraph data model and corresponding hypergraph matrix H can be represented, as shown in Fig. 8.

Performance metrics

To evaluate the performance, the error will be measured by the following metrics:

$$MSE = \frac{1}{m}\sum_{i=1}^{m}\left(y_i - \hat{y}_i\right)^2 \tag{13}$$

the full name of MSE is mean squared error, which measures the average squared difference between the estimated values and the actual value.

Table 3 Training and test dataset numerical statistics

Type	Training dataset	Test dataset
Total number of observations	535,591	137,153
Number of wafers	1584	397
Number of wafers of stage A	932	234
Number of observations of stage A	301,045	75,814
Number of wafers of stage B	652	163
Number of observations of stage B	234,546	61,339

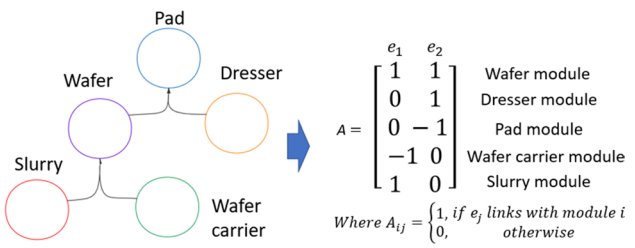

Fig. 8 The matrix of the CMP hypergraph structure

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|y_i - \hat{y}_i|, \tag{14}$$

the full name of MAE is mean absolute error, which measures the errors between the estimated values and the actual value expressing the same phenomenon.

Hyperparameters

This experiment uses an Adam optimizer with an initial learning rate of 0.01, the dimensions number of each vector is 128, the dropout rate of 0.1 for all feedforward layer, the MSE as the loss function, and the heads number of graph attention mechanism is 1, the number of HGCN layer is 2, the epoch is 100, and the batch size is 128.

Comparable cutting-edge models

To validate the advantages of the proposed model, it is compared with cutting-edge models adopted in the prognostic and health management field with the same hyperparameters, as listed below:

CNN-MR: Deep convolutional neural network-based regression approach (Babu et al., 2016).

LSTM-MR: Long Short-Term Memory approach for prediction (Zheng et al., 2017).

GRU-MR: Gated Recurrent Unit model for prediction (Yan et al., 2019).

Auto-Encoder + DNN: Using Auto-Encoder to generate additional features and feed them with the original features into DNN (Ren et al., 2018).

The experiment is conducted with a fivefold cross-validation mechanism, data normalization, and early stop for generating a stable and better result. The comparison results with cutting-edge approaches can be seen in the upper part of Table 4 of each metric.

Meanwhile, the effectiveness of the HGCN-based model is validated by comparing the proposed model with the models without different submodules, and the verification result can be seen in the lower part of Table 4 of each metric. Hereby, (1) the proposed model without the HGCN layer represents the node embedding vectors remain the same after the construction of hypergraph and before the hypergraph attention mechanism; (2) the proposed model without hypergraph means all the operations related to the graph will be removed, such as hypergraph convolution layer and graph attention mechanism, and graph readout process; (3) the proposed model without statistical features represents the model does not concatenate with the statistical feature before DNN; and (4) the proposed model without temporal features represents train the DNN model only with the statistical features.

Table 4 Performance comparison

Model	Stage A	Stage B
MSE		
CNN-MR (Babu et al., 2016)	0.000098	0.037468
LSTM-MR (Zheng et al., 2017)	0.000105	0.037490
GRU-MR (Yan et al., 2019)	0.000149	0.038041
Auto-Encoder + DNN (Ren et al., 2018)	0.000094	0.037589
Proposed model without HGCN layer	0.000077	0.037357
Proposed model without hypergraph	0.000078	0.038248
Proposed model without statistical features	0.000093	0.037757
Proposed model only with statistical features	0.000080	0.037392
Proposed model	0.000075	0.036672
MAE		
CNN-MR (Babu et al., 2016)	0.009669	0.162875
LSTM-MR (Zheng et al., 2017)	0.008618	0.161538
GRU-MR (Yan et al., 2019)	0.009353	0.163359
Auto-Encoder + DNN (Ren et al., 2018)	0.009467	0.162468
Proposed model without HGCN layer	0.008572	0.161841
Proposed model without hypergraph	0.008590	0.163655
Proposed model without statistical features	0.009487	0.163352
Proposed model only with statistical features	0.008561	0.160379
Proposed model	0.007523	0.159504

Table 5 HGCN-based model performance of different matrix

Different hypergraph matrix	Stage A	Stage B
MSE		
Proposed matrix	0.000084	0.036816
Proposed matrix without direction	0.000086	0.036852
Identity matrix	0.000094	0.037134
Random matrix	0.000091	0.037219
MAE		
Proposed matrix	0.009034	0.150175
Proposed matrix without direction	0.009150	0.150220
Identity matrix	0.009519	0.150494
Random matrix	0.009403	0.150715

Furthermore, to validates the correctness of the hypergraph matrix form, the experiment compares the proposed hypergraph matrix with other matrix and the random matrix, of which the experiment results are shown in Table 5.

Discussion

Based on the experiment results obtained from Tables 4 and 5, some further analysis can be conducted as follows.

Comparison with baselines

According to Table 4, the proposed HGCN-based model outperforms the other cutting-edge models: CNN-MR, LSTM-MR, GRU-MR, and Auto-Encoder + DNN, in both Stage A and Stage B of MRR. The result shows that by combining the equipment structure as a hypergraph form into a deep learning approach, this structure can provide meaningful and beneficial knowledge for the prediction task, and hence the proposed hypergraph construction method is effective. Theoretically, the hyperedge links with more than two nodes, representing the synergistic mechanism involves more than two components in the complex equipment. The convolution layer exploits the complex and high-order relationships in the hypergraph for representation learning. Therefore, the proposed model outperforms other cutting-edge models which neglect the structural knowledge.

Effective of HGCN-based structure

One unique characteristic of the proposed HGCN-based model is that it contains a hypergraph structure and uses hypergraph convolution layers to learn the hidden data correlation. To validate its effectiveness, four scenarios are considered as shown in Table 4, where the proposed model achieves the lowest MSE and MAE compared with the ones without different submodules. Also, the HGCN, hypergraph, statistical features have positive contributions to the prediction accuracy.

The correctness of hypergraph matrix

The experiment also compares the difference performance brought by mechanism-based hypergraph matrix and different matrices. As shown in Table 5, the proposed hypergraph matrix achieves better performance than the proposed undirected hypergraph matrix (all hyperedges are undirected), identity matrix, and random matrix. This experiment verifies the correctness of the proposed hypergraph matrix and further proves that the proposed hypergraph construction method can express the impact relationship efficiently.

Limitations

Despite the above advantages, some parts of the model in this research work are simplified, for instances: (1) *Weighting*. The proposed model only reflects the different impaction by training the node's weight matrix, but assuming all the hyperedge have the same weight. However, the impact relationship is varying from different components, which are influenced by its nodes and hyperedges. (2) *Hyperedge*. The hypergraph attention mechanism treats the nodes connected with the same hyperedge as the first order neighbor, which may not be precise enough as a fully connected edge.

In summary, this proposed model can effectively predict MRR in the CMP tool, by learning the complex and high-order correlations among the heterogeneous data in the representative hypergraph. As a generic methodology proposed, it can also be further implemented in similar scenarios in the manufacturing process with complex impact relationships.

Conclusion

MRR prediction plays a critical role in the CMP process. However, existing methodologies normally neglect the structural knowledge of the CMP tool, which contains a large amount of hidden information that can also improve the MRR prediction. To tackle this challenge, this paper firstly provided a novel framework to construct a CMP hypergraph data model, which represents the impact relationship of different components and parts in the CMP tool. Secondly, this paper proposes a novel HGCN-based model to learn the data correlation and to aggregate the node information in hypergraph for MRR prediction with temporal data. A case study was conducted revealing that the proposed HGCN-based model is capable to combine the hypergraph structure and node features effectively, and it outperforms the cutting-edge models in MRR prediction. The key contributions of this research can be summarized as follow:

1. Proposed a systematic manner to transform the complex equipment structure into the representative hypergraph data model, which can reflect the complex impact relationship among components and parts effectively.
2. Introduced a novel approach to embedding the node with various features and different time lengths into the fixed dimensions and time length, which benefits subsequent model training effectively and rapidly.
3. Proposed the HGCN-based model for MRR prediction. This model integrated the HGCN, hypergraph attention mechanism and GRU, which can learn the heterogeneous data correlation more efficiently. As the experiment result shown, it outperformed previous cutting-edge models in several metrics.

Apart from the case study of MRR prediction in the CMP tool, it is envisioned that this research can also bring insightful ideas or guide to relevant tasks among other complex manufacturing process. However, this research work still has some limitations as pointed out in "Discussion" section. Taking all these factors into consideration, it is recommended that future works can be done to: (1) involve the environmental effect of the complex equipment (e.g., the

chamber pressure), which may also affect the performance of equipment; (2) consider the weightings of hyperedge; and (3) describe the neighbor relationship of different orders in the hypergraph.

Funding This research work was partially supported by the grants from the National Natural Research Foundation of China (No. 52005424), and Research Committee of The Hong Kong Polytechnic University (G-UAHH).

Availability of data and material Data can be found via: https://www.phmsociety.org/events/conference/phm/16/data-challenge.

Code availability Not applicable.

Declarations

Conflict of interest We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome. We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed.

References

Alsafi, Y., & Vyatkin, V. (2010). Ontology-based reconfiguration agent for intelligent mechatronic systems in flexible manufacturing. *Robotics and Computer-Integrated Manufacturing, 26*(4), 381–391. https://doi.org/10.1016/j.rcim.2009.12.001

Babu, G. S., Zhao, P., & Li, X. L. (2016). Deep convolutional neural network based regression approach for estimation of remaining useful life. In *International conference on database systems for advanced applications* (pp. 214–228). Springer, Cham. https://doi.org/10.1007/978-3-319-32025-0_14.

Evans, C. J., Paul, E., Dornfeld, D., Lucca, D. A., Byrne, G., Tricard, M., et al. (2003). Material removal mechanisms in lapping and polishing. *CIRP Annals - Manufacturing Technology, 52*(2), 611–633. https://doi.org/10.1016/S0007-8506(07)60207-8

Feng, Y., You, H., Zhang, Z., Ji, R., & Gao, Y. (2019). Hypergraph neural networks. In *Proceedings of the AAAI conference on artificial intelligence* (vol. 33, no. 01, pp. 3558–3565). https://doi.org/10.1609/aaai.v33i01.3301358.

Hamilton, W. L., Ying, R., & Leskovec, J. (2017). *Inductive representation learning on large graphs*. arXiv preprint https://arxiv.org/abs/1706.02216

He, X., Deng, K., Wang, X., Li, Y., Zhang, Y. D., & Wang, M. (2020). LightGCN: Simplifying and powering graph convolution network for recommendation. In *SIGIR 2020—Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval* (pp. 639–648). https://doi.org/10.1145/3397271.3401063.

Hedberg, T. D., Bajaj, M., & Camelio, J. A. (2020). Using graphs to link data across the product lifecycle for enabling smart manufacturing digital threads. *Journal of Computing and Information Science in Engineering, 20*(1), 1–15. https://doi.org/10.1115/1.4044921

Hong, S., Han, D., Kwon, J., Kim, S. J., Lee, S. J., & Jang, K.-S. (2020). Influence of abrasive morphology and size dispersity of Cu barrier metal slurry on removal rates and wafer surface quality in chemical mechanical planarization. *Microelectronic Engineering*. https://doi.org/10.1016/j.mee.2020.111417

Hu, L., Liu, Z., Hu, W., Wang, Y., Tan, J., & Wu, F. (2020). Petri-net-based dynamic scheduling of flexible manufacturing system via deep reinforcement learning with graph convolutional network. *Journal of Manufacturing Systems, 55*, 1–14. https://doi.org/10.1016/j.jmsy.2020.02.004

Jia, X., Di, Y., Feng, J., Yang, Q., Dai, H., & Lee, J. (2018). Adaptive virtual metrology for semiconductor chemical mechanical planarization process using GMDH-type polynomial neural networks. *Journal of Process Control, 62*, 44–54. https://doi.org/10.1016/j.jprocont.2017.12.004

Kim, S., Jang, J., & Kim, C. O. (2020). A run-to-run controller for a chemical mechanical planarization process using least squares generative adversarial networks. *Journal of Intelligent Manufacturing*. https://doi.org/10.1007/s10845-020-01639-1

Kong, Z., Oztekin, A., Beyca, O. F., Phatak, U., Bukkapatnam, S. T. S., & Komanduri, R. (2010). Process performance prediction for chemical mechanical planarization (CMP) by integration of nonlinear Bayesian analysis and statistical modeling. *IEEE Transactions on Semiconductor Manufacturing, 23*(2), 316–327. https://doi.org/10.1109/TSM.2010.2046110

Lee, H., & Jeong, H. (2011). A wafer-scale material removal rate profile model for copper chemical mechanical planarization. *International Journal of Machine Tools and Manufacture, 51*(5), 395–403. https://doi.org/10.1016/j.ijmachtools.2011.01.007

Lee, H. S., Jeong, H. D., & Dornfeld, D. A. (2013). Semi-empirical material removal rate distribution model for SiO_2 chemical mechanical polishing (CMP) processes. *Precision Engineering, 37*(2), 483–490. https://doi.org/10.1016/j.precisioneng.2012.12.006

Lee, K. B., & Kim, C. O. (2020). Recurrent feature-incorporated convolutional neural network for virtual metrology of the chemical mechanical planarization process. *Journal of Intelligent Manufacturing, 31*(1), 73–86. https://doi.org/10.1007/s10845-018-1437-4

Li, X., Chen, C.-H., Zheng, P., Jiang, Z., & Wang, L. (2021). A context-aware diversity-oriented knowledge recommendation approach for smart engineering solution design. *Knowledge-Based Systems, 215*, 106739. https://doi.org/10.1016/j.knosys.2021.106739

Li, X., Chen, C.-H., Zheng, P., Wang, Z., Jiang, Z., & Jiang, Z. (2020). A knowledge graph-aided concept–knowledge approach for evolutionary smart product–service system development. *Journal of Mechanical Design, 142*(10), 1–19. https://doi.org/10.1115/1.4046807

Li, Z., Wu, D., & Yu, T. (2019). Prediction of material removal rate for chemical mechanical planarization using decision tree-based ensemble learning. *Journal of Manufacturing Science and Engineering, Transactions of the ASME, 141*(3), 1–14. https://doi.org/10.1115/1.4042051

Ludwig, C., & Kuna, M. (2012). An analytical approach to determine the pressure distribution during chemical mechanical polishing. *Journal of Electronic Materials, 41*(9), 2606–2612. https://doi.org/10.1007/s11664-012-2151-1

Narwariya, J., Malhotra, P., Vishnu, T. V., Vig, L., & Shroff, G. (2018). *Graph neural networks for leveraging industrial equipment structure: An application to remaining useful life estimation*. arXiv preprint https://arxiv.org/abs/2006.16556.

Ren, L., Sun, Y., Cui, J., & Zhang, L. (2018). Bearing remaining useful life prediction based on deep autoencoder and deep neural networks. *Journal of Manufacturing Systems, 48*, 71–77. https://doi.org/10.1016/j.jmsy.2018.04.008

Tiacci, L. (2020). Object-oriented event-graph modeling formalism to simulate manufacturing systems in the Industry 4.0 era. *Simulation Modelling Practice and Theory, 99*, 102027. https://doi.org/10.1016/j.simpat.2019.102027

Ushakov, S., & Zhang, H. (2019). A comprehensive survey of prognostics and health management based on deep learning for autonomous ships. *IEEE Transactions on Reliability, 68*(2), 720–740. https://doi.org/10.1109/TR.2019.2907402

Velicković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2017). *Graph attention networks*. arXiv preprint https://arxiv.org/abs/1710.10903.

Wang, C., Samari, B., & Siddiqi, K. (2018). Local spectral graph convolution for point set feature learning. In *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics), 11208 LNCS* (pp. 56–71). https://doi.org/10.1007/978-3-030-01225-0_4

Wang, P., Gao, R. X., & Yan, R. (2017). A deep learning-based approach to material removal rate prediction in polishing. *CIRP Annals - Manufacturing Technology, 66*(1), 429–432. https://doi.org/10.1016/j.cirp.2017.04.013

Wang, Z., Chen, C. H., Zheng, P., Li, X., & Khoo, L. P. (2019). A graph-based context-aware requirement elicitation approach in smart product-service systems. *International Journal of Production Research, 59*(2), 635–651. https://doi.org/10.1080/00207543.2019.1702227

Weise, J., Benkhardt, S., & Mostaghim, S. (2019). A survey on graph-based systems in manufacturing processes. In *Proceedings of the 2018 IEEE symposium series on computational intelligence, SSCI 2018* (pp. 112–119). https://doi.org/10.1109/SSCI.2018.8628683.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Yu, P. S. (2021). A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems, 32*(1), 4–24. https://doi.org/10.1109/TNNLS.2020.2978386

Xu, Q., Chen, L., Liu, J., & Cao, H. (2020). A wafer-scale material removal rate model for chemical mechanical planarization. *ECS Journal of Solid State Science and Technology, 9*(7), 074002. https://doi.org/10.1149/2162-8777/abadea

Yan, H., Yang, J., & Wan, J. (2020). KnowIME: A system to construct a knowledge graph for intelligent manufacturing equipment. *IEEE Access, 8*, 41805–41813. https://doi.org/10.1109/ACCESS.2020.2977136

Yan, S., Xiong, Y., & Lin, D. (2018). Spatial temporal graph convolutional networks for skeleton-based action recognition. In *32nd AAAI conference on artificial intelligence, AAAI 2018* (vol. 32, no. 1). https://doi.org/10.1186/s13640-019-0476-x.

Yan, Y., Fang, H., & Li, Z. (2019). Lithium-ion battery remaining useful life prediction based on an integrated method. In *Proceedings of 2019 IEEE 8th data driven control and learning systems conference, DDCLS 2019* (pp. 592–597). https://doi.org/10.1109/DDCLS.2019.8908992.

Zhang, Y., Li, Y., Wei, X., & Jia, L. (2020). Adaptive spatio-temporal graph convolutional neural network for remaining useful life estimation. In *2020 International joint conference on neural networks (IJCNN)* (pp. 1–7). https://doi.org/10.1109/IJCNN48605.2020.9206739.

Zheng, S., Ristovski, K., Farahat, A., & Gupta, C. (2017). Long short-term memory network for remaining useful life estimation. In *2017 IEEE international conference on prognostics and health management, ICPHM 2017* (pp. 88–95). https://doi.org/10.1109/ICPHM.2017.7998311.