https://dx.doi.org/10.1016/j.cie.2021.107864

A node2vec-based graph embedding approach for unified assembly process information modeling and workstep execution time prediction

Qiangwei Bao[a,b], Gang Zhao[a,c], Yong Yu[a,d,*] , Pai Zheng[e]

[a]School of Mechanical Engineering and Automation, Beihang University, Beijing 100191, PR China

[b]School of Energy and Power Engineering, Beihang University, Beijing 100191, PR China

[c]Key Laboratory of Aeronautics Smart Manufacturing, Beihang University, Beijing 100191, PR China

[d]Beijing Engineering Technological Research Center of High-Efficient & Green CNC Machining Process, Beijing 100191, PR China

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

Abstract

The trend of customized production results in the demand for higher level of automation, in which artificial intelligence decision-making dominates. As the core of smart manufacturing systems, intelligent services stand a significant role in the analysis, prediction, and adjustment of production process, which is inseparable from the effective semantic modeling of the procedures and elements involved. However, there is an absence of unified modeling of assembly process, including both geometric and non-geometric information, leading to the incomprehensiveness when providing data support for intelligent services. To fill this gap, a generic node2vec-based parameterized representation of geometric elements and assembly constraints approach is proposed. Firstly, the information structure of assembly process is established, in which the geometric elements and topological relationships of the product are abstracted into a network. Secondly, node2vec is adopted for the graph embedding to generate preset dimension vectors corresponding to the nodes in the geometric network. As the edges in the network, the vectors corresponding to the assembly constraints, which are regarded as the parameterized representations, can be obtained through node vector calculation. Moreover, an assembly workstep execution time prediction method based on historical data is introduced with the parameterized representations of assembly constraints as the carriers of geometric topological information. At last, an industrial case study is illustrated to show the entire process of constraint parameterized representations and workstep execution time prediction, indicating the feasibility and availability of the method proposed.

Keywords

Assembly process modeling, assembly constraint, parameterized representation, node2vec, workstep prediction

1 Introduction

In the last decades, the continuous promotion of product customization puts forward higher requirements

*Corresponding author: Yong Yu
E-mail: yuyong@buaa.edu.cn
ORCID: 0000-0001-9524-7230

for real-time control and adjustment in production (Wang et al., 2019). To accommodate the increasing pace of production, the decision-making pattern has transformed from manual to artificial intelligence in optimization (Cui et al., 2020), further eliciting the paradigm of self-optimization in cyber-physical systems (Lee et al., 2015). With the pursuit of higher automation and intelligence degree of industrial production, cognitive manufacturing (IBM, 2017) has received more attention as a new manufacturing pattern. Compared with the method of rational or perceptual manufacturing, cognitive manufacturing embraces cognitive computing, the industrial Internet-of-Things, and advanced analytics to optimize manufacturing processes (Chen et al., 2018), which endows manufacturing systems with human-level perception and judgment abilities (Zheng et al., 2021) to perceive the alterations of production processes and respond to the dynamic fluctuations of production links (Terziyan et al., 2018).

It is obvious that information modeling and circulation play an essential role in the aspects of cognitive-based production (Zolotová et al., 2020), including modeling of processes and resources, description of production status, mathematical preparation for advanced analytics, et al. The integration and transformation of production information reflect the indispensability of semantization in smart manufacturing systems (Zheng et al., 2018; Zhang et al., 2021). Meanwhile, the industrial big data-based cognitive computing requires better commonality in semantic modeling to expand coverage apart from consistency (Belhadi et al., 2019). As an essential part, process information contains correlative methods, parameters, and resources, which can restore the execution of production tasks and further provide guidance for the process formulation of subsequent products (Bao et al., 2019). Nevertheless, there still lacks a unified semantic-based information modeling method for processes due to the multiformity of products and richness of production modes considering the massive heterogeneous information (Lu et al., 2020), especially in the field of assembly.

Among the manufacturing domains, assembly occupies a crucial position with 40%-60% of the time cost in product research, development, and production (Liu et al., 2018). However, the diversity and complexity of assembly process (Bao et al., 2021) raise two significant difficulties in semantic modeling, which are detailed as follows:

(1) *There are obvious differences among disparate assembly processes for the diversity of product structures and characteristics, leading to the discreteness when modeling separate assembly links.* Traditional assembly analysis usually requires a separate solution for a specific workstep or operation of a product, which results in poor universality of assembly process modeling, further increasing the workload of data partitioning and reducing the richness of historical data sets when facing idiographic demands. In the trend of personalized manufacturing, repetitive production will be gradually replaced by variety of customized tasks (Cohen et al., 2019), highlights the urgent need for the modeling approach that can describe and abstract multiple types of assembly operations within limits. In this context, the first step of assembly process information modeling is to analyze the possible actions and classify the assembly behaviors. Similar behaviors should be semantically unified into a standard data form to serve potential and congeneric assembly operations.

(2) *The assembly process is documented by both geometric and non-geometric data, with its specific descriptive approach respectively, which is hard for a unified information modeling.* Most of the non-geometric data can be translated into the form of parameter sets by semantic means (Zhuang et al., 2021), while the geometric topological relationships are difficult to be directly parameterized. In assembly, geometric topological relationships are represented as the internal structure of parts and the actual performer of constraints (Chen et al., 2020). Previous semantic modeling of geometric information in products mostly relies on the organization of structures and relationships

(Ramos, 2015), ignoring the topological characteristics of specific geometric elements and constraints, leading to the deficiency of structural description in the semantic modeling of them.

To meet the data requirements of intelligent services in assembly, the foremost task of assembly process semantic modeling is to establish a generic information model to uniformly describe both non-geometric and geometric information, including the topological characteristics of geometric elements and assembly constraints involved. Meanwhile, the structural characteristics of geometric elements and assembly constraints can be abstracted with the consideration of overall structures of products, instead of recording by separate entities, which increases the granularity of information to some extent (Yao et al., 2013).

With the deepening of deep learning applied on graphs and networks, graph-embedding (Cai et al., 2018) provides new possibilities for the information extraction of assembly topology network to form semantic descriptions for the structural features of elements. Therefore, there lies great potential to expand the coverage of assembly process semantic modeling, providing more comprehensive data support for decision-making by artificial intelligence in assembly, which however, has been little discussed to-date.

In this manuscript, a unified parameterization method for geometric elements and assembly constraints in assembly is proposed to fill the gaps. First, the achievements of assembly information modeling and the applications of graph-embedding technology in manufacturing field are reviewed, after which the research gaps are given. Then, the assembly structure is considered as the network with the geometric elements as nodes and relationships as edges after the assembly process information been expressed with a hierarchical framework. The assembly topological relationships between geometric elements can be abstracted as the indefinite dimension adjacent matrix. On this basis, node2vec can be used to generate fixed dimensional vectors for the nodes and edges in the assembly topological network to describe the local structural features, which is considered as the key contribution of the presented research. As one of the application ideas, the approach of assembly workstep execution time prediction is shown to introduce the usage of node2vec-based parameterized representation of assembly constrains in assembly process information modeling and intelligent services. Finally, a case is illustrated to validate the rationality and feasibility of the method proposed, in which the process of algorithm tuning and training in specific scenarios can be found.

The remainder of this work is organized as follows: Section 2 reviews the related work about assembly information modeling and graph-embedding. Section 3 analyses the assembly process modeling and topological relations of assembly from the angle of geometric elements. On this basis, Section 4 introduces the parameterized expression method of geometric elements and assembly constraints. Section 5 shows the process and method of assembly workstep execution time prediction as an application scenario of parameterized representation of assembly constraints. In addition, a case study of the specific product is provided in Section 6. Finally, Section 7 presents the conclusions and discusses future work.

2 Literature Review

This research aims to extend the modeling of assembly process information with the parameterized geometric elements and assembly constraints. This section presents a systematic literature review of the latest development in the areas of assembly information modeling and graph-embedding approaches in manufacturing and assembly.

2.1 Assembly information modeling

With the continuous improvement of assembly digitalization and intelligence, the importance of assembly information modeling has been highlighted to provide data support for computer-aided decision making (Liu et al., 2011). Rachuri et al. (2006) analyzed the description method and data structure of assembly information. Considering the diversity and complexity of assembly information, hierarchical modeling method was proposed based on the product structures and dependencies (Lv et al., 2011; Chen et al., 2012). Furthermore, semantic methods like ontology (Kim et al., 2006) were applied for information modeling to improve machine-readability (Wang et al., 2007). Meanwhile, CAD assembly models were also utilized with the expansion of multi-source attributes to improve the construction of assembly routings (Vilmart et al., 2018) or retrieve the assembly models (Han et al., 2019). In engineering applications, existing CAD tools usually described assembly information centered on constraint relationships between parts, while the rules for data exchange were regulated by relative ISO standards like STEP (STandard for Exchange of Product Model Data) AP203 or AP233.

As an important part of assembly information, the modeling of assembly process has also attracted additional attentions. In terms of modeling languages, the modeling approaches of assembly process mainly contained object-oriented modeling (Qiao et al., 2011), database-based modeling (Wang et al., 2011), XML-based modeling (Bloomfield et al., 2012), ontology-based modeling (Bao et al., 2022) and Petri net-based modeling (Zha, 2000). Generally, the modeling of assembly process included non-geometric information describing attributes of involved resources or methods and geometric information describing assembly structure. The unified modeling of assembly process depended on whether there was a suitable method to map assembly structure. Therefore, the deconstruction and analysis of assembly structure have been important components in representing geometric information in assembly information modeling (Johansson et al., 2019), which has always attracted the attention of scholars. Mullins et al. (1998) proposed a graph-based assembly constraint description method concerning the topological structure of the geometrical elements involved in the parts. On this basis, Zhao et al. (2012) used the adjacency matrix in graph theory to express the assembly topological relationships in the product, further presented an integrated assembly model including assembly constrains equations and an assembly process structure tree. Based on the graph theory, various graphs that can describe assembly relations were defined and constructed, focusing on product functions (Zhang et al., 2005), assembly sequences (Vigano et al., 2012), constraint transformations (Jiang et al., 2012), assembly resources (Da et al., 2013), dimensional variation propagations (Yu et al., 2016), et al. In addition to graph-based assembly modeling, ontology technology has also been applied to the expression of assembly structure with the development of semantic modeling. Kim et al. (2009) presented a method of developing an ontology-based framework that can represent morphological characteristics related to assembly constraints, by which various similar constraints can be successfully differentiated in a standard and machine-interpretable manner. Furthermore, various knowledge and rules in the assembly field have been continuously refined and expressed through ontology, which expanded and standardized product ontology modeling research to develop further applications such as automatic assembly sequence planning (Qiao et al., 2018).

From the above research on assembly information modeling, the difficulty lies in the inconsistency of the description of assembly process considering the non-geometric attributes of resources or methods and geometric information with assembly structures as the core. Normally, the assembly relationships reflect in the constraints among geometric elements. The emphasis of assembly structure modeling is to describe the correlation among geometric elements whether graph or other semantic methods are used. When modeling geometric information of assembly processes, the traditional graph-based modeling mainly uses adjacency matrix and list to record the relationships among geometric elements at the mathematical level. However, adjacency matrix and list may have high dimension and dimensional diversity when describing different products, which restricts the storage and reuse of the assembly structure model to a large extent. Fortunately, the technology of graph embedding (Yan et al., 2006) is proposed to map nodes within networks into low-dimensional vectors with the research of deep learning for network structure, which provides the new possibility for parameterized representation of assembly relationships. By preset parameterized representation of topological structures of geometric elements and assembly constraints, the unified process modeling framework can be established considering both geometric and non-geometric information, which alleviates the heterogeneous problem of multi-source information in process modeling.

2.2 Graph-embedding approaches in assembly

The method of graph-embedding has been paid much attention in various fields since proposed for that vectors or vector sets generated by graph-embedding can be used to capture the structure of graphs, node-to-node relationships, and other relevant information about graphs, subgraphs, and nodes. At present, graph-embedding technology has been widely used in prediction of relationships in social or commodity networks (Yin et al., 2019), prediction of protein function and interaction in biomolecules (Yue et al., 2020), and forecasting or monitoring of abnormal events and network traffic in communication networks (Kriegel et al., 2008).

However, the application of graph-embedding technology in the assembly field is not consummate at present, while the applications of graph and network are widely used in various aspects of manufacturing and assembly, continuously receiving full attention. To reach Industry 4.0, the most urgent demand for manufacturing industry is to construct KGs (Knowledge Graphs) to realize intelligent decision with the support of AI (Artificial Intelligence) technology (Buchgeher et al., 2021). Motivated by this need, various KGs have been established to semantically model specific and often complex domains (Feilmayr et al., 2016) to support and enhance the accuracy of downstream tasks like assembly sequence planning (Hsu et al., 2011), assembly timing planning (Qian et al., 2021), assembly process generation and evaluation (Zhou et al., 2021), resource allocation (Zhou et al., 2021), et al. In addition to the semantic modeling method of ontology which is similar to graph structure, there are still some researches based on traditional graph or network theory. Since the graph or network can clearly represent the nodes and relationships, it has certain irreplaceability in areas such as hierarchical expression of assembly structure (Borenstein, 2000), possible workflow modeling of the workshop (Ahn et al., 2019), upstream and downstream modeling of supply chain (Hamta et al., 2018), process execution sequence modeling (Kuhn et al., 2021).

It can be summarized that the research and applications of the graph in the field of assembly have two furcate tendencies. Firstly, semantic methods are used to describe the objects and relationships in a macro perspective, after which the knowledge within can be defined manually or automatically and the framework can realize self-iteration. In this genre, the nodes of the graph can refer to various types of objects while the structural characteristics itself no longer show valid information. The other idea aims to model specific structures and processes in the production with graphs or networks. However, most of these modeling methods remain at the level of information modeling and do not make proper use of the structural characteristics contained in graphs or networks. Therefore, it is necessary to choose an appropriate method with normalization and expandability to represent and utilize the structural characteristics in the graph or network.

There are many specific algorithms that can achieve parameterized expression of nodes in the network, among which the representative approaches are shown in Table 1 with the enumerations of advantages and disadvantages. Each algorithm corresponds to certain appropriate scenarios, which is not the emphasis of this manuscript. Node2vec (Grover et al., 2016) is chosen for generating the parameter sets that correspond to the geometric elements and assembly constraints for the characteristics of: (1) the algorithm parameters can be used to control the searching process to favor the strategy of breadth-first search (BFS), which can preferably express the distribution of nodes' local neighbors; (2) node2vec provides a method of edge modeling with adjacent nodes, by which the parameterized expression of assembly constraints can be realized; (3) the dimensions of parameter sets can be manually defined to ensure that all kinds of assembly relationships can be modeled uniformly for further historical data learning algorithms; (4) the main drawback of node2vec is that it requires major computational expense when facing a huge network structure, but the number of nodes in common mechanical products cannot reach that magnitude.

Table 1 Representative graph embedding approaches

Category	Year	Method	Advantages	Defects
Factorization	2000	Locally Linear Embedding (LLE) (Roweis et al., 2000)	Available on locally linear low dimensional manifolds of any dimension; low computational complexity	Parameter sensitivity; high sample requirements
	2013	Graph Factorization (GF) (Ahmed et al., 2013)	Good adaptability for new nodes; adaptive on large scale graph data	Limited model expression; ignorance of node characteristics
	2015	GraRep (Cao et al., 2015)	Available on capturing neighbor connections of any order	Unavailable on large scale graph data; ignorance of node characteristics
	2016	High-Order Proximity preserved Embedding (HOPE) (Ou et al., 2016)	Good commonality; good training results	Unavailable on large scale graph data; ignorance of node characteristics
Random Walk	2014	DeepWalk (Perozzi et al., 2014)	Strong scalability; good adaptability for new nodes	High computational cost; ignorance of node characteristics
	2016	Node2vec (Grover et al., 2016)	Strong scalability; good adaptability for new nodes; available on adjusting migration strategy	High computational cost; ignorance of node characteristics
	2017	Struc2vec (Ribeiro et al., 2017)	Strong scalability; good adaptability for new nodes; strong ability to capture structural similarity	High computational cost; ignorance of node characteristics
Deep Learning	2016	Structural deep network embedding (SDNE) (Wang et al., 2016)	Available on retaining both local and global network structure;	Parameter sensitivity; high sample requirements; high computational cost
	2016	Deep neural networks for learning graph	Available on capturing potentially complex nonlinear	Parameter sensitivity; high sample requirements; high

		representations (DNGR) (Cao et al., 2016)	relationships between nodes	computational cost
	2017	Graph convolutional networks (GCN) (Marcheggiani et al., 2017)	Available on retaining both local and global network structure; better effect when considering node attributes	Parameter sensitivity; high sample requirements; high computational cost
Miscellaneous	2015	Large-scale Information Network Embedding (LINE) (Tang et al., 2015)	Strong scalability; Available on directed weighted graph	Ignorance of global network structure

2.3 Research gaps

In summary, the literature above has explored the methods of assembly information modeling and applications of graph-embedding in assembly. However, there are still some critical challenges in the modeling and utilization of assembly processes, as follows:

(1) *There are limitations in the unified modeling process of assembly constraints.* The mathematical modeling methods describing geometric topological characteristics of assembly constraints mainly rely on adjacency matrix and list at present. However, adjacency matrix and list have inherent shortcomings of inconsistent data dimensions when describing different objects.

(2) *There is a lack of assembly process-oriented mathematical modeling framework that can consider both geometric and non-geometric information.* Due to heterogeneity, geometric and non-geometric information are usually recorded in different forms on various carries, which affects the integral modeling and analysis of assembly process.

(3) *The preservation and utilization of assembly historical data are incomprehensive.* Based on historical data and knowledge extracted, the current approaches in the field of assembly mainly focus on specific processes or operations. The research on workshop-level production is still in its preliminary stage, which is insufficient compared with the conceive of delicacy management in the workshop.

Based on the foregoing observation, it is obvious that there are still some deficiencies in the semantic modeling of assembly process oriented to information storage and utilization. The semantic expressions of geometric information in assembly processes of multiple products differ greatly, leading to the need to build multifarious models for different products when using historical data for analysis and prediction. The commonalities of similar structures in different products are not fully utilized. Motivated by this need, this manuscript proposes a generic node2vec-based parameterized representation method of geometric elements and assembly constraints in assembly process modeling. With the method proposed, a unified assembly process modeling framework is established facing various processes of multiple products. By considering both geometric and non-geometric information in semantic modeling, assembly process can be better modeled mathematically, providing data support for assembly oriented intelligent services from the aspect of process. Meanwhile, an approach of assembly workstep execution time prediction is illustrated as the feasible applications of the method presented to demonstrate the directions in assembly workstep management.

3 Modeling of assembly process

Generally, assembly process is formulated by process designers which can expands in order of assembly operations in production to guide or stipulate assembly tasks to be executed according to predetermined trajectories. The assembly processes corresponding to specific tasks consist of sublevel units like procedures and worksteps to increase the readability and executability. The hierarchical relationships and attributes of assembly tasks, procedures and worksteps are shown in Fig. 1.

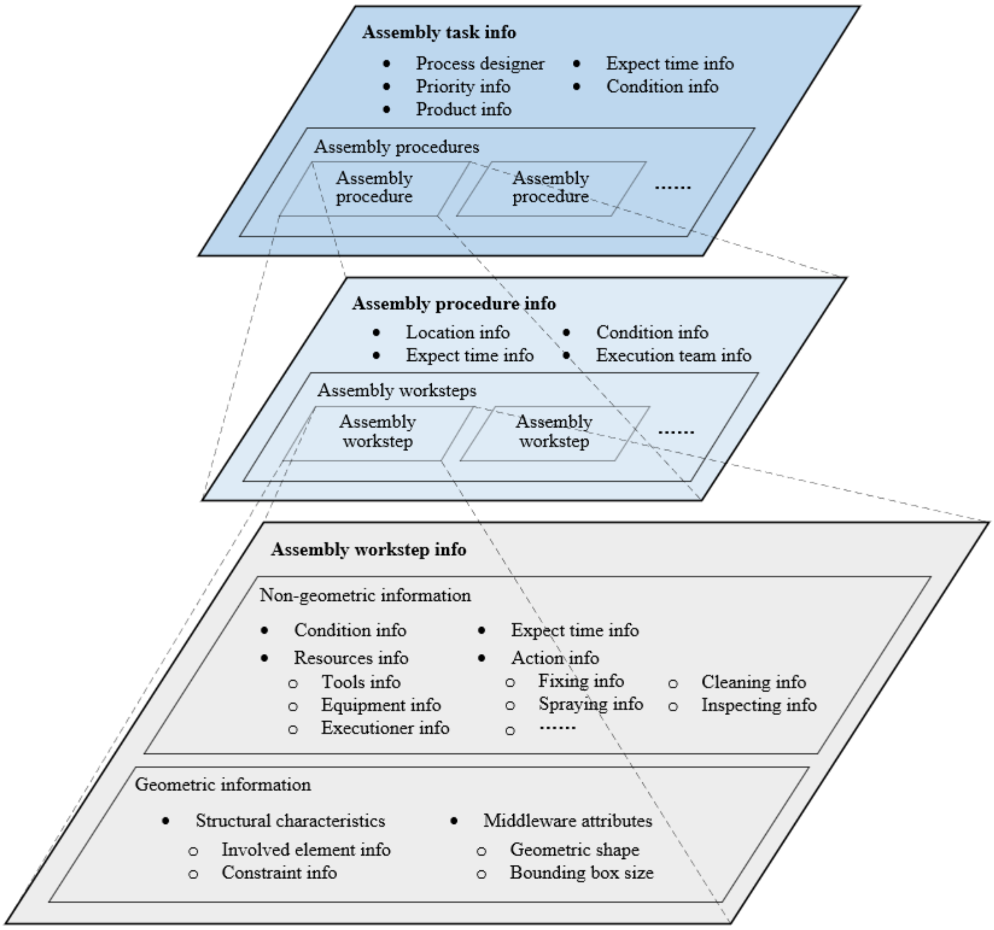
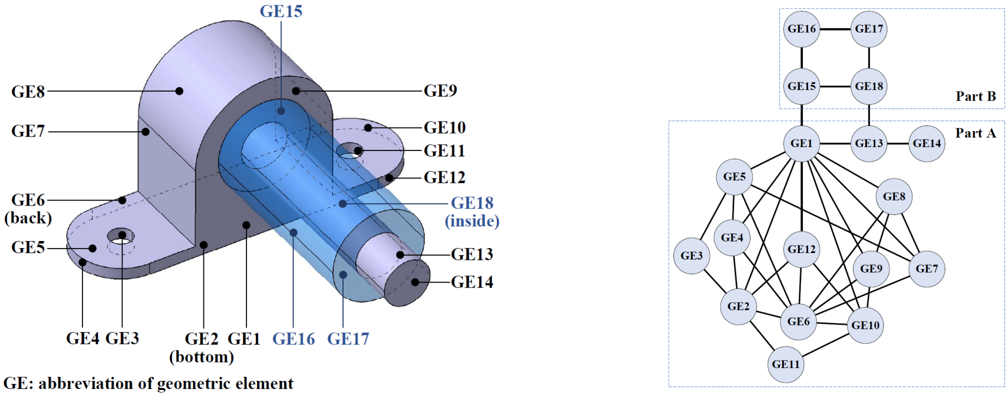


Fig. 1 The hierarchical relationships and attributes of assembly tasks, procedures and worksteps

As the primary description level of assembly process, some macro information should be considered in the modeling of assembly tasks like process designer, task priority, corresponding product, time arrangement, and task condition. Meanwhile, the subordinate relationships between the assembly task and contained procedures should be established to form the hierarchical structure. Assembly worksteps, as the basic modeling units of assembly process, undertake the detailed description of the assembly execution. Thereinto, the non-geometric information should contain involved resources and actions apart from condition and arrangement. Resources like tools, equipment and executioner applied in production should be linked with the assembly workstep, while the parameters of physical operation method should be recorded considering the diversity of operation types. On the other hand, the geometric information mainly describes the topology characteristics of geometric elements and assembly constraints involved in the assembly workstep, as well as the geometric properties of middleware that participate in.

Different from non-geometric information, the recording and preservation of geometric information highly relies on 3D assembly models, especially the description of involved geometric elements and constraints. Non-geometric information can be standardized by recording or processing according to predefined data forms as parameter sets, while structural characteristics need further treatment to form parameter sets that can describe geometric elements and topological relationships. In most assembly worksteps, one or more assembly constraints are implemented on physical entities. Parameterized modeling of assembly constraint information is indispensable for the purpose of summarizing the knowledge or rules from the historical data of assembly workstep execution.

The topological relationships among geometric elements in products mainly include two portions: the adjacent relations among geometrical elements inside parts and the assembly constraints between parts. Firstly, if geometric elements and relationships are presented in the form of graph, the adjacency graph in Fig. 2(b) describes the structure of the parts shown in Fig. 2(a).



(a) The example parts (b) The adjacency graph

Fig. 2 Example parts and the adjacency graph describing geometrical structure

With geometric elements as nodes, the assembly topological relationships can be expressed by an unweighted undirected graph. The edges in the graph represent the adjacent relations of geometric elements within parts and assembly constraints between parts. Meanwhile, the adjacency matrix can also reflect the topological relationships between geometric elements. For the sample parts in Fig. 2, the adjacency matrix can be denoted as shown in Fig. 3.

	GE1	GE2	GE3	⋯	GE16	GE17	GE18
GE1	0	1	0	⋯	0	0	0
GE2	1	0	1	⋯	0	0	0
GE3	0	1	0	⋯	0	0	0
⋮	⋮	⋮	⋮	⋱	⋮	⋮	⋮
GE16	0	0	0	⋯	0	1	0
GE17	0	0	0	⋯	1	0	1
GE18	0	0	0	⋯	0	1	0

Fig. 3 The adjacency matrix of example parts

The dimension of the adjacency matrix depends on the quantity of geometric elements in the product. The dimensional uncertainty of adjacency matrix prevents it from being directly used in the machine learning process of assembly history data as the carrier of assembly topological relationships. Therefore, an appropriate method is needed to model the assembly relationships of arbitrary scale with a parameter set of fixed length.

When node2vec is used for network learning, the network structure needs to be transformed into the form of adjacence list. The steps of construction and transformation are as follows:

Step 1. The geometric elements of the products should be individually assigned an identification number. Usually, this step needs to be completed in CAD software with the design models of the products. It is necessary to note that geometric elements on different products should be differentiated with the same rules.

Step 2. The adjacencies between geometric elements within parts should be traversed and recorded. Based on the geometric elements and topological relations inside the parts, part-level unauthorized undirected networks can be constructed as subnetworks of the product level network.

Step 3. Similarly, the assembly constraints in the products should be traversed and recorded, especially the geometric elements that involved in the constraints. With constraints, the part-level networks can be connected and form the product-level topology networks.

Step 4. For various types of products that can be analyzed, the topological networks can be placed in the same environment, constituting an aggregate network with the widest possible coverage. With the increase of the number of product-level sub-networks, the parameterized representation trained by the network will better represent the topological characteristics of geometric elements.

Step 5. The network referring to topological structures of products should be transformed into groups of geometric elements with topological relationships to meet node2vec's demand for data form.

With the assembly process modeling, the information related to the process can be consistently expressed in a hierarchical information framework, including assembly task information, assembly procedure information and assembly workstep information. In order to improve the versatility of the assembly process model, structural characteristics of geometric elements and assembly constraints are envisaged to be expressed by preset length vectors. This section illustrates the extraction and abstraction of assembly topological relationships, while the parameterization method will be proposed in the next section.

4 Parameterized representation of assembly structure

Parameterized representation of assembly structure needs to deal with geometric information during assembly, such as the structure and distribution of geometric elements, which cannot be evaluated by some measurable magnitudes. With the development of graph embedding, the algorithm of node2vec provides an effective solution to this problem.

As mentioned above, the assembly topology can be abstracted to a network if the geometric elements are regarded as nodes and the relations as edges. Node2vec is a model that outputs node vectors in a network, which can be used for fixed dimensional parameterization of geometric elements. The main idea of node2vec is to generating corresponding sequences for each node by sampling with a specific way of walking, and then drawing on the CBOW (Continuous Bag-Of-Words) or Skip-gram model in word2vec (Mikolov et al., 2013) with regarding the sequences as texts to get the vector of each node.

The advantage of node2vec is that it adopts a biased random walk strategy when dealing with assembly networks. For a network *G (V, E)*, given the current node *v*, the probability of walking to the next node *x* is:

$$P(c_i = x \mid c_{i-1} = v) = \begin{cases} \pi_{vx} / Z & \text{if } (v, x) \in E \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where π_{vx} is the unnormalized transition probability from node v to node x, while Z is normalization constant. The edges in the assembly network do not contain weights, therefore $w_{vx}=1$. If the previous node of random walk is node t, $\pi_{vx}=\alpha_{pq}(t,x)\cdot w_{vx}=\alpha_{pq}(t,x)$, in which:

$$\alpha_{pq}(t,x)=\begin{cases} 1/p & \text{if } d_{tx}=0 \\ 1 & \text{if } d_{tx}=1 \\ 1/q & \text{if } d_{tx}=2 \end{cases} \tag{2}$$

where d_{tx} is the shortest path distance between node t and node x. Node2vec introduces two hyper-parameters p and q to control the random walk strategy. Return parameter p controls the probability of repeating a node that has just been accessed. With the higher value of p, there is a low probability of repeated access. In-out parameter q controls whether the walk is outward or inward. If q>1, the random walk tends to visit nodes that are close to node t. Otherwise, the random walk tends to visit nodes far from node t. The specific walking probability is shown in Fig. 4 (Grover et al., 2016).

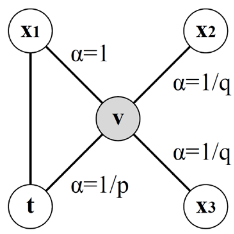


Fig. 4 Illustration of the random walk strategy in node2vec.

According to the above analysis, the random walk strategy of node2vec can be similar to the breadth-first search (Jungnickel, 2005) when selecting the appropriate value of the hyper-parameters (p<1, q>1). This controllable walk strategy enables node2vec to effectively model the adjacent topological structures of geometric elements when generating node vectors, which is the primary concern of geometric element parameterized representation.

For a specific network structure, the edge vector can be further calculated after the generation of node vectors. Given two nodes u and v with the node vectors $f(u)$ and $f(v)$, there are four choices given by Grover (2016) for the calculation of edge vectors $V(u, v)$, as shown in Table 2.

Table 2 Choices of edge vector calculation.

Method	Definition
Average	$V_i(u,v)=\frac{f_i(u)+f_i(v)}{2}$
Hadamard	$V_i(u,v)=f_i(u)*f_i(v)$
Weighted-L1	$V_i(u,v)=\lvert f_i(u)-f_i(v)\rvert$
Weighted-L2	$V_i(u,v)=\lvert f_i(u)-f_i(v)\rvert^2$

For the assembly structure, the parameterized expression of the correlation between geometric elements can be obtained, including the internal geometric topology and the assembly constraints between parts. By this means, vectors of predetermined length can be used to refer to geometric elements and topological relations in the assembly instead of traditional adjacencies graph and matrix. The problem that different products cannot be unified in modeling due to dimensional inconsistencies can be solved to a certain extent. In graph embedding, the structural characters of the geometric elements in the adjacent neighborhood are recorded by preset length vectors. Along with other attributes, such as geometry type, bounding box size, et al., it is possible to achieve a normalized parametric description of geometric elements. In a similar way, topological relations in assembly can also be expressed by a parameter set of rated length. From this point of view, the parameterization of geometric elements and assembly constraints can be considered as a practice of granular computing in assembly process modeling, for that heterogeneous information containing geometric and non-geometric information is unified in the same dimension. With parameterization, the structural characteristics of geometric elements and assembly constraints can be abstracted from the entire assembly topology instead of modeling as separate entities. The modeling of multiple production process of various products can be mathematically unified considering overall structure to form a more comprehensive and general database in mining knowledge or discovering patterns. Parameterization of geometric elements and relationships can be applied in various aspects, such as:

1) Parameterized representation of assembly constraints can be used as the substitution of geometric information to participate in assembly process modeling. Traditional assembly process modeling often relies on 2D drawings, 3Dmodels and simulation animations to carry assembly relations. Such graphical description methods can effectively guide the assembly personnel to operate or write equipment automation control instructions, but it is difficult to accept and understand for computers. The vector-based parametric representation can describe the assembly constraints with fixed length vectors in the context of recording the topological characteristics. Parameterized assembly constraints provide a solution for assembly process modeling regardless of product type, offering a new idea for the construction and utilization of assembly process big data.
2) The representation of nodes and edges can be used for link prediction with appropriate algorithms. Under the trend of collaborative manufacturing, complex products are usually divided into multiple assembly units (Wang et al., 2009) to be designed and manufactured by different enterprises. The rationality of product modularization largely determines the performance of the product (Li et al., 2019). Node2vec-based graph embedding can predict the existence of edges by parameterizing objects in the network. With this specialty, guidance on assembly unit partitioning can be given at the product design stage based on statistical methods if numerous samples of partitioning results are used for training.
3) In the stage of part design and machining, parameterized representation can be used for the classification of geometric elements, and further provides an idea of feature recognition to avoid repeatedly traversing the geometric topological network. By parameterized representation of nodes and edges in the network, machining features in parts can be separated with the additional information of adjacent geometric structures to further refine the types. With the promotion and improvement of STEP-NC technology, the compilation and implementation of machining process have been gradually transformed to be based on the machining features of parts (Xu, 2006). The classification and enrichment of description will lay the foundation for the formation and utilization of process knowledge based on machining features.

5 Assembly workstep execution time prediction

In order to meet the new requirements for smart manufacturing, the assembly process should be pre-analyzed and predicted in specific steps and aspects. Although the introduction of automation equipment and human-robot collaborative assembly have reduced the uncertainty in process, manual assembly is still essential especially for complex products (Fang et al., 2020). The traditional assembly scheduling depends more on experience in determining the execution time, which is inadequate when facing personalized production tasks. The prediction of assembly workstep execution time can promote the rationality of production scheduling to a large extent, furthermore improve production efficiency.

5.1 Modeling of the influencing factors of workstep execution

As analyzed in the modeling of assembly process, there may be multiple execution actions in the assembly workstep, such as cleaning, inspecting, fixing, spraying, et al. When diverse actions are performed, the factors affecting the execution of assembly worksteps are also different. In this manuscript, fixing operation is chosen to demonstrate how parameterized assembly constraints support the prediction of assembly workstep execution time.

The sub-assemblies and parts involved in assembly are collectively referred to as assembly middleware. The factors affecting the execution time of fixing operation mainly include modalities of assembly constraints, usage of assembly resources, properties of assembly middleware, fastening parameters, as shown in Fig. 5.

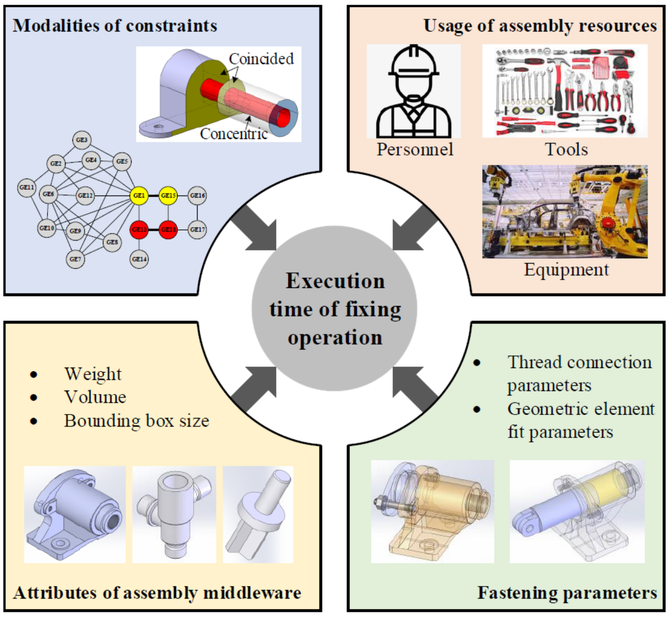


Fig. 5 Analysis of the factors that may affect the execution time of fixing operation.

Among the factors, adjacent structures of assembly constraints can be parameterized by learning topological network, which is often ignored in the analysis of assembly process. The complex neighbor structure of assembly constraints will inevitably affect the execution of assembly operations. Meanwhile, the parameterization of assembly constraints makes the modeling of assembly operation more comprehensive with the consideration of geometric information. In addition, discrepant manufacturing capacities lead to the execution of workstep being affected by assembly resources such as personnel, tools and equipment. The attributes of the assembly middleware also have an impact on the execution time of assembly operation. Particularly, the size of middleware bounding box is critical for its decisive role of equipment and personnel movement. At present, the fixing of mechanical products mostly depends on the thread connection to achieve, with the fits between geometric elements for auxiliary positioning. Therefore, it is necessary to record the fastening parameters during the middleware installation, such as the nominal diameter of the thread, the axial length of the thread, the number of threaded connections, et al.

The non-geometric information, such as involved assembly resources and fastening parameters, can be obtained from assembly process documents or assembly MES (Manufacturing Execution System) with detailed description of the method and process of assembly. Meanwhile, the geometric information is usually carried by 3D models. Attributes of assembly middleware can be directly acquired by the build-in functions in designing tools, while topological structures can be obtained by abstracting the geometric elements, adjacent relations and assembly constraints contained in the 3D models. After parameterization, the structural characteristics can be expressed by preset length vectors, forming semantic data modeling along with other attributes related to the execution of worksteps.

5.2 Workstep execution time prediction based on historical data

With information modeling of factors that may affect the execution time of assembly workstep, historical data can be processed as the training set to match the requirements of machine learning models. A set of data in the training set should simultaneously contain parameterized assembly constraints, usage of assembly resources, attributes of assembly middleware and fastening parameters, as shown in Fig. 6.

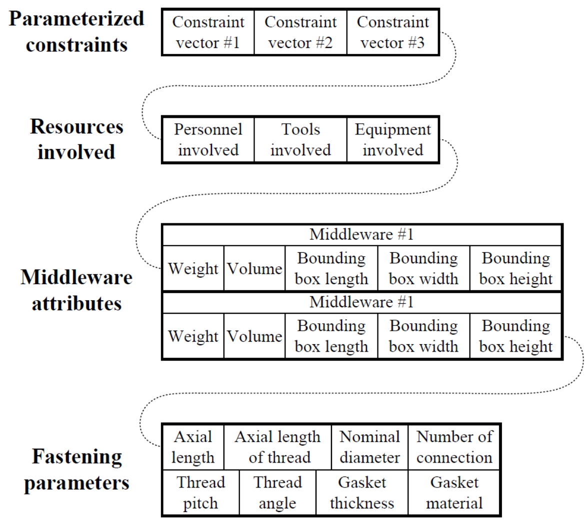


Fig. 6 Characteristic variables of assembly time prediction data set.

It is well-known that the degree of freedom between two assembly middleware can be limited by up to three appropriate assembly constrains. Therefore, the space for three constraint vectors is reserved for the parameterized constraints in the prediction data set. The dimension of the constraint vector depends on the preset parameter of node2vec-based network learning, the appropriate value of which should be determined through experiment. The main process of workstep execution time prediction model establishment is shown in Fig. 7.

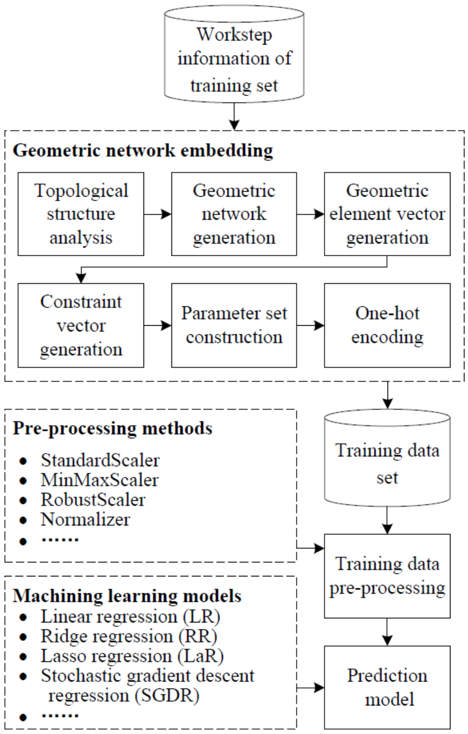


Fig. 7 The establishment of workstep execution time prediction model.

Firstly, the parameter set of each assembly workstep sample is constructed combined with the constraint parameterized representation of geometric topology structure and non-geometric information that may affect the execution time. Secondly, the parameter sets should be processed by one-hot encoding to allow the discrete variables participating in the regression task. After the mathematical preparation, a suitable machine learning model should be selected and trained with the appropriate pre-processing methods to fit the relationship between the assembly workstep execution time and the influencing parameters. The parameter sets from new assembly tasks can be processed in the same way to predict the workstep execution time with the trained machine learning model.

In the traditional assembly mode, the execution time of task scheduling is mostly estimated based on experience, which is different from the actual situation in most cases. With the growing popularity of customized production, the inaccuracy will be amplified because of the limitations of human experience. The inconsistencies between the scheduling and execution will lead to the decrease in the utilization of production resources and the decline in productively. To solve this problem, delicacy management is introduced into the manufacturing field, aiming to control the production from the segment as detailed as possible. If the number of assembly historical data entries is sufficient, the method proposed in this manuscript can predict the execution time of any assembly workstep before practical production, providing suggestions of assembly workstep level for assembly plan formulation and real-time scheduling during assembly execution.

6 A case study

In this section, a case study is illustrated to verify the validity of the method proposed. The assembly process of the sample product shown in Fig. 8 and geometrical topology structure (as shown in Fig. 9) are analyzed, after which parameterized representation of assembly constraints and construction of workstep time prediction model are realized.

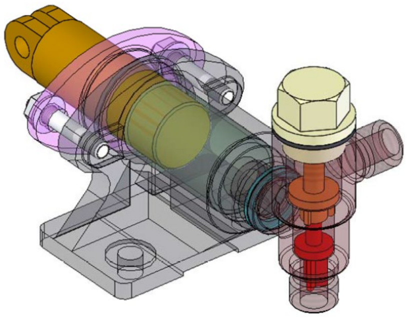


Fig. 8 The sample product

The sample product consists of 8 parts, subdivided into 104 geometric elements. The topological relationships among geometric elements can be obtained through secondary development of CAD software, and the network structure can be formed based on the adjacent relationships and assembly constraints abstracted from 3D models. In the generation of node vectors, there are five primary parameters: number of walks, walk length, hyper-parameter p, hyper-parameter q, and embedding size. Number of walks and walk length represents the number of random steps generated by each node and the number of nodes passed by each random walk, controlling the iteration of the random walk. Hyper-parameters p and q restrict the moving strategy of the random walk, as discussed in Section 4. Embedding size refers to the dimension of the generated node vectors. In order to evaluate the modeling of node vectors on the structural characteristics, geometric elements are divided into two categories according to whether they are involved in assembly constraints. Furthermore, the node vector-based classification accuracy with the kernel of linear regression is used for the evaluation of embedding. Since there is uncertainty in the random walk, all the evaluation results in this manuscript take the average of 100 tests under the same parameter setting. The classification accuracy (Micro-F1 score) is 68.29% with a set of predefined parameters (number of walks=80, walk length=10, p=0.25, q=4, embedding size=20) to classify geometric elements. With the help of t-SNE (t-distributed Stochastic Neighbor Embedding) (Van et al., 2008), vectors of geometric elements can be projected onto two-dimensional space, with the distribution as shown in Fig. 10. It can be clearly seen that the geometric elements related to assembly constraints have obvious aggregation on the diagram, demonstrating that the geometric element representations based on node2vec can reflect the topological structure to some extent.

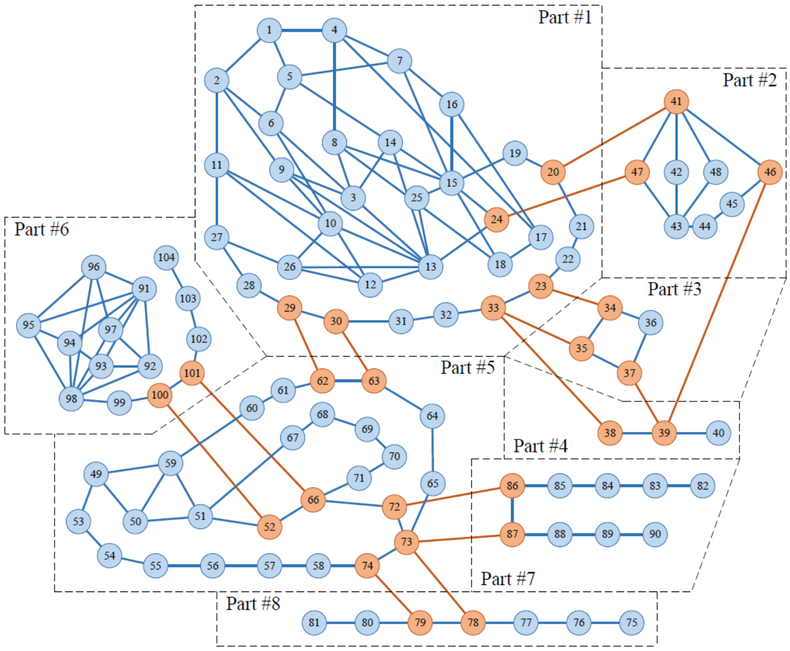


Fig. 9 The geometrical network of sample product

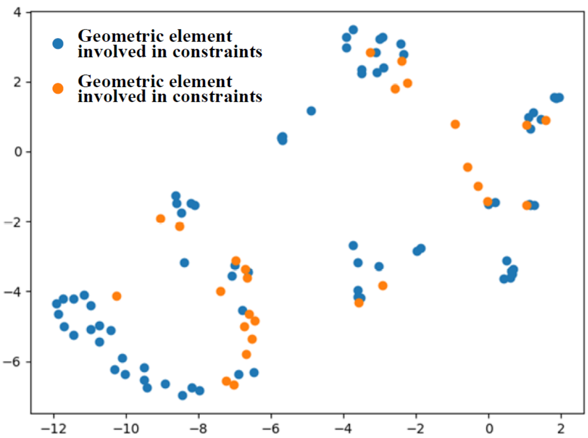
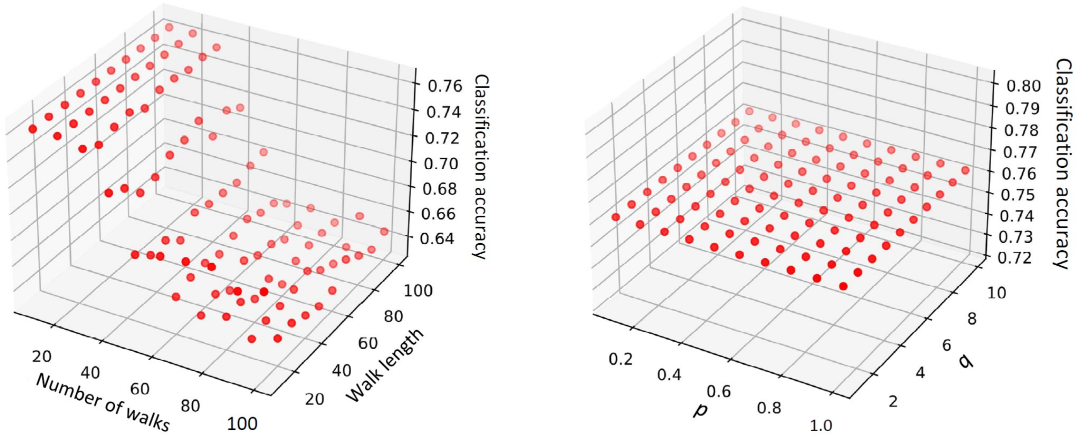


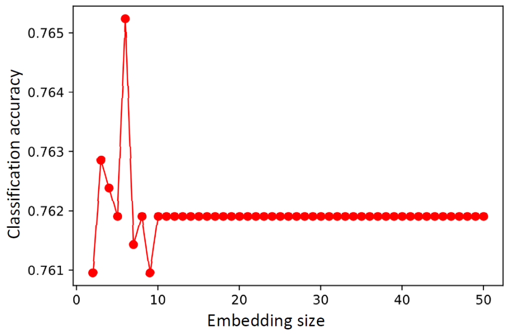
Fig. 10 The distribution of geometric elements of sample product (number of walks=80, walk length=10, p=0.25, q=4, embedding size=20).

However, the parameters of the embedding may affect the expression effect of node vectors on geometric topological relationships. Therefore, the optimal parameter set should be determined by repeated experiments. According to the actual control link, the approximation of the optimal parameters is divided into three steps. Firstly, number of walks and walk length, as the controller of the random walk iteration, are altered on the scale of 1 to 100 to search for the best combination. The result is shown in Fig. 11(a). Secondly, the hyper-parameters p and q can also be altered on the scale of 0 to 1 and 1 to 10, respectively. The results of the combinations of various parameter values are shown in Fig. 11(b). Similarly, the classification accuracy under different preset embedding sizes is obtained and shown in Fig. 11(c). Through the above experiments, it can be determined that for the network in the case study, the optimal parameter of node2vec algorithm is set as follows: number of walks=20, walk length=70, p=0.5, q=4, embedding size=6. This set of parameters will be used in subsequent node2vec-related calculations. It should be noted that the above parameter sets are only valid for this data set. For different training sets, a unique parameter set should be set to ensure the optimal algorithm effect.



(a) The searching results of number of walks and walk length

(b) The searching results of hyper-parameters p and q



(c) The searching results of embedding size

Fig. 11 Parameter searching results of number of walks & walk length, hyper-parameters p and q, and embedding size

After the parameter settings are determined, the node2vec algorithm can be used to obtain the geometric element vectors. The relationships between geometric elements, which are the edges in the network, can also be divided into two categories: the adjacent relationships within parts and the assembly constraints between parts. Similarly, the effect of edge parameterization can be evaluated by the classification accuracy of the relationships between geometric elements. Since the geometric element vectors are settled, four methods as shown in Table 2 are used to calculate the edge vectors respectively. The classification accuracy of the edge vectors obtained by all four method is 87.10%. Therefore, the average method with the least amount of computation is chosen for the parameterized expression of assembly constraints.

Based on the parameterized representation of assembly constraints, the prediction data set of assembly workstep execution time can be constructed together with other information contained in the workstep. The assembly process of the sample product consists of 7 main assembly worksteps. As an example, the modeling process and result of the third workstep are shown in Fig. 12. Hierarchical assembly process modeling is illustrated with the separate information constitution of the assembly task, procedure and workstep. When forming the unified prediction data set, various factors that may affect the execution time of the sample assembly workstep are processed by appropriate methods. Thereinto, structural characteristics of assembly constraints can be parameterized by node2vec as introduced. Resources involved should be preprocessed by one-hot encoding for the discreteness. Middleware attributes and fastening parameters should be semantic processed according to uniform data arrangement and measure units, while the material properties of fasteners should be preprocessed by one-hot encoding likewise.

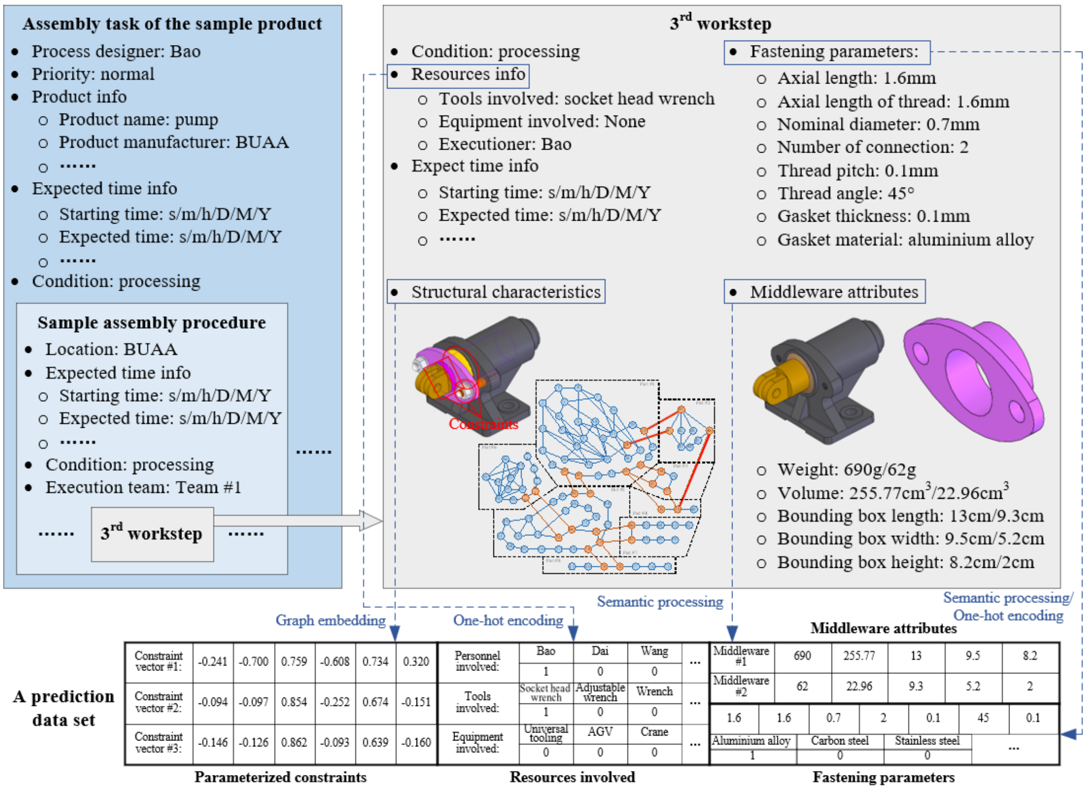


Fig. 12 The modeling process and result of the sample workstep

In a similar way, 266 sets of prediction data are collected for training models under different conditions such as personnel, tools and equipment in the experiment, together with the measured execution time. Five common regression models are trained using the training data sets, and the corresponding pre-processing methods and prediction accuracy obtained by 10-fold cross-validation are shown in Table 3. Besides, the prediction accuracy of the same training sets after removing the parameterized constraints is also given as a contrast under uniform machining learning models and pre-processing methods.

Table 3 Prediction accuracy of assembly workstep execution time based on multiple machine learning models.

Machining learning models	Pre-processing methods	Prediction accuracy with parameterized constraints	Prediction accuracy without parameterized constraints
Linear regression	-	77.35%	72.28%
Ridge	Standard scaler	75.53%	69.70%
Lasso	Standard scaler	74.77%	69.76%
Stochastic Gradient Descent with Warm Restarts (SGDR)	Standard scaler	78.37%	73.82%
Support Vector Machine (SVM) with linear kernel	-	77.45%	72.46%

With the comparison above, prediction accuracy with parameterized constraints can be improved to some extent. The improvement of accuracy proves that there is certain influence in the execution of assembly operation considering the complexity of assembly positions, which can be parameterized with the method proposed. The parameterized assembly constraints can reflect the geometric topology information and participate in the modeling of assembly worksteps, providing support for the subsequent analysis of assembly big data.

7 Conclusions and future works

With the popularization of artificial intelligence algorithm and the continuous improvement of high-level automation production systems, the generality of process information semantic modeling needs to be emphasized to anticipate the trend of customized production. The topological characteristics of geometric elements and assembly constraints, as the geometric information which is difficult to describe uniformly, are the challenging objects in assembly process modeling. In this manuscript, a method of generic node2vec-based parameterized representation of assembly process information is presented with the sample application of assembly workstep execution time prediction. With the method proposed, the semantic modeling of assembly processes can be jointed with the semantic representation of the topological characteristics of elements involved. Meanwhile, the historical data of assembly process can break through the original mode of storage and utilization, which is being classified according to specific products and operations. The vectorial expression of geometric elements and assembly constraints expand the application possibility of assembly process big data, which can be used for intelligent services in personalized production mode. The main contributions of this research are highlighted as follows:

(1) *The organization form of assembly process information is defined considering geometric and non-geometric information.* The information contained in the assembly process is divided according to hierarchy while the data form and specific steps of the construction and transformation of the geometric information that meets the requirements of node2vec are presented on the basis of analyzing and deconstructing CAD models to provide mathematical preparation for the algorithm of graph embedding.

(2) *A generic node2vec-based parameterized representation method of geometric elements and topological relationships in assembly is proposed.* The vector with preset length can be used to reflect the structural characteristics of geometric elements or constraints, providing a highly generalized geometric information modeling method for assembly process. The adjustment and optimization of the parameters of node2vec are illustrated considering the applied scenario.

(3) *The execution time of assembly workstep is predicted based on historical data.* As the sample application of constraint parameterized representation, this approach shows the applicability and effectiveness of node2vec. Meanwhile, a new path of workshop delicacy management is introduced with the unified modeling of both geometric and non-geometric information of products.

Compared with other links in manufacturing, the modeling, storage, and utilization of assembly process data are difficult, due to the diversity, uncertainty, and complexity of assembly processes. This manuscript explored the parameterized representation of geometric elements and assembly constraints with graph embedding technology, and hence provided a new idea for the unified modeling of geometric and non-geometric information of assembly process.

The vectors referring to geometric elements and assembly constraints held the ability to record adjacent geometric structures instead of the topological network, which makes it possible to serve multiple assembly analysis requirements that can be further attempted. The parameterized representation of geometric elements and assembly constraints brings in new possibilities for assembly analysis in the scenes of product development and workshop management. In the future, the similarity assessment based on the parameterized representation needs to be studied, by which the historical processes can be quickly reused and the adjustment scheme in assembly can be recommended. Besides, the method of production scheduling and distribution with the support of workstep execution time prediction should be further studied to reach the prospect of delicacy management.

Declarations

Funding

This research is supported by the 2017 Special Scientific Research on Civil Aircraft Project of China.

Conflicts of interest/Competing interests

The authors declare no competing interests.

Availability of data and material

Not applicable.

Code availability

Not applicable.

Authors' contributions

Not applicable.

Ethics approval

Not applicable.

Consent to participate

Not applicable.

Consent for publication

Not applicable.

References

Ahmed, A., Shervashidze, N., Narayanamurthy, S., Josifovski, V., & Smola, A. J. (2013, May). Distributed large-scale natural graph factorization. In *Proceedings of the 22nd international conference on World Wide Web* (pp. 37-48). https://doi.org/10.1145/2488388.2488393

Ahn, K., & Park, J. (2019, December). Idle vehicle rebalancing in semiconductor fabrication using factorized graph neural network reinforcement learning. In *2019 IEEE 58th Conference on Decision and Control (CDC)* (pp. 132-138). https://doi.org/10.1109/CDC40024.2019.9030245

Bao, J., Guo, D., Li, J., & Zhang, J. (2019). The modelling and operations for the digital twin in the context of manufacturing. *Enterprise Information Systems*, 13(4), 534-556. https://doi.org/10.1080/17517575.2018.1526324

Bao, Q., Zhao, G., Yu, Y., Dai, S., & Wang, W. (2021). The Ontology-Based Modeling and Evolution of Digital Twin for Assembly Workshop. *The International Journal of Advanced Manufacturing Technology*. https://doi.org/10.1007/s00170-021-07773-1

Belhadi, A., Zkik, K., Cherrafi, A., & Sha'ri, M. Y. (2019). Understanding big data analytics for manufacturing processes: insights from literature review and multiple case studies. Computers &

Industrial Engineering, 137, 106099.
https://doi.org/10.1016/j.cie.2019.106099

Borenstein, D. (2000). A directed acyclic graph representation of routing manufacturing flexibility. *European journal of operational research*, 127(1), 78-93.
https://doi.org/10.1016/S0377-2217(99)00324-0

Buchgeher, G., Gabauer, D., Martinez-Gil, J., & Ehrlinger, L. (2021). Knowledge Graphs in Manufacturing and Production: A Systematic Literature Review. *IEEE Access*, 9, 55537-55554.
https://doi.org/10.1109/ACCESS.2021.3070395

Cai, H., Zheng, V. W., & Chang, K. C. C. (2018). A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering*, 30(9), 1616-1637.
https://doi.org/10.1109/TKDE.2018.2807452

Cao, S., Lu, W., & Xu, Q. (2015, October). Grarep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM international on conference on information and knowledge management* (pp. 891-900).
https://doi.org/10.1145/2806416.2806512

Cao, S., Lu, W., & Xu, Q. (2016, February). Deep neural networks for learning graph representations. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 30, No. 1).
https://ojs.aaai.org/index.php/AAAI/article/view/10179

Cohen, Y., Naseraldin, H., Chaudhuri, A., & Pilati, F. (2019). Assembly systems in Industry 4.0 era: a road map to understand Assembly 4.0. *The International Journal of Advanced Manufacturing Technology*, 105(9), 4037-4054.
https://doi.org/10.1007/s00170-019-04203-1

Cui, Y., Kara, S., & Chan, K. C. (2020). Manufacturing big data ecosystem: A systematic literature review. *Robotics and computer-integrated Manufacturing*, 62, 101861.
https://doi.org/10.1016/j.rcim.2019.101861

Chen, J., & Jia, X. (2020). An approach for assembly process case discovery using multimedia information source. Computers in Industry, 115, 103176.
https://doi.org/10.1016/j.compind.2019.103176

Chen, M., Herrera, F., & Hwang, K. (2018). Cognitive computing: architecture, technologies and intelligent applications. Ieee Access, 6, 19774-19783.

Da Xu, L., Wang, C., Bi, Z., & Yu, J. (2013). Object-oriented templates for automated assembly planning of complex products. *IEEE Transactions on Automation Science and Engineering*, 11(2), 492-503.
https://doi.org/10.1109/TASE.2012.2232652

Feilmayr, C., & Wöß, W. (2016). An analysis of ontologies and their success factors for application to business. *Data & Knowledge Engineering*, 101, 1-23.
https://doi.org/10.1016/j.datak.2015.11.003

Fang, W., & Zheng, L. (2020). Shop floor data-driven spatial–temporal verification for manual assembly planning. *Journal of Intelligent Manufacturing*, 31(4), 1003-1018.
https://doi.org/10.1007/s10845-019-01491-y

Grover, A., & Leskovec, J. (2016, August). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 855-864).
https://doi.org/10.1145/2939672.2939754

Hamta, N., Shirazi, M. A., Behdad, S., & Ghomi, S. F. (2018). Modeling and measuring the structural complexity in assembly supply chain networks. *Journal of Intelligent Manufacturing*, 29(2), 259-275.
https://doi.org/10.1007/s10845-015-1106-9

Hsu, Y. Y., Tai, P. H., Wang, M. W., & Chen, W. C. (2011). A knowledge-based engineering system for assembly sequence planning. *The International Journal of Advanced Manufacturing Technology*, 55(5-8), 763-782.
https://doi.org/10.1007/s00170-010-3093-5

IBM (2017). Cognitive Manufacturing: An Overview and Four Applications that are Transforming Manufacturing Today.

Jiang, D., Na, G., & Ji, K. (2012). Research on assembly model based on assembly constraints transformation. *Procedia Engineering*, 29, 770-774.
https://doi.org/10.1016/j.proeng.2012.01.039

Johansson, P. E. C., Malmsköld, L., Fast-Berglund, Å., & Moestam, L. (2019). Challenges of handling assembly information in global manufacturing companies. *Journal of Manufacturing Technology Management*, 31(5), 955-976.
https://doi.org/10.1108/JMTM-05-2018-0137

Jungnickel, D. (2005). *Graphs, networks and algorithms*. Berlin: Springer.

Kim, K. Y., Chin, S., Kwon, O., & Ellis, R. D. (2009). Ontology-based modeling and integration of morphological characteristics of assembly joints for network-based collaborative assembly design. *AI EDAM*, 23(1), 71-88.
https://doi.org/10.1017/S0890060409000110

Kriegel, H. P., Kröger, P., Renz, M., & Schmidt, T. (2008, July). Hierarchical graph embedding for efficient query processing in very large traffic networks. In *International Conference on Scientific and Statistical Database Management* (pp. 150-167).
https://doi.org/10.1007/978-3-540-69497-7_12

Kuhn, M., Funk, F., Zhang, G., & Franke, J. (2021). Blockchain-based application for the traceability of complex assembly structures. *Journal of Manufacturing Systems*, 59, 617-630.
https://doi.org/10.1016/j.jmsy.2021.04.013

Liu, J., Sun, Q., Cheng, H., Liu, X. K., Ding, X. Y., Liu, S. L., & Xiong, H. (2018). The state-of-the-art, connotation and developing trends of the products assembly technology. *Journal of Mechanical Engineering*, 54(11), 2-28.
https://doi.org/ 10.3901/JME.2018.011.002

Li, Y., Wang, Z., Zhong, X., & Zou, F. (2019). Identification of influential function modules within complex products and systems based on weighted and directed complex networks. *Journal of Intelligent Manufacturing*, 30(6), 2375-2390.
https://doi.org/10.1007/s10845-018-1396-9

Lee, J., Bagheri, B., & Kao, H. A. (2015). A cyber-physical systems architecture for industry 4.0-based manufacturing systems. Manufacturing letters, 3, 18-23.

Lu, Y., Xu, X., & Wang, L. (2020). Smart manufacturing process and system automation–a critical review of the standards and envisioned scenarios. Journal of Manufacturing Systems, 56, 312-325.
https://doi.org/10.1016/j.jmsy.2020.06.010

Marcheggiani, D., & Titov, I. (2017). Encoding sentences with graph convolutional networks for semantic role labeling. *arXiv preprint arXiv:1703.04826.*

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781.*

Mullins, S. H., & Anderson, D. C. (1998). Automatic identification of geometric constraints in mechanical assemblies. *Computer-Aided Design*, 30(9), 715-726. https://doi.org/10.1016/S0010-4485(98)00026-8

Ou, M., Cui, P., Pei, J., Zhang, Z., & Zhu, W. (2016, August). Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1105-1114). https://doi.org/10.1145/2939672.2939751

Perozzi, B., Al-Rfou, R., & Skiena, S. (2014, August). Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 701-710). https://doi.org/10.1145/2623330.2623732

Qian, J., Zhang, Z., Shi, L., & Song, D. (2021). An assembly timing planning method based on knowledge and mixed integer linear programming. *Journal of Intelligent Manufacturing*. Advance online publication. https://doi.org/10.1007/s10845-021-01819-7

Qiao, L., Qie, Y., Zhu, Z., Zhu, Y., uz Zaman, U. K., & Anwer, N. (2018). An ontology-based modelling and reasoning framework for assembly sequence planning. *The International Journal of Advanced Manufacturing Technology*, 94(9), 4187-4197. https://doi.org/10.1007/s00170-017-1077-4

Ramos, L. (2015). Semantic Web for manufacturing, trends and open issues: Toward a state of the art. Computers & Industrial Engineering, 90, 444-460. https://doi.org/10.1016/j.cie.2015.10.013

Ribeiro, L. F., Saverese, P. H., & Figueiredo, D. R. (2017, August). struc2vec: Learning node representations from structural identity. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 385-394). https://doi.org/10.1145/3097983.3098061

Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500), 2323-2326. https://doi.org/10.1126/science.290.5500.2323

Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015, May). Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web* (pp. 1067-1077). https://doi.org/10.1145/2736277.2741093

Terziyan, V., Gryshko, S., & Golovianko, M. (2018). Patented intelligence: Cloning human decision models for Industry 4.0. Journal of Manufacturing Systems, 48, 204-217.

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Vigano, R., & Gómez, G. O. (2012). Assembly planning with automated retrieval of assembly sequences from CAD model information. Assembly Automation, 32(4), 347-360. https://doi.org/10.1108/01445151211262410

Wang, C., & Jiang, P. (2019). Deep neural networks based order completion time prediction by using real-time job shop RFID data. *Journal of Intelligent Manufacturing*, 30(3), 1303-1318.

https://doi.org/10.1007/s10845-017-1325-3

Xu, X. W. (2006). Realization of STEP-NC enabled machining. *Robotics and Computer-Integrated Manufacturing*, 22(2), 144-153. https://doi.org/10.1016/j.rcim.2005.02.009

Yan, S., Xu, D., Zhang, B., Zhang, H. J., Yang, Q., & Lin, S. (2006). Graph embedding and extensions: A general framework for dimensionality reduction. *IEEE transactions on pattern analysis and machine intelligence*, 29(1), 40-51. https://doi.org/10.1109/TPAMI.2007.250598

Yin, H., Wang, Q., Zheng, K., Li, Z., Yang, J., & Zhou, X. (2019, April). Social influence-based group representation learning for group recommendation. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)* (pp. 566-577). https://doi.org/10.1109/ICDE.2019.00057

Yu, J. F., Tang, W. B., Li, Y., & Zhang, J. (2016). Dimensional variation propagation modeling and analysis for single-station assembly based on multiple constraints graph. *Assembly Automation*, 36(3), 308-317. https://doi.org/10.1108/AA-11-2015-090

Yue, X., Wang, Z., Huang, J., Parthasarathy, S., Moosavinasab, S., Huang, Y., Lin, S. M., Zhang, W., Zhang, P., & Sun, H. (2020). Graph embedding on biomedical networks: methods, applications and evaluations. *Bioinformatics*, 36(4), 1241-1251. https://doi.org/10.1093/bioinformatics/btz718

Wang, D., Cui, P., & Zhu, W. (2016, August). Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1225-1234). https://doi.org/10.1145/2939672.2939753

Wang, Y., Liu, J. H., & Li, L. S. (2009). Assembly sequences merging based on assembly unit partitioning. *The International Journal of Advanced Manufacturing Technology*, 45(7-8), 808-820. https://doi.org/10.1007/s00170-009-1993-z

Zhang, J., Wang, Q., Wan, L., & Zhong, Y. (2005). Configuration-oriented product modelling and knowledge management for made-to-order manufacturing enterprises. *The International Journal of Advanced Manufacturing Technology*, 25(1-2), 41-52. https://doi.org/10.1007/s00170-003-1871-z

Zhang, X., & Ming, X. (2021). An implementation for Smart Manufacturing Information System (SMIS) from an industrial practice survey. Computers & Industrial Engineering, 151, 106938. https://doi.org/10.1016/j.cie.2020.106938

Zhao, S., Hong, J., Zhao, H., Li, S., & Ding, L. (2012). Research on assembly sequence planning method for large-scale assembly based on an integrated assembly model. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 226(4), 733-744. https://doi.org/10.1177/0954405411429248

Zhou, B., Bao, J., Chen, Z., & Liu, Y. (2021). KGAssembly: Knowledge graph-driven assembly process generation and evaluation for complex components. *International Journal of Computer Integrated Manufacturing*. Advance online publication. https://doi.org/10.1080/0951192X.2021.1891572

Zhou, B., Bao, J., Li, J., Lu, Y., Liu, T., & Zhang, Q. (2021). A novel knowledge graph-based optimization approach for resource allocation in discrete manufacturing workshops. *Robotics and Computer-*

Integrated Manufacturing, 71, 102160. https://doi.org/10.1016/j.rcim.2021.102160

Zheng, P., Wang, H., Sang, Z., Zhong, R. Y., Liu, Y., Liu, C., ... & Xu, X. (2018). Smart manufacturing systems for Industry 4.0: Conceptual framework, scenarios, and future perspectives. *Frontiers of Mechanical Engineering*, 13(2), 137-150. https://doi.org/10.1007/s11465-018-0499-5

Zheng, P., Xia, L., Li, C., Li, X., & Liu, B. (2021). Towards Self-X cognitive manufacturing network: An industrial knowledge graph-based multi-agent reinforcement learning approach. Journal of Manufacturing Systems, 61, 16-26.

Zhuang, C., Gong, J., & Liu, J. (2021). Digital twin-based assembly data management and process traceability for complex products. Journal of manufacturing systems, 58, 118-131. https://doi.org/10.1016/j.jmsy.2020.05.011

Zolotová, I., Papcun, P., Kajáti, E., Miškuf, M., & Mocnej, J. (2020). Smart and cognitive solutions for Operator 4.0: Laboratory H-CPPS case studies. Computers & Industrial Engineering, 139, 105471.

Liu, Z. J., Wan, J. J., & Wang, P. (2011). Research on Technology of Process-Oriented Virtual Assembly Modeling for Product Information. In Advanced Materials Research (Vols. 314–316, pp. 2177–2184). Trans Tech Publications, Ltd.

Rachuri, S., Han, Y. H., Foufou, S., Feng, S. C., Roy, U., Wang, F., Sriram, R. D., & Lyons, K., W. (2006). A model for capturing product assembly information. Journal of Computing and Information Science in Engineering, 6(1), 11-21.

Lv, M. Y., Hou, W. J., & Li, X. J. (2011). The research of hierarchy assembly semantic model based on intelligent assembly process planning. In Key Engineering Materials (Vol. 467, pp. 1933-1939). Trans Tech Publications Ltd.

Chen, X., Gao, S., Yang, Y., & Zhang, S. (2012). Multi-level assembly model for top-down design of mechanical products. Computer-Aided Design, 44(10), 1033-1048.

Kim, K. Y., Manley, D. G., & Yang, H. (2006). Ontology-based assembly design and information sharing for collaborative product development. Computer-Aided Design, 38(12), 1233-1250.

Wang, H., Xiang, D., Duan, G., & Zhang, L. (2007). Assembly planning based on semantic modeling approach. Computers in Industry, 58(3), 227-239.

Vilmart, H., Léon, J. C., & Ulliana, F. (2018). From CAD assemblies toward knowledge-based assemblies using an intrinsic knowledge-based assembly model. Computer-Aided Design and Applications, 15(3), 300-317.

Han, Z., Mo, R., & Hao, L. (2019). Clustering and retrieval of mechanical CAD assembly models based on multi-source attributes information. Robotics and Computer-Integrated Manufacturing, 58, 220-229.

Qiao, L., Han, F., & Bernard, A. (2011). A Unified Assembly Information Model for Design and Manufacturing. In Global Product Development (pp. 465-472). Springer, Berlin, Heidelberg.

Wang, J., Chang, Q., Xiao, G., Wang, N., & Li, S. (2011). Data driven production modeling and simulation of complex automobile general assembly plant. Computers in industry, 62(7), 765-775.

Bloomfield, R., Mazhari, E., Hawkins, J., & Son, Y. J. (2012). Interoperability of manufacturing applications using the Core Manufacturing Simulation Data (CMSD) standard information model. Computers & Industrial Engineering, 62(4), 1065-1079.

Bao, Q., Zhao, G., Yu, Y., Dai, S. (2022). Ontology-based assembly process modeling with element extraction and reasoning. Computer-Aided Design and Applications, 19(2), 280-292.

Zha, X. F. (2000). An object-oriented knowledge based Petri net approach to intelligent integration of design and assembly planning. Artificial Intelligence in Engineering, 14(1), 83-112.

Yao, J. T., Vasilakos, A. V., & Pedrycz, W. (2013). Granular computing: perspectives and challenges. IEEE Transactions on Cybernetics, 43(6), 1977-1989.