Engineering, 2026. https://doi.org/10.1016/j.eng.2025.11.034

Research

Intelligent Manufacturing—Review

Machine Learning-Based Cyber Manufacturing Services: A Review of Manufacturing Process Selection, Process Planning, and Design for Manufacturing

Xiaoliang Yan [a], Zhichao Wang [a], Shreyes N. Melkote [a], David W. Rosen [a,b,*]
[a] *The George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA*
[b] *Institute of High Performance Computing, Agency for Science, Technology, and Research (A*STAR), Singapore 138632, Singapore*

* Corresponding authors.
E-mail addresses: rosendw@a-star.edu.sg (D.W. Rosen).

Abstract: Cyber manufacturing services, which aim to connect geographically distributed designers and manufacturing service providers via the internet, are emerging to address the market shift from mass production to mass personalization. Recent advances in the Internet of Things (IoT) and machine learning enable new capabilities that promise improved efficiencies across the cyber manufacturing ecosystem. In this paper, we focus on machine learning methods that facilitate cyber manufacturing services in the areas of manufacturing process planning and design for manufacturing (DFM). To enable automated manufacturing process planning, we review recent advances in manufacturing capability modeling, manufacturing process selection, and feature recognition for process planning. To facilitate DFM, data-driven tools for generative design are reviewed and new methods and results presented. In the context of the literature review, we summarize work from our research group and present some new methods and results in the DFM area. Critical summaries of research challenges are provided to set the stage for recommendations on future research directions toward realizing cyber manufacturing services.
Keywords: Cyber manufacturing; Intelligent manufacturing; Machine learning; Manufacturing process planning; Design for manufacturing

1. Introduction

Cyber manufacturing, aimed at overcoming geographical barriers between manufacturing service users and service providers through information technology, is enabling a shift from mass production to mass personalization in manufacturing. The concept of cyber manufacturing emerged in the 1990s (for example, Ref. [1]) as manufacturing became more computerized and networked. Terms like "smart manufacturing" and "digital manufacturing" also became popular during this time. Recently, the term cyber manufacturing refers to a system that translates data from digitalized manufacturing systems into predictive and prescriptive operations to achieve resilient performance [2]. This marked a shift towards more interconnected, data-driven manufacturing powered by the industrial Internet of Things (IoT) and machine learning (ML). Recently, cyber manufacturing has gained traction with the rise of advanced technologies, including but not limited to cloud computing, ML, artificial intelligence (AI), and IoT. The digital model-based approach can enable on-demand and customized production. Concepts like smart factories, intelligent manufacturing [3], intelligent machine tools [4], digital twins [5], and collaborative robots have become the hallmarks of cyber manufacturing. The trend is towards more intelligent, resilient, and sustainable production through digital transformation. Cyber manufacturing has evolved considerably from early computerization to highly connected, automated, and data-driven factories.

The rise of digitization improves the accessibility of manufacturing data, enabling researchers and industries to employ ML for tasks like knowledge extraction, decision-making, design generation, simulations, and process monitoring and control. ML offers higher efficiency through automation of workflows compared to traditional methods that may rely on significant human intervention. In addition, formulating the mathematical relationships intrinsic to manufacturing knowledge is challenging due to the diverse factors involved, which span different process stages and encompass a wide range of heterogeneous knowledge. Therefore, developing fine-grained physical models is often impractical due to their significant computational requirements. ML-based surrogate models can supplement fine-grained physical models for many tasks, providing one approach to address this difficulty. To illuminate the potential applications of ML to facilitate cyber manufacturing, an in-depth review is crucial to summarize and analyze current research topics as well as identify the current research gaps.

The cyber manufacturing services context in which this paper is situated is illustrated in Fig. 1. A designer develops a preliminary design of a product, module, or subsystem. For each part, manufacturing processes and materials must be selected, this is Step 1. In Step 2, manufacturing planning and sequencing are performed. Subsequently, design for manufacturing (DFM) fine-tunes the part designs so that they are suitable for the selected processes and materials. Iterations between Steps 2 and 3 are expected as both the part and its manufacturing plan are progressively refined. Along this product development process, suppliers are identified who will fabricate each part. This paper focuses on Steps 1–3, as well as an element of Step 4, in Fig. 1. Topics covered in Sections 3–6 are shown at the bottom of Fig. 1 under their corresponding Steps. Manufacturing capability modeling (Section 3.1) and feature recognition (Section 3.2) are basic elements that enable manufacturing process selection (Section 3.3), which is the objective of Step 1. For Step 2, the topic of manufacturing operation sequencing (Section 4.1) is presented; for each operation, process parameter selection (Section 4.2) is needed. In Step 4, only the topic of operation sequencing is covered in a short Section 5. This topic was chosen since it builds on the contents of Section 4 and utilizes ML techniques that are similar to those used for operation sequencing. The contents of Sections 3–5 are literature surveys, while DFM in Section 6 represents new research from our group that has not been published before. The most common ML and deep learning (DL) techniques used for each topic are shown at the bottom of the figure.

In the context of cyber manufacturing, ML-based research advances for each of Steps 1–4 are reviewed with the purpose of answering the following three research questions:

(1) Can ML models learn the process capabilities of individual manufacturing processes in terms of the ranges of geometric shapes produced by that process?

(2) What types of ML models are capable of sequencing manufacturing processes in a process chain and developing process plans for each process?

(3) Can ML techniques be used to perform DFM, that is, modifying part shapes to improve their manufacturing characteristics?

These questions are central to the objectives of cyber manufacturing. Regarding the first question, the authors have found that it is difficult to articulate the range of shapes that a given manufacturing process can produce. Some efforts have been made to mathematically and computationally define manufacturing process capabilities using carefully crafted sets of machining features [7] and rules [8] on material removal volumes, accessibility directions, tool choices, etc. An extensive literature emerged on feature recognition from three dimensional (3D) part models based on machining volume decomposition [9] or reasoning from boundary representations (BReps) [10]. Good results were achieved in representing machining capability and in process planning for 3-axis machining and drilling, for example. However, these approaches were found to have limited scalability to complex feature interactions [11], more complex machining (e.g., 5-axis) processes and to other manufacturing processes, such as molding and casting. Furthermore, the proliferation of process variants, such as micro-milling vs. large scale milling, vertical axis vs. horizontal axis machine tools, etc., increases the difficulties associated with the analytical and manual approaches to process capability modeling and feature definitions. Recent research demonstrates that process capabilities can be represented by latent spaces generated by various ML encoders. The importance of this work is that it is foundational to manufacturing feature recognition and process selection. Research on these topics will be presented in Section 3.

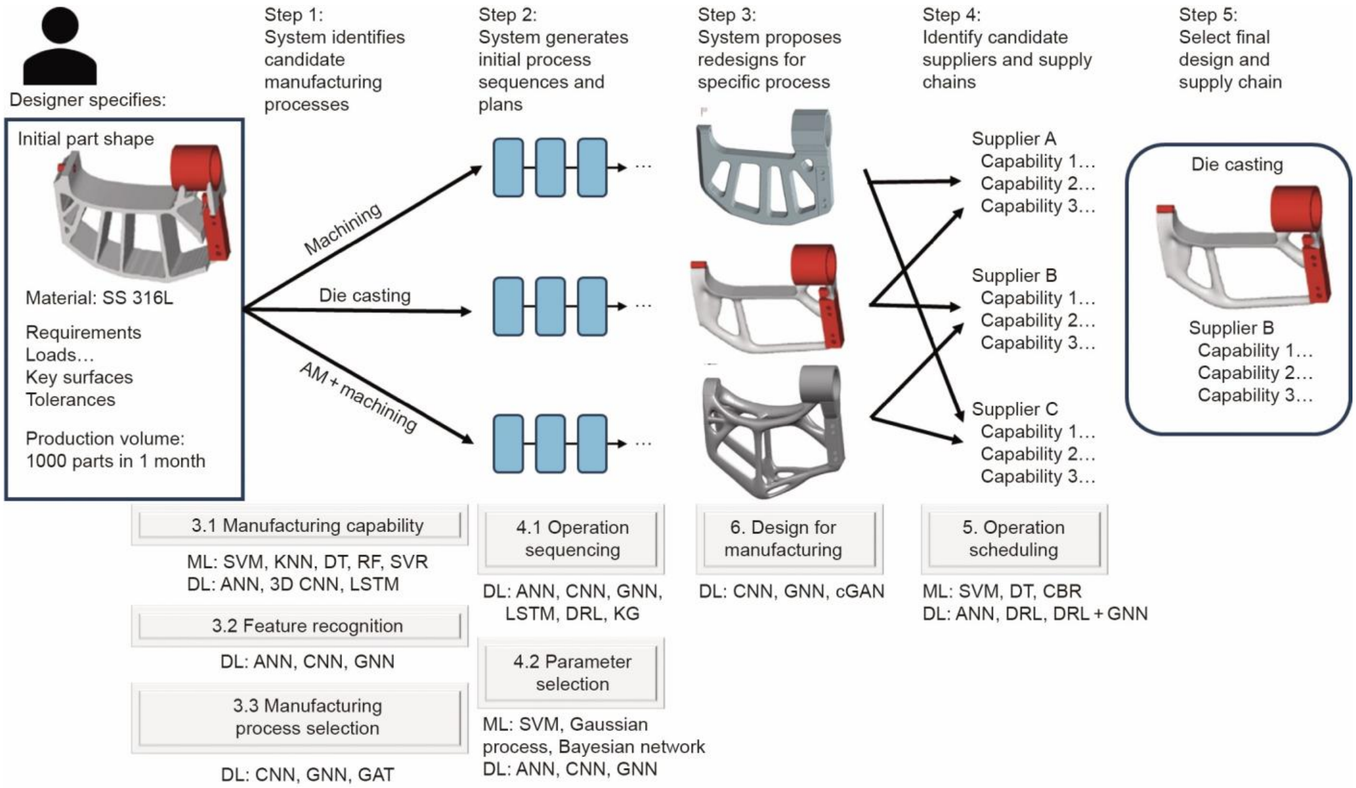


Fig. 1. Cyber manufacturing context for this paper. DL: deep learning; ML: machine learning; ANN: artificial neural network; CBR: case-based reasoning; CNN: convolutional neural network; DRL: deep reinforcement learning; DT: decision tree; GAN: generative adversarial network; cGAN: conditional generative adversarial network; GAT: graph attention network; GNN: graph neural network; KG: knowledge graph; KNN: k-nearest neighbor; LSTM: long short-term memory; SVM: support vector machine; SVR: support vector regression. (Part images from [6])

If a part requires more than one manufacturing process to be produced, then those processes need to be sequenced in the order that ensures all specifications are met while minimizing manufacturing costs and difficulties. Then, each process needs to be designed: toolpaths and process variable values should be determined. Traditionally, both tasks have relied on human expertise, applying experiences with similar parts and features using variant process planning approaches. In the machining domain, geometric reasoning about feature interactions guided decisions about feature precedence. However, limitations arose that were similar to those encountered for manufacturing process capability modeling and feature recognition, in particular, the combinatorial explosion of interacting feature types, many possible cutting tools, many possible accessibility paths, etc. [12]. Instead, with advances in ML, the opportunity for software models to learn from these experiences offers new possibilities for automation in Step 2. A related topic is operation scheduling, which becomes important after process planning is completed and suppliers are selected. This is a well studied, but difficult combinatorial optimization problem [12]. A small but growing body of ML-based literature is emerging on this topic to try to address the inherent computational challenges. A wide variety of research literature is reviewed on these topics in Sections 4 and 5.

DFM raises other types of issues. Again, traditionally DFM is performed by experienced designers who understand manufacturing process capabilities and limitations well. Typical DFM guidance is offered through Do/Don't examples and feature dimension examples [13]. An example of a Do/Don't rule for injection molding is to modify features on a part so that they do not cause undercuts. For feature dimension examples, recommended dimensions for snap-fit features, minimum feature sizes, and minimum feature spacing are common. For an ML model to learn DFM rules and practices, different approaches are likely needed since DFM requires the ability to apply abstract concepts and rules to a new part design, something that has likely never been seen before. Some initial work investigated product design in an intelligent manufacturing context [14]. However, newly developed generative AI models hold promise in being able to generate part designs according to rules learned from a range of examples. A challenge not yet mentioned is the need to ensure part functionality is met. Expert designers assess functionality while proposing design modifications; this capability must be built into ML-based models. Little literature has been published on ML-based DFM approaches. However, related work is surveyed briefly in Section 6, then some emerging example research is presented that has promise in performing DFM.

2. Background on ML and DL

A brief introduction will be provided to the conventional ML and DL techniques identified in Fig. 1.

2.1. ML techniques

Several conventional ML techniques were applied to manufacturing process capability modeling. Techniques such as *k*-nearest neighbor (KNN), support vector machine (SVM), decision tree (DT), and random forest (RF) were applied to classify parts according to their manufacturing process (Section 3.1 for descriptions and references). Hence, the representation of process capability was indirect; if a part was classified into a specific class, then the manufacturing process represented by that class could manufacture that part. KNN is an unsupervised method that groups data into *K* clusters based upon their distances from one another, as measured by an appropriate distance metric. As presented in Section 3.1, the metrics were based on shape descriptors computed from geometric part models, such as curvatures. In contrast, DT, RF, and SVM techniques are supervised learning methods that rely on labeled data. DTs learn rules by splitting data based on feature values to minimize error; RF is an ensemble of multiple DTs trained on different subsets of data and tend to be more accurate and less susceptible to overfitting than single DTs. SVMs compute boundaries between classes as hyperplanes that maximize the distance between classes. They can also be used for regression to compute a function that fits the data to within a margin of error, which was applied to parameter selection and operation scheduling tasks, as presented in Sections 4.2 and 5, respectively.

Artificial neural networks (ANN) can also be considered a conventional ML technique. However, their application to cyber manufacturing as discussed here is typically as a deep network, that is, one that has multiple hidden layers of neurons.

2.2. DL

DL ML techniques represent significant advances in the past decade that resulted from fundamental progress in computer vision, natural language processing (NLP), and real-time cyber-physical communication. In the remainder of Section 2, brief introductions of the primary neural network types, including convolutional neural networks (CNN) and graph neural networks (GNN) will be provided. Then, more specific learning paradigms will be introduced, including recurrent neural networks and reinforcement learning (RL) methods.

2.3. CNN

CNNs are a form of neural network that has enabled significant advances in a wide range of applications, particularly in computer vision-based tasks. A key element in a CNN is the convolutional layer, in which convolution operations and nonlinear activation functions are utilized to extract intrinsic features from the input image, voxel field, or video. The convolutional layers can be stacked together to extract higher-level information from the original input. However, using convolutional layers alone does not rapidly reduce the dimension of the input matrix. To overcome this, pooling layers are typically applied after convolutional layers to downsample the data. For example, max pooling, which searches for local maximum values, is commonly adopted in pooling layers. It is common for convolutional and pooling layers to alternate in a CNN to extract features and compress irrelevant information, which reduces computational effort. After convolutional and pooling layers, fully connected layers, also known as a multilayer perceptron (MLP), can be used to match the desired output dimension. During training of CNNs, the weights associated with each layer are adjusted to enhance the quality of the generated results.

A CNN as a type of DL approach usually comprises a sequence of layers of neural networks. Fig. 2(a) illustrates a typical CNN architecture. Note that the pairs of convolutional and pooling layers, depicted by the dashed rectangle in Fig. 2, can be repeated, or the convolutional layer itself can be stacked. The number of hidden layers is chosen based on the sophistication of the learning task. The architecture shown in Fig. 2 is commonly called an encoder since it effectively encodes the input image into a latent (condensed) representation. Convolution operations can be inverted to decode latent representations and regenerate images. For further insights into CNNs, several well-known references are available [15,16].

As applied to cyber manufacturing, CNNs operate on two dimensional (2D) images of parts or 3D voxel models. Their output, latent representations encode high level features (although not explicitly) that can be used to represent process capability and manufacturing features, which can be applied to process selection and operation sequencing tasks. CNNs are not limited to operating on images, however. Their convolution operations can be applied to recognize patterns in any datasets that are represented using vectors or matrices. Hence, they are also applied to other types of part information and shape descriptors, such as sampled points, curvature distributions, and their combinations.

2.4. GNN

In contrast to CNNs that process grid-structured data, GNN operate on datasets where the data elements are related by more general relationships, enabling a much wider range of datasets and applications [17]. Both triangle meshes and BReps are represented as graphs so much more informationally rich part representations can be processed directly using GNNs, compared to CNNs. Most applications reviewed here utilize a type of GNN that performs convolutions, similar to CNNs, based on neighboring nodes, in analogy to adjacent grid elements in a CNN. In a few cases, attention mechanisms are used to enhance GNN performance; these are denoted as graph attention networks (GAT).

2.5. Generative adversarial network (GAN)

As illustrated in Fig. 2(b), a GAN is composed of two neural networks, a generator and a discriminator [18]. GANs generate new designs using random noise as input. During the training phase, random noise is generated and fed into the generator, resulting in the production of new designs known as fake data. Both the fake data and real data are then passed to the discriminator, which tries to determine which is authentic. The penalties determined by the discriminator are subsequently transmitted back to update the generator. Throughout this training process, the generator and discriminator engage in a competitive interplay, continually enhancing their performance. Ultimately, the generator becomes proficient at generating realistic designs, which enables generative design.

2.6. Recurrent neural networks

Recurrent neural networks process sequential data, where context and order are important. As such, they are often applied to NLP, video analysis, and speech or audio processing. In this paper, a specific type of RNN was applied known as long short-term memory (LSTM) that has mechanisms for learning long-term dependencies in sequential data, while forgetting some information [19]. They utilize feedback connections and maintain an internal state. In this review, they have two main applications, processing time-series signals from machine tools and operation sequencing. Some implementations on the latter topic utilized bidirectional LSTMs to support reasoning in both the forward and backwards directions. The architecture of a typical LSTM cell is shown in Fig. 2(c), where the forget, input, and output gates, as well as the candidate memory are all neural networks. The input $\boldsymbol{X}_t$ is the time series data that is input to the RNN, while the memory and hidden state inputs are generated by the LSTM network.

2.7. RL

In contrast to CNN and generative models that rely on labeled data that requires ground-truth information for training, RL operates without such labels. Instead, RL is trained in a trial-and-error manner to learn an effective policy (set of rules) to decide which action to take given the current state of the environment [20] to maximize a cumulative reward. Stated differently, an agent learns to achieve a goal in an uncertain environment by performing actions and receiving feedback through rewards and possibly penalties. The general framework of RL encompasses two fundamental components: the agent and the environment as shown in Fig. 2(d). The agent serves as the focus of training and is capable of taking actions that cause the system to transition from state to state. The environment represents the world in which the agent exists and operates. As the agent takes actions based on the current state and its policy, the environment conducts observations, generates new states from the initial state, and derives the corresponding rewards. RL has been implemented in many different variants, with major differentiators being whether a model of the environment is developed (model-based) or no model is used (model-free) and whether learning is primarily focused on value learning [21] or policy learning [22].

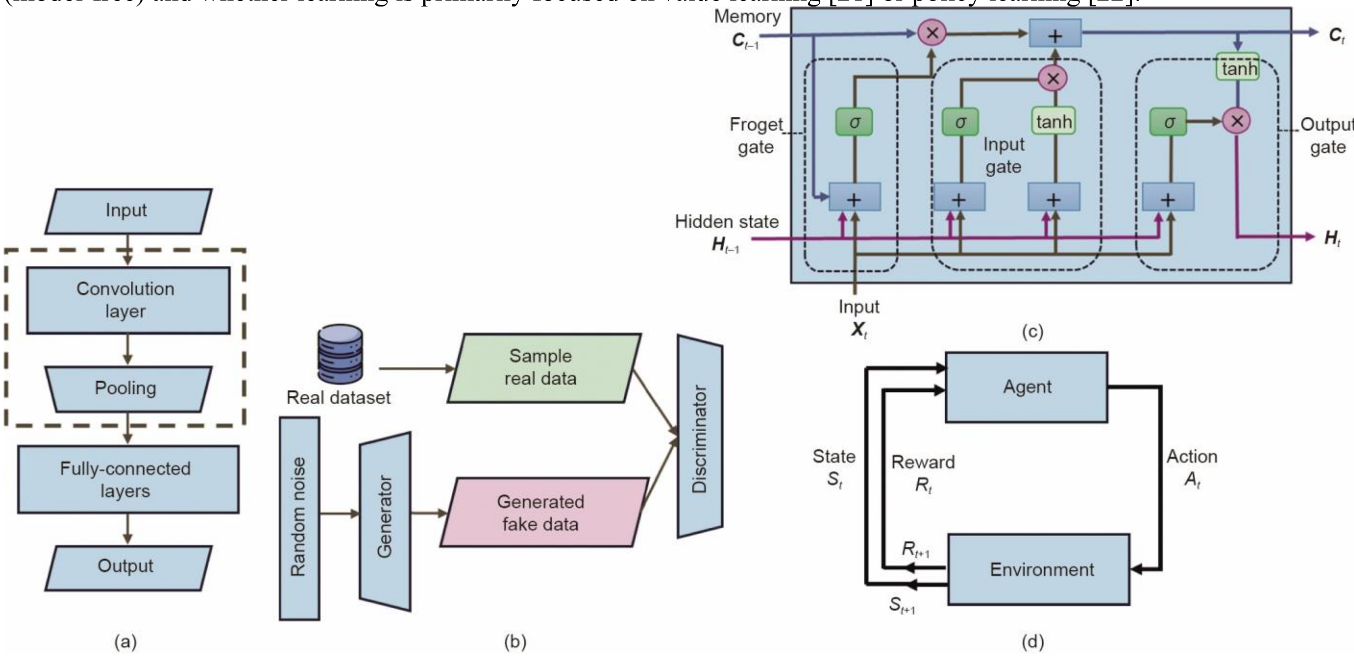


Fig. 2. General architectures of the ML techniques surveyed: (a) CNN, (b) GAN, (c) LSTM, and (d) RL.

3. Manufacturing capability modeling for process planning automation

The vision of cyber manufacturing aims to break the geographical barrier between designers and manufacturers by connecting manufacturing demands and resources via the internet. As a result, end users are provided with a growing number of available resources to physically realize part designs. The challenge is to select appropriate manufacturing resources and develop manufacturing plans that achieve technical and economic objectives. Addressing these challenges requires a knowledge base that captures the capabilities of manufacturing resources to achieve these objectives. By manufacturing capability, we mean the ability of a manufacturing process to achieve design specifications such as part shape, material properties, and quality attributes. Recent ML research related to aspects of manufacturing capability modeling, operation sequencing, and process planning such as feature recognition, process selection, process sequencing, and process parameter selection are critically reviewed and research gaps identified in the following sections. A typical workflow for manufacturing process planning is shown in Fig. 3 [23] that expands upon Steps 1 and 2 from Fig. 1 and assumes that DFM is not needed. Iterations between steps are not shown. Sections 3.1 and 3.2 correspond to the first activity of Step 1 in Fig. 3 on recognizing manufacturing features, while Section 3.3 focuses on the second activity on determining manufacturing processes.

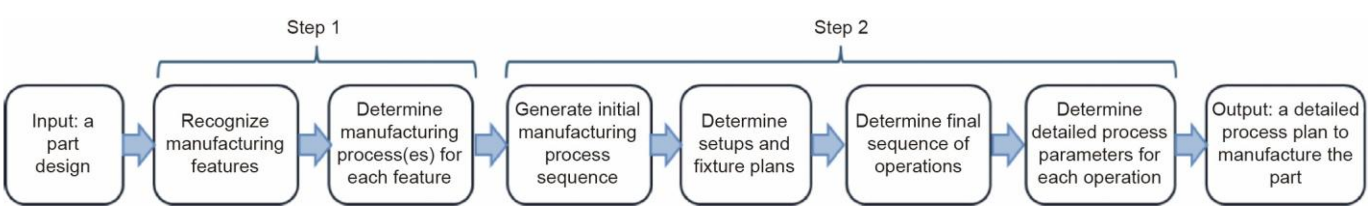


Fig. 3. Typical manufacturing process planning workflow. Reproduced from [23] with permission.

3.1. Manufacturing capability extraction and representation

The problem of selecting manufacturing processes for a given part design is an important topic in manufacturing. With the growing trend of utilizing digital/cyber manufacturing platforms to link manufacturing resources and capabilities with customers looking to source customized products, the problem of automatic process selection is more difficult due to the large number of manufacturing resources readily available online. The need is apparent: scalable representations of manufacturing process capability knowledge must be developed to facilitate efficient search, comparison, and selection of manufacturers capable of producing the product. Instead of researchers or practitioners mathematically and computationally defining manufacturing process capabilities, features, and rules, ML enables knowledge inference from data which has made it possible to learn the capabilities of manufacturing processes and resources automatically. An extensive review of research on process capability extraction and modeling is included here.

Ip et al. [24] utilized KNN and SVM [25] techniques to learn the shape generation capabilities of casting and machining processes. In these approaches, the D2 shape distribution (in Ref. [24]) and part curvature distribution (in Ref. [25]) were chosen as shape descriptors, which were calculated from part geometries represented using triangle tessellations. A drawback of using KNN [24] is that the computational requirement of the method grows significantly when the dataset becomes large. Also, the normalization of curvature values in Ref. [25] results in loss of information about extreme curvature values and thus loss of local shape details. In addition, high mesh resolution is required for these methods to compute accurate shape descriptors.

Hoefer and Frank [26] extended Ip and Regli's work by introducing more part shape descriptors to which they applied DT, KNN, and RF classifiers to predict if a part could be made by general machining or casting processes. Given a part represented by a tessellated surface mesh, the aggregate geometry metrics considered in their work included ① a volume to surface area ratio, ② a bounding box to part volume ratio, ③ a bounding box surface area to part surface area ratio, ④ an aspect ratio of longest to shortest dimension, and ⑤ a ratio of number of facets (in the tessellated mesh) to surface area. The slice-based metrics include ⑥ a visibility score that describes the tool accessibility in terms of achievable tool angles in x, y, and z directions, ⑦ a tool reachability depth defined by the maximum of the shortest distances from the part surface to each faceted edge of the part, ⑧ the maximum diameter of the tool that can reach every facet of the part without any obstacles, ⑨ the ratio between ⑦ and ⑧, ⑩ the number of required principal directions (i.e., x, y, and z directions) to manufacture the part, and ⑪ the number of rotations (apart from the principal directions) required to manufacture the part. The approach taken to determine the slice-based metrics is illustrated in Fig. 4. The facet-based orientation metrics included ⑫ the angle between the tool (of either end milling or face milling) and the normal of each facet of the tessellated mesh of the part, and ⑬ the maximum deviation angle between the facet normal to the principal directions of each facet. For each part, these 13 metrics were calculated, and then the manufacturing process (either milling or casting) was predicted using the built classifiers. While the use of multiple shape descriptors ensures that the classifiers can capture the shape generation and tooling capabilities of milling and casting, many of the metrics were specifically designed for the milling process and may not be applicable when other manufacturing process labels are introduced to the classifiers.

Zhao et al. [27] utilized interpretable decision tree based classifiers to learn the capabilities of discrete manufacturing processes including turning, milling, and casting in terms of their capability to generate part shape, to handle different input materials (defined in terms of their mechanical and physical properties), and to generate the specified quality. Their work is limited by the assumption that all features in the part are produced using the same manufacturing process, ignoring the possibility that different features can be made using different discrete manufacturing processes.

Wu et al. [28] introduced RF-based classifiers to describe the milling process capability in terms of tool wear rate. The inputs to their classifier are the instantaneous cutting forces and vibrations in x, y, and z directions and acoustic emission sensor readings. A dataset of 315 milling operations under the same cutting condition was used to train and test the classifiers. Their results show that the proposed RF classifier outperformed the ANN and support vector regression (SVR) classifiers in terms of prediction accuracy. However, the authors did not test the generalization performance of the classifiers for predicting tool wear rates based on varying cutting conditions. No information about input stock or the target manufacturing feature to be generated was considered by the proposed classifiers. Furthermore, the authors did not compare their method against time series forecasting methods (e.g., LSTM [19]), which would have been reasonable since the tool wear rate they predicted is a time-variant attribute.

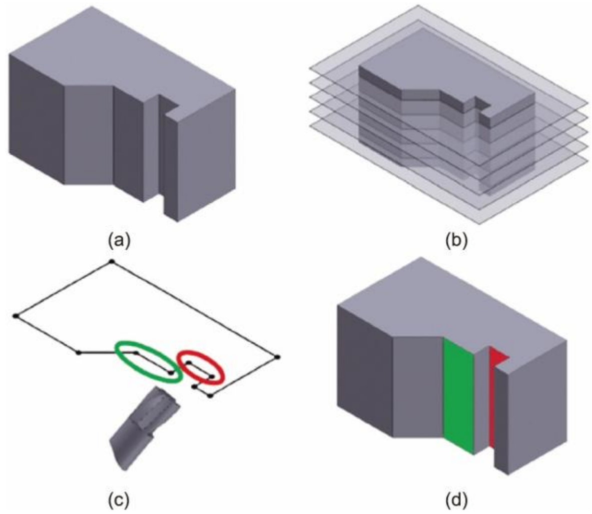


Fig. 4. Slice-based machining analysis of (a) an example part, that is (b) sliced; (c) each slice is analyzed for manufacturability, and (d) results are mapped back to the Stereolithography Language (STL) model. Reproduced from [26] with permission.

Angrish et al. [29] represented the capabilities of a manufacturing service provider by the parts successfully manufactured by the provider in the past. Parts were described in terms of their geometry, material name, and part tolerance. Given a query part that needs to be manufactured, they utilized the KNN algorithm to find the provider with the greatest number of similar parts in their repository. On the other hand, part material and quality in this method are described in a categorical manner (e.g., metal or non-metal, high tolerance or low tolerance) rather than quantitatively. Part shape was described by only global shape descriptors. Moreover, their system presumed the presence of multiple comparable products to the queried part, which may not be always true.

Ghadai et al. [30] utilized a 3D CNN to learn the DFM rules from voxelated part design geometries and used the rules to predict the manufacturability of drilled hole features in part designs. However, the approach considers part design as a whole without locating the local source(s) of non-manufacturability.

Rønsch et al. [31] implemented DL and regression algorithms for predicting the quality of injection-molded plastic parts based on material information and process parameters (e.g., mold temperature). They compared ANN, CNN, and LSTM techniques on their sensor data streams and found similar performance; however, they demonstrated that a combination of sensor data and material type yielded better prediction accuracy than sensor data alone.

Guo et al. [32] proposed a framework to extract knowledge of machining process capabilities from text-based descriptions using NLP methods, including the LSTM technique, to construct a knowledge graph. To apply the knowledge base to a specific part, the user develops a text description of features and quality requirements (tolerances, surface finish specifications). Han et al. [33] proposed a data-driven model for mapping the relationship between a part CAD model and manufacturing process data supported by a knowledge graph and an LSTM network. Kumar and Starly [34] proposed an NLP framework for extracting manufacturing process knowledge from abstracts of manufacturing articles. These proposed approaches combined a bidirectional LSTM and a conditional random field to extract both qualitative (e.g., processable material types) and quantitative descriptions (e.g., achievable ranges of surface roughness and size tolerance) of machining process capabilities. However, these methods only consider text-based information as input for knowledge extraction while ignoring information in other forms (e.g., 3D models of the part shapes) so detailed information of part shape is lacking.

Zhang et al. [35] proposed a DL approach that combined a CNN and an ANN to learn and predict manufacturability of a laser powder bed fusion (PBF) process. Based on voxelated part design geometry and text-based descriptions for the material (e.g., material supplier, material name, and material density in loose form) and printing machine (e.g., machine supplier and type), the proposed approach predicts if each voxel, which reflects the input part design, can be manufactured or not. Barrionuevo et al. [36] proposed using regressors such as SVM, DT, RF, KNN, gradient boosting, Gaussian process (GP), and MLP to predict the relative density of parts made using a PBF process based on process parameters including laser power, scanning speed, layer thickness, hatch spacing, and particle size. The predictions were limited to 316L stainless steel parts.

Table 1

Manufacturing capability extraction literature.

Manufacturing process	ML techniques	Input	Shape descriptors	Reference
Part type	KNN	Tri mesh	D2	[24]
Casting, machining	SVM	Tri mesh	Curvature	[25]
Milling, casting	DT, RF	Tri mesh	Geometric metrics	[26]
Turning, milling, and casting	DT	Tri mesh	Curvature, D2, and symmetry	[27]
Milling	RF, ANN, SVR	Cutting forces, vibration	—	[28]
Part type	KNN	CAD model	Global shape desc.	[29]
Drilling	3D CNN	Voxel	—	[30]
Injection molding	ANN, CNN, LSTM	CAD model	Features	[31]
Machining	NLP, LSTM	Text	Knowledge graph	[32]
Machining	Knowledge graph, LSTM	CAD model	Grammar	[33]
All	NLP: LSTM, RF	Technical text	Word2vec	[34]
PBF	CNN, ANN	Voxel, text	—	[35]
	SVM, DT, RF, KNN, GP	Process variables	—	[36]

The approaches taken on this topic are summarized in Table 1 [24-36], where the manufacturing processes, ML techniques, inputs, and shape descriptors are documented. The first four [24–27] took similar approaches of applying standard ML techniques to well known shape descriptors, such as curvature distributions and D2, computed from a geometric part model to capture the class of shapes that are producible by a manufacturing process. Ref. [28] took a different approach for milling, where process capability was captured based on cutting forces and vibrations, not part geometry or shape. RF, ANN, and SVR were the ML techniques investigated for this purpose. In ref. [29], the objective was the functional classification of parts. The researchers used a set of global shape descriptors computed using part CAD models, then applied a KNN classifier to sort the parts into appropriate classes. Ref. [30] captured drilling capabilities using a 3D CNN with voxel models of parts as inputs. This approach is similar to many taken for feature recognition (Section 3.2). Refs. [32,33] utilized knowledge graphs to characterize process capability but supported reasoning using different approaches. In both cases, an LSTM network was employed to learn from text (which is a sequence of words). Ref. [34] also used NLP and an LSTM network to learn from technical text describing manufacturing processes, but did not build an explicit knowledge graph. LSTM networks are also useful for dealing with sensor data, which is again sequential, as investigated in Ref. [31]. Finally, results from ref. [35] reinforced the benefits of utilizing some information beyond part shape to capture process capability, in this case, adding process settings and material information as inputs.

3.2. Manufacturing feature recognition

An essential aspect of cyber manufacturing is automatic process planning which aims to automatically generate a process plan, including the sequence of processes and other process details (e.g., machine, fixture setting, process parameter setting, etc.), from a product design. First, ML approaches to feature recognition will be surveyed, which is a prerequisite for process planning, sequencing, and scheduling, topics to be addressed in the sequel.

Much of the recent work on manufacturing feature recognition has pursued 2D or 3D CNNs as their primary ML method, along with a range of part representations and shape descriptors. More recently, GNNs have shown great promise for feature recognition. Research approaches are summarized in Table 2 [5,37–56]. Note that all research reviewed here focused on machining processes, except for one paper. A clear evolution is evident in the research, from early ANN to CNN to GNN approaches.

Table 2
Manufacturing feature recognition research.

Manufacturing process	ML techniques	Input	Shape descriptors	References
Machining	ANN	BRep	AAG	[37,38]
Machining	ANN (PointNet++)	Tri mesh → point cloud	Normal vector clusters, FAG	[39]
Machining	ANN, transformer	CAD model → point cloud	Point patches	[40]
Machining	3D CNN	CAD model → voxel	—	[41]
Machining	3D CNN	Dexel	—	[42]
Machining	2D CNN	2D views	ResNet	[5,43,46]
Machining	2D CNN	Cross-section views	Feature segmentation	[44]
Machining	3D CNN	STEP → voxel	AAG	[45]
Machining	3D CNN	Voxel, octree	—	[47]
Machining	GNN	BRep	AAG	[48,51]
Machining	GNN	BRep, tri mesh	FAG	[49]
Machining	GNN	BRep, points	FAG	[50]
Machining	GNN	BRep	AAG	[52]
Machining	GNN, GAT	BRep	FAG	[53]
Machining	GNN	BRep, points	AAG	[54,55]
Sheet metal	GNN	STEP, points	AAG	[56]

FAG: face adjacency graph; STEP: standard for exchange of product model data; AAG: attributed adjacency graph;

Jian et al. [37] proposed a feature recognition method for prismatic part designs represented using the STEP format. Given a design, an attributed adjacency graph (AAG) was built, and subgraphs were identified by trimming the edges associated with convex edges. Finally, each subgraph was input to an ANN to predict its corresponding manufacturing feature. They adopted an evolutionary optimization method, called the novel bat algorithm, for training the ANN. Zhang et al. [38] improved Jian's method by introducing the Artificial Bee Colony algorithm, instead of the novel bat algorithm, to further enhance the performance of the ANN. More recent research using ANNs has utilized point cloud representations, sampled from triangle meshes or CAD models, as inputs. Yao et al. [39] added normal vector information for each point, then clustered those vectors, as well as a face adjacency graph (FAG) as shape descriptors. These shape descriptors were input into a well known ANN, PointNet++ [40], to recognize features. In contrast to all the other research surveyed in this section, Zhang et al. [40] took a self-supervised learning approach to feature recognition. They sampled CAD models to generate point clouds, then clustered points into point patches which served as inputs to their transformer-based ANN. Training was performed in two stages, with the first utilizing unlabeled data and the second using a small amount of labeled data. In experiments, they demonstrated semantic segmentation accuracies better than 99%, outperforming PointNet++.

Zhang et al. [41] proposed FeatureNet which utilized a 3D CNN for recognizing voxelated machining features. Fu et al. [42] proposed a dexel representation for 3D part geometry that utilized varying spatial resolutions to efficiently capture both global shape and detailed local shape information to facilitate training of a 3D CNN for feature recognition. However, the proposed representation requires much larger storage space than a standard voxel representation since additional dimensions (other than x, y, and z) are required. Zhang and Zhou [43] proposed a 2D CNN to recognize manufacturing part categories based on a set of 2D views of the part. However, the method classifies the part as a whole and lacks recognition of detailed features in the part. Shi et al. [44] proposed a CNN for recognizing manufacturing features in a part from cross-sectional views. Ning et al. [45] utilized a graph-based method to segment BRep models into individual features based on convex edges, and they utilized a CNN to classify each segmented feature. Example feature recognition results are shown in Fig. 5. A limitation of these feature recognition methods in the context of automatic process planning is that they only consider part shape information and do not relate the recognized features to manufacturing processes that can produce them. An exception is the digital twin framework proposed by Zhou et al. [5]. As an extension of Ref. [43], detailed cutting tool selections were made based on a set of if-then rules that considered the relationship between recognized feature types, manufacturing processes, and the corresponding cutting tool. However, in their method, identification of manufacturing features in the part and manufacturing process selection need to be performed by the user. Zhou et al. [46] associated manufacturing feature recognition with cutting tool selection. They utilized ResNet [48] to recognize special manufacturing features, represented using multiple views of engineering drawings, and associated cutting tools as class labels. Therefore, given new part designs with similar features, their trained CNN can recognize the feature and assign a feasible cutting tool to machine it. Most of the prior works on ML-based manufacturing feature recognition reviewed here utilized a voxel-based part geometry representation, which necessitates a tradeoff between feature resolution and computational cost. In recent work, Vatandoust et al. [47] examined this tradeoff in more detail by comparing the performance of 3D CNN models utilizing octree-based sparse voxel part representation (O-CNN) with the performance of conventional voxel-based 3D CNN models such as FeatureNet. Their results showed that the O-CNN model yields higher feature recognition accuracy while requiring lower GPU memory consumption than a voxel-based model of comparable feature resolution. The automation of workflow starting from feature recognition to detailed process planning considering different design and manufacturing information (e.g., geometry, material, and quality) is still lacking.

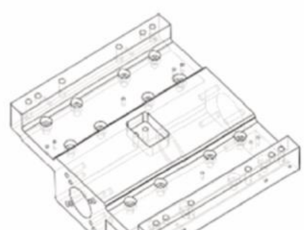
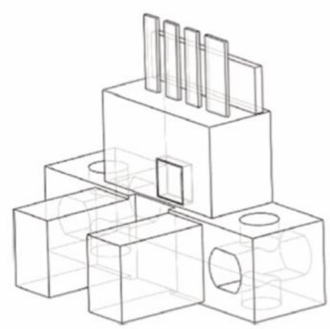
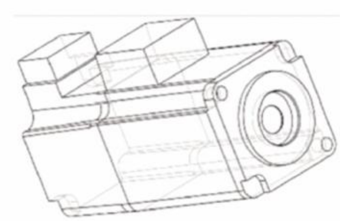
	Part 1	Part 2	Part 3
3D model			
Characteristic quantity	77	14	14
Number of separation features	77	13	14
Number of correct convex features identified	0	8	5
Number of correct concave features identified	71	5	8
Accuracy	100%	92.86%	92.86%

Fig. 5. Example feature recognition results for three parts using a 3D CNN and voxel representations. Reproduced from [45] with permission.

Others have experimented with GNN approaches to feature recognition. In 2020, Cao et al. [48] was one of the first groups to investigate GNN-based machining feature recognition of prismatic parts. They utilized a FAG derived from the BRep of the part, along with planar surface models, and sampled points. A large dataset of parts was generated with a variety of machining features, including slots, passages, pockets, steps, holes, chamfers, and rounds. They demonstrated that their GNN approach performed better than a voxel-based method, called submanifold sparse convolutional network (SSCN) [] that won a 2017 segmentation competition. In 2022, members of the same research group [49] presented the Hierarchical CADNet, again for machining feature recognition, where their graph had two levels, one that represented the topological structure using a FAG, and a second level that represented the geometry using a triangle mesh. In 2021, a research team from Autodesk [50] proposed a novel GNN called UV-Net that encodes part BReps using a FAG, as well as sampling points on surfaces and curves. UV-Net can be considered a hybrid of graph and CNN, where the CNN preprocessed sampled points into node embeddings for input into the GNN. Another Autodesk publication [51] proposed a somewhat different GNN called BRepNet that operated directly on BReps without the sampled point processing. The authors developed a novel message passing mechanism to exchange information among the topological entities (faces, edges, and vertices) to recognize features in parts with prismatic and curved geometry. Inspired by graph-based and mesh-based approaches, a different hybrid approach was presented in 2023, called DeepFeature [52], that utilized an AAG part representation, derived from a BRep, as input to their GNNs. A subgraph extraction algorithm separated single features which were input into GNNs, each of which recognized a specific feature type from CAD models, and then six GNNs were employed to identify the type of the isolated features. An approach that utilized FAGs with a novel graph attention network, called BRepGAT [53], did not rely on sampling points, but still achieved better than 99 percent feature recognition accuracy. Most recently, a research group presented AAGNet, a GNN that proved very successful in recognizing machining features [54]. They called their approach a multi-task neural network since it performed semantic segmentation, instance segmentation, and bottom face (of features such as pockets) segmentation to recognize machining features, faces associated with each feature, and their bottom faces. In their extensive testing, they demonstrated feature recognition accuracies of 98%–99.8%. Xia et al. [55] also investigated a GNN approach with an AAG-based part representation, but their GNN had three output heads, one each for semantic segmentation, instance segmentation, and topological relationships among features. This structure enabled their network to correctly recognize a wide variety of interacting features.

Finally, Ma and Yang [56] proposed a GNN approach to feature recognition in sheet metal parts where an enhanced AAG was used for part representation. They also experimented with a two-stage training approach. They first trained their network on a large training dataset (49 000 models) with a subset of 12 features, then they introduced 11 new features into the smaller second training dataset (7000 models). Experiments demonstrated that their network learned much faster than the benchmark approaches and demonstrated excellent recognition capabilities on their datasets.

The three broad approaches surveyed here, using ANN, CNN, or GNN techniques, each utilize mostly different types of input part information. The approaches using ANNs utilized either AAG or point cloud part representations as inputs. The CNN approaches used geometrically discretized representations (pixel images in 2D, voxel models in 3D), while GNN approaches used BRep-derived face-edge adjacency graphs, along with point clouds in many cases. Strikingly, all three broad approaches performed well, with some accuracy results exceeding 99% in each case. The combination of surface connectivity and geometric (point cloud) information used in the recent GNN approaches [56–59] enables comprehensive representations of parts that can perhaps be expected to yield excellent results. However, results that were almost as good were achieved with only point cloud information, without explicit connectivity relationships. This suggests that connectivity information is not as important in recognizing features, or CNN technology is so advanced that the implicit connectivity inherent in grid-based representations is good enough. Further work is needed to clarify the underlying reason for the similar performance of CNN and GNN approaches in representing part geometry.

3.3. Manufacturing process selection

Automated process selection is necessary in generative Computer-Aided Process Planning (CAPP) systems, which seek to automatically generate a detailed process plan for the input part design. Previous CAPP systems (e.g., Ref. [60,61]) have focused heavily on machining. Recent process selection systems [62–64] often employ a feature recognition module to acquire part shape information that is associated with manually encoded machining operation capability. Hamouche and Loukaides [65] proposed a 2D CNN for sheet forming process selection to decide among several bending and stretch-forming methods. Part geometry was sampled into point clouds and several curvatures were computed (principal, Gaussian, and mean) and input to the CNN. The output was a manufacturing process label. Peddireddy et al. [66] proposed a 3D CNN for selecting milling and turning processes based on voxelated geometry of the part, an approach that was similar to those presented in the feature recognition section. The model considers part shape information only while other information like material and quality attributes were not included.

Wang and Rosen [67,68] utilized the heat kernel signature [69] and rotation invariant shape descriptors to represent part geometry and utilized 2D and 3D CNNs to select if machining, laser cutting, or injection molding can manufacture the given part design. The heat kernel signature approach used triangle mesh part models, while the second approach [68] used point clouds to represent parts. Interestingly, the point cloud-based shape descriptors proved better at representing shapes for manufacturing process classification than mesh models, even though they have less shape information directly represented. The rotation invariant shape descriptor comprised of seven distances that capture both local and global shapes. These distances and point coordinates were fed into an MLP, followed by a max pooling operation to comprise one layer, denoted dRIConv (distance Rotation Invariant Convolution) of the neural network that was constructed for this work, shown in Fig. 6. The entire network consisted of 3 dRIConv layers followed by 3 fully connected layers, the outputs of which were probabilities for each of the four candidate manufacturing processes. Compared to other part classification methods, such as RSCNN [70], DensePoint [71], and DGCNN [72], the dRIConv method was superior, particularly under arbitrary rotations.

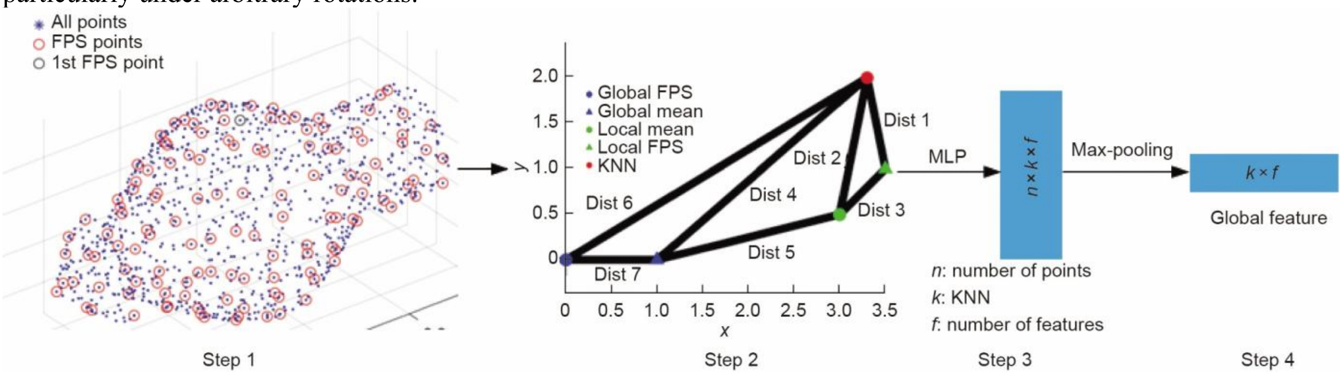


Fig. 6. One layer of the dRIConv shape descriptor and neural network used for point cloud-based manufacturing process classification. Reproduced from [68] with permission. FPS: farthest point sampling.

Yan and Melkote [73,74] proposed a 3D GAN to describe the shape generation capability of turning and milling operations and thus assist process selection for given part designs using voxel representations. In their approach, they trained separate 3D GANs for different machining operations (turning, grooving, and chamfering), where training data were limited to parts producible by each operation type separately. With this approach, they did not need to label their training datasets, other than to specify the operation type. Zhao and Melkote [75] proposed a 3D CNN for machining process selection based on part geometry and quality attributes, and they extended the work by adding material information [76]. However, the model was limited to selecting processes for part designs consisting of a single manufacturing feature.

Of course, many parts require more than one manufacturing process to fabricate all shapes and achieve all requirements. Recognizing features for more than one manufacturing process requires dividing part geometry into regions, where the different regions correspond to geometry that is produced by different processes. From an ML viewpoint, this corresponds to a segmentation task. Furthermore, it is important to distinguish different features of the same type, meaning that instance segmentation is preferred. For example, if two slots are recognized, it is helpful to distinguish between the two since they will have to be treated separately for operation sequencing, process planning, and scheduling purposes. This issue is related to some of the feature recognition work presented in Section 3.2, but that work was limited to only machining processes. Research is underway to address manufacturing process selection for parts requiring multiple processes by extending the work reported in [67,68] using an extension of the rotationally invariant convolution methods in [77]. Process classification results of better than 96% accuracy have been achieved on die cast parts that also require milling and/or turning [78].

Recently, a different approach was taken to manufacturing process selecting using a variant of GNN known as GNN with self-attention capabilities, also known as GAT (graph attention NN). In Ref. [79], the authors proposed a GAT architecture with three input heads for BRep (part topology), triangle mesh (geometry), and quality (tolerances, surface finish) information. Part features and surfaces were classified according to three roughing processes, milling, turning, and drilling, and two finishing processes, grinding and reaming. The architecture of their GAT network, called MaProNet, is shown in Fig. 7. 60 000 CAD models were constructed with quality information associated with their surfaces. Results demonstrated accuracy of about 92%–93%.

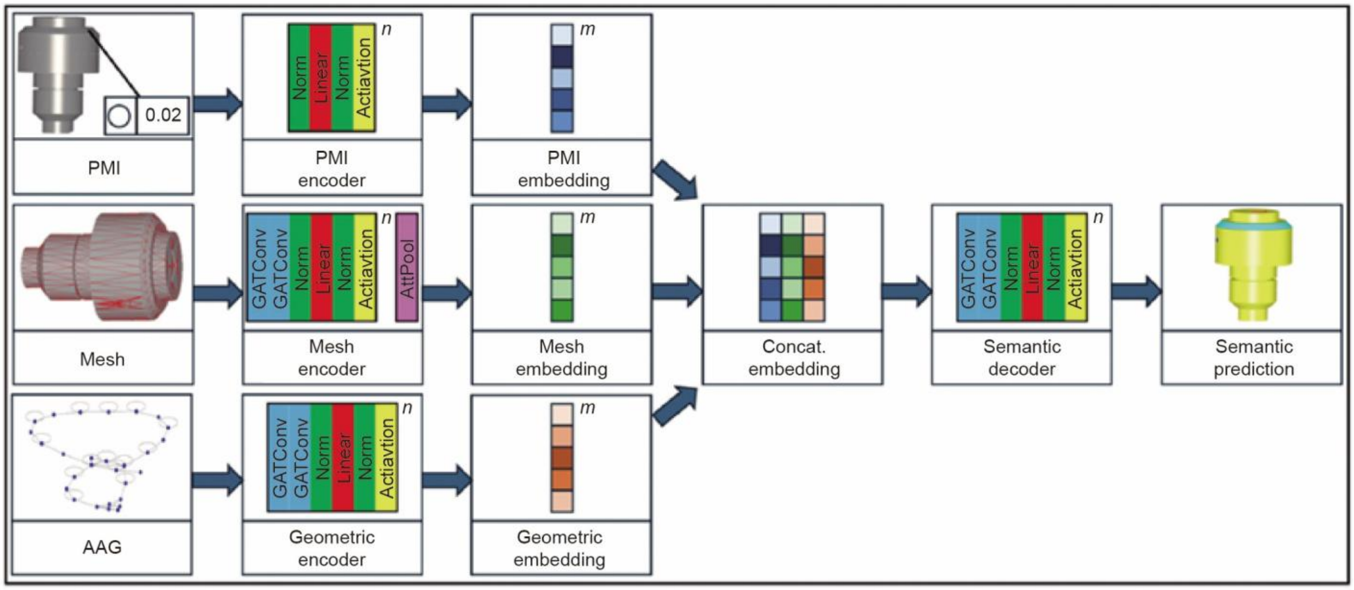


Fig. 7. Architecture of MaProNet GNN for manufacturing process selection. *m*: size of embeddings; *n*: number of layers in encoders and decoders; PMI: part manufacturing information. Reproduced from [79] with permission.

These manufacturing process selection methods are summarized in Table 3 [65–68,73–76,78,79]. As mentioned, they all utilized some form of CNN; however, their inputs varied widely, from point clouds to voxel models. Some included quality information as well, such as tolerances, while most focused on purely geometric information. A limitation of this approach is that the fusion of information regarding both design and manufacturing is lacking, which has been shown to be advantageous in other applications [80,81]. Regarding the two references that utilized generative models [66,67], the authors claim their GANs "…model the shape transformation capability as a latent probability distribution from which visualizations of realistic machinable features can be sampled for shape decomposition and reconstruction", which represents a different approach than the rest of the research efforts. For the feature recognition task, GNNs and variants seem to be emerging as the most promising approach; it will be interesting to see if GNNs also become preferred for process selection. To a large extent, this issue is related to the type of inputs that are used. If CNN-based methods work the best, then point clouds, mesh models, or voxel models should be used inputs. However, if graph-based BReps become the preferred inputs, then the usage of graph-based neural networks will likely become widespread.

Table 3
Manufacturing process selection literature.

Manufacturing process	ML techniques	Input	Shape descriptors	References
Sheet forming	2D CNN	Point cloud	Curvature	[65]
Milling, turning	3D CNN	Voxel, tolerances	Features	[66]
Milling, turning, laser cutting, and inj. molding	2D CNN	Tri mesh	Heat kernel signature	[67]
Milling, turning, laser cutting, and inj. molding	2D CNN	Point cloud	Rotational invariant shape descriptor	[68]
Milling, turning	3D GAN	Voxel	—	[73,74]

Milling	3D CNN	Voxel, quality, material	—	[75, 76]
Machining, casting	2D CNN	Point cloud	Rotational invariant shape desc.	[78]
Machining	GAT	BRep, tri mesh, quality	AAG	[79]

inj.: injection molding;.

4. ML for manufacturing operation sequencing, planning, and scheduling

Most research efforts in process operation sequencing are rule-based and require optimization (e.g., [82,83]), though direct usage of ML methods has also appeared in recent advances. Related work reviews knowledge graph methods for process planning [84], although the scope of that work covers machining process planning to fault diagnosis planning to generic process models using large language models. This section reviews ML methods in operation sequencing, planning (process variable setting), and scheduling.

4.1. Manufacturing operation sequencing

Zhang et al. [85] proposed a DL approach for predicting feature precedence in process planning. The method converted the part design information including voxelated part geometry, dimensional tolerance, and surface roughness to a fourth-order tensor. The relationships (as defined in Ref. [86]) between the faces of the part were represented using a relation matrix. The tensor and relation matrix were put into an encoder to generate high-level representations, and an LSTM network-based decoder was used to predict an ordered manufacturing process sequence to realize the part.

Some research utilized manually encoded machining features, tool access directions (TAD), and quality information (tolerances and surface finish specifications), rather than a 3D geometric part model. Wu et al. [87] proposed a process planning approach using deep RL (DRL). The proposed approach covers both tool selection and operation sequencing by optimizing the overall cost, which is modeled as a summation of machine cost, tool cost, cost associated with setup changes, cost associated with machine change, and cost associated with tool change. Each manufacturing operation is described by a TAD, machine, and cutting tool. The achievable tolerance for each operation is also considered. The Monte Carlo (MC) sampling method was used to evaluate the policy, and the actor-critic algorithm was used to find the optimal policy. On the other hand, the achievable tolerance needs better modeling, and other factors such as material processability should be considered when determining if an operation is feasible. Zhang et al. [88] proposed a setup planning method for prismatic part designs based on an ANN. Given the part design, the manufacturing features were detected and grouped based on TAD. The sequence for machining each group was then ordered based on (a) the number of features in the group, (b) the cross-sectional area of the step feature in the group, and (c) the cross-sectional area of the open pocket feature in the group. Finally, the locating datum of each feature group was determined using an ANN considering the surface roughness, dimensional tolerance, and tool achievability. The method was limited to producing features using a 3-axis milling machine, and the rule-based sequencing approach lacked scalability in terms of accommodating more features.

Sugisawa et al. [89] utilized deep inverse RL (DIRL) to generate machining sequences to realize a given part design. Given a part design represented as a BRep, a graph is generated to represent the information of each edge (e.g., length) and face (e.g., area), their connectivity, and their associated quality requirements including surface roughness, concentricity, and dimensional tolerance information. A set of primitive removal volumes was identified, and the DIRL was used to estimate the priority of each removal volume, thus predicting an ordered list of volumes to be removed from the stock. Zhang et al. [90] used stacked auto-encoders to learn the relationships between the part design and manufacturing process analysis rules. Given a part design described in terms of geometry, tolerance, and surface roughness, the process analysis rules were selected by the autoencoder, each of which is associated with a predefined machining process step, and the series of process steps was formed into a manufacturing process graph as a search space. A swarm optimization algorithm was applied to generate an ordered sequence of process steps that can realize the given design part.

Zhao et al. [91] proposed a manufacturing process sequence suggestion method using a data-driven framework, where input information included part geometry (STL file), mechanical properties of the material, and quality requirements in terms of tolerances and surface finishes. A collection of 315 part examples across 109 unique manufacturing processes served as the training dataset. The method determined the frequently occurring manufacturing process sequence patterns, each consisting of an ordered list of manufacturing processes, and learned the capability of each pattern defined in terms of the part shape, processable material, and achievable part quality metrics. Given a 3D part design query, the framework suggests process sequence patterns that are potentially capable of realizing the part design. In subsequent work [23], this module was integrated with a GNN to identify manufacturing features and a CNN to identify the manufacturing processes for those features, as shown in Fig. 8. The resulting system could automatically select manufacturing processes and generate process sequences given a part CAD model.

In other recent work from our research group, a recurrent CNN approach was used for operation sequencing for milling and drilling operations [92]. Traditionally, manufacturing engineers specified precedence rules or constraints among operations manually for various part shapes and for all operation combinations. In contrast, this work pursued a sequential learning approach that trained a ML model on valid sequences of operations that have historically produced parts that meet specifications. For example, the traditional approach to thin-wall interactions between slotting and drilling is to drill the hole first, then mill the slot, pocket or other feature that produces the thin wall. This is intended to prevent excessive vibration during drilling. In a practical scenario, companies with long histories of part manufacturing will have many examples that can serve as the training dataset. In this work, 3D voxel part models were used as inputs. The ML architecture consisted of a 3D CNN encoder, a bidirectional recurrent unit, and a 3D CNN decoder and was called the 3D convolutional recurrent neural network (3D-ConvRNN). By embedding the precedence relations from successfully produced parts as latent recurrent vectors, experiments demonstrated that the proposed 3D-ConvRNN model yields 97.6% precedence relation validation accuracy, which outperformed a 3D-CNN binary classifier.

Some recent work applied knowledge graph technology to process sequencing and planning. Liang et al. [93] developed a comprehensive knowledge graph representing process design intent integrated with information about machining resources (machines, fixtures) and operations (tool paths, cutting parameters), as well as part features and quality characteristics. An extensive ontology was developed to support this research and was implemented in web ontology language (OWL) and Protégé. Given a CAD model of a part with tolerance and surface finish requirements, an interactive process occurs that results in manufacturing feature annotations of the part geometry. These annotations are then linked to process knowledge captured in the knowledge graph. Reasoning steps occur to develop a detailed process plan suitable for the part. An example demonstrated with their system developed a process plan with separate roughing and finishing operations for hole and pocket features and sequenced the operations accordingly. Another research group integrated DRL with their knowledge graph to develop process plans [94]. This work focused on a knowledge-based process planning method for diesel engine components; much of the knowledge capture and modeling focused on collecting knowledge in the form of case studies. Case study knowledge had a higher priority than more general rule-based knowledge. Their knowledge modeling approach was the concept space–decision space–knowledge space (CDK) model which is sufficient to represent concepts, relationships, and rules to support RL reasoning. Individual elements of the knowledge graph were represented as triplets in an entity-relationship manner. Part geometry and requirements were manually encoded into the knowledge graph. Zhang et al. [95] presented another approach that combined a knowledge graph with RL. They formulated process planning as a problem of sequential recommendation of knowledge and used an actor-critic algorithm to solve the problem. However, they noted that this solution method tended to get trapped in local minima and future work will test different algorithms.

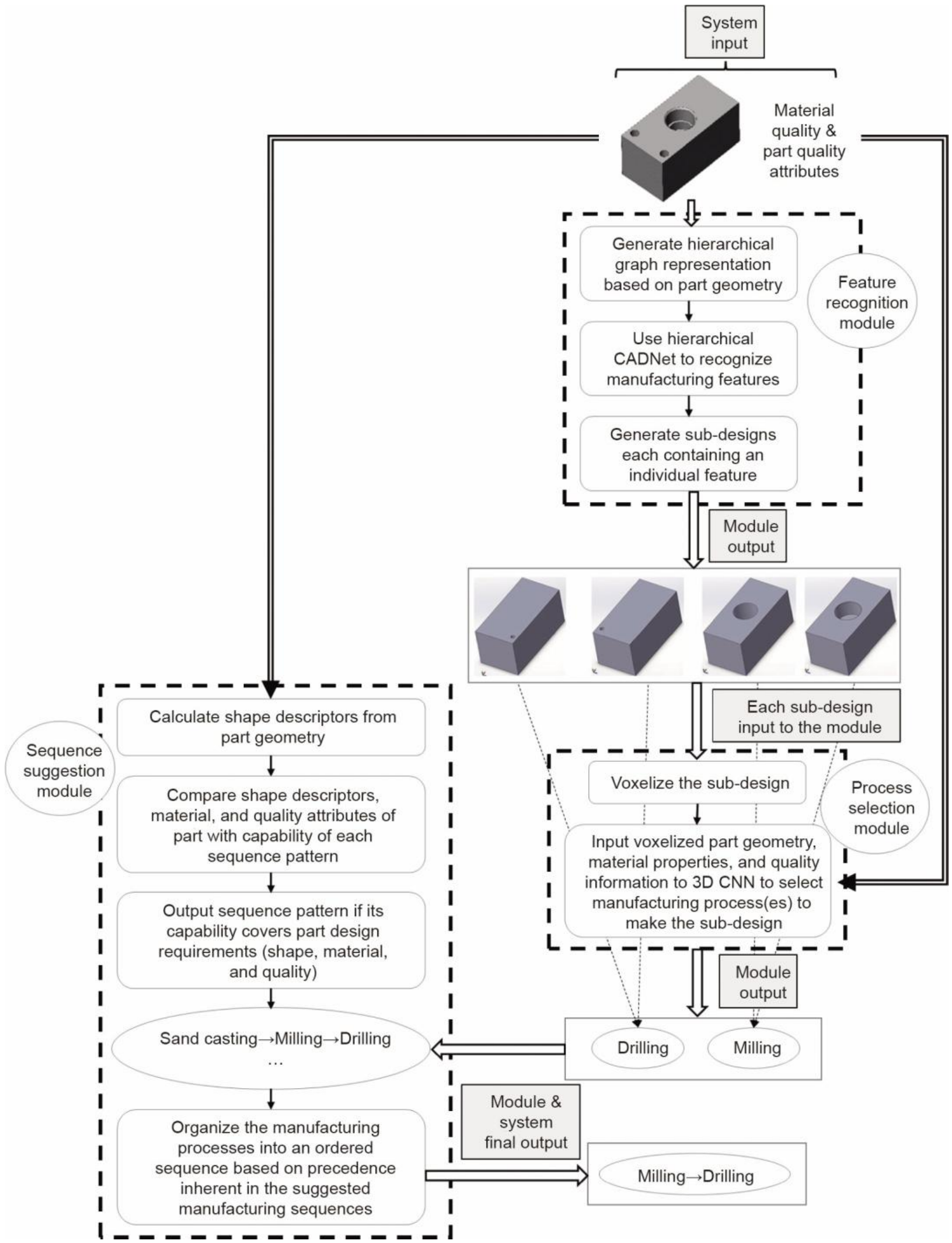


Fig. 8. Generalized workflow of the integrated manufacturing process sequencing framework. Reproduced from [23] with permission.

Literature on operation sequencing is summarized in Table 4 [23,85,87–95], which shows the broad range of approaches taken. Even for the research that automatically recognizes features or removal volumes from geometric part models [23,89–92], no consensus has emerged on superior approaches, given that RL, auto-encoders, GNNs, and CNNs have been investigated. A wide range of shape descriptors or intermediate representations have been explored as well. Separately, the knowledge graph-based research [93–95] resulted in much more detailed process plans than those produced by the other ML approaches, but required much more extensive knowledge modeling. This work would benefit from integration with automatic feature recognition methods so that manual interpretation of part geometry and requirements is no longer needed.

Table 4
Manufacturing operations sequencing literature.

Manufacturing process	ML techniques	Input	Shape descriptors	References
Machining	LSTM	Voxel, tolerances, roughness	FAG	[85]
Machining	DRL	Features, TAD, quality	—	[87]
3-axis milling	ANN	Features, TAD, quality	Geometric metrics	[88]
Machining	DIRL	BRep, quality	AAGs of removal volumes	[89]
Machining	Stacked auto-encoders	CAD model, quality	Knowledge graph	[90]
Machining	GNN, CNN	Tri mesh, material, quality	Sequence patterns	[23,91]
Milling, drilling	3D recurrent CNN	Voxel	—	[92]
Machining	Knowledge graph	CAD model, quality	Relations based on ontology	[93]
Machining	Knowledge graph, DRL	Unknown	Relations based on ontology	[94,95]

4.2. Manufacturing process parameter selection

After operations have been identified and sequenced, process settings for each operation should be determined which is investigated in this section. Hashimoto and Nakamoto [96] proposed a DL model for cutting tool selection and tool path type selection for milling processes. Their input contained voxelated part shape together with a categorical (i.e., high vs low) accuracy requirement at each voxel. This information was put into a 3D CNN to predict, per each voxel, the cutting tool (ball endmill vs flat endmill) and tool-path pattern (contour-line pattern, scanning-line pattern, or along-surface pattern) used to generate the part design. On the other hand, the categorical descriptions of accuracy requirements (e.g., high accuracy vs low accuracy) limit the granularity of the description. Similarly, Komura et al. [97] used a 3D CNN to select machining operations (cutting using a large tool, cutting using a small tool, or electrical discharge machining) to realize a part design represented by voxels. To further explain the selection, they utilized MC dropout to evaluate the level of certainty for the selection per each voxel. On the other hand, only shape information was considered. Deng et al. [98] proposed a GNN for planning machining process parameters based on part AAG representations. Based on the feature geometry and required operation (pocketing or profile contouring), the trained model will predict process parameters including width of cut, depth of cut, and feed rate. However, factors such as material and tolerance requirements were not considered in the approach at its current form. As an extension, they [98] proposed another GNN to predict cutting depth, cutting width, feed rate, and spindle speed for machining operations. Pre-defined constraints regarding cutting parameters and surface roughness were introduced into the loss function to further improve training accuracy.

Regarding additive manufacturing processes, most applications of ML focused on process monitoring and defect detection as reported by several survey articles [100–102]. Some have focused on process parameter setting and manufacturability analysis. Given this purpose and the complex geometry capability of AM processes, inputs to the ML model are desired properties or characteristics, rather than part geometry. Caiazzo and Caggiano [103] proposed an ANN for relating process parameters (i.e., laser power, scanning speed, and powder feed rate) to the width, depth, and height of the deposited metal trace in a laser direct metal deposition process. The ANN was used to identify the process parameters to achieve desired trace geometry.

For PBF processes, Zhang et al. [104] proposed an ANN to model the performance for Ti-6Al-4V material. The ANN was used to map (a) roughness of a spread layer and (b) volume of powder spread per unit time per unit width of spreader as a function of spreader translation and rotation speeds. This mapping function was used inversely to select the spreader parameters for achieving the desired roughness and speed. Similarly, Aoyagi et al. [105] utilized an SVM to map the relationship between process parameters (e.g., scan speed, line offset of the laser path, and layer thickness) and quality of built surface for a CoCr alloy. The mapping was then used to select process parameters to reduce pore density in the fabricated part. Meng and Zhang [106] utilized a GP-based model to learn the relationship between remelted depth and laser power and scan speed process parameters. The relationship was then used to suggest process parameters for 316L and 17–4 PH stainless steel to promote conduction mode melting and reduce keyhole pore formation during printing. Hertlein et al. [107] proposed a Bayesian network model to relate final part properties (e.g., yield strength, ultimate tensile strength, surface roughness, and hardness) to process parameters including laser power, scan speed, hatch spacing, and layer thickness. The model can suggest process parameters for building parts made of 316L stainless steel. Tapia et al. [108] proposed a GP-based surrogate modeling framework to learn and predict melt pool depth of based on process parameters including laser power, scan speed, and size of the laser beam. The model was then used to facilitate process planning by identifying processing windows that lead to desirable thermal conditions for the melt pool. On the other hand, the generalizability of the model was questionable since the model prediction deteriorated when predicting melt pool depth for process parameters given in other literatures. Park et al. [109] proposed a DL-based approach for selecting process parameters using a four-layer ANN to learn the relationship between part quality (i.e., desired density ratio and surface roughness) and process parameters (i.e., laser power, laser scanning speed, layer thickness, hatch distance). Then, based on the learned relationship, a process parameter repository that maps combinations of part quality values to a set of process parameters was built. Finally, based on the input desired part quality, the approach searched through this repository and determined the corresponding set of process parameters to achieve the part quality. The approach is limited to parts made of Ti–6Al–4V, and the computational cost for the repository generation and searching could be high when the method is expanded to include other materials. Dong et al. [110] proposed a 3D CNN with U-net architecture for modeling temperature and residual stress distributions in metal parts represented as voxel models. They investigated a feature-based approach where their training dataset consisted of parts with shape features commonly produced by topology optimization (TopOpt) methods, such as thin walls and struts of various sizes, shapes, and angles. Their model could be used to fine-tune process parameters to reduce distortions caused by residual stresses.

For fused filament fabrication (FFF) material extrusion processes, Mohamed et al. [111] proposed an ANN to relate creep compliance and recoverable compliance of parts to process parameters (e.g., layer thickness, raster angle, and number of contours), and the model was used to optimize process parameters to achieve better viscoelastic properties of the built part. In later work, the researchers investigated the effects of FFF fabrication conditions on the dimensional accuracy of cylindrical parts [112]. Second-order definitive screening design and an ANN were combined to quantify how the dimensional accuracy of length and diameter dimensions were affected by six important manufacturing variables: slice thickness, raster to raster air gap, deposition angle, part print direction, bead width, and number of contours. Jiang et al. [113] utilized an ANN to relate printable bridge length with process parameters including print speed, cooling fan speed, and print temperature for the polylactic acid material.

Research on parameter selection is summarized in Table 5 [96–99,103–113]. ML applications of process planning for machining consider part shape, while approaches for AM processes consider quality requirements, rather than part shape. Presumably this is because AM processes can produce just about any shape, so process parameter settings are not very shape dependent. In contrast, process settings for machining are highly dependent on feature shapes. In terms of ML techniques for parameter selection, a wide variety has been explored. Even in machining, both CNN and GNN approaches have been investigated and yielded good results. The range of ML methods for AM processes is considerably wider, from SVM and GP to ANN and 3D CNN techniques, all of which demonstrated good parameter prediction results.

Table 5
Manufacturing operations sequencing literature.

Manufacturing process	ML techniques	Input	Shape descriptors	References

Milling, drilling	3D CNN	Voxel	—	[96,97]
Machining	GNN	CAD model	AAG	[98,99]
DED	ANN	Desired deposit geometry	Relations between deposit size and parameters	[103]
PBF	ANN	Powder bed roughness, spreading speed	Relations between spreader parameters and bed roughness	[104]
PBF	SVM	Desired deposit quality	Relations between process parameters and quality	[105]
PBF	GP	Desired deposit quality	Relations between process parameters and quality	[106,108]
PBF	Bayesian network	Desired deposit quality	Relations between process parameters and quality	[107]
PBF	ANN	Desired deposit quality	Relations between process parameters and quality	[109]
PBF	3D CNN	Voxel, simulation results	—	[110]
FFF	ANN	Desired compliance	—	[111,112]
FFF	ANN	Desired bridge length	Relations between feature size and process parameters	[113]

DED: directed energy deposition;

5. Manufacturing operation scheduling

With reference to Fig. 1, the focus will shift to Step 4 on supply chains. However, as indicated earlier, the only topic to be covered here is operation scheduling using ML methods. Many papers have addressed operation scheduling across supply chains of various types as indicated by recent review articles [114,115]. However, most of this work focused on issues of combinatorial optimization that are outside the scope of our interests. Two different approaches to the scheduling issue will be covered here, one that uses conventional ML techniques and one that uses RL.

Priore et al. [116] utilized ensemble methods including bagging, boosting, and stacking for selecting dispatching rules for machine selection and operation scheduling in a flexible manufacturing system. The individual classifiers included in the ensemble method were SVM, DT, ANN, and CBR. The factors that affect machine selection and operation sequencing included deadline of the job, number of alternative machines for an operation, mean utilization of the system, utilization of each machine, mean number of production parts in the system, ratio of the utilization of the bottleneck machine to the mean utilization of the entire plant, ratio of the standard deviation of the individual machine utilization to the mean utilization, mean tardiness, and mean flow time.

Interest in RL and DRL in smart manufacturing has exploded in recent years. Review article [115] from 2023 surveyed over 40 papers on the application of DRL in job shop scheduling. They categorized papers into the three areas of order dispatching, job shop flow scheduling, and process rescheduling. Some of these papers directly addressed operation scheduling in a single job shop and some included automated guided vehicle (AGV) routing considerations. Job shops were characterized by random order, small batches, and large variety, which caused challenges in managing manufacturing resources and achieving high production efficiency. For example, Yang and Xu [117] investigated scheduling and reconfiguration of a flow line under dynamic job arrival conditions using DRL. Specifically, they utilized the advantage actor-critic algorithm and showed it performed better than iterated greedy and genetic algorithm approaches. In other work, Jing et al. [118] proposed a combination of RL and graph convolutional network for flexible job shop scheduling. In their approach, the scheduling problem was modeled as the process of topology graph structure prediction and the scheduling strategy was adjusted by predicting the connection probability among edges in the graph. They demonstrated that their novel graph-based multi-agent system approach performed better than traditional DRL and conventional scheduling algorithms. Other recent work utilized DRL for dynamic task scheduling in cloud manufacturing, where the researchers updated the scheduling policy by considering the distribution distance between the pre-training dataset and the training policy [119].

For factory floor, cloud manufacturing, and job shop production problems, RL approaches appear to perform better than more conventional solutions. RL techniques appear to be well suited to operation sequencing problems since RL deals with series data and operation scheduling is about learning how to adapt (that is, learning a policy) to consequences of earlier decisions. It should be noted that the research reviewed here did not use part CAD or geometric models as inputs. Instead, operation types and tasks were the inputs, which were manually determined. Whether these approaches can be integrated with upstream applications, including process selection and process planning as covered in Sections 4.1 and 4.2, to automatically determine operation types and tasks remains an open question.

6. Generative design for manufacturability

In the context of cyber manufacturing, the generation of designs must go beyond functionality and consider manufacturability. It is essential to prioritize the automatic generation of manufacturable designs to meet practical application requirements. In this section, three approaches to combining generative design with design for manufacturability will be presented. First, an approach is presented that embeds TopOpt within a GAN model, while the second utilized an image modification approach to modify part representations to improve manufacturability. In the third approach, the image modification method is extended to 3D and combined with design generation to achieve both functional and manufacturing objectives. The second and third approaches are from our research group and will be presented in some detail.

6.1. Embed GAN in TopOpt

One approach to automatically generate manufacturable designs utilizes a dataset of manufacturable designs to train a GAN. Greminger [120] proposed an extension of a multi-scale gradient structure GAN (MSG-GAN) [121] to generate 3D manufacturable designs in a TopOpt problem formulation. This was realized by restricting the training dataset to designs that are known to be manufacturable. To generate enough manufacturable designs, a scripting approach utilizing CadQuery was applied to generate voxel synthetic training datasets for parts that could be manufactured on a 3-axis computer numerically controlled (CNC) milling machine. In this approach, the trained GAN was embedded into the TopOpt algorithm by using the latent feature vector (input to the GAN) as the design variable, rather than element densities. Element densities were then generated by the GAN. Though this work can generate manufacturable designs, limitations included the need for 4 different voxel models per part (each part was represented using four voxel scales, 4, 8, 16, and 32 voxels per direction) and that volume fraction constraints could not be applied directly. One consequence of the first limitation is that training with 20 000 parts models required 23 days of computer time.

6.2. DFM via image modification

Previous work generated manufacturable designs by restricting the training dataset to be manufacturable. However, existing designs could not be modified to improve their manufacturability corresponding to Step 3 of our cyber manufacturing scenario (Fig. 1). Wang et al. [122] proposed the Manufacturable conditional GAN (McGAN) to modify existing unmanufacturable part designs to make them manufacturable, as shown in Fig. 9. The strategy was to identify manufacturing features in the part, then modify their shape to ensure consistency with manufacturability rules for a given process (injection molding in this case). This strategy was implemented for 2D part representations (images) by first carrying out instance segmentation using Mask R-CNN to segment complex designs into multiple simple designs consisting of individual features [123]. After a resizing scheme, multiple conditional GANs, called Pix2Pix, were applied to modify these segmented designs to make them manufacturable [124]. Eventually, the resizing and recombination schemes were utilized to merge these modified simpler but manufacturable designs into the original complex designs. As illustrated in Fig. 9 [122], McGAN was applied to 2D representations of injection molded parts that represented product housing components. Part designs consisted of 3 or 5 vertical walls and ribs on a horizontal wall. Design changes included rounding sharp corners, adding draft angles, and adjusting feature aspect ratios to ensure injection moldability. The results of McGAN were promising as modified designs were of high quality from the viewpoint of manufacturability [125].

Although McGAN worked well on a wide range of part designs consisting of 3 or 5 vertical walls, it has some limitations. This image modification approach was performed on a per-feature basis, with separate Pix2Pix GANs trained for each type of feature. Each new feature type requires significant efforts to generate training datasets. Feature interactions, other than vertical-to-horizontal wall interactions, were not considered. Successful modifications require successful segmentation into individual feature pairs, which may be difficult if complex feature interactions exist. Hence, generality and scalability to additional design rules and manufacturing processes remain promising, but unproven. Also, extension to 3D part representations is essential; this will be presented next.

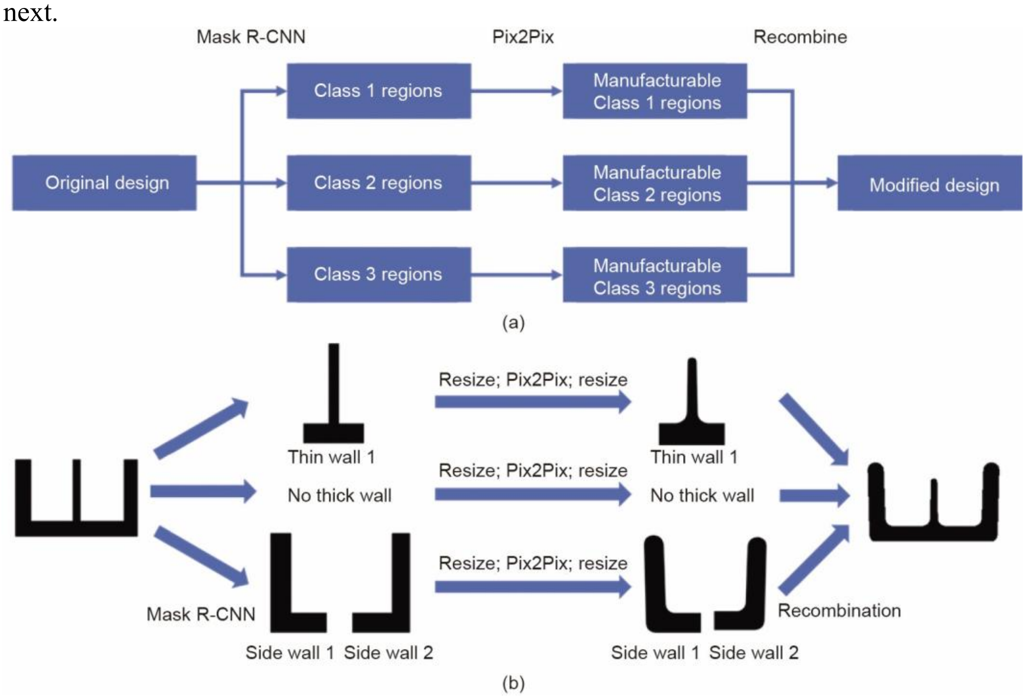


Fig. 9. The framework of McGAN: (a) framework of McGAN, and (b) application of McGAN for injection molding. Reproduced from [122] with permission.

6.3. Design for functionality and manufacturing

An initial study was conducted on the integration of functional design, using a generate-and-test approach, and DFM using generative design techniques [125]. The study utilized 3D voxel models of parts and represented an extension from 2D to 3D of the McGAN DFM method presented in Section 5.2.

A specific example was developed to demonstrate the proposed design method of a product housing component consisting of two internal ribs and a hole surrounded by an internal wall as shown in Fig. 10. The functional design problem is to maximize stiffness subject to a distributed load on the bottom surface, a fixed boundary condition around the tops of the vertical walls, and a volume fraction constraint, similar to a typical TopOpt problem. In this case, however, design variables were feature sizes and locations. The part design domain was discretized into a 128 × 128 × 32 voxels space. This discretized design space yielded a total of 107 000 000 possible designs.

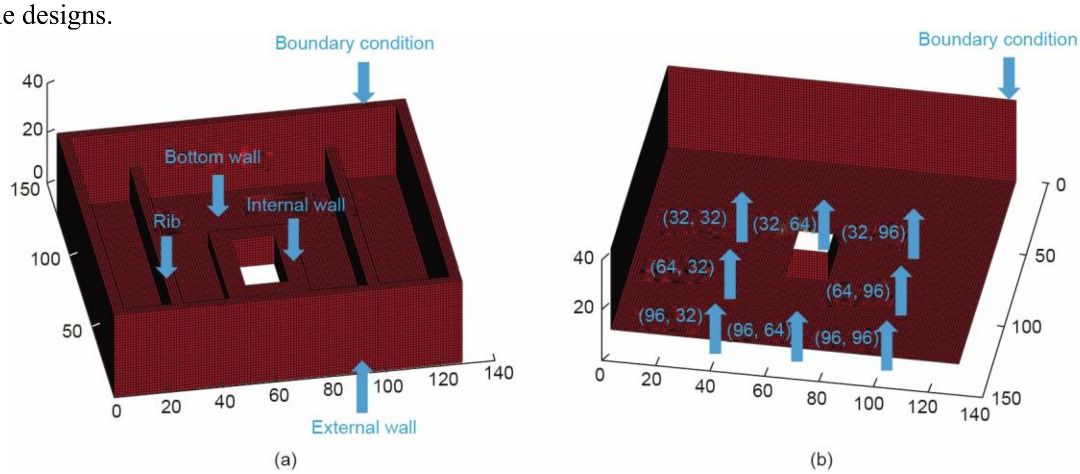


Fig. 10. 3D housing part example problem: (a) the composition of 3D housing parts, and (b) boundary and load conditions of 3D housing parts.

For functional design, a variational auto encoder (VAE) was constructed and trained on a sampling of 10 000 part designs, from the design space of 107 M designs. After considerable experimentation, a VAE architecture was established, where the encoder consisted of five 3D convolution layers, resulting in $1 \times 1 \times 256$ output representing 128 mean values and 128 standard deviations, and the decoder had 6 transpose convolution layers, with LeakyReLu activation functions used throughout.

After design generation, an evaluation and selection process was used to identify designs that best met requirements. Specifically, compliance was evaluated by a straightforward 3D structural finite element analysis and volume fraction was computed by taking the fraction of solid voxels in the design domain. From all generated designs, the Pareto front was identified and the top 100 designs were selected for subsequent DFM. Each was modified using the 3D cGAN counterpart of the 2D McGAN presented in Section 6.2. Three 3D Vox2Vox GANs were trained to modify features, analogous to the Pix2Pix GANs for rounded corners, draft angles, and feature aspect ratios.

The functional design VAE was integrated with selection and DFM to automatically and iteratively generate, evaluate, and modify designs until designs emerged that satisfied both functional and manufacturability requirements. To determine when the iterative generation-to-DFM process converged the differences between DFM inputs and outputs were quantified. Specifically, the number of voxels that DFM modified was counted and compared with the total number of voxels in the design space. For the example in Fig. 10, a total of 11.4% of the voxels was modified in the first DFM iteration. Subsequently, 2.97% and 2.06% of voxels were modified in iterations 2 and 3, respectively. After 3 iterations, the percentage of modified voxels was deemed low enough to consider the iterative process converged. A sampling of four resulting part designs is shown in Fig. 11, one design in each row. The left column represents inputs to the DFM cGAN, while the middle and right columns show output designs from the top and bottom views. Note that all output designs have rounded corners and draft angles on the vertical walls and rib features. The ribs and internal walls satisfy aspect ratio limits. Also of note are the slots that were added to the undersides of the thick rib features in Designs 1 and 4 (see in the bottom views in the right column). In injection molding, such slots are often added under thick features to achieve uniform wall thicknesses throughout a part. Such consideration was built into the training dataset for the aspect ratio Vox2Vox model.

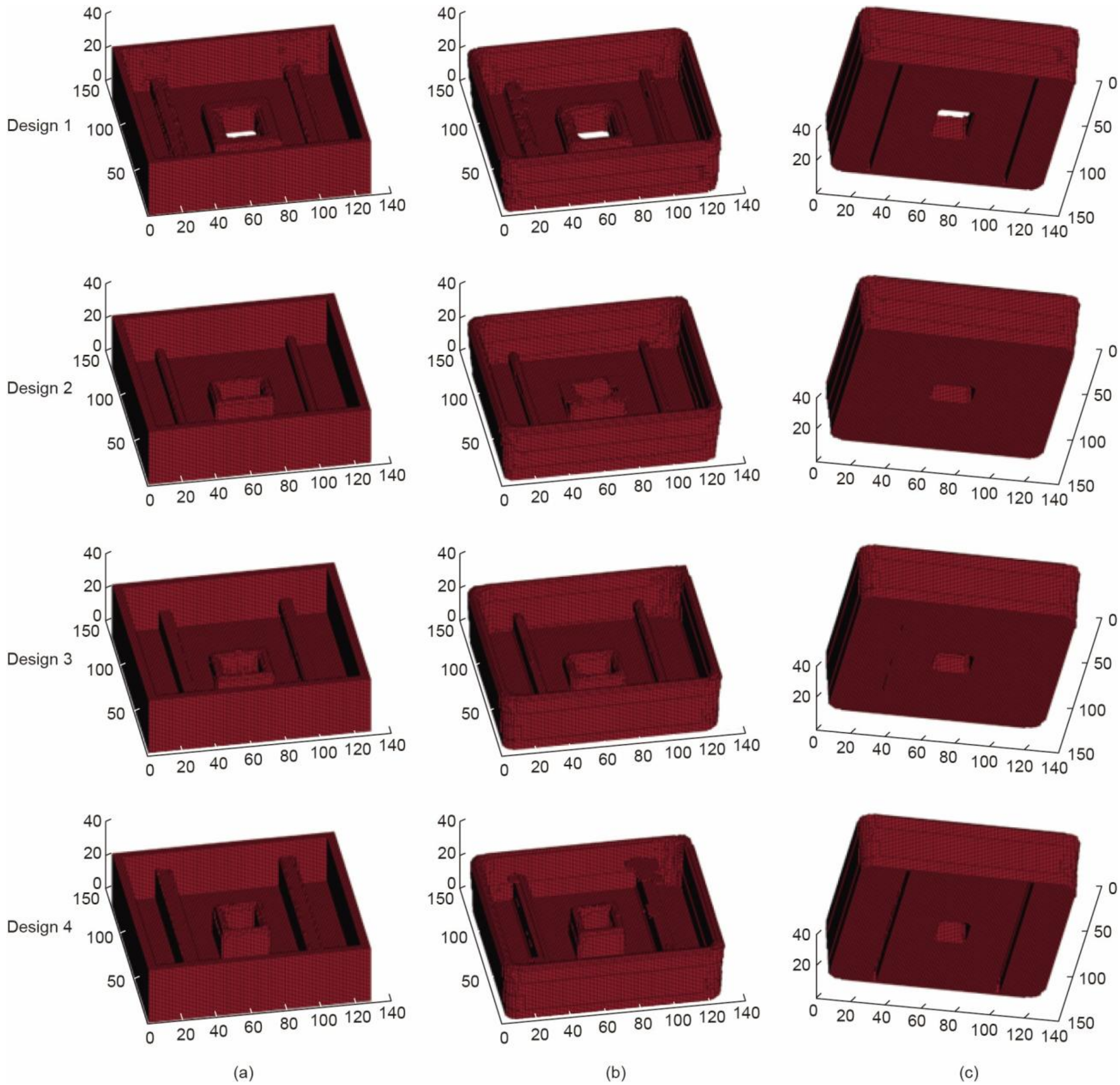


Fig. 11. Sampling of results of the iterative functional design and DFM process for the design problem: (a) input to DFM, (b) output of DFM view (−10, 65), and (c) output of DFM view (−10, −65).

7. Research issues

Based on the literature review throughout this paper, a number of research issues has been identified. We will start with a consideration of issues related to each major topic covered in Sections 3–6, then address larger issues around process planning and DFM.

7.1. Manufacturing process classification and process planning

Good ML based manufacturing process classification results have been achieved, upwards of 98% accuracy, with several different approaches and with different part representations, including triangle meshes and point clouds. It is unclear if more informationally complete representations, such as BReps, can lead to even better accuracy. Also, a move away from strictly CNN-based ML models to hybrid graph and convolutional networks, which have achieved very good results in other domains, may also help improve accuracy.

BReps and GNNs have proven to yield very high accuracies in machining feature recognition [54], including instance segmentation of feature faces. However, considerable advances are required to support process planning and operation sequencing. Set-ups and fixturing need to be determined and cutting tools and toolpaths are needed. It is not clear if more complete process planning can be achieved by simply composing workflows from existing tools or if additional research is needed.

Decent progress has been made in operation sequencing problems related to machining processes. It is interesting to see the application of RL and other time-series-based methods, such as LSTM and recurrent learning techniques, being applied for sequencing problems. Knowledge graph techniques are frequently utilized for more detailed tool selection and operation sequencing. However, little progress seems to have been made on part manufacturing that requires significantly different manufacturing processes, such as die casting followed by machining. Also, to date, limited identification and sequencing of interacting features has been achieved. It is not clear if current approaches will be sufficient for the most complex configurations of interacting features in machining or other combinations of processes.

Very good success has been achieved in using ML models as surrogate models for process parameter selection for a variety of processes. Many researchers have used traditional ML methods, such as SVM, GP regression, and Bayesian networks, to model relationships between process variables and part or material properties [126]. CNNs have become prevalent for modeling relationships between spatially distributed characteristics or properties and process variables. For example, color maps in 2D or 3D are often used to represent temperature or mechanical property distributions; they are computed using process simulations given certain process settings. By processing many of these maps using a CNN, relationships between process variables and property distributions can be learned effectively. Issues remain, however, in increasing the accuracy of the learned models and the efficiency with which they are learned. New CNN, GNN, and hybrid architectures are being explored. An open issue is the type of neural network to use and, related to that, the type of input models. The emerging GNN and hybrid GNN-CNN approaches used BRep and hybrid BRep-point cloud representations and achieved excellent accuracies in feature recognition. These representations contain much more information than point clouds and mesh models so it makes sense that the methods that use them perform well. Voxel models contain considerable unneeded information (all the voxels that do not define a surface); since they are very memory intensive their resolutions must be kept low for computational reasons. Even so, they can work well due to the maturity and effectiveness of 3D CNN methods. Perhaps a larger issue is the generality and scalability of these ML models to a wider range of process conditions, coupled relationships among several properties, other materials, other manufacturing equipment, etc.

Manufacturing operation scheduling is often treated as a problem of operations management and applied across supply chains. For our purposes in a cyber manufacturing setting, the problem is more circumscribed, reflecting a perspective of job shop scheduling. Many traditional ML techniques have been applied for machine selection and flexible manufacturing system operations. Recently, many research groups applied RL techniques for these scheduling problems, taking advantage of the unsupervised learning property of RL and its iterative approach to learning decision policies. What is needed is integration with upstream tools such as process selection and process planning for more complete cyber manufacturing workflows.

Throughout all this research, an issue repeated by many researchers is the lack of appropriate training datasets. Many groups developed their own synthetic datasets then applied them to train and test their ML models with considerations of data quality, diversity, and balance [126]. Some groups leveraged existing datasets such as Model40, MFCAD, and MFCAD++, but often existing datasets do not include all of the labels necessary for an application, so researchers had to annotate them. Of course, this highlights a major advantage of the unsupervised or semi-supervised learning approaches that some researchers took. As the cyber manufacturing field matures, the research community should focus on developing and adopting some standard datasets to enable direct comparisons of model performance.

7.2. Design for manufacturability

Limited progress has been made in applying ML technology to DFM, as indicated by the small number of research efforts reviewed here (3, with 2 from our group). On the other hand, tremendous progress has been made in generative AI and generative design, so it is likely that additional works will be reported in the near future. In the generative engineering design area, many papers have been published on the integration of generative ML and TopOpt. Various methods for integrating ML into TopOpt [120] or TopOpt into ML [128] have been proposed. AutoDesk has offered their generative design capability commercially for many years, which appears to be TopOpt based, but where they can generate many variants. Commercial offerings have started to incorporate some manufacturing constraints into TopOpt or generative design tools, such as ensuring 3-axis machinability or avoiding overhangs in additive manufacturing.

In our cyber manufacturing scenario from Fig. 1, the need for DFM arises because it is unlikely that designers would have added manufacturing features and details before the manufacturing process was selected. The part included in Fig. 1 is a good example of the potential significant design changes during DFM. For such design changes, the designer must ensure that functional requirements are met, as well as reducing manufacturing challenges, so a DFM tool cannot simply make manufacturability changes and assume that the design process is complete. The research generative design systems required extensive training datasets and significant training time yet have quite limited scope. Several major research issues are evident. First, good computational manufacturability analysis tools are needed across many manufacturing processes. These should be used to generate training datasets and potentially during usage of a cyber manufacturing system. Second, significant efficiency improvements are needed for generative DFM; current generative design tools typically operate in a generate-and-test manner, typically generating and discarding many infeasible or inferior designs [122,129,130]. Third, DFM capabilities need to be extended from consideration of a single manufacturing process to process sequences and process chains. Fourth, a much wider variety of engineering applications and physical phenomena should be addressed by generative design, since most work focuses only on structural applications.

7.3. Generative and variant process planning

Finally, a few thoughts on process planning are warranted, from both generative and variant perspectives. An underlying assumption regarding process planning is that if the capabilities (shapes, sizes, properties, etc.) of a manufacturing process can be modeled, then generative process planning should be achievable. At present, ML models of process capabilities have demonstrated very good performance on several aspects of variant process planning. These models should be integrated to provide additional capabilities across cyber manufacturing workflows. Even so, scaling to high levels of part complexity with many interacting features and complex tolerance chains remains an unsolved problem. Will ML models created to interpolate throughout their training spaces be able to extrapolate to unseen types of part designs?

Another question: with the use of ML models, will generative and variant process planning converge such that no distinguishable difference remains between the two approaches? Given a part design, a generated process plan (generative process planning) was likely produced by a ML model that was trained with many similar parts, so perhaps internally the process planner was operating in a variant manner, generating a slight variant of a process plan that it was trained on. Although this may be a philosophical issue, the research community may benefit from a perspective that a continuum exists between the generative and variant extremes and that the best approach is an appropriate combination of the two.

When the entirety of the cyber manufacturing scenario in Fig. 1 is considered, it is clear that much research remains. Excellent progress has been made on several of the steps illustrated, but each study had limitations. To make significant progress, we suggest that more complete design specifications should be provided as the input to Step 1 in Fig. 1: geometric part model, geometric tolerances, surface finish specifications, and mechanical properties. Tools to aid subsequent steps should be able to reason with all of these types of information. Iterations between steps should occur so that decisions can be updated with more complete information from downstream decisions and analyses. During DFM, rather than simply generating many designs, each should be evaluated, so that designers can gain a holistic understanding of trade-offs across the entire manufacturing process chain.

8. Conclusion

This paper provides an integral review of state-of-the-art AI and ML-based digital manufacturing frameworks for cyber manufacturing. We addressed specifically a cyber manufacturing scenario that starts with a preliminary part design and progresses through the steps of manufacturing process selection, process planning, DFM, and supplier selection, with a focus on the first three of those steps. While it is evident from the review that significant progress has been reported in manufacturing process selection, process sequencing, and process planning, significant challenges remain in establishing a ubiquitous design-to-manufacturing pipeline. Some convergence has been achieved on establishing suitable part representations for some of the steps or sub-steps, such as BRep models being suitable for manufacturing feature recognition. On the other hand, good manufacturing process selection results have been achieved with part representations ranging from point clouds to BReps, so additional research is needed for convergence. Regarding AI and ML techniques, some specific techniques appear to be appropriate for some decisions, such as RL being effective for manufacturing operation scheduling, or CNNs being effective at processing spatial distributions of part properties for process parameter selection.

Challenges and research issues were discussed in the previous section. Progress across the cyber manufacturing workflow remains uneven, with most results of limited scope. The research community will benefit from more holistic formulations of the research challenges and approaches to address them. Recent progress in developing foundation models in AI enables consideration of cyber manufacturing from a fundamental, first principles basis since these foundation models attempt to capture the basis for general intelligence. Such consideration could provide a step change in automated cyber manufacturing capabilities.

Acknowledgements

The authors gratefully acknowledge funding from the US National Science Foundation Future Manufacturing Research Grant (2229260). DWR gratefully acknowledges funding from the Singapore RIE2025 Manufacturing, Trade and Connectivity (MTC) Programmatic Fund (M24N3b0028).

References

[1] Smith CS, Wright PK. CyberCut: a world wide web based design-to-fabrication tool. J Manuf Syst 1996;15(6):432–42.

[2] Lee J, Bagheri B, Jin C. Introduction to cyber manufacturing. Manuf Lett 2016;8:11–5.

[3] Zhou J, Li P, Zhou Y, Wang B, Zang J, Meng L. Toward new-generation intelligent manufacturing. Engineering 2018;4(1):11–20.

[4] Chen J, Hu P, Zhou H, Yang J, Xie J, Jiang Y, et al. Toward intelligent machine tool. Engineering 2019;5(4):679–90.

[5] Zhou G, Zhang C, Li Z, Ding K, Wang C. Knowledge-driven digital twin manufacturing cell towards intelligent manufacturing. Int J Prod Res 2020;58(4):1034–51.

[6] Grossmann A, Weis P, Clemen C., Mittelstedt C. Optimization and re-design of a metallic riveting tool for additive manufacturing—A case study. Additive Manufacturing 2020; 31:100892.

[7] Gupta SK, Nau DS. Systematic approach to analysing the manufacturability of machined parts. Comput-Aided Des 1995;27(5):323–42.

[8] Raman R, Marefat MM. Integrated process planning using tool/process capabilities and heuristic search. J Intell Manuf 2004;15:141–74.

[9] Kim YS. Recognition of form features using convex decomposition. Comput-Aided Des 1992;24(9):461–76.

[10] Joshi S, Chang TC. Feature extraction and feature based design approaches in the development of design interface for process planning. J Intell Manuf 1990;1:1–15.

[11] Vandenbrande JH, Requicha AA. Geometric computation for the recognition of spatially interacting machining features. In: Shah JJ, Mäntylä M, Nau DS, editors. Manufacturing research and technology. Amsterdam: Elsevier; 1994. p. 83–106.

[12] Tan W, Khoshnevis B. Integration of process planning and scheduling—a review. J Intell Manuf 2000;11:51–63.

[13] Bralla JG. Design for Manufacturability Handbook. 2nd ed. New York City: McGraw-Hill; 1999.

[14] Rosen DW. Thoughts on design for intelligent manufacturing. Engineering 2019;5(4):609–14.

[15] Goodfellow I, Bengio Y, Courville A. Deep Learning. Cambridge: MIT Press; 2016.

[16] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521(7553):436–44.

[17] Wu Z, Pan S, Chen F, Long G, Zhang C, Yu PS. A comprehensive survey on graph neural networks. IEEE Trans Neural Netw Learn Syst 2021;32(1):4–24.

[18] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial nets. In: Proceedings of the 28th International Conference on Neural Information Processing Systems; 2014 Dec 8–13; Montreal, Canada. Cambridge: MIT Press; 2014. p. 2672–80.

[19] Sherstinsky A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D 2020;404:132306.

[20] Sutton RS, Barto AG. Reinforcement learning: an introduction. 2nd ed. Cambridge: MIT Press; 2018.

[21] Mnih V, Kavukcuoglu K, Silver D, Graves A, Antonoglou I, Wierstra D, et al. Playing Atari with deep reinforcement learning. 2013. arXiv:1312.5602.

[22] Sutton RS, McAllester D, Singh S, Mansour Y. Policy gradient methods for reinforcement learning with function approximation. In: Solla S, Leen T, Muller K, editors. Advances in neural information processing systems. Cambridge: MIT press; 1999. p. 1057–63.

[23] Zhao C, Dinar M, Melkote SN. Deep learning and sequence mining for manufacturing process and sequence selection. Int J Prod Res 2024;62(14):5293–314.

[24] Ip CY, Regli WC, Sieger L, Shokoufandeh A. Automated learning of model classifications. In: Proceedings of the eighth ACM symposium on Solid modeling and applications; 2003 Jun 16–20; Washington, DC, USA. New York City: Association for Computing Machinery; 2003. p. 322–7.

[25] Ip CY, Regli WC. A 3D object classifier for discriminating manufacturing processes. Comput Graph 2006;30(6):903–16.

[26] Hoefer MJ, Frank MC. Automated manufacturing process selection during conceptual design. J Mech Des 2018;140(3):031701.

[27] Zhao C, Dinar M, Melkote SN. Automated classification of manufacturing process capability utilizing part shape, material, and quality attributes. J Comput Inf Sci Eng 2020;20(2):021011.

[28] Wu D, Jennings C, Terpenny J, Gao RX, Kumara S. A comparative study on machine learning algorithms for smart manufacturing: tool wear prediction using random forests. J Manuf Sci Eng 2017;139(7):071018.

[29] Angrish A, Craver B, Starly B. “FabSearch”: a 3D CAD model-based search engine for sourcing manufacturing services. J Comput Inf Sci Eng 2019;19(4):041006.

[30] Ghadai S, Balu A, Sarkar S, Krishnamurthy A. Learning localized features in 3D CAD models for manufacturability analysis of drilled holes. Comput Aided Geom Des 2018;62:263–75.

[31] Rønsch GØ, Kulahci M, Dybdahl M. An investigation of the utilisation of different data sources in manufacturing with application in injection moulding. Int J Prod Res 2021;59(16):4851–68.

[32] Guo L, Yan F, Li T, Yang T, Lu Y. An automatic method for constructing machining process knowledge base from knowledge graph. Robot Comput-Integr Manuf 2022;73:102222.

[33] Han Z, Huang R, Huang B, Jiang J, Li X. Data-driven and knowledge-guided approach for NC machining process planning. Comput Aided Des 2023;162:103562.

[34] Kumar A, Starly B. “FabNER”: information extraction from manufacturing process science domain literature using named entity recognition. J Intell Manuf 2022;33(8):2393–407.

[35] Zhang Y, Yang S, Dong G, Zhao YF. Predictive manufacturability assessment system for laser powder bed fusion based on a hybrid machine learning model. Addit Manuf 2021;41:101946.

[36] Barrionuevo GO, Ramos-Grez JA, Walczak M, Betancourt CA. Comparative evaluation of supervised machine learning algorithms in the prediction of the relative density of 316L stainless steel fabricated by selective laser melting. Int J Adv Manuf Technol 2021;113(1):419–33.

[37] Jian C, Li M, Qiu K, Zhang M. An improved NBA-based STEP design intention feature recognition. Future Gener Comput Syst 2018;88:357–62.

[38] Zhang Y, Zhang Y, He K, Li D, Xu X, Gong Y. Intelligent feature recognition for STEP-NC-compliant manufacturing based on artificial bee colony algorithm and back propagation neural network. J Manuf Syst 2022;62:792–9.

[39] Yao X, Wang D, Yu T, Luan C, Fu J. A machining feature recognition approach based on hierarchical neural network for multi-feature point cloud models. J Intell Manuf 2023;34(6):2599–610.

[40] Zhang H, Wang W, Zhang S, Wang Z, Zhang Y, Zhou J, et al. Point cloud self-supervised learning for machining feature recognition. J Manuf Syst 2024;77:78–95.

[41] Zhang Z, Jaiswal P, Rai R. FeatureNet: machining feature recognition based on 3D convolution neural network. Comput-Aided Des 2018;101:12–22.

[42] Fu X, Peddireddy D, Aggarwal V, Jun MBG. Improved dexel representation: a 3-D CNN geometry descriptor for manufacturing CAD. IEEE Trans Industr Inform 2021;18(9):5882–92.

[43] Zhang C, Zhou G. A view-based 3D CAD model reuse framework enabling product lifecycle reuse. Adv Eng Softw 2019;127:82–9.

[44] Shi P, Qi Q, Qin Y, Scott,PJ, Jiang X. A novel learning-based feature recognition method using multiple sectional view representation. J Intell Manuf 2020;31:1291–309.

[45] Ning F, Shi Y, Cai M, Xu W. Part machining feature recognition based on a deep learning method. J Intell Manuf 2023;34:809–21.

[46] Zhou G, Yang X, Zhang C, Li Z, Xiao Z. Deep learning enabled cutting tool selection for special-shaped machining features of complex products. Adv Eng Softw 2019;133:1–11.

[47] Vatandoust F, Yan X, Rosen D, Melkote SN. Manufacturing feature recognition with a sparse voxel-based convolutional neural network. J Comput Inf Sci Eng 2024;25(3):031002.

[48] Cao W, Robinson T, Hua Y, Boussuge F, Colligan AR, Pan W. Graph representation of 3D CAD models for machining feature recognition with deep learning. In: Proceedings of the ASME 2020 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference; 2020 Aug 17–19; Virtual, Online. New York City: ASME; 2020. p. DETC2020-22355, V11AT11A003.

[49] Colligan AR, Robinson TT, Nolan DC, Hua Y, Cao W. Hierarchical CADNet: learning from B-reps for machining feature recognition. Comput-Aided Des 2022;147:103226.

[50] Jayaraman PK, Sanghi A, Lambourne JG, Willis KD, Davies T, Shayani H, et al. UV-net: learning from boundary representations. In: Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2021 Jun 20–25; Nashville, TN, USA. Piscataway: IEEE; 2021. p. 11698–707.

[51] Lambourne JG, Willis KD, Jayaraman PK, Sanghi A, Meltzer P, Shayani H. BRepNet: a topological message passing system for solid models. In: Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2021 Jun 20–25; Nashville, TN, USA. 2021. p. 12773–82.

[52] Wang P, Yang WA, You Y. A hybrid learning framework for manufacturing feature recognition using graph neural networks. J Manuf Process 2023;85:387–404.

[53] Lee J, Yeo C, Cheon SU, Park JH, Mun D. BRepGAT: graph neural network to segment machining feature faces in a B-rep model. J Comput Des Eng 2023;10(6):2384–400.

[54] Wu H, Lei R, Peng Y, Gao L. AAGNet: a graph neural network towards multi-task machining feature recognition. Robotics Comput-Integr Manuf 2024;86:102661.

[55] Xia M, Zhao X, Hu X. Machining feature and topological relationship recognition based on a multi-task graph neural network. Adv Eng Inform 2024;62(Pt B):102721.

[56] Ma L, Yang J. Adaptive recognition of machining features in sheet metal parts based on a graph class-incremental learning strategy. Sci Rep 2024;14(1):10656.

[57] Qi CR, Yi L, Su H, Guibas LJ. Pointnet++: deep hierarchical feature learning on point sets in a metric space. In: Proceedings of the 31st International Conference on Neural Information Processing Systems; 2017 Dec 4–9; Long Beach, CA, USA. New York City: Curran Associates Inc.; 2017. p. 5105–14.

[58] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016 Jun 27–30; Las Vegas, NV, USA. Piscataway: IEEE; 2016. p. 770–8.

[59] Graham B, Engelcke M, van der Maaten L. 3D semantic segmentation with submanifold sparse convolutional networks. In: Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA. Piscataway: IEEE; 2018. p. 9224–32.

[60] D'Souza RM, Sequin C, Wright PK. Automated tool sequence selection for 3-axis machining of free-form pockets. Comput-Aided Des 2004;36(7):595–605.

[61] Sormaz DN, Khoshnevis B. Modeling of manufacturing feature interactions for automated process planning. J Manuf Syst 2000;19(1):28–45.

[62] Deja M, Siemiatkowski MS. Machining process sequencing and machine assignment in generative feature-based CAPP for mill-turn parts. J Manuf Syst 2018;48(Pt A):49–62.

[63] Ndip-Agbor E, Cao J, Ehmann,K. Towards smart manufacturing process selection in Cyber-Physical Systems. Manuf Lett 2018;17:1–5.

[64] Shen W, Hu T, Zhang C, Ye Y, Li Z. A welding task data model for intelligent process planning of robotic welding. Robotics Comput-Integr Manuf 2020;64:101934.

[65] Hamouche E, Loukaides EG. Classification and selection of sheet forming processes with machine learning. Int J Comput Integr Manuf 2018;31(9):921–32.

[66] Peddireddy D, Fu X, Shankar A, Wang H, Joung BG, Aggarwal V, et al. Identifying manufacturability and machining processes using deep 3D convolutional networks. J Manuf Process 2021;64:1336–48.

[67] Wang Z, Rosen D. Manufacturing process classification based on heat kernel signature and convolutional neural networks. J Intell Manuf 2023;34:3389–411.

[68] Wang Z, Rosen D. Manufacturing process classification based on distance rotationally invariant convolutions. J Comput Inf Sci Eng 2023;23(5):051004.

[69] Sun J, Ovsjanikov M, Guibas L. A concise and provably informative multi-scale signature based on heat diffusion. Comput Graph Forum 2009;28(5):1383–92.

[70] Liu Y, Fan B, Xiang S, Pan C. Relation-shape convolutional neural network for point cloud analysis. In: Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2019 Jun 15–20; Long Beach, CA, USA. Piscataway: IEEE; 2019. p. 8887–96.

[71] Liu Y, Fan B, Meng G, Lu J, Xiang S, Pan C. DensePoint: learning densely contextual representation for efficient point cloud processing. In: Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV); 2019 Oct 27–Nov 2; Seoul, Republic of Korean. Piscataway: IEEE; 2019. p. 5238–47.

[72] Wang Y, Sun Y, Liu Z, Sarma SE, Bronstein MM, Solomon JM. Dynamic graph CNN for learning on point clouds. ACM Trans Graph 2019;38(5):1–12.

[73] Yan X, Melkote S. Generative modeling of the shape transformation capability of machining processes. Manuf Lett 2022;33:794–801.

[74] Yan X, Melkote S. Automated manufacturability analysis and machining process selection using deep generative model and Siamese neural networks. J Manuf Syst 2023;67:57–67.

[75] Zhao C, Melkote SN. Learning the part shape and part quality generation capabilities of machining and finishing processes using a neural network model. In: Proceedings of the ASME 2022 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference; 2022 Aug 14–17; St. Louis, MO, USA. New York City: ASME; 2022. p. DETC2022-91115, V002T02A010.

[76] Zhao C, Melkote SN. Learning the manufacturing capabilities of machining and finishing processes using a deep neural network model. J Intell Manuf 2023;35:1845–65.

[77] Zhang Z, Hua BS, Yeung SK. RIConv++: effective rotation invariant convolutions for 3D point clouds deep learning. Int J Comput Vis 2022;130(5):1228–43.

[78] Liu X, Wang Z, Melkote SN, Rosen DW. CAD model point cloud semantic segmentation for identification of multiple manufacturing processes. 2022. arXiv:2207.01218v2.

[79] Hussong M, Ruediger-Flore P, Klar M, Kloft M, Aurich JC. Selection of manufacturing processes using graph neural networks. J Manuf Syst 2025;80:176–93.

[80] Sharp M, Ak R, Hedberg T. A survey of the advancing use and development of machine learning in smart manufacturing. J Manuf Syst 2018;48(Part C):170–9.

[81] Sharma A, Zhang Z, Rai R. The interpretive model of manufacturing: a theoretical framework and research agenda for machine learning in manufacturing. Int J Prod Res 2021;59(16):4960–94.

[82] Jing X, Zhu Y, Liu J, Zhou H, Zhao P, Liu X, et al. Intelligent generation method of 3D machining process based on process knowledge. Int J Comput Integr Manuf 2022;33(1):38–61.

[83] Liu Q, Li X, Gao L, Li Y. A modified genetic algorithm with new encoding and decoding methods for integrated process planning and scheduling problem. IEEE Trans Cybern 2021;51(9):4429–38.

[84] Xiao Y, Zheng S, Shi J, Du X, Hong J. Knowledge graph-based manufacturing process planning: a state-of-the-art review. J Manuf Syst 2023;70:417–35.

[85] Zhang Y, Zhang S, Huang R, Huang B, Yang L, Liang J. A deep learning-based approach for machining process route generation. Int J Adv Manuf Technol 2021;115(11):3493–511.

[86] Huang R, Zhang S, Bai X, Xu C. Multi-level structuralized model-based definition model based on machining features for manufacturing reuse of mechanical parts. Int J Adv Manuf Technol 2014;75(5):1035–48.

[87] Wu W, Huang Z, Zeng J, Fan K. A fast decision-making method for process planning with dynamic machining resources via deep reinforcement learning. J Manuf Syst 2021;58:392–411.

[88] Zhang Y, Yu X, Sun J, Zhang Y, Xu X, Gong Y. Intelligent STEP-NC-compliant setup planning method. J Manuf Syst 2022;62:62–75.

[89] Sugisawa Y, Takasugi K, Asakawa N. Machining sequence learning via inverse reinforcement learning. Precis Eng 2022;73:477–87.

[90] Zhang Y, Zhang S, Huang R, Huang B, Liang J, Zhang H, et al. Combining deep learning with knowledge graph for macro process planning. Comput Ind 2022;140:103668.

[91] Zhao C, Dinar M, Melkote SN. A data-driven framework for learning the capability of manufacturing process sequences. J Manuf Syst 2022;64:68–80.

[92] Yan X, Wang Z, Rosen DW, Melkote SN. Learning precedence relations for manufacturing operations sequencing using convolutional recurrent neural networks. Manuf Lett 2025;44:91–101.

[93] Liang J, Zhang S, Zhang Y, Huang R, Xu C, Wang Z, et al. A knowledge graph-based approach to modeling & representation for machining process design intent. Adv Eng Inform 2024;62(Part A):102645.

[94] Hua Y, Wang R, Wang Z, Wang G, Yan Y. Knowledge graph with deep reinforcement learning for intelligent generation of machining process design. J Eng Des 2024;36(11):2072–106.

[95] Zhang L, Wu H, Chen Y, Wang X, Peng Y. A knowledge-guided process planning approach with reinforcement learning. J Eng Des 2024;36(7–9):1527–50.

[96] Hashimoto M, Nakamoto K. Process planning for die and mold machining based on pattern recognition and deep learning. J Adv Mech Des Syst Manuf 2021;15(2):JAMDSM0015-JAMDSM0015.

[97] Komura N, Matsumoto K, Igari S, Ogawa T, Fujita S, Nakamoto K. Computer aided process planning for rough machining based on machine learning with certainty evaluation of inferred results. Int J Automot Technol 2023;17(2):120–7.

[98] Deng T, Li Y, Liu X, Wang P, Lu K. A data-drivenParameter planning method for structural parts NC machining. Robot Comput-Integr Manuf 2021;68:102080.

[99] Deng T, Li Y, Chen J, Liu X, Wang L. Informed machine learning-based machining parameter planning for aircraft structural parts. Int J Adv Manuf Technol 2021;117(11):3563–75.

[100] Qin J, Hu F, Liu Y, Witherell P, Wang CCL, Rosen DW, et al. Research and application of machine learning for additive manufacturing. Addit Manuf 2022;52:102691.

[101] Jiang J. A survey of machine learning in additive manufacturing technologies. Int J Comput Integr Manuf 2023;36(9):1258–80.

[102] Toprak CB, Dogruer CU. A critical review of machine learning methods used in metal powder bed fusion process to predict part properties. Int J Precis Eng Manuf 2024;25(2):429–52.

[103] Caiazzo F, Caggiano A. Laser direct metal deposition of 2024 Al alloy: trace geometry prediction via machine learning. Materials 2018;11(3):444.

[104] Zhang W, Mehta A, Desai PS, Higgs Iii CF. Machine learning enabled powder spreading process map for metal additive manufacturing (AM). in 2017 International Solid Freeform Fabrication Symposium, 2017.

[105] Aoyagi K, Wang H, Sudo H, Chiba A. Simple method to construct process maps for additive manufacturing using a support vector machine. Addit Manuf 2019;27:353–62.

[106] Meng L, Zhang J. Process design of laser powder bed fusion of stainless steel using a gaussian process-based machine learning model. JOM 2020;72(1):420–8.

[107] Hertlein N, Deshpande S, Venugopal V, Kumar M, Anand S. Prediction of selective laser melting part quality using hybrid Bayesian network. Addit Manuf 2020;32:101089.

[108] Tapia G, Khairallah S, Matthews M, King WE, Elwany A. Gaussian process-based surrogate modeling framework for process planning in laser powder-bed fusion additive manufacturing of 316L stainless steel. Int J Adv Manuf Technol 2018;94(9):3591–603.

[109] Park HS, Nguyen DS, Le-Hong T, Van Tran X. Machine learning-based optimization of process parameters in selective laser melting for biomedical applications. J Intell Manuf 2022;33(6):1843–58.

[110] Dong G, Wong JC, Lestandi L, Mikula J, Vastola G, Jhon MH, et al. A part-scale, feature-based surrogate model for residual stresses in the laser powder bed fusion process. J Mater Process Technol 2022;304:117541.

[111] Mohamed OA, , Masood SH, Bhowmik JL. Influence of processing parameters on creep and recovery behavior of FDM manufactured part using definitive screening design and ANN. Rapid Prototyp J 2017;23(6):998–1010.

[112] Mohamed OA, Masood SH, Bhowmik JL. Modeling, analysis, and optimization of dimensional accuracy of FDM-fabricated parts using definitive screening design and deep learning feedforward artificial neural network. Adv Manuf 2021;9:115–29.

[113] Jiang J, Hu G, Li X, Xu X, Zheng P, Stringer J. Analysis and prediction of printable bridge length in fused deposition modelling based on back propagation neural network. Virtual Phys Prototyp 2019;14(3):253–66.

[114] Sahoo S, Kumar S, Abedin MZ, Lim WM, Jakhar SK. Deep learning applications in manufacturing operations: a review of trends and ways forward. J Enterp Inf Manag 2023;36(1):221–51.

[115] Li C, Zheng P, Yin Y, Wang B, Wang L. Deep reinforcement learning in smart manufacturing: a review and prospects. CIRO J Manuf Sci Technol 2023;40:75–101.

[116] Priore P, Ponte B, Puente J, Gómez A. Learning-based scheduling of flexible manufacturing systems using ensemble methods. Comput Ind Eng 2018;126:282–91.

[117] Yang S, Xu Z. Intelligent scheduling and reconfiguration via deep reinforcement learning in smart manufacturing. Int J Prod Res 2022;60(16):4936–53.

[118] Jing X, Yao X, Liu M, Zhou J. Multi-agent reinforcement learning based on graph convolutional network for flexible job shop scheduling. J Intell Manuf 2024;35:75–93.

[119] Wang X, Zhang L, Liu Y, Laili Y. An improved deep reinforcement learning-based scheduling approach for dynamic task scheduling in cloud manufacturing. Int J Prod Res 2024;62(11):4014–30.

[120] Greminger M. Generative adversarial networks with synthetic training data for enforcing manufacturing constraints on topology optimization. In: Proceedings of the ASME 2020 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference; 2020 Aug 17–19; Virtual, Online. New York City: ASME; 2020. p. DETC2020-22399, V11AT11A005.

[121] Karnewar A, Wang O, Iyengar RS. MSG-GAN: multi-scale gradient GAN for stable image synthesis. 2019. arXiv: 1903.06048.

[122] Wang Z, Yan X, Melkote SN, Rosen D. McGAN: Generating manufacturable designs by embedding manufacturing rules into conditional generative adversarial network. Adv Eng Inform 2025;64:103074.

[123] He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV); 2017 Oct 22–29; Venice, Italy. Piscataway: IEEE; 2017. p. 2980–8.

[124] Isola P, Zhu JY, Zhou T, Efros AA. Image-to-image translation with conditional adversarial networks. In: Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV); 2017 Jul 21–26; Honolulu, HI, USA. Piscataway: IEEE; 2017. p. 5967–76.

[125] Wang Z. Generative design using deep learning methods for functionality and manufacturability. Report. Atlanta: Georgia Institute of Technology; 2022.

[126] Jiang J, Xiong Y, Zhang Z, Rosen DW. Machine learning integrated design for additive manufacturing. J Intell Manuf 2022;33:1073–86.

[127] Xie J, Sun L. Zhao YF. On the data quality and imbalance in machine learning-based design and manufacturing—a systematic review. Engineering 2024;45:105–31.

[128] Wang Z, Melkote SN, Rosen DW. Generative design by embedding topology optimization into conditional generative adversarial network. J Mech Des 2023;145(11):111702.

[129] Oh S, Jung Y, Kim S, Lee I, Kang N. Deep generative design: integration of topology optimization and generative models. J Mech Des 2019;141(11):111405.

[130] Shu D, Cunningham J, Stump G, Miller SW, Yukish MA, Simpson TW, et al. 3D design using generative adversarial networks and physics-based validation. J Mech Design 2020;142(7):071701.