The International Journal of Advanced Manufacturing Technology (2020) 106:2363–2375
https://doi.org/10.1007/s00170-019-04770-3

ORIGINAL ARTICLE

STEP-NC feature-oriented high-efficient CNC machining simulation

Gang Zhao[1,2] · Xian Cao[1] · Wenlei Xiao[1,2] · Qiang Liu[1] · Martin Byung-Guk Jun[3]

Received: 1 July 2019 / Accepted: 29 November 2019 / Published online: 19 December 2019

Abstract

During the development of CNC machining, the computer-based NC machining simulation has played an important role in verifying the NC programs economically and safely so as to avoid scraps or even accidents. In order to balance the accuracy, efficiency, and fidelity, various simulation methods have been proposed in historical researches, including wireframe, solid, object space-based, and image space-based simulation. All the mentioned methods use G-code as input and take the workpiece blank as a whole part, which causes the difficulty for achieving real-time simulation with acceptable accuracy under limited computing resources. This paper proposed a STEP-NC feature-oriented machining simulation method, which takes the workpiece blank as the combination of several machining features plus the workpiece, and implements customized algorithms for individual features. Using the divide and conquer strategy, high efficiency with guaranteed accuracy and fidelity can be obtained during the real-time simulation. Two demo workpieces with 2.5D features from the STEP-NC standard were tested using the proposed method, and the efficiency was validated by comparing results with the conventional methods.

Keywords STEP-NC · Feature-oriented simulation · Machining simulation · Machining feature

1 Introduction

In order to detect cutting errors, potential collisions, or improve efficiency, the computer-based CNC machining simulation has been implemented in the industry for decades. With the arising concepts of intelligent manufacturing, Cyber-Physic System(CPS), and Digital Twin, the machining simulation plays more important roles not only before but also during the real cutting process for monitoring and observing purposes. Real-time cyber-physical reflection definitely requires the system to run more efficiently with acceptable accuracy than the off-line simulation version. Conventional machining simulation methods take the workpiece blank as a whole part, and consider only toolpaths (mostly G-code) during the simulation, hence cannot guarantee both high efficiency and accuracy under limited computing resources. STEP-NC provides much more abundant information than G-code, and can represent the workpiece as a combination of machining features. In view of this, this paper proposed a feature-oriented machining simulation framework based on STEP-NC standard, which supports high-efficient and high-fidelity performance.

Various machining simulation methods have been researched during the last decades, which mainly fall into four categories [32]: wireframe-based, solid-based, object space-based, and image space-based. Due to its pretty simple data structure [28], the wireframe-based method is widely used in early systems [5, 27, 34]. Nonetheless, because of its inability in representing the solid model of the workpiece, it is rarely used in contemporary commercialized systems [1, 4]. The most accurate simulating result with the highest fidelity can be obtained by the solid-based method [32], which requires the CSG or B-rep model for analytical intersection [20, 23]. However, extreme accuracy requires heavy computational efforts. Hence, this method is rarely used for the fast toolpath verification [2]. More effective simulation methods that are frequently used in industrial production are the object space-based method (Octree, Z-map, etc.) [3] and the image space-based method (Z-buffer, Tri-dexel, etc.) [4]. In these methods, workpiece is subdivided into limited numbers of

✉ Wenlei Xiao
xiaowenlei@buaa.edu.cn

[1] School of Mechanical Engineering & Automation, Beihang University, Beijing, 100191, China

[2] MIIT Key Laboratory of Aeronautics Intelligent Manufacturing, Beihang University, Beijing, 100191 China

[3] Department of Mechanical Engineering Technology, Purdue University, West Lafayette, IN 47907-2021, USA

simple elements that balancing accuracy and space complexity [2, 32]. Compared with the image space-based method, the object space-based method consumes much more memory, because over-fine subdivision can generate dramatic numbers of elements [24]. Moreover, the triangle mesh generation in object space-based method is also time consuming [9, 11, 30].

Besides the geometry verification, physical state prediction is also important for machining. Cutting force and surface quality are predicted in physical simulation while taking the result of geometry simulation as input [15, 19]. Efficient and accurate geometry simulation guarantee the credible physical simulation result [21]. And some researches focused on reducing computation time to improve simulation efficiency [6].

All the mentioned methods take G-code as input, so the whole workpiece blank is conducted in intersection computing or subdivision. In this paper, STEP-NC features rather than the whole blank are taken as individual simulation "blanks," so that the divide and conquer strategy can be used to isolate the computation inside features. STEP-NC have been researched in machining simulation to improve the data stream integrity and fidelity closing to practical situation [10]. And the STEP-NC standard application in other NC machining have been promoted by many research works [17].

In contrast to conventional simulation methods, the feature-oriented method has several significant advantages. (1) The complexity of a feature is irrelevant to the workpiece, so usually it is much simpler than the whole workpiece. (2) The blank is separated into features, so materials that are not located inside features have no need to intersect with tool, which saves a lot of computation. (3) The computation of individual features are mutually independent, which greatly decreases the computation consumption and helps the real-time simulation. (4) When a feature is completely machined during the simulation, it can be directly hidden and even removed from the data structure; hence, the time and space complexities are gradually decreased during the simulation. (5) Since the STEP-NC provides diverse definitions for features, operations, strategies, and operations, customized simulation algorithms with optimized performance can be developed for different features and operations. (6) As the features are irrelevant, tailored parameters and methods can be applied to obtain the highest accuracy, efficiency, and fidelity.

A series of methods are developed to achieve feature-oriented machining simulation, and this system is verified on two demo parts in the STEP-NC standard document (ISO 14649-11) and two customized freeform parts. The simulation results prove that real-time simulation with high efficiency and accuracy can be guaranteed, comparing with conventional methods.

2 Feature-oriented machining simulation

The STEP-NC standard provides diverse machining features definitions [29, 33, 35]. For a single feature, there is the best suited simulation method that can be an existing methods or a customized method. Based on the analysis of conventional simulation methods, the available methods for feature-oriented simulation system are enumerated in this section. Some of these methods are customized and optimized to attain high efficiency and accuracy. Additionally, features in STEP-NC standard are classified according to the characteristics of material removing process during simulation. Finally, some criterions of simulation methods choosing are summarized for different types of features.

2.1 Available simulation methods

In view of the diversity of STEP-NC features, a series of simulation methods should be readjusted and redesigned to obtain the optimum performance. Obviously, existing simulation methods are not enough to meet all the feature-oriented requirements. In this section, the characteristics of existing methods (solid-based, image space-based, and object space-based) are analyzed, while the legacy wireframe-based method is out of discussion. Moreover, other customized simulation methods are introduced to expand the family of machining simulation methods and fulfill conditional requirements of high efficiency.

For solid-based method, the main advantage is the accuracy they can achieve. Representing by CSG, B-Rep or triangular meshes, a machining workpiece can be cut with the most accurate way. However, the complexity of Boolean operation increases during the simulation for the engagement of tool and workpiece is changing in every simulation step. Moreover, visualization model generation of CSG, B-Rep method is also a compute-intensive task [18]. For object space-based method, the simple discrete model of blank makes it efficient. And the limitations of the object space-based methods are similar: the number of discrete elements grows dramatically which causes the increasing memory consumption [8]. For image space-based method, this method is most efficient since the Boolean operation is simplified into one dimension. But this method only produces the image of blank which is not suitable for further analysis [7, 25, 26].

Based on the analysis of existing simulation methods mentioned before, five algorithms are presented here as the available algorithms to be used in feature-oriented simulation system. Different simulation methods are suitable for different situations.

(1) CSG or B-Rep method

In STEP-NC feature-oriented simulation, the complexity of Boolean operation using CSG or B-Rep model can be significantly reduced because the calculation is limited inside a local area and the regulation of engagement between tool and blank can be exploited for some kinds of features. This method is suitable for modeling the removed volume with a consist engagement between tool and blank which has been analyzed in detail by Yip [31]. In this case, the cutting process is not available to compromise the requirement of efficiency. There is another use of this method is to model the geometry of removed material for all features. This result is the precondition to apply other simulation methods independently. The modeling process is discussed in details in Section 3.

(2) Mesh stretch method

This method is a modification of the mesh interaction method but avoiding the complex mesh interaction calculation. For simple machining feature, the triangle mesh is stretched inside the blank synchronized with the tool feeding process and vertexes of mesh are translated to new position to achieve the deforming effect. Obviously, this method is efficient to calculate the mesh model of machined feature because the Boolean operation is quite simple. With low computation consumption, high discrete precision of mesh model is allowed to get accurate simulation result. Another advantage is that the triangle mesh model can be used directly in visualization and conversion is not needed.

(3) Extended quadtree method

As another customized method, the hierarchical quadtree is extended with signed distance field to achieve fast Boolean operation. The principle of this method is inspired by the research works of Liu [16] and Li [14] which belonging to *Z*-map-based method. This method is used to simulate the machining process in which the tool moves along a 2D toolpath and removes material with uniform depth value. The supported machining process is also a common situation in 2.5D machining. One of the evident advantages is decomposing the volume of a feature into multi-layers, so that quadtree model of the machined layer can be directly dropped from the memory. Additionally, to improve the accuracy, the mesh model is used inside the quadtree node to calculate the tool intersection.

(4) Voxel list

This method is designed for finish milling operation, in which the geometry of removed material is a thin-rounded pieces. To represent this special shape, voxels are arranged as a list along the boundary of the removed material and the discrete model is built. This method can be regarded as a special case of extended quadtree, while only the quadtree node with minimum size appears which is defined as voxel. As the feeding direction is confirmed, it is easy to pick out the voxels participating the Boolean operation, so that the simulation is very efficient.

(5) *Z*-map-based method

Z-map simulation method has been applied and improved in many research works which belongs to the object space-based methods. In this paper, the method proposed by Li [14] is adopted as an available method to simulate the 3-axis milling process. In feature-oriented simulation framework, this method is only applied for a restricted area defined by machining feature and the memory consumption is limited as a low level while hight accuracy is supported.

(6) Octree-based method

The octree-based simulation method has been relatively mature according to the existing research works [9, 11, 30]. The method proposed by Yau [30] is adopted as an available method which supports 3-axis and 5-axis milling process. Inherent drawbacks of this method are about two aspects: the discrete precision notably affects the efficiency and the sharp edge geometry is lost easily which is needed to be handled specially [12, 13, 24]. However, by applying this method for restricted area, these problems can be resolved which is similar to the *Z*-map method.

2.2 Feature classification

The simulation methods mentioned above are suitable for different situations. As the basic criterion for simulation method selection, machining features need to be classified considering the characteristics of the machining process and requirements of machining simulation. It is obvious that the classification of features in STEP-NC standard cannot be used directly because it is originally designed to conveniently describe machining process. Therefore, machining features defined in STEP-NC standard are summarized and analyzed to classify for simulation purpose.

As showed in Fig. 1, the blank consists of multiple machining features. This is a typical 2.5D part which has three machining features: planar face, round hole, and closed pocket. Different machining feature has different machining process and geometry of removed material. Features are mutual independent and the simulation methods mentioned in Section 2.1 can be selected for different features to achieve high accuracy and efficient.

The integrated data of machining process illustrated in Fig. 1 can be described using STEP-NC standard according with object-oriented concept. A STEP-NC file contains 5 types of data including working step, feature, operation, tool, and workpiece. Machining feature defines the removed material volume which is the main interest of this paper. Operation and tool influence the way of material removing.

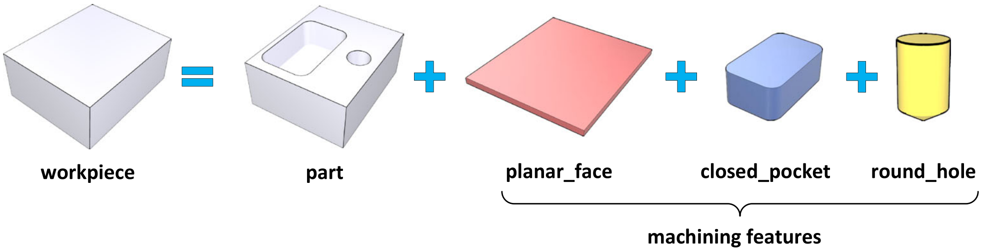


Fig. 1 The relationship between workpiece blank and machining features

This three types of data are taken into account in feature-oriented simulation.

The data in STEP-NC standard is not used directly in feature-oriented machining simulation. So the feature classification criterions are proposed which is inspired by the concept of engagement shape between tool and blank in Yip's work [31]. Features with similar material removing regulation are regarded as the same type and suit same simulation method.

As listed in Table 1, the regulations of engagement deforming are summarized as four classes: one dimension, two dimension, three dimension, and freeform. The characteristics of these four types are discussed as follows.

One dimension: The geometry of one-dimension feature is formed by the tool moving along a line and intersecting with blank. In this situation, Boolean operation is one dimension and it is very easy to calculate the geometry of machined feature volume. In STEP-NC standard, "round_hole" and "slot" are two typical one-dimension features. Additionally, geometry of removed material is different in roughing and finishing operation for the same one-dimension feature, but the characteristics of engagement deforming process are same.

Two dimension: The main characteristic of two-dimension features is that the machining process repeats for every layer of feature volume. For example, the "closed_pocket" feature can be divided into multi-layers which are arranged along z-axis. Inside one layer, the tool only moves in the x-y plane, and for which the calculation of engagement becomes a 2D problem. Additionally, if finishing operation is taking into consideration, the removed material only distributes around the boundary of the feature and tool only moves along a profile in the x-y plane. This characteristic may lead to use different simulation methods comparing with roughing operation.

Three dimension: Three-dimension feature is also belonging to 2.5D machining feature, but deformation of tool-workpiece engagement exists in three directions. In STEP-NC standard, there are two classes of three-dimension feature. The first is the compound feature, which can be decomposed into several one- or two-dimension features. So the machining process is also decomposed into multiple parts that simulated separately. The second is non-compound while the shapes of cross-section along z-axis are different. For example, the feature "Shape_profile" showed in Table 1 satisfies this situation. Generic simulation method is needed because the machining process is more complex than one or two dimension.

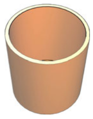
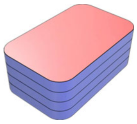
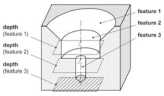
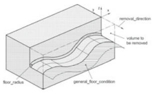
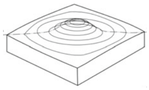
Freeform: In this kind of feature, the machined surface is freeform surface, either 3-axis or 5-axis. The deformation of engagement between tool and workpiece blank is complex and does not have noticeable regulation, so more generic simulation method is needed.

2.3 Simulation methods selection criteria

Based on the classification of machining features, the suitable simulation methods can be selected for different features. The matching rules of simulation methods and features are summarized in Table 1, in which the relationships between features and available simulation methods are presented. The simulation methods selection criteria are explained in detail as follows.

1. For one-dimension feature, the mesh stretch method is suitable no matter the difference in roughing and finishing operation.
2. For two-dimension feature with roughing operation, the extended quadtree method is used. The removed material volume is divided into multiple layers and represented by the assembly of quadtree models. One layer is corresponding with a quadtree model.
3. For two-dimension feature with finishing operation, the voxel list method is applied. Voxel list model is built around the boundary of the remaining volume after roughing operation.
4. For compound three-dimension features, the feature is divided into several one- or two-dimension feature. And the appropriate simulation methods are applied for the simple sub features.
5. For non-compound three-dimension features, Z-map method is used. The machining process is similar with common 3-axis machining, for which the Z-map method is suitable.
6. For freeform feature, Z-map method or octree-based method is used. Z-map method is more suitable for

Table 1 The result of feature classification and simulation methods selection

Feature		Operation	Type index	Simulation method	Example
One dimension		Roughing	1	Mesh stretch	Round_hole & Drilling
		Finishing	2	Mesh stretch	 Round_hole & Reaming
Two dimension		Roughing	3	Extended quadtree	 Closed_Pocket & Bottom_and_side_rough_milling
		Finishing	4	Voxel list	Closed_Pocket & Bottom_and_side_finish_milling
Three dimension	Compound	–	–	–	 feature 1 feature 2 feature 3 depth (feature 1) depth (feature 2) depth (feature 3) counterbore_hole
	Non-compound	–	5	octree	 removal_direction volume to be removed floor_radius general_floor_condition shape_profile
Freeform			6	*Z*-map or octree	 freeform

3-axis machining, and octree-based method is selected for 5-axis machining.

3 STEP-NC feature modeling

To apply selected simulation method for machining feature independently, the geometry of each feature is important which defines the removed material volume. Computation consumption of a simulation method is limited, because the adapted volume is only a part of workpiece blank. Therefore, it is important to propose a method to build the accurate geometry of machining feature.

As the STEP-NC standard is designed to describe the machining process, the geometry parameters of machining features cannot be obtained directly from STEP-NC file. The geometry parameters are not only related to features element, but also the relevant elements including operation, tools, and strategy. In addition, the extraction method of geometry parameters from STEP-NC elements is inconsistent for different types of machining features.

3.1 Feature geometry modeling

Considering these difficulties, modeling methods are designed for different types of machining features. The modeling process is summarized as 2 steps: extract geometry parameters from STEP-NC definitions; construct the geometry model by calling a CAD kernel.

To explain the modeling process, hole and pocket features in Fig. 1 are taken as examples. The modeling results are shown in Figs. 2 and 3. In these figures, geometry

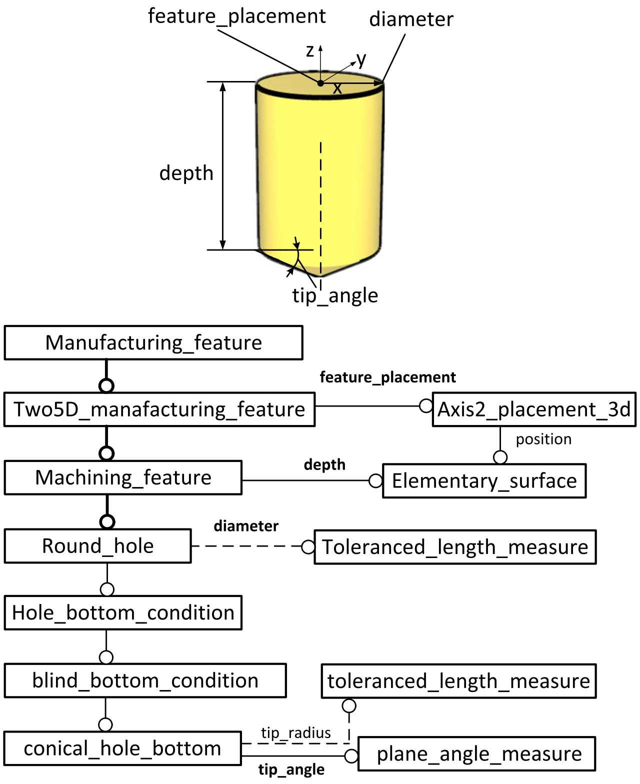


Fig. 2 The geometric parameters and modeling result of “round_hole”

parameters are labeled to illustrate the relationship between feature definitions and modeling results.

“round_hole”: Modeling method of “round_hole” feature is shown in Fig. 2. The basic geometry parameters of this feature are “diameter” and “depth.” The bottom condition is defined by “conical_hole_bottom” and “tip_angle” is used in modeling. While calling the CAD kernel, an outline is built firstly which is revolved around the hole axis to form the geometry.

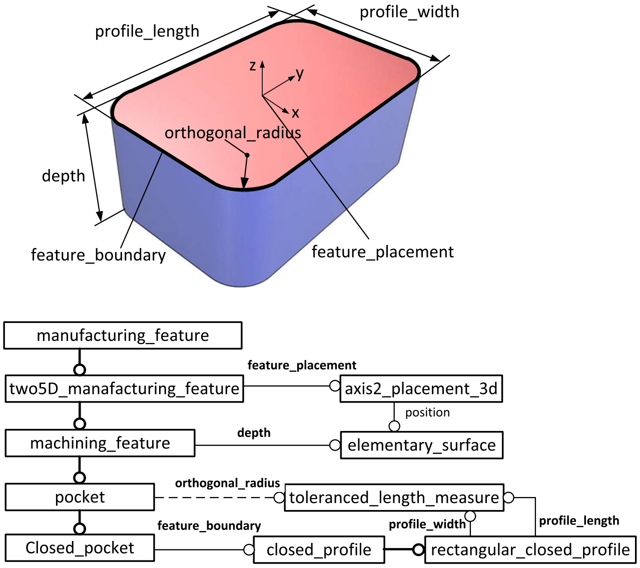


Fig. 3 The geometric parameters and modeling result of “closed_pocket”

“closed_pocket”: Geometry parameters of this feature are labeled in Fig. 3. The “closed_profile” element defines the shape of sketch which is a rounded rectangle. Then extrude modeling operation is performed along the direction provided by “depth” which is an attribute of “machining_feature.” The geometry parameters in Fig. 3 are delivered to the CAD kernel to build the final shape.

In addition to the mentioned 3 features, all 2.5D machining features in STEP-NC standard are supported by the proposed modeling method. To demonstrate this capability, the two demo parts in STEP-NC standard document are tested and the result is illustrated in Fig. 4.

Geometry modeling of machining features from the STEP-NC file is an important preprocessing phases before applying the selected simulation methods. The initial simulation model is build based on the geometry modeling result.

3.2 Tool modeling

Geometry of tool is also needed in NC simulation to perform the Boolean operation with workpiece blank. In STEP-NC, the tool geometry definition is complete enough to meet the requirement of tool modeling. The modeling result is converted into triangle mesh which is used in visualization. For Boolean operation, the simulation model of tool is build based on the modeling result.

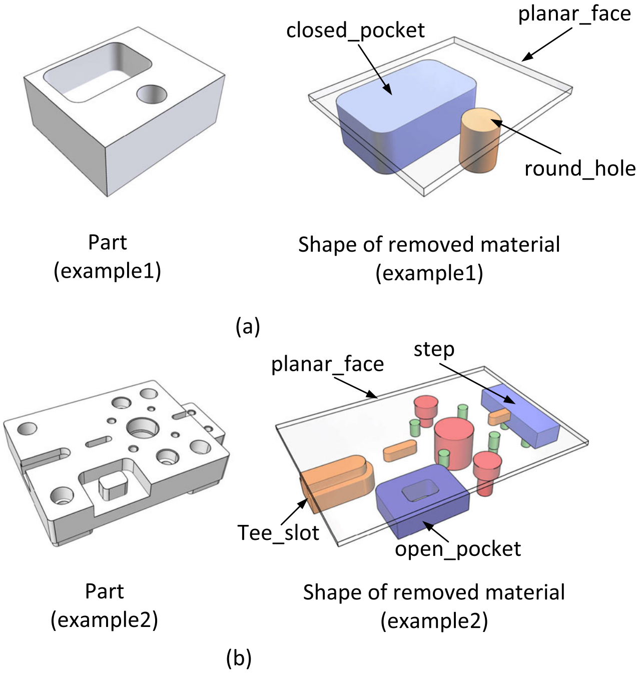


Fig. 4 **a**, **b** The modeling results of demo parts provided by STEP-NC standard (ISO14649-11)

The tool definitions in STEP-NC are analyzed to extract geometry parameters. A general tool modeling result is illustrated in Fig. 5 and the related parameters are labeled. Additionally, different tool models are needed that adapting to different simulation methods. A specified tool simulation model is established (Fig. 5) considering the available simulation methods mentioned previously.

The signed distance field is used and improved in tool simulation model shown in Fig. 5. Given a point in 3D space, three distance values are calculated from this model. The first value is the minimum distance from point to the boundary of tool the same as the definition of distance field [30]. The second value is the minimum distance in cross section of tool as illustrated in Fig. 5. For point p3, the cross section of tool passing over p3 is found firstly which is labeled as c2. Then the minimum distance from p3 to the outline of c2 is calculated as the second value. The situation for point p1, p2 is similar, but the outline of cross section is different and the minimum distance value is different. The third value is defined as the minimum distance from point to the boundary of tool at *z*-axis. Only for points inside the tool, this value is computed. For example, point p2 is translated along *z*-axis to point p2' which is located on the tool boundary and the distance between p2 and p2' is returned.

The simulation model of tool is categorized as explicit definition and is suitable for Boolean operation. For different simulation methods, the three distance values mentioned above are used in different manner.

3.3 Modeling with operation information

In general, a machining feature is associated with more than one operation, for example the roughing and finishing. Considering the influence of different operations, the geometry of removed material for one feature can be further divided into smaller parts. In STEP-NC standard, the remaining volume after one operation is not defined explicitly and the method to extract these geometry parameters is different for different operation. Three possible situations are summarized while modeling from operation information.

(1) The first situation is that operation element contains “allowance” attribute which describes the thickness of left material. In this situation, a smaller geometry shape is build that its boundary offset from the feature boundary.

(2) The second one is that the geometry parameter is not provided by operation directly, while the tool dimension must be considered. According to the tool dimension, the removed material volume is calculated by perform Boolean difference operation of the feature geometry with tool geometry.

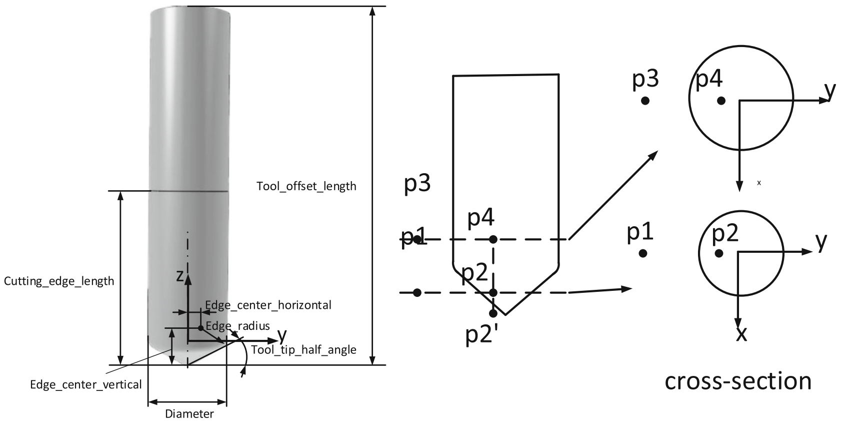
(3) The third one is mainly applied on the two-dimension features. As discussed in Section 2, two dimension features are divided into multi-layers inside which the machining process is repeated. The distance between two adjacent layers is not provided by “machining_feature” element, and it is defined in operation.

Two features, “close_pocket” and “round_hole,” are taken as examples to explain the modeling process with operation information.

For “round_hole,” this feature is associated with two operations: “drilling” and “reaming.” This is the second situation that allowance distance is not defined in operations. Consequently, tool dimension is used to build geometry. The modeling result of these two operations is shown in Fig. 6. The geometry of “drilling” is the result of Boolean difference operation between feature volume and tool volumes along the depth direction. And for reaming operation, the geometry is built by performing a Boolean difference between feature volume and the shape of drilling operation.

For “closed_pocket,” the modeling result is shown in Fig. 6. The first situation is satisfied and two allowance distance values are defined on the bottom and side surface. Using these two distance values, the geometry shapes of

Fig. 5 The geometry modeling result of tool from STEP-NC standard



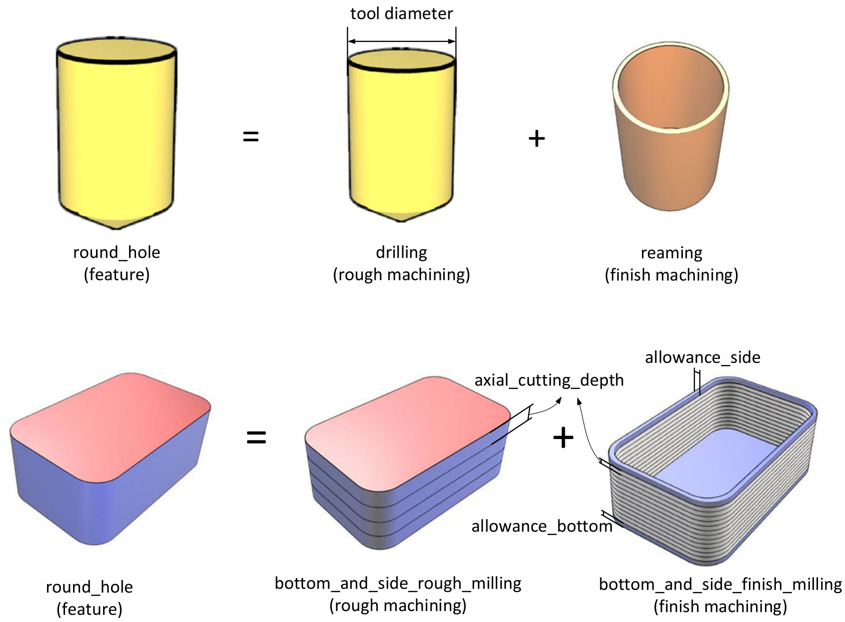


Fig. 6 Geometry modeling of "closed_pocket" and "round_hole" using operation information

roughing and finish operations are build. In addition, the third situation is also suitable and the geometry can be further divided into multi-layers arranged along the z-axis.

4 Simulation methods realization

In this section, realization of simulation methods is explained which are listed in Table 1 as alternative methods for feature-oriented simulation. This section focuses on the three customized simulation methods: mesh stretch, extended quadtree, and voxel list, designed to reduce the computation consumption by taking full advantage of the characteristics of machining features. The three machining features shown in Fig. 1 are used as examples to explain intuitively. The two methods, z-map and extended octree, are pre-existing and the significant difference on using compared with common simulation situation is that the feature modeling result must be used in initialization to identify the local area. At last, some criteria are proposed for the collaboration of multiple simulation methods.

4.1 Mesh stretch method

Mesh stretch method is suitable for one-dimension feature no matter roughing or finishing. Here, the hole feature is taken as an example to explain the principle. With the tool moving along the depth direction, the deformation of surface is just like the mesh point stretch along the same direction. So the shape of machined hole is determined by the tool position.

According to the regulation of deforming, the specified mesh model is built utilizing the feature modeling result presented in Section 3. The critical step is to assign some dynamic vertexes in mesh model, and the position of these vertexes is updated according to the tool position.

The mesh structure for drilling operation is illustrated in Fig. 7. Vertexes located on the circular edge are static and keep position during simulation. Because the tool tip contacts with feature volume, there are a group of dynamic vertexes between center vertex and circular edge. Dynamic vertexes are translated along feed direction. For reaming operation, dynamic vertexes are located on the two circular edges labeled as red line in Fig. 7. When the tool tip touched the bottom, all vertexes are converted to dynamic. The positions of dynamic vertexes are computed using the minimum distance from vertex to the boundary of tool at z-axis which is the third distance of tool model proposed in Section 3.2.

In the mesh stretch method, Boolean operation is simple and the position of dynamic vertexes is easy to calculate. This characteristic ensures the high efficiency, based on which the accurate mesh model with dense segmentation is

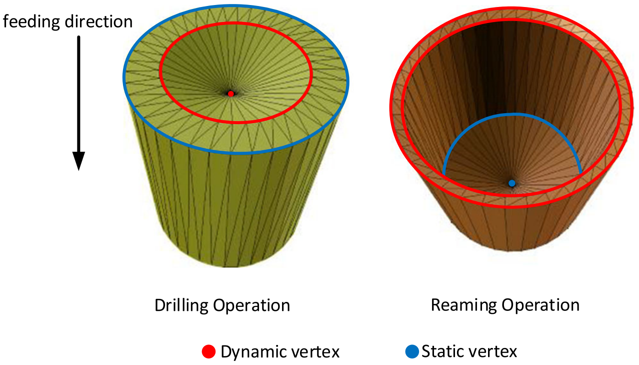


Fig. 7 Mesh structure of "round_hole" feature in simulation

allowed to achieve high accuracy. Mesh model is also an advantage that it can be directly used in the visualization.

4.2 Extended quadtree method

As discussed in Section 2, the extended quadtree method is suitable for two-dimension feature under roughing operation. The pocket feature with rough milling operation is taken as an example to explain this method. This method is inspired by existing quadtre-based simulation methods [14, 16, 22]. The main differences are as follows: quadtree models are built for every layer of feature volume; signed distance field and mesh intersection are combined to perform Boolean operation.

The distribution of quadtree models is illustrated in Fig. 8 that coordinates with layer structure of pocket feature volume. There are three types of flags indicating the position relationship between node and feature volume, including INSIDE, OUTSIDE, and UNDECIDED. For pocket roughing, this structure helps to search intersection area efficiently.

For the Boolean operation, signed distance field is utilized to search the intersected nodes rapidly. Inside a quadtree node, a polygon model is built to represent the details of machined shape. Polygon data structure is similar to the half edge as shown in Fig. 9b. Vertexes, edges, and faces are linked by pointers so that it is efficient to search for vertex and edge. Once cut by a tool, a polygon is divided into two parts and the new generated polygons are organized as a binary tree structure shown in Fig. 9c. In Fig. 9a, the polygon model of a node is cut by tool 4 times. The process of tool cutting is regarded as deformation of polygon intersected by tool.

To generate the 3D model illustrated in Fig. 9d, side surface that connecting two polygon nodes is constructed. The face f5 is located inside the tool and vertex v2', v3', and v6' are translated down to fit the tool tip position. Then the 4 vertexes, v2', v3', v2, v3, form a new polygon which is regarded as the side surface.

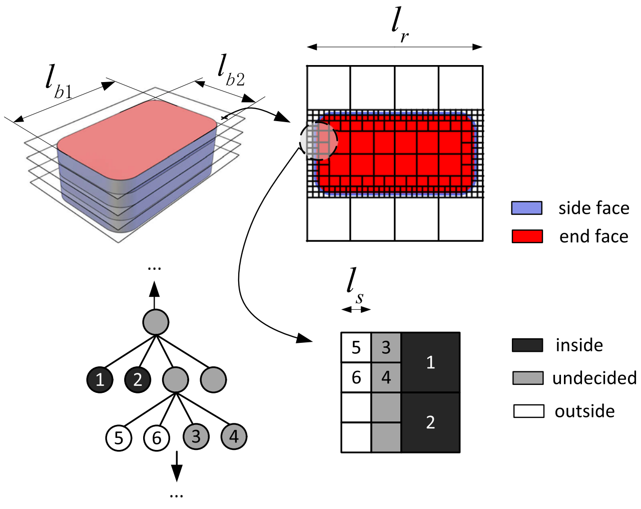


Fig. 8 The extended quadtree model

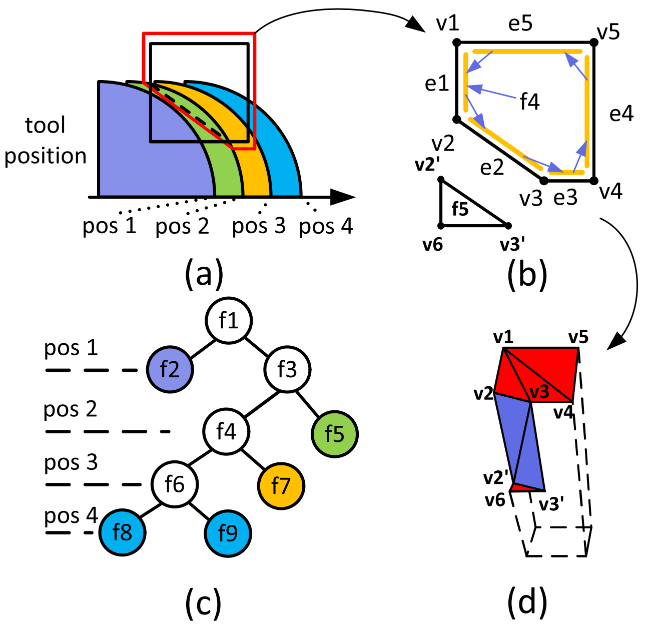


Fig. 9 The polygon model inside a quadtree node

4.3 Voxel list method

Voxel list method is another customized method only for two-dimension features with a finishing operation. Here, the finish milling operation of pocket is taken as an example to explain the principles. As shown in Fig. 10, there is only a thin rounded pieces of material is removed in this machining process.

Voxel is a special quadtree node that the four edges are not required to form a rectangle and do not have a parent node. Voxels are arranged into a list along the boundary

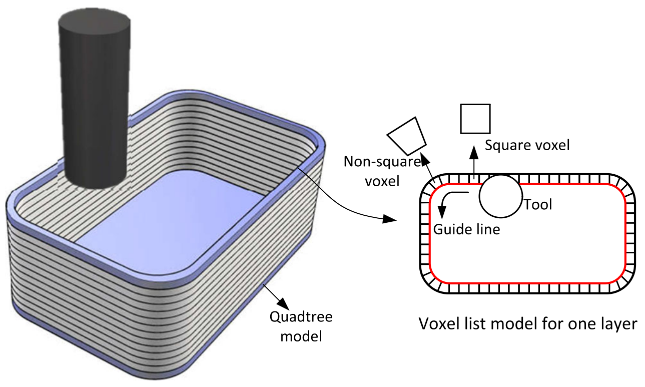


Fig. 10 Voxel list model of removed material volume in pocket finishing

of removed material volume as shown in Fig. 10. Boolean operation between voxel and tool is same with extended quadtree method. It must be mentioned that the removed volume in bottom layer is similar to the two-dimension feature with roughing operation and a single layer quadtree model is constructed. Computation consumption of this method is very low because the number of voxel participating the Boolean operation is small.

4.4 Extended octree method and *z*-map method

Extended octree method and *z*-map method have been proposed by existing research works [14, 16, 30]. The specified methods used in feature-oriented simulation are introduced in Section 2.1.

Belong to the object-space-based simulation method, there is a conflict between high discrete precision and low memory consumption for these two methods. But in feature-oriented simulation framework, this conflict is eased because these methods are applied only for local areas. Before applied for simulation, the extended octree model or *z*-map model must be initialized with the geometry of machining feature.

4.5 Collaboration of multiple simulation methods

The entire simulation process is the combination with different simulation methods. Under this feature-oriented framework, an important issue that must be handled is to output the uniform simulation result to achieve an integrated machining simulation system. And the execution sequence of machining operation must be correct to ensure that the appropriate simulation method is used to simulation. The two principles are discussed below to deal with these problems.

(1) Uniform result of different simulation methods.

Although the simulation model and Boolean algorithm are different, the simulation result is same for the four methods mentioned in this section, which is the triangle mesh model. The triangle mesh is the uniform data structure for visualization and it is easy to achieve a complete simulation process.

(2) Execution sequence of simulation methods.

The execution sequence can be obtained from the "workplan" element defined in the STEP-NC file. The "workplan" element and the related "workingstep" element define the machining sequence. And this sequence determines the order of machining features and operations to be employed in simulation. Every combination of machining feature and operation is associated with the simulation model and the suitable method is selected before the simulation started.

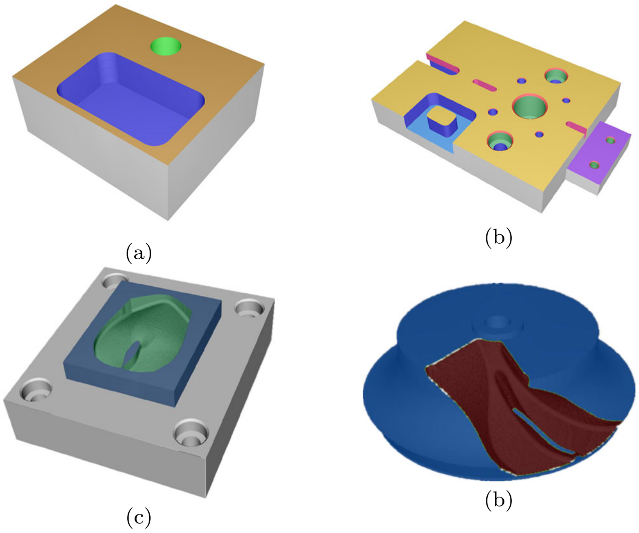


Fig. 11 The simulation result of example parts using feature-oriented method. **a**, **b** 2.5D parts in STEP-NC standard document. **c** A mouse model combining 3-axis freeform feature and 2.5D features. **d** An impeller model that indicates the ability of 5-axis simulation

5 Case study and experiment

The proposed feature-oriented simulation system has been implemented and coded in C++. Two 2.5D parts in STEP-NC standard document (ISO14649-11) and two freeform parts are validated in this simulation system running on a PC with 3.6GHz Intel Core CPU and NVIDIA Geforce GT 705 graphics card. Real-time interaction is supported in a 3D view window.

To demonstrate the advantages on efficiency and accuracy, statistics of time consumption, memory consumption, and geometry error are listed using three simulation methods. Feature-oriented method are compared with two common used methods: octree-based method [30] and *z*-buffer-based method [13]. In addition, visualization effect of sharp edge is compared and discussed to indicate that this simulation system can provide high-quality geometry model of machined workpiece (Fig. 11).

The comparison of efficiency is listed in Tables 2 and 3. Two types of efficiency are investigated: time and space.

Table 2 Comparison of average time consumption in one update process

Example	Time consumption (ms)		
	Feature-oriented	Octree	*Z*-buffer
1	22	112	47
2	24	191	70
3	107	267	176
4	126	126	110

Table 3 Comparison of max memory consumption of whole simulation process

Example	Memory (MB)		
	Feature-oriented	Octree	*Z*-buffer
1	75	318	150
2	114	866	237
3	82	487	93
4	816	816	136

For time efficiency, the computation time of one updating step is recorded. The time consumption includes Boolean operation and triangle mesh generation which are the most complex computation process. In order to get credible result, the minimum size of discrete element is set to 1 mm for the three simulation methods and four example. For space efficiency, the max memory consumption is recorded and compared.

It is obvious that feature-oriented simulation method takes less time in updating than other two methods. This phenomenon is even more obvious in 2.5D machining simulation. And the comparison of memory consumption is similar to time consumption. The main reason is that several customized algorithms are applied in feature-oriented simulation using the definition in STEP-NC file and the divide and conquer strategy helps to reduce the computational complexity. But for freeform part, the *z*-map or octree is used and it is not easy to achieve a strong advantage in efficiency.

The statistics are arranged in Table 4 to compare simulation accuracy of the three simulation method. The designed part geometry and generated geometry are compared to calculate the error value. The two freeform parts are not listed because the principle of simulation is similar and it is meaningless to compare. But for 2.5D machining process, the feature-oriented method can generate much more accuracy geometry model.

The plain triangle mesh model does not prevent the feature-oriented methods from producing accurate geometric models. Taking the open pocket milling simulation as an example, the geometric models generated by the three simulation methods are shown in Fig. 12. It is obvious that the sharp edge features have been lost in octree-based methods and *z*-buffer-based method, while the feature-oriented method proposed in this paper preserved the sharp features well and the visual effect is close to the *z*-buffer-based method.

Table 4 Accuracy of generated workpiece geometry model

Example	Error (mm)		
	Feature-oriented	Octree	*Z*-buffer
1	0.15	0.48	0.36
2	0.23	0.50	0.43

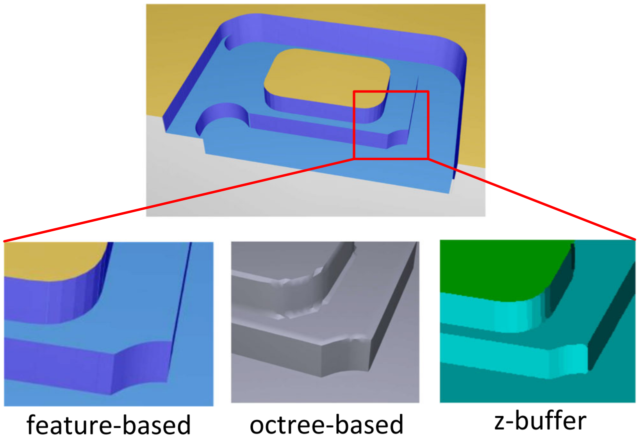


Fig. 12 Comparison of sharp edges generated by different simulation method

The frame rate is also recorded during the simulation process as shown in Fig. 13. It remains approaching 60 HZ, which is almost equal to the frame rate of rendering static triangle mesh. So the real-time interaction operation by user is supported.

In summary, the results of the experiment support that the feature-oriented machining simulation method proposed in this paper has following advantages: high simulation efficiency, small computational resource consumption, and accurate geometric model generation. These advantages are also helpful for high-quality physical simulation which will be implemented in future work.

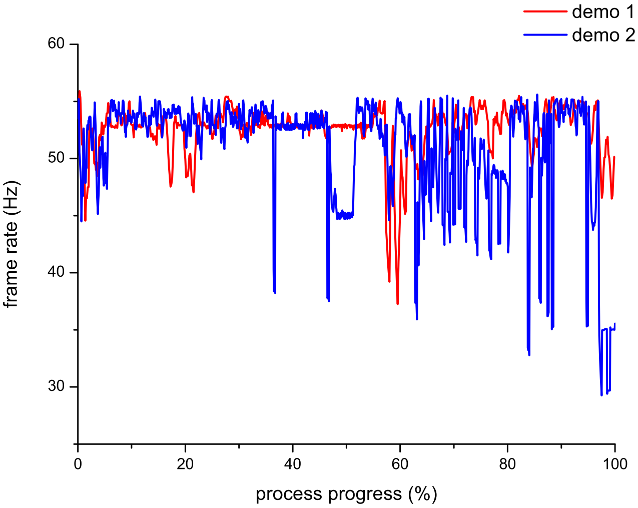


Fig. 13 Statistics of frame rate during simulation

6 Conclusion and future works

In feature-oriented machining simulation system, the algorithm is optimized through the idea of divide and conquer. It makes improvement in both efficiency and accuracy. Low level memory consumption and real-time visualization make this simulation system highly practical.

However, there are some limits for this simulation system. The input of this system must be a STEP-NC file that contains necessary machining process information. The application of STEP-NC standard is mainly limited to research works at present. Nonetheless, as the next generation of machining information exchange standard, the STEP-NC file will be supported widely in the future. A special case is not discussed in this paper that the geometry shapes of the two features are intersecting and different simulation methods cannot be used directly. This problem may be solved by dividing the machining features into smaller partition that is mutually independent.

The advantage of this paper is the ability to efficiently and accurately generate geometric model of machined workpiece. Based on this, more research works can be carried out in the future. For example, an efficient physical simulation can be developed using the result of feature-based method. Taking advantage of independent data structure in STEP-NC, tolerances and surface finish requirements are evaluated in physical simulation so that machining sequences can be inferred. This system is also valuable for online monitoring of machine tools by demonstrating geometry of machined workpiece to factory managers.

References

1. Cnc software inc. mastercam (2019). https://www.mastercam.com
2. Aras E, Yip-Hoi D (2012) State-of-the-art in geometric modeling for virtual machining. In: ASME 2012 international design engineering technical conferences and computers and information in engineering conference, pp 737–750
3. CAMotics (2018) https://camotics.org/manual.html
4. CGTech: Vericut (2018) http://www.cgtech.com/
5. EMC2 (2010) http://www.linuxcnc.org
6. Gul E, Joseph VR, Yan H, Melkote SN (2018) Uncertainty quantification of machining simulations using an in situ emulator. J Qual Technol 50(3, SI):253–261. https://doi.org/10.1080/00224065.2018.1474689
7. Huang Y, Oliver JH (1995) Integrated simulation, error assessment, and tool path correction for five-axis NC milling. J Manuf Syst 14(5):331–344
8. Joy J, Feng HY (2017) Efficient milling part geometry computation via three-step update of frame-sliced voxel representation workpiece model. Int J Adv Manuf Technol 92(3):1–14
9. Joy J, Feng HY (2017) Frame-sliced voxel representation: an accurate and memory-efficient modeling method for workpiece geometry in machining simulation. Comput Aided Des 88:1–13
10. Kadir AA, Xu X (2011) Towards high-fidelity machining simulation. J Manuf Syst 30(3):175–186
11. Karunakaran KP, Shringi R (2007) Octree-to-BRep conversion for volumetric NC simulation. Int J Adv Manuf Technol 32(1–2):116–131
12. Kobbelt LP, Botsch M, Schwanecke U, Seidel HP (2001) Feature sensitive surface extraction from volume data
13. Lee SW, Nestler A (2012) Virtual workpiece: workpiece representation for material removal process. Int J Adv Manuf Technol 58(5–8):443–463
14. Li JG, Ding J, Gao D, Yao YX (2010) Quadtree-array-based workpiece geometric representation on three-axis milling process simulation. Int J Adv Manuf Technol 50(5–8):677–687
15. Li JG, Yao YX, Xia PJ, Liu CQ, Wu CG (2008) Extended octree for cutting force prediction. Int J Adv Manuf Technol 39(9-10):866–873
16. Liu SQ, Ong SK, Chen YP, Nee AYC (2006) Real-time, dynamic level-of-detail management for three-axis NC milling simulation. Comput Aided Des 38(4):378–391
17. Lukic D, Zivanovic S, Vukman J, Milosevic M, Borojevic S, Antic A (2018) The possibilities for application of step-nc in actual production conditions. J Mech Sci Technol 32(7):3317–3328
18. Miao Y, Song X, Jin T, Shan Y (2016) Improving the efficiency of solid-based NC simulation by using spatial decomposition methods. Int J Adv Manuf Technol 87(1–4):421–435
19. Nishida I, Okumura R, Sato R, Shirase K (2018) Cutting force simulation in minute time resolution for ball end milling under various tool posture. J Manuf Sci Eng, 140(2 SI)
20. O'onnell J, Jablokow (1993) Construction of solid models from nc machining programs. ASME Production Engineering Division (Publication) 64:157
21. Rivièrelorphèvre E, Huynh HN, Verlinden O (2018) Influence of the time step selection on dynamic simulation of milling operation. Int J Adv Manuf Technol 95(7):1–16
22. Roy U, Xu Y (1998) 3-D object decomposition with extended octree model and its application in geometric simulation of NC machining. Robot Comput Integr Manuf 14(4):317–327
23. Spence AD, Altintas YA (1994) A solid modeller based milling process simulation and planning system. J Eng Indust 116(1):61
24. Sullivan A, Erdim H, Perry RN, Frisken SF (2012) High accuracy NC milling simulation using composite adaptively sampled distance fields. Comput Aided Des 44(6):522–536
25. Sun YJ, Yan C, Wu SW, Gong H, Lee CH (2018) Geometric simulation of 5-axis hybrid additive-subtractive manufacturing based on tri-dexel model. Int J Adv Manuf Technol 99(9–12):2597–2610
26. Takata S, Tsai MD, Inui M, Sata T (1989) A cutting simulation system for machinability evaluation using a workpiece model. CIRP Ann Manuf Technol 38(1):417–420
27. Wang Xiaoyu Chen Jihong TX (2010) Component implementation technology of wire-frame simulation based on vtk. Comput Eng 36(4):226–228
28. Weik MH (2001) Wire-frame modeling. Springer, US
29. Xu X (2006) Realization of step-nc enabled machining. Robot Comput Integr Manuf 22(2):144–153
30. Yau HT, Tsou LS (2009) Efficient NC simulation for multi-axis solid machining with a universal APT cutter. J Comput Inform Sci Eng 9(2):375–389
31. Yip-Hoi D, Huang X (2006) Cutter/workpiece engagement feature extraction from solid models for end milling. J Manuf Sci Eng 128(1):339–349
32. Zhang Y, Xu X, Liu Y (2011) Numerical control machining simulation: a comprehensive survey. Int J Comput Integr Manuf 24(7):593–609

33. Zhao G, Liu Y, Xiao W, Zavalnyi O, Zheng L (2018) Step-compliant cnc with t-spline enabled toolpath generation capability. The International Journal of Advanced Manufacturing Technology

34. Zhou Xiangdong TX (2009) Numerical control machining wireframe simulation based on vtk. Mach Tool Hydraul 37(7):208–210

35. Zhu W, Hu T, Luo W, Yan Y, Zhang C (2018) A step-based machining data model for autonomous process generation of intelligent cnc controller. Int J Adv Manuf Technol 96(1-4):271–285

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.