Automated Process Planning System for End-Milling Operation by CAD Model in STL Format

Nishida, Isamu
Shirase, Keiichi

(Citation)
International Journal of Automation Technology,15(2):149-157

(Issue Date)
2021-03-05

(Resource Type)
journal article

(Version)
Version of Record

Paper:

Automated Process Planning System for End-Milling Operation by CAD Model in STL Format

Isamu Nishida† and Keiichi Shirase

Kobe University
1-1 Rokko-dai, Nada-ku, Kobe, Hyogo 657-8501, Japan
†Corresponding author, E-mail: nishida@mech.kobe-u.ac.jp
[Received August 11, 2020; accepted October 16, 2020]

A method for extracting the machining region from a 3D CAD model in Standard Triangulated Language (STL) format and automatically generating a tool path is proposed. First, a method is proposed for extracting the machining region and obtaining the geometrical features such as a convex or concave shape from only the 3D CAD model in STL format. The STL format uses only triangular mesh data and drops all information, which is necessary for extracting the removal volume for the machining and geometrical characteristics. Furthermore, the triangular mesh size is non-uniform. A contour line model is proposed in which the product model is minutely divided on the plane along any one axial direction and is represented by points at intervals below the indicated resolution obtained from the contour line of the cross section of the product. Subsequently, a method is proposed to determine the machining conditions for each extracted machining region and automatically generate a tool path according to the geometrical features of the machining region obtained. A machining experiment was conducted to validate the effectiveness of the proposed method. As a result of the machining experiment, it was confirmed that the tool path automatically generated from the 3D CAD model in STL format can be machined without any problems and with a practical level of accuracy.

Keywords: CAM, STL format, end milling, process planning

1. Introduction

While industry is shifting from mass production to small volume production or customized production, the automation of preparation tasks such as NC program generation is strongly desired. The time and cost for preparation tasks, such as NC program generation, directly affect the production efficiency and cost for customized production. Some previous studies have attempted to reduce the time required to generate an NC program, and have realized digital copy milling without such a program [1–4] or computer-aided process planning (CAPP) systems. To achieve automation of the NC program generation, it is necessary to recognize the machining shape and determine the machining conditions. To recognize the machining shape, a computer is not so sophisticated that it cannot understand the geometric features of a complicated product shape. It is therefore necessary to recognize the machining region and features of such shape. There are two main methods for machining feature recognition: recognizing the target shape [5–9] and recognizing the removal volume [10–19]. Furthermore, to achieve automation of NC program generation, it is necessary to automatically determine the machining conditions, which depends significantly on the know-how of experienced operators because conditions such as the feed speed and the axial and radial depths of the cut significantly affect the product efficiency and cost. Some previous studies have attempted to automatically determine the machining conditions and others have realized the automatic determination of the machining conditions by depending on the geometric features of the product shape [20–22]. Other studies have realized this by reusing machining case data [23, 24]. Although some previous studies have supported the NC program generation, it is difficult to apply such generation in the field because the product shape is usually too complicated to recognize the machining features, and the combination among the workpiece material, machine tool, and cutting tool is not always constant. Furthermore, the diversity of the 3D CAD model format is one of the major problems in realizing automated NC program generation. Various CAD software packages are used in the field. Owing to the difference in these data formats, the 3D CAD data designed by the ordering source are often manually re-created by different CAD software used at the manufacturing site with data compatibility.

This study focuses on the data format of a 3D CAD model and proposes a method for extracting the machining region and geometrical features from the 3D CAD model in STL format and automatically generating a tool path. First, a method is proposed for extracting the machining region and obtaining the geometrical features, such as convex or concave shapes, from only the 3D CAD model in STL format. Subsequently, this study proposes a method for determining the machining conditions for each extracted machining region and automatically generating a tool path according to the geometrical features

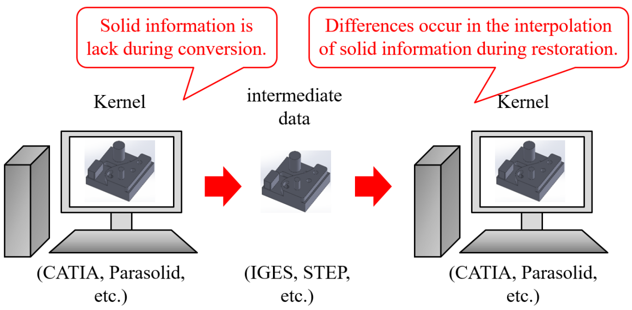


Fig. 1. Lack of information when converting between kernel and intermediate data.

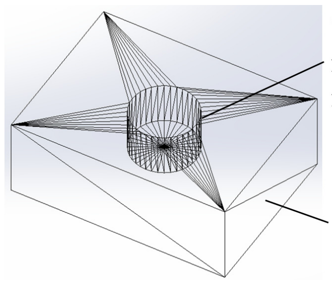


Fig. 2. STL format.

of the machining region obtained. In this study, a machining experiment is conducted to validate the effectiveness of the proposed method.

2. Automated Tool Path Generation by CAD Model in STL Format

2.1. Three-Dimensional Shape Representation Using Contour Line Model

In general, the difference in format of 3D CAD data is a major issue when handling such data. The data format of a 3D CAD model can be classified into three types, i.e., kernel data, intermediate data, and mesh data, depending on the amount of information and the features of the compatibility. Kernel data are in a format that depends on CAD software, and is the core in creating a three-dimensional model. Such data hold the points, lines, surfaces, and solid information, and have a large amount of information. The CAD data created by different kernels are incompatible in their present form, but can be converted into intermediate data. Such intermediate data can hold the CAD data created by different kernels in a format that does not depend on the data format. In general, the intermediate data store only the surface information, including points and lines that construct a 3D shape. The solid information held by the kernel data is dropped. The mesh data are the data in which only the points and lines are stored. In the mesh data, the surface information is represented by a set of minute triangles. The surface information held by the intermediate data is dropped.

Various CAD software packages are used in the field. Owing to the difference in these data formats, the 3D CAD data designed by the ordering source are often manually re-created by different CAD software used at the manufacturing site. In addition, as shown in **Fig. 1**, there is a lack of information when converting the kernel data that depend on CAD software into intermediate data, and thus the more complex the shape is, the greater the difference from the original shape that may occur during the restoration. Therefore, it takes time and effort to re-create the CAD data at the manufacturing site.

The mesh data are also in a Standard Triangulated Language (STL) format because they represent the shape with a set of triangular elements. Owing to the representation of triangular elements, the format is highly versatile. However, all historical information when creating the CAD model is removed. In other words, this data format drops all information necessary to extract the removal volume for machining as well as the geometrical characteristics. In this study, a method to analyze only the STL format and overcome the problems related to the extraction of the machining region is proposed to overcome the difficulties related to the difference in data format.

In the STL format, a plane is represented by much larger triangular meshes, and a curved surface is represented by much smaller triangular meshes in order to represent surfaces with the minimum number of triangular meshes, as shown in **Fig. 2**.

Our research group has proposed a method called a contour line model, which represents the surface of the product shape as discrete points at intervals below the indicated resolution to overcome the non-uniformity of the triangular meshes [25, 26]. The contour line model is one in which the product model is minutely divided on the plane along any one axial direction, and the contour line of the cross section is superimposed. In the STL format, the triangular mesh size is non-uniform, and thus the distance between the intersections with the triangular mesh obtained on the divided plane is not constant. Therefore, the product surface needs to be represented by points at intervals below the indicated resolution.

The algorithm for the contour line model is described as follows. The product model expressed by the STL format is divided finely on the XY plane at intervals of the indicated resolution, as shown in **Fig. 3**. For each surface of the divided XY plane, the intersection between the plane and all triangular meshes is geometrically extracted. The line segment, where the triangular mesh and the target plane intersect, is calculated. The contour line of the divided plane is calculated by searching for a line with the same start of the end point and connecting the same points, as shown in **Fig. 4**. The line segments composing the contour line obtained are then of non-uniform lengths, as shown in **Fig. 5**. In particular, a plane of the shape is represented by a single long line segment, whereas a curved surface is represented by a plurality of minute line segments. Therefore, when the length of each line segment is longer than the indicated resolution, the points are inserted using a linear interpolation. When the length of

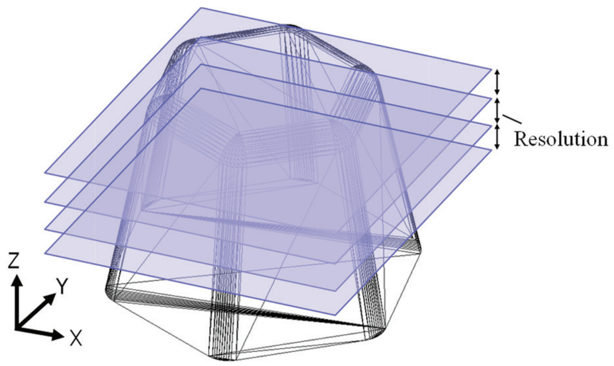


Fig. 3. Product model expressed using STL format divided finely on the XY plane at intervals of the resolution.

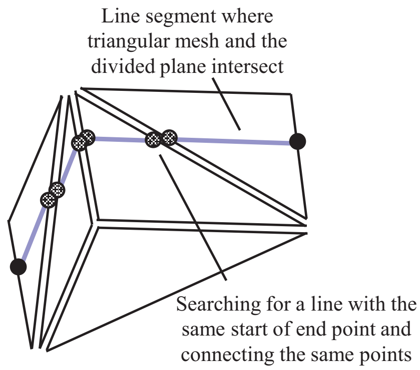


Fig. 4. Geometrical relationship between triangular mesh and the divided plane.

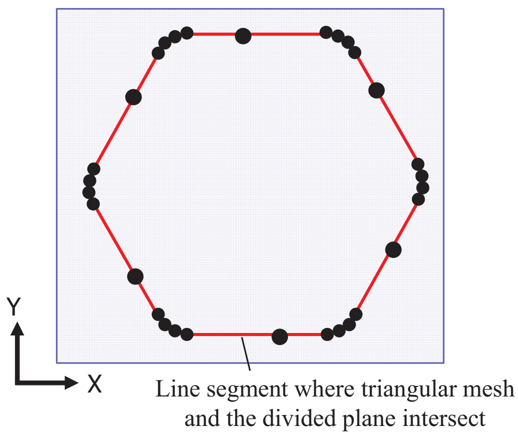


Fig. 5. Contour line obtained from the line segment where the triangular mesh and divided plane intersect.

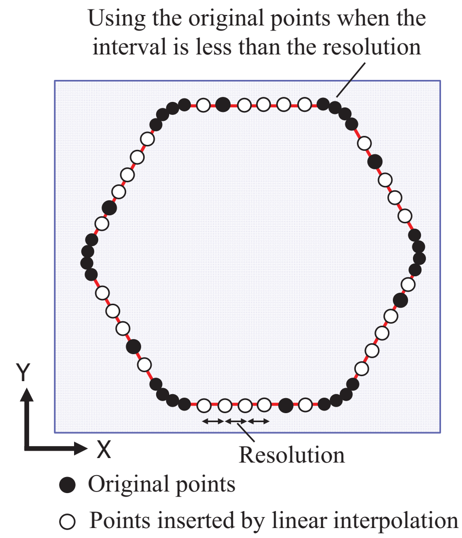


Fig. 6. Contour line expressed using points at intervals below the resolution.

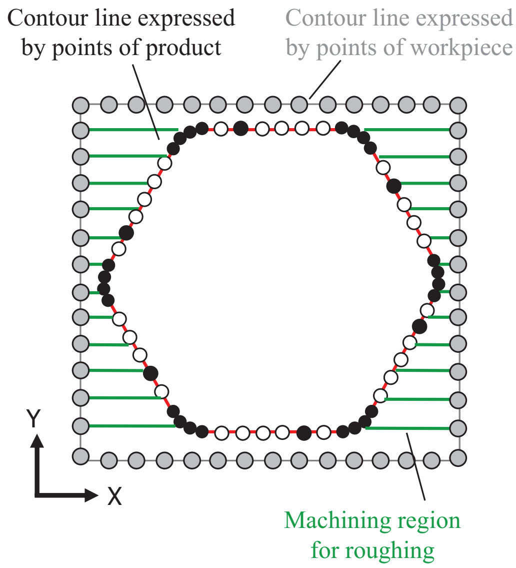


Fig. 7. Machining region obtained from the difference in the points of the contour line between workpiece and product.

the line segment obtained from the STL format is less than the resolution indicated, it is expressed using the line segment shown in **Fig. 6**. As described above, the product surface expressed using a non-uniform triangular mesh can be expressed by points at intervals below the indicated resolution. The accuracy of this model depends on the maximum value of the tolerance specified when expressing the 3D CAD model in STL format and the indicated resolution of the contour line model.

2.2. Extraction of Machining Region and Features by Contour Line Model

During the roughing, the machining region can be easily calculated from the difference between the points of the contour line obtained from each plane divided at the minute intervals of the workpiece and those of the product, as shown in **Fig. 7**.

However, when applying a finishing for convex regions, drilling for holes, or finishing or roughing for concave regions that are independently machined, it is necessary to extract the machining region independently. Therefore, in this study, a method to extract these machining regions from a 3D CAD model in STL format is proposed.

First, the contour line is obtained from each plane divided finely along the XY plane at minute intervals of the

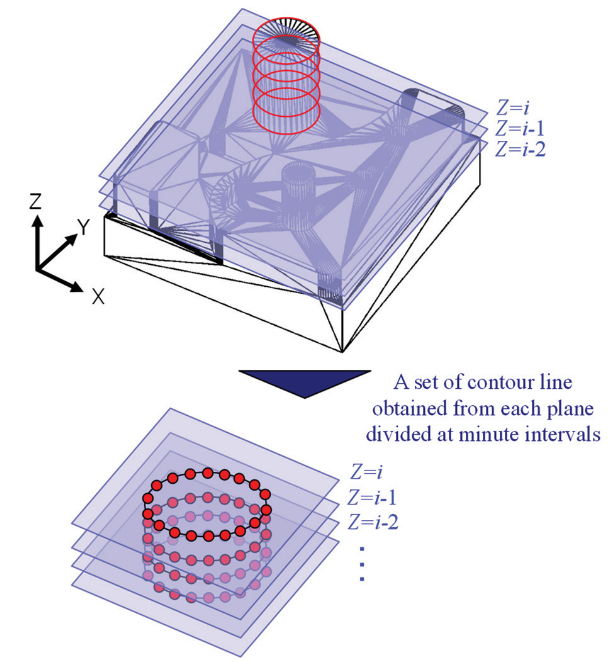


Fig. 8. Extraction of machining region from contour lines obtained from each plane divided along the *XY* plane.

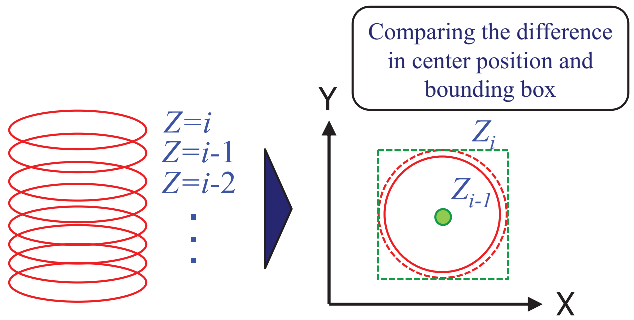


Fig. 9. Determination of independent convex or concave machining region based on the difference in the bounding box of the contour line on each divided plane.

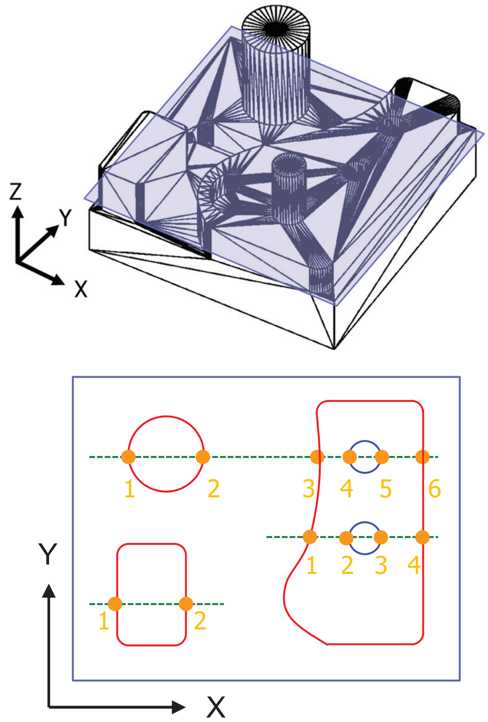


Fig. 10. Determination of geometrical features (convex or concave) of the machining region.

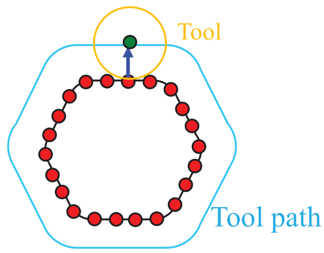


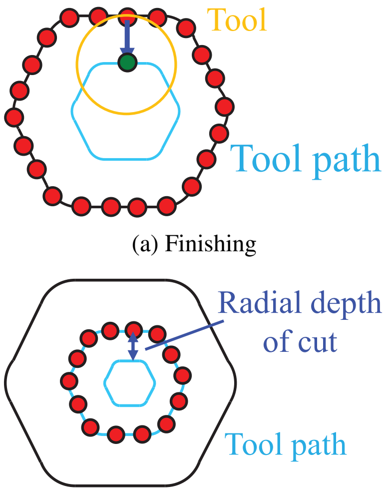
Fig. 11. Tool path for finishing of convex machining region.

product. The contour line on each divided plane is independently obtained for each closed loop, as shown in **Fig. 8**. Subsequently, if the difference at the center position of the bounding box of the contour line between *Z*-values and the difference of the bounding box of the contour line are within the threshold, it is determined as an independent convex or concave machining region, as shown in **Fig. 9**. Through repetition to obtain the contour lines until the *Z*-value of the divided plane is at minimum, it is possible to obtain all independent convex or concave machining regions.

Although it is possible to obtain an independent machining region as described above, it is not easy to determine whether the machining region obtained is convex or concave because the STL format consists of triangular mesh data. Thus, this study aims to determine the geometrical features of the machining region obtained as follows. First, the intersection of a straight line parallel to the *X*-axis at the *Y*-value of the center of the bounding box of the machining region and the contour line is calculated for the *XY* plane at any *Z*-value where the obtained machining region exists. Subsequently, the intersections obtained are numbered from the smallest *X*-value, as shown in **Fig. 10**. When the number of intersections with the target contour line is odd, the machining region can be determined to have a convex shape. By contrast, when the number of intersections with the target contour line is even, the machining region can be determined as a concave shape. By applying these processes for all machining regions obtained, the machining region and its features can be extracted from the 3D CAD model in STL format.

2.3. Tool Path Generation

When the obtained machining region is a convex shape, the tool path for finishing can be calculated by offsetting the obtained contour line toward the outside based on the radius of the tool, as shown in **Fig. 11**. When the obtained machining region has a concave shape, the tool path for



(a) Finishing

(b) Roughing

Fig. 12. Tool path of concave machining region.

the finishing can be calculated by offsetting the obtained contour line toward the inside using the radius of the tool, as shown in **Fig. 12(a)**. When the obtained machining region has a concave shape, the tool path used for roughing can also be calculated by offsetting the points representing one outer tool path with intervals toward the inside based on the radial depth of the cut, as shown in **Fig. 12(b)**. Drilling can be conducted when the obtained machining region is a concave shape and the tool approaching surface is a circle. Although it is also possible to calculate the tool path for roughing by offsetting the points representing one inter-tool path with intervals toward the outside based on the radial depth of the cut when the obtained machining region has a convex shape, it is difficult to calculate the tool path for roughing within a product shape with multiple convex regions, as shown in **Fig. 10**, owing to interference with the tool. Therefore, when the product shape has multiple convex regions, the roughing is conducted using a scanning operation.

The tool paths for the scanning operation are calculated based on the amount of interference between the product surface represented by the contour line model obtained from each plane divided finely on the *ZX* plane at minute intervals of the product and tool. The tool position can be determined by calculating the amount of interference between the product surface and the tool in the *Z*-axis direction at every minute interval with respect to the tool feed direction, and offsetting the maximum value of the interference amount in the *Z*-axis direction. First, the tool is divided by the planes with the same interval as the product, as shown in **Fig. 13**. In the case of a square-end mill, the tool is shown as a rectangle on the plane. Next, the amount of interference on each divided plane is calculated by detecting the product of the surface points existing in the tool, as shown in **Fig. 14**. The maximum amount of interference is calculated from the amount of interference on each divided plane. The tool position can be calculated by offsetting the tool in the *Z*-axis direction by the

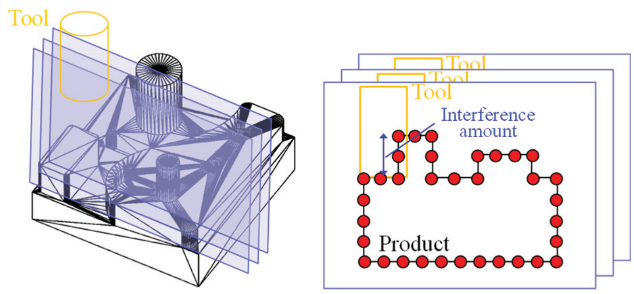


Fig. 13. Description of product and tool with contour line model for scanning operation.

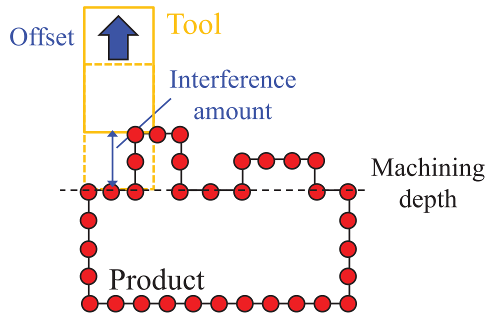


Fig. 14. Tool offset according to the calculated amount of interference.

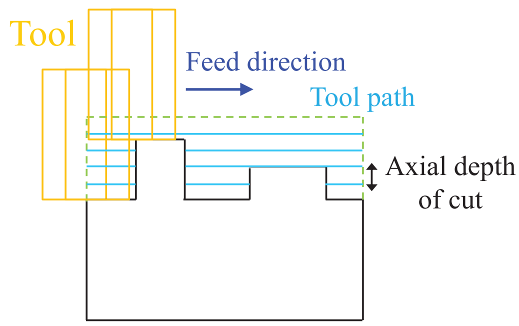


Fig. 15. Tool path for scanning operation obtained by calculating the tool position at each minute interval.

maximum amount of interference obtained. By calculating the tool position for every minute interval with respect to the tool feed direction, the tool path can be calculated as shown in **Fig. 15**.

2.4. Determination of Machining Conditions

The tool path can be calculated as described in the previous section when the machining conditions are determined for each machining region extracted. Here, the determination of the tool diameter for the region is described.

When the machining region is concave and its approaching surface is circular, the region is machined using a drill with the same diameter as that of the circle on the approaching surface. When the approaching surface is not a circle or there is no drill with the same diameter as

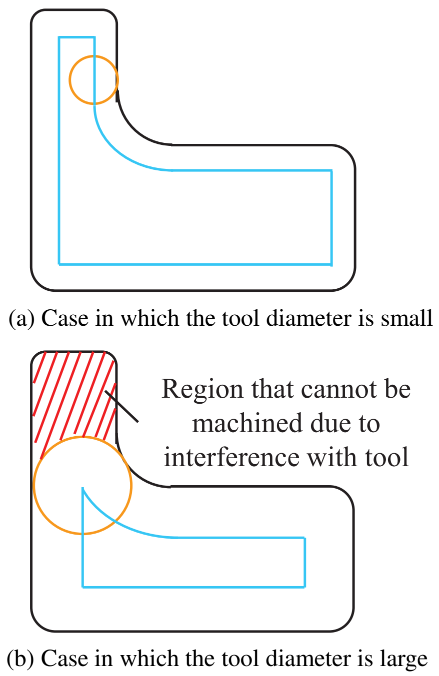


Fig. 16. Determination of tool diameter for concave machining region.

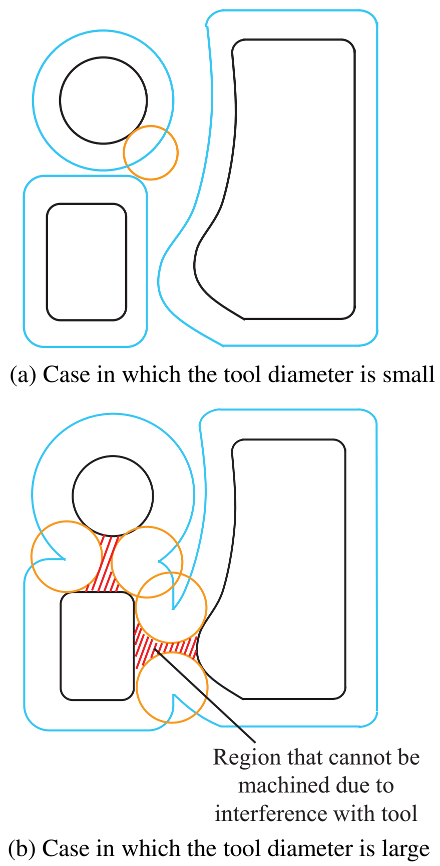


Fig. 17. Determination of tool diameter for convex machining region.

the diameter of the circle, contour machining is selected. Therefore, the tool with the largest tool diameter that can be machined without interfering with the contour line of the approaching surface obtained is selected, as shown in **Fig. 16**.

When the machining region is convex, a tool with the largest tool diameter that can be machined without interfering with the neighboring convex region is selected, as shown in **Fig. 17**. In the scanning used for roughing, the same tool diameter as that applied for a convex region is used because it is necessary to achieve a machining without interfering with any convex regions.

As described above, the machining region can be automatically extracted from the STL format of the product model by the contour line model, and the tool diameter for machining can be selected from the tool list. In this study, the machining conditions, such as the spindle speed, feed speed, and axial and radial depths of the cut, were determined by referring to our previous study [24], the parameters of which are determined from the machining case data by recognizing the most similar geometrical features of the machining region.

3. Case Study

To validate the method for extracting the machining region and features from the 3D CAD model in the STL format and generating a tool path, a tool path was generated for a product model, as shown in **Fig. 18**, and a machining experiment was conducted. **Fig. 18** also shows the machining region and features extracted by the proposed contour line model. **Table 1** lists the tools required

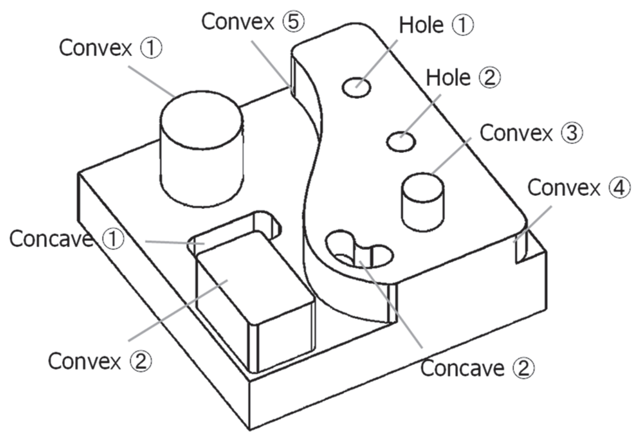


Fig. 18. 3D CAD model used in the case study.

Table 1. Tool list for case study.

No.	Diameter	Tool type
1	2 mm	Square end mill
2	4 mm	Square end mill
3	6 mm	Square end mill
4	4 mm	Flat drill
5	6 mm	Flat drill

Table 2. Extracted machining region and the tool determined.

Region	Tool No.	Region	Tool No.
Convex ①	3	Hole ①	4
Convex ②	2	Hole ②	4
Convex ③	3	Concave ①	1
Convex ④	2	Concave ②	1
Convex ⑤	2		

Table 3. Machining conditions for each tool.
(D, diameter; R_d, radial depth of cut; A_d, axial depth of cut; Spindle, spindle speed; Feed, feed speed)

No.	D [mm]	R_d [mm]	A_d [mm]	Spindle [min^{-1}]	Feed [mm/min]
1	2	0.5	0.25	10000	200
2	4	1.0	0.5	10000	200
3	6	2.0	0.5	10000	200
4	4	–	–	5000	100
5	6	–	–	5000	100

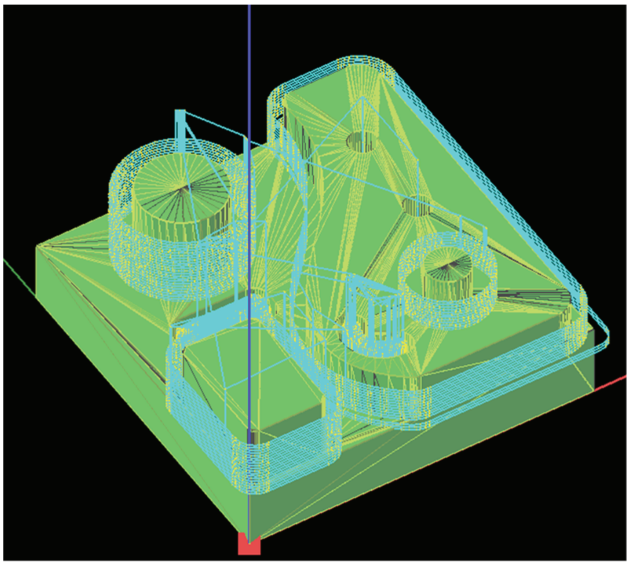


Fig. 19. Tool path generated for convex and concave regions.

to determine the machining conditions. **Table 2** shows the tools determined using the proposed method. **Table 3** shows the machining conditions for each tool when the workpiece material is aluminum alloy A2017. **Fig. 19** shows the tool path calculated from these machining conditions. The machining experiment was conducted using a vertical-type machining center (Modia Systems Co., Ltd., Mini Miller100). **Fig. 20** shows the results of the machining experiment using the generated tool path. **Table 4** shows the length in the X-, Y-, and Z-directions when comparing the CAD model and those models measured using digital calipers. According to the results shown in **Table 4**, it was confirmed that the tool path automatically generated from the STL format of the CAD model can

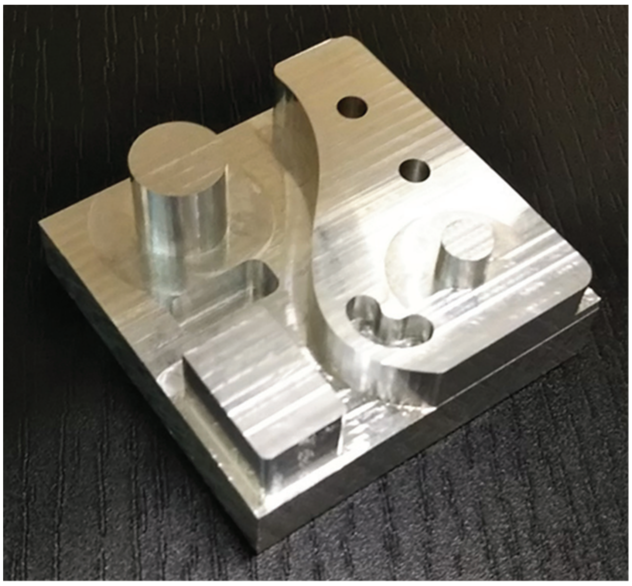


Fig. 20. Results of machining experiment for case study.

Table 4. Extracted machining region and the determined tool.
(NM, not measured)

Region	Model [mm]			Measured [mm]		
	X	Y	Z	X	Y	Z
Convex ①	ϕ12.00		12.00	ϕ11.94		12.01
Convex ②	11.00	17.00	10.00	10.99	16.98	10.02
Convex ③	ϕ6.00		6.00	ϕ5.96		6.00
Convex ④	NM	50.00	6.00	NM	49.94	5.94
Convex ⑤	NM	50.00	8.00	NM	49.94	7.96
Hole ①	ϕ4.00		8.00	ϕ3.95		8.01
Hole ②	ϕ4.00		8.00	ϕ3.95		8.02
Concave ①	14.00	6.00	6.00	13.96	5.97	6.04
Concave ②	NM	NM	5.00	NM	NM	5.05

be machined without any machining problems and with practical accuracy. It was also confirmed that a polygonal finished surface appears on the side wall of convex ① and convex ⑤. With the proposed method, a tool path is generated according to the polygon expressed by the STL format of the CAD model. However, when convex ① is intended to be a cylinder, a finished surface along the circle may be desired. A circular interpolation will be an area of our future research, and may be improved by using the circular interpolation function on the CNC controller.

4. Conclusions

This study proposed a method for extracting a machining region from the STL format of a 3D CAD model and generating a tool path. This study was conducted as follows.

1. The machining region can be extracted, and its geometrical features can be obtained from the STL format of the 3D CAD model by a three-dimensional shape

expressed by superimposing the contour line of the cross sections obtained by dividing the product along any one axial direction.

2. The machining conditions for each machining region extracted from the STL format of the 3D CAD model can be determined, and the tool path can be automatically generated.

3. The results of the machining experiment showed that the proposed method makes it possible to automatically generate a tool path that can be machined without any machining problems and with practical accuracy.

Acknowledgements

This work was partially supported by the Tateishi Science and Technology Foundation, Hyogo Prefecture COE Program, Suzuki Foundation, Mazak Foundation, and Kawanishi Memorial ShinMaywa Education Foundation.

References:

[1] K. Nakamoto, K. Shirase, H. Wakamatsu, A. Tsumaya, and E. Arai, "Development of Digital Copy Milling System to Realize NC Programless Machining: 3rd Report, Machining Strategy for In-Process Adaptation of Cutting Conditions," Trans. of the Japan Society of Mechanical Engineers, Series C, Vol.69, No.677, pp. 270-277, 2003.

[2] K. Shirase and K. Nakamoto, "Direct Machining Operation Performed by Autonomous NC Machine Tool Controlled by Digital Copy Milling Concept," Trans. of the Japan Society of Mechanical Engineers, Series C, Vol.74, No.743, pp. 1901-1906, 2008.

[3] K. Shirase and K. Nakamoto, "Simulation Technologies for the Development of an Autonomous and Intelligent machine Tool," Int. J. Automation Technol., Vol.7, No.1, pp. 6-15, 2013.

[4] K. Shirase, T. Kondo, M. Okamoto, H. Wakamatsu, and E. Arai, "Development of Virtual Copy Milling System to Realize NC Programless Machining: Real Time Tool Path Generation for Autonomous NC Machine Tool," Trans. of the Japan Society of Mechanical Engineers, Series C, Vol.66, No.644, pp. 298-303, 1999.

[5] D. Hamada, K. Nakamoto, T. Ishida, and Y. Takeuchi, "Development of CAPP system for multi-tasking machine tool," Trans. of the Japan Society of Mechanical Engineers, Series C, Vol.78, No.791, pp. 2698-2709, 2012 (in Japanese).

[6] L. Wang, M. Holm, and G. Adamson, "Embedding a process plan in function blocks for adaptive machining," CIRP Annals – Manufacturing Technology, Vol.59, Issue 1, pp. 433-436, 2010.

[7] Y. Woo, E. Wang, Y. S. Kim, and H. M. Rho, "A hybrid feature recognizer for machining process planning systems," CIRP Annals – Manufacturing Technology, Vol.54, Issue 1, pp. 397-400, 2005.

[8] K. Nakamoto, K. Shirase, H. Wakamatsu, A. Tsumaya, and E. Arai, "Automatic production planning system to achieve flexible direct machining," JSME Int. J., Series C, Vol.47, No.1, pp. 136-143, 2004.

[9] A. Ueno and K. Nakamoto, "Proposal of machining features for CAPP system for multi-tasking machine tools," Trans. of the JSME, Vol.81, No.825, 15-00108, doi: 10.1299/transjsme.15-00108, 2015 (in Japanese).

[10] E. Morinaga, M. Yamada, H. Wakamatsu, and E. Arai, "Flexible process planning method for milling," Int. J. Automation Technol., Vol.5, No.5, pp. 700-707, 2011.

[11] E. Morinaga, T Hara, H. Joko, H. Wakamatsu, and E. Arai, "Improvement of computational efficiency in flexible computer-aided process planning," Int. J. Automation Technol., Vol.8, No.3, pp. 396-405, 2014.

[12] K. Dwijayanti and H. Aoyama, "Basic study on process planning for turning-milling center based on machining feature recognition," J. of Advanced Mechanical Design, Systems and Manufacturing, Vol.8, No.4, JAMDSM0058, 2014.

[13] H. Sakurai and P. Dave, "Volume decomposition and feature recognition, part 1 – polyhedral objects," Computer-Aided Design, Vol.27, Issue 11, pp. 793-869, 1995.

[14] H. Sakurai and P. Dave, "Volume decomposition and feature recognition, part II – curved objects," Computer-Aided Design, Vol.28, Issues 6-7, pp. 519-537, 1996.

[15] T. Inoue and K. Nakamoto, "Proposal of a recognition method of machining features in computer aided process planning system for complex parts machining," Trans. of the JSME, Vol.83, No.850, 16-00574, doi: 10.1299/transjsme.16-00574, 2017 (in Japanese).

[16] M. M. Isnaini, Y. Shinoki, R. Sato, and K. Shirase, "Development of a CAD-CAM interaction system to generate a flexible machining process plan," Int. J. Automation Technol., Vol.9, No.2, pp. 104-114, 2015.

[17] I. Nishida, R. Sato, and K. Shirase, "Proposal of process planning system for end-milling operation considering product design constraints," The Institute of Systems, Control and Information Engineering, Vol.30, No.3, pp. 81-86, 2017 (in Japanese).

[18] I. Nishida, T. Hirai, R. Sato, and K. Shirase, "Automatic process planning system for end-milling operation considering CAM operator's intention," Trans. of the JSME, Vol.84, No.860, 17-00563, doi: 10.1299/transjsme.17-00563, 2018 (in Japanese).

[19] I. Nishida and K. Shirase, "Automated process planning system for end-milling operation considering constraints of operation (1st report Process planning to minimize the number of times of tool change)," Trans. of the JSME, Vol.84, No.866, 18-00242, doi: 10.1299/transjsme.18-00242, 2018 (in Japanese).

[20] Y. Shinoki, M. M. Isnaini, R. Sato, and K. Shirase, "Machining operation planning system which utilize past machining operation data to generate new NC program," Trans. of the Japan Society of Mechanical Engineers, Series C, Vol.81, No832, 15-00280, 2015 (in Japanese).

[21] M. El-Mehalawi and R. A. Miller, "A database system of mechanical components based on geometric and topological similarity. Part I: representation," Computer-Aided Design, Vol.35, No.1, pp. 83-94, 2003.

[22] M. El-Mehalawi and R. A. Miller, "A database system of mechanical components based on geometric and topological similarity. Part II: indexing, retrieval, matching and similarity assessment," Computer-Aided Design, Vol.35, No.1, pp. 95-105, 2003.

[23] S. Kobayashi, "Present and future of case-based reasoning," J. of Japanese Society for Artificial Intelligence, Vol.7, No.4, pp. 559-566, 1992 (in Japanese).

[24] I. Nishida and K. Shirase, "Automatic Determination of Cutting Conditions for NC Program Generation by Reusing Machining Case Data based on Geometric Properties of Removal Volume," J. of Advanced Mechanical Design, Systems, and Manufacturing, Vol.12, No.4, JAMDSM0093, doi: 10.1299/jamdsm.2018jamdsm0093, 2018.

[25] I. Nishida and K. Shirase, "Proposal of contour line model for high-speed end milling simulation," Int. J. Automation Technol., Vol.14, No.1, pp. 38-45, 2020.

[26] I. Nishida and K. Shirase, "Machining time reduction by tool path modification to eliminate air cutting motion for end milling operation," Int. J. Automation Technol., Vol.14, No.3, pp. 459-466, 2020.

Name:
Isamu Nishida

Affiliation:
Assistant Professor, Department of Mechanical Engineering, Graduate School of Engineering, Kobe University

Address:
1-1 Rokko-dai, Nada-ku, Kobe, Hyogo 657-8501, Japan

Brief Biographical History:
2012- Sysmex Corp.
2016- Assistant Professor, Kobe University
2018- CEO, BESTOWS Co., Ltd. (Concurrently)

Main Works:
• "Machining time reduction by tool path modification to eliminate air cutting motion for end milling operation," Int. J. Automation Technol., Vol.14, No.3, pp. 459-466, 2020.
• "Machine tool assignment realized by automated NC program generation and machining time prediction," Int. J. Automation Technol., Vol.13, No.5, pp. 700-707, 2019.
• "Sequence planning of on-machine measurement and re-machining," J. of Advanced Mechanical Design, Systems, and Manufacturing, Vol.13, No.1, JAMDSM0014, 2019.

Membership in Academic Societies:
• Japan Society of Mechanical Engineers (JSME)
• Japan Society for Precision Engineering (JSPE)

Name:
Keiichi Shirase

Affiliation:
Professor, Department of Mechanical Engineering, Graduate School of Engineering, Kobe University

Address:
1-1 Rokko-dai, Nada-ku, Kobe, Hyogo 657-8501, Japan

Brief Biographical History:
1984- Research Associate, Kanazawa University
1995- Associate Professor, Kanazawa University
1996- Associate Professor, Osaka University
2003- Professor, Kobe University

Main Works:
• K. Shirase and K. Nakamoto, "Simulation Technologies for the Development of an Autonomous and Intelligent Machine Tool," Int. J. Automation Technol., Vol.7, No.1, pp. 6-15, 2013.
• T. Kobayashi, T. Hirooka, A. Hakotani, R. Sato, and K. Shirase, "Tool Motion Control Referring to Voxel Information of Removal Volume Voxel Model to Achieve Autonomous Milling Operation," Int. J. Automation Technol., Vol.8, No.6, pp. 792-800, 2014.
• M. M. Isnaini, Y. Shinoki, R. Sato, and K. Shirase, "Development of CAD-CAM Interaction System to Generate a Flexible Machining Process Plan," Int. J. Automation Technol., Vol.9, No.2, pp. 104-114, 2015.
• K. Shirase, "CAM-CNC integration for innovative intelligent machine tool," Proc. of the 8th Int. Conf. on Leading Edge Manufacturing in 21st Century (LEM21), A01, 2015.
• I. Nishida, R. Okumura, R. Sato, and K. Shirase, "Cutting Force Simulation in Minute Time Resolution for Ball End Milling Under Various Tool Posture," ASME J. of Manufacturing Science and Engineering, Vol.140, No.2, 021009, 2018.

Membership in Academic Societies:
• American Society of Mechanical Engineers (ASME)
• Society of Manufacturing Engineers (SME)
• Japan Society of Mechanical Engineers (JSME), Fellow
• Japan Society for Precision Engineering (JSPE), Fellow