



A Level-set Non-uniform Offset Method for Iso-scallop Toolpath Planning Based on Mesh Surface Conformal Mapping

Yue-Feng Li¹ , Qing-Hui Wang²  and Wen-Feng Gan³ 

¹ South China University of Technology, lucaslyf@foxmail.com

² South China University of Technology, wqh@scut.edu.cn

³ ZWSOFT CO., LTD (Guangzhou), gwf421@zju.edu.cn

Corresponding author: Qing-Hui Wang, wqh@scut.edu.cn

Abstract. The mesh surface conformal mapping algorithm has been already used in the field of toolpath planning due to its dimensionality reduction properties. Based on this algorithm, toolpath planning can be carried out in two-dimensional (2D) domain, avoiding many three-dimensional (3D) geometric calculations. The key problem in planning iso-scallop toolpath in 2D domain is to realize the non-uniform offset of point sequence curves (PS-curves), which is difficult to be solved. To address this issue, a level-set non-uniform offset method is proposed. This method makes use of the implicit expression characteristics of level-set curve motion and does not need to consider the calculation of effective loop in the PS-curves non-uniform offset. Based on the calculation method of non-uniform offset parameters proposed in this paper, the level-set non-uniform offset method can plan the high-precision iso-scallop toolpath in 2D domain. Then, the 2D toolpath is inversely mapped back to the 3D surface to generate the final iso-scallop toolpath. Finally, the toolpath fast evaluation method is used to evaluate the generated path, and the results verify the effectiveness of this method.

Keywords: Toolpath planning, Conformal mapping, Level-set, Mesh surface, Non-uniform Offset.

DOI: <https://doi.org/10.14733/cadaps.2023.1288-1299>

1 INTRODUCTION

The most traditional method for free-form surface machining path generation is based on parametric surface [3] [9] but it only applies to a single surface patch [5]. Compared with parametric surface, mesh surface is simpler and more robust. More importantly, mesh surface can be applied to composite surface. As a result, the mesh surface representation model has received a lot of attention and is widely used for toolpath planning [6] [10].

The conformal mapping algorithm based on mesh surface can map 3D mesh surface to 2D mesh plane, so that toolpath planning can be carried out in 2D domain. Although this method decreases the dimension of toolpath planning from 3D to 2D and considerably reduces computational

complexity, it still requires a high number of 2D intersection calculations in PS-curves non-uniform offset algorithm. Most of the existing PS-curves offset methods are based on the premise of uniform offset and are not applicable for non-uniform offset [1]. Compared with uniform offset, non-uniform offset will produce more complex self-intersection problems. A non-uniform offset method based on tracking is proposed in reference [2], which needs to traverse all offset points and intersection points in each offset process, with a huge amount of calculation, and still needs some additional pre-processing and post-processing to eliminate special local invalid loops.

Level-set is a method for calculating and analyzing interface motion proposed by S. Osher and J. A. Sethian [12]. It is simple, universal, and easy to expand to higher dimension space. At present, it is widely used in many fields, such as shape optimization, material deposition, curvature movement, path planning, medical image processing and so on [13]. In the planar toolpath planning, level-set function can naturally deal with the problem of 2D PS-curves self-intersection in the process of path offset. Hong et al. used level-set for isometric filling of three cross sections, simple polygon, multi-island polygon and multi-hole polygon [4]. Xiong et al. employed level-set to plan toolpaths for lightweight structures with thin walls and slender features [14]. Both of these works generated uniform offset tool paths.

In this work, based on the conformal mapping algorithm, a level-set non-uniform offset method is proposed to replace the 2D PS-curves non-uniform offset algorithm, to avoid the complex self-intersection problems of 2D PS-curves. Firstly, the 2D mesh plane is generated by using the conformal mapping algorithm, and a series of discrete points are generated by equidistantly dispersing the plane's area. To calculate the mapping stretch factor, the shortest distance direction from the discrete points to the 2D mesh plane boundary is used to approximate the 2D line spacing direction. Then, the approximate stretching mapping coefficient is used to calculate the 2D line spacing value at each discrete point, and the value is used as the normal velocity in the level-set function. In this way, the non-uniform offset based on level-set is realized, and iso-scallop toolpath can be planned in 2D domain. Then, the final toolpath can be generated by inverse mapping the 2D toolpath back to the 3D mesh surface. Finally, a fast evaluation algorithm is used to quickly evaluate the generated toolpath. This toolpath planning method is applicable to most free-form mesh surfaces and has higher robustness than other methods using PS-curves non-uniform offset algorithm.

2 TOOLPATH PLANNING BASED ON MESH SURFACE CONFORMAL MAPPING

The basic strategy of toolpath planning based on conformal mapping of mesh surface is shown in Figure 1. Firstly, the appropriate conformal mapping algorithm of mesh surface is used to map the 3D mesh surface M_{3D} into a mesh plane M_{2D} , then use the mapping distortion compensation (MDC) method to convert the 3D toolpath parameters into the 2D domain and plan the 2D toolpath on the plane mesh [2]. In this way, the dimension reduction characteristics of mesh surface conformal mapping algorithm are fully utilized, and the toolpath can be planned directly in the 2D domain according to the demand, which reduces the complexity of geometric calculation and improves the robustness. In addition, the mature and rich 2D related technologies can be directly used to solve the corresponding problems in toolpath planning. For the last inverse mapping step, it is necessary to determine the triangle to which each 2D toolpath point belongs, and then the inverse mapping can be carried out in two ways: the way of using area coordinates and the way of calculating the linear mapping matrix of each triangle, among which the way of using area coordinates is the most common, simple, and efficient [15].

In the toolpath planning based on conformal mapping, to meet the process requirements of iso-scallop height, it is necessary to calculate the corresponding 3D line spacing and mapping deformation at each toolpath point. This makes each 2D toolpath point have different offset distance. Therefore, the key problem of iso-scallop toolpath planning in 2D domain based on conformal mapping algorithm is to realize the non-uniform offset of PS-curves. Compared with uniform offset, non-uniform offset will produce more complex self-intersection problems, which requires additional post-processing to calculate the effective loop.

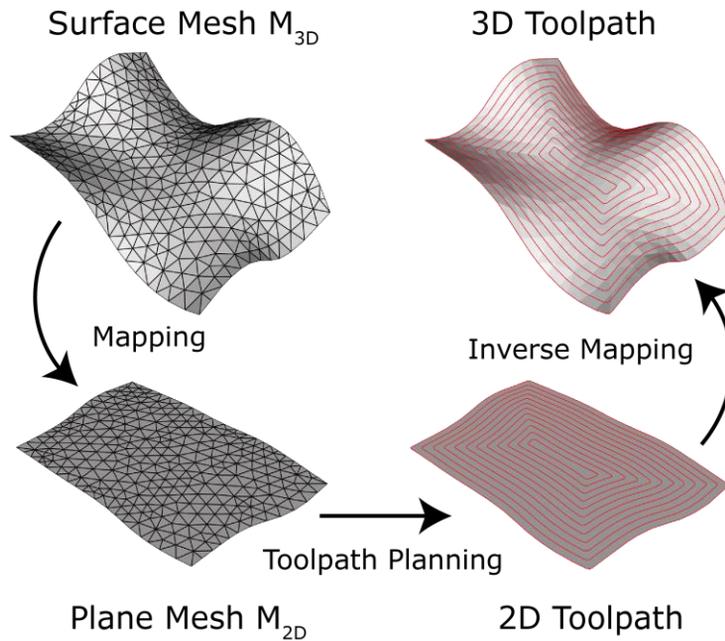


Figure 1: Toolpath planning based on conformal mapping.

In the PS-curves offset algorithm, the effective loop is usually judged based on three ways: (1) based on the rotation direction of the loop; (2) based on the number of offset points or segments contained in the loop; (3) distance based. Due to the complex self-intersection of non-uniform offset, it is often necessary to combine the above three methods to adopt multiple judgment. The first judgment is to use the rotation direction. The initial loop toolpath is treated as counterclockwise rotation. For all sub loop toolpaths, only those whose rotation direction is still counterclockwise are retained, and those clockwise are invalid. As shown in Figure 2 is a complex self-intersection situation. After the first judgment, L_1 , L_2 and L_3 are retained. At this time, the invalid loop L_2 can be removed by using the third way. However, sometimes there are multiple efficient loops, it is difficult to correctly judge whether loop L_3 is an invalid loop. It is necessary to use various methods to judge the invalid loop in different situations, which reduces the algorithm's robustness and efficiency.

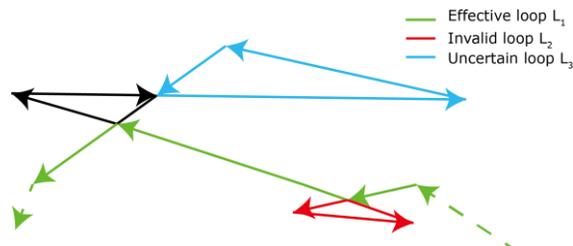


Figure 2: Loop self-intersection in non-uniform offset.

After calculating the effective loop correctly, the extracted effective loop needs to be smoothed. Especially for the case of non-uniform offset, some distorted and sharp local shapes will appear in the offset PS-curves, as shown in Figure 3.

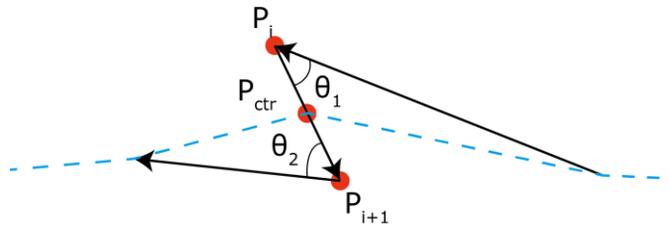


Figure 3: Sharp local shape in non-uniform offset.

Improper handling of such sharp problems will seriously affect the subsequent toolpath accuracy, and the scallop height of the toolpath changes greatly at the sharp shape points. The sharp points can be determined by judging the angle θ_1 and angle θ_2 , then replace point P_i and point P_{i+1} with the midpoint P_{ctr} . In this process, θ_1 and θ_2 will affect the overall toolpath accuracy. To generate high-quality iso-scallop toolpath, they need to be modified according to different models, which reduces the robustness of the algorithm. The double arc approximation method has better results, but it will reduce computational efficiency.

The level set non-uniform offset algorithm proposed in this paper does not need to consider the intersection of PS-curves offset process. In theory, it is also applicable for multi boundary model (with inner hole). Therefore, it is simpler and more robust.

3 NON-UNIFORM OFFSETTING BASED ON LEVEL-SET

At present, this method is suitable for three-axis and five-axis finishing toolpath planning of ball-end cutter, and the input surface adopts STL format. To facilitate understanding, the level-set algorithm will be briefly described next, and the key contents of this paper will be described in detail: the implementation of non-uniform offset of level-set and the calculation of offset parameters based on mapping stretch factor.

3.1 Implementation of Non-uniform Offsetting Based on Level-set

The level-set function has the following properties:

$$\begin{aligned} \varphi(x, t) &> 0 \text{ for } x \in \Omega \\ \varphi(x, t) &< 0 \text{ for } x \notin \bar{\Omega} \\ \varphi(x, t) &= 0 \text{ for } x \in \partial\Omega = \Gamma(t) \end{aligned} \quad (3.1)$$

where $\varphi(x, t)$ is the level-set function, $x = x(x_1, \dots, x_n) \in R^n$, Ω is an open area and $\bar{\Omega}$ is a closed area.

When the interface $\Gamma(t)$ is constantly moving, the motion interface $\Gamma(t)$ can be captured by finding where $\varphi = 0$. The topological change is implied in the change of value φ , so that the intersection problem in the path offset procedure does not need to be considered.

Symbolic distance function is the most used level-set function, which can be expressed as:

$$\varphi(x, t = 0) = \pm d(x) \quad (3.2)$$

where $d(x)$ is the minimum distance from point x to the closed interface $\Gamma(0)$. When point x is inside interface $\Gamma(0)$, the sign of $d(x)$ is negative. When point x is outside interface $\Gamma(0)$, the sign of $d(x)$ is positive.

The starting interface $\Gamma(0)$ is embedded in a higher one-dimensional level-set function φ as a zero level-set. Then, the interface motion problem is turned into an initial value problem, which uses the zero level-set of a time-varying function to calculate the motion of the interface $\Gamma(t)$.

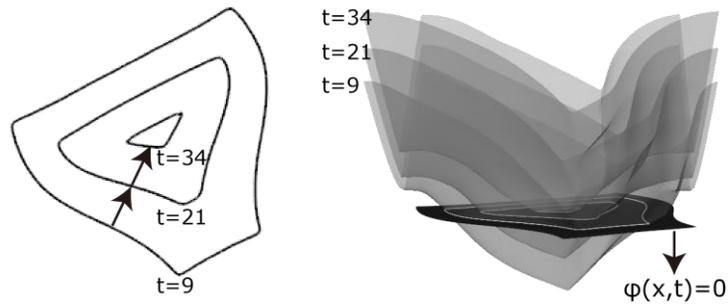


Figure 4: Level-set function and boundary propagating.

As shown in Figure 4, the zero level-set of the function is always consistent with the motion interface $\Gamma(t)$, which can be expressed as:

$$\varphi(x, t) = 0 \tag{3.3}$$

through the chain rule, it can be calculated as:

$$\frac{\partial \varphi}{\partial t} + \nabla \varphi \cdot v = 0 \tag{3.4}$$

if v_N is the velocity in the normal direction, then:

$$v_N = v \cdot \frac{\nabla \varphi}{|\nabla \varphi|} \tag{3.5}$$

the motion equation of φ is:

$$\frac{\partial \varphi}{\partial t} + v_N |\nabla \varphi| = 0 \tag{3.6}$$

specify conditions:

$$\varphi(x, t = 0) = 0 \tag{3.7}$$

This is the level-set equation introduced by S. Osher and J. A. Sethian [12]. When the normal velocity v_N is only a function related to \vec{x} , t and $\nabla \varphi$, the Equation (3.6) is a standard Hamilton-Jacobi equation. By solving the Hamilton-Jacobi equation, the value of $\varphi(x, t)$ at any moment can be calculated, and the points at $\varphi(x, t) = 0$ can be connected to generate a new toolpath.

To facilitate understanding, try to use visual methods to describe the above process, take the half vase model as an example. The plane mesh is generated by the Angle Based Flattening (ABF++) algorithm [11], and the discrete points $x_{i,j}$ are produced by discretizing the plane mesh area which can be expressed as:

$$x_{i,j} = p_{i,j}(x_i, y_j) \text{ for } p_{i,j} \in R^2 \tag{3.8}$$

where i is the number of discrete points in the x-axis direction, j is the number of discrete points in the y-axis direction, as shown in Figure 5(a). The value $\varphi(p_{i,j}, t = 0)$ calculated by Equation (3.2) is used as the z-axis value of each point, then these discrete points can implicitly construct a surface in 3D space (Figure 5(b)). The normal velocity v_N is the velocity at which the z-axis value of each point $p_{i,j}$ changes with time.

Enlarge the local boundary in Figure 5(a), as shown in Figure 6(a). When the interface $\Gamma(t)$ moved, the $p_{i,j}$ unchanged, the minimum distance $d(p_{i,j})$ to the interface changed. Therefore, the zero level-set in 3D space and the shape of the curve changed, the new curve can be calculated by solving the Hamilton-Jacobi Equation (3.6) [12]. To describe in the simplest and most intuitive way for ease of understanding: detect the adjacent discrete points, if the signs of the two points are different, a point in zero level-set is obtained by linear interpolation in the two points, and a new 2D toolpath can be generated by connecting all the points in zero level-set, as shown in Figure 6(b).

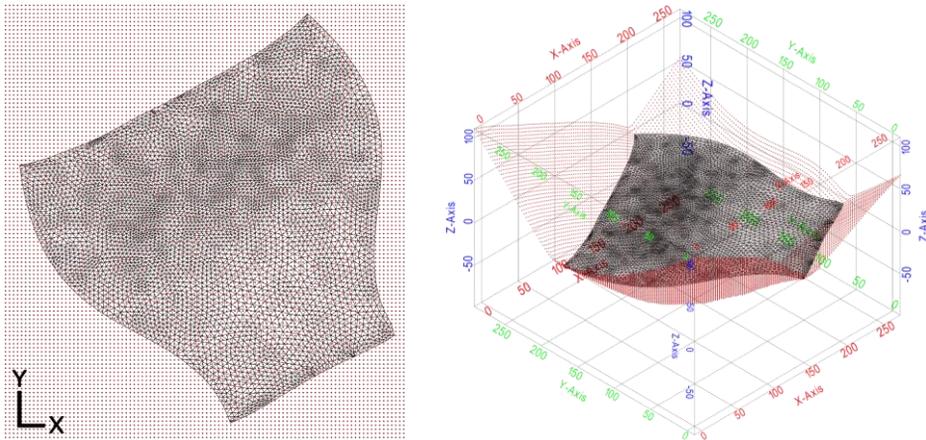


Figure 5: Discrete point set of plane mesh:(a) 2D discrete point set, and (b) representation of discrete point set in 3D space.

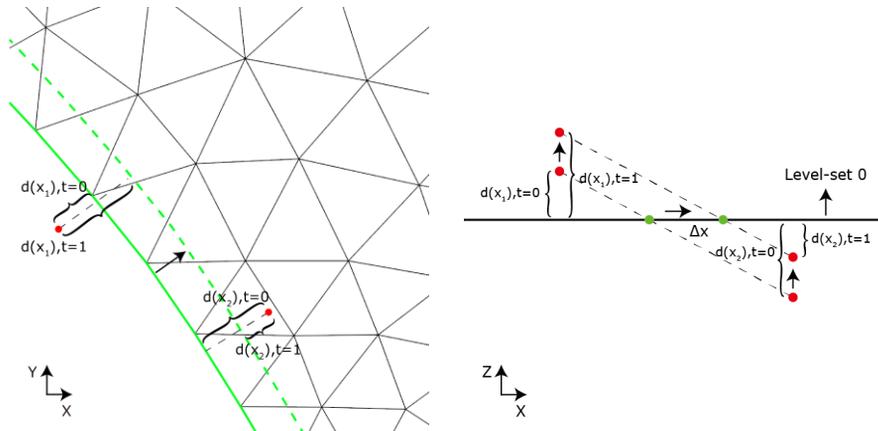


Figure 6: Interface motion:(a) interface motion in 2D domain, and (b) discrete point motion in 3D domain.

Visualize the above process, as shown in Figure 7. Take the discrete points state when $t = 1$, $t = 9$, $t = 21$ and $t = 34$ respectively, as shown in Figure 7(a). The discrete point set at different times implicitly represent a surface, as shown in Figure 7(b). The corresponding 2D toolpath can be generated by calculating the zero level-set of discrete point set at different times, as shown in Figure 7(c).

Therefore, the effective loop change in the offset process of 2D PS-curves is implicitly included in the change process of z-axis value of discrete points, and the intersection problem of PS-curves is perfectly avoided. The non-uniform offset can be realized by providing different normal velocity $v_{N(i,j)}$ for each discrete point $p_{i,j}$.

3.2 Calculation of Non-uniform Offset Parameter

The iso-scallop toolpath based on level-set can be constructed in a 2D plane by setting the normal velocity v_N of each point in the calculation domain to the line spacing l_{side} that satisfies the scallop

height constraint, which can be calculated directly using the equation $l_{side} = \sqrt{8hr}$, h is the preset maximum scallop height and r is the cutter radius.

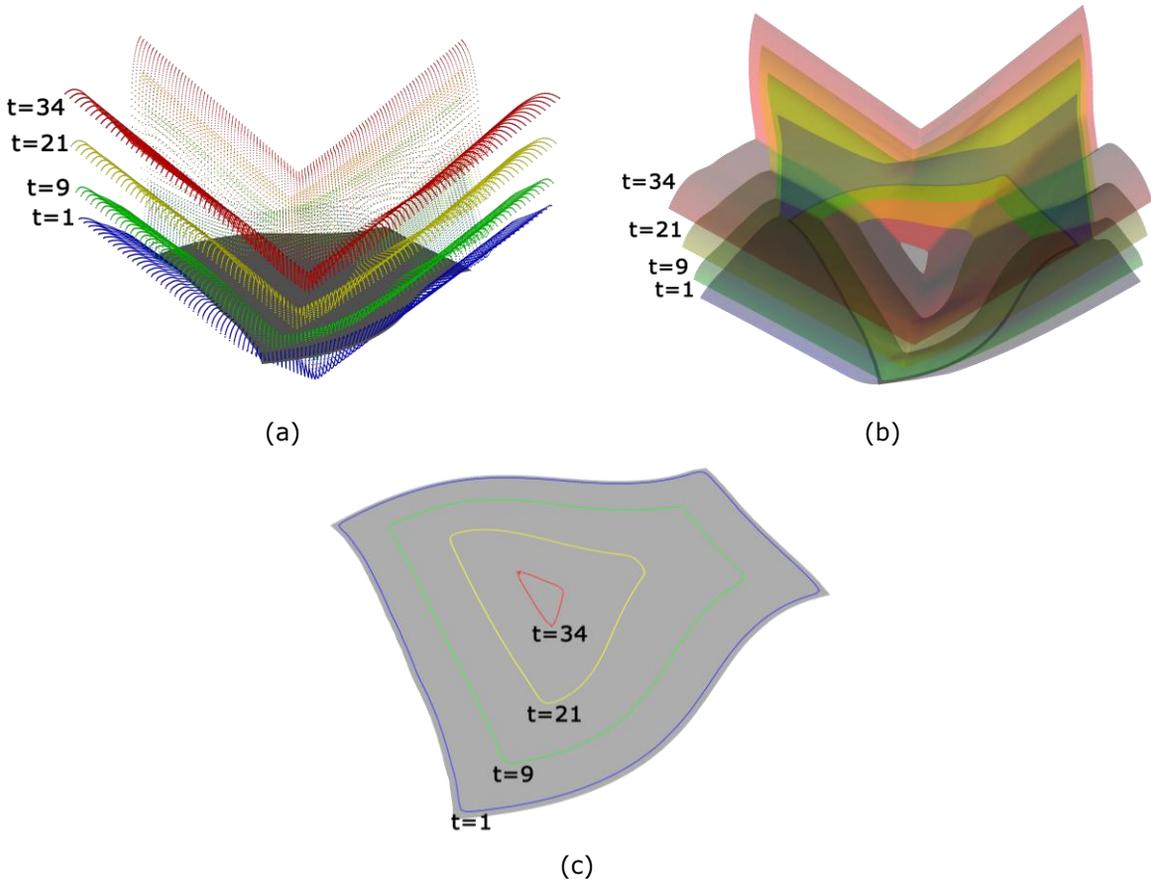


Figure 7: Visualization of non-uniform offset based on level-set:(a) 3D discrete point sets, (b) surfaces implicitly represented by discrete Point sets, and (c) zero-level set of discrete point sets.

Therefore, the normal velocity v_N in Equation (3.6) can be defined as:

$$v_N = l_{side} \tag{3.9}$$

In this paper, the conformal mapping method of mesh surface is used to map a 3D mesh surface into a 2D mesh plane. The 2D mesh will deform during this process, so the 2D line spacing l_{side} cannot be calculated directly. Therefore, the mapping stretch factor σ_θ needs to be introduced to calculate the 2D line spacing:

$$l_{side} = L_{side}/\sigma_\theta \tag{3.10}$$

This algorithm only applies to ball-end cutters, so only the radius of cutters and curvature of parts need to be considered, which is described in Reference [7]:

$$L_{side} = \sqrt{8hr\rho_{side}/(\rho_{side} \pm r)} \tag{3.11}$$

L_{side} is 3D line spacing, ρ_{side} is radius of curvature. It is worth noting that ρ_{side} must be greater than r , otherwise the algorithm cannot work. When the point $p_{i,j}$ is not in the closed interface $\Gamma(0)$, there is no mapping stretch deformation: $\sigma_\theta = 1$, and ρ_{side} is infinite, so $l_{side} = \sqrt{8hr}$.

To sum up, v_N can be expressed as:

$$\begin{aligned} v_N &= \sqrt{8hr\rho_{side}/(\rho_{side} \pm r)}/\sigma_\theta & \text{for } d(x) \leq 0 \\ v_N &= \sqrt{8hr} & \text{for } d(x) > 0 \end{aligned} \tag{3.12}$$

the mapping stretch factor σ_θ of anisotropy in triangle can be expressed from [2] as:

$$\sigma_\theta = \|w'\| = \sqrt{(\sigma_1 \cos \theta)^2 + (\sigma_2 \sin \theta)^2} \quad (3.13)$$

where:

$$\|w'\|^2 = w^T w' = w^T J_f^T J_f w = w^T I w \quad (3.14)$$

$$w = w_1 \cos \theta + w_2 \sin \theta \quad (3.15)$$

σ_1 and σ_2 are singular values of matrix J_f , θ is the angle between w and w_1 , w is the direction of 2D line spacing, w_1 and w_2 are unit orthogonal eigen vectors of matrix I . Where σ_1 , σ_2 , w_1 and w_2 can be calculated from reference [2]. The simplified algorithm is used to approximate the direction of w , when calculating $d(x)$ in Equation (3.2), taking the direction of the shortest distance from point x in the calculation domain to the closed interface $\Gamma(0)$ as the direction w , so the value of θ can be calculated quickly. In reference [2], the author considers the length deformation and direction deformation in the mapping process. This paper adopts the level set non-uniform offset method, if the direction deformation is considered, the amount of calculation will be greatly increased. In fact, the direction deformation generated by the conformal mapping method is small, which will not have a great impact on the final toolpath accuracy and the subsequent simulation results also prove this idea.

Now v_N in Equation (3.12) can be determined, and then the algorithm needs to be verified with models.

4 CASE STUDY

This paper uses the MATLAB level-set toolbox written by Ian M. Mitchell to calculate the level-set part and uses a half Bing Dwen Dwen model to verify the proposed toolpath generation method.

Firstly, the conformal mapping algorithm is used to map the surface mesh to a plane mesh, then the plane mesh is discretized into a series of discrete points. The initialization zero level-set can be calculated from Equation (3.2) as:

$$\mathbf{Data}_0 = \begin{bmatrix} \varphi(p_{(0,0)}, t = 0) & \cdots & \varphi(p_{(0,j)}, t = 0) \\ \vdots & \ddots & \vdots \\ \varphi(p_{(i,0)}, t = 0) & \cdots & \varphi(p_{(i,j)}, t = 0) \end{bmatrix} \quad (4.1)$$

Through Equation (3.12), let $h = 0.2mm$ and $r = 2mm$, the normal velocity matrix can be calculated as:

$$\mathbf{V}_0 = \begin{bmatrix} v_{N(0,0)} & \cdots & v_{N(0,j)} \\ \vdots & \ddots & \vdots \\ v_{N(i,0)} & \cdots & v_{N(i,j)} \end{bmatrix} \quad (4.2)$$

Take \mathbf{Data}_0 and \mathbf{V}_0 as input parameters and use level-set toolbox for iterative calculation to generate the non-uniform offset toolpath on the parameter plane as shown in Figure 8. Finally, the 2D toolpath is inversely mapped to the 3D mesh to generate the final toolpath, as shown in Figure 9.

To sum up, the steps of toolpath generation can be summarized as follows:

- Step1: use conformal mapping to map 3D mesh surface \mathbf{M}_{3D} to 2D mesh plane \mathbf{M}_{2D} . Then extract boundary $\Gamma(0)$ from \mathbf{M}_{2D} .
- Step2: generate the discrete points $p_{i,j}$ by discretizing the mesh plane \mathbf{M}_{2D} area.
- Step3: calculate the initial data matrix \mathbf{Data}_0 of level-set function from Equation (4.1). At the same time, calculate the normal velocity matrix \mathbf{V}_0 from Equation (4.2).
- Step4: input \mathbf{Data}_0 and \mathbf{V}_0 into the level-set toolbox to iteratively generate the toolpath \mathbf{Path}_{2D} in the plane of the parameter domain.
- Step5: using the inverse mapping algorithm, \mathbf{Path}_{2D} is inversely mapped to 3D space to generate the final toolpath \mathbf{Path}_{3D} .

Taking the Bing Dwen Dwen model as an example, the toolpath planning results are shown in Figure 8 and Figure 9. It can be seen from the figures that the non-uniform offset algorithm effectively uses different line spacing in areas with different mapping deformation.

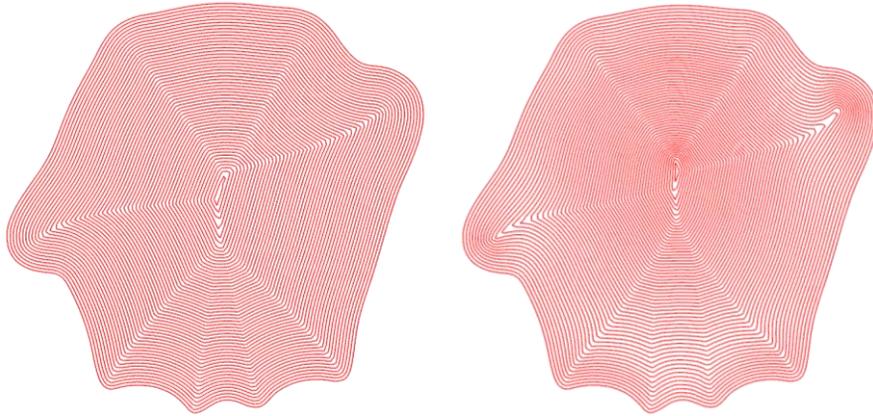


Figure 8: 2D toolpath of Bing Dwen Dwen model: (a) uniform offset, (b) non-uniform offset.

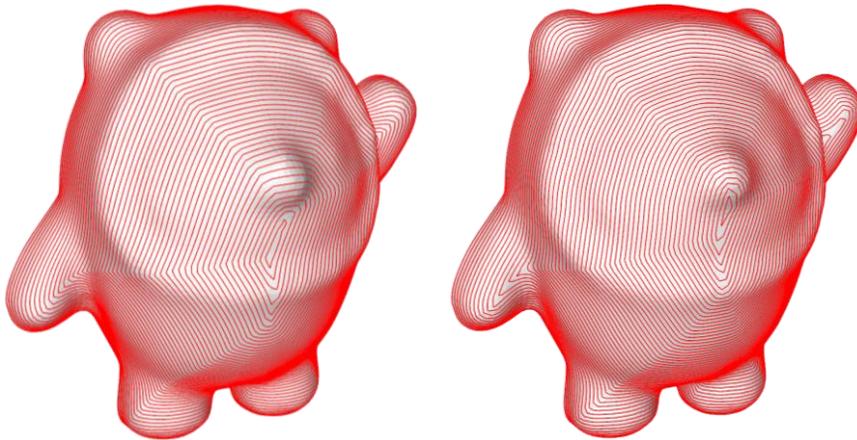


Figure 9: 3D toolpath of Bing Dwen Dwen model: (a) uniform offset, (b) non-uniform offset.

Then quickly evaluate the toolpath. The evaluation algorithm can be described as follows:

- Definition: m is the index of single toolpath (from inside to outside), n is the point index on the current single toolpath. M is the number of toolpaths, N is the number of points in each single toolpath.
- Start: $m = 0, n = 0$.
- Step1: calculate the normal curvature radius ρ_{side} , feed direction V_{feed} , normal direction V_{nor} and the opposite direction of line spacing $V_{rside} = V_{feed} \times V_{nor}$ at point p_{mn} .
- Step2: construct plane P with V_{rside} and V_{nor} .
- Step3: point p_{mn}' is calculated by intersection of plane P and adjacent toolpath. The distance from point p_{mn} to point p_{mn}' is line spacing L_{side} .
- Step4: calculate $h = \frac{\rho_{side} \pm r}{8 \cdot r \cdot \rho_{side}} \cdot L_{side}^2$ from Equation (3.11).
- Step5: if $n < N, n = n + 1$, jump to step 1. Otherwise, skip to step6.
- Step6: if $m < M, m = m + 1$, jump to step 1. Otherwise, end the calculation.

The Scallop height distribution is shown in Figure 10, and the specific evaluation results are shown in Table 1.

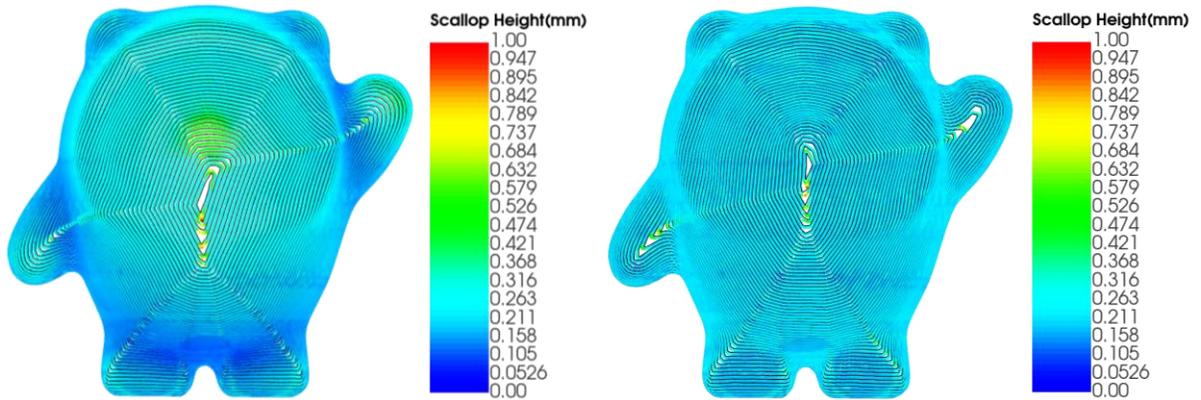


Figure 10: Scallop height distribution of half Bing Dwen Dwen model: (a) uniform offset, (b) non-uniform offset.

| <i>Code</i> | <i>h</i> | <i>r</i> | \bar{h} |
|-------------|----------|----------|-----------|
| a | 0.2 | 2 | 0.221099 |
| b | 0.2 | 2 | 0.202443 |

Table 1: Results of half vase model toolpaths evaluation.

Where \bar{h} is the average scallop height. It can be seen from the scallop height distribution diagram that the level-set non-uniform offset method proposed in this paper can generate iso-scallop height toolpath with uniform scallop height distribution, so it is effective.

5 CONCLUSIONS

In the previous research of our team, the iso-scallop toolpath planning is realized in the 2D parameter domain by using the mesh surface conformal parameterization technology, which greatly simplifies the relevant geometric calculation. However, it is necessary to continuously improve the 2D non-uniform offset algorithm to apply to different mesh surfaces, toolpath parameters and tool parameters. Based on the previous research, this paper proposes a non-uniform offset algorithm based on level-set, which can completely avoid the intersection problems in the process of 2D non-uniform offset and greatly improve the robustness of toolpath generation algorithm. More importantly, the algorithm is very simple, and the accuracy is controllable. Theoretically, by increasing the density of discrete points, the toolpath accuracy can be continuously improved, but this will increase the computational cost. How to select the appropriate discrete point density under the requirement of toolpath accuracy is one of the subsequent problems to be solved.

The algorithm in this paper only compensates the length deformation in the mapping. For some models with large amount of angular deformation after mapping, the generated toolpath maybe has large errors. This is one of the problems that need to be optimized.

From model test results, the toolpaths generated by this algorithm have uniform scallop height distribution, which is an important factor to evaluate the quality of toolpath. However, for the model with continuous deformation aggregation (model in which large amounts of deformation accumulate at a point after mapping), the algorithm still has some limitations. This is one of the problems that need to be optimized.

In theory, this method can be used for the multi-boundary model (with internal holes) and an iso-scallop toolpath can be generated. However, since the current toolpath evaluation algorithm cannot be applied to the multi-boundary model, this paper does not give this kind of model test

cases. The toolpath evaluation algorithm is also one of the problems that need to be optimized later. All the above issues are currently being solved by the author team.

ACKNOWLEDGMENTS

This work was supported by the Key-Area Research and Development Program of Guangdong Province, China [grant number 2021B0101190002].

Yue-Feng Li, <http://orcid.org/0000-0002-3730-1160>

Qing-Hui Wang, <http://orcid.org/0000-0003-3691-7374>

Wen-Feng Gan, <http://orcid.org/0000-0003-1698-465X>

REFERENCES

- [1] B.K. Choi; S.C. Park.: A pair-wise offset algorithm for 2D point-sequence curve, *Computer-Aided Design*, 31(12), 1999, 735-745. [https://doi.org/10.1016/S0010-4485\(99\)00060-3](https://doi.org/10.1016/S0010-4485(99)00060-3)
- [2] Chen-Yang Xu; Jing-Rong Li; Qing-Hui Wang; Guang-Hua Hu.: Contour parallel tool path planning based on conformal parameterisation utilising mapping stretch factors, *International Journal of Production Research*, 57(1), 2019, 1-15. <https://doi.org/10.1080/00207543.2018.1456699>
- [3] Han Z; D. C. H. Yang; Jui-Jen Chuang.: Isophote-based ruled surface approximation of free-form surfaces and its application in NC machining, *International Journal of Production Research*, 39(9), 2001, 1911-1930. <https://doi.org/10.1080/00207540110024663>
- [4] Hong X; Xiao G; Zhang Y et al.: A New Path Planning Strategy Based on Level Set Function for Layered Fabrication Processes, *International Journal of Advanced Manufacturing Technology*, 2021. <https://doi.org/10.1007/s00170-021-08239-0>
- [5] Lasemi, A.; D. Y. Xue; P. H. Gu.: Recent development in CNC machining of freeform surfaces: a state-of-the-art review, *Computer-Aided Design*, 42(7), 2010, 641-654. <https://doi.org/10.1016/j.cad.2010.04.002>
- [6] Lee; S.-G.; H.-C. Kim; M.-Y. Yang.: Mesh-based Tool Path Generation for Constant Scallop-height Machining, *International Journal of Advanced Manufacturing Technology* 37 (1-2), 2008, 15-22. <https://doi.org/10.1007/s00170-007-0943-x>
- [7] Lin; R.; Koren, Y.: Efficient Tool-Path Planning for Machining Free-Form Surfaces, *ASME. J. Eng. Ind*, 118(1), 1996, 20-28. <https://doi.org/10.1115/1.2803642>
- [8] Lin Z; Fu J; He Y et al.: A robust 2D point-sequence curve offset algorithm with multiple islands for contour-parallel tool path, *Computer-Aided Design*, 45(3), 2013, 657-670. <https://doi.org/10.1016/j.cad.2012.09.002>
- [9] Li; H.; H.-Y. Feng.: Efficient Five-axis Machining of Free-form Surfaces with Constant Scallop Height Tool Paths, *International Journal of Production Research* 42 (12), 2004, 2403-2417. <https://doi.org/10.1080/00207540310001652905>
- [10] Lu, J.; R. Cheatham; C. G. Jensen; Y. Chen; B. Bowman.: A Three-dimensional Configuration-space Method for 5-axis Tessellated Surface Machining, *International Journal of Computer Integrated Manufacturing*, 21 (5), 2008, 550-568. <https://doi.org/10.1080/09511920701263313>
- [11] Sheffer; A.; B. Lévy et al.: ABF++: Fast and Robust Angle Based Flattening, *ACM Transactions on Graphics*, 24 (2), 2005, 311-330. <https://doi.org/10.1145/1061347.1061354>
- [12] S. Osher; J. A. Sethian.: Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *Journal of Computational Physics*, 79 (1), 1988, 12-49. [https://doi.org/10.1016/0021-9991\(88\)90002-2](https://doi.org/10.1016/0021-9991(88)90002-2)
- [13] S. Osher; R. P. Fedkiw.: Level Set Methods: An Overview and Some Recent Results, *Journal of Computational Physics*, 169(2), 2001, 463-502. <https://doi.org/10.1006/jcph.2000.6636>
- [14] Xiong Y.; Park S.-I.; Padmanathan S. et al.: Process planning for adaptive contour parallel toolpath in additive manufacturing with variable bead width, *International Journal of Advanced*

- Manufacturing Technology, 105 (10), 2019 , 4159-4170. <https://doi.org/10.1007/s00170-019-03954-1>
- [15] Yuwen; S.; Dongming; G.; zhenyuan; J. et al: Iso-parametric tool path generation from triangular meshes for free-form surface machining, International Journal of Advanced Manufacturing Technology, 28(10), 2006, 721-726. <https://doi.org/10.1007/s00170-004-2437-4>