The Development of Shop Documentation Plug-In Based on NX Software

Jiaqi Zhao*, Xiaodong Yang and Ligeng Wu
Department of Precision Manufacturing Centre, Tianjin Institute of Aerospace Mechanical and Electrical Equipment, Tianjin, China
*royjuventus_29@126.com

Abstract. Shop documentation is a document filled with all relevant information of product processing, which is playing an important role in CNC machining. At present, although shop documentation could be generated from 3D modelling software, the contents in shop documentation are usually not up to operators’ requirement. On this condition, the data structure of shop documentation should be rebuilt, and a new approach of generating shop documentation should be created as soon as possible. In this paper, the data structure of shop documentation is redesigned to make sure that the document contains the most suitable information, and a new kind of shop documentation plug-in is developed in Siemens NX software at the same time, which could generate PDF-format shop documentation very fluently. The plug-in is programmed by C++, using the function library of NX software which is called NX Open C. Finally, an example of generating shop documentation in NX software is given to prove that the shop documentation plug-in works properly.

1. Introduction

In the last decades, with the great development of computer technology, CAD and CAM have made great contributions to CNC machines. As the complication of workpiece shape is increasingly growing, multi-axis machines are being used more widely [1]. It has become particularly vital of rationality of using equipment and equipment utilization. Reasonable scheduling scheme makes the use of production resource more efficient, reduces the production cost of enterprises, and enhances the competitiveness of enterprises [2]. To this point, Bin Li has provided an insight into reasonable scheduling scheme, and he commits to analysing process parameters, reducing process time and waste of tools [3].

With the advance of NC technology, the majority of modern CNC machines are characteristic of high speed, high precision, high reliability, intelligence and compounding [4]. Complex machine structure makes CNC programming require software assistance [5]. For the purpose of maximizing the performance of CNC machines, the separation of CNC programmers and operators is a major trend.

At present, companies establish digital and paperless workshops [6], including models, database-based information management and equipment management and so on [7]. The workshop management is applied for CNC machine management and tool management. It is also convenient for transmitting CNC program [8], and they use the CNC program list to send message of process to operators [9]. However, it is difficult to transmit some key information to operators and operating-complex software elevates operators’ learning burden and company training cost.

In order to convey information more effectively, shop documentation is extremely necessary. So
far, shop documentation has been suitable for the products of similar shapes and the specific CNC machines [10]. However, most of shop documentation is accomplished manually, which is easier to contain human errors. Besides, the editable documents are likely to be modified by mistakes during delivery process. Shop documentation could be generated within NX software [11], but the shop documentation could not list important information, including clamping method, WCS, step description and some useful process parameters.

Compared to traditional research results, the shop documentation plug-in in this paper overcomes the shortcomings mentioned above. Firstly, the information, including model, tool parameters, WCS and other parameters of the manufacturing process, could be fully extracted by the plug-in. Secondly, the plug-in could generate read-only shop documentation in PDF-format. Thirdly, the shop documentation is suitable for duplication, storage and delivery.

2. System planning of shop documentation plug-in
The shop documentation plug-in could automatically extract views and manufacturing information of the model. The views include front view, side view, top view and axonometric view, while the manufacture information includes operation names, tool parameters, cutting parameters and non-cutting parameters. The data structure and workflow of the plug-in are illustrated as follows.

2.1. Data structure
The data structure of the plug-in is as shown in figure 1.

![Figure 1. The data structure of shop documentation plug-in](image-url)
2.2. Workflow
The workflow of the plug-in is as shown in figure 2.

Open NX software → Click “Shop Documentation” button in the main menu of NX software → Input manufacturing information in the coming out UI dialog → Shop documentation in PDF format is generated → Do not click “OK” button until the information is checked to be correct

Figure 2. The workflow of shop documentation plug-in

3. The development of shop documentation plug-in
The plug-in is developed by C++ on the platform Microsoft Visual Studio version 2012, and it could be used in NX software version 10 or above. The key technologies of shop documentation plug-in development are as follows.

3.1. The design of UI dialog
The UI dialog is created under “Block UI Styler” module of NX software. In “Block UI Styler” module all the controls could be dragged straightforwardly to the UI dialog, and the UI dialog could be easily made as the designer’s wish. The UI dialog interface is as shown in figure 3.

Figure 3. The UI dialog interface of shop documentation plug-in

3.2. The creation of model views
In NX software, the ACS (absolute coordinate system) is a conceptual location and orientation in model space. It is invisible and cannot be moved. All of the model’s information is related to the ACS. The Machine Coordinate System (MCS) determines the orientation and origin of tool paths for all operations in the orient group. The MCS has an initial position which is matched to the ACS.

In actual processing, each face of the part may be machined. In order to express cutting area properly, the plug-in must set appropriate views to show the MCS and cutting faces. After this, the CNC programmer should set the top view which he wants to use, then the plug-in will calculate the remaining views. The calculation process is shown in figure 4.
Figure 4. The flow chart of obtaining the screenshots of model views

a) Get the view matrix. The view matrix is a unit matrix, which is like \[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\]. It is orientation of the current view relative to the ACS.

b) The plug-in analyses whether the axis of the view is parallel (or vertical) to ACS or not. If the view is eligible, the plug-in will keep going. Otherwise, it will remind the user to adjust views. The view matrix is a unit matrix. In 3D space, it is a ball with a radius of 1, and the spherical centre coordinates is \((0, 0, 0)\) (figure 5). When the axis of the view is parallel (or vertical) to ACS,
it intersects the unit ball at the quadrant point. The vector may be (1, 0, 0), (0, 1, 0), (0, 0, 1), (-1, 0, 0), (0, -1, 0) or (0, 0, -1). In order to determine the eligibility of the views conveniently, the function \( |x| + |y| + |z| = 1 \) (figure 5) is established. In other words, when the vector meets the condition, the plug-in will proceed.

Figure 5. Unit ball and function \( |x| + |y| + |z| = 1 \) graph

c) In the previous step, the plug-in obtains the view matrix
\[
\begin{bmatrix}
  n_{11} & n_{12} & n_{13} \\
  n_{21} & n_{22} & n_{23} \\
  n_{31} & n_{32} & n_{33}
\end{bmatrix},
\]
in which \( |n_{11}| + |n_{12}| + |n_{13}| = 1 \), \( |n_{21}| + |n_{22}| + |n_{23}| = 1 \), \( |n_{31}| + |n_{32}| + |n_{33}| = 1 \). Then, for the purpose of displaying parts clearly, the plug-in will change rendering style to wireframe. In NX software, using the function by the parameters \( n_1 \) and \( n_2 \), it will return the view of X-Y plane, according to the user-defined view. In this way, the plug-in obtains the front view, side view and top view of the part.

d) The cutting area including several faces, the plug-in could present the cutting area in the trimetric view, which makes the processing information more intuitive.

The transformation method is \( M_c \times M_p \). In this function, \( M_c \) is the view matrix. \( M_p \) is defined as
\[
\begin{bmatrix}
  M_r & M_t \\
  0 & 1
\end{bmatrix},
\]
\( M_r \) is a rotate matrix. \( M_r = R_xR_yR_z \).

The \( R_x \) is defined as:
\[
\begin{bmatrix}
  1 & 0 & 0 \\
  0 & \cos(\text{roll}) & \sin(\text{roll}) \\
  0 & -\sin(\text{roll}) & \cos(\text{roll})
\end{bmatrix}
\]

The \( R_y \) is defined as:
\[
\begin{bmatrix}
  0 & 1 & 0 \\
  \sin(\text{pitch}) & 0 & \cos(\text{pitch}) \\
  \cos(\text{pitch}) & 0 & \sin(\text{pitch})
\end{bmatrix}
\]

The \( R_z \) is defined as:
\[
\begin{bmatrix}
  \cos(\text{yaw}) & \sin(\text{yaw}) & 0 \\
  -\sin(\text{yaw}) & \cos(\text{yaw}) & 0 \\
  0 & 0 & 1
\end{bmatrix}
\]

(roll = 45°, pitch = 45°, yaw = 45°).
$M_t$ is a translation matrix. As trimetric view does not require translation, $M_t = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$. After that, the colour of cutting area in trimetric view changed into dark colour so that cutting area could be figured out.

e) Finally, the plug-in will direct view to the X-Y plane and change rendering style to shade.

3.3. The extraction of manufacturing parameters
In manufacturing process, a large amount of information is necessary to be passed to CNC machine operators, including the information of task, model, WCS, clamping, operation and tools. The plug-in requires proper CNC program, cutting area and essential basic data.

In the NX manufacturing application, most of operation data is set in the same way, including tools data, part stock, floor stock, spindle speed and feed rate. The functions and parameters needed by extracting data are as shown in table 1.

| MFG info          | Data sort | Function                        | Parameter of function          |
|------------------|-----------|---------------------------------|--------------------------------|
| Operation type   | String    | UF_OPER_ask_oper_type()         | Tag of operation               |
| Tool name        | String    | UF_OBJ_ask_name()               | Tag of tool                    |
| Tool No.         | Integer   | UF_PARAM_ask_int_value()        | UF_PARAM_TL_NUMBER             |
| Tool length      | Double    | UF_PARAM_ask_double_value()     | UF_PARAM_TL_HEIGHT             |
| Tool flute length| Double    | UF_PARAM_ask_double_value()     | UF_PARAM_TL_FLUTE_LN            |
| Part stock       | Double    | UF_PARAM_ask_double_value()     | UF_PARAM_STOCK_PART             |
| Floor stock      | Double    | UF_PARAM_ask_double_value()     | UF_PARAM_STOCK_FLOOR            |
| Spindle speed    | Double    | UF_PARAM_ask_double_value()     | UF_PARAM_SPINDLE_RPM            |
| Feed rate        | Double    | UF_PARAM_ask_subobj_ptr_value() | UF_PARAM_FEED_CUT               |
| Cutting time     | Double    | UF_PARAM_ask_double_value()     | 142                            |
| Cut level        | Double    | UF_PARAM_ask_double_value()     | Depending on operation type     |

The cut levels are double numbers, but the method of setting different. Planar/face milling use Depth per Cut. Fixed/variable contour milling uses the Stock Part Offset. Cavity/zlevel/zlevel_5axis uses Global Cut Depth. Hole milling uses Axis Pitch. Thus, when the plug-in extracts values of the above-mentioned operations, it will identify the types of the operations and use the matched parameters.

3.4. The generation of shop documentation
The shop documentation is generated from a template in the format of excel, and saved as a PDF format file. The program execution procedure of the generation of shop documentation is as shown in figure 6.

![Figure 6. The program execution procedure of the generation of shop documentation](attachment:image.png)
4. Experiment and verification
This paper will use a part model to test and verify the plug-in, which is as shown in figure 7.

4.1. The description of the experiment
In this workpiece, the MCS is set at center and all surfaces need milling-finishing. This processing work step as follows:

a) Use a D35R5 cutting tool to process four fillets and open rough.
b) Use a D17R0.8 cutting tool to process the rest of upper surface and semi-finished side.
c) Use a D6 cutting tool to finish the ground and side.
d) Use a R4 cutting tool to finish the slender groove on the top of the part.

Figure 7. The part model used for the experiment

4.2. The verification of shop documentation plug-in
When users run the plug-in, they need to fill in a dialog, which is as shown in figure 8.

Figure 8. Plug-in operation interface
After filling in the UI dialog, selecting the cutting area, and setting the model views, shop documentation will be generated. The shop documentation generated from the plug-in is as shown in figure 9, which could verify the shop documentation plug-in is working properly and friendly to use.

### Shop Documentation

| File No. | Program No. | Cutting Time |
|----------|-------------|--------------|
| 2019062333153 | dxzsc-2_30_5 | 654.07min |

| Task No. | Task Name | Body Size |
|----------|-----------|-----------|
| 20190623-01 | Mosquito Repelle | 330×340×64.782 |

| Part No. | Part Name | Model File Name |
|----------|-----------|----------------|
| 50 | Mill | dxzsc-2.prt |

| Process No. | Process Name | Step No. | Step Description |
|-------------|--------------|----------|------------------|
| 30 | Mill | 5 | Rough> Semi> Finish |

**Programmer**: Mr. Wu  
**Computer**: cxd_s  
**Date**: 23/6/2019 23:31:33

**Clamping**: Air pressure  
**WCS**: X:Mid, Y:Mid, Z: Top

| Operation Name | Tool Name | Tool No. | Tool/Flute Len. | Cut Levels | Part/Floor Stock | Time | Speed | Feed |
|----------------|-----------|----------|----------------|------------|-----------------|------|-------|------|
| ZLEVEL_PROFILE | D35R5     | 1        | 120/40         | 0.3        | 0.0             | 98.23min | 2500 | 1500 |
| CAVITY_MILL   | D35R5     | 1        | 120/40         | 1          | 0.30/15         | 112.05min | 2000 | 2300 |
| CAVITY_REST_MILL | D17R0.8 | 2        | 80/40          | 0.25       | 0.350/15        | 57.88min | 2000 | 2500 |
| ZLEVEL_PROFILE_1 | D17R0.8 | 2        | 80/40          | 0.35       | 0.150           | 86.46min | 2500 | 1500 |
| ZLEVEL_PROFILE_2 | D6        | 6        | 60/30          | 1          | 0.120           | 33.23min | 3500 | 1500 |
| ZLEVEL_PROFILE_3 | D6        | 6        | 60/30          | 1          | 0.05            | 33.20min | 3500 | 1500 |
| FACE_MILLING  | D6        | 6        | 60/30          | 0          | 0.0             | 21.45min | 2000 | 1000 |
| FACE_MILLING_2 | D6        | 6        | 60/30          | 0          | 0.0             | 2.08min | 2000 | 1000 |
| FACE_MILLING_3 | D6        | 6        | 60/30          | 0          | 0.0             | 1.07min | 2000 | 1000 |
| CONTOUR_AREA  | R4        | 4        | 75/40          | 4          | 0.0             | 8.3/min  | 2000 | 1000 |

**Figure 9.** The shop documentation generated by the plug-in
5. Conclusion

In this paper, a kind of shop documentation plug-in has been successfully developed, which could extract all the relevant information of the manufacturing product from NX software, and combine all the information to generate shop documentation in PDF format. This development of the shop documentation plug-in makes all manufacturing information shown in a file, which not only reduces times of the communication between programmers and machine operators, but also makes NC programming mistakes easily to be found out and corrected. Overall, the development of this kind of shop documentation plug-in makes manufacturing process more effective and more efficient.

Acknowledgments

We wish to acknowledge the encouragement from our department leaders, Zhaohui Zhu and Huadong He. Besides, we also appreciate for the support from Xiaobao Li Skill Master Studio of China.

References

[1] Yibo Liu and Fangrong Liu 2013 Classification and trend of NC programs Metal Processing 7 25
[2] Xin Zhou 2014 Design and implementation of workshop scheduling system on distributed numerical control system (China: University of Chinese Academy of Sciences) p 7
[3] Bin Li 2007 Basic on Lean Production of CNC Machining (China: The College of Automation Chongqing University) p 62
[4] Wei Liang and Xian Wang 2010 Development trend of CNC machine tools Journal of Guilin college of Aerospace Technology 58 164
[5] Zhaoliang Liu and Zanbo Yang 2018 Deeply study the application of NX software in NC machining World Nonferrous Metal 2018(9) 285-286
[6] Song Jin 2006 Development of Paperless Production System Based on Embedded Terminal In Tool workshop (China: the College of Automation Chongqing University) p 50
[7] Yu Zhang 2009 Design and Implementation of CNC Workshop Management System (China: Heilongjiang University of China) p 54
[8] Limao Mu 2003 The software development of long NC machining code transmission direct from PC machine to NC machine Modular Machine Tool & Automatic Machining Technology 2003(6) 53-55
[9] Guoli Gao and Jiaqiang Huang 2010 Automatic generation of machining program list and network post-process for die & mould Die & Mould Industry 2010(4) 16-17
[10] Haifei Wang, Bin Wang and Xuehua Li Research on CNC Machining Template and Program Automation (China: High Level Equipment for Civil Aircraft Manufacturing Technology and Equipment) p 8-9
[11] Siemens NX10 Documentations (Germany: SIEMENS)