Visualization of metalworking processes in the development of simulation programs for equipment with numerical control

A N Yurov¹, M I Chizhov¹, A V Bredikhin¹ and V V Lavlinskiy²

¹Voronezh State Technical University, 14, Moscow ave., Voronezh, 394026, Russian Federation
²Voronezh State University of Forestry and Technologies named after G.F. Morozov, 8 Timiryazeva str., Voronezh, 394000, Russian Federation

E-mail: yurovalex@mail.ru

Abstract. The article proposes the concept and implementation of soft-ware solutions that allow to present milling and turning processes. The software simulates metal-looping using a pre-created control program in G-code. Software products based on the use of functional capabilities of the open Cascade geometric core allow not only to visualize the processes of milling and machining, but also to transfer models with the results of processing through export formats to other CAD systems.

1. Introduction

Production processes for obtaining new products often depend on the capabilities and novelty of technological equipment. One of the ways to significantly increase the efficiency of manufacturing processes in the manufacturing and engineering industries of industrial enterprises is the use of CNC machines at enterprises (equipment with numerical control - CNC).

The enterprises actively use CNC equipment and machining centres. The use of numerical control has undeniable advantages in the production of finished products, and in the case of re-equipment of production. The main advantage of CNC machines is a higher level of automation of production. The process of human intervention in the technological process is reduced exclusively to the loading of the control program and subsequent control in case of emergency situations. These systems can operate almost autonomously, producing high quality products. At the same time, as noted earlier, the task of the service personnel will be preparatory and final operations, such as adjustment and inspection of tools, installation and removal of work pieces, which can be done by one person, and on several machines at once.

Another significant argument is the following fact: CNC machines have high accuracy and repeatability of machining parts. This means that the quality of parts in manual processing can be influenced by the human factor (the duration of the change of the operator, the complexity of manufacturing, etc.), the CNC are devoid of such defects, they will complete the processing of parts in a given time interval and the quality of processing will be at the same level. And another advantage can be attributed to the fact that CNC machines allow you to process complex parts, the manufacture of which is impossible on conventional equipment. An example of such parts is stamps and molds.
Since the processing time of the work piece using the numerical control is regulated, this factor allows you to more accurately determine the processing time of the batch of parts, which allows you to more accurately plan the production.

In addition to the advantages, there are also negative factors regarding CNC equipment—it is the cost. CNC machines are a significant investment in the production of products. In addition to the cost of purchase and installation, they are still quite expensive to maintain, so not all enterprises can afford them. But these disadvantages are easily overlaid by high performance and precision machining.

When using modern equipment working in symbiosis with CAD solutions at the model level, there is a need to create and prepare the control programs for numerical control systems. In addition, the current and the machining process, or rather an imitation of it. It is important to see the result of the control program, to look at the progress of operations, which clearly demonstrate all the stages of processing according to the created G-code from the stage of loading the workpiece into the machining center (or the technological remainder of the machine) to the completion of the processing process.

2. Requirements for the design decision

Statement to the creation of software solutions that implement processes for visualization of Metalworking processes can be formulated as follows: the created control program is used to process the work piece, which was imported into the system. The process must be implemented in real time. At the same time, for the milling module, it is necessary to solve the following tasks:

- load the model into the system;
- prepare dialog boxes with parameters for the specified machining contours, including the diameter of the cutter, the starting point with coordinates responsible for the positioning of the tool, the starting point of the contour, working feed, etc.;
- implement methods to create control programs for the required contours;
- to visualize the process of milling by means of the dynamic movement of the machining tool, plotting of track lines and subtracting material from a solid model of the work piece;
- prepare the model for recording in export format.

Based on the formulated tasks, it is possible to offer the concept of the software product and algorithms for the functional part of the application.

To implement a project that allows you to simulate and visualize the process of turning, it is necessary to solve the following problems:

- obtaining models (their development and loading into the application);
- description of turning cycles;
- development of tools for species transformations in real time;
- animation of the turning process;
- synchronization of movements of tool and technological parts of the equipment during processing;
- uploading the results to the media.

In addition, it is useful to add different services to the application, which will allow you to control the color of models and the background of the application, access to the menu when the processing is underway, and a number of other features.

3. Requirements for the design decision

For the design solution, you can offer a formal description for the previously assigned tasks.
In a windowed application, you must create an instrumental panel that contains a number of contour types. Each circuit needs to interact with the dialog box in which we determine the parameters for the preparation of the control program.

Making a choice from a predefined set of parameters, the transition to the next stage is carried out - the creation of the control program with its output in a separate information window.

The visualization process itself is available under the following factors: the solid-state model is loaded into the system, the control program is prepared and shown, all parameters are set and correct.

Visualization is performed with the display and movement in real time of the milling tool (which can be conventionally represented by a cylinder), showing the trajectory of the cutter and removal of material from the work piece.

During processing, all options are available: to change the color of the model and tool, zoom in and out of scene objects, specify the current processing position in the control program, and a number of other features.

To visualize the machining operations required to implement a screening processing of the work piece in real time. The original model (work piece) can be obtained by importing from an external file, and all other elements (models of cutters, auxiliary machinery) can be integrated into the application. It is easier to use models that can be prepared in a third-party CAD. Next, load models into the system as needed.

In addition to visualization services for loading and unloading data, painting models need dialog boxes. They must also be implemented.

View transformations and changes are useful both at the stage of preparation for processing processes and at the stage of visualization. You need tools that allow you to interrupt the simulation calculations and do the transformation. The development diagram of the application for visualization of turning operations can have the layout shown in figure 1.

**Figure 1.** Scheme of work of the module on visualization of turning processing.
4. The implementation of the application

On the basis of the previously proposed approaches, software modules responsible for turning and milling processes with real-time visualization were developed. Simulation of Metalworking processes allows you to observe the modification of the original part, which is imported into the software. Implementation of milling and turning, working with models, algorithmic designs were developed on the basis of classes and methods of the geometric kernel Open Cascade. Functionality and implementation of projects using Open Cascade are considered in [1-3]. The developed software module for visualization of milling processes has the following distinctive features:

the output screen of the scene with the model. All manipulations with the model, construction of track lines, construction of channels of contours of processing, display of a trajectory of movement of the tool and the tool are made in the main window;

elements of the dashboard, responsible for the species changes to the model within the scene. The motor can be rotated and moved both during operations and after the milling process. Available modes of representation of the model in the carKass version, as well as in solid state. For the convenience of monitoring the processing process, the optimal view of the work piece in the axonometric projection is proposed. There is also an option to increase the model to the screen size;

processing of a given contour type, preparation of the control program and the milling process itself are placed on a separate tool panel, which is placed horizontally at the bottom of the application. The following types of contour processing are available: circle, rectangle, triangle. The functionality of the contours can be extended by including additional methods in the project responsible for the implementation of processing;

location of the control program. The program consists of a set of G-code commands for CNC machines. In the process of processing and visualization of the process, a special cursor-marker shows the location of the part of the code that is currently involved.

Previously, for the milling process, you need to specify the type of machining contour by calling the dialog box with parameters.

A visualization of the milling process for the contour defined by the circle according to the specified control program is shown in figure 2.

![Figure 2. Plotting the path of the cutter along the contour of the circle.](image-url)
parameters in the module are adjusted, new function blocks are opened. At the last step, the option to perform a simulated turning process becomes available.

The General view of the graphical interface of the created software module is shown in figure 3. The figure shows that the module provides a brief information on a given turning cycle, shows the scheme of the tool, the main application window displays a solid model of machine tools and cutter, as well as a model of the work-piece, which will be modified over time according to the control program. The interface of the application contains interactive tips that make it easier to work with the software module.

The implementation of part of the algorithms, as noted earlier, was carried out by the methods of the geometric kernel Open Cascade, as well as classes and components Qt + SDK in C++.

The process of visualizing turning cycles has been defined in several methods void runG90 (), void runG94 () and void runG75(). Void cutMaterialsGXX (int x, int y, int z, int R, int r, int deep) was developed to remove the work piece material during a given turning process. In addition, the components responsible for the movement of the tooling of the machine were created:

- int timerObject;
- bool breakTimer;
- double anglePatron;
- void timerEvent (QTimerEvent *event).

![Figure 3. General interface of the turning visualization software module.](image)

The following formal approach was used for the G90 longitudinal roughing cycle, which is designed to cut long cylindrical sections of the part according to the diagram in figure 3:

- several passes of the cutting tool are performed, after which the external cycle moves the tool to a new position;
- during operation, the material is removed from the model by the cutMaterialsGXX method(part_lengthdeepX+25,0,0,radiusBig,radiusBig-j-step,deepX), which acts as a blank;
- at the end, the turning equipment is stopped.
The cutMaterialsGXX() method is universal and can be used for other turning cycles. Its essence consists in the construction of a hollow cylinder and the subsequent subtraction from the workpiece after the complete passage of the cutter. This implementation is not quite correct (it is desirable to do the removal of the material at shorter intervals of the tool), but the means of Open Cascade for this approach will not be productive. The following code snippet uses constructible primitives of Open Cascade objects for the above recommendations.

```cpp
#include <BRepPrimAPI_MakeCylinder.hxx>
#include <BRepAlgoAPI_Cut.hxx>

void docObject::cutMaterialsGXX(int x, int y, int z, int R, int r, int deep)
{
    TopoDS_Shape cyl = BRepPrimAPI_MakeCylinder (gp_Ax2(gp_Pnt(x, y, z), gp_Dir(1, 0, 0)), R, deep).Shape();
    TopoDS_Shape cyl2 = BRepPrimAPI_MakeCylinder (gp_Ax2(gp_Pnt(x, y, z), gp_Dir(1, 0, 0)), r, deep).Shape();
    TopoDS_Shape ShapeCut = BRepAlgoAPI_Cut(cyl, cyl2);
    ShapeCut = BRepAlgoAPI_Cut(myData[1]->Shape(), ShapeCut);
    Handle(AIS_Shape) temp = new AIS_Shape (ShapeCut);
    temp->SetColor (Quantity_Color (0.67, 0.67, 0.0, Quantity_TOC_RGB));
    myVision->getContext()->Display(temp, true);
    myVision->getContext()->Remove(myData[1],true);
    myData[1] = temp;
}
```

Similarly, other turning cycles based on the above approach have been developed for the visualization process.

5. Conclusion

The proposed ways and concepts related to the simulation of processing processes using existing control programs led to the creation of a program solution with visualization of the processes of milling solid-state models. The application demonstrates the processing according to the specified control program for the selected contour. The application was developed with an intuitive interface, sufficient speed and advanced functionality. Support for export CAD data formats is included, allowing you to use the original models from different sources, as well as transfer the results to other applications for further application. The application provides functionality not only to visualize the process itself, but also developed additional options for graphical output. For example, adding track lines to the tool as it moves around the perimeter of the selected path. In addition, the model is modified during processing due to the functionality of the geometric core Open Cascade, thereby making the process of simulation processing more realistic.

The software module for visualization of turning processes of solid-state models imitates Metalworking according to the given turning cycles. The application was prepared using an intuitive interface, sufficient speed and advanced functionality. There is support for the well-known CAD data format, which allows you to import and export models, as well as transfer the results to other applications for further application.

The application provides functionality not only for visualization of the process itself, but also provides additional options for graphical output. For example, removing material from the turning area. Thus, the model is modified during processing due to the functionality of the geometric core Open Cascade, thereby making the simulation process more realistic.

References

[1] Yurov A N 2019 Design of automated system of production planning Modeling of systems and processes 12 87-93

[2] Yurov A N and Eremin I A 2017 Software modeling and visualization of geometric bodies by
means of the OCCT core. **Informatics: problems, methodology, technology: int. scientific-methodical conf.** (Voronezh: Voronezh State University Press) 340-5

[3] Yurov A N 2017 Development of Autonomous solutions based on functional capabilities of OCCT. **New technologies in research, design, management, production: proceedings of the Int. scientific and tech. conf.** (Voronezh) 189-91

[4] Bedaka A K and Lin C Y 2018 CAD-based robot path planning and simulation using OPEN CASCADE. **Procedia computer science** 133 779-85

[5] Jianqing Y et al. 2010 NC Machine Interference and Collision Checking System Based on OpenCASCADE. **International Conference on Digital Manufacturing & Automation** 976-979

[6] Liang Q et al. Developing CAM software with open CASCADE. **Advanced Materials Research** 753-755 1266-9

[7] Zhang M D and Wang J L 2015 The simulation and reconstruction of the robotic grinding trajectory on complex surface based on OPEN CASCADE. **Modul Mach Tool Autom Manuf Tech** 5 6-9

[8] Yuan Y, Wang Y H, Jiang L, Lin C D and Wang Y 2013 Research on modeling technology based on Qt and OpenCASCADE. **Modern Electronics Technique** 10

[9] Zhou Q et al. 2014 Research and Implementation of CAD-INP Integration System Based on OpenCASCADE. **Multimedia and Ubiquitous Engineering** (Berlin: Heidelberg) 227-32

[10] **OpenCascade - an open source library for BRep solid modeling** Retrieved from https://www.opencascade.com/