# VMS-I: An Internet-based Virtual NC Milling System

Anping XU*, Yunxia QU, Yanping GAO, Hongye HOU and Guolin DUAN
School of Mechanical Engineering
Hebei University of Technology
Tianjin 300130
P. R. CHINA

*Abstract:* - This paper describes an Internet-based virtual NC milling system. The model of virtual NC milling machine and 3D milling scene were established by using 3DS MAX and Virtual Reality Modeling Language (VRML), respectively. The dexel-based blank modeling method suitable for VRML language was developed. The interaction between Java Applet and VRML scene was realized by applying the External Authoring Interface (EAI) of VRML. An Internet-based virtual NC milling system named VMS-I was developed based on Tomcat service platform, which makes users vividly operate the virtual NC machine tool through Web browser. On a certain machining condition, the data of interest to users such as milling force and milled surface topography can be acquired. According to this system, the reasonability of the process parameters can be evaluated.

*Key-Words:* - Virtual manufacturing, NC milling, virtual reality, Internet, VRML, Java, EAI

## 1 Introduction

Time reduction and improvement of quality in the field of industrial manufacturing and material forming require more and more the use of a virtual approach coupled with the classical trial-and-error method. Indeed, numerical tools can minimize the number of expensive trials performed for the optimization of a given process. In the early 1990s, Virtual Manufacturing (VM) was firstly proposed. [1] Since then, many companies and research groups have been developing this activity [2-4]. VM makes it possible to estimate manufacturing processes previously without using real facilities; therefore, VM is expected to be used for many applications in manufacturing.

In Ref. [5], Onosato and Iwata presented the concept of VM systems, discussed the relations with other concepts in manufacturing and the requirements for VM systems, explained the architecture of VM systems, and introduced some systems developed for VM systems. Ref. [6] dealt with a modeling and simulation architecture for virtual manufacturing systems for the practical development and described a method of modeling and simulation for distributed simulation of virtual manufacturing systems. It was point out that the distributed simulation is the most effective way to get the real-time machining in VM. El Mounayri et al. [7] established a geometrical model of a five-axis milling system. Abrari et al. [8] extended the system model and took the dynamic effect into consideration. Tlusty and Ismail [9] added the milling force into the system. W. S. Yun and J. H. Ko et al. [10-12] also developed a virtual machining system, which was able to physically simulate the peripheral end-milling process, such as milling force and surface error. Xu et al. [13] investigated the end milling process considering the cutter flexibility. A novel model of the dynamic end milling system was proposed, and the milling force and the milled surface topography were predicted.

Virtual Reality (VR) emerged as a popular technology for human-computer interface in order to master the shortcomings of CAD systems [14]. It can simulate a realistic environment and allow a designer to interact with it. The essence of VR is immersion and interactivity, which differentiates it from CAD systems. Immersion means to give the designer a “being there” feeling, as if a real object is being manipulated. This provides the designer with a tool to perform what he or she studies, and broadens the designer’s understanding of the product. Interactivity implies the ability of humans to interact with events in the virtual world [15, 16].

VR has been applied to industries in recent years. Fang et al. [17] developed a virtual machining lab for knowledge learning and skills training, in which students could operate the virtual machine with the aid of a stereo helmet and digital gloves. After setting machining parameters, the machining process can be automatically conducted under the control of the generated G codes. Finally, the machining parameters such as cutting force, cutting power, surface roughness and tool life, could be predicted. Ong et al. [18] described an Internet-based virtual CNC milling system. Suh et al. [19] developed an Internet-based virtual machine tools.

In the documents mentioned above, most of them only focused on the physical simulation issues such as establishment of simulation model and algorithm. In this paper, the model of virtual NC milling machine and 3D milling scene are established by using 3DS MAX and Virtual Reality Modeling Language (VRML), respectively. The dexel-based blank modeling method suitable for VRML language is developed. The interaction between Java Applet and VRML scene is realized by applying the External Authoring Interface (EAI) of VRML. An Internet-based virtual NC milling prototype system named VMS-I is developed based on Tomcat service platform.

## 2 Structure of the System

An ideal virtual NC milling system should start with the CAD model of workpiece. As shown in Figure 1, the system selects the blank shape and its dimensions according to the CAD model given. After the process parameters such as the number of tooth, tool diameter, and spindle speed etc., are initially determined, the NC codes will be generated accordingly. The geometrical and physical simulation can, therefore, be done according to the NC codes. In the process of geometrical simulation, the system will be able to check whether the interference occurs, whether the tool path is right, and whether the part size and precision are acceptable. On the other hand, in the process of physical simulation, the milling force, tool life, surface quality and stability can be predicted. If the terms above are satisfactory, then the process parameters selected will be returned and used for practical fabrication. Otherwise, the process parameters need to be repeatedly modified until the concerned items are satisfactory.

Considering the fact that the existing commercial CAD/CAM packages are capable of automatically generating NC file, the presented system named VMS-I is started with NC file as the input, and so far it can predict the milling force, milled surface topography, vibration of the cutter relative to the workpiece, and real-time workpiece changes.

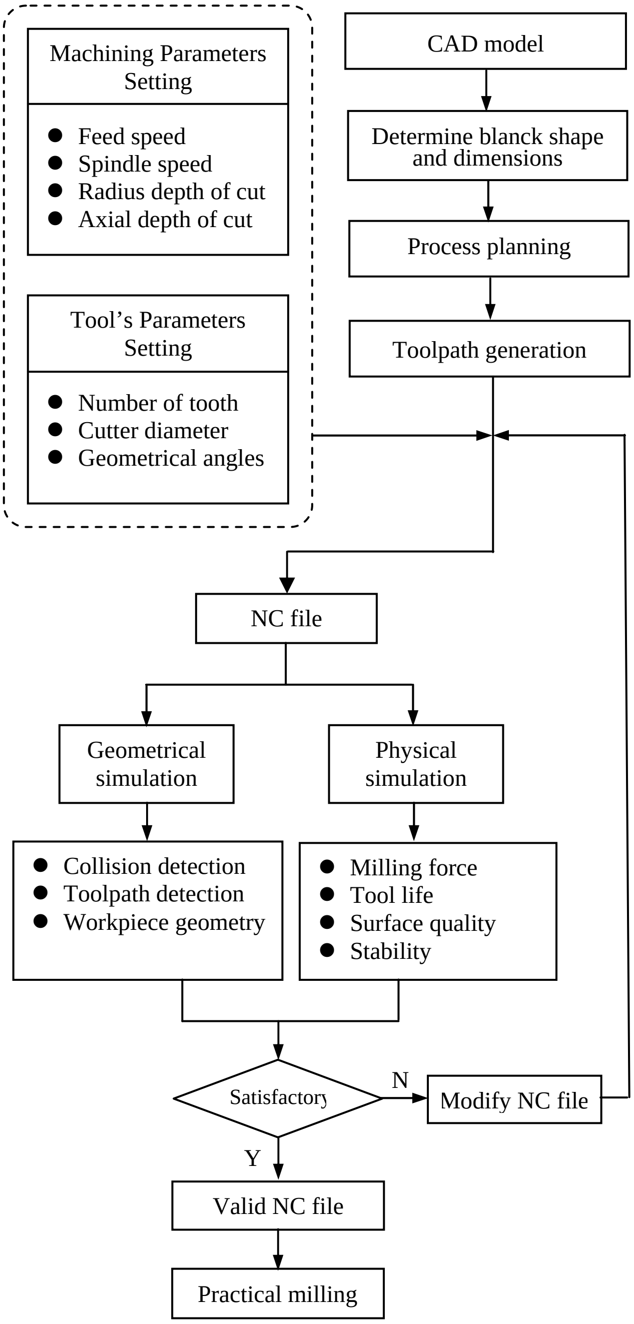


Fig. 1 Working flow of an ideal virtual NC milling system

## 3 Development Tools

Virtual Reality Modeling Language (VRML) is a file format developed for describing and interacting with 3D objects on the Internet. It has been published in two editions successively, i.e., VRML 1.0 and VRML 2.0, when it was brought forward at the W3 conference at Geneva, Switzerland in 1994. In 1997, VRML was formally issued as an international standard after a few editing and functional adjustments based on VRML 2.0 specifications. Then it obtained ISO authorization in 1998. It is usually called VRML 97. As VRML matured, different viewers have provided better and better interfaces. At the time of our research, VRML 2.0 viewer and Cortona VRML Client were selected. Henceforth, VRML in the remainder of the paper refers to VRML 2.0.

### 3.1 Interface Methods of VRML

VRML provides many nodes for 3D environment presentation, such as points, lines, facets and geometric objects, which endow VRML with powerful modeling ability. However, the animation and interactivity of VRML is limited by key-frame basis. The realization of complicated animations and interactivities should rely on integrating VRML with Java. Since Java Applet is popular in networking, Java is on the rise and is widely used for its platform-independent, powerful sustainable and web-based programming. The combination of Java and VRML is an ideal integration, which supplies our VMS with platform-independent, Internet-based capabilities and programmable control. At present there are two interface methods between VRML and Java: SAI (Script Authoring Interface) and EAI (External Authoring Interface). The former interface controls object movement in VRML environments by integrating Java with the Script and Route nodes of VRML; the latter controls VRML scene nodes through direct interactivities between Java Applet and users. Applet can interact with HTML, database and other media to actualize Internet control. Thus the function and scope of EAI are better than that of SAI. In the proposed VMS, EAI method was used.

### 3.2 Basic Nodes of VRML

VRML program is composed of HEAD, BODY, NODE and ROUTE lines. Each node defines a 3D shape and its properties through the bracketed parameters. Animation and interactivity can be conducted by the sensor node, interpolator node and ROUTE.

VRML scenes can be roughly divided into the following categories: geometrical shapes, vision attribute, scenic hierarchy and behavior attribute. They are described by their own nodes, respectively. For example, the geometrical shapes can be described by Box, Cone, Sphere, IndexedFaceSet, Text etc.; the vision attribute can be described by the nodes such as Color, Texture, and Material etc.; the scenic hierarchy can be described by the node Transform; the behavior attribute can be realized through the data stream mechanism. These nodes can achieve the simple tasks such as interpolation and sensor events. Most of the complex functions of the data stream are realized with the aid of the other powerful programming languages such as Java, JavaScript etc. in the Script node.

### 3.3 Data Stream Model of VRML

VRML uses data stream model to describe the attributes of the object in the given scene, and the data transmission between sensors, interpolators, scripts and paths in order to implement the communication with one another. Each VRML node has its own data-receiving interface (EventIn). When data are input, the internal computational functions are triggered, and then the generated values are endowed to EventOut domain, and EventOut transmits the data to the corresponding nodes. Accordingly, the desired effect can be obtained. Shown in Figure 2 is the data stream of VRML, which shows how the scenic object nodes work.

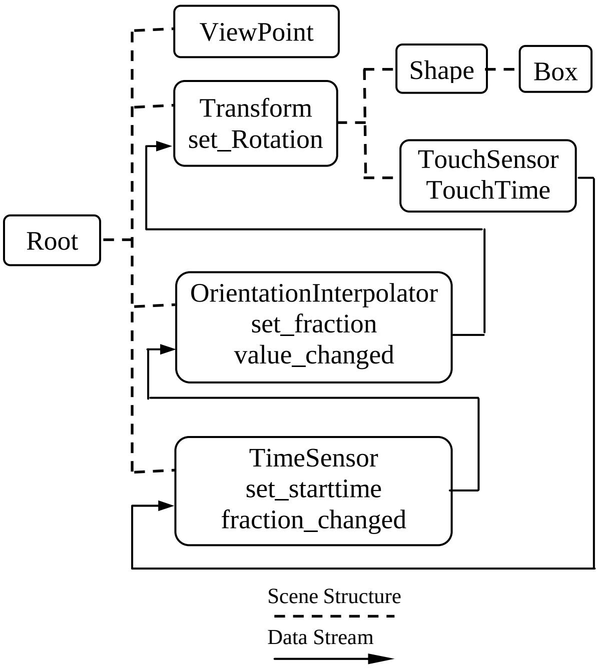


Fig. 2 Data stream of VRML

## 4 The Scene Modeling

The model built in virtual environment is memorized in VRML format (.wrl file). There are three approaches to building this kind of file: the first one is to use a text editor (i.e., VrmlPad) to directly write it, the second one is to use a visual authoring tool to create it, and the third one is the converting approach from other graphic file formats.

Comparatively, the first one is relatively direct, and its disadvantage is that it can only create simple objects and scenes. So the creation of complex scenes usually doesn't use this approach.

The prominent advantage of the second approach is the convenient use. It does not require a great amount of manually-input commands. Several commonly-used visual authoring tools are Cosmo World 2.0, Paragraph's Virtual Home Space Builder, Home Space, Caligari's Pioneer, Virtus WalkThrough Pro and 3D Website Builder etc.

The third method is commonly used to build a complex model. On the Web page, there are a variety of 3D model file formats, and the conversion tools are in charge of conversion from one format to another. These converters include Syndesis's InterChange, Macromedia's Extreme 3D and DoomToVrml2, Sony's VRML1to2 and SGI's VRML1toVRML2, which can convert VRML 1.0 into VRML 2.0. In addition, some 3D CAD modeler such as 3DS MAX and UG etc. can directly export their models in VRML format (.wrl). Using such a way can effectively avoid the error resulting from the conversion. It should be noted that all of those converters have their own limitations, e.g., the file after conversion is usually big; therefore, the lighting and texture mapping in the post-processes is liable to fail. So it is necessary to optimize the file after conversion.

In this system, the text editor of VrmlPad was used to create the simple parts of the milling machine. Since the workshop and environment modeling need to comprehensively consider not only the position of the machine but the memory space as well, and thus it is hard to model using the text editor, the workshop and environment were modeled using the visual authoring tool of Cosmo World. The whole body of the milling machine was obtained through the third approach. In this study, the 3DS MAX was selected because it is relatively simple to use and the directly-exported .wrl file has much smaller size than the one exported from UG.

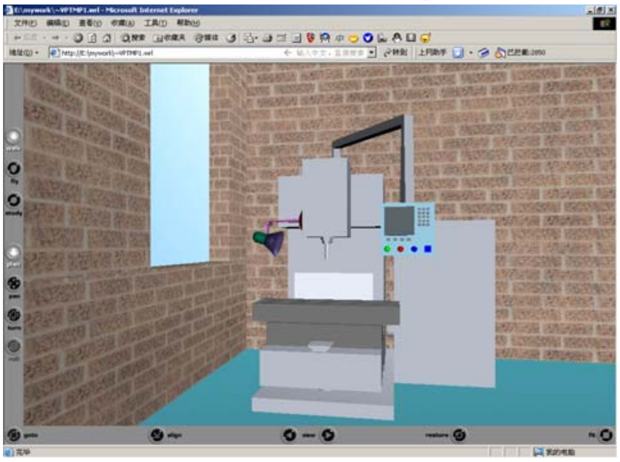


Fig. 3 VRML model of the milling machine

It is convenient to transmit through the Internet. Shown in Figure 3 is the presented milling machine model in the VRML format exported from 3DS MAX. The workshop and lamp models were built using the Cosmo World, and the control panel was obtained through writing the text programs in the VrmlPad.

# 5 Simulation Models in the VMS-I

## 5.1 Depth Element Model in VRML

Depth Element (dexel) was first proposed by Van Hook [20]. This representation is very suitable for set operation. Dexel model is actually the extension of Z-buffer technique [20, 21]. That is to say, each grid on the screen is associated with a chain table node that memorizes the depth information of the specified object in the certain view direction. As shown in Figure 4, the cuboid-like element was called a depth element (abbreviated dexel). The set of all these dexels can provide enough information for object display in the specified view direction.

After building the dexel part model, it needs to be imported into the VRML scene to display. Among the many nodes of the VRML, the ElevationGride node can be used for implementation of the dexel model [22]. However, each grid element of the ElevationGrid node is only associated with a dexel so that it is hard to describe the complicated shape as shown in Figure 4. In VRML, only IndexedFaceSet node can describe complex shapes. The syntax of the IndexedFaceSet node can be referred to Ref.[22].

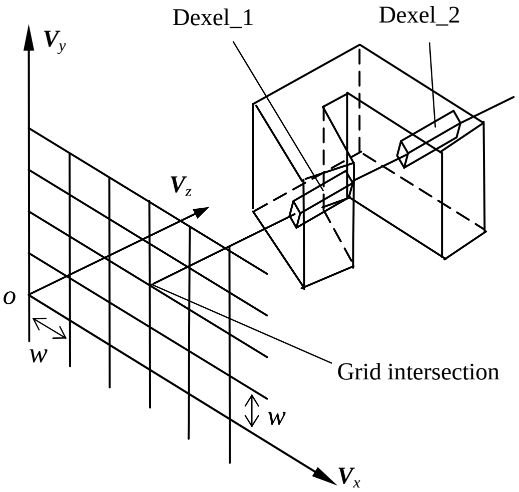


Fig. 4 Dexel model

The ElevationGrid node of VRML can be used for rapid subtract operation to the objects in the virtual scene. It is very important to guarantee the real-time ability of the dynamic machining simulation. However, the ElevationGrid node itself can only form a piece of surface other than an integrated body. In order to be able to show users the integrated solid object after simulation, the ElevationGrid model of the solid object needs to be converted into the IndexedFaceSet model [22].

### 5.2 Surface Generation Model

Machined surface is generated by the combination of several motions that not only facilitate the chip-forming processes but also move the point of engagement along the surface. The geometric shape of machined surface and its texture (called surface topography by a joint name) depend in the final analysis on the mutual positional relationships between cutting tool and workpiece in machining process. It is related not only to the concrete cutting method and machining conditions, including tool's path, but also to dynamic and material properties of machining system. So, without considering the effect of material properties of workpiece on machined surface, if it is requested to accurately describe machined surface topography, it is necessary to develop the tool's geometrical model and the kinematical and dynamic model of the machining process.

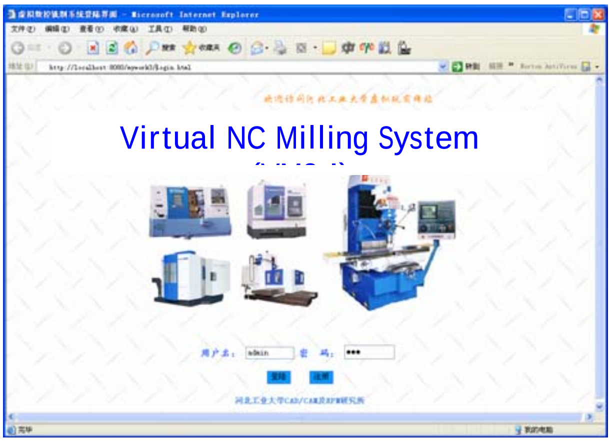
In this study, the milled surface generation model considering the cutter flexibility was employed. The details about the surface generation model and the corresponding algorithm can be referred to References [13] and [22].

### 5.3 Milling Force Model

The milling force modeling has always been a research focus since a few decades ago. Different kinds of milling force models have been reported [22]. In the proposed system, a generalized model for milling force simulation was employed [22]. Due to the page limitation, it cannot be stated here in detail.

## 6 Introduction to the System

Shown in Figure 5 is the login interface of the system. Users can browse the Web page with the Internet Explorer 5.0 or higher version. User ID and passport need to be input in the corresponding field before entering the system environment as shown in Figure 6. In Figure 6, the right part is in charge of controlling the system, through which the user can give the blank shape and size, as well as the position of the blank on the table. If the user enter a NC file name in the "NC File" field and click on the "Read in" button, the NC file will be displayed in the "NC codes" window. Of course, the NC file also needs to be validated by calling the G-code verifying program before dynamic machining simulation. If any error occurs, it will be highlighted in red and requests the user to correct it. The buttons on the control panel can be used to start or stop the operation even justify the cutter position. After the machining simulation, the milling force and milled surface topography will be displayed in their own windows.

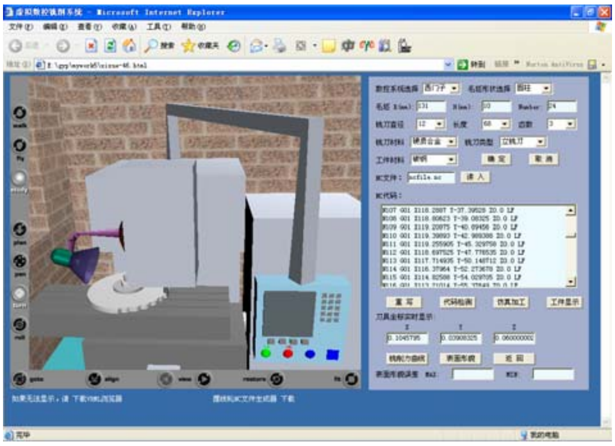


Microsoft Internet Explorer

Virtual NC Milling System

用户名： admin 密 码： •••

Fig. 5 Login interface of the system



Microsoft Internet Explorer

NC文件： ncfile.nc

Fig. 6 System interface in manufacturing

## 7 Conclusion

In this paper, an Internet-based virtual NC milling prototype system named VMS-I was developed based on Tomcat service platform, which makes users vividly operate the virtual NC machine tool through Web browser. On a certain machining condition, the data of interest to users such as milling force and milled surface topography can be acquired.

According to this system, the reasonability of the process parameters can be evaluated. The future work should be focused on the improvement of the system, adding some functional modules such as collision detection, tool path detection, and tool life, etc.

*Acknowledgements*

The authors gratefully acknowledge the financial support provided by the Provincial Natural Science Foundation of Hebei under Grant No: 500046.

*References:*

[1] Cheng Wu, *Introduction to Contemporary Integrated Manufacturing System*, Tsinghua University Press and Springer Press, 2002

[2] S. Ressler, Applying Virtual Environments to Manufacturing, *NISTIR*, 1995, p.5343

[3] http://www.isr.umd.edu/Labs/CIM/virtual.html, Feb. 2006.

[4] http://www.cenaero.be/v2/txt/20045185648.htm, Feb. 2006.

[5] M. Onosato, K. Iwata. Development of a Virtual Manufacturing System by Integrating Product Models and Factory Models. *Annals of the CIRP* , Vol.42, No.1, 1993, pp.475-478

[6] K. Iwata, M. Onosato, K. Teramoto, S. Osaki, Modeling and Simulation Architecture for Virtual Manufacturing Systems, *Annals of the CIRP*, Vol.44, No.1, 1995, pp.399-402

[7] H. El Mounayri, A. D. Spence, M. A. Elbestawi. Milling Process Simulation – A Generic Solid Modeler Based Paradigm, *ASME Journal of Manufacturing Science and Engineering*, Vol.120, No.2, 1998, pp.213-221

[8] F. Abrari, M.A. Elbestawi, A. D. Spence. On the Dynamics of Ball End Milling – Modeling of Cutting Forces and Stability Analysis, *International Journal of Machine Tools and Manufacturing*, Vol.38, No.3, 1998, pp.215-237

[9] J. Tlusty, F. Ismail. Basic Non-linearity in Machining Chatter. *Annals of the CIRP*, Vol.30, No.1, 1991, pp.299-304

[10] J. H. Ko, W.-S. Yun, D.-W. Cho, K. F. Ehmann, Development of a Virtual Machining System, Part 1: Approximation of the Size Effect for Cutting Force Prediction, *International Journal of Machine Tools & Manufacture*, 42, 2002, pp.1595-1605

[11] W.-S. Yun, J. H. Ko, D.-W. Cho, K. F. Ehmann, Development of a Virtual Machining System, Part 2: Prediction and Analysis of a Machined Surface Error, *International Journal of Machine Tools & Manufacture*, 42, 2002, pp.1607–1615

[12] W.-S. Yun, J. H. Ko, H. U. Lee, D.-W. Cho, K. F. Ehmann, Development of a Virtual Machining System, Part 3: Cutting Process Simulation in Transient Cuts, *International Journal of Machine Tools & Manufacture*, 42, 2002, pp.1617–1626

[13] A.-P. Xu, Y.-X. Qu, D.-W. Zhang, T. Huang, Simulation and Experimental Investigation of the End Milling Process Considering the Cutter Flexibility, *International Journal of Machine Tools and Manufacture*, Vol.43, No.3, 2003, pp.283-292

[14] A. Jezernik, G. Hren, A Solution to Integrate Computer-Aided Design and Virtual Reality Database in Design and Manufacturing Processes, *International Journal of Advanced Manufacturing Technology*, 22, 2003, pp.768-774

[15] S. H. Choi, S. Samavedanm, Modelling and Optimization of Rapid Prototyping, *Computers in Industry*, 47, 2002, pp.39-53

[16] S. H. Choi, A. M. M. Chan, A Layer-Based Virtual Prototyping System for Product Development, *Computers in Industry*, 51, 2003, pp.237-256

[17] X. D. Fang, S. Luo, N. J. Lee, Virtual Machining Lab for Knowledge Learning and Skills Training, *Computer Applications in Engineering Education*, Vol.6, No.2, 1998, pp.89-97

[18] S. K. Ong, L. Jiang, A. Y. C. Nee. An Internet-based Virtual CNC Milling System, *International Journal of Advanced Manufacturing Technology*, 20, 2002, pp.20-30

[19] S.-H. Suh, Y. Seo, S.-M. Lee, Modeling and Implementation of Internet-based Virtual Machine Tools, *International Journal of Advanced Manufacturing Technology*, 21, 2003, pp.516-522

[20] V. Hook, Real-Time Shaded NC Milling Display, *Computer Graphics*, Vol.20, No.4, 1986, pp.15-20

[21] T. Satio, NC Machining with the G-buffer Method, *Computer Graphics*, Vol.25, No.4, 1991, pp.206-217

[22] Anping Xu, Physical Simulation of NC Milling Process by Considering the Cutter Flexibility, [Ph.D. Dissertation]. Tianjin: Tianjin University, P. R. China, 1998