Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia CIRP 3 (2012) 103 – 108

www.elsevier.com/locate/procedia

45th CIRP Conference on Manufacturing Systems 2012

Realistic Machine Simulation with Virtual Reality

R. Neugebauer[a,b], P. Klimant[a,]*, M. Witt[a]

[a]Institute for Machine Tools and Production Processes, Chemnitz University of Technology,
Reichenhainer Strasse 70, 09126 Chemnitz, Germany
[b]Fraunhofer Institute for Machine Tools and Forming Technology Chemnitz,
Reichenhainer Strasse 88, 09126 Chemnitz, Germany
* Corresponding author. Tel.: +49-371-531-36911; fax: +49-371-531-836911 .*E-mail address*: philipp.klimant@mb.tu-chemnitz.de

Abstract

Today highly complex components are manufactured on NC-controlled machine tools. The NC programs, controlling these machines, are usually automatically generated by CAM software. This automatic processing is often erroneous. The VR-based realistic machine simulation, presented in this paper, extends the usual content of a machine simulation, like material removal and collision detection, by various new aspects. The coupling of a real NC unit allows the recognition and elimination of all process- as well as controller-caused errors. The integration of the multi-body simulation enables the consideration of inertia, machine rigidity and milling cutter deflection.

Keywords: Machine Simulation; Multi-Body Simulation; Virtual Reality

1. Introduction

The objective of the research project described in this article is to use VR technology, which was previously used mostly as a pure visualization medium, in a targeted manner as a development tool for mechatronic systems. In order to achieve this objective an Hardware-in-the-Loop (HiL) coupling between a Siemens SINUMERIK 840D machine tool control unit and a VR model of a 3-axis milling machine was used. This HiL coupling enables the utilization of the complete functional scope of the NC control unit and also allows for a variety of tests to be carried out on a VR machine model, prior to the production of the actual machine.

This article will first of all demonstrate how the coupling between a real NC control unit and a VR machine model can be established effectively. The Second main part in this article shows how the virtual machine model should be prepared and structured to realize material removal with a VR model. It will then be shown how process-related deviations in the work piece can be calculated and integrated into the VR simulation environment, firstly by means of the inertia of the moving masses and secondly by means of the tool deflection. The moving masses of the machine tools are observed using a multi-body system (MBS) model. The deflection of the tool is calculated using a bending beam model. As everything is to be integrated into a Hardware-in-the-Loop simulation environment, care must be taken during these simulations that the entire simulation fulfills the requirement regarding real-time capability.

2. State of the Art in NC Simulations

It is frequently the case that automatically generated NC programs have a great deal of optimization potential such as, for example, shorter travelling distances for the individual axes, reduced downtimes or optimized milling strategies. The NC programs can also contain errors that only occur upon the realization of the real NC control unit. For example, collisions may result if the tools previously defined in the NC program do not concur

doi:10.1016/j.procir.2012.07.019

with the real tools projected in the control unit. In addition, the tools may collide with a clamping device not considered in the NC programming, which could cause considerable damage to the machine. The correction of these errors and the optimization of the automatically generated NC program require considerable time expenditure and could also produce many defective tools. A simulation of the NC program using the VR machine tool model can therefore be used to identify and correct these errors prior to real production.

The majority of programs for checking automatically generated NC programs are desktop-based and are divided into two parts – machine and control unit. If the NC control unit is available as a program on the simulation computer, this is a so-called Software-in-the-Loop (SiL) coupling. The workspace of the machine tool is reproduced with these simulation programs. Thus, particular aspects such as speed, the range of movement of the tool and the accuracy of the geometry of the work piece to be produced can be checked. The software that reproduces the NC control unit reacts to errors. However such simulations only ever reproduce certain aspects of the overall system. Most of all, NC control units that have undergone considerable redevelopment in recent years and, as is standard practice in automation technology, work internally in real-time, are difficult to reproduce in their full scope of function (cf. [1]). For this reason, another type of simulation has been established in recent years. HiL couplings use the real NC control unit and reproduce the machine tool as a virtual model. In doing so, the largest challenge is the coupling of the NC control unit with the virtual machine model, as NC control units work in real-time and the response of the simulation model must take place after a specific time interval. (cf. [2])

The depth of the simulation model plays a crucial role in these HiL simulations. If the distances travelled by the machine are optimized and checked for collisions, the connection of the control unit is relatively simple. However, if the individual drives of the machine, particularly with regard to their masses and inertias, are to be reproduced, this type of modeling can be very expensive as most drive models use complex, abstract simulation models created, for example, using Matlab® Simulink®. With the help of these models, the individual parameters of the drives can be thoroughly checked and reproduced. However, due to their high degree of abstraction, checking for collisions of the machine axes can be rather difficult. Because the requirements placed on this coupling of the control unit are so enormous, the NC control unit is generally only operated in the so-called simulation mode. This means that its outgoing target values are returned directly to the internal position regulator as actual values. With this coupling variant the NC control unit runs the NC program as usual and the connected VR machine model runs precisely according to specification. This allows an early inspection of the NC program and the identification of potential collisions of the machine axes. The only disadvantage of this variant is that the internal controller structures of the control unit cannot be tested.

3. VR Supported HiL-Simulation Environment for Milling Machines

3.1. HiL Coupling

An HiL coupling between a virtual model and a real control unit can be established using various methods. The coupling variants are distinguished based on their performance capacity and requirements. For the majority of NC control units there are two possible ways of connecting the model to the control unit. One variant is the use of a machine manufacturer-specific automation bus. Another is to read the axis specifications directly from the control unit and to send this information via Ethernet (TCP/IP). With the second variant a precise knowledge regarding the setup and functionality of the control unit is required. The advantage of this variant over the variant using the automation bus is that no change to the hardware configuration of the NC control unit is required.

It is possible to read the current axis values via the local network as the NC control units sends all updated actual position values to the Human Machine Interface (HMI) at the same time. In the case of the Siemens SINUMERIK 840D this internal data transfer takes place via the Windows service Dynamic Data Exchange (DDE).

Feedback to the NC control unit is also possible via this interface. Collision monitoring of the real machine could, for example, be realized through this feedback. To do so, the simulation would have to be implemented on the machine tool parallel to real operation. In the event of an increased risk of collision in the virtual machine model, the NC control unit would be stopped prior to the collision actually occurring. Because the simulation uses the actual values from the NC control unit, the machines required reaction time must be observed to ensure that it stops in time. The required time gain could, for example, be achieved by increasing the body casing models of the moving parts or by pre-calculating the possible axis movement.

For the VR simulation presented here the axis values and the number of the currently selected tool must first be read from the HMI and sent to the simulation environment. These axis values must then be transmitted to the prepared VR machine model.

3.2. Preparation of the VR Machine Model

The starting point for the VR model of the machine tool is generally its CAD data. Various data formats are supported depending on the type of VR software. Virtual Reality Modeling Language, abbreviated to VRML, was developed as a neutral exchange format for 3D data. Most design and modeling programs support exporting in VRML (cf. [3], P. 247). Thus, for example, all current CAD programs can also directly export VRML. In addition, the majority of VR programs support the direct loading of VRML. It is therefore also recommended that the VR model of the machine tool be developed using VRML. The VRML 3D data structure consists of a scene graph, which is successfully used to reproduce the structure of a serial produced machine tool (cf. [4], P. 84-86).

A VRML model must then first be exported based on the CAD data of the real machine tool. Following the export this is completely static, i.e. without kinematic and animation information. The complete CAD dataset of the machine, and thus also the VRML dataset, is very extensive and contains an enormous quantity of polygons. The VR system can, however, only visualize a limited number of polygons. Therefore, as an initial step, the parts that are not relevant for the simulation, such as the housing or non-visible components, must be removed. This step is very costly, but worthwhile as the number of polygons can be vastly reduced (cf. [5]). If this is still not sufficient, individual components can be further reduced using automatic polygon reduction.

Both the static and moving machine parts should then be divided up into individual groups. Static assembly groups are, for example, the machine bed and the housing. Moving assembly groups are the rotary and linear axes, such as the X-axis for example. This hierarchical structure allows later movement specifications to be transmitted from the NC control unit to these assembly groups. For an optimal simulation, the hierarchy of the virtual machine tool must be structured analogous to the real machine tool. In the VRML this can be successfully realized using so-called transform nodes. Each transform node receives a separate name. Transformation specifications can then be transmitted to the real NC control unit via these nodes. Figure 1 shows the hierarchy of the VRML file, realized using the transform nodes, of the Deckel-Maho DMP 45 V linear 3-axes milling machine as an example.

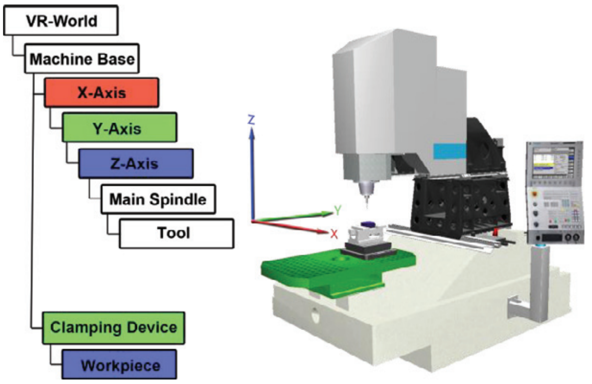

Fig. 1. Hierarchy of the model of the DMP 45 V linear 3-axes milling machine.

3.3. Material Removal Using VRML ElevationGrid

CAD models are generally volume-based. This makes them well suited for a material removal simulation using the sweep volume algorithm (cf. [5], [6]). The disadvantage of the use of a CAD-based sweep volume algorithm is that the calculation time of the sweep volume cannot be precisely predicted. This is largely due to the fact that the calculation of the intermediate steps during the process simulation of the work piece and the complexity of the work piece model can vary considerably and the render time of the graphics cards required for real-time visualization vastly increases as a result. This render time is largely determined by the number of polygons in the model. If the work piece changes, the number of polygons of the overall model also changes and so too the render time. This effect limits the use of CAD-core-based sweep volume algorithms to simple processing procedures and work pieces. In the case of more complex work pieces other material removal processes must be used, in particular for real-time capable VR visualization.

In Chapter 3.2 it was explained that the model of the entire machine tool was created directly in the standard VR format, VRML. VRML is a surface-based data format and is thus not suitable for a material removal simulation using sweep volumes, because the sweep volume generated by the tool movement in surface-based models would always be 0. However, the ElevationGrid node offers the possibility of creating a real-time capable VR-based material removal simulation directly within the VRML. The ElevationGrid is a defined elevation grid, whereby each point on the grid can be allocated a separate height value. The definition of the ElevationGrid is shown in Figure 2.

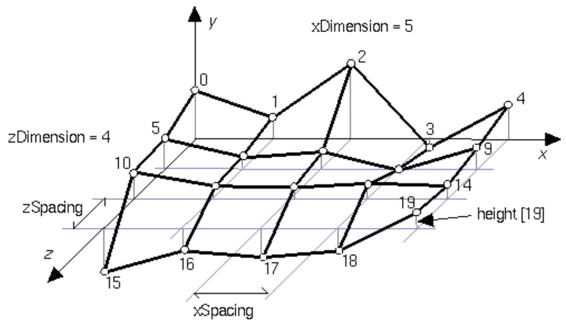

Fig. 2. Definition of the ElevationGrid [7].

The position of any point in the array of the ElevationGrid can be described in the coordinate system as follows (cf. [7] and [8])

$$P[i,j].x = xSpacing \cdot i \tag{1}$$

$$P[i,j].y = height[i + j \cdot xDimension] \tag{2}$$

$$P[i,j].z = zSpacing \cdot j \tag{3}$$

For 0 < i < xDimension and 0 < j < zDimension

The ElevationGrid must always be defined on the X-Z level in the VRML. If the work piece in the VR is on another level, the entire VR model must be adjusted accordingly. The xDimension and zDimension values specify the granularity and thus the intended resolution of the work piece.

The ElevationGrid is initialized directly upon loading the VR model. During the simulation only the individual elevation values change through the processing. The number of initialized points does not change. If the simulation can render the entire VR scene real-time capable with the generated ElevationGrid at the beginning of the simulation, then it can also do so during the simulation. This is an enormous advantage over sweep volume-based material removal algorithms. However, these material removal algorithms are very well suited for an offline simulation of the work piece, because they can achieve a considerably higher level of detail. The ElevationGrid-based material removal simulation is better suited for a real-time capable VR visualization which, due to the projection, has a limited resolution anyway.

4. Simulation of the Tool Deflection

4.1. Behavior of Milling Machines Under Load

During milling, depending on the process parameters and cutting method, varying forces can occur in the cutting zone. These forces have an effect on both the work piece and the tool, including their kinematic chains. Depending on the stiffness of the various components these forces lead to elastic deformations. Moreover, displacements between the components can occur due to backlash. These deformations and displacements result in changes in the cutting zone. One consequence of the process forces is the deflection of the tool, which therefore produces measuring inaccuracies in the component to be produced (cf. [9]).

Although machine tools are generally already backlash-free and relatively bend-proof designed, machine stiffness should not be neglected. The stiffness of the work piece is dependent on its shape and material and can therefore vary considerably (cf. [9]). Due to the unfavorable, rod-shaped form of the mill this is generally the most flexible link in the kinematic chain. The tool is therefore deflected due to bending particularly when milling with end mills.

4.2. Analytic Estimation of the Bending of the Mill Using Technical Bending Gauges

Albeit a considerable simplification, the mill can be viewed as a rod that is clamped on one side, whereby the shaft is generally cylindrical and the blade is cut or milled into a cylinder. The tool mount represents a fixed bearing and the projecting length of the mill represents the length of the model of the bending beam. As the first gross simplification, a resulting point force can be assumed at the end of the mill. If the bending of a rod clamped on one side is observed in this case, according to the bending beam model (cf. [10]), this will result in the warping of the rod at position x by:

$$u(x) = \frac{FI^3}{6EI_y}\left[2 - 3\frac{x}{I} + \left(\frac{x}{I}\right)^3\right] \tag{4}$$

The E-Module (E) and the moment of inertia of area about the y-axis (I_y) are hereby dependent on the mill.

4.3. Implementation of the Bending Beam Model in Matlab Simulink

In order to integrate the analytical approach of mill deflection into a model of the entire machine tool, the first step is to implement Equation (4) in Matlab® Simulink® as a block diagram. In this way the deflection of the milling cutter can now be calculated at each point in full knowledge of the forces.

This transition to Matlab® Simulink® is required as preparation for the simulation in the multi-body system (MBS) of the machine tool using SimMechanics™. The multi-body simulation serves to reproduce the machine tool with rigid bodies. Bearings and joints can be defined between bodies. Bodies can move by means of sensors and actuators, and their position and orientation can be specified. Values recorded by sensors can be analytically processed via Simulink block diagrams and returned to the bodies via actuators. The deformation of bodies should not occur in an MBS. In order to reproduce the deflection of the mill, the simplest method is to model the mill as a rod onto which the bend can be applied as a rotation about the clamping point. The angle of rotation is produced as a result of the displacement at the milling end. The mill is divided into segments in order to increase the level of simulation accuracy. The segment transitions are each positioned on the bend line of the analytical solution. Rotation joints are modeled between the individual segments, producing a kinematic chain of the individual segments. By way of an example, Figure 3 shows an end mill divided into segments.

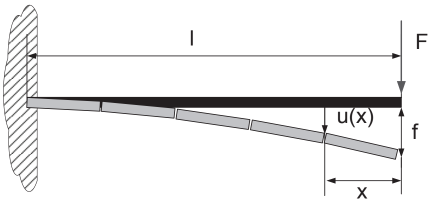

Fig. 3. Segmentation of the mill for the MBS model.

The segmentation can be increased in accordance with the level of simulation. However, the cutting calculation of the material removal simulation is subject to certain restrictions, which should have already been taken into consideration when designing the model in the MBS. Relevant restrictions are the use of simple geometric bodies, clear cutting conditions and gapless cutting. Failure to comply with these restrictions may lead to a drop in the performance level of cutting calculations and ultimately faulty results.

The individual segments are therefore modeled as simple cylinders for the cutting calculations, which can only be displaced and rotated in accordance with the simulation. Taking into account previous mentioned restriction for material removal, the mill is only subdivided into two segments (blade and shaft) in the simulation model.

5. Mechanical Replacement Model of the Machine Tool for the MBS Simulation

The forces described in Chapter 4.1 not only lead to the bending of the mill, but also to a softening in the kinematic chain of machine tools. This flexibility can be calculated using an MBS simulation model of the machine tool. The use of SimMechanics^{TM} allows rigid bodies to be combined with bearings, thus forming a kinematic chain of machine tool. The level of freedom of the bearings is thereby defined in accordance with the data for the real machine tools. Movements are defined for these bearings using actuators. The size and form of the elements are defined by their mass and their inertia tensor. The original position is defined by the position of the element's center point in space. The bearings between the elements allow for translational movements or rotations of the bodies, whereby, similar to the hierarchy in Figure 1, the bodies following the bearings follow these movements. The simulation model can be moved in this way by the directed bearings and actuators. The movements detected in the coupling serve as an input to the individual actuators of the respective bearings, meaning that the simulation model follows the axis values.

Up to now, this moving simulation model of the machine tool has only taken into consideration masses and movements in accordance with the real axis values of the NC control unit. A flexible model requires that the stiffness of the axes and drives are considered. In order to make this possible, simulation modules are integrated between the bodies and the bearings. An additional bearing will be added into each of these modules, which will allow for additional degrees of freedom according to the flexibility levels. The forces acting on the bearings can be determined using sensors and converted into movements through the use of actuators. By considering the specific stiffness between sensor and actuator a movement of the machine is generated in accordance with its levels of flexibility.

The movement of the rigid bodies, which are a result of the forces and rigidity, allow the total flexibility of the machine to be estimated. Via the kinematic chain, the tool mount in the shaft is subject to the translational movement that results from the forces and the flexibility of the machine. In the simulation model, the point at which the process forces act lies at the tool center point.

In order to integrate the tool deflection described in Chapter 4.3 both mill segments are connected to the kinematic chain of machine tools with a fixed bearing. The mill thereby follows the movement specifications of the axis values calculated in the coupling and is displaced in accordance with the machine rigidity. Because the process forces are known, a milling process, including static stiffness and tool deflection, can be simulated.

In the simulation model the position and orientation of each body can be read using sensors. The individual, new axis values are thereby transmitted to the VR model of the machine tool for the coupling to the VR software. These values are then used in the VR software to control the virtual machine model. In this way the flexibility and tool deflection, in the case of the 3-axes milling machine, act as pure translational movement on the mill. The limitation to pure translational movement of the mill in the VR machine simulation is required for material removal simulation. The previously used removal simulation for the 3-axes milling machine did not consider the tilting of the mill so far, but rather anticipated the translational movement of the axes.

6. Simulation of a Real Cutting Trial

In the simulation, the machine position as well as the deflections and flexibility levels were calculated using the recorded axis values, on the basis of the measured process forces. The positions of the tool center points of the tool calculated in this way will be saved in a new axis file. This calculated axis file will be used in the VR machine simulation to show the axial movements of the machine. The integrated material removal simulation calculates the geometry, which is affected by the flexibility and tool deflection, based on these movements. Real tool deflections and flexibility lie in the range of just a few µm, so particularly high requirements with regard to accuracy and the level of detail must be placed on the simulation. The material removal simulation described in Chapter 3.3 is designed primarily for real-time capability in the HiL coupling. However, for a higher level of accuracy it is recommended that a material removal simulation based on a CAD core will be used. This material removal simulation using sweep volumes (cf. [5]) is to be calculated at the position of precise cutting geometries, however it requires a greater computational effort and is no longer real-time capable. The removal simulation based on the CAD core already provides a calculated volume model of the work piece, which, using CAD applications, can be compared with a CAD model of the target geometry without a costly conversion process.

7. Summary

This article demonstrated that, if the setup of an NC control unit is known, an HiL coupling that is capable of, for example, reading the axis values of the control unit and sending commands to the control unit (e.g. Stop) can be established. The calculated axis values can be used in simulations to simulate the production process on a virtual machine. The reproduction of the complete machine tool, including tools, axial movements and the workspace allows collisions to be identified and NC programs or manual entries to be checked for errors. The virtual machine movements are the basis for carrying out cutting calculations on a virtual work piece, and thus for simulating material removal. Depending on the application, the machine simulation, including material removal, can take place in the VR with the associated requirements regarding real-time, or alternatively using the exact sweep volume calculation of a CAD core and with the required computational effort. Furthermore, with the knowledge of the process forces and with the aid of the axial values, the tool deflection and the static flexibility of a machine tool can be calculated by means of an MBS simulation. Previously, measured process forces formed the basis for the MBS simulation model. In future research projects, however, these should be replaced by calculated cutting forces (e.g. cutting force calculation as developed by Kienzle [11]).

Acknowledgements

The Cluster of Excellence „Energy-Efficient Product and Process Innovations in Production Engineering“ (eniPROD®) is funded by the European Union (European Regional Development Fund) and the Free State of Saxony.

The project KF2232904PRO "Virtual Reality-based simulation of material removal for cutting processes" is funded by the German Federal Ministry of Economics and Technology.

Supported by:

Federal Ministry
of Economics
and Technology

European Union

Europe funds Saxony.
EFRE
European Regional
Development Fund

on the basis of a decision
by the German Bundestag

References

[1] Zaeh, M. F., Poernbacher, C., 2005. A Model-Based Method to Develop PLC Software for Machine Tools, CIRP Annals – Manufacturing Technology 54, p. 371-374

[2] Pritschow, G., Röck, S., 2004. “Hardware in the Loop” Simulation of Machine Tools, Annals of the CIRP 53, p. 295-298

[3] Sherman, W. R., Craig, A. B., 2003. Understanding Virtual Reality: Interface, Application, and Design. M. Kaufmann. Amsterdam

[4] Ong, S. K., Nee, Y.-C. A. ed., 200., Virtual and Augmented Reality Applications in Manufacturing, Springer, London

[5] Neugebauer, R., Klimant, P., Wittstock, V., 2011. Virtual-Reality-Based Simulation of NC Programs for Milling Machines, In (Bernard, A. ed.): Global Product Development. Proceedings of the 20th CIRP Design Conference, Ecole Centrale de Nantes, Nantes, France, 19th-21st April 2010. Springer-Verlag Berlin, Heidelberg, p. 697-703

[6] Leu, M., Maiteh, B., Blackmore, D., Fu, L., 2001. Creation of Freeform Solid Models in Virtual Reality, CIRP Annals – Manufacturing Technology 50, p. 73-76

[7] ISO International Organization for Standardization, 1997. Information technology – Computer graphics and image processing – The Virtual Reality Modeling Language (VRML) – Part 1: Functional specification and UTF-8 encoding

[8] Bell, G., Carey, R., Marrin, C., 1996. The Virtual Reality Modeling Language Specification, Version 2.0, ISO/IEC WD 14772

[9] Altintas Y., 2006. Manufacturing Automation: Metal Cutting Mechanics, Machine Tool Vibrations, and CNC Design, Cambridge University Press, p. 65ff.

[10] Grothe, K.-H., 2009. Springer handbook of mechanical engineering, New York, Berlin, Heidelberg, p. C22

[11] Kienzle, O., 1951. Bestimmung von Kräften und Leistungen an spanenden Werkzeugen und Werkzeugmaschinen, VDI-Z, 94: 11/12