

Development of virtual environments for Mechanical Engineering

ANTONIO VALERIO NETTO

University of São Paulo – ICMC/USP

Av. Trabalhador São-carlense, 400 13560-241 - São Carlos – São Paulo – Brazil

aneto@icmc.sc.usp.br

MARIA CRISTINA FERREIRA DE OLIVEIRA

University of São Paulo – ICMC/USP

Av. Trabalhador São-carlense, 400 13560-241 - São Carlos – São Paulo – Brazil

cristina@icmc.sc.usp.br

Abstract: - We outline a procedure for implementing a virtual CNC lathe prototype using software for creating virtual environments. The prototype constructed focus on the lathe's interlocking system (its functionality) and on its geometric model (its physical design). This project allowed us to identify the possibilities and limitations of applying current Virtual Reality technology for virtual prototyping of manufacturing machines, and evaluate the complexity associated to product prototyping in manufacturing or assembly. In additional, we suggest the possibility that this new approach can use for three areas: Training, Marketing/Sale and Product development.

Key Words: Virtual Reality, Virtual Prototype, Petri Net, Training and Product development.

1 Introduction

Although the entertainment industry is certainly responsible for leveraging the interest of the great public in virtual reality, most innovative technology in the area still comes from applications committed to its professional usage. As research in VR matures and creates increasingly powerful software tools and improved (and cheaper) hardware devices, the technology attracts greater interest from different industrial segments and an increasing number of users and applications. This is reinforced by the fact that, to face competition in a global market, companies must improve the effectiveness of their whole production system. Today's high competitiveness levels pose a demand for a total remodeling of productive systems, starting with the replacement of obsolete and inefficient equipment by modern and productive ones and reaching other stages of product development, such as layout restructuring of existing plants and transportation flows at the shop floor; production planning and control, up to the need of improving labor qualification to ensure better product quality and reduce production lead time and waste [1].

Searching for new ways of providing modernization of the companies' productive systems, the use of VR technology has been proposed for equipment

simulation, training of personnel, production planning validation, plant layout and product prototyping [2].

Several authors report on the use of VR in training and education, demonstrating its efficiency for such purposes [3]. Others [4][5] present studies showing that training supported by virtual reality technology has clearly superior results compared to those obtained with conventional systems.

Sophisticated software for creating VR environments enables to model machinery, vehicles and devices, and to simulate the hardware's actual behavior. Based on geometrical and topological information, on the results of simulation obtained from modeling tools combined with kinematics' calculations, material, tolerance and other available product information, it is possible to generate computerized prototypes for realistic presentations.

The approach can save money and reduce development cycles, allowing designers to perform validation and training sessions [9]. Virtual prototyping is now a major VR application in project automation [6], providing an important step towards the efficient development of the final product and allowing interaction with the product even in the early stages of development [8]. Several researchers illustrate and justify its practical application, particularly in the automotive industry [7]. There are reports on the perspectives of virtual prototyping in

the market [16], on advanced techniques for product modeling based on virtual prototyping [17], and also examples of industrial applications [8].

Major advantages of virtual prototyping in industrial processes are:

- **Time Reduction:** The time parameter is a most important factor in a production cycle, as *time-to-market* is the marketing key differentiating industrial competitors, and virtual prototypes can significantly reduce development cycle times.
- **Cost reduction:** Virtual prototypes can reduce the number of physical prototypes that must be constructed, thus enabling a reduction in development time and human effort spent in the project. There is also a reduction in the quantity of tools and materials needed to build physical prototypes. Results can be obtained faster from the virtual prototype, therefore providing early feedback to the project, before production costs are established.
- **Quality Improvement:** different alternatives to a project can be quickly pursued and investigated, at a lower cost, thus improving the validation of appropriate solutions that comply with the parameters specified by the client.

This paper describes an experience in the development of a virtual prototype and investigates the strengths and limitations of virtual prototyping in face of the complex data involved in the production of a prototype for a practical application. A virtual CNC lathe is presented, and the development approach adopted in its construction is generalized into a systematic approach that may be applied to the development of other virtual industrial machines. We also discuss applications of this technology in marketing (sales) and user training.

This paper is organized as follows. In Section 2 we present an overview of the development approach adopted in the virtual lathe project. In Section 3 the development stages along the development procedure are described. In Section 4 we discuss some practical issues related to the project, such as the input and execution of the NC (Numeric Command) program and the introduction of a failure generator. In Section 5 we present the conclusions and final remarks.

2 Overview of a virtual prototyping project

We designed and implemented a CNC (Computerized Numeric Command) lathe¹ virtual prototype emphasizing the design of its geometric model (physical design) and its interlocking system (functionality). A geometric model has been created that precisely represents all the components of a lathe and their interconnection, and on top of this model the interlocking system of a real lathe is simulated. The interlocking system detects conditions out of the predefined work area or invalid command sequences and either aborts the action in course or initiates the necessary corrective actions [10].

Many interlocking operations may be triggered when a part is being machined. As in a real lathe, the interlocking system of the virtual prototype may be triggered as the equipment is used, for example, when the CNC door is opened during a machining operation, or when there is an error caused by the NC (Numeric Command) program. The virtual prototype designed allows the user to check which procedure was adopted by the machine, depending on the failure that caused the errors. It also allows observation of the process functionality and interaction with it, the exchange of existing tools in the lathe tool holder and the animation of the scrap and cooling liquid appearance and withdrawal. There are also interlocking operations for situations where failures occur due to hardware problems, such as failure in the spindle brakes, failure in the electrical energy feeding or problems in the hydraulic and pneumatic systems of the lathe.

A mouse is used as the input device for navigation and interaction with the virtual environment, and a CRT display (Cathode Ray Tube) with a graphics board accelerator is used as the viewing device. The result is a non-immersive environment as far as the feeling of presence is concerned, but an interactive one that provides a user-guided exploration through the 3D environment and its existing virtual entities. The environment responds and reacts according to the user's actions. Figure 1 depicts an overview of the virtual CNC lathe.

The model developed by Lobão [11] was adopted for implementing the interlocking system of the lathe. This model had its interlocking modules simulated and validated by logical diagrams and Petri Nets. The prototype developed is interesting to

¹ A Lathe is equipment for performing a lathing operation, a mechanical process that produces surfaces of revolution with the aid of one or multiple tools. The part is rotated (swept) around a principal axis, and the tool simultaneously moves along a trajectory coplanar with the given axis [18].

support the development of similar equipment. Moreover, it can be used for training operators for the equipment without spending real tools and materials, with no risks of accidents and also with the possibility of simultaneously training several operators without increasing costs.

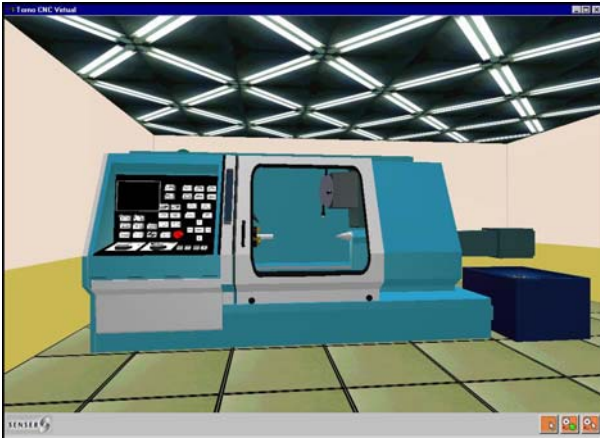


Figure 1 - Overview of the CNC lathe developed

3 Procedure for the CNC lathe implementation

The stages of the procedure adopted for the design and implementation of the virtual lathe are depicted in Figure 2. Basically, the development and implementation procedure has been split into two groups of activities that were conducted in parallel at the beginning of the project, but converged later during the implementation of the logical project characteristics (interlocking). The first group comprises the graphics/geometric development stages, further detailed in the subsequent items. The second group handles the system validation and its functionality simulation using some modeling tool such as Petri Nets to implement the operational logic of the virtual lathe. Once most stages of the first group are finished, it is possible to execute the last stage of the second group, which deals with the implementation of the lathe's functional characteristics. These include the opening and closing of the feeding door; and the operation of the tool-holder cart's X and Z-axes, the turret lathe, the spindle, and the counterpoint motions. Other functional characteristics are the animation of the cooling liquid, the scrap remover, the buttons and displays in the lathe control panel, etc. Sub-sections 3.1 to 3.7 describe the activities embedded in each stage shown in Figure 2. The virtual environment that implements the lathe prototype has been developed with *3DStudio Max* [14] for defining the

lathe's geometric model and *World Up* [15] to create the operational virtual prototype.

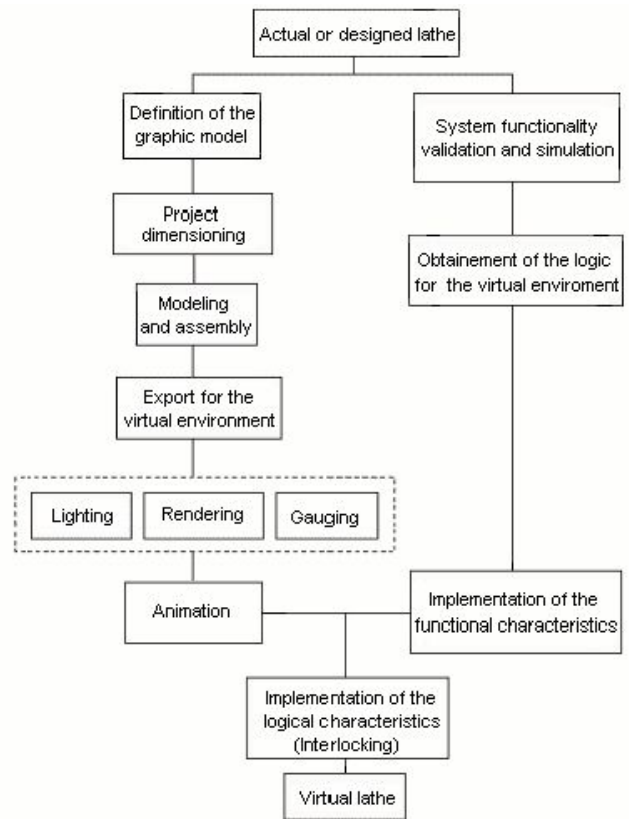


Figure 2 - The different stages of the general procedure adopted for creating the lathe [15]

3.1 Definition of the graphics model

The geometric model that defines the lathe's format and configuration was defined in accordance with its purpose. This particular project does not allow the user to explore the interior of the machine tool, making it unnecessary to model the internal non-visible parts. Many details such as bolts, junctions and external details without functional action were suppressed so as to limit the number of polygonal faces in the model and therefore improve rendering speeds.

Two Traub lathe models were chosen to serve as graphics models, a TND360 and a TNA480 (Traubomatic, 1993). A hybrid of both lathes was created using the external appearance of the TND360 and the machining system of the TNA480, where the tool-holder cart moves on the X and Y axes, not hindering the view of the machining.

3.2 Project dimensioning

A sketch of the lathe was drawn both to show its form and to allow the independent handling of its functional parts in the virtual environment, as the functions to be performed by each element in the virtual system were already known. This allowed to define which components of the machine should be modeled separately. These were the turret lathe, the tool-holder cart, the access door, all the buttons, the spindle, the nuts and the counterpoint. Except for those, all the remaining components were merged and modeled as a single one.

The scale employed in the sketch was 1:10 both from the frontal and lateral point of view that showed only a body section including the lathe door. This information was sufficient to start the modeling and assembling process. It was decided to work with a 1:1 scale, with the modeling environment unit equivalent to one millimeter, thus avoiding the calculation of the scaled model dimensions. The number of decimal digits can be configured in 3DStudio Max, and we adopted a precision of four decimal digits.

3.3 Modeling and assembly

Modeling started by creating a 2D model of the side view of the lathe, from which the body, the door and panel module were extruded along the X -axis. Buttons, keys and display were later added to the panel. This approach has the advantage of ensuring that all the objects created from the same profile would fit each other perfectly.

The objects created were assembled using *3DStudio MAX* [14] alignment tool, which allows objects to be aligned in any of the axes (x , y , z) using as reference the maximum (or minimum) point of an object on a chosen vector, a pivot point or an object's geometric center. From the 2D side view model that defined the window opening (Figure 3), an ellipse was extruded to form the sealing rubber of the windowpane. The windowpane was suppressed in the modeling, as it should be totally transparent to allow observation of the machining process.

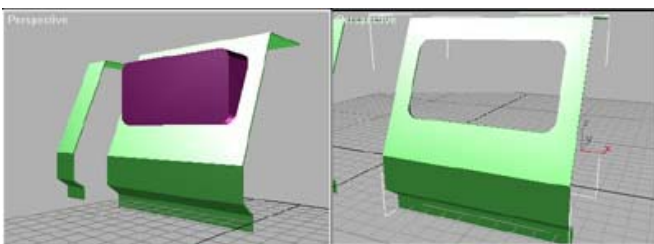


Figure 3 - A detail of the creation of the lathe's window panel [15]

In the following the spindle and the counterpoint were modeled, basically made up from cylinders. After creating the cylinders, a side view enabled to align those objects on the x and y -axes using the alignment tool. The cart that moves the counterpoint was also created from a side model through an extrusion process, and the counterpoint axis was created from a 3D solid model. This modeling approach split into stages allowed many objects to be created exactly in the spot where they should be placed, such as the counterpoint cart and its axis. It simplified the assembly of the lathe's geometric model when exporting it to World Up[®], the software used for creating the virtual environment.

At this point the lathe model reached a geometric complexity that made it difficult to model the remaining components. *3DStudio MAX* viewing tools were used to overcome this difficulty, as they allow hiding or freezing graphical objects. Such facilities supported the creation of the tool-holder cart and the turret. The latter was created using the same process used for creating the counterpoint cart. The turret lathe was initially created from a cylinder segmented into two halves.

3DStudio MAX allows objects to be linked together into a hierarchy. This enabled us to test the lathe's geometric functionality even before it was exported to the virtual environment. For instance, the turret lathe was linked with the cart that moves along the y -axis, and later linked with the cart that moves along the x -axis. Then, moving the cart along the x -axis, the cart on the y -axis and the turret lathe has their movements synchronized accordingly. When only the cart on the y -axis was moved, only the turret lathe followed its movements. The same test was applied to the spindle and to the lathe door, as both have other objects whose movements depend on them. Finally, the cutting tools, the tablets/chips and holders (later assembled on the turret lathe using the alignment tool) were modeled.

3.4 Exporting to the virtual environment

The World Up[®] software [13] features a resource, the so-called *resource browser*, for importing geometry from external programs. It allowed us to import all the objects at the same time, maintaining each one of them as an independent component of the whole model. Once the importing procedure is finished objects may be rescaled through a multiplication factor. In this case all the objects were rescaled by a factor of four, thus changing the original 1:1 scale to a 4:1 one. This was necessary because the lathe showed up too small in the virtual

environment, thus compromising navigation. World Up[®] also has an object hierarchy, which is assembled in the *scene graph browser*. It is represented as a cascading list with multiple levels, enabling an object to be placed “within” another, so that an object responds to scaling and translation operations performed on its parent object in the list.

3.5 Lighting, rendering e checking

The elements within the virtual environment must be properly illuminated in order to be visible. Because the lathe has “internal” components, such as the chamber where the machining is performed, a single light source could not illuminate all its components. Therefore, three directional light sources were added to the default room light which provides the universal standard illumination: one above the lathe’s body (to illuminate all the exterior portion, including the panel), another one inside the machining chamber (to illuminate the part being machined, the counterpoint, the tool-holder and the spindle), and another one to illuminate the rear portion of the lathe.

Rendering the virtual models requires the definition of the visual properties of their surfaces, such as their color and how they reflect incoming light. It may also involve the application of textures (bitmaps) to the object’s surfaces to achieve a more realist effect [19]. All graphical models representing components of the lathe were transferred to the World Up[®] MODELER to have their material properties defined, and then brought back into World Up[®]. This is a simple operation: it is just a matter of selecting an object and choosing *Edit Geometry* from the *Object* menu. The MODELER provides a simple material editor that allows the user to choose the basic material color and define how it reacts to directional lighting. It enables the definition of the material’s opacity level and also the selection of a bitmap for texturing the object’s surface. Textures may be applied to planar, cylindrical or spherical surfaces or to level surfaces, and the positioning and scaling of the bitmap on the object’s surface may be adjusted. In the lathe’s model the buttons, the display panels and the doors received textures. The bitmaps for the buttons were created in Windows 95 Paint, and those for the doors and panels were created in Photoshop 4.

Gauging is the process of checking the limits for the movement of the lathe’s mobile components within the virtual space. The maximum and minimum (x, y, z) values which the moving objects, such as the door and the tool-holder, could reach in their course, were

defined. Such information was later used to program the behavior of those objects in the virtual environment.

3.6 Animations

Animations are programmed in World Up[®] by defining how the properties of the graphical objects vary along a time vector. Some objects were created to represent the movement of the cooling liquid and scrap production by the lathe. These objects are small squares representing instances of the scrap and cooler dynamics in space. These instances were divided in groups, so that it is possible to represent the movement of those elements just by sequentially enabling and disabling different groups.

3.7 Definition of Functionality, Validation, Simulation

To support definition, simulation and validation of the operation of the virtual CNC lathe’s interlocking system logical diagrams and Petri Net models of its behavior were produced [11][15]. Figure 4a and 4b illustrates one of such models for a particular function. The figure on the left shows the logical diagram of the interlocking functions that implement the response to the command for Opening/Closing the lathe’s feeding door. The figure on the right describes the same functionality as a Petri Net. From these models it was possible to derive the procedure that implements the simulation of the corresponding behavior in the virtual model. The same overall approach, whose steps are depicted in Figure 5, was employed to define the behavior of the different functions of the interlocking system.

4 Relevant design aspects in the CNC virtual lathe project

To support the development of the virtual environment a first lathe prototype was created so that designers could get acquainted with the techniques available for programming the *trigger* — a technique that enables the handling of buttons and triggers/drivers in the virtual environment) —, the *avatar* and the *virtual lathing* procedure.

With regard to the avatar, when one is navigating on the virtual environment created by the *software* World Up[®], it is possible to pass through all the solid objects that make up this environment. In this case, there is the programming technique, which prevents this from occurring.

This technique disables the camera motion link with the mouse. When navigating on a virtual environment using the mouse, what happens is the following: There is a virtual camera that follows all the movements made by the mouse. The motion link is the object class of the World Up[®] responsible for the connection.

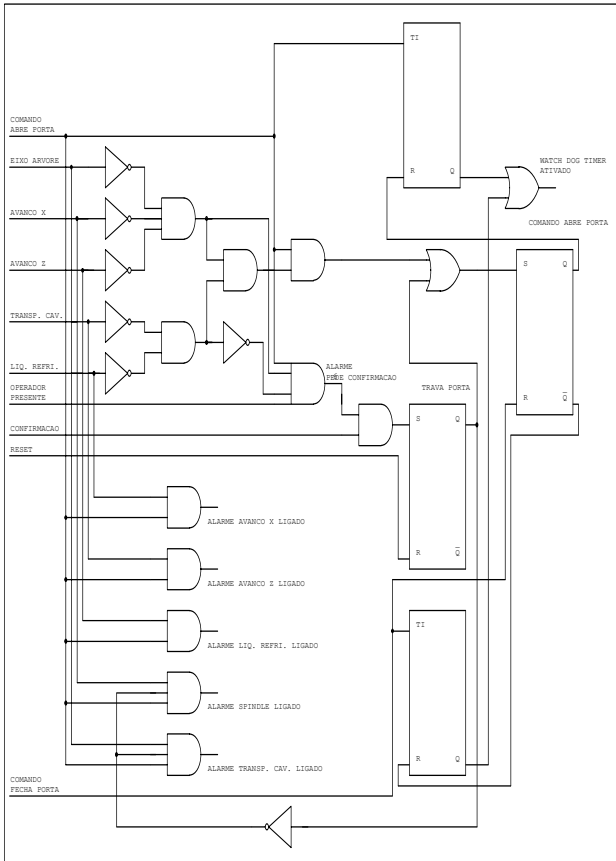


Figure 4a - Modeling of the functionality of the commands for opening/closing the feeding door of the lathe as a logical diagram [15]

Thus, it is possible to detect the collision of this cube when one is navigating among other objects on the virtual environment. By means of the programming in *BasicScript*, the solid with which one navigates (in this case, the cube), designated as “AvatarBody”, is always eject the position immediately before the position where there is a collision, thus preventing from crossing over the solids that make up the virtual environment. These *scripts* also prevent the occurrence of rotations on the camera axis (rolling), keeping the virtual horizon always in the horizontal position.

When a dowel is undergoing a machining process in a real lathe, it takes a different shape at every machining step, and the final part is totally distinct from the initial one. It was very difficult to simulate this process in the virtual environment in real time

and in accordance with the lathe’s NC programming. The virtual objects are described as polygonal meshes consisting of three basic elements, vertices, edges and polygonal faces [19], but model’s geometry and topology change as a result of the machining process. As changes occur, it is very difficult to ensure the geometrical e topological consistency of the intermediate models when operating on polygonal mesh models. An artifact was employed to overcome this problem. Multiple juxtaposed cylinders were created so that the whole set appeared as a single cylinder that was used to represent the dowel body. Since the machining of a part by the lathe is only capable of producing symmetric changes along its longitudinal axis, this solution simplified manipulation of the dowel body, as it was much simpler to modify the radii of each component cylinder and then generate the corresponding polygonal approximation than to manipulate the model’s vertices, edges and faces directly.

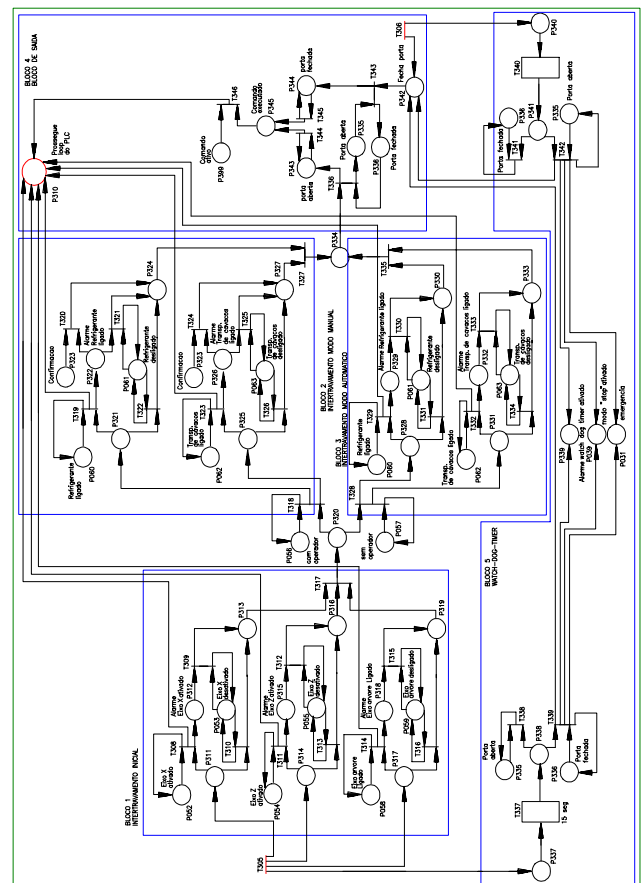


Figure 4b - Modeling of the functionality of the commands for opening/closing the feeding door of the lathe as a Petri Net [15]

By juxtaposing small cubes over the tools’s surface, as depicted in Figure 6, it was possible to

realistically simulate the lathing process. A *BasicScript* program checks the occurrence of a collision between the cubes of a certain tool and some of the cylinders making up the dowel. If a collision occurred, the radius of the cylinder touched decreases as the tool forwards along its longitudinal axis. The thickness of the cylinders' and of the cubes responsible for the collision detection defines the lathing resolution.

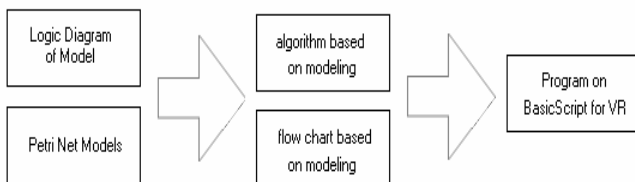


Figure 5 - Steps for deriving the procedures that implement the functionality of the virtual lathe prototype [15]

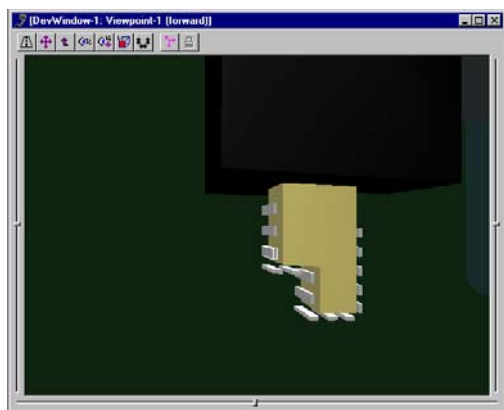


Figure 6 - Detail of the cubes created for simulating the virtual lathing process [15]

4.1 Referencing the virtual part

An EXCEL[®] spreadsheet was used to insert the NC program in the virtual environment. EXCEL exchanges information with the virtual environment's editor through Microsoft[®]'s ODBC 32 *bits* data sharing facility. The procedure is simple: the user accesses an Excel spreadsheet with multiple columns, each one specifying an NC programming command, following the standard approach for programming this type of machine tool. An internal compiler — programmed only for an existing set of CN commands — recognizes the CN commands so that the user can select the desired procedure from the set of pre-defined commands.

4.2 Referencing the virtual part

The geometric model of the CNC lathe was created with reference to a point (0,0,0) defined in the virtual environment and positioned in the interior of the tool cart. This point, the so-called zero-point of the machine, is used for referencing the part for the execution of the CN program. The appropriate referencing of the part (the dowel) in the lathing process is responsible by the correct detection of the physical boundaries of the part being machined. From a point interpolation equations (1) and (2) below, that define the parameterized movement of the tool that performs the lathing process, were obtained for programming the NC program compiler.

(1) $X^V = -2.X + 410$, that is, given an X value in the NC program, this equation determines how the tool advances to a corresponding X value in the virtual environment.

(2) $Z^V = -6,75.Z - 2895$, that is, given a Z value in the NC program, this equation determines how the tool advances to a corresponding Z value.

4.3 Failure generator

The virtual lathe is an ideal tridimensional model in a graphical environment that strictly follows a programming *script*. Its operation is perfect and it never fails. Failures occur in a real lathe due to the influence of external factors in the environment, such as electric power peaks, highly polluted air, human failure, and also due to internal failures, such as fatigue of parts and tools, lack of fluid, etc.

To check the functioning of the Interlocking System with regard to the malfunction of the machine tool, a failure generator was created. This generator consists is accessed by buttons positioned outside of the rendering window depicting the virtual environment. By triggering these buttons one is can change the value of the initialization variables in the programming of the virtual prototype, thus generating a failure when running a simulation of the virtual lathing process.

5 Conclusions

The major difficulty faced in the implementation of the interlocking system responsible for the operation of the virtual lathe was to obtain the algorithms and flowcharts that describe its behavior. The problem here is that algorithms and flowcharts, necessary for deriving the *BasicScript* programs, follow a sequential logic, whilst the Petri Network models are parallel in nature, being very similar to the Ladder

language used for programming industrial CLP [15].

Strength of the software used for creating the virtual environment is that its functional execution model proved very suitable for describing the parallel nature of the machine tool's interlocking system. The environment allows the simultaneous execution of multiple procedures, including the continuous checking of several machine conditions required for further decision-making. However, rendering speed was a major issue, as the final geometric model of the lathe is reasonably complex. The slow rendering updates can be noticed when the spindle is on; its visual rotation speed is much slower than the desired one. This also hinders the user's interactive navigation and the shift of mobile objects (door, back center, etc.) in the virtual environment. Thus, complex environments described by a high number of polygonal faces require a graphics accelerator board for better performance.

Another negative aspect was the lack of a "second hand" for navigation, for it is a limiting condition to manipulate the buttons and the dowel with the *mouse* only. There were situations which required pressing two buttons at the same time, as when interacting with the speed driver of the tool cart, which requires pressing the speed and steering buttons simultaneously, and when placing the machined part out of its niche, a procedure which requires both hands. The suggestion for further developments the project would be the use data gloves to create virtual simulations closer to the desired reality [15].

This project has applicability in three different areas. The virtual lathe can be used for training operators in industries and students in schools, with obvious advantages when compared to training in real machines. The purchaser can also explore it in product marketing, offering the virtual lathe for the potential clients to handle prior to its acquisition, and thus enabling a better interaction and product knowledge. Finally, a most relevant application is to support product development, replacing (or reducing the number of) real prototypes.

The current version of the virtual lathe is robust enough to be used in such application areas. However, some aspects of the current design can be improved to achieve better results. For example, it would be desirable to extend the CN program compiler within the virtual environment to process a greater number of commands and also to perform a semantic verification on the input commands before running them. Also, it would be interesting to add to the virtual environment a program editor in the lathe's panel, which would allow options for editing,

modifying and exchanging CN programs within the virtual environment. This solution would make the environment independent of EXCEL[®].

Another interesting extension would be to refine the graphical and functional detail of the lathe's model. Extensions could be made, for example, to allow users to remove the machined dowel and place it somewhere else. The failure generator could include additional error situations, such as running out of fluid, a situation that might cause problems with the opening/closing of the door, etc. The environment could also be extended to include a variety of lathing tools and a lathing process with improved precision. It could also include an option to model and monitor the wearing out and break off of a tool and the influence of such conditions on the lathing of a part.

6 Acknowledgments

The authors are grateful to Professor **Pat Banerjee** of the Industrial Virtual Reality Institute (IVRI) of University of Illinois at Chicago (UIC). They also would like to acknowledge the support of **T-Systems do Brazil**, company of the group Deutsche Telekom and DaimlerChrysler, and the sponsorship of **CNPq** — The Brazilian Research Funding Agency — and **FAPESP** — The State of São Paulo Research Funding Agency.

References:

- [1] Valerio Netto, A. et al. (1998). Realidade virtual e suas aplicações na área de manufatura, treinamento e desenvolvimento de produto, Revista GESTÃO & PRODUÇÃO, v. 5, n. 2, pp. 104-116, São Carlos - SP, August. (in Portuguese)
- [2] Ressler, S. (1997). Virtual reality for manufacturing - case studies, National Institute of Standards and Technology. <http://www.nist.gov/itl/div894/ovrt/projects/mfg/mfgVRcases.html> (September).
- [3] Rosembaum, L. J. (1995). Mission visualization for planning and training, IEEE Computer Graphics and Application, pp. 12-14, September.
- [4] Wittemberg, G. (1995). Training with virtual reality, Assembly Automation, pp. 12-14, v.15, n. 3.
- [5] Kirner, C. (1996). Apostila do ciclo de palestras de realidade virtual, Atividade do Projeto AVVIC-CNPq (Protem - CC - fase III) - DC/UFSCar, São Carlos, pp. 1-10, October. (in Portuguese)
- [6] Leston, J. (1996). Virtual reality: the it perspective, Computer Bulletin, pp. 12-13, June.

- [7] Dvorak, P. (1997). Engineering puts virtual reality to work, *Machine Design*, pp. 69-73, February.
- [8] Rix, J. et al. (1995). Virtual prototyping - virtual environments and the product *design* process, IFIP Chapman & Hall, 348 Pp.
- [9] Valerio Netto, A. et al. (1999). As novas perspectivas da engenharia com a utilização de ambientes em realidade virtual, *Revista MÁQUINAS & METAIS*, n. 397, pp. 72- 85, Ed. Aranda,- São Paulo – SP, February. (in Portuguese)
- [10] Dombrowski, R. et al. (1991). “Separation of interlocking and regulatory control in programmable electronic systems”, *ISA Transactions*, pp. 19-25, vol.30, n.4.
- [11] Lobão, E. C. (1995). Intertravamento de máquinas-ferramenta: proposta de roteiro para projeto em ambiente de manufatura flexível, Dissertation (Master thesis), EESC-USP, São Carlos-SP, 175 Pp., February. (in Portuguese)
- [12] Traubomatic. (1993). Manual de programação e operação de tornos automáticos, Traubomatic indústria e comércio Ltda., 240 Pp. (in Portuguese)
- [13] Sense8 (1996). Customer applications: information management and analysis, Sense8 corporation, <http://www.sense8.com/> (September).
- [14] Elliott, S. D. & Miller, P. L. (1995). 3D Studio 4.0 técnicas avançadas, Ed. Berkeley, São Paulo. (in Portuguese)
- [15] Valerio Netto, A. (1998). Prototipação de um torno CNC utilizando realidade virtual. São Carlos, 1998. 131p. Dissertation (Master thesis) – EESC, Universidade de São Paulo. (in Portuguese)
- [16] Teresko, J. (1995). Customers transform virtual prototyping, *IW Electronics & Technology*, pp. 35-37, May.
- [17] Ribarsky, W. et al. (1994). Visualization and analysis using virtual reality, *IEEE Computer Graphics and Application*, pp. 10-12, January.
- [18] Gonçalves Filho, E. V. (1994). Comando numérico de máquinas ferramentas, Notas de aula da disciplina de graduação do SEM-EESC/USP, 43 Pp., abril. (in Portuguese)
- [19] Hearn D. & Baker, M. P. (1994). *Computer Graphics*, Prentice-Hall international, INC.