Architecture for Direct Model-to-Part CNC Manufacturing

Gilbert Poon, Paul J. Gray, Sanjeev Bedi
Department of Mechanical Engineering, University of Waterloo
Waterloo, Ontario, N2L 3G1, Canada

and

Daniel Miller
Department of Electrical and Computer Engineering, University of Waterloo
Waterloo, Ontario, N2L 3G1, Canada

ABSTRACT

In the traditional paradigm for Computer Numerical Control (CNC) machining, tool paths are programmed offline from the CNC machine using the Computer-Aided Design (CAD) model of the workpiece. The program is downloaded to the CNC controller and the part is then machined. Since a CAD model does not exist inside the CNC controller, it is unaware of the part to be machined and cannot predict or prevent errors. Not only is this paradigm labor intensive, it can lead to catastrophic damage if there are errors during machining. This paper presents a new concept for CNC machine control whereby a CAD model of the workpiece exists inside the controller and the tool positions are generated in real-time by the controller using the computer's graphics hardware without human intervention. The new concept was implemented on an experimental lathe machine specifically designed to machine complicated ornamental wood workpieces with a personal computer. An example workpiece was machined and measured using a 3D camera. The measured data was registered to the CAD model to evaluate machining accuracy.

Keywords: CNC Manufacturing, Computer Graphics, Automated Machining, Real-time Tool Positioning, Machine Control, Surface Machining.

1. INTRODUCTION

The paradigm for CNC machining (Figure 1) has changed little since its inception. In this paradigm, a CAD model is imported into the CAM software and the tool path is generated offline from the machine controller on a separate computer. The output is a tool path consisting of a list of tool positions that the CNC machine will interpolate between to cut the part.

An immediate observation is that the processes are essentially independent with one-way communication between CAD software, Computer-Aided Manufacturing (CAM) software and the CNC machine. If errors are detected, the individual operators or programmers are responsible for closing the loop and correcting the errors. However, the cost of a mistake can be disastrous for the machine, workpiece or even the operator.

To prevent machining errors, the tool path is sometimes simulated offline before machining. However, the communication is still one-way and any errors detected during simulation are generally fixed by the CNC programmer. The corrected path must then be re-simulated. Generally the simulator can only check if the program is correct. It cannot signal if any of the process parameters in the actual setup of the part and tools are incorrect and it cannot prevent manual operation errors. Furthermore, with this paradigm it is difficult and expensive to integrate new sensors into the system to improve productivity or verify part quality.

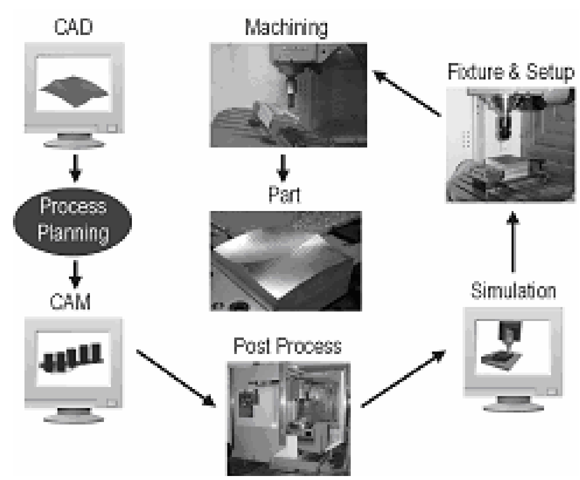

Figure 1 CNC machining process
(reproduced from Gray et. al. [1])

The limitation of this paradigm comes from the division of labor. Without two-way communication between the processes, it will be difficult and maybe even impossible to optimize the process as a whole and to achieve fully autonomous direct model-to-part manufacture.

1.1. Project Objectives

The work by Gray et al. [1] exposed the problems with the current CNC machining paradigm. A new paradigm was proposed to improve the intelligence of the CNC controller for predicting and preventing errors by amalgamating all the processes outlined in Figure 1 into the CNC controller. The main goal of that proposal was to eliminate machining errors. The current goal of our research now is to machine parts directly from their CAD model with minimal human intervention in an optimal fashion. Realizing this goal is not a small task thus, it has been broken-down into smaller manageable components.

The objective of the work in this paper is to present a system for direct model-to-part manufacturing. The new CNC controller is attached to a simple lathe that was designed for machining complex ornamental wood objects. The lathe machine has two agendas: (1) To serve as a safe platform for new control developments directed towards realizing the project objectives; (2) To simplify the process of CNC machining thereby reducing its cost. The reduced cost would make CNC machining attractive to artists and hobbyists.

1.2. Graphics-Assisted CNC Controller

The special feature presented in this paper is the development of a computer graphics-assisted CNC controller for the experimental lathe. The CAD model of the part is downloaded to the controller and the controller computes the tool path in real-time as the part is being machined using the computer's graphics hardware. A Live Model simulation of the workpiece and tool motions resides inside the controller and is updated according to encoder readings from the machine. Knowing the current position of the tool and workpiece, the next tool position is computed and motion commands are sent to the motors. This process is done many times per round. With this machine, no human intervention is necessary to machine the part. There is no offline tool path programming or post-processing.

The following section describes the design features of the Single Axis Lathe that the CNC controller is connected to. The subsequent sections discuss the direct model-to-part manufacturing system developed for this paper. Section 4 presents a machining experiment and accuracy measurement results to verify the system. Future work to expand the capabilities of the direct model-to-part system is presented in Section 5 and the final conclusions are presented in Section 6.

2. THE SINGLE AXIS LATHE

The developments for the work in this paper were implemented on a simple Single Axis Lathe shown in Figure 2. The Single Axis Lathe was constructed with reduced complexity over a conventional CNC machine to simplify implementation and testing of new control algorithms. The simplification comes from the mechanical coupling of the rotational axis (*A*) and feed axis (*C*), which generates a fixed helical tool path around the workpiece (Figure 3).

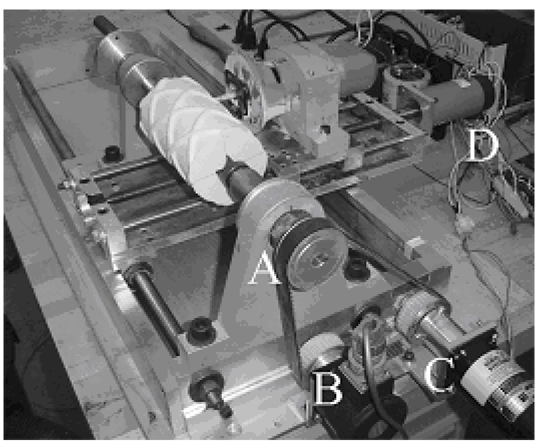

Figure 2 Single Axis Lathe

Motion is generated with stepper motors on both the *A* and *C* axes. An encoder (*B*) on the rotational axis indicates the workpiece rotation, which is used to update the Live Model motion simulator in the controller so that the tool location is always known.

The PC controller communicates with the motor amplifiers via the computer's parallel port. The feed and rotational axes are simultaneously commanded to reach the tool positions. The next tool position is not commanded until the current tool position has been achieved. The sequential output regulates the machining speed by allowing higher feed rates in low curvature sections and slowing down at higher curvature sections of the part.

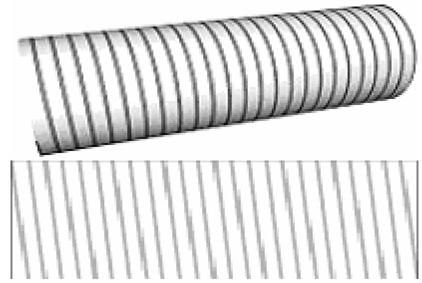

Figure 3 Helical tool path

3. DIRECT MODEL-TO-PART ARCHITECTURE

The Single Axis Lathe was initially designed to offer woodworking hobbyists a simple and inexpensive CNC machine to create complex ornamental workpieces [2]. Though Kaplan et al. [3] used the Single Axis Lathe to machine ornamental tiled patterns, the tool paths were generated offline with specialized software by mapping the helical path to the designed part. The paths were then downloaded to the controller to machine the part similar to conventional CNC machining.

In our approach, the CAD model is downloaded to the CNC controller instead of a tool path. An autonomous CAM system is integrated into the controller, which generates the tool path in real-time during machining without human intervention. With the fixed helical tool path, the Single Axis Lathe does not require tool path planning intelligence, which simplifies our initial implementation of a direct model-to-part manufacturing system. The subsequent sections further describe the architecture of the direct model-to-part system.

3.1. CAD Model Input

The input to our controller is a triangulated STL model of the workpiece. Most CAD software provide triangulation algorithms and STL file output is customizable to user-specified accuracy settings. Machining accuracy need not be compromised using triangulated data. Austin et al. [4] studied discretization of surfaces into triangulated data for tool path generation and gouge detection. Guidelines were provided to achieve a user-specified accuracy based on the cutter dimensions.

The advantage to using triangulated data is that the computer's graphics hardware can be harnessed to assist in computing the tool positions. By using the graphics hardware, some of the computational load on the computer's CPU is reduced, freeing the CPU for other tasks. Since the graphics hardware uses triangle vertices to render surfaces, the STL CAD model does not require any additional software processing for geometric interpretation.

3.2. Tool Path Generation

Using triangulated surface data for tool path generation is not a new phenomenon. Jun et al. [5] developed a method for 3-axis machining and Li and Jerard [6] developed one for 5-axis machining. When using triangulated data, the computer's graphics hardware is ideally suited to simplify some of the tool path calculations since it uses triangle vertices for rendering surfaces. Saito and Takahashi [7] used the graphics hardware for

3-axis tool positioning and Gray et al. [8,9] used the graphics hardware to compute optimal 5-axis tool orientations and positions for surface machining.

For the Single Axis Lathe, tool positioning is similar to 3-axis machining in that the tool orientation is fixed by the kinematics of the machine. To position a tool to the surface, the minimum distance along the tool axis between the tool and the surface must be found (Figure 4). The minimum distance can be found using the graphic hardware's depth buffer by simply rendering the surface region in the shadow of the tool.

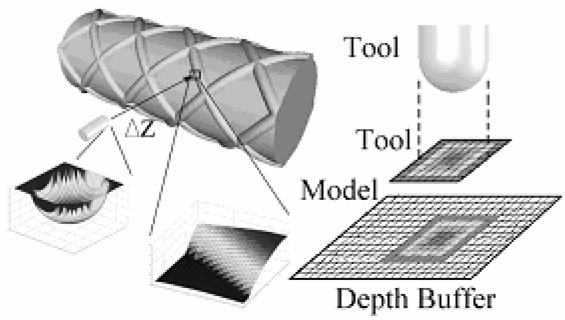

Figure 4 Depth buffer tool positioning

The first step to rendering a surface is to define a volume in space to be rendered. This is known as the viewing volume (Figure 5). All objects outside the viewing volume will be clipped from the rendered scene. The viewing volume can be defined with a position for the eye, a target point for the viewing direction, a view width and height, and positions for the near and far clipping planes.

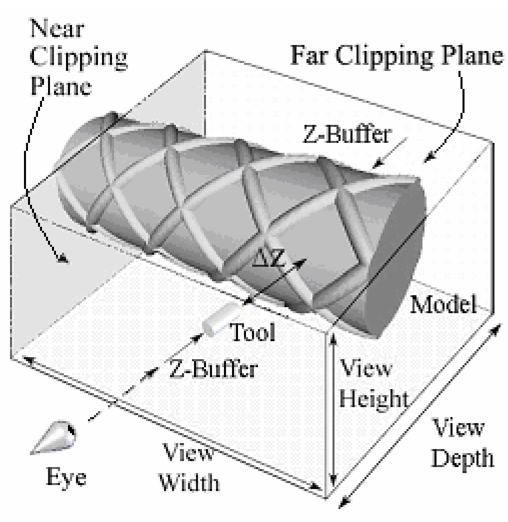

Figure 5 Viewing Volume

The viewing volume's width and height are discretized into a rectangular array of pixels and the depth of the viewing volume is discretized by the depth buffer by the graphics hardware. The depth buffer is usually used for hidden surface removal: A depth value is computed for each pixel, and if the pixel's depth is closer to the eye position, then the color of that pixel replaces the current pixel's color in the frame buffer. Once the scene is rendered, the depth buffer contains the depth position of all the rendered pixels that are closest to the eye position in the viewing volume.

Finding the Cartesian coordinate of a rendered pixel simply requires a backwards transformation of the horizontal and vertical pixel coordinates and the depth buffer value for the pixel to the Cartesian coordinate system of the viewing volume. With an orthographic projection (as opposed to a projective projection), the backward transformation of the pixel coordinates and depth buffer values is only a simple scaling. To compute the tool position, the tool and the surface beneath the tool shadow are rendered separately and their respective frame and depth buffers are captured. The minimum distance between the tool and the surface is computed by comparing their respective depth buffers for pixels that are shaded in both frame buffers. This minimum distance represents the distance that the tool must move along the tool axis to tangentially touch the part surface without gouging. Since the tool geometry and orientation are fixed for the Single Axis Lathe, the tool only needs to be rendered once and the frame and depth buffers can be stored for use with all the tool positions.

The OpenGL API was used in the graphics-assisted tool positioning software. Though OpenGL is a software interface to the graphics hardware, the system is not limited by the graphics hardware. If the graphics hardware does not support the functions needed, the operations are supported in software. Also, memory limitations are not a problem, as the computer will substitute RAM or virtual memory if necessary. However, if the card memory is exceeded or operations are performed in software, the processing speed can be significantly impaired.

3.3. CNC Controller Section

Since the CAD model resides within the controller, the controller can maintain a Live Model simulation of the tool and part motion. This is done by mapping the physical workspace of the machine to the virtual workspace inside the controller. Gray et al. [10] demonstrated such a mapping of an offline material removal simulator to the actual machined part. Currently in our controller, only the tool and workpiece motions are modeled in the simulation. The motion is simulated in the controller with the encoder providing motion feedback from the machine in real-time. The real-time update from the machine is the reason behind the terminology Live Model. Figure 6 shows the simulation during machining.

As the tool and part positions are updated, the tool position can be computed from the graphical display of the Live Model. The tool moves along a helical footprint as the tool path and the process repeats itself until the part is completely machined. Thus, the part is automatically machined without any human intervention simply by downloading the workpiece CAD model to the machine controller.

4. EXPERIMENT IMPLEMENATION AND DISCUSSIONS

An example part was machined to demonstrate the direct model-to-part system. The CAD model of the workpiece is shown in Figure 6, which was created using SolidWorks. A triangulated STL-format file was generated in SolidWorks and downloaded to the machine controller. The workpiece stock material was an 80mm diameter cylinder.

A 6.35mm diameter ballnose endmill was selected and rendered in the machine controller at a known position in the virtual workspace and the corresponding tool depth buffer was stored. With the current machine setup, tool positions are computed every 1.44º of rotation of the part. The helical path has a pitch of 0.00635mm along the feed axis for each revolution of the part. Figure 7 shows the final machined part.

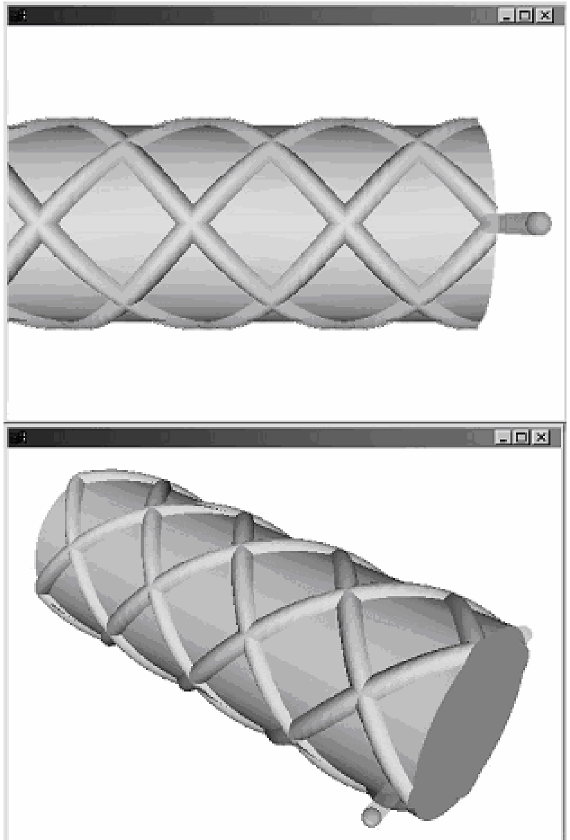

Figure 6 Live model motion simulation

Once a tool position is computed with the graphics hardware, the tool axis traveling distance is converted into a number of pluses for the tool axis stepper motor to move. The Live Model is then updated based on encoder reading and the next tool position is computed. This process continues until the tool shadow has scanned over the entire CAD model.

To investigate the accuracy of the system, the machined part was scanned with a Minolta Vivid 900 3D camera. The scanned data was registered to the CAD STL model data using Raindrop Geomagic Qualify software. The machined part was cut short from the CAD model because the length of the stock material was slightly shorter than the CAD model.

Figure 8 shows the data registration results, which indicates that the machined part matches with the CAD model data at the beginning of the tool path. However, the figure also shows a gradually increasing undercut as the tool path progressed to a maximum of approximately 2.5mm. This gradual increasing undercut is actually due to inaccurate construction of the machine. The feed axis on the Single Axis Lathe was found to be skewed with respect to the workpiece's rotational center. Thus, as the tool carriage proceeds along the feed axis, the undercut linearly increases accordingly thereby increasing the cut part's diameter.

The skew was measured by machining a purely cylindrical part by fixing the tool axis position and advancing the feed axis without any adjustment along the tool axis. The cylinder diameter showed an increase of approximately 1.8mm along its axis over a length of 140mm. If the skew direction between the two axes is in the plane of the workpiece's axis of rotation and the tool axis, then the skew angle would be approximately 0.375º. The remaining error in diameter of the ornamental workpiece shown in the 3D scan can be attributed to the fact that the tool axis is open-loop. Thus, a cumulative error in the tool position can increase during machining if the motor misses pulses because of large velocity changes or high forces.

Figure 7 Machined part

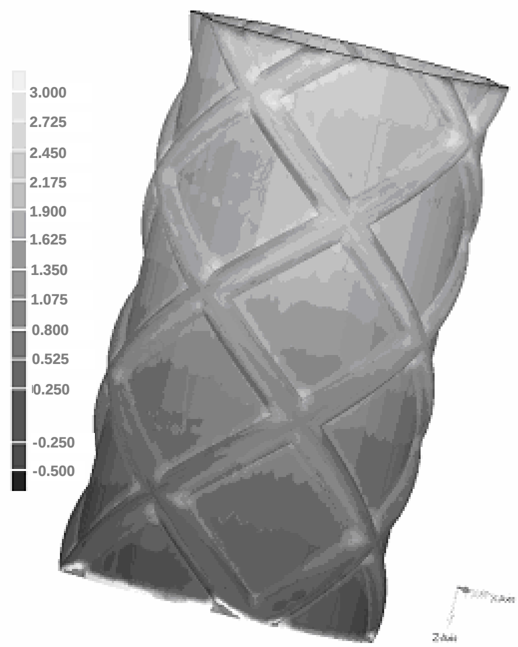

Figure 8 3D scanned data registered to CAD STL model

5. FUTURE WORK

The work in this paper outlines our initial developments towards a more generalized system for direct model-to-part manufacturing to machine industrial components autonomously.

The system was demonstrated on the Single Axis Lathe, which has a fixed helical tool path trajectory. Although this system demonstrates the concept, it is understood that it is an initial implementation and that the system must be applied to a machining platform with more degrees of freedom for industrial implementation. We are currently constructing a new 4-Axis Router CNC Machine (Figure 9), which the direct model-to-part system will be adapted to control using a similar control architecture as the Single Axis Lathe; the CAD model will reside inside the controller, and the controller will autonomously determine the tool path.

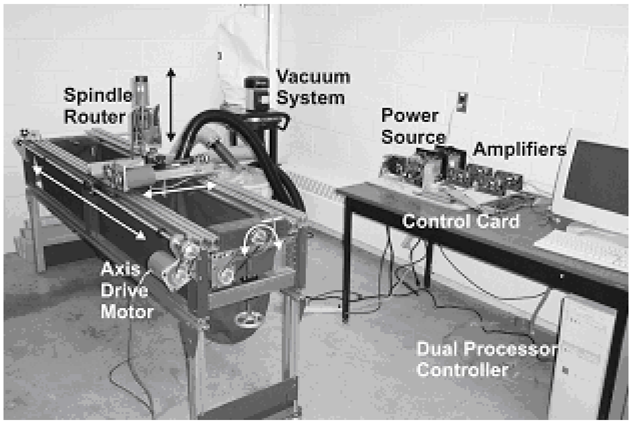

Figure 9 4-Axis Router machine

The greater degrees of freedom in the 4-Axis Router Machine means that an infinite variety of tool path trajectories can be used (e.g. spiral, zig-zag, contour, isoparametric passes). Thus, the system must have process planning capabilities to compute an appropriate or optimal tool path. Furthermore, the process planning intelligence must account for rough and finish machining operations and their appropriate sequence.

A key component of the direct model-to-part system is the Live Model, which must actively monitor the activities in the machine's workspace and model them in the controller's virtual workspace. Our next step will be to integrate the material removal simulation module demonstrated in the work by Gray et al. [10] into the Live Model motion simulator currently in the Single Axis Lathe controller. If error prediction is incorporated into the simulation module, then the system will have the necessary tools to predict and correct errors before they occur. Intelligence algorithms can then be developed for error prediction and correction. The addition will also allow the cutting process to be modeled and the machining parameters could then be dynamically adjusted to optimize machining.

6. CONCLUSIONS

The presented work demonstrates that it is feasible to integrate the part CAD model into the CNC controller to achieve autonomous direct model-to-part manufacturing. Thus, a single unified CAM, simulation and control system reside inside the machine controller.

The direct model-to-part architecture was implemented on a Single Axis Lathe using a PC to control the machine. The controller uses the computer's graphics hardware to assist in computing tool positions in real-time and to maintain a Live Model simulation of the motion of the tool and workpiece during cutting. Machining experiments showed the system to be reasonably accurate, though the machine was not constructed to high tolerances.

The helical tool path limitation of the Single Axis Lathe will be eliminated by the construction of a 4-Axis Router machine. The 4-Axis Router will be the basis upon which the direct model-to-part architecture will be expanded to machine industrial parts. The new system will require intelligence and decision-making capabilities to find an appropriate way to machine the part and eventually to optimize the whole manufacturing process.

7. ACKNOWLEDGEMENTS

The authors would like to thank Kevin Moule for assisting in the 3D part scans and data registration. We would also like to thank the funding agencies that have generously supported this project: Natural Sciences and Engineering Research Council of Canada, the Ontario Innovation Trust and the Canada foundation for Innovation.

8. REFERENCES

[1] P.J. Gray, S. Bedi, S. Mann, "A new paradigm for numerically controlled machines", **Virtual Concept**, Biarritz, France, 2003, pp. 128-134.

[2] N.P. Manos, "Single Controlled Axis Lathe Mill", **Thesis for Master of Applied Science in Mechanical Engineering**, University of Waterloo, 2002.

[3] C.S. Kaplan, S. Bedi, S. Mann, G. Israeli, G. Poon, "A New Paradigm for Woodworking with NC Machines", **The Journal of Computer-Aided Design and Applications**, Vol. 1, 2004, pp. 217-222.

[4] S. Austin, R.B. Jerard, S. Drysdale, "Comparison of discretization algorithms for NURBS surfaces with application to numerically controlled machining", **Computer-Aided Design**, Vol. 29, No. 1, 1997, pp. 71-83.

[5] C.S. Jun, D.S. Kim, S. Park, "A new curve-based approach to polyhedral machining", **Computer-Aided Design**, Vol. 34, 2002, pp. 379-389.

[6] S.X. Li, R.B. Jerard, "5-axis machining of sculptured surfaces with a flat-end cutter", **Computer-Aided Design**, Vol. 26, No. 3, 1994, pp. 165-178.

[7] T. Saito, T. Takahashi, "NC Machining with G-Buffer Method", **SIGGRAPH Computer Graphics**, Vol. 25, No. 4, 1991, pp. 207-216.

[8] P.J. Gray, F. Ismail, S. Bedi, "Graphics-assisted rolling ball method for 5 axis machining", **Computer-Aided Design**, Vol. 36, No. 7, 2004, pp. 653-663.

[9] P.J. Gray, S. Bedi, F. Ismail, "Arc-intersect method for 5 axis tool positioning", **Computer-Aided Design**, Vol. 37, No. 7, 2005, pp. 663-674.

[10] P.J. Gray, G. Poon, G. Israeli, S. Bedi, S. Mann, D. Miller, "Model to Part: A road map for the CNC machine of the future," **7th International Conf. on Adv. Manuf. Systems and Technology**, Udine, Italy, 2005.