To be published on II. International Conference
of PhD Students, Miskolc, 8-14 August, 1999.

ADVANCED SIMULATION OF NC MACHINING OPERATIONS

Oliver Hornyák
PhD Student
University of Miskolc, Institute of Information Science, Department of Information Engineering

ABSTRACT:

This paper is dedicated to the major levels of simulation tasks. It summarises the basic services of NC programming and verification systems. To satisfy complex technological demands (for example time, quality and cost engineering) there are no appropriate simulators therefore new simulators have to be developed with higher performance and capability. The expected tool life is a stochastic variable function of cutting intensity i.e. the rate of stock-removal. In non-stationary case, real time measurement or simulation is required. This requirement necessitates a new generation of simulators. The second part of this paper focuses on the object oriented modelling and simulation architecture for simulation of NC machining operation. In this area some new results will be presented.

INTRUDUCTION

In the course of generating the NC part program setting-out from the geometry of part it is to be done the program file interpreting by the given NC controller. Because of the continuous growing of process complexity, negative consequences of program errors, e.g. damage of worker, machine, workpiece and tool have become extremely serious. Therefore the demand on computerised simulator offering NC program verification became more and more important.

The main levels of simulation tasks are as follows:

(1) Syntactic verification
(2) Semantic analysis
(3) Geometrical undercut and collision test
(4) Tool life simulation
(5) Complex technological simulation.

The main types of simulators are:

- built into the NC programming system
- standalone.

The syntactic verification and the semantic analysis are necessary preconditions for the further investigations. These are basic services of the NC programming and verification systems. The recognition of undercut and collision events as a pragmatic level of simulation requires investigation of the geometrical model of the part and tools too.

SOME NEW TASKS FOR SIMULATION

The modern simulators give reliable estimation on the operation time which is a function of the tool path generation method. By means of simulation some different versions can be tested, and the optimal method can be selected. Considering the term of average or extreme program running conditions the technology bottlenecks of the manufacturing process can be identified.

The latest generation of simulators comprises estimating of tool wears which is the most fundamental activity of machine monitoring systems. The expected tool life is a stochastic variable, therefore machine tools must be equipped with monitoring system which can conclude the necessity of tool change from measuring the main motor current, vibration, force, wear, acoustic emission, etc. Tool life is a stochastic variable function of *cutting intensity* that is also named the *rate of stock-removal* in special literature. Research work of the late years carried-out at the University of Miskolc has shown the fact that the two main state variables namely *tool life* and *machining time* are well manageable functions of cutting intensity [5].

The effect of the NC override of cutting is that the operation cost change in a non-linear way. Because of the increased cutting intensity, the tool life decreases. It causes more frequent tool or cutting edge change. Balancing the machining and tool costs is an optimum searching problem. The advanced simulation of NC machining operations should comprise the modelling of in-process cutting intensity, machining cost, machining power, tool wear.

The complex technological simulation includes the simulation of quality assurance and quality control as well. The major factors affecting quality are as follows:

- force
- vibration
- quantity of heat
- tolerances
- deviations
- surface roughness.

All of these factors can retraceable to elementary volumes and integration of these. The simulation of quality assurance can be executed on the base of this principle. This requirement introduces a new generation of simulators.

SOFTWARE TECHNOLOGY FOR ADVANCED SIMULATION

Development of a new simulator software is a complex problem. Developing engineers should utilize the advantages of the object oriented (OO) technology.

The systems constructed in object oriented way are built from co-operating objects instead of reducing these objects to a procedural set of steps. The object oriented paradigm expounds three major ideas that are necessary for an object oriented programming language to support:

- *Encapsulation* - the hiding of an object's implementation details
- *Inheritance* - the ability to reuse existing class in the creation of new, more specialized objects
- *Polymorphism* - the ability of code to exhibit multiple behaviours depending on the object being used.

The encapsulation makes the ability to decompose the system into co-operating object. The object internal implementation, which is code and data is nobody's business but its own. The object is a sort of „black box". The object has a public interface through which the data can be accessed. This interface constitutes the contract between the object and its users.

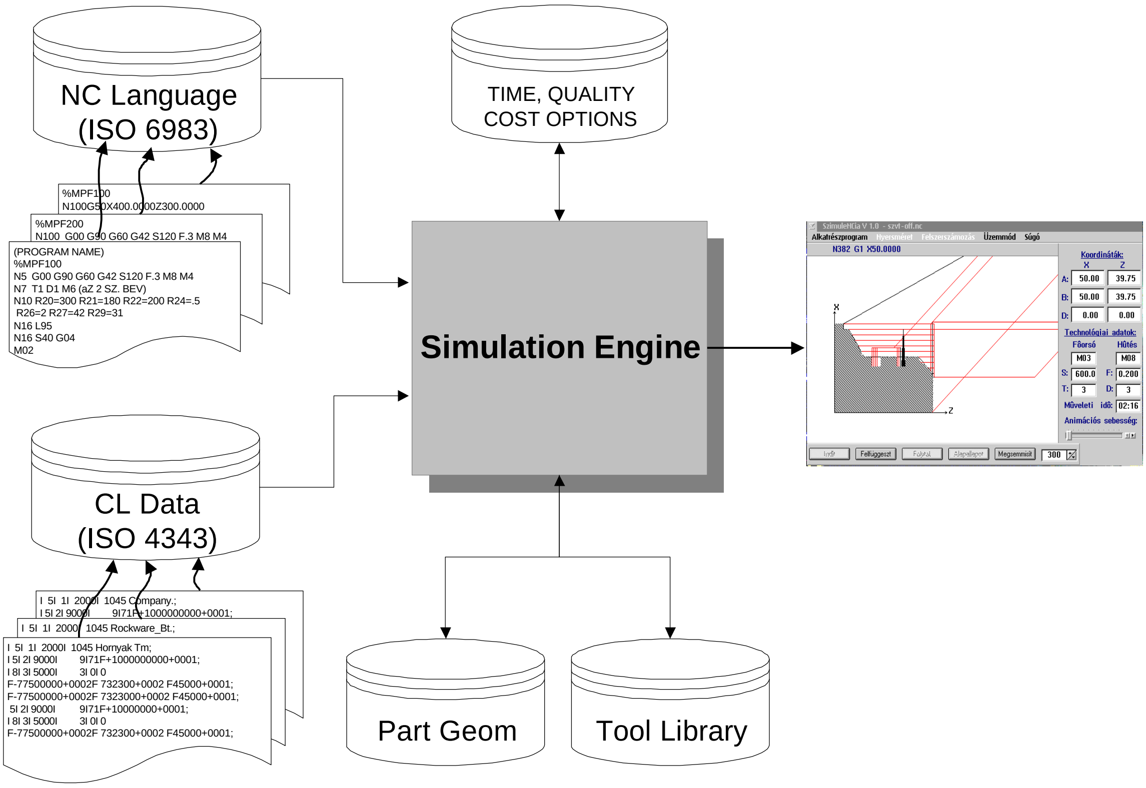

Fig. 1. Simulator for graphical verification of NC part program

During my research work I have developed a simulator for graphical verification of NC part program. The most important step of the object oriented design was to build-up the object hierarchy. The nature of simulation problems is that proceeding of the processes in time should be examined. From the point of view of software technology simulators are aggregations of static and dynamic objects. Such a static object is the *simulator* object. This is the central object of the simulation. It contacts the auxiliary objects for example the *user interface* object, and manage the graphical animation.

Other objects can arise only dynamically, in runtime. The object of simulation should only be dynamic objects. During the object oriented design I have made up a hierarchy of class to decompose the NC program into sequence of *entities*. Such entities are the movement cycles (fast feed, linear interpolation, circular interpolation), coolant, spindle rotation on/off and even the end of the sentence sign. The entities visualization one after the other yields the graphical animation. The ability of object's polymorphism is observable in this example. The visualization method of the object *coolantOn* different to the object *linearInterpolation*. The main purpose of the graphical animation is semantic analysis. The use of the object oriented technology makes it possible to imply additional simulation levels without modifying the code. Unvarying existing interfaces of the existing, well written objects the software can be extended adding further interfaces and components.

To improve performance of simulators they should be modular. It is expedient to use DCOM (Microsoft Distributed Component Object Model) or CORBA (Common Object Request Broker Architecture) as the extension of OO technology.

A significant problem of simulator software is graphical environment. Developing a new standalone graphic library is a notable task. It can not be a part of development of the new simulator software. The basic graphic libraries only support elementary 2D objects. This is the code reusing. Performance of the modern computers permits the use of real time 3D animation. The OpenGL graphical library regards as a quasi-standard in this area. It is accessible to the most frequented operating systems. To improve performance, some of the video accelerators support hardware OpenGL acceleration. An other possible and popular method is using the Virtual Reality. The Virtual Reality Modelling Language is standardized, platform-independent and suitable for network.

In the course of developing a simulator software the network environment must be examined. The access to machining tools database and download of the NC programs from the DNC server are carried-out through network in the real machine tools environment. The simulator must be equipped with the ability of sending and receiving standard manufacturing messages, accessing databases, communicating with other computers.

CONCLUSIONS

Requirements set against the simulators have recently been increasing. An attempt has been made in this paper to show necessity of the object oriented technology supporting multiple CAxx functions, for example CAPP, NCP, PPS, etc.

ACKNOWLEDGEMENTS

The scientific investigation results obtained so far and summarised in this paper are belonging to the PhD (Doctoral) Programme of the University of Miskolc, Subprogramme Production Systems and Information Science. The author is indebted to the Academic Staff of the Department of Information Engineering, especially to *Dr. Ferenc Erdélyi* and to *Prof. Tibor Tóth* for their advice, encouragement and the useful discussions.

REFERENCES

[1] AVGOUSTINOV N.: **Virtual Shaping and Virtual Verification of NC-Programs (Simulation of Part Manufacturing by Means of NC Machines in Virtual Reality)**
[2] CARLSSON, T.E. STRAND ,F.: **A Statistical Model for Prediction of Tool Life as a Basis for Economical Optimization of the Cutting Process**. Annals of the CIRP Vol. 41/1/1992
[3] EDDON G., EDDON H.: **Inside Distributed COM** pp. 11-18.
[4] HORNYÁK, O.: **NC program ellenõrzõ szimulátor fejlesztése**, 1997
[5] TÓTH T., ERDÉLYI F.: **The Role of Optimization and Robustness in Planning and Control of Discrete Manufacturing Process**. The Second World Congress in Intelligent Manufacturing. CIRP Budapest, Hungary, 1997., pp. 205-210.