The Design of a STEP-NC Compliant Agent Based CAD/CAM System


Wolfson School of Mechanical & Manufacturing Engineering,
Loughborough University
Loughborough, Leicestershire, LE11 3TU, UK

ABSTRACT
The introduction of Artificial Intelligence (AI) and knowledge-based systems has provided researchers with a wide range of opportunities over the last 50 years. Today the next generation of AI systems are being developed based on a new and exciting area of research in manufacturing termed “Agent Technology.” This paper describes the design and development of a STEP-NC compliant Agent-Based Computer Aided Manufacture System termed AB-CAM. The system combines STEP-NC compliant feature-based design with agent-based computer aided process planning (CAPP). The proposed system is outlined, consisting of a STEP-NC compliant, Agent based CAPP system with supporting component and resource information models.

1. INTRODUCTION
Customised, rapid and efficient automated NC code generation has been sought after for many decades since the development of the first NC machine tool introduction by MIT in 1952. Machine tools have evolved from simple machines with controllers that had no memory, driven by punched tape readers to today’s highly sophisticated multi-processor controllers. Part programming has developed significantly from the days of manual part programming to sophisticated computer aided part programming systems which are now commonplace in industry. Today, CNC manufacture is approaching the next significant advancement in machine tool part programming since the development of APT, thanks to the rapid development of computer hardware, the availability of more sophisticated software and the development of a common standard specifically aimed at NC programming. The goal of creating a standardised CNC controller and NC code generation facility is now a feasible reality.

This paper presents a methodology to design a STEP-NC compliant CAM system utilising agent technology and feature based manufacture, it is divided into four sections. The first part of the paper provides a brief review of agent technology, the second briefly overviews feature based manufacture together with the development of new CNC standards. The third part of the paper defines a generic STEP-NC compliant CAM system. The structural design of the CAM system is shown through IDEF0 representation. The IDEF0 diagrams illustrate the major activities within a STEP compliant structure. The final part of the paper outlines a proposed agent-based CAM system based on the generic STEP-NC compliant CAM system, termed AB-CAM and outlines the implementation methods undertaken.

2. AGENT TECHNOLOGY
The introduction of Artificial Intelligence (AI) and knowledge-based systems has provided researchers with a wide range of opportunities over the last 50 years. Today the next generation of AI systems are being developed based on a new and exciting area of research termed “Agent Technology”. Agents exhibit many of the attributes required to meet the increasing demand for agility in conventional approaches to design and manufacture.
Chiariglione [1] provides seven attributes to define an agent: autonomy, social ability, reactivity, pro-activeness, mobility, temporal continuity and adaptivity. These seven attributes make the use of agent technology in the development of CAPP/CAM systems particularly useful. The agents can be programmed to accomplish various planning activities such as selecting machine types, cutting tools and cutting parameters to machine specific geometries or features. Three major forms of agents are recognized in the literature and are briefly outlined below.

2.1. MULTI-AGENT SYSTEMS (MAS)

A system in which several agents interact is known as a Multi-Agent-System (MAS) [2]. MAS’s can be separated into two main systems, namely Software MAS’s and Technical MAS’s. Software MAS’s exists currently in the form of web crawlers, search engines in application for the web. Technical MAS consist of both hardware and software, which are usually applied to robot systems.

2.2 MOBILE AGENTS

Mobile agent’s were first established in 1994 with the release of a white paper [3] the computation environment described is known as “Telescript” [4]. Mobile agents are used to access resources that are not located at the host; they are able to migrate to the new host under its own control to carry out tasks. Papaioannou [5] in his PhD thesis provides an excellent overview of mobile agent technology.

2.3 HIGH PERFORMANCE AGENTS

High performance agents can be utilized to improve system performance in MAS’s where duplication of tasks is a problem. The term agent fusion describes the runtime optimizations across multiple, co-operating agents. Fusion’s may be applied repeatedly, to create a single efficient agent from multiple collaborating agents [6]. Applications that are required to run on multi-platform systems can use method known as agent morphing. Morphing, meaning “changing form” is a method in which an agent can change its form from a platform independent form – termed neutral form – and a platform dependant form – termed its native form. Zhou and Schwan [6] explain the mechanics of agent morphing.

3. EVOLUTION OF NC TECHNOLOGY

The current standard of programming NC machine tools has had no significant change since the early 1950’s when the first NC (numerical control) machine was developed at M.I.T. (Massachusetts Institute of Technology), U.S.A. These early NC machines and today’s NC machines continue to use the same standard for programming namely G & M codes based on the ISO 6893 standard [7].

Since the 1970’s significant developments have been made towards more automatic and reliable computer numerically controlled machines with new processes such as punching & nibbling, laser cutting, and water jet cutting which are now common place. The advent of the Computer Numerical Control (CNC) brought a massive improvement in the capabilities of these machines. Currently CNC machines provide the ability of multi-axis, multi-tool, and multi-processes manufacture. These capabilities have made the programming task increasingly difficult and off-line software tools for CAD/CAM a necessity for efficient code generation. Though these developments have revolutionized CNC processes and capabilities, the programming language has basically stayed the same with G/M code programming which was developed in the 1950’s and later became the ISO 6983 standard that is based on the tool path and machine status description.

Ever since the beginning of CAD and CAM software, the problem of a model’s portability from system to system was one of the key issues to spread the use of these tools. Many solutions were proposed in the direction of a standard way of data exchange such as SET, VDA, and IGES, which were partially successful [8] but were not totally suitable to all the needs of the CAD/CAPP/CAM industry. Thus, the international community have developed...
the ISO10303 [9] set of standards, well known as STEP, which has its foundations in many of the earlier aforementioned standards [8].

In parallel with these developments the use of solid modeller’s has brought the possibility of a more realistic and reliable CAD model, however the pure geometric solid modelers (GSM) were not developed enough to take the designer’s intent to the manufacture engineers. Feature technology has been researched since 1976 [9] and was expected to fill this gap between the design and manufacturing environment. However, neither the design intent nor the production planning could be transferred completely to the CAM systems. Most of the information created by the designer was lost, or needed to be re-entered, during the process-planning phase, which included the creation of NC programs by CAM systems. Some researchers have tried to solve this problem using feature recognition [10,11,12]. The essential problem is in the creation and exchange of models where different feature recognizers’ give different results for the same problem. By using a design by features approach, within a GSM based CAD system, where the modelling history is generally stored via feature design interactions. Though once the data is exchanged from CAD to CAM the feature information is lost and then most of the work must be done again with the additional problem of not understanding or assuming the designers intent. This problem has been addressed in the standard ISO 10303 with the Application Protocol 224 (AP224) [13], which provides a set of standardized machining features for use in process planning.

4. STEP-NC COMPLIANT PROGRAMMING

The software evolution for NC programming has seen a number of generations, from the beginning of hardwired machines, with manual block to block programming, to APT (Automated Programming Tool), ADAPT[14], AUTOMAP, COMPACT II, and UNIAPT[15] and the extensions of APT such as EXAPT[16], EXAPT II, and EXAPT III [14] to the modern graphic interactive Computer Aided Manufacture (CAM) systems. Today the software and hardware available at machine tools makes it possible to simulate graphically the tool motion, material removed, and use adaptive control for on-line improvement. The current trends are towards open architectures such as OSACA [17] and OMAC [18] where third party software can be used at the controller working within a standard PC operating system.

One further industrial development is the application of software controllers, where PLC logic is captured in software rather than in hardware. Such systems for example the MDSI CNC architecture [19], provides many opportunities to implement open control capabilities, but they are mainly used in retrofitting applications for older CNC & NC machines. Although these developments have improved software tools and the architecture of CNC machine tools, vendors and users are still seeking a common language for CAD, CAPP, CAM, and CNC, which integrate and translates the knowledge of each stage. It is with this aim that the STEP-NC Compliant programming is being developed to provide consistent standards for automatic and quality oriented CNC component manufacture. To this end in the second half of the 1990’s an effort from the international community backed by the International Organization for Standardisation started a major change in the concept of NC programming. Today a new data model for CNC called ISO 14649 [20] is being developed under the ISO technical Committee TC184 in the Sub-Committees SC1 and SC4 in combination with the support of the IMS project named STEP-NC in Europe and Asia, and Super Model in the USA. Contrary to the current NC programming standard ISO 6983, known by G/M codes, the ISO 14649 is not a method for programming and does not describe the tool movements for a CNC machine. Instead, ISO 14649 provides a object oriented data model for CNC’s with a detailed and structured data interface that incorporates feature based programming where there is a range of information such as the feature to be machined, type of tools used, the operations to perform, and the work plan [21].

For each operation performed on one or more features, a statement called a workingstep is defined. These workingsteps provide the basis of the workplan to manufacture the component. Figure 1 shows the general structure of the ISO 14649 data structure. Figure 2 illustrates the physical file extract of such data for a part with a workplan consisting of workingsteps, for facing, drilling, and pocketing.
in an object oriented high level language as shown in Figure 3 [22].

Programs created or changed at the machine tool can be seen back at the remote workstation as all the data is stored in the CNC controller, which will have all the functionality of the remote workstation, being able to create and modify programs (MDI) which enable the operator to program new components at the machine using standard features. The future of functionality though there have been some attempts to improve this through Manual Data Interface techniques (MDI) which enable the operator to program new components at the machine using standard features. The future of functionality though there have been some attempts to improve this through Manual Data Interface techniques (MDI) which enable the operator to program new components at the machine using standard features.

CAD/CAM functionality (G&M Codes) and the future functionality (STEP-NC).

The Design of a STEP-NC Compliant Agent Based CAD/CAM 4

...suitably termed CAD/CAPP (Computer Aided Process Planning) systems. These systems will concentrate on the generic process plan and ignores the lower level detail. In many respects future CAD/CAM systems will be more similar. On the surface not much as changed however the way in which data is stored and the method in which part details are included (Workplans).

...on the surface not much as changed however the way in which data is stored and the method in which part details are included (Workplans).

4.1 The Functionality of STEP-NC verses G&M Codes

It is clear that CAD/CAM is changing. The functionality of CAD/CAM is moving away from the low level detail of generating tool paths to a higher process planning level. Figure 3 illustrates the key differences between current CAD/CAM functionality (G&M Codes) and the future functionality (STEP-NC).

At the remote workstation the available functions of both current day and future CAD/CAM system is very similar. On the surface not much as changed however the way in which data is stored and the method in which part programs are created and interpreted are very different. Traditional CAD/CAM systems are aimed at simplifying the job of generating tool paths. The future CAD/CAM system though still capable of this activity focuses more on a generic process plan and ignores the lower level detail. In many respects future CAD/CAM systems will be more suitably termed CAD/CAPP (Computer Aided Process Planning) systems. These systems will concentrate on the role of planning a component at a feature level.

An even greater change will be seen at the machine tool controller. Conventional controllers have very little functionality though there have been some attempts to improve this through Manual Data Interface techniques (MDI) which enable the operator to program new components at the machine using standard features. The future of CNC controller will have all the functionality of the remote workstation, being able to create and modify programs at the machine. An additional function never seen before is a bi-directional information flow which means that programs created or changed at the machine tool can be seen back at the remote workstation as all the data is stored in an object oriented high level language as shown in Figure 3 [22].
5. DESIGN OF A GENERIC STEP-NC CAM SYSTEM

The overall design of the STEP-NC CAM system is illustrated in the IDEF0 diagram in Figure 4. This outlines the main input to the system, which is a product model of the desired work piece, which would generally take the form of CAD model containing information relating to raw material final product design and tolerances. The outputs from the system are two fold. An ISO 14649 part program would be used to control the next generation of intelligent machine tool controllers. In addition the STEP-NC CAM system has the facility to post process this information into a conventional ISO 6983 part program for use on existing machine tool controllers. The reason for the second output is that the author believes that although it is important to design new systems that can generate ISO 14649 programs there will be a significant amount of time before industry invests in new controllers to support the new standard. For this reason, any new CAM systems currently being designed should support both standards.

Additional controls include ISO10303 application protocols (AP) 203/214. These standards are used for interpreting component model geometry and more commonly known in the CAD/CAM world as the STEP language interface. AP213 is an application protocol for use in process planning. AP219 is used for inspection data and results and AP224 for manufacturing feature information. ISO10303-21 describes the physical file format for the STEP-NC output. ISO10303-22, 23, 24 etc describe implementation methods for use with languages such as C++ and Java. Process capability and cutting tool information data structure is supported by ISO14649 parts 10/11 and 111. The machine tool library contains information regarding the machine tool capability and a fixture library support Fixturing capability. Additional mechanisms include a data interface translator to translate ISO10303 AP203/214 component models. A feature recogniser is used to recognise features in the component model and store them in an AP 224 format. The CAPP (computer aided process planning) system provides an infrastructure, which will help the user to plan and program their component. An ISO14649 interpreter would be used to read STEP-NC files. The system is designed as an additional component, which can be added to a commercial CAM system.
5.1 OPERATIONAL STRUCTURE OF A GENERIC STEP-NC COMPLIANT PROGRAM GENERATOR

The sub-level design of the STEP-NC Compliant Program Generator in Figure 5, illustrates the activities, which transform a product model into a process plan and part program. These activities are divided into three major parts namely: a feature extractor activity, generation of STEP-NC compliant process plan and the generation of a controller specific part program. The feature extractor is used to generate process information. The STEP-NC compliant process plan is generated using the information from the feature extractor and interfaces with process parameter libraries, to output an ISO 14649 (STEP-NC) part program [22]. It is envisaged that the next generation intelligent machine tool controllers will handle the tool trajectory calculations. For today’s CNC machine controllers some traditional CAM functionality has been included which will generate tool trajectory information from the STEP-NC compliant process plan and part program resulting in the output of a ISO 6983 part program [7].

5.1.1 EXTRACT FEATURES:

The IDEF0 representation ‘Extract Features’ is required to enable a feature based component model to be generated. This is only necessary if the CAD model is not created from features or the feature tree information has been lost due to translation. The authors propose that an AP224 translator could be developed to capture the feature information from a feature-based CAD model. Currently the feature tree information is lost during the translation from propriety to AP203/214 format.

5.1.2 GENERATE STEP-NC COMPLIANT PROCESS PLAN

The IDEF0 representation ‘Generate STEP-NC compliant Process Plan’ requires two inputs, (i) feature tree information, and (ii) the product model of work piece. The functionality of this IDEF0 representation is to select a process type, e.g. rough mill, drill etc. Select the machine tool, cutting tool and cutting parameters. Define fixture methods and clamping locations and finally to determine the ISO 14649 workingsteps and workplan. The workingstep defines an operation e.g. rough pocket and the workplan defines the sequence of workingsteps. The output is a STEP-NC compliant process plan and part program.
5.1.3 GENERATE A CONTROLLER SPECIFIC PART PROGRAM

The functionality of IDEF0 representation ‘Generate a Controller Specific Part Program’ is required only for today’s CNC controllers. The main input is the STEP-NC compliant process plan/part program. The system also needs the product model of a workpiece as input in order to generate tool path trajectories and check for collisions. The first activity is to generate a data structure from the STEP-NC process plan/part program which can be understood by a commercial CAM system. The commercial CAM system is used to generate the tool trajectories. An NC simulator is used to check for part and machine tool collisions and finally the commercial CAM system will use a post processor to generate a specific ISO 6983 part program. NC simulation, which is now commonplace in commercial CAM systems, represents a considerable problem. The STEP-NC part program is designed to function on many CNC controllers, therefore the CAM systems need to simulate every controller to ensure accuracy.

![Diagram of the Design of a STEP-NC Compliant Agent Based CAD/CAM System](image)

Figure 5: The Sub-level IDEF0 Representation for the Design of a STEP-NC compliant Program Generator

6. OVERVIEW OF THE AB-CAM METHODOLOGY

This research is the culmination of 3 major research topics resulting in a novel approach to the design of a computer aided manufacturing system. The approach termed AB-CAM system uses manufacturing features as a basis for the component model, the evolving STEP-NC (ISO14649) standard for the manufacturing data structure and a multi agent system for intelligent decision-making. The authors’ view of STEP-NC is that it is more of a detailed process plan than traditional machine code therefore the logical application of agents to generate STEP-NC code has been adopted. STEP-NC is a very new standard; some commercial CAM vendors are adapting their software to generate the code, which is based on features as the structure of STEP-NC relies heavily on features. However, there is no function to generate code automatically. Current implementations are generating STEP-NC code in the form of additional post processor with dialogs, which provide any additional data. The AB-CAM system generates the process plan and therefore the STEP-NC code automatically. A feature model is created which contains some simple manufacturing information such as corner radii; the agents then take responsibility to populate the required data to generate a STEP-NC part program. The AB-CAM methodology is based on the framework described in section 5 and the operational structure of a generic STEP-NC CAM system described in section 5.1. The
The Design of a STEP-NC Compliant Agent Based CAD/CAM methodology can be broken down into four major parts; Component Model, Resource Model, Multi-Agent process planner and a post processor. Figure 6 illustrates the relationships between the major components.

The component model contains information relating to the component geometry and tolerances. Each component comprises of a number of features which are used to describe the geometry. This feature information is fed into the multi-agent process planner.

The process planner consists of a CAM facilitator. This is the main communication agent within the multi-agent framework its primary role is to interoperate the component model and from it generate Manufacturing Feature Agents (MFA). There are two kinds of MFA, an autonomous MFA and a cooperating MFA. If the component model contains a single feature or several features such as holes or pockets etc. the facilitator agent will generate a new autonomous agent such as a ‘hole’ agent and give it a name e.g. ‘hole1’. The ‘hole1’ agent is created with all the feature information from the component model. The MFA contains a STEP-NC compliant data structure in which it will store the STEP-NC manufacturing data. When all the MFA’s have been generated (1 for each feature) Each MFA sends a message to the facilitator to ask if any other MFA’s interact. Interactions can take the form of overlaps where two or more features cross each other, precedence where one feature is above the other, e.g. a hole in the bottom of a pocket. Thin wall where two features are very close together and the material left between them is less than say 3mm. If the ‘Are there any interactions’ answer is no the autonomous MFA searches the relevant databases to find all the required STEP-NC data required to machine the feature as an autonomous entity. If the answer returns with a yes the facilitator combines each interacting MFA into a cooperating MFA. The cooperating MFA will then determine the correct STEP-NC manufacturing information for that region of features. The final component of the multi-agent process planner is to optimise the operation sequence or in STEP-NC terms optimise the workingsteps. This agent contains various rules that could be user defined to determine the best solution. For the purpose of this research the agent is programmed to minimise tool changes and therefore calculates the sequence of workingsteps, which necessitates the least number of tool changes.

The resource model contains all the required information needed by the MFA such as machine tools, cutting tools, feeds/speeds and Fixturing. For example a pocket that has a corner radius of 5mm will therefore require a 100mm diameter tool to finish the corner. The MFA can select the tool from the cutting tool library, then cross-reference the tool with the component material and determine the required cutting speeds and feeds. The resource model takes the form of an access database, named the ‘AB-CAM Database’. The database is structured to comply with the STEP-NC data structure. Each MFA creates a permanent record of the manufacturing information generated during the process-planning phase within the AB-CAM Database.

A post processor has been developed that interrogates the AB-CAM database for a selected component in order to generate a physical STEP-NC file. This file can then be passed to the CNC machine tool to manufacture the part.
6.1 IMPLEMENTATION OF AB-CAM

The AB-CAM system incorporates a design by features methodology. The commercial CAD/CAM system PowerMill from Delcam Plc has been used as the base product to implement the AB-CAM system. Dialogs have been created in PowerMill to define ISO14649 compliant feature information and an OLE link is used to create these features within the PowerMill software, as shown in figures 7. In the future it is envisaged that an AP224 file will be read to define the feature geometry so that the designer will not be limited by the functions within PowerMill, which is primarily a CAM system, not a CAD system. On creation of a feature within PowerMill the information is stored in the Microsoft® Access database. The AB-CAM system has been developed using the Java language, due to its ability to be platform independent and Java’s supporting libraries methods for the development of intelligent agents. The system’s data structure is based upon ISO14649 and is able to take information from Microsoft® Access databases which are populated by the PowerMill dialogs to generate the component model. Once the agent-based process planning is complete the user may view and edit the manufacturing solution via a number of dialogs. Finally the data is post processed into a functional STEP -NC part program as illustrated in figure 8.

![Figure 7: STEP-NC Pocket Definition within the AB-CAM System](image1)

![Figure 8: STEP-NC Part Program produced from the AB-CAM System](image2)

6. CONCLUSIONS

This paper has presented a methodology for an agent-based STEP-NC compliant CAM system. The authors believe that this approach to implementing STEP-NC has significant advantages. It can be bolted on to many commercial CAM systems providing there is an ability to retrieve feature information. The multi-agent framework provides a method to automatically generate STEP-NC code by performing process planning decisions. There is also potential to modify the rules to represent the local working practices of a machining company. Finally the system is platform independent due to its implementation in the Java programming language that can operate on any platform that has installed the java virtual machine.

7. ACKNOWLEDGEMENTS

The authors wish to thank the Engineering and Physical Science Research Council (EPSRC), Loughborough University, UK and UDESC-State University of Santa Catarina, Brazil in support of the studentships to undertake this research. In addition the authors wish to thank Dr S Hobbs and Ms M Hills of Delcam Plc for their support.
REFERENCES


