Journal for Technology of Plasticity, Vol. 26 (2001), Number 2

MANUFACTURE OF INDUSTRIAL FORMING TOOLS FOR PLASTIC PARTS WITH INDUSTRIAL ROBOTS

Pahole I., Drstvenšek I., Balič J.
University of Maribor, Faculty of Mechanical Engineering, Laboratory for Intelligent Manufacturing Systems, Maribor, Slovenia

ABSTRACT

The paper describes a universal postprocessor initially intended to translate NC code for non-standard NC controllers. By adding arithmetic modules and connecting it to the technological database (TDB) it's functionality is broaden to a wider range of production planning problems, such as quick calculations of costs, evaluation of production possibilities and disposability of tools, etc. Thus, an automation tool is gained that enables a company to participate in a virtual manufacturing market and quickly and effectively evaluate jobs offered.

1. INTRODUCTION

Since the beginning of the NC technology development we are witnessing a "flood" of different "standards" for NC code that aggravate a uniform control of different devices. The problem becomes even more distinctive because of ever faster development of NC machines and CAD/CAP software on one side, and on account of increasingly complicated designers' demands. To make the problem more "colourful" industrial robots (IR) are becoming popular as machine tools for tasks where the tolerance field is wide enough to cover some control errors. Controlling devices on such robotic machine tools are usually unique and totally different from anything that we are used to on NC machines. They can be placed somewhere between NC and typical IR controls with languages that strongly differ from any NC standard.

Besides these common troubles a new principle of offering and accepting of production tasks, a so-called virtual manufacturing, recently emerged. This "production" method takes place in the World Wide Web, where one side offers a job - places an order to the web, and the other answers with an offer for realisation. If it will also take it over, and accomplish it depends on whether it is fast and good enough to compete with other subjects that also entering the web in a search for

opportunities. An advantage in such circumstances mostly depends on a reaction time in which the company is able to effectively evaluate an order's demands and it's own possibilities.
All the listed situations require a quick evaluation of concrete working demands. In the role of order's performer one usually doesn't need to know all designing details so a model of a product isn't needed. In most cases an NC program, which one can transfer into domestic code and evaluate it, is completely satisfactory. In any case a tool is needed which will enable a quick and simple way for code transformation and it's evaluation.

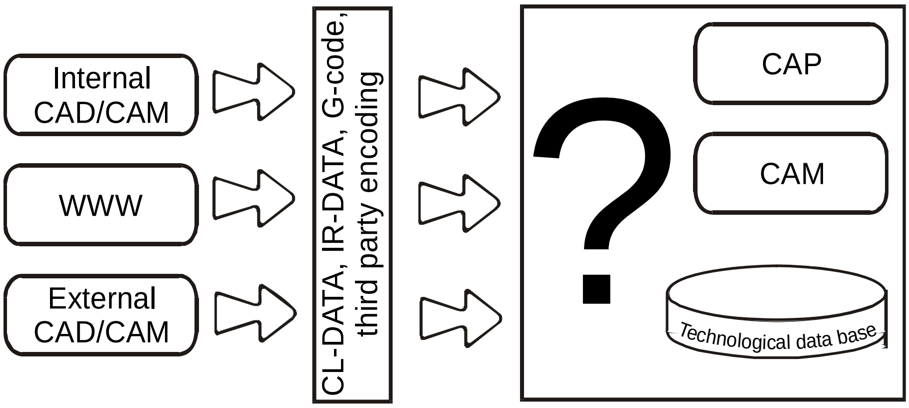

Figure 1. The problem statement

1.1. NC CODE'S APPLICABILITY

Nowadays we should ask ourselves why to use such an archaic thing as NC code at all? In a computer networks' era and with all kinds of machine tools' communication terminals on a market the CAD/CAM transfer should be done directly without such a historic intermediator. Modern CNC machine tools are equipped with powerful computers with capabilities far beyond reading a tape and performing simple interpolations. They can take over the network transfer tasks, calculations, and even processing of CAD files. On the other hand the market is filled with different CAP software for tool path calculation that originally writes the data in their own CL-DATA like formats. The latter again requires a postprocessor to be translated into a proper NC "standard" code. So, why not skipping this step and directly control the machine via the original file? Why preserving the NC code? Or even, why using a file at all? The answer is versatile and hidden in many areas involved in the process of production automation. Firstly, CAP software developers are gaining their market advances by introducing slight differences into their code. So one cannot rely on them to stick at only one standard. Secondly, machine tool manufacturers need an exact code to control movements of the machine for what the CL-DATA like codes are to cumbersome and complicated. And at last but not the least, making a new standard requires a lot of effort, costs a lot, and inevitably lead to a new code that will be history the next moment after it's invention.
A solution of this problem is either in a co-operative work of software developers and machine tool manufacturers or in universal software, which will enable knowledge, based translation among different types of code.

2. INTENTION OF THE NC INTERFACE

The NC interface is a kind of software mentioned above. It is intended for post procession of all types of tool path files without consideration to a source or format of the record. Output file can have an optional format, and can be divided into many smaller parts suitable for completion on some older NC machine tools equipped with insufficient RAM space. The interface also solves the problem of control devices without arithmetic units - interpolators and controls with special demands for the format of arithmetic parameters. It contains an arithmetic module, which can optionally transfer among different formats of conics' definitions.

2.1. REQUIREMENTS

Before development of the interface some requirements have to be accomplished to assure that the developed software will cover wide demands of machine tool users as well as production planning engineers and production managers. Principal requirements are shown below.

1. Open architecture.
   The open architecture of the developed software should assure its growth and improvements to fulfil new demands of users. [1]

2. Knowledge acquisition.
   The interface has to gain some knowledge from a user and store it for later use. Knowledge in this case means rules and instructions for translation of one code into another. This requirement is essentially connected to the open architecture, which has to assure a flexible environment for appropriate knowledge acquisition.

3. Technological database (TDB) support.
   Connection to the TDB is a crucial demand in an automated environment [2]. The TDB stores all the data needed in the production process, thus the interface can collect all required data to perform calculations from it. The database architecture is also most suitable to store gathered knowledge and to adapt to possible changes required during the application of the interface.

3. GOALS OF THE NC INTERFACE

The main idea described in former paragraphs mainly shows the goals which interface has to fulfil. But to clarify the importance of the idea for nowadays manufacturing needs, in continuation, main goals are explicitly stated and explained in more details;

1. A post procession of heterogeneous NC files with a common software.

2. Acceptation of any NC program (without any additional demands for the offerer) and its quick and effective evaluation or transformation into a suitable format, respectively.

3. Incorporation of companies with insufficient CAD/CAP/CAM equipment into the virtual manufacturing market.

4. Automation of tool resources verification via a technological database connection.

5. Calculation of main production time via evaluation of tool data and technological parameters gained from the NC file.

6. Accommodation of technological data to the requirements of the tool data gathered from the domestic TDB.

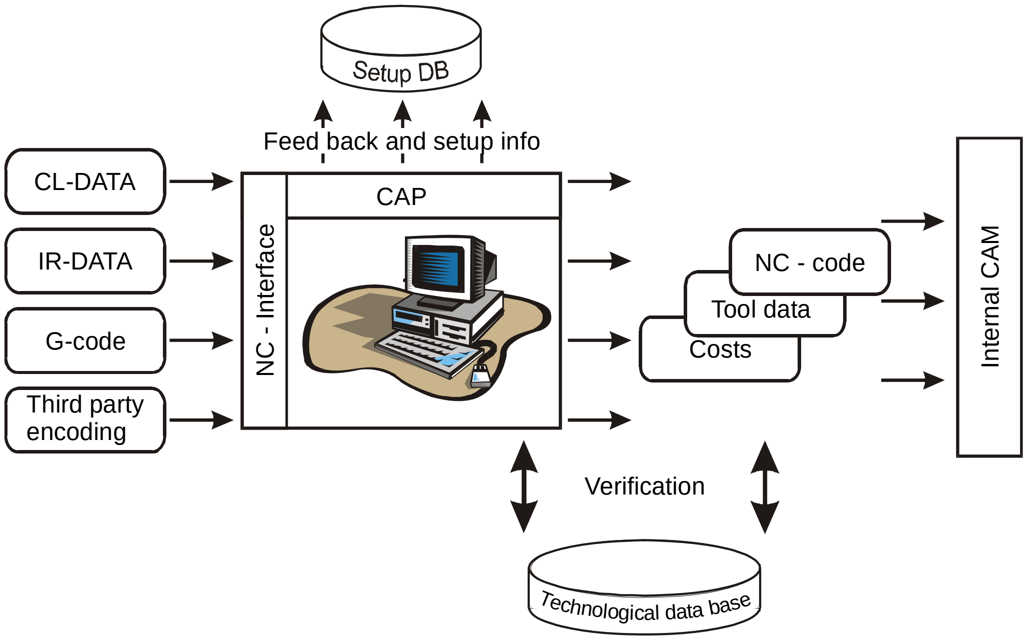

Figure 2. Idea of the NC interface

Accordingly to the requirements listed in the paragraph 2.1 the interface must be able to read any type of a NC code and translate it into an optional file of records. It has to be able of quick adaptations in case of many different inputs as well as output code types. This way the interface will enable small companies to equally compete in the virtual manufacturing market without costly and timely software adaptations. The possibilities of evaluation, cost calculation, and adaptation of the code are far beyond the scope of simple postprocessor. They give a new point of view to the usability of the NC code, from which it becomes a reach data carrier for quick cost calculations.

4. COMPOSITION

The open architecture of the NC interface is gained by a modular structure. It assures simple maintenance and upgrade ability in a case of significant changes in the NC machine tools development. It also makes possible to disconnect a module in a case when it isn't needed for the end user and save some computer space. Also crucial for the user openness and friendliness is a connectivity to the database. Here we talk about two different databases. One is a technological database previously installed at the users for production control purposes. The NC interface can optionally be connected to this database to gain its full usability and use it's calculation module. The second database is a constitutional part of the NC interface. A so-called set-up database is a kind of knowledge-database, which stores the set-up data gathered from the user. Such a composition of pure code and a database engine enables quick adaptation of the interface and optimal search and storage of gathered set-up data. Another advantage of the set-up database is that it can be easily converted into a technological database, what is especially useful in the environment where the technological database isn't employed yet.

The NC interface consists of 5 logical units designed for separate successive functions of post processing as it is depicted in figure 3. The first unit is an input module, which reads the input file and recognises its format. The second is a processor, which translates the read file and performs needed arithmetical operations. The third unit practically concludes basic functions of the interface with formatting and writing of the translated record in the shape demanded by user. The remanding two units cover extended tasks of the interface.

The first -fourth- enables an on-line evaluation of demanded tools properties and comparison of the demands with an actual situation in the technological database. In a case of tool shortage an alarm is triggered and user's action demanded. On the other hand if the geometry of a tool existing in the database is suitable but it's technological parameters (speed, feedrate, and depth) deviate from those required in the input file, the verification module will change the parameters to adopt those required by the tool in the database. It will than execute the calculation of main production time and compare it with the result of the original file's verification [3].

The last, fifth unit communicates with peripheral devices, machine tools, internet, etc. Beside the basic output into the file, there is also a possibility to directly transfer the data into the machine's memory or to the web server, or to any other kind of equipment if so demanded. Each of described units has to be prepared for its activity. To this task a so-called set-up module is dedicated which is a constitutive part of each unit, but can also act as an independent unit. The interface uses it for input file procession, for definition of its format and for preparation of the output file.

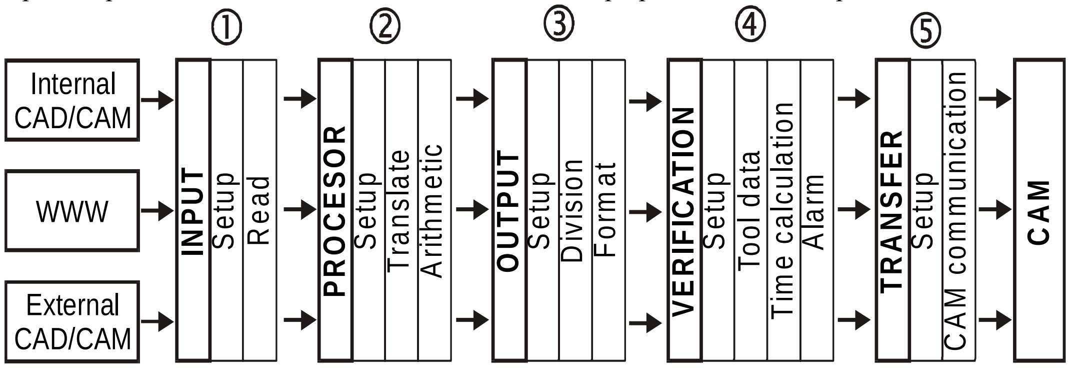

Figure 3. The composition of the NC interface

Before the first use, the set-up module contacts the user via a user interface to gather unknown information. It activates the user every time when a new type of code is detected. The set-up module also takes care for a preparation of transfer parameters for other equipment (RS232, Ethernet and TCP/IP, etc.).

4.1. THE KNOWLEDGE

Knowledge database, knowledge based systems, intelligent manufacturing, etc, are "heavy" words frequently abused by scientists working in a field of production technology. Namely "knowledge" is hard to store. Erudition, proficiency, expertise, know-how, whatever term we use, is information and they are a human domain that can't be stored by a machine. A machine can only store data without any regards to their source. Therefore, knowledge is information gained and stored in their owner's mind that can store some parameters of gained knowledge into files or, in our case, into the database. [4] A sense of the knowledge is in its ability to upgrade by sensing new data and converting them according to conventions incorporated in it. Consequently knowledge is self-upgradeable and as such a characteristic for intelligence. Thus, any system, no matter what its name is only as intelligent as it's user and only as useful as it's creator made it. So-called "intelligent systems" can certainly help their users to act more intelligently, however they cannot act as«standalone intelligence".

The NC interface's structure is based on the knowledge gathered in "standards" for NC code and upgraded by the experience of it's owners. The set-up database is therefore prepared to store five types of data needed to control a NC machine tool. Since the database is of a relational type, terms used further on refer to the relational database's terminology [5], [6]. Each of five data types is stored in a separate table joined to the main table prepared to store NC sentences.
The first table stores *programming functions* that define a programming mode (G90 absolute mode, G91 incremental mode, etc.).

In the second table *movement functions* are stored (G00 rapid move, G01 working move, G02 clockwise circular move, etc.).

The third table stores *switching functions* that enable some tasks like switching peripheral devices on and off.

The fourth table is prepared for *auxiliary functions* such as starting the main spindle, tool changing etc.

The last table stores *geometric information* to which the *movement functions* are related.

A run of the interface starts by reading the first line of the NC file. The user needs to determine single parts of the line and give them an appropriate meaning according to five data types listed above. Parts of the NC line - sentence - are then stored into tables building a set-up base for further use.

Number of tables can of course be changed if the future would bring some significant changes in the NC coding area. Number of fields in tables is also a matter of changes if so required but most important is that none of these changes can seriously affect the interface's effectiveness because of its divided structure. Namely only the set-up units interact with the set-up database all other parts of the interface's structure are logically separated from the database.

5. RESULTS

The testing phase of the NC interface's use gave some promising results in real production circumstances of a Slovene company which deals with forming of plastic materials. It has been used for a transformation of standard NC files (ISO) into the control language of the 6-axis robot RIKO with control device Stefan PC-106 and control language RRL. Beside the particularities in circular interpolation definition the RRL consists of two modules - programme-, and location- module [7]. The first contains control and technological data, and the other is filled with geometrical data of points among which the orders of the first module are executed. This particularity shows a great flexibility of the interface since no special modifications were needed to accomplish this task [8], [9],11].

Figure 4. Manufacture the industrial tool for forming plastic part with robot RIKO with control device Stefan PC-106 and control language RRL

The company has already adopted the interface as the transformation tool that enabled them to use their robot also as an NC machine tool.

Table 1. Translation of NC code into the RRL

Original NC code	Programme code	Location code
G1 X-6.0 Y-4.0 Z0.0	LINE TO 4 CON	FRAME 4 = -6 -4 0 0 0 0
G1 X-6.0 Y-4.0 Z-2.0	LINE TO 5 CON	FRAME 5 = -6 -4 -2 0 0 0

Unfortunately the evaluation module hasn't got its chance to prove its usefulness in the real production circumstances yet. Therefore only some laboratory tests were done that can hardly be called significant.

6. CONCLUSION

The idea of building the NC interface arose from practical demands of the industry. Many companies deal with the problem of sufficient hardware equipped with insufficient control devices and software support. Usual problems are a lack of RAM and outdated CAP software with inappropriate postprocessors. The solution of these problems is of course much simpler than the one shown in this paper, but considering the possibility of using some "non-standard" equipment as a machine tool lead to the NC interface development.

On the other hand, old standards for NC encoding hide a lot of restrictions that are an issue of the past. To overcome standardisation problems and restrictions an interface has been developed that solves most of known problems that emerged in a long history of successful collaboration among industrial companies and research institutes. It will enable a rapid adaptation to the coming changes without painful and costly modifications of production resources.
The interface solves a problem of:

Non-standard control devices,
Control devices with insufficient RAM space,
Workshops with many heterogeneous machine tools.
Further on the interface gives a competitive chance to enterprises with insufficient CAD/CAP/CAM equipment.

An automatic control of machines that usually aren't controlled by a use of the NC code is also enabled (industrial robots, etc.)

The interface enables checking of suitability of production resources regarding to an order and supervision of tools in the technological database. It practically means a technological evaluation of an order over the NC file [10].

A direct effect of the NC interface application into the production process is in a partial automation of production procedures without a costly adaptation of remaining software.

7. REFERENCES

1. Singh, N.; Computer Integrated Design and Manufacturing, J. Wiley & s., New York, 1996

2. Drstvenšek, I.; Model tehnološke baze obdelovalnih operacij v postopkih optimiranja rezalnih pogojev z uporabo genetskih algoritmov, dissertation, Faculty of Mechanical Engineering, Maribor, 1997

3. Drstvenšek, I., Balič, J.; Technological database as an optimisation environment, Proceedings of the 8th International DAAAM Symposium, Katalinic (ed.) DAAAM International, Vienna, 1997, pp 67-77
4. Pratt,P., Adamsky, J.; Database systems: management and design, International Thompson Publishing, Danvers, Massachusetts, 1994
5. Mohorič, T., Uvod v podatkovne baze, BI-TIM d.o.o., Ljubljana, 1995
6. Tansel, A., Clifford, J., Gadia,S., Jajodia, S., Segev, A., Snodgrass, R.; "Temporal Databases - Theory, Design, and Implementation", The Benjamin/Cummings Publishing Company, Inc. Redwood City, 1993
7. user's guide, ; Robotski krmilnik Stefan PC-106, Institut Josef Stefan, Ljubljana, 1997
8. Balič, J.; Manufacturing systems for the third milenium, Faculty of Mechanical Engineering Maribor, Maribor, 2001
9. Brezočnik, M.; Balic, J., Gusel, L.; Artificial intelligence approach to determination of flow curve; Journal for Technology of Plasticity; Vol. 25; Number 1-2; Novi Sad; 2000
10. PAHOLE, Ivan. Determination of flexibility of manufacturing systems. V: BALIČ, Jože. Manufacturing systems for the third millenium; Faculty of Mechanical Engineering,; Maribor 2001,; 117-129
11. Balič, J.; Contribution to Integrated Manufacturing, DAAAM International Vienna (Ed. B. Katalinic, Vienna, 1999.

PROIZVODNJA INDUSTRIJSKIH ALATA ZA IZRADU PLASTIČNIH DELOVA SA INDUSTRIJSKIM ROBOTIMA

Pahole I., Drstvenšek I., Balič J.

REZIME

Od početka primene NC tehnologije pa sve do sada svedoci smo pojave raznih vrsta "standarda" za NC kod koji otežavaju uniformno upravljanje različitih uređaja. Problem se pojačavao sve bržim razvojem NC mašina i CAD/CAP software-a sa jedne strane, a sve većim i kompleksnijim zahtevima projektanata procesa sa druge strane. Istovremeno, pojavili su se i roboti u sferama gde je tolerantno polje dovoljno veliko da "pokrije" određene upravljačke greške. Kontrolni i upravljački uređaji na takvim robotiziranim mašinama su unikatni i različiti od onoga što se koristi na NC mašinama.

Ovaj rad opisuje univerzalni postprocesor koji je, inicijalno zamišljen da bi "prevodio" NC kodove za ne-standardne NC kontrolere. Dodavanjem aritmetičkog modula i njegovim povezivanjem sa tehnološkom bazom podataka (TDB) njegova funkcionalnost se proširuje na širi krug aplikacija, kao npr. brz proračun troškova, ocenjivanje proizvodnih mogućnosti alata i dr. Na taj način dobijena je jedna mogućnost da kompanija participira na virtualnom proizvodnom tržištu kao i da brzo i efikasno proceni ponuđene poslove.

Razvijeni interfejs omogućava brzu adaptaciju na potrebne promene bez da se moraju obaviti skupe modifikacije proizvodnih resursa. Interfejs rešava probleme:

- *ne standardnih upravljačkih jedinica*
- *upravljačkih jedinica sa nedovoljno RAM memorije*
- *radionica sa puno heterogene opreme (mašina).*

Kreirani NC interfejs je testiran u industrijskim uslovima u jednoj slovenačkoj kompaniji za proizvodnju plastičnih masa.