STEP-NC based Optimization and Smart Industrialization of NC Machining in the context of the Factory of the Future

Christophe Danjou[a], Shixin Xu[b], Julien Le Duigou[a], Nabil Anwer[b]

a. Sorbonne Universités, Université de Technologie de Compiègne, Mechanical Engineering Systems Department, Mechanical laboratory Roberval
UMR UTC/CNRS 7337, CS 60319, 60203 Compiègne cedex, France.
christophe.danjou@utc.fr; julien.le-duigou@utc.fr

b. LURPA, ENS Cachan, Univ. Paris-Sud, 61 avenue du Président Wilson,
F-94235 Cachan, France
sxu@ens-cachan.fr; anwer@lurpa.ens-cachan.fr

Résumé :

L'article propose un nouveau concept pour assurer le retour d'information depuis les Machines-Outils à Commandes Numériques vers les systèmes FAO et la simulation d'usinage. Le but principal de la proposition est de gérer les connaissances d'usinage extraites de la machine pour les réutiliser au niveau de la simulation d'usinage et ainsi fournir une aide à la décision pour la réalisation de futur programme d'usinage.

Pour assurer la capitalisation des connaissances d'usinage, la proposition s'appuie sur une structure basée sur les entités d'usinage. Bien que la structure du standard STEP-NC permette l'exploitation d'une telle structure, son utilisation est impossible du fait de sa faible implémentation. C'est pourquoi la proposition se base sur la chaîne numérique actuelle et propose une reconnaissance des entités d'usinage directement depuis les fichiers STL extrait de la simulation d'usinage et propose une conversion des programmes de Code-G en STEP-NC. Cela permet alors d'interagir avec l'implémentation d'OntoSTEP-NC, une ontologie basée sur STEP-NC, comme base de connaissance.

Abstract:

The paper proposes a new concept to support information feedback from CNC machines to CAM/NC simulation systems. The main goal is to manage knowledge from the CNC machine to the NC simulation to provide guidelines and help programmers in making choices for planning new machining sequences.

The proposal uses a feature oriented approach for knowledge capitalization. As STEP-NC which is manufacturing features oriented, is not implemented yet, the proposal is based on actual manufacturing technologies and proposes a feature recognition directly on STL file from NC simulation and proposes to convert G-Code to STEP-NC in order to interact with OntoSTEP-NC, an ontology based on STEP-NC, implemented as a knowledge database.

Mots-clefs: Feature Recognition, STEP-NC, Knowledge Based System.

1 Introduction

To be more competitive and due to globalization context, aeronautical manufacturers need to improve the production engineering phase by focusing on the triptych: Cost, delay and quality. One of the main aspects which has been highlighted by French aeronautical industries is the barriers of cutting knowledge capitalization. Moreover the use of new materials as Titanium and Inconel add uncertainty on the machining accuracy due to little knowledge on these materials.

As an answer the French research program FUI14 ANGEL (Atelier Numérique coGnitif intEropérable et agiLe) focuses on the machining capitalization and on the control of the digital chain process in order to manufacture the right part the first time. This has been enhanced with the development of the digital chain, with new standard that has been created to exchange data between CAx systems. As a consequence, the main questions this paper addresses is: How to integrate machining knowledge from manufacturing in the production engineering phase for mechanical parts? And how to capitalize the extracted information to provide guidelines for future machining sequences?

To answer these two questions, the next section exposes first a background on the data exchange on the digital chain and presents STEP-NC standard as a promising solution. As STEP-NC standard is not yet implemented, the third section proposes a way to integrate knowledge directly into current G-Code digital chain by using the results of NC simulation. Section 4 presents a solution for machining parameters capitalization based on a conversion of G-Code to STEP-NC in order to be feature oriented which would help to fill OntoSTEP-NC, an ontology based on STEP-NC, as the knowledge database. Section 5 proposes a case study based on STEP-NC reference parts. Finally, section 6 provides a small discussion on the work carried out in ANGEL project and conclude this paper.

2 Related works on STEP-NC

Since the 1970's, machining has evolved transforming conventional machining into numerically control machining. Those CNC machines require information concerning toolpath and cutting parameters compiled into the CNC program. The machining program has to be generated based on a geometric model description obtained thanks to CAD systems. Once the part geometry has been defined, the CAM systems allow generating the toolpath and cutting parameters. To translate this CAM program into readable CNC machine program, a Post-Processor is used based on CNC machine specificity – each machine has its own capability and the Post-Processor adapts the CAM program to the chosen CNC machine. This sequence of tools and systems defines the manufacturing digital chain which is composed by CAD/CAM/Post-Processor/CNC machine. A machining simulation step can be added between the Post-Processor and the CNC machine to simulate the program execution based on the dynamic of the chosen machine.

For long years, expertises from CAx systems (CAD, CAM, Post-Processor, CNC machine, etc.) have been enclosed in specific files. The software uses their own language and disable to have exchanges between the other editors' tools. For example, CATPart and CATProcess will be used respectively for CAD and CAM systems of Dassault Systems, Top'CAM for TopSolid CAM system, G-Code and M-Code led by the ISO6983 standard will be used as specific inputs for the CNC controller, etc.

As explained before, the main problematic is to achieve the data exchange between the CAx systems and the use of this data by other systems. The use of standard formats seems to be a relevant solution to ensure the bi-directionality of information flow between software. This approach allows having the same language for all the CAx systems.

To first achieve the data exchange between CAD and CAE, STEP ISO10303 standard has been developed [1]. The STEP standard is an open and standardized format that aims at promoting the data exchange in a format which is understandable and shared by all. According to Rachuri [2], the STEP standard provides a neutral, sustainable and scalable data exchange format.

In the same way, for data exchange between CAM system and Post-Processor, two main standard formats have been developed: APT and CL-File ISO 4343-1978. Although those standards allow exchanging data with the other systems, they do not allow having information feedback from the post-processor and CNC machine to the CAM system.

To solve this feedback issue, STEP-NC standard (ISO14649 and ISO10303AP238) has been developed – in the last decade – in order to improve the systems interoperability by integrating the process machining data [3]. The STEP-NC standard encompasses machining process, cutting tools description, and a description of CAD features and requirements. This enriched standard format allows having in the same file all the information required for the whole product development process from the early design phase to the machining operation [4].

Contrary to the G-Code program, the STEP-NC program is manufacturing feature oriented [5]. The CNC program consists of a manufacturing operation list organized into entities. The STEP-NC file is structured with entities and each manufacturing sequences are defined as manufacturing feature (Hole, Pocket, PlanarFace…). In Figure 1 are exposed the schema of the different manufacturing features defined in STEP-NC.

Figure 1: EXPRESS-G representation of manufacturing feature in STEP-NC [6]

Numerous works use STEP-NC structure in order to improve the manufacturing efficiency [7]. Newman et al. [8] proposes a literature review on STEP-NC works. It clearly appears that STEP-NC standard enhances the digital chain. Consequently, a lot of research works use STEP-NC standard to process optimization directly on the CNC code. Below are listed major works which use STEP-NC standard and provide enhancement of manufacturing efficiency.

- Xu et al. [9] use STEP-NC as a universal programming for CNC machines. The same CAM program can be spread to many CNC machines. It allows using the same CAM program into many different CNC machines with the intelligent transfer from CAM system to CNC machines.

- Through the use of STEP-NC Ridwan & Xu [10] define an automatic correction of cutting parameters based on the Machine Condition Monitoring. They have developed optiSTEP-NC system which helps to perform cutting parameters optimization.

-Nassehi et al. [11] based on MASCAPP platform propose to use the STEP-NC manufacturing environment to define the best machining sequence based on a multi-agent system. Hence, it provides an optimized CNC program.

- Borgia et al. [12] based on STEP-NC allow automated recognition of feature and generating toolpath based on machining working step. Then a mathematical optimization is conducted.

- In the same way, Zhao et al. [13] define closed-loop machining thanks to STEP-NC to achieve online inspection. To succeed this inspection they have developed a closed-loop between CAPP and CNC machine.

It has been highlighted that the use of STEP-NC standard enhances the optimization of CNC program. In one hand the STEP-NC structure allow reasoning on manufacturing feature thanks to explicit syntax, in the other hand the optimization of CNC program is due to computation. The combination of these two aspects allows having more manufacturing efficiency.

However, most part of this optimization is supported by computing power and STEP-NC structure; this could be improved by adding knowledge re-use. In this way, Zhang et al. [14] propose through UPCi system a process comprehension based on STEP-NC in order to have a knowledge re-use. Moreover, although STEP-NC allows sharing and propagating data in both ways – CAD to CNC and CNC to CAD – STEP-NC is not yet a solution for archiving CNC machining information in order to be integrated in the CAx systems at the right time to help programmers and manufacturers to choose the right parameters for machining sequences. To integrate the machining data and cutting parameters at the right time, information has to be classified and organized previously. Hence a knowledge database would allow improving future CNC program based on similar sequences and features.

Currently as the digital chain and the CAx systems are not STEP-NC ready, this work cannot be implemented yet. Consequently, the previous proposals exposed are not relevant for today application. A solution for using STEP-NC works in current state can be the merge of current digital chain and those works. In this way, there would be interactions between both technologies.

To face the STEP-NC implementation problem, the two next paragraphs propose to have a feature recognition directly from the CNC simulation and to bring back expertise and knowledge guidelines based on feature-oriented analysis in order to improve the CNC program, and then to have a G-Code translation to STEP-NC to allow manufacturing knowledge capitalization into OntoSTEP-NC; Thus despite a non-STEP-NC digital chain (Figure 2).

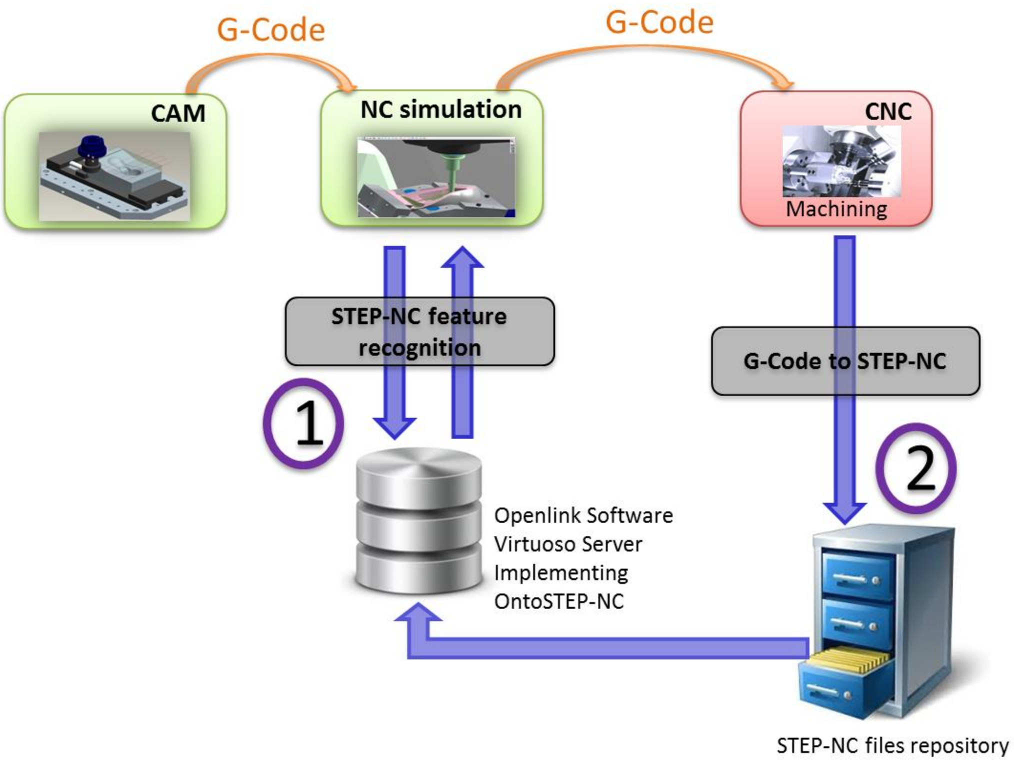


Figure 2: Closed-Loop Manufacturing process (1) Re-use (2) Capitalization

3 Knowledge re-use

3.1 Manufacturing feature recognition

This section aims to present the method of feature recognition from the NC simulation results. The geometric results of a NC simulation are in-process models (IPM), which are most commonly represented by triangle meshes (e.g. in STL/OFF files). The IPMs often have scallops, offcut components, and their triangles are highly non-uniform (Figure 3-1). The recognition method is consisted of two major phases: mesh segmentation and feature recognition. Taking account of the IPM characteristics, a method is proposed to carry out the task. The method includes the following three procedures, as illustrated in Figure 3.

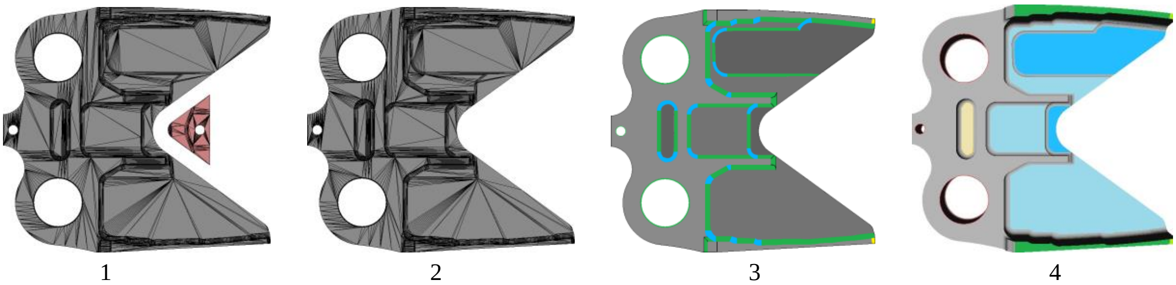


Figure 3: Illustration of feature recognition procedures from NC simulation: (1) an IPM; (2) pre-process; (3) mesh segmentation; (4) feature recognition.

(Figure 3-2) IPM pre-process. a) Build a manifold polyhedral B-rep model, including offcut removal, IPM data error correction; b) Computation of vertex normal vectors, vertex curvatures, edge-concavities/convexities, some attributes of triangle mesh, etc. These values will be used for subsequent processing and for determining some thresholds.

(Figure 3-3) Mesh segmentation. a) Coarse segmentation: identification of sharp edges on the part’s boundary, discernment of narrow triangle clusters, segmentation of planar and most of cylindrical faces. After this operation, an IPM’s triangles are grouped into three categories of regions—narrow triangle regions, small regular triangle regions, and large triangle regions; b) Shape index based segmentation: segmentation of non-ruled faces (such as blending surfaces), mainly corresponding to small regular triangle regions. Shape index is a value calculated from the two principal curvatures to describe the local shape around a triangle’s vertex in the region [15], [16]. A segment is a group of neighboring vertices whose shape index values are within an interval. Nine surface types are defined based on shape index (Figure 4). Note: value 2 is isolated, which does not belong to the shape index interval, for labeling flat vertices. Every vertex of the shape is assigned a surface type label during local surface type recognition. Then connected regions are generated from these clustered vertices. For details, refer to reference [17].

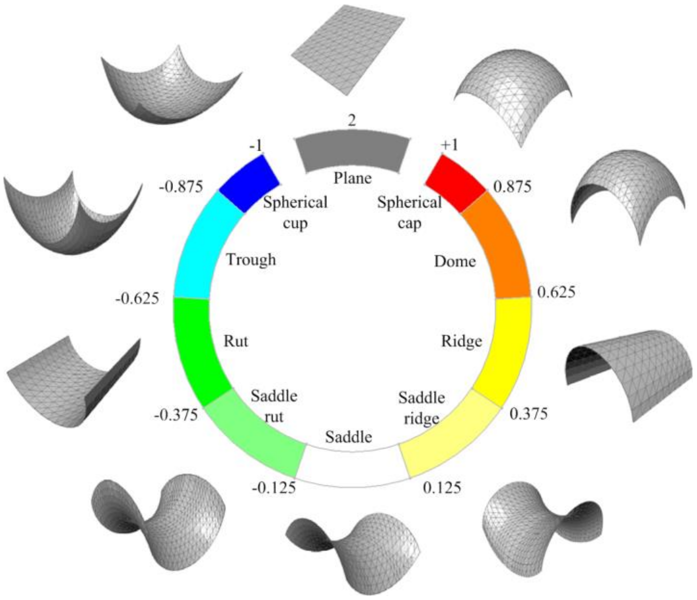


Figure 4: Surface types, shape index scales and color scales.

(Figure 3-4) STEP-NC manufacturing feature recognition. Generally speaking, the feature recognition is performed in two phases. One phase is to identify the feature class (passage, depression, k-protrusion, or k-slot [18]), and then determine the corresponding STEP-NC feature class; the other phase is to extract the STEP-NC feature parameters according to the standard’s specifications. For instance, a blind round hole has the following extracted data: feature id, feature type, axis direction and position, hole diameter, hole depth, bottom type, bottom angle. A blind, closed pocket has the following extracted data: feature id, feature type, profile and its shape, maximum depth, draft angle, access direction, minimum orthogonal radius, planar radius, bottom type. Figure 5 shows the flowchart of the feature recognition algorithm. For details, refer to reference [17].

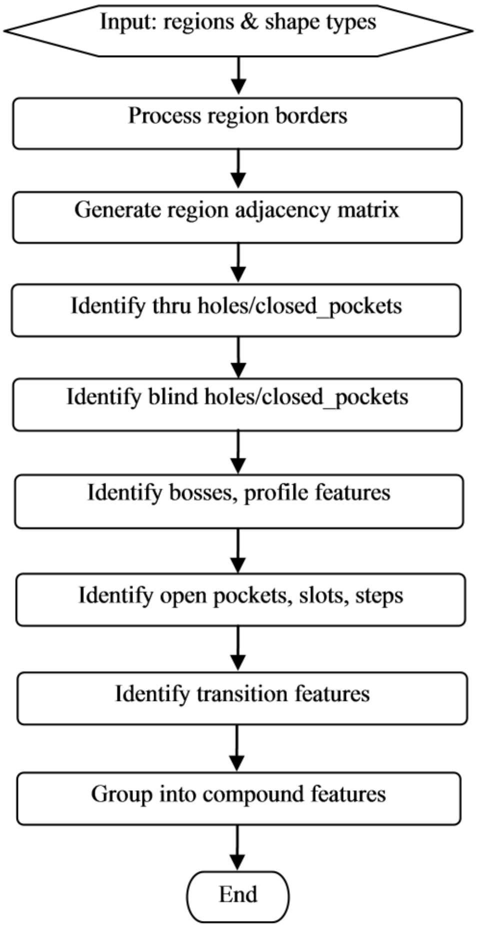


Figure 5: Flowchart of the feature recognition algorithm

3.2 OntoSTEP-NC

As highlighted in paragraph 2, the STEP-NC standard provides an interesting structure which allows having all the information needed for machining. Even though STEP-NC standard allow sharing information in the whole digital chain, its format doesn't allow knowledge capitalization. To provide a data model with a sufficient accuracy level, a new model has been created: OntoSTEP-NC. This approach proposes to create an ontology based on STEP-NC standard – ISO14649 [19]. Thus the ontology provides all the necessary data for CNC machines accuracy.

The challenge to create OntoSTEP-NC consists in translating the STEP-NC standard into ontology. Indeed STEP-NC standard is written in EXPRESS language whereas it has been chosen to use OWL (Web Ontology Language) standard language for the ontology. Each language has its own syntax. That is why ENTITIES in EXPRESS has been defined as CLASSES in OWL. Similarly, ATTRIBUTES and CARDINALITY in EXPRESS are not defined in OWL. To translate them into OWL OBJECT PROPERTIES and DATA PROPERTIES have been used. The complete methodology and process are detailed in Danjou et al. [20].

To set OntoSTEP-NC as a knowledge database it has been chosen to implement the ontology model using the OPENLINK Software VIRTUOSO Universal Server® [21]. Thus OntoSTEP-NC becomes a knowledge database which can wear various cutting parameters according the projects, the cases and the shapes. By using the STEP-NC information structure OntoSTEP-NC implementation provides a way to store all the information concerning the sequences for each feature as: spindle, cutting tools, feedrate, feed per tooth, approach and retract macro, radial and axial depth of cut, overhang ratio, center path overlap, toolpath strategy, etc.

The main point to respect for information feedback consists in sending back the right answer at the right time. Based on this assertion it is mandatory that the information be selected before it is proposed to the programmer. OntoSTEP-NC provides a model in which the needed information for machining is structured and implemented as a knowledge database. In this way, the implementation allows generating queries directly on the OntoSTEP-NC model using SPARQL syntax.

As OntoSTEP-NC is structured feature-oriented, the queries are based on entities of the OntoSTEP-NC model. The feature recognition is the basis of the query. To be relevant those queries must be contextualized. The query refers to the following ordered criterion list: Type of machine-tool, material nature, cutting tool and the manufacturing feature and its geometrical dimensions. It can be launch directly from the virtual machining simulation system.

The query answer then returns all the cutting parameters as a sorted list linked to each manufacturing feature which can be exactly similar or have approximate dimensions. Those lists of parameters can be directly proposed to the programmer in order he can explore and choose the parameters which best fits with regards to the case and sequence planning. This query answer can be interfaced directly in the simulation software to be imported.

The query answers are directly linked with the ability to fill the knowledge database. The more the database is furnished the more the queries answers will be relevant and will give exact match on the manufacturing features and associated dimensions. This is why the next paragraph details a way to capitalize the knowledge from the CNC machine using G&M-Code.

4 Knowledge capitalization

4.1 G-Code to STEP-NC translation

Generally, in a G-code program there is no information about cutting tools, the rawpiece' shape and dimensions, its location, and the CNC controller type. This is first point to supplement for ensuring successful recognition. Then by analyzing hints (such as tool changes, speed changes, machining regions) in the G-code, workingsteps can be generated, in which all operations are treated as freeform operations and features as toolpaths, except those (like canned cycles) that can be easily attached to operations. Finally, machining features are extracted from the toolpath data by analyzing machining regions, machining strategies, etc., and the extracted feature data are organized by the ISO 14649.

An interpreter was developed for the toolpath generation [22]. The interpreter emulates the execution of the given G-code one block by one block: if it meets a “G0” command, a “rapid movement” entity is created; if it meets one or several consecutive “G1”s or “G2|3”s, a “machining workingstep” which includes a freeform operation is created. The parameters of these commands are used as cutter location data stored in the toolpath list of the freeform operation.

Then the extraction of manufacturing (machining) features from toolpaths and tool’s geometry is done. In this phase, one freeform operation corresponds to one machining workingstep. Many freeform operations might correspond to one manufacturing feature because often there are several layers of a rough machining and a finish machining, which are required to make a final feature. Hence one major procedure is to merge those freeform operations that machine the same feature, as well as the relevant workingsteps and rapid movements.

Some features can be easily extracted by the tool used. For example, if the tool type is for drilling, the feature is a hole; if the tool is a facemill, the feature is a planar face. If the tool is an endmill**,** which is a general and complex case, then analyse the x, y, z-values of the CL (cutter location) data in the toolpath list. If z-value varies and x, y-values keep constant, it mills a hole feature. If x, y, z-values all vary, it is a freeform milling operation for making a region of surface. If z-value keeps constant and x, y-values vary, it is a 2½D milling operation. The main grounds to find the remaining features (which can be planar faces, general profiles, closed pockets and open pockets in STEP-NC.) in a 2½D milling operation are the cutting area (the tool's covering region for cutting movements) and the milling strategy, which are computed based on toolpath's CL data.

4.2 Closed-Loop Manufacturing

This step of the knowledge capitalization process consists in filling the knowledge database by using the data extracted from the CNC machines converted in STEP-NC. Capitalization is based on STEP-NC manufacturing features and on Closed-Loop Manufacturing process which aims at reasoning on cases [23]. First designed for enabling feedback from CNC machines to CAM systems as showed in Figure 6 (1st track), the Closed-Loop Manufacturing process has been adapted for providing guidelines to the NC Simulation step (2nd track).

The capitalization is based on manufacturing features. This is why the cutting program initially in G-code language has to be translated into STEP-NC. The process proposes a case-based approach to determine if the guidelines are yet capitalized in the database.

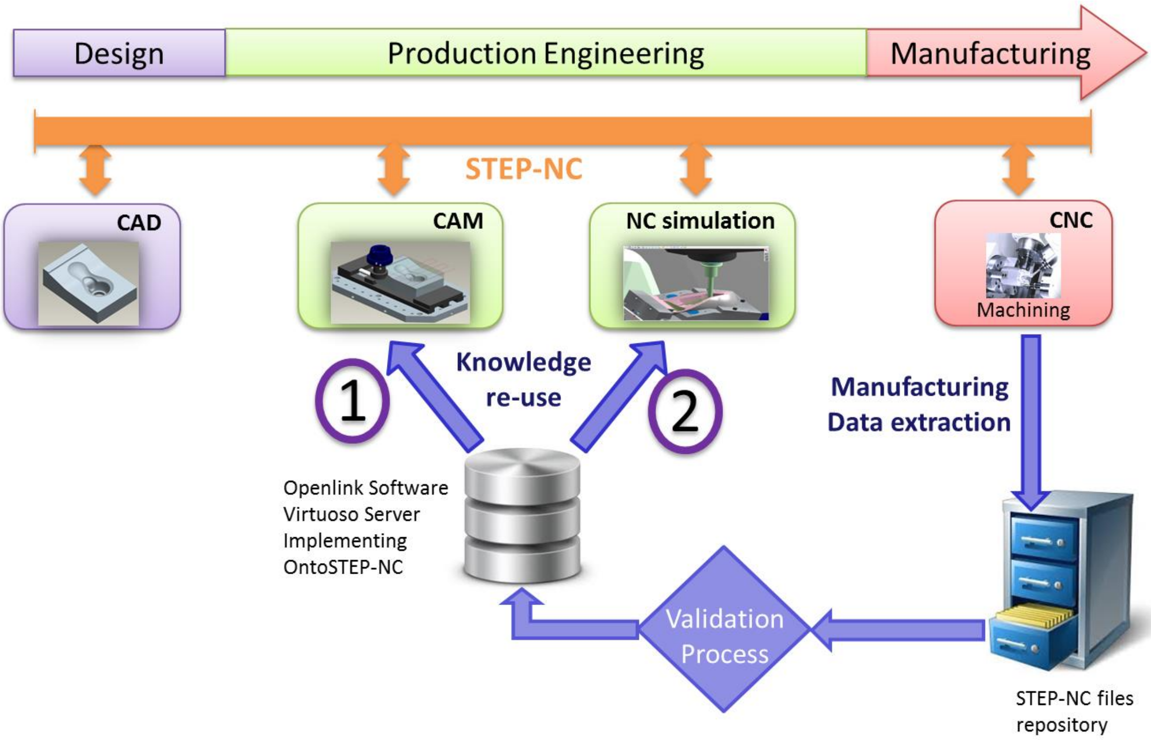
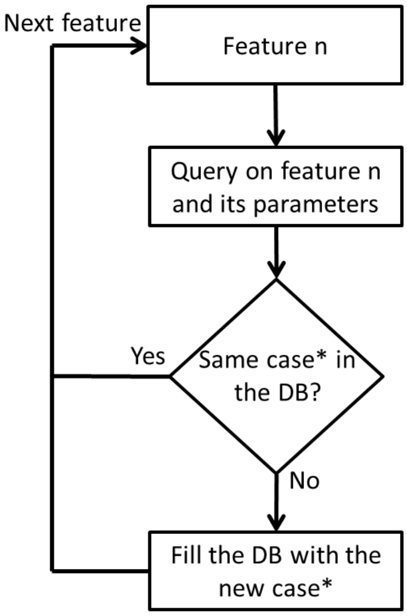


Figure 6: Closed-Loop Manufacturing process

The comparison is made on each case which groups the manufacturing feature and its entire criterion as machine type, material, cutting tool, etc. If the case does not exist in the knowledge database, then the manufacturing feature is stored in OntoSTEP-NC to become new guidelines for future CNC programs. In Figure 7 is detailed the database filling process.



*Case=feature+parameters

Figure 7: Filling database process

The last 2 parts has presented a way to enable feedback from CNC machines to NC simulation and providing help to verify cutting parameters in NC simulation. The next part exposes a case study in order to demonstrate the proposal presented previously.

5 Case study

Figure 8 presents the chosen part which allows testing the proposal. This part is the ISO 14649 STEP-NC reference part which provides 2 independent manufacturing features – a hole and a closed pocket – and a surfacing operation. In this study we are focusing on the two defined manufacturing features:

- The hole diameter is 20mm and has through end conditions (depth=50mm).
- The closed pocket boundary is a rectangle with planar bottom condition. The pocket's dimensions are 80*50*30mm and the orthogonal radius is R10.

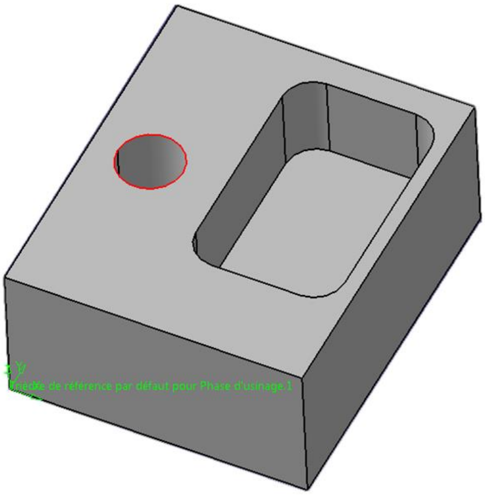


Figure 8: Testing part

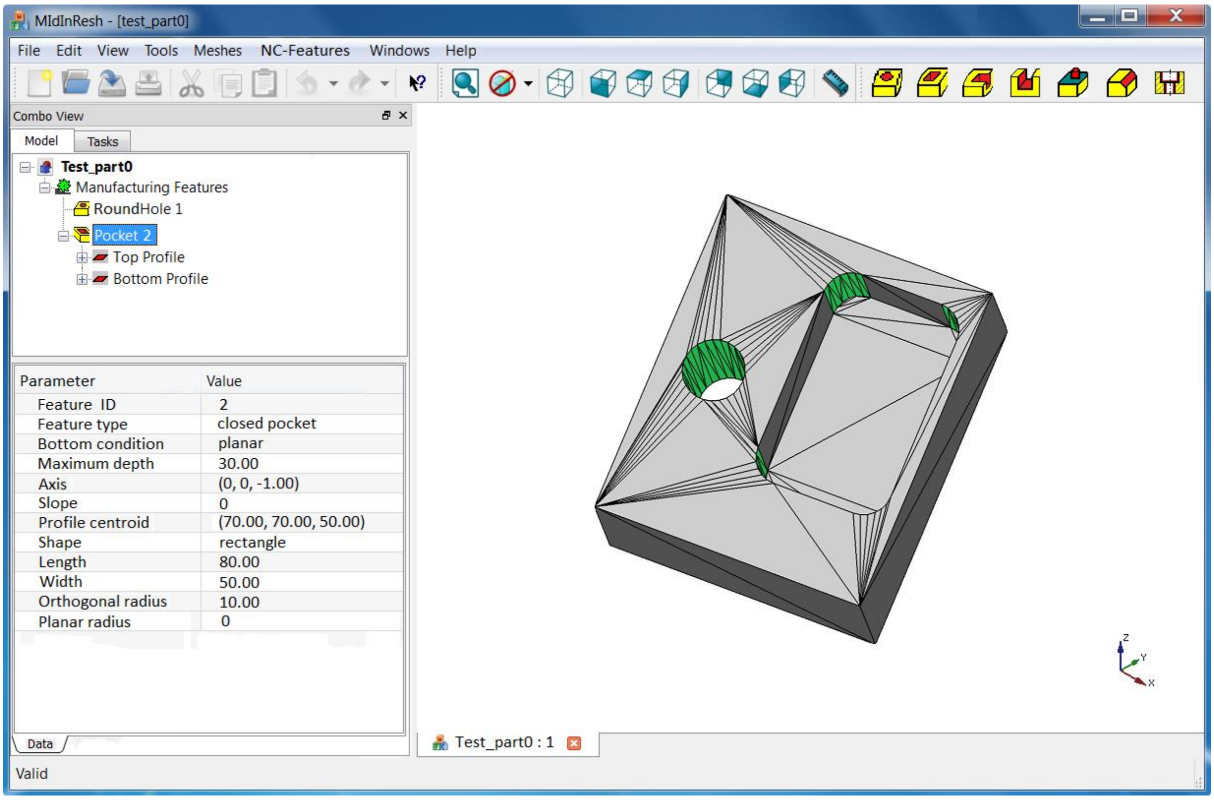
To test the proposal with this part, the scenario respects the following hypothesis. The CAD model and the CAM model has been done previously and those steps are not taken into account in the tests.

The test is performed in the following order:

- While the NC simulation is running, the simulation results are extracted and can be then re-used for starting the feature recognition. The feature recognition gives then a table in which all the manufacturing features are listed with their parameters.
- Based on this table, a query is launched by using the previous extraction on the knowledge database to find the exact or the closest solutions which have the same features but with different dimensions. This provides guidelines to the NC simulation step which can be followed or not if it doesn't appear as the best choices for machining the part.
- Once the program has been run in the machine, the G-Code program is extracted from the CNC machine and translated into STEP-NC program which leads the capitalization cutting knowledge.

5.1 Feature recognition and knowledge re-use in NC simulation

The IPM of the test part and the software interface is shown in Figure 9. The function for the feature recognition is under menu "NC-Features". For this simple IPM of NC simulation, the software has recognized 2 features —1 round hole and 1 rectangular pocket. The extracted parameters of these features are listed in Table 1.



Parameter	Value
Feature ID	2
Feature type	closed pocket
Bottom condition	planar
Maximum depth	30.00
Axis	(0, 0, -1.00)
Slope	0
Profile centroid	(70.00, 70.00, 50.00)
Shape	rectangle
Length	80.00
Width	50.00
Orthogonal radius	10.00
Planar radius	0

Figure 9: Screenshot of the software for feature recognition from IPM of NC simulation

Table 1: Recognized features of the test part

Feature name	Feature type	Parameters	Values
RoundHole 1	round hole	axis	(0, 0, 1.)
		axis position	(20., 60., 25.)
		diameter	22.
		depth	50.
		bottom condition	thru
Pocket 2	closed pocket	axis	(0, 0, -1.)
		profile centroid	(70., 70., 50.)
		profile shape	rectangle
		length	80.
		width	50.
		orthogonal_radius (minimum corner radius)	10.
		maximum depth	30.
		bottom condition	planar

Based on the previous results of recognized feature, SPARQL queries can be launched directly on the implementation of OntoSTEP-NC into Virtuoso Server. The query answer can be compiled as a list of parameters corresponding to each instance find in the database.

According the cases, 4 different scenarios can be found for the integration of data into the digital chain from the knowledge database. These 4 scenarios are detailed below:

- Scenario 1: The NC simulation has detected no collision and no problem with the current CNC program. Moreover the same case is yet stored in the database and provides the same parameters. Hence, the CNC program is validated and can be run on the CNC machine.
- Scenario 2: Similarly to scenario 1, the NC simulation tells that the program can be run. Even though in this scenario, no similar case has been yet stored into OntoSTEP-NC and some approximate (parameters, dimensions, tools…) cases exists, the process can be run after a last verification by analyzing the other solutions stored in the database.
- Scenario 3: The sequences planned do not pass the NC simulation step and the program sequences must be modified. In the database, a similar case has yet been stored and can be re-used by modifying directly the sequences on the NC simulation software. In this scenario, the knowledge stored in the database allows improving the CNC program in order it become runnable on the CNC machine.
- Scenario 4: Similarly to scenario 3, the simulation step has been aborted due to problems (collision, speed, dimensions…). Moreover there is no similar case in the database therefore a loop from NC simulation to CAM system is mandatory. As we have yet capitalized approximates cases in the knowledge database, guidelines and help is provided to the CAM programmer thanks to Closed-Loop Manufacturing (Track 1 in Figure 6). Then new machining sequences can be written and the program is then run again on NC simulation. If it passes NC simulation the program can be then run on the CNC machine, if not, a new loop between simulation and CAM has to be performed.

5.2 Knowledge capitalization on STEP-NC program

Once the program has been run on the machine and if the part is quality accepted, the CNC program can be considered as a reference for capitalization. As explained in section 4, the capitalization is feature oriented and it is mandatory the program be transformed in STEP-NC for filling the knowledge database.

The G-code shown in Figure 10 machines the test part presented in Figure 8. Three cutting tools (a drill, a reamer and an endmill) are used. By the tool types and the canned cycles in the code, we can extract a hole feature. By the cutting area and the milling strategy, we can identify a pocket feature.

```
G54 G90 G21 G40 G49 M5 M9
G0 Z100. (Move to the secure plane)
(To drill and ream a thru hole)
T2 M6 (Use a spiral drill, diameter 20mm)
G43 H2 (Length compensation by 70mm)
M8 M3 F900. S720
G0 Z30.
G90 G99 G81 X20. Y60. Z-18. R10.
G99 G81 X20. Y60. Z-36. R10. F1800.
G99 G81 X20. Y60. Z-60. R10. F1350.
G1 Z10. F1800.
G80 G49 M5 M9 (end of drilling cycle)
T3 M6 (Use a reamer, diameter 22mm)
G43 H3 (Length compensation by 50mm)
M8 M3 S1080
G90 G99 G85 X20. Y60. Z-60. R10.
G80 G49 M5 M9 (End of reaming cycle)

(To cut a pocket, rough & finish)
T1 M6 (Use an endmill, diameter 18mm)
G43 H1 (Length compensation by 50mm)
M8 S1200 M3 F2400.
G0 Z30.
X64.754 Y50.069
Z15.
(To rough pocket in 5 layers, 5.9/layer)
(First 2 blocks: to run helical approach)
G2 X77.2 Y55. Z-5.9 I5.246 J4.932
G2 X70. Y55. I-3.60 J0.
G1 Y90.
X75.
Y50.
X65.
Y90.
X70.
Y95.
X80.
Y45.
X60.
Y95.
X70.
Y100.
X85.
Y40.
X55.
Y100.
X70.
Z0.
G0 X69.532 Y47.815 (End of 1st layer)
……… (Code of next 4 layers omitted)

(To finish pocket in 6 layers. 5mm/layer)
(Bottom allowance 0.5,side allowance 1)
G0 Z30.
X74.890 Y60.285
Z15.
(First 2 blocks: to run helical approach)
G2 X77.2 Y55. Z-2. I-4.891 J-5.285
G2 X70. Y55. I-3.60 J0.
G1 Y93.
X78.
Y47.
X62.
Y93.
X70.
Y101.
X85.
G2 X86. Y100. I0. J-1.
G1 Y40.
G2 X85. Y39. I-1. J0.
G1 X55.
G2 X54. Y40. I0. J1.
G1 Y100.
G2 X55. Y101. I1. J0.
G1 X70.
Z0. (End of 1st layer)
............(Code of rest of layers omitted)
G2 X55. Y101. I1. J0. (now Z-30.)
G1 X70.
Z15. (End of finishing)
M30
```

Figure 10: G-code program of test part

After the recognition from the G-code, we got two features: a hole and a pocket. Structuring the feature data by the STEP-NC standard, we can get the ISO14649 part reference file for the recognition result, as shown in Figure 11.

The knowledge capitalization is directly made by analyzing the STEP-NC program and checks the identified features one by one; first querying the database on "manufacturing_feature" and then on "machining_operation". In this case, the manufacturing feature ROUND_HOLE calls two different machining operation DRILLING and REAMING. Both machining operation are mandatory to machine this hole. Similarly for each machining operation, the parameters has to be check crossed in order to compare with accuracy the cases in order the feedback guidelines be relevant.

As a consequence of this comparison before capitalization, the scenario 1 and scenario 3 are found in the database as existing cases while scenario 2 and scenario 4 are absent. To capitalize these new good practices, the cases must be entered in the knowledge database with all their entire parameters. By focusing here in the hole feature, the following information is capitalized: machining operation, associated tool, bottom condition, strategy, diameter, tolerance, security plane, spindle speed, approach and retract, direction and axis placement. All this information will be then re-used when a similar case will occur.

```
………… (Header section omitted)
DATA;
#1= PROJECT ('EXECUTE EXAMPLE1',#2,(#3),$,$,$);
#2= WORKPLAN ('MAIN WORKPLAN',(#11,#12,#13,#14),$,#4,$);
#3= WORKPIECE ('BLOCK WORKPIECE',$,0.01,$,$,$,(#91,#92,#93,#94));
#4= SETUP ('MAIN SETUP',#62,#60,(#5));
#5= WORKPIECE_SETUP (#3,#63,$,$,());
#11= MACHINING_WORKINGSTEP ('WS DRILL HOLE1',#60,#21,#32,$);
#12= MACHINING_WORKINGSTEP ('WS REAM HOLE1',#60,#21,#33,$);
#13= MACHINING_WORKINGSTEP ('WS ROUGH POCKET1',#60,#22,#34,$);
#14= MACHINING_WORKINGSTEP ('WS FINISH POCKET1',#60,#22,#35,$);
#21= ROUND_HOLE ('HOLE1 D=22MM',#3,(#32,#33),#67,#70,#111,$,#25);
#22= CLOSED_POCKET ('POCKET1',#3,(#34,#35),#69,#71,(),$,#26,$,#112,#27);
#25= THROUGH_BOTTOM_CONDITION ();
#26= PLANAR_POCKET_BOTTOM_CONDITION ();
#27= RECTANGULAR_CLOSED_PROFILE ($,#113,#114);
#32= DRILLING ($,$,'DRILL HOLE1',15.,$,#44,#54,#51,$,$,$,$,$,#55);
#33= REAMING ($,$,'REAM HOLE1',15.,$,#47,#54,#51,$,$,$,$,$,#56,.T.,$,$);
#34= BOTTOM_AND_SIDE_ROUGH_MILLING ($,$,'ROUGH POCKET1',15.,$, #40,#57,#51,$,$,$,#58,6.5,5.,1.,0.5);
#35= BOTTOM_AND_SIDE_FINISH_MILLING ($,$,'FINISH POCKET1',15.,$,#40,#57,#51,$,$,$,#59,2.,10.,$,$);
#40= MILLING_CUTTING_TOOL ('ENDMILL_18MM',#41,(#43),80.,$,$);
#41= TAPERED_ENDMILL (#42,4,.RIGHT.,.F.,$,$);
#42= MILLING_TOOL_DIMENSION (18.,$,$,29.,0,$,$);
#43= CUTTING_COMPONENT (100.,$,$,$,$);
#44= MILLING_CUTTING_TOOL ('SPIRAL_DRILL_20MM',#45,(#43),90.,$,$);
#45= TWIST_DRILL (#46,2,.RIGHT.,.F.,0.84);
#46= MILLING_TOOL_DIMENSION (
#47= MILLING_CUTTING_TOOL ('REAMER_22MM',#48,(#43),100.,$,$);
#48= TAPERED_REAMER (#49,6,.RIGHT.,.F.,$,$);
#49= MILLING_TOOL_DIMENSION (22.,$,$,40.,$,$,$);
#51= MILLING_MACHINE_FUNCTIONS (.T.,$,$,.F.,$,(),.T.,$,$,());
#54= MILLING_TECHNOLOGY (0.03,.TCP.,$,-18.,$,.F.,.F.,.F.,$);
#55= DRILLING_TYPE_STRATEGY (0.75,0.5,2.,0.5,0.75,8.);
#56= DRILLING_TYPE_STRATEGY ($,$,$,$,$,$);
#57= MILLING_TECHNOLOGY (0.04,.TCP.,$,-20.,$,.F.,.F.,.F.,$);
#58= CONTOUR_PARALLEL ($,$,.CW.,.CONVENTIONAL.);
#59= CONTOUR_PARALLEL (0.05,.T.,.CW.,.CONVENTIONAL.);
#60= PLANE ('SECURITY PLANE',#61);
#61= AXIS2_PLACEMENT_3D ('PLANE1',#90,#81,#82);
#62= AXIS2_PLACEMENT_3D ('SETUP1',#80,#81,#82);
#63= AXIS2_PLACEMENT_3D ('BLOCK WORKPIECE',#80,#81,#82);
#67= AXIS2_PLACEMENT_3D ('HOLE1',#97,#81,#82);
#68= AXIS2_PLACEMENT_3D ('DEPTH PLANE',#98,#81,#82);
#69= AXIS2_PLACEMENT_3D ('POCKET1',#99,#81,#83);
#70= PLANE('DEPTH SURFACE FOR ROUND HOLE1',#68);
#71= PLANE('DEPTH SURFACE FOR POCKET1',#68);
#80= CARTESIAN_POINT('ORIGIN',(0.00,0.00,0.00));
#81= DIRECTION('K-VECTOR',(0.00,0.00,1.00));
#82= DIRECTION('I-VECTOR',(1.00,0.00,0.00));
#83= DIRECTION('J-VECTOR',(0.00,1.00,0.00));
#90= CARTESIAN_POINT('SECURITY PLANE:LOCATION',(0.00,0.00,100.00));
#91= CARTESIAN_POINT('CLAMPING_P1',(0.00,20.00,25.00));
#92= CARTESIAN_POINT('CLAMPING_P2',(100.00,20.00,25.00));
#93= CARTESIAN_POINT('CLAMPING_P3',(0.00,100.00,25.00));
#94= CARTESIAN_POINT('CLAMPING_P4',(100.00,100.00,25.00));
#97= CARTESIAN_POINT('HOLE1:LOCATION',(20.00,60.00,0.00));
#98= CARTESIAN_POINT('FEATURE DEPTH',(0.00,0.00,-30.00));
#99= CARTESIAN_POINT('POCKET1:LOCATION ',(70.00, 70.00, 0.00));
#111= TOLERANCED_LENGTH_MEASURE(22.00,#115);
#112= TOLERANCED_LENGTH_MEASURE(10.00,#116);
#113= TOLERANCED_LENGTH_MEASURE(50.00,#116);
#114= TOLERANCED_LENGTH_MEASURE(80.00,#116);
#115= PLUS_MINUS_VALUE(0.30,0.30,3);
#116= PLUS_MINUS_VALUE(0.10,0.10,3);
ENDSEC:
```

Figure 11: STEP-NC program of test part

6 Discussion and conclusion

This paper presents a part of the results of the ANGEL project concerning the evolution of the digital chain. It details in one hand the direct direction by exploring the technical solution to improve the data exchange from CAD systems to CNC machines and in the other hand it details the capitalization process which allows having feedback to the CAM system and the NC simulation.

As the STEP-NC is not yet implemented, the bi-directional data exchange from CAD systems to CNC machines is not totally achieved. Moreover, as the STEP-NC digital chain cannot be implemented this paper has proposed to convert when necessary the existing formats and language into STEP-NC which is more reliable for working on feature oriented structure. This is why it has been first proposed to perform feature recognition directly from the NC simulation in order guidelines could be provide for programmers. Secondly to capitalize the knowledge it has been chosen to convert the G-Code extracted from the CNC machine into STEP-NC. This feature oriented structure allows capitalizing the machining knowledge by using case-based verification and validation. Even though this proposal allows having feedback from CNC machine to CAM and NC simulation this would not be optimal until STEP-NC is implemented.

OntoSTEP-NC provides a way to capitalize and to re-use machining knowledge but one limit of the Closed-Loop Manufacturing consist of filling the knowledge database. In fact to send back relevant information it is necessary that cases yet processed be referenced as good practices. Although the Closed-Loop Manufacturing is a promising solution to help programmer to make better decisions, when implementing the knowledge database there is no cases stored. Two ways can be explored: to wait that the database be filling automatically or to explore the old cases and to fill manually the knowledge database.

The proposal presented in this paper provides a way to capitalize expertise and manufacturing knowledge, this work could be completed by taking into account all the freeform features. In fact, this proposal only focuses on manufacturing features which are defined as "two5D_manufacturing_feature" but not all the freeform operations as spleen, surface projection, etc. defined by "region". This work is going to be carried out in other works and will be enriched with data mining for extracting more parameters from the behavior of CNC machines.

7 References

[1] S.-H. Lee and Y.-S. Jeong, "A system integration framework through development of ISO 10303-based product model for steel bridges," *Autom. Constr.*, vol. 15, no. 2, pp. 212–228, 2006.

[2] S. Rachuri, E. Subrahmanian, A. Bouras, S. J. Fenves, S. Foufou, and R. D. Sriram, "Information sharing and exchange in the context of product lifecycle management: Role of standards," *Comput. Des.*, vol. 40, no. 7, pp. 789–800, 2008.

[3] R. Laguionie, M. Rauch, and J. Hascoet, "Toolpaths programming in an intelligent STEP-NC manufacturing context," *J. Mach. Eng.*, vol. 8, no. 1, pp. 33–43, 2008.

[4] X. W. Xu, "Realization of STEP-NC enabled machining," *Robot. Comput. Integr. Manuf.*, vol. 22, no. 2, pp. 144–153, 2006.

[5] T. Kramer and X. Xu, "STEP in a nutshell," in *Advanced Design and Manufacturing Based on STEP*, pp. 1–22, 2009. (doi: 10.1007/978-1-84882-739-4_1)

[6] T. Dipper, X. Xu, and P. Klemm, "Defining, recognizing and representing feature interactions in a feature-based data model," *Robot. Comput. Integr. Manuf.*, vol. 27, no. 1, pp. 101–114, 2011.

[7] X. W. Xu, H. Wang, J. Mao, S. T. Newman, T. R. Kramer, F. M. Proctor, and J. L. Michaloski, "STEP-compliant NC research: the search for intelligent CAD/CAPP/CAM/CNC integration," *Int. J. Prod. Res.*, vol. 43, no. 17, pp. 3703–3743, 2005.

[8] S. T. Newman, A. Nassehi, X. W. Xu, R. S. U. Rosso, L. Wang, Y. Yusof, L. Ali, R. Liu, L. Y. Zheng, S. Kumar, P. Vichare, and V. Dhokia, "Strategic advantages of interoperability for global manufacturing using CNC technology," *Robot. Comput. Integr. Manuf.*, vol. 24, no. 6, pp. 699–708, 2008.

[9] X. W. Xu, L. Wang, and Y. Rong, "STEP-NC and function blocks for interoperable manufacturing," *IEEE Trans. Autom. Sci. Eng.*, vol. 3, no. 3, pp. 297–308, 2006.

[10] F. Ridwan and X. Xu, "Advanced CNC system with in-process feed-rate optimisation," *Robot. Comput. Integr. Manuf.*, vol. 29, no. 3, pp. 12–20, 2013.

[11] A. Nassehi, S. T. Newman, and R. D. Allen, "The application of multi-agent systems for STEP-NC computer aided process planning of prismatic components," *Int. J. Mach. Tools Manuf.*, vol. 46, no. 5, pp. 559–574, 2006.

[12] S. Borgia, A. Matta, and T. Tolio, "STEP-NC compliant approach for setup planning problem on multiple fixture pallets," *J. Manuf. Syst.*, vol. 32, no. 4, pp. 781–791, 2013.

[13] F. Zhao, X. Xu, and S. Xie, “STEP-NC enabled on-line inspection in support of closed-loop machining,” *Robot. Comput. Integr. Manuf.*, vol. 24, no. 2, pp. 200–216, 2008.

[14] X. Zhang, A. Nassehi, M. Safaieh, and S. T. Newman, “Process comprehension for shopfloor manufacturing knowledge reuse.,” *Int. J. Prod. Res.*, vol. 51, no. 23/24, pp. 7405–7419, 2013.

[15] J. J. Koenderink and A. J. van Doorn, “Surface shape and curvature scales,” *Image Vis. Comput.*, vol. 10, no. 8, pp. 557–564, 1992.

[16] H. Zhao, N. Anwer, and P. Bourdet, “Curvature-based registration and segmentation for multisensor coordinate metrology,” *Procedia CIRP*, vol. 10, pp. 112–118, 2013.

[17] S. Xu, N. Anwer, and C. Mehdi-Souzani, “Machining Feature Recognition from In-Process Model of NC Simulation,” *Comput. Aided. Des. Appl.*, vol. 12, no. 4, pp. 383–392, 2015.

[18] N. Anwer, “Méthodologie d’analyse de raisonnement pour la génération automatique des gammes d’usinage en fraisage,” PhD thesis from ENS de Cachan, 2000.

[19] “International Standards Organization, ISO 14649-1. Industrial automation systems and integration — Physical device control — Data model for computerized numerical controllers — part 1: Overview and fundamental principles.” 2003.

[20] C. Danjou, J. Le Duigou, and B. Eynard, “OntoSTEP-NC : une approche ontologique pour la capitalisation des connaissances en usinage,” in *14° Colloque National AIP-Primeca*, pp. 1–7, 29 March - 2 April 2015, La Plagne, France.

[21] “Virtuoso universal server.” [Online]. Available: http://virtuoso.openlinksw.com/. [Accessed: 15-Dec-2014].

[22] S. Xu, N. Anwer, and S. Lavernhe, “Conversion of G-code part programs into STEP-NC,” in *Proceedings of Joint Conference on Mechanical, Design Engineering & Advanced Manufacturing,*18-20 June 2014, Toulouse, France.

[23] C. Danjou, J. Le Duigou, and B. Eynard, “Closed-loop manufacturing process based on STEP-NC,” *Int. J. Interact. Des. Manuf.*, pp. 1–13, 2015. (doi: 10.1007/s12008-015-0268-1)