

13th CIRP Conference on Intelligent Computation in Manufacturing Engineering, CIRP ICME '19

## IC.IDO as a tool for displaying machining processes. The logic interface between Computer-Aided-Manufacturing and Virtual Reality

Michele Giannuzzi<sup>a,\*</sup>, Gabriele Papadia<sup>a,\*</sup>, Claudio Pascarelli<sup>a</sup>

<sup>a</sup>Department of Engineering for Innovation, University of Salento, via per Monteroni, 73010 Lecce, Italy

\* Corresponding author. E-mail address: [michele.giannuzzi@unisalento.it](mailto:michele.giannuzzi@unisalento.it), [gabriele.papadia@unisalento.it](mailto:gabriele.papadia@unisalento.it)

### Abstract

This scientific communication investigates the logic interface of a CAM solver, i.e., MasterCAM, into a Virtual Reality (VR) environment. This integration helps in displaying machining operations in virtual reality. Currently, to partially visualize the results of a simulation in an immersive environment, an import/export procedure must be done manually. Here, a software plugin integrated into IC.IDO (by ESI Group) has been realized and fully described. This application allows the complete integration of CAM solver into the VR environment. In particular, the VERICUT solver has been integrated into VR. This kind of integration has never been done yet.

© 2020 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Peer review under the responsibility of the scientific committee of the 13th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 17-19 July 2019, Gulf of Naples, Italy.

**Keywords:** CAM solver; NC file verification; Virtual Reality

### 1. Introduction

This study aims to integrate the Computer Aided Manufacturing (CAM) with the Virtual Reality (VR), i.e., being able to visualize, in the semi-immersive environment and 1:1 scale a machining process.

CAM relates to the use of software technologies by facilitating and automating the process plans used by the CNC machines. Process planning is known to be a crucial procedure in manufacturing and is considered the bridge between design and product manufacturing. Once the design of the CAD model is completed, it has to be imported into CAM software, where the machine-independent Cutter Location DATA (CLDATA) file will be created. Via post-processing, the CLDATA is converted into a Numerical Control (NC) Program that can be read by a Control Unit. They are referred to as the G-Code. The G-Code is standardized under DIN 66025 and ISO 6983. An example of CAM software which can do this work is Mastercam®.

Once obtained, NC Programs may be run on the NC Control Unit without prior testing. Frequently, however, these programs have considerable potential for optimization and may even

contain errors. Possible errors caused by the NC Control Unit are: reading a point outside the machine's axes and/or the potential collisions between the machine's axes. The NC file verification simulation is a critical task before real CNC machining. Usually, the software packages developed for testing NC file are desktop-based software. Simulation tool like this maps the workspace of the machine tool, permitting verification of speed, tool's movement path, and verifying the material removal at the workpiece. An example of software like this is Vericut®. To provide the verifying G-Codes functionality, Vericut® uses its libraries to decode the G-Code from the most established NC Control Units.

The target here is to integrate the technology of Virtual Reality (VR) with desktop-based software, such as Vericut®, to be used as a tool for NC file verification.

Among the Digital Transformation technologies, VR is one of the most promising [1] since it involves the transformation of business operations and affects products and processes. The use of VR brings benefits in different mechanical fields: design, maintenance, management, and manufacturing. Many studies describe the VR potential applications for manufacturing

applications, most of them for interfacing CAD authoring tools. Few studies explore the potentials of directly integrating VR environments with CAM-related simulations and information, and even fewer proved to solve real industrial problems [4]. In machining simulations, the virtual reality it is applied in the inspection methods and collision avoidance [2, 6, 8 10]. In studying the factors affecting the quality, and the time of the material removal process [9]. In coupling between a real NC control unit and a virtual machine and its potentials application for vocational training [7]. In the application of haptic virtual reality for machining [5]. Moreover, VR helps in the assistance of CNC machine tool demonstration and exhibition [3].

Hence, as seen, the VR technology visualizes 3D (stereoscopic) scale one-to-one models, e.g., studying and using different types of machine tool and their kinematics. So, it can be said that the Virtual Manufacturing paradigm is the integration of Virtual Reality and manufacturing technologies, i.e., the integration of virtual environment for enhancing decision-making and control in the manufacturing system. IC.IDO® is a software package for the Windows operating system that provides a Virtual Reality (VR) environment designed for the industrial environment. It allows users to interact in an immersive environment with mechanical components, assembled components, prototypes or entire work environments in a more realistic way.

In this paper, a plugin application for the NC programs verifications based on human-machine interaction is presented. The Virtual Machining Environment (VME) allows users to test the NC process planning and predict the processing quality.

This article will, first of all, present the research and explain the kind of CAM-VR integration intended and the solutions introduced, i.e., the interface logic thought and performed will be described. Following a simulation test case will be presented. The future potentialities, the limitations, and problems will be addressed. Finally, the conclusion will close the paper.

## 2. Methods

### 2.1. Scope

This research aims to create a software integration that, starting from the Mastercam® NC file and an initial IC.IDO® session (without animations), can automatically generate the animations giving a final and complete virtual reality session in IC.IDO®. The user will be able to navigate through a stereoscopic scene, where the machine tool moves according to the lines of the NC file produced by Mastercam®, and he will be able to carry out the verification checks. This tool, hence, is designed to provide greater user-friendliness and to allow more expert modeling and design to bring simulations to a higher level of solution. When it was born, VR technology was developed as a visualization tool. Now, virtual reality technology should be integrated into every section of the product lifecycle.

The scope, here, is to recreate the cutting operations of CNC machines in virtual reality, i.e., building a Virtual Machining

Environment (VME). Most of the programs which are used to simulated automatically generated G-codes are desktop-based. Within this Software-in-Loop simulation system, the workspace of the machine, including aspects such as speed, range of movement of the tool and the accuracy of the workpiece geometry is reproduced. Among the commercial software platforms, one of the most famous is Vericut® which, in this paper, is integrated into a Virtual Reality software IC.IDO®. In the case here presented, the virtual reality is considered as a semi-immersive simulation and interactive technology which require 3D eyewear and motion trackers. The University of Salento, in this project, has made available its virtual room whose features are listed below:

- Dimensions of approximately 21 x 6 meters, of which about 4.5 meters are dedicated to the control room and laboratory
- A stereoscopic projector in rear projection with ultra-short optics, to allow interaction between user and display without interference with the light beam.
- Two workstations for the generation of stereoscopic images
- Two-camera system (Smartrack) able to detect the position and orientation, thanks to markers, of the eyes and hand into space.
- Flystick, which in addition to containing markers, contains a series of buttons and an analog stick, with which the user can issue commands to the engine.
- Main room equipped with a larger screen, with front projection, able to display the images generated by the PCs installed in the control room/laboratory.

As a case study, a simple milling operation on a 3-axis machine has been selected, see Fig.1. It is significant to know that the G-code only drives the movements of the tool and nothing says about the required actions of the machine parts. For example, if the code tells that the tool must move from point A to point B, it does not say how this movement should be achieved, for example by moving the tool holder rather than the workpiece table, or by a combination of both actions. The interpretation is made by the machine itself, based on its construction characteristics. Hence, the aim of the was to reproduce, in a Virtual Environment, what Vericut® does in a bi-dimensional environment to improve the understanding of cutting operations and machine movements thanks to the user's coexistence in the same context of the simulated machine.

The standard VR data format is the Virtual Reality Modeling Language (VRML). After the exporting the VRML geometry, the model is entirely static, i.e., without kinematic and animation information. The VR machine tool has to be divided at least into a moving axis and the static part of the machine tool. The hierarchy of the VR machine tool must be structured analog to the real machine tool.

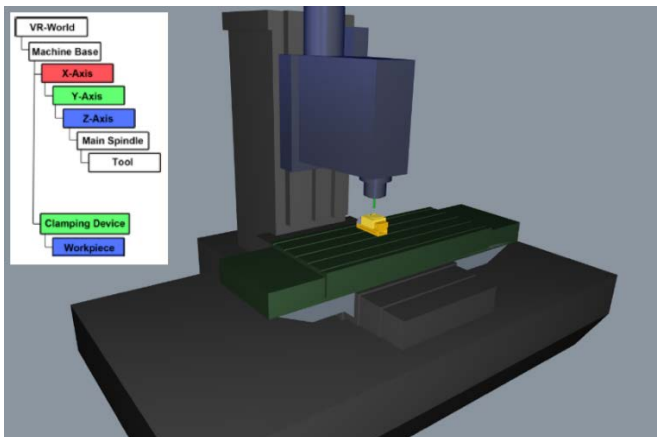


Fig. 1. A 3-axis milling machine shown in Vericut®.

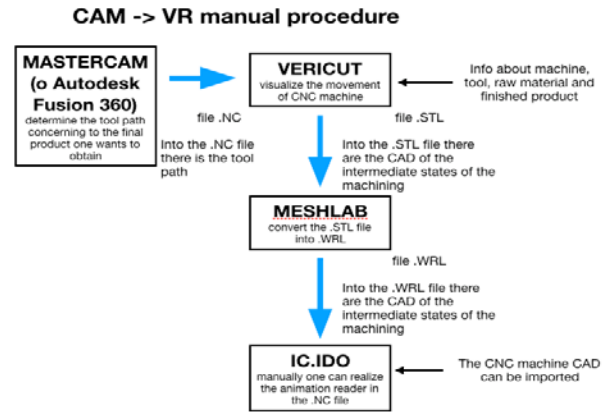


Fig. 3. Manual procedure for CAM-VR integration

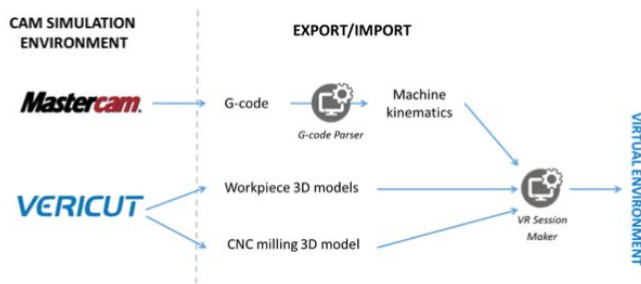


Fig. 2. Software chain for VR-CAM integration.

## 2.2. The VR-CAM integration

Two types of CAM solver were used to obtain the necessary elements, see Fig.2:

- Mastercam® that allows the design of manufacturing operations for a CNC machine and that gives the output G-code;
- Vericut® that has been used to obtain the CAD models of the workpiece to reconstruct its modification during machining.

In particular, Vericut® libraries were the starting point for modeling the 3-axis CNC milling. The resource library contains a digital model of various types of machine tools, fixtures, and cutting tools. Based on sufficient machining ability provided by the resource library, a multi-part process plan of different means can be carried out in the virtual machining environment in Vericut®. Mainly two categories of the machine tools are defined in the library currently, single-task machine and multi-task machine. The machine components are split into two-group: movable components and fixed components. The software itself provides advanced multitasking and complex structural machines from the famous corporation as MAZAK, DMG, MAKINO, WFL and so on.

Currently, there is no easy way to visualize a machining scene in a VR environment. IC.IDO® can import the geometry models of machine parts, tools, and workpieces, but it is not able to load and interpret automatically an NC file containing a

## CAM -> VR automatic procedure

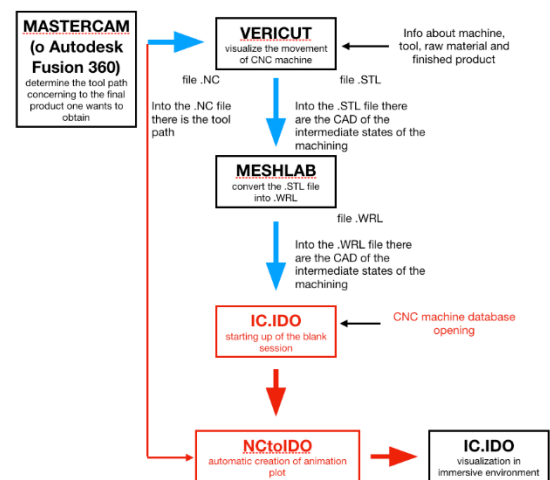


Fig. 4. Automatic procedure for CAM-VR integration.

G-Code program. Manually generating large number of animations necessary to replicate the machining scene in an immersive environment is time expensive, even for modest complexity. Generally, the steps required for the user are:

- Performing a Mastercam® simulation. The simulation generates an NC file which will subsequently be loaded into the Vericut® software together with a series of data concerning the machining (machine, tool, raw, finished),
- in Vericut® one exports (manually) the STL file of the steps of the workpiece (advancing through individual steps in the animation),
- the STL file must be converted to WRL format, for example through MeshLab software and imported into IC.IDO®,
- importing machine, tool and workpiece models,
- importing the models of the semi-finished products, which represent the progressive processing status of the piece,
- set the correct coordinate system,
- set the correct units of measurement,
- analyze the G-Code program line by line. If the current line contains a move or cuts instruction, create an animation that faithfully replicates the movement of the machine. If a cut to the workpiece happens the replacement of the old

geometry, currently shown, with the new configuration must be processed.

In Fig.3 is shown a diagram representing the process just described. This process is too long even if a few lines of the NC file are considered since it involves a burdensome and complex user-work to customize IC.IDO. Hence, to realize the visualization in an immersive environment of a CAM simulation, especially if of high complexity, it is necessary software that takes care of automating these operations.

The solution introduced is summarized in Fig.4. With this new procedure, the starting point is still the generation of the NC program from Mastercam®, i.e., the file with the process plan information to get the final product. All the models (the machine tool, the tools selected and the workpiece - the initial, the semi-finished and finished product) will be imported from the database (Vericut®, MeshLab). Afterward the IC.IDO will start. At the same time, the tool/parser, called NCtoIDO/ToolEDS, acquires the NC file and creates the scene, the animations of the process and the various inter steps in XML format, in such a way as to respect the timeline, according to the instructions received. The XML file is imported into IC.IDO and quickly it will link all VRML models within the scene. Thus the animations panel will show the entire processing from the raw geometry to the finished product.

Hence, the solution proposed was to create, from scratch, a parser, i.e., the NCtoIDO/ToolEDS tool, described in the next section, that translate the general instruction of the G-code in specific commands for the specific machine. The same tool automatically creates and “assembles” all the necessary elements for a complete VR session in IC.IDO

### 2.3. NCtoIDO: a NC-VR coupling strategy

The purpose of the NCtoIDO tool is to read and analyze a program in G-Code, written manually or auto-generated by a CAD / CAM software like Mastercam, and to insert in an IC.IDO session the necessary information useful to visualize in the immersive environment the desired workpiece processing plan.

The tool was developed in C++ language creating both a standard interface compatible with Windows desktop environment and it is a highly optimized executable that allows processing G-Code programs very quickly even if they are too complicated. The integrated development environment (IDE) chosen is Microsoft Visual Studio (VS).

When creating a tool that acts as an interpreter for a G-Code program to an IC.IDO virtual reality session, a fundamental concept must take into account: a G-Code does not contain any useful information on the structure and how the CNC operates. For example, the G-Code instructions only describe a relative displacement between the machine (and its tool) and the parts to be machined, but they say nothing about how the machine should perform that action.

Based on the above, it is clear that there is no univocal way to “translate” a G-Code program into a file readable by IC.IDO.

The interpreter to the machine specifications can be adopted only after the machine itself has been chosen. The machine to

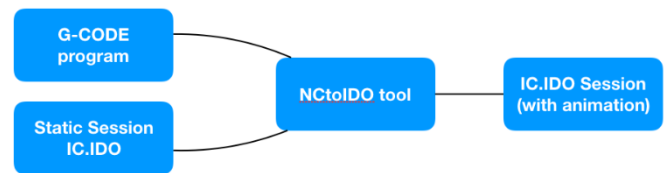


Fig. 5. Automatic procedure for CAM-VR integration.

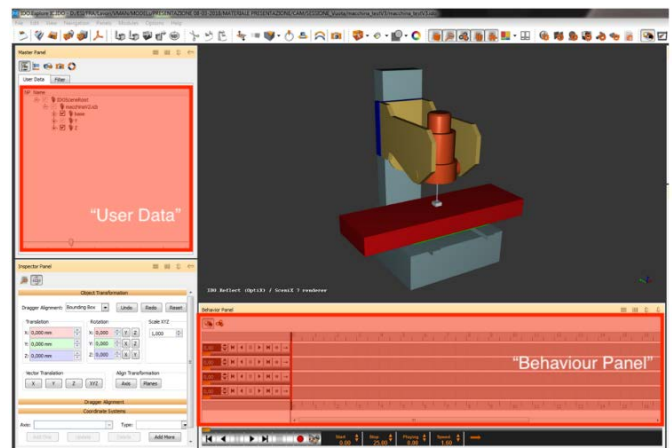


Fig. 6. IC.IDO work session.

be simulated in IC.IDO is chosen from a library of available “starting” work sessions, prepared suitably to be used with the NCtoIDO tool. As from the “User Data” panel of IC.IDO (see Fig. 5) the machine part models are grouped according to a precise hierarchy. The X-axis moving parts go into the group called “base”, while the parts that move along Y and Z must be inserted into the homonymous groups. Note, also, in the “Behavior Panel” there is a timeline with empty tracks, they will later be populated with the necessary animations.

When it starts up, the tool reads the NC file provided as input line by line. Each line of the G-Code is preprocessed in one or more CMC instructions. The Canonical Machining Commands (CMC) allow describing the operation of any numerical control machine from 3 to 6 axes. Each command is atomic, i.e., it produces a single movement or action; therefore, a line of the NC file can be interpreted as one or more CMC commands.

The preprocessing phase of the instructions in CMC has two advantages:

- It allows having an intermediate standard interpretation layer. Which means that if one chooses a machine with more than three axes, only the part that translates the G-Code into CMC must be modified.
- It allows a clear and simplified view of the processing to be included in the IC.IDO session.

At the end of the parsing of the NC file, the NCtoIDO tool has in memory an array with all the movements to be transformed into animations in the IDO session. In the end, hence, an XML must be created.



```

<FileHeader name="FileHeader">
  <String ICIDO_GmbH_Session_File_Version="11.2.0"/>
  <IDFile ModelFile="macchina_testV3.icb"/>
  <IDFile StatesFile="state.xml"/>
  <IDFile ReferenceTreeFile="referenceTree.xml"/>
  <Boolean HasExternalReferences="false"/>
  <Boolean SavedExternalReferences="true"/>
  <Boolean CombinerActive="false"/>
  <String Metric="m"/>
  <String InitialState="state:id/4099"/>
  <String RecentState="state:id/4099"/>
</FileHeader>

```

Fig. 7. "FileHeader" block of the IDO file.

Microsoft Visual C++ 2017 is not equipped with a standard with a class capable of manipulating a XML files: it is then necessary to include in the NCtoIDO code the TinyXml-2 external library, which is popular open-source software. Since the IDO file is an XML, the choice to use already widely tested free software has guaranteed reduced implementation times and it is in line with expectations in terms of stability of software performance. TinyXml-2 opens the IDO file for reading and writing and verifies that the content is a well-formed XML; otherwise, it returns an error to the screen. In this way, the software can search and find the elements that it is interested in modifying, verifying the hierarchical level, the tag and the presence of attributes with specific values.

The IDO file is an XML file composed by two blocks, the "FileHeader" block and the "State Container" block, which is composed of 14 State containers. The "FileHeader" block, see Fig. 7, is composed of lines that refer to the version of IC.IDO installed on the machine, to the 3D model of the machine tool and to the STATE ID (a code that distinguishes the session). The block of the "State Container", is composed of 14 State Containers. The ones needed in this case are the State 0 / State 7 / State 11. Each of them has a fundamental task:

- STATE CONTAINER 0: Contains the animation declaration. For each animation frame at least "TreeNode" of movement is created, one for each axis on which the position is changed: each movement is therefore decomposed on the axes.
- STATE CONTAINER 7: when the animation is inserted inside the "behavior panel" (the timeline of IC.IDO) the SC 7 will create the "Track" block, namely all the clips inserted into that animation track. In this section, NCtoIDO

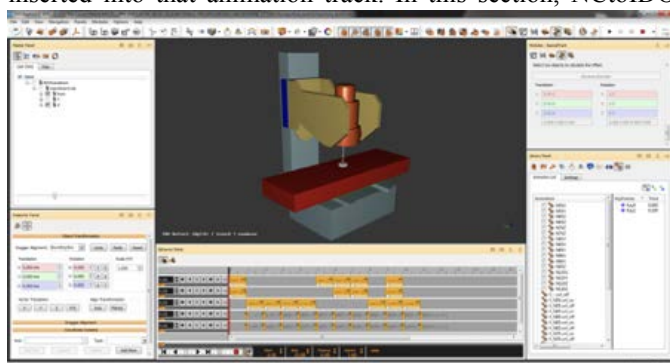


Fig. 8. IDO session completed.

```

N10 G90 G94 G17 G49 G40 G80
N15 G21
N20 G28 G91 Z0
N25 G90

(20 COUNTOUR1)
N30 T1 M06
N35 S1070 M03
N40 G54
N45 M08
N55 G00 X50. Y35.
N60 G43 Z15. H01
N65 G00 Z5.
N70 G01 Z1. F30
N75 Z-5.
N80 Z10. F214.
N85 G03 X9.988 Y-35. J.35.
N90 G01 X50.002 Y-35.013
N95 G00 Z15

N105 M09
N110 G28 G91 Z0.
N115 G49
N120 G28 X0. Y0.
N125 M30
%
```

Fig. 9. Part Program.

reconstructs the processing timeline based on the "Animation" tags already created and the information contained in the "TreeNodes" related to them.

- STATE CONTAINER 11: Here all the references for the user data (the node tree in the session) will be found. These codes will be indispensable to make the animation be linked to the right 3D model.

Once the processing is complete, NCtoIDO calls the TinyXml-2 library again to save the IDO file modified in the path specified by the user. The IDO session is created, see Fig. 8

## 2.4. CAM Simulation

The CAM-VR integration aforementioned is an application to check the tool path obtained by MASTERCAM, to investigate collision problems and optimize the Part Program.

A computerized experiment was performed. A simple milling process has been designed and simulated: 3-axis machining in which the tool performs a circular interpolation movement was chosen, see Fig. 9. The processing consists of a "U" of width equal to that of the cutter (10mm) and depth 5mm. The generated Part Program is visible in Fig.8. The central part is the one where the processing movements are.

A 3-axis milling machine was chosen (basic\_3ax\_vmill.mch). The Part Program and the tool libraries were created accordingly. The relative control file is the "fan0m.ctl". Both the MCH and CTL files are present in the Vericut library. This folder also contains the CAD files of the machine, in various formats (.ply, .swp, .stl).

Within VERICUT software, the components that are present in the tree under the heading "Base" are the components that always remain fixed during machining. While the ones in "X", "Y" and "Z" axes are the components that have the movements along their respective axes. When Vericut reads a command line of the Part Program that commands a movement of, for example, 20 mm along Y, all the components of the machine placed under "Y" will move 20 mm along the Y-axis. Under

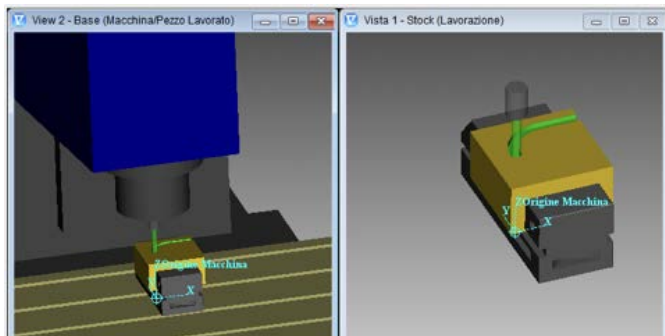


Fig. 10. (Left) BASE view; (Right) STOCK view.

the heading "Fixture", on the other hand, there are the CAD models relating to the fixture/grips.

In Vericut, generally, there are two different views (see Fig.10). In the "Stock" view (right), only the workpiece, the gripping equipment and the tool are visible. In this view, the piece is always stationary and the movement is always possessed only by the tool. Therefore, the actual movement could be different, since generally some movements are possessed by the tool and others by the piece. The workpiece table moves along X and Y and the tool along Z. In fact, in the "Base" view (left) one sees the real movement of piece and tool.

The workpiece table can be seen moving along X and Y, while the movement along Z is given by the tool holder. The view "Base" is the one shown in the Virtual Reality Environment.

### 3. Results and Discussion

Concerning the material removal simulation, only the representation scene is updated in real time, while the workpiece geometry is updated later from the material removal sequence. Better said, the swept volume is not fast rendered, but the material removal is subtracted from the original volume on completion of the cut. Indeed, the real-time removal technique is only possible when using a volume-based model (e.g., CAD models). Since the VR models are based on the VRML format (the models are based solely on their surface) no material removal is shown. The influence of the tool on the workpiece is only possible for workpiece generated quasi-virtually in the course of a simulation run.

### 4. Conclusion

VR-CAM integration has been a challenging scenario. The main difficulty was having to deal with Vericut®, i.e. a closed commercial software, therefore not customizable. Many commercial software have restrictions in their implementation. Each construction process of a new virtual machining environment system is akin to the reinvention of "wheel". The problems encountered are both purely geometric and numerical ones. For instance, a geometric problem has been the right coincidence of the reference systems and orientation when passing from MasterCAM software to Vericut/IC.IDO platform. The reference systems are of considerable importance since even a minimal alignment error leads to the corruption of all the animations, thus giving the final session completely incorrect.

In Vericut, the export of the CAD models of the workpiece during machining is not yet well developed, and the export of the CAD model of the CNC machine does not exist as functionality. This made it necessary, on the one hand, to compromise on the quality of simulations in the Virtual Environments and, on the other hand, to manually model a large part of the machine.

The parsing activity was also very challenging. Two are the reasons. The complexity of the kinematics machine is the first. The amount of data to be processed the second: every line of the G-code is translated into dozens of IC.IDO graphics engine instructions.

Another problem has been the different programming language used by MASTERCAM and IC.IDO. Hence, there has been a need to develop a third specific software, NCtoIDO, to interface the two software.

The conducted VR-CAM integration allows the industrial companies to simulate in a realistic way the machining operations, improving their understanding of the process, and consequently reducing the time requested to validate the CNC part-programs.

### Acknowledgements

This work was funded by InnovaPuglia, the inhouse company of the Apulia region, Italy, fostering technological innovation. The funding was the "Virtual MANufacturing" grant (n. T227BY5) of the Regional Technological Clusters Call.

### References

- [1] Rüßmann M. et al. "Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries." Boston Consulting 62.4 (2015): 40–41
- [2] Caputo, F., Di Gironimo, G., & Marzano, A. *A structured approach to simulate manufacturing systems in virtual environment*. In Congresso International de Ingegneria Grafica, 2006, pp. 73-80.
- [3] Kao, Y. C., Lee, C. S., Liu, Z. R., & Lin, Y. F. *Case study of virtual reality in CNC machine tool exhibition*. In MATEC Web of Conferences, 2017, Vol. 123, p. 00004.
- [4] Choi, S., Jung, K., & Noh, S. D. *Virtual reality applications in manufacturing industries: Past research, present findings, and future directions*. Concurrent Engineering, 2015, 23(1), 40-63.
- [5] Fletcher, C., Ritchie, J., Lim, T., & Sung, R. *The development of an integrated haptic VR machining environment for the automatic generation of process plans*. Computers in Industry, 2013, 64(8), 1045-1060.
- [6] Chang, W. Y., Hsu, B. Y., & Hsu, J. W. *Real-Time Collision Avoidance for Five-Axis CNC Machine Tool Based on Cyber-Physical System*. In 2018 IEEE International Conference on Advanced Manufacturing (ICAM), 2018, pp. 284-287. IEEE.
- [7] Pürzel, F., Klimant, P., Wittstock, V., & Kuhl, M. *Real NC Control Unit and Virtual Machine to Improve Operator Training*. Procedia Computer Science, 2013, 25, 98-107.
- [8] Neugebauer, R., P. Klimant, and V. Wittstock. "Virtual-Reality-Based Simulation of NC Programs Formilling Machines." Global Product Development - Proceedings of the 20th CIRP Design Conference. Springer Berlin, 2011. 697–703.
- [9] Neugebauer, R., Klimant, P., & Wittstock, V. *Virtual-Reality-Based Simulation of NC Programs for Milling Machines*. In Global Product Development, 2011, pp. 697-703. Springer, Berlin, Heidelberg.
- [10] Luo, L. F., Sun, S., Meng, Q. G., & Li, Q. Q. *The process planning simulation of multi-axis numerical control based on virtual reality*. In Advanced Materials Research, 2010, Vol. 97, pp. 3146-3150). Trans Tech Publication