KBE Application for the Design and Manufacture of HSM Fixtures

J. Ríos, J. V. Jiménez, J. Pérez, A. Vizán, J. L. Menéndez, F. Más

The design of machining fixtures for aeronautical parts is strongly based in the knowledge of the fixture designer, and it comprises certain repetitive tasks. An analysis of the design process allows us to state its suitability for developing Knowledge Based Engineering (KBE) applications in order to capture the knowledge, and to systematize and automate the designs.

This work justifies the importance of fixtures for High Speed Milling (HSM), and explains the development of a KBE application to automate the design and manufacturing of such elements. The application is the outcome of a project carried out in collaboration with the company EADS.

In the development process, a specific methodology was used in order to represent the knowledge in a semi-structured way and to document the information needed to define the system. The developed KBE application is independent of the parts design system. This makes it necessary to use an interface to input the part geometry into the KBE application, where it is analyzed in order to extract the relevant information for the fixture design process. The results obtained from the application come in three different ways: raw material drawings, fixture 3D solid models, and text files (Bill Of Materials – BOM, and Numerical Control - NC programs). All the results are exported to other applications for use in other tasks. The designer interacts with the application through an ad hoc interface, where he is asked to select or input some data and where the results are also visualized.

The prototype KBE application has been carried out in the ICAD development environment and the main interface is with the CAD/CAM system CATIA V4.

Keywords: KBE, ICAD, HSM fixtures design.

1 Introduction

High Speed Milling (HSM) has been widely used in the aerospace sector for several years. The benefits of using HSM in manufacturing of components made of aluminium, titanium or hardened steel alloys have also been clearly identified. It is in addition recognized that HSM allows us to mill complex structures in aluminium that previously were neither practical nor possible to achieve. Mainly, it allows us to achieve thinner walls to reduce the weight and to machine monolithic structures that save on assembly. Frames and components of different sizes and with different degrees of geometrical complexity made of aluminium and titanium are very common in the aeronautical sector. In general, the benefits come from the use of higher feed rates and cutting speeds, and lighter depths of cut, making this combination cost-effective. However, it is also widely admitted that the use of HSM demands a change in the philosophy and the approach to machining. In addition to the change in the cutting parameters both in rough and finishing operations, dedicated tooling and production equipment incorporating the necessary design features is needed. In this sense, rigidity to avoid vibrations is a key factor, set-up stability is part of the solution, and fixtures are the elements responsible for part of this increased stability. In addition to the stability requirement, some other factors affect the design of fixtures for HSM, mainly the thinner part walls, the new machining strategies, the reduction in the number of set-ups, and the increased amount of material to become chips. Erdel [1] provides a comprehensive view of the HSM process, but more specific information can be obtained from tool manufacturers such as Sandvik and Iscar.

As mentioned in the previous paragraph, work holding is a key element in the machining process. Some of the factors to be considered when designing jigs and fixtures are: quantity of work, production rate, machine capacity, sequence of operations, tolerances, interferences, cutting forces, chip evacuation, part dimensions and shape, etc. At the same time, requirements to consider are: simplicity, rigidity, accuracy, durability, set-up times, and economy, just to name some of the most relevant factors [2].

In the case of HSM for aeronautical parts, production is frequently carried out in a machining cell made up of several

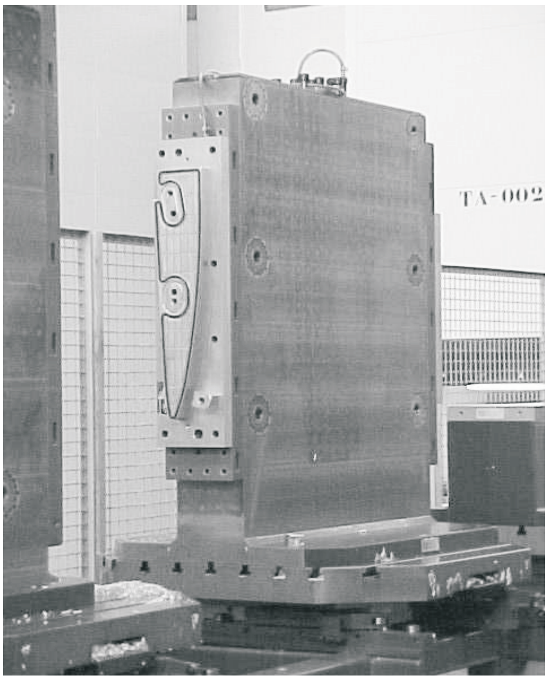

Fig. 1: Fixture assembly for HSM

HSM machines. Precision pallets, hole-based two or four--sided tooling blocks and base plates, together with vacuum/non-vacuum specific purpose fixtures, magnetic clamping elements (when ferrous materials are machined), and modular fixture elements are the main components of the system. Fig. 1 shows a machining fixture with a vacuum circuit for CNC machining in a typical assembly for HSM made up of the following elements: pallet, tooling block, base plate, and vacuum specific fixture.

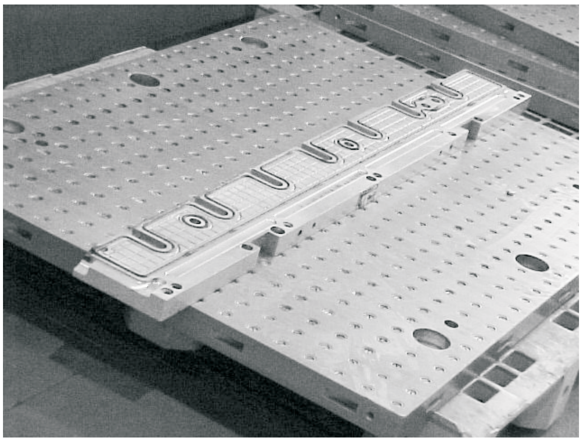

Fig. 2: Example of vacuum specific fixture

Fig. 3: Example of a vacuum specific fixture

Vacuum specific fixtures are frequently used for machining parts that require a single set-up, which cannot be achieved with any other kind of locating and clamping elements. As an example, in the aerospace sector fixtures of this kind are used for parts that are machined from a block of raw material and in which NC planning, pocketing, contouring and drilling operations are performed (Fig. 2), and for skin panels on which NC trimming and drilling operations are performed (Fig. 3). Cutting strategies are also designed in order to achieve maximum stability and rigidity during the machining process, in order to avoid deformation of the part and generation of vibrations, as one of the requirements for HSM is to be chatter-free. For example, during the contouring operation a thin foil web of about 0.01 mm of material is left to keep the part attached to the vacuum fixture. Once the machining is completed and the pallet is out of the machining stations, the vacuum is released and the part can be snapped out of the surrounding material. In order to help in getting a chatter-free machining environment, vacuum fixtures also provide a very fast clamping and un-clamping cycle time.

Looking at the design process of fixtures, there is wide recognition of the extensive use of heuristic knowledge during such a process, as well as the dependencies between some of the information used. As an example, when defining the machining operations, the fixture solution should be kept in mind, and vice versa. Experience and skills, gathered and kept by designers in the form of 'explicit' and 'tacit' knowledge for several years, are a key factor in achieving a good fixture design. This fact, together with:

1. the extensive information needed during the design process, mainly related to the part, the machining process, resources, and production; and
2. the complexity of the design itself, which implies that we must determine the locating supporting and clamping positions and the corresponding physical fixture elements, considering mainly the requirements of stability, rigidity, deformability, accuracy, accessibility, interference, availability and cost;

make it extremely difficult to automate the design process of fixtures completely.

This is the main reason why much research work focuses on specific issues of fixture design (locating and clamping layout, fixture force analysis, fixture tolerances analysis, etc.), and address a specific kind of fixtures, e. g., modular fixtures, or consider only parts with a specific kind of geometry, e. g., prismatic forms.

The first attempts to develop a Computer Aided Design application for machining fixtures date back around twenty years [3]. With the improvements in feature based design, geometry analysis algorithms, knowledge capture and representation, and artificial intelligence techniques, the development of such applications has been facilitated, but the extensive expertise needed during the process makes this area of research still extremely challenging. A comprehensive study of different systems can be found in [4] and [5], in the work carried out by Hou and Trappey on modular fixtures [6], one the latest studies in this area, which also provides an interesting literature survey.

This paper presents the development of a KBE system applied to the design of fixtures for HSM of aeronautical components. Due to the special characteristics of the fixture design process and the complexity of the knowledge involved, the application integrates knowledge represented in the form of design rules with knowledge provided by the designer in the form of input parameters. Because of the specific requirements, the development framework encompasses two different systems: CATIA V4 (CAD/CAM system) and ICAD (KBE development software).

2 Automating the design of fixtures

As stated previously, automation of fixture design has been pursued for several years. Lately, the topic seems to have returned, basically due to the application of various techniques for the reasoning process needed to provide a possible solution to the problem. Genetic algorithms [7], agent based systems [8], and machine-learning techniques [9] are being applied to automation of the fixture design process. The use of these techniques, framed under the general discipline of Artificial Intelligence, claims to facilitate the capture of the 'what', the 'how' and the 'why' needed to carry out the design reasoning. At this point, it seems relevant to point out that KBE systems can be considered a particular technique for automating design processes, and in fact this term is currently more likely to be used than the former Expert System. In this sense, one of the main aims of any KBE system is to automate routine designs that require a significant amount of time when performed manually [10].

One of the main questions, when addressing the automation of any design process, is what degree of automation should be achieved. To answer this question is not an easy task, since many human, technical and economical factors should be taken into account. In particular, there is one key factor to consider, which is the *amount of creative or routine work involved in the process*. However, it should be considered that the automated design process should help the designer to carry out more creative work but at the same time to make use of part of his 'knowledge'. Above all, this concerns knowledge that the designer, can input easily, while failure to provide such knowledge would involve much extra effort.

In this sense, when addressing the automation of fixture design, one of the main tasks is to analyse the geometry of the part. It is essential to know the raw or initial material and the final part that is to be obtained, in order to determine the volumes of material to remove. These volumes will help to determine the necessary machining operations, but the order in which they will be performed cannot be defined without taking into consideration many other factors like the tolerances, the machine tool and the fixture solution that it will be used. When dealing with aeronautical parts, the analysis of the part geometry turns out to be really complicated, because the parts themselves are geometrically complex. In this particular case, this complexity does not allow the wide use of design features. The general practice for many aerospace components is to use complex curves and surfaces as the starting point to design a 3D solid model. Fig. 4 shows three examples of parts considered in the project.

The method used to design the part, in particular the kind of geometric primitives and functions used, influences tremendously on the later analysis of the component. This consideration is even more relevant when a translation of the geometric model between different systems is needed. The problem that usually arises is the loss of the geometry modelling history of the part. This implies an upper degree of complexity when analyzing the received geometric model in the system where the automated design will be carried out. In particular, prior to the design of the fixture solution, we need to identify, dimension and locate the supporting face, the supporting face contour, the wall thicknesses, holes, pockets, and possible internal and external remnants from the part.

In HSM, the possible remnants are particularly important because they need to be clamped to avoid vibrations or to avoid snapping during the machining process. The supporting face contour is also relevant, for two main reasons: the first is to dimension the specific support fixture to the minimum size to support the part properly, in order to reduce the length of the tool, and in consequence the possibility of chatter, when machining the part with a tool the axis of which is parallel to the supporting plane; and the second reason is that the contour is used as an input to determine the vacuum system geometry, if needed.

In order to minimize part of the complexity of the geometrical analysis to be carried out by the automated application, two different kinds of actions can be performed by the fixture designer during preparation of the geometry of the part: geometry generation practices and addition of information. In terms of geometry generation, one of the practices is to make sure that there are neither gaps nor duplication between the curves and surfaces that define the faces of the part. In terms of adding information to the model, one example is to locate the part with the intended supporting face with one of its coordinates at value 0. The selection of the coordinate should be based on the configuration of the NC machine tool to be used.

3 KBE development environment

The development of KBE applications implies the use of a particular environment providing a programming language that allows us to represent product and design process knowledge, and a reasoning process. The latter is referred to in general as the inference engine, and its basic function is to derive answers from the knowledge applied to the initial data. In the engineering design context, the relation between CAD systems and what used to be called several years ago 'expert systems', currently referred to as KBE systems, dates back around twenty years. The reason for this relation lies in the need for any KBE system to be used in the design environment of a link with a geometric modelling kernel. In this

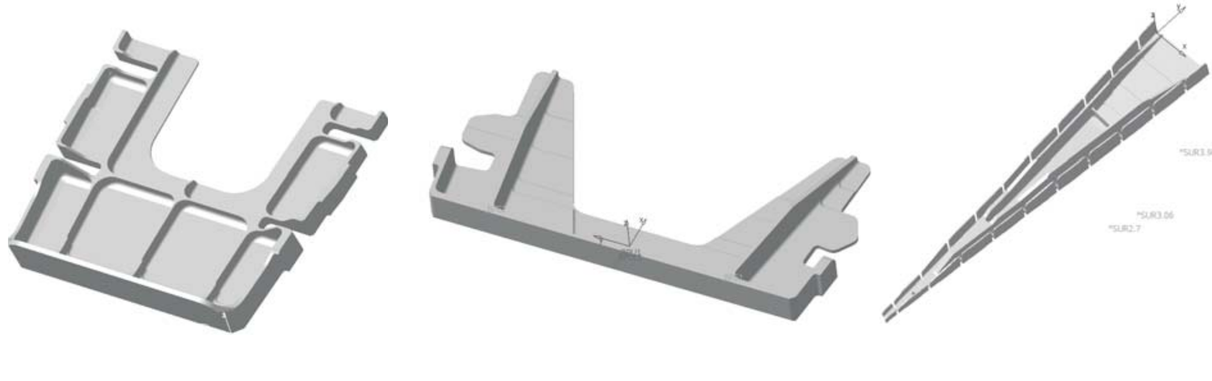

Fig. 4: Examples of aeronautical parts used in the project

sense, Parasolid, ACIS, and Open Cascade are some of the geometric modellers currently used by some KBE systems.

In some way, the application program interface provided by some CAD systems in the form of programming subroutines, which can be called from programs developed mainly in Fortran, may be considered as the first version of a KBE programming environment. In fact, rule-based systems, one of the typical Artificial Intelligence (AI) methods, which constitutes the basis for several of the current KBE systems, such as ICAD and INTENT!, can be implemented using the construction 'IF … THEN' in Fortran language. However, with this approach it is necessary to define the reasoning process as well, which in fact would be a forward reasoning method based on the arrival of new facts represented by variable values. The reasoning process should be defined in the flow of the program. In this sense, it is important to point out that, currently, Object Oriented languages like C++ allow the development of 'expert systems' even though their development environments do not provide any reasoning process or inference engine, and this has to be implemented basically through the definition of methods and messages between objects. Callan [11] provides a comprehensive introduction to AI techniques.

Coming back to KBE applications, it can be considered that there are currently five main commercial KBE development environments:

- ICAD (Intelligent Computed Aided Design) based on LISP language, and using mainly Parasolid as the geometric modelling kernel.
- INTENT! based on LISP language and using ACIS as the geometric modelling kernel.
- PACELAB based on C++ language and using Open Cascade as the geometric modelling kernel.
- Unigraphics Knowledge Fusion based on INTENT! as the modelling language and using Parasolid as the geometric modelling kernel.
- CATIA V5-Component Application Architecture based on C++ and having its own geometric modelling kernel.

In the particular case of our project, CATIA V4 and ICAD constituted the development environment. CATIA V4 was used as the geometry modeller of the aeronautical part. And ICAD was used for the development of the KBE application. The geometry of the part was imported into ICAD via a CATIA model processor and later the resulting fixture design, generated by ICAD, was exported into CATIA.

As has been previously mentioned, ICAD is based on LISP language. It provides what is called ICAD Design Language (IDL) and a forward/backward inference engine, which means that the reasoning process can be done from the start state to the goal state, and vice versa. For the geometric modelling functions, ICAD provides two options, the Parasolid kernel and a surface based kernel. The development of the KBE application user interface is done independently of the knowledge modelling.

The basic components of an ICAD application are the IDL code elements named DEFPART and DEFUN. The first element can be used to define the geometric and non-geometric entities, and it is structured in a hierarchical tree. The highest level of the tree is a root DEFPART that encompasses the root DEFPART of the user interface and the root DEFPART of the geometrical components. A DEFPART has a basic type and several sections, the main ones being: part-documentation, inputs, optional-inputs, modifiable-inputs, query-attributes, attributes, attribute-types, descendant-attributes, pseudo-parts, and parts. The later element, DEFUN, allows us to code the traditional functions, which take the parameters as input, apply an algorithm, and return a result. In addition to the DEFPART files, an ICAD application may include catalog files. These are text files where the parameters defined for the application are stored together with their possible values. In particular, this kind of file can be used to define the parameters needed for the decision rules and for the generation of library components.

4 Capturing and representing product and design process knowledge

The capture and representation of knowledge is a discipline that has attracted considerable attention from the research perspective in recent years, mainly due key role in the area of knowledge management. Particularly, in the subarea related to KBE systems there are two main initiatives that must be considered: CommonKADS [12], and MOKA [10].

CommonKADS is a methodology for the development of Knowledge-Based Systems (KBS). It considers the use of the tool set PC-PACK as a knowledge elicitation tool, and in different parts it leans on the use of the Unified Modelling Language (UML). It proposed three main modelling steps:

- Context Modelling. This encompasses three different models: organization, task and agent models.
- Knowledge Modelling. This also includes three different models: domain knowledge (static view), inference knowledge (reasoning process), and task knowledge (application goals).
- Communication Modelling. This encompasses the definition of the information exchange procedures to perform the knowledge transfer between agents.

MOKA is another methodology influenced by CommonKADS, amongst other techniques; it focuses on the development of KBE applications. The main objective of MOKA is to help to reduce the effort needed and the risk associated with the development of KBE applications. It defines two levels of knowledge representation: an informal level and a formal level. The informal level is based on the use of specific forms named ICARE forms: Illustrations, Constraints, Activities, Rules and Entities. The formal level comprises the transformation of the knowledge defined in the ICARE forms into UML based diagrams. The idea is to use these object oriented models as an input in the coding of the KBE application. In this sense, it is relevant to note that the company that commercializes ICAD software was one of the major partners in the development of the MOKA methodology. Currently, the PC-PACK tool supports the generation of ICARE forms.

In the development of this project, in addition to the documents defined by the industrial partner of the project to represent the HSM fixture requirements, only the MOKA informal level was considered. The ICARE forms were mainly used to represent constraints and rules associated with the fixture design process.

5 A knowledge based application for the design and manufacture of fixtures

The design of the fixture solution for a particular aeronautical component to be machined by HSM implies the work of various technicians and the development of a considerable number of steps. Most of these steps are carried out making use of a CAD/CAM tool. As it was stated previously, information about the part, the possible machining process, and available resources, has to be gathered and put together. A detailed analysis of the whole process reveals that its automation should be addressed partially, with the main objective of providing an application to the fixture designer, which will help him in his most routine tasks.

A fundamental element due to be considered is the fixture philosophy, which in fact depends on the configuration of the production resources and the family of parts to be machined. This determines the kind of fixture elements that need to be considered. In this particular case, and as previously stated, the HSM of aeronautical parts is frequently carried out in a machining cell made up of several HSM machines. This

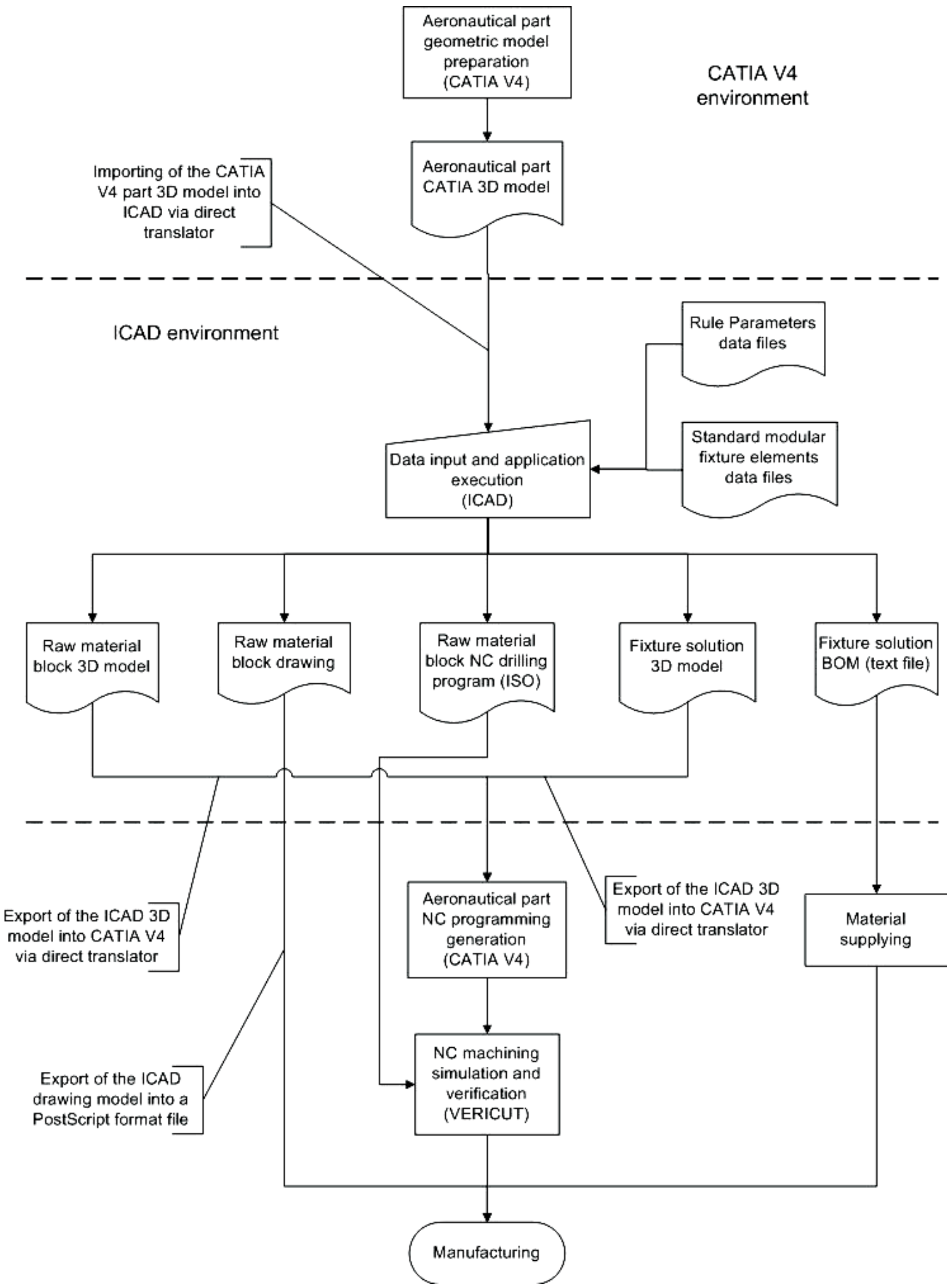

Fig. 5: General view of the developed ICAD application context

implies the use of precision pallets, hole-based two or four-sided tooling blocks and base plates, together with vacuum/non-vacuum specific purpose fixtures, and some modular fixture elements. In relation to the family of parts, the project initially addressed parts that have a planar supporting face, which can be defined by one or more geometric surfaces (Fig. 4).

Fig. 5 presents a general view of the development application context, and of the outputs. The knowledge needed for its operation is defined and codified in the form of two main kinds of files: code files, based mainly on the DEFPART and DEFUN constructions, and parameter files for decision rules, catalogs, and for generating standard components.

The general requirements specified for the application development are summarized in the following terms:

- The geometry of the aeronautical part will be generated in CATIA V4. An output model in CATIA format will be created and it will be imported into ICAD via a direct translator.
- Unless specified in a different way, the ICAD application will calculate and generate all the geometry needed based on: geometric analysis, application interface and parameter files inputs, and design rules.
- The ICAD application will generate the raw material of the part. This part implies the following outputs:
  - A 3D solid model of the raw material needed for the part, including all the holes for fixing it and for handling on the specific base fixture. The holes will be designed according to the company standards. The kind of holes to be considered are as follows:
    - Holes for guiding/locating pins.
    - Holes for fixing screws.
      - Standard fixing holes.
      - Non-standard fixing holes. These can be of two types:
        - Those set by the designer. They will be situated close to the location points specified by the designer in the CATIA model.
        - Those needed for fixing the pocket remnants, external remnants and narrow strips or inlets of material. The situation of these holes will be calculated by the application based on a geometric analysis and the design rules.
      - Holes for lifting screws.

    This ICAD 3D solid model will be exported to a CATIA V4 model.
  - A drawing of the raw material for the part, according to the company standards and including all the annotations, dimensions and details needed for its machining. The drawing will be exported into a PostScript format file.
  - A text file with the NC program in ISO format for the stock material drilling operations.
- The ICAD application will generate the specific base fixture for machining the raw material for the part. This part implies the following outputs:
  - A 3D solid model of the specific base fixture. The model will include the following elements:
    - A base plate.
    - A fixture block with the following kinds of holes, and the corresponding threaded inserts, guide bushes and screws:
      - Holes for fixing screws.
        - To fix the raw material for the part to the fixture.
        - To fix the base fixture to a base plate.
      - Holes for guiding pins.
        - To guide the assembly of the raw material for the part on the base fixture.
        - To guide the assembly of the base fixture on the base plate.
      - Holes for lifting screws. To handle the base fixture during the assembly process.
    - The fixture block will have the following additional elements when requested by the designer:
      - A vacuum system, including: vacuum nozzle, vacuum channels, vacuum grid, connecting holes, connecting grooves, and sealing grooves.
      - A clamping system. Three kinds of clamps are considered: plain clamps, adjustable clamps and bridge clamps. This system includes all the needed fixing holes, inserts, and components of the clamps.

    This ICAD 3D solid model will be exported to a CATIA V4 model.
  - A bill of materials (BOM). A text file including all the elements that make up the fixture solution.

Due to the complexity of the application, the development, was divided into six main areas: an analysis of the imported geometry, calculation and generation of the raw material for the part, calculation and generation of the fixture solution, generation of drawings, generation of text files, and the user interface. In each of these areas, specific functions were developed to resolve the different geometric analyses needed and to overcome the problems due to the complexity of the geometry of the parts.

In this sense, one of the most challenging tasks was to develop a method to locate, without interferences and conforming to the standards of the industrial partner, all the elements to be included in the drawing of the raw material for the part. Fig. 6 shows an example of a created drawing6.

To address this issue a methodology based on the concept of a 'quality engine' was developed, and implemented in the form of decision rules. This concept of quality, evaluated on the basis of quantifiable parameters, was applied to each element of the drawing and to the whole drawing as a single element. Additionally, the elements were classified into three groups: with a fixed location, with a restricted location, and with a free location.

The generation of the drawing was defined in five main phases:

- Phase 0. Identifying of the geometry of the part.
- Phase 1. Drawing preparation. Locating the fixed elements.

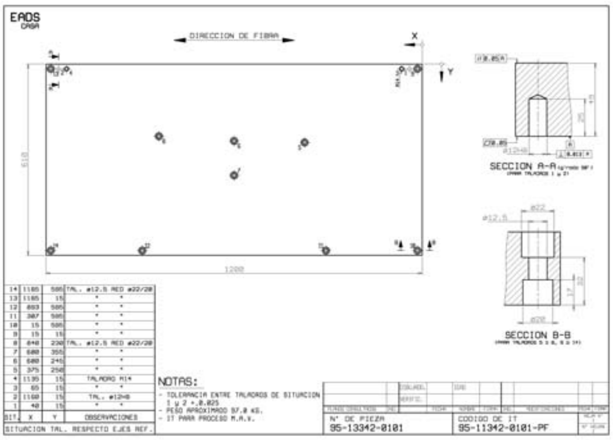

Fig. 6: Example of a drawing of the raw material for a part generated with the developed application

- Phase 2. Determining of the number and kind of views and sections.
- Phase 3. Generating the fixed and mobile elements in pseudo-drawings (virtual intermediate drawing).
- Phase 4. Locating all elements in the drawing. Final configuration.

Phase 3 is where the interference analysis and the quality quantification are performed.

In the fixture solution, in addition to the development of decision rules, the generation of the curves for the vacuum system needed a specific development. The generation of these curves is based on inner offsetting supporting face external contour of the aeronautical part and the outer offsetting of the contour of the through pockets and holes. The family of parts considered allows us to have a supporting face geometrically defined by more than one geometric surface or geometric face, which adds another element to the complexity of the problem. Although ICAD provides a function to generate such curves, in reality due to the special requirements imposed by the geometric contour of the part, the offsetting of some curves resulted in curves with self-intersected loops, or in curves that were just one of the loops of the whole offset curve. This kind of result was totally invalid for generating the vacuum grooves and the vacuum grid. It is relevant to be pointed out that the problem of self-intersected loops in offset curves has been studied by various authors, due to its importance in the generation of surfaces starting from such curves, and in the generation of tool paths for NC machining. In this particular case, the solution adopted was based on discretizing the curves by points, and application of the offset distance to such points in the direction of its normal to the curve. The pitch between points was defined as a parameter, allowing a finer or rougher value depending on the geometric characteristics of the contour. Once the offset points are generated, a list of segments is created. The evaluation of the intersection between segments allows us to identify all the possible self-loops. An algorithm was developed to determine and eliminate the points that are included in the loops. As a result, an approximated curve is generated with the remaining points. This process was applied first to the most critical offset curve, which is the offset curve used to define the connection of the vacuum grid, since it is the most inner offset curve to the external contour of the part.

Fig. 7 shows two examples of a vacuum fixture solution for two of the parts depicted in Fig. 4.

6 Conclusions

Several findings and conclusions can be extracted from the project presented in this paper:

1. Due to the complexity of the fixture design process, its automation should be addressed in partial stages, combining designer decisions with automatic knowledge-based expert judgments.
2. The knowledge elicitation process plays a key role in the success of the development process. The use of a formalized technique like MOKA, or even a company specific one, results in better and faster application development, maintenance and possible future extension.
3. The translation of complex geometric models of parts between dissimilar systems is still a problematic issue when a deep analysis of the geometry is necessary in the receiving system.
4. The correct decision between using forward reasoning, which implies the application of rules from the start conditions followed by the necessary calculations and reasoning until the goal is achieved, or using backward reasoning, which implies the application of rules to a calculated general final state until a compliant solution is

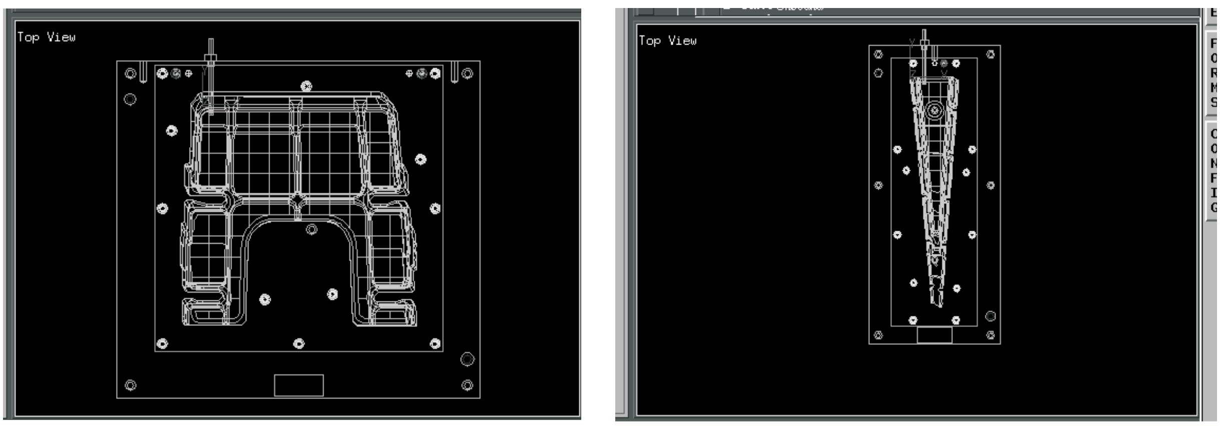

Fig. 7: Examples of 3D models of specific vacuum fixtures obtained with the developed ICAD application

achieved, is another key factor to consider when developing a KBE application.

5. The benefits obtained from the development of a KBE application go beyond the use of the application itself, since a deep rationalization and systematization of the design process helps to improve it and to capture tacit knowledge that in regular circumstances is not formally documented in the company.
6. Finally, in order to facilitate the application of KBE techniques, there is a need to develop and integrate knowledge-based tools which do not demand strong programming skills from the designers, in general purpose CAD systems.

Acknowledgments

The authors want to express their most sincere gratitude to the fixture designers from MTAD EADS-CASA at the Tablada factory, who kindly collaborated in this project.

References

[1] Erdel, B.: *High Speed Machining*. USA: SME, 2003.

[2] Boyes, W. E.: *Handbook of Jig and Fixture Design*. 2nd Edition. USA: SME, 1989.

[3] Miller, A. S., Hannam, R. G.: *Computer Aided Design Using a Knowledge Based Approach and its Application to the Design of Jigs and Fixtures*. Proc. I.Mech.E., 1985.

[4] Nee, A. Y. C. et al.: *Advanced Fixture Design for FMS*. Springer-Verlag, London, 1995.

[5] Rong, Y., Zhu, Y.: *Computer Aided Fixture Design*. Marcel Dekker Inc., 1999.

[6] Hou, J.-L., Trappey, A. J. C.: "Computer-Aided Fixture Design System for Comprehensive Modular Fixtures." *Int. Journal Production Research*, Vol. **39** (2001), No. 16, p. 3703–3725.

[7] Senthil, A. et al.: "Conceptual Design of Fixtures Using Genetic Algorithms." *Intl. Advanced Manufacturing Technology*, Vol. **15** (1999), p. 79–84.

[8] Senthil, A. et al.: "A Multi-Agent Approach to Fixture Design." *J. of Intelligent Manuf.*, Vol. **12** (2001), p. 31–42.

[9] Senthil, A. et al.: "Conceptual Design of Fixtures Using Machine-Learning Techniques." *The Int. J. of Advanced Manufacturing Technology*, 2000, p. 176–181.

[10] Stokes, M. (Ed.) et al.: *Managing Engineering Knowledge: MOKA Methodology for Knowledge Based Engineering Applications*. ASME Press, 2001.

[11] Callan, R.: *Artificial Intelligence*. UK: Palgrave MacMillan, 2003.

[12] Schreiber, G. et al.: *Knowledge Eng. and Manag. The CommonKADS methodology*. MIT press, 1999.

Dr. José Ríos
e-mail: j.rios@cranfield.ac.uk

Eng. Juan V. Jiménez

Dr. Jesús Pérez

Dr. Antonio Vizán

Dept. of Mechanical Engineering and Manufacturing

Industrial Engineering Superior College
Polytechnic University of Madrid
José Gutiérrez Abascal 2
Madrid, 28006 Spain

Eng. José L. Menéndez

Eng. Fernando Más

Dept. of CAD/CAM-PDM Systems, MTAD

EADS-CASA
Factory of Tablada,
Av. García Morato S/N
Sevilla, 41011 Spain

