



Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <https://sam.ensam.eu>
Handle ID: <http://hdl.handle.net/10985/16785>

To cite this version :

Otto SALOMONS, Henri PARIS, Lionel ROUCOULES - Process planning as an integration of knowledge in the detailed design phase - International Journal of Computer Integrated Manufacturing - Vol. 16, n°1, p.25-37 - 2003

Any correspondence concerning this service should be sent to the repository

Administrator : scienceouverte@ensam.eu



Process planning as an integration of knowledge in the detailed design phase

ROUCOULES Lionel*, **SALOMONS Otto****, **PARIS Henri*****

** Laboratory of Mechanical System and Concurrent Engineering
University of Technology of Troyes
12, rue Marie Curie BP2060
10 010 Troyes
France
Tel: (+33) (0)3 25 71 80 17
Fax: (+33) (0)3 25 71 56 75
Email: lionel.roucoules@univ-troyes.fr*

*** Laboratory of Design, Production and Management
University of Twente
Department of Mechanical Engineering
P.O. Box 217
7500 AE Enschede
The Netherlands
Tel: (+31) (0)53 489 25 32
Fax: (+31) (0)53 489 36 31
Email: o.w.salomons@wb.utwente.nl*

**** Laboratory Sols, Solides, Structures of Grenoble
Domaine Universitaire BP 53
38 041 Grenoble Cedex 9
France
Tel: (+33) (0)4 76 82 52.74
Fax: (+33) (0)4 76 82 70 43
Email: paris@geo.hmg.inpg.fr*

Abstract

This paper deals with knowledge integration during product design. An integrated design approach is presented based on two phases: the initial design and the detailed design phase. Those two phases must be as concurrent as possible. Nevertheless, minimal structural definition of the product is necessary prior to manufacturing process planning as part of the detailed design phase.

During the initial design phase, the structural technology is chosen according to the product main functions. This phase aims at defining minimal functional data that are used as a starting point for other knowledge integration in the detailed design phase.

During the detailed design, other people that are actively involved in the product development process evaluate the product with their own point of view and their own tools. This paper presents the use and the limits of two features based tools for process planning knowledge integration: a What-if system and a CAPP system. Both tools make an analysis of the product. As a result of these analyses designers add new data to the product definition. Thus the design solution is progressively defined. A Co-operative Design Modeller supports this knowledge integration and gathers the data provided by the What-if and the CAPP systems.

1. Introduction

Nowadays industry aims at applying Concurrent Engineering in practice. Integrated design makes part of concurrent engineering like simultaneous or distributed design methods. The main objectives of integrated design are to reduce the product development time, to increase product quality and to reduce cost.

In order to achieve these objectives a means is to improve the communication within the design group. This communication allows the concurrent integration of more knowledge during the design process. As a result, all the product life cycle issues are addressed in time (concurrent aspect) and in relation with other disciplines (integrated aspect). In integrated design, design iterations are reduced, because constraints of different disciplines are taken into account as soon as possible. This means that conflicting constraints can be identified and solved earlier than in a non-integrated design approach.

In industry, concurrent engineering is used more and more but too few methods and even less (software) tools exist to assist the design. This paper briefly presents the result on the development of such a method using different kind of software. The paper mainly focuses on the communication in order to integrate process planning information during the integrated design process.

In current design practice, the integrated aspects are hard to achieve. Most CAD tools require a user to define a geometrically complete model, as do most CAPP tools. However, process planning can in principle be done with a geometrically incomplete model. At least within the design by least commitment approach, functionally (incomplete) geometry can later be completed by using manufacturing information. Then the process planning integration can be achieved earlier in the design process. In addition, this process can start with a minimum of functional information.

The integrated design method as presented in this paper is composed of two design phases: the initial design phase and the detailed design phase. These two phases are not sequential but concurrent as much as possible. For example, incomplete geometrical information serves as an input to process planning. This means that some tasks have to be realised before others. For instance, minimal structural definition of the product (initial design phase) is necessary prior to process planning as part of the detailed design phase.

Indeed specific knowledge integration has to be made as soon as possible otherwise these constraints (manufacturing, assembly, structure analysing constraints, etc.) are integrated too late and are not taken into account. In other words, specific knowledge integration has to be made as soon as design experts (manufacturing, assembly, etc.) have enough data to make their own evaluation of the product.

The paper is divided into four parts. The second part of the paper briefly presents the main concepts and the method of integrated design. The concept of knowledge integration is then achieved using a product model for integration and a product model to map product functions to product structure. The third part explains the use of a Co-operative Design Modeller (CoDeMo) in the initial design phase. The modeller assists the designers in constructing the product models. These models provide the minimal functional surfaces needed for other integration. The paper presents the design of a robot grip as an example. The fourth part focuses on applying specific process planning tools in detailed design. The connection between those tools (a What-if system and a CAPP system) and CoDeMo assists machining knowledge integration in the product design. The fifth part presents the conclusions and the recommendations.

2. Concepts and methods

Integrated design is presented as a special point of view of concurrent engineering. This method focuses on knowledge integration during design process and is coupled with designers' co-operation. The concept and the means to achieve both integration and co-operation are now presented.

2.1. Integrated design concepts

Concurrent engineering (Sohlenius 1992) leads the stream in design methodology evolution in industries. Above all, it has a positive effect on design time (Aldanondo et al. 2000). Nevertheless concurrent engineering is described in this paper with a knowledge point of view. Two sub-methods are described to better understand how the knowledge is taken into account during the design process:

- Integrated design,
- Distributed design.

2.1.1. Integrated design

Before concurrent engineering much more design iterations appeared during the design process. Those iterations were the results of the too late integration of knowledge on the product design. It was often the case between the two main parts of the industry, the design department and the manufacturing department.

Integrated design aims at taking into account as much data as possible during the design process. These data must be related to the whole product life cycle (Tichkiewitch 1996). In other words, product design has to pay attention to manufacturing, assembling, recycling, etc.

Thus, a designer does not have to wait until he knows there will be a conflict. He must enter the design group and describe the problem with his point of view. He therefore relates his data with other designer's data and formalises the coupled phenomenon in design.

2.1.2. Distributed design

In the design group, each expert has his own knowledge and therefore is not able to provide information outside his experience domain. The integration of new knowledge and the objective of distributed design then means that new experts have to be integrated in the design group. The knowledge is distributed among every designer.

Integrated design and distributed design have really to be studied together. It then appears that integrated design is actually a problem of making several designers communicate during the design process (Brissaud et al. 1996). The communication must allow the data to be reachable and shared by everyone.

2.2. Integrated design method

The integration of knowledge concept has just been presented and aims at reducing design iterations. The integrated design method is based on two phases of integration: the initial design phase and the detailed design phase. During the two phases the designers handle on the one hand a product model for knowledge integration and on the other hand a product model to map product functions to product structure. The final result of the integrated method is to provide the product geometry and not any more to use the product geometry as a starting point (cf. figure 1). This design by least commitment approach has been used by (Mäntylä et al. 1989) for example.

‘[Insert figure 1]’

2.2.1. Two design phases for the integrated design method

Integrated design aims at taking into account all the product life cycle knowledge as soon as possible. However, it is sometimes impossible because of a lack of information. For example, it is impossible to choose a manufacturing process and even less so the manufacturing data without a minimal definition of the product.

Actually, the design begins by choosing an (or a list of) available technological solution according to the product main functions. In this phase the *technologist designer* (expert in technology) integrates his knowledge. This phase is called initial design phase. The initial design phase provides minimal information and minimal functional surfaces (cf. 3.2).

From the minimal data provided by the initial design phase, other designers can integrate their knowledge. This phase is called the detailed design phase. Then, specific designers (manufacturing, assembly expert, etc.) add new data and new functional surfaces on the product (cf. 4.3). For example the manufacturing expert can define initial shape provided by stamping.

The integration concept is respected once all the data are gathered in the same database to be later shared by everybody. Figure 2 shows the central position of the database among the two design phases. The design phases are not sequential any more as they were in sequential design (Beitz et al. 1994). On the contrary, they have to be treated simultaneously as far as possible.

‘[Insert figure 2]’

2.2.2. The product models for the integrated design method

The product model for knowledge integration used in the integrated design method is fully explained in (Tichkiewitch 1996). This model is described with:

- Components that represent the description of the product.
- Links that represents a part of the component that can be hanged from another component.
- Relations that represents a relation between several (at least two) components.

The product model is constructed according to several operations:

- The decomposition of component that allows the product description at different hierarchical level.
- The substitution that substitutes one relation with a group of components, links and relations.
- The multiple-views decomposition that allows the description from a specific point of view. All those views are relative to a specific knowledge (machining for example) excepted the *Frame*¹ and *Geometric* views that result from the knowledge integration (sections 3.2 and 3.3 detail the creation of both *Frame* and *Geometric* product breakdown).

Another product model used to map product functions to product structure is based on the functional model developed by Constant (Tollenaere et al. 1997). This model basically represents the energetic flows in the product; these flows can be mechanical, magnetic, electrical or thermal.

Figure 3 presents a representation of both product models and their connection. Part 3 and 4 present how those models are then used during the design process. The functional product model is only used during the initial phase of design whereas the multiple -views product model is used in both initial and detailed design phases.

‘[Insert figure 3]’

As presented in part 3, the links are progressively mapped to the functional surfaces and to the product geometry. This geometrically incomplete model is then used as a starting point for knowledge integration as presented in the design approach (figure 1).

2.2.3. The formal level of communication using a CAID² system

The integrated design concept presents the interest in integrating a maximum of knowledge in product design. Indeed, this integration reduces the design iteration number and as consequence the time of design. Nowadays, it remains a real problem in making designers communicate with each other. Moreover when foreign partners are involved in a design project it results quite expensive to gather everybody on a single site. The industrial stream is therefore to have remote formal and informal communications instead of gathering every design partners on the same spot. From general concepts of a CAID system presented in (Tichkiewitch 1996), a Co-operative Design Modeller (CoDeMo) has been developed (Roucoules et al. 2000) to supply the formal level of communication.

As shown in figure 4 the concept of integration is, first of all, based on a real exchange of data. If each partner works alone and never talks about his result or his problem he will never be integrated in the design project. Then the

¹ In the present design context, the product *frame* actually represents a set of functional surfaces (Tollenaere et al. 1995). These features lead all quality information on the product (roughness, tolerances etc.).

² Computer Aided Integrated Design.

designer would rather share and compare his data. By this way he needs a formal exchange of data. In CoDeMo a shared database is the central spot of the communication.

‘[Insert figure 4]’

At the first step of the integration the designer should pick up the minimal data (the information) he needs to create his personal evaluation; in other words to create new data. For instance, to calculate how to manufacture a part, the geometry and the quality of the part are needed. Thus the manufacturing process can be chosen (tooling, milling or stamping, etc.) and manufacturing parameters (feed and cut speed, etc.) can be calculated. In CoDeMo each designer can either collect the information editing it with the graphic user's interface (GUI) or using a neutral file automatically generated from the database. The advantage of creating such kind of exchange file is to directly connect the database to any kind of software. However the input and output formats have to be known and it is not always obvious. STEP for example studies such kind of standard format but nowadays they are only really reliable in CAD systems for geometric data (STEP-AP203).

The second step of the integration consists in putting back information in the shared database. This step allows each designer to integrate his own data on the project. Either those data define more and more the product or constrain it according to relations between new and already-integrated data. For example the manufacturing expert can, with his knowledge, constrain the roughness of a machined part to a minimum.

Every user currently makes this operation manually using CoDeMo GUI.

3. Applying a software tool in initial design

Now presented the context of our studies this part of the paper details the initial design phase. In this phase the Co-operative Design Modeller (CoDeMo, cf. 2.2.3) is used as a CAID system. The presentation is based on the development of a mechanical grip handled by a robot.

3.1. Functions and technology of the grip

The grip is designed to take an object with a robot and to release it once moved. Let us assume that the robot leads the global motion of the object. The grip has then to achieve only one function: to realise a total kinematics joint between it and the object to be moved. The joint must not be permanent to keep the possibility to release the object. Figure 5 shows the representation of the grip technological decomposition in its global environment (robot, grip and object). This figure is a screenshot of CoDeMo.

‘[Insert figure 5]’

3.2. Minimal functional surfaces definition

CoDeMo (in the initial design phase) is currently connected to a *Technologist Module* that assists the mapping between functions and technological structure (Tichkiewitch et al. 1999). Step by step the product is then decomposed according to a specific technology.

As described on figure 6, the two product models are progressively constructed. The functional product model leads the breakdown in the initial phase. Every energetic flow is described using the GUI, and the *technologist* view is automatically created. As soon as the product functions are mapped to the structural technology, every link is associated to an energetic flow. Every link therefore represents a functional surface absolutely necessary to the flow conduction. These surfaces are fully relative to the structural technology. The functional surfaces are managed in CoDeMo using the product model breakdown in what we call the *Frame* view. The features are as consequence relative to the surface quality (roughness, tolerances, etc.).

The breakdown realised in the *Frame* view during the initial design is what we call the minimal functional surfaces definition. Extra functional surfaces relative to machining, assembly, etc., will be added later on. Indeed, minimal structural definition of the product is necessary prior to manufacturing process planning as part of the detailed design phase.

‘[Insert figure 6]’

3.3. From the functional surfaces to the geometrically incomplete model and toward specific knowledge integration

The initial design is not only a functional design but provides surfaces that progressively define the product. Obviously a surface has a shape; it is then logical that the initial design also provides part of the geometry. This first geometrical featuring is the result of technological knowledge integration. As presented in the integrated design method, it defines the geometrically incomplete model as a starting point for extra knowledge integration (manufacturing, structure analysing, etc.). This new integration is done in the detailed design phase as soon as possible and progressively provides new surface in order to totally complete the geometric model. The soonest time can even mean at the functional stage as presented in (Gerbaud et al. 1998, Troussier et al. 1999). Part 4 talks about machining knowledge integration. It treats two different approaches that actually provide complementary information on the product during the detailed design phase.

In CoDeMo, each functional surface is associated with one or several geometrical features that define the shape of the surface. Figure 7 presents the geometric decomposition of the product model. This decomposition stores every feature that describes the nominal attributes of the geometry (length, radius, etc.). From time to time, the shape can be constrained according to the technology. The designer has to create relations in the product model. For example the cylindrical kinematics joint between the grip casing and the grip finger provides two cylindrical surfaces on each part. If the shape is not constrained in the initial design phase, it will then be defined and constrained with other knowledge in the detailed design phase.

‘[Insert figure 7]’

4. Applying a software tool in detailed design

The initial design phase provides functional surfaces for the technological structure. At this stage, the manufacturer should have enough data to make his evaluation of the product. This evaluation has to be made as soon as possible to take into account machining capabilities. Thus, current manufacturing problems are avoided before the end of the design.

Two software tools have been studied to show the process planning integration in the detailed design phase (analysis). Then those tools are connected to CoDeMo that supports the knowledge integration (synthesis).

4.1. Presentation of the CAD systems for process planning integration

This part presents two kinds of CAD systems. Those systems are based on machining knowledge and assist the evaluation of parts from a process planning point of view.

4.1.1. A What-if system

The what-if system is developed in the laboratory of Design, Production and Management at the University of Twente³ -The Netherlands-.

The What-If system for machining integration is a prototype of a generative design system that allows users to ask 'What-If' questions on the design. Based on the What-if question that the user has asked, the system automatically generates one or more design alternatives. The system is based on the use of feature technology as well as on functional faces. Based on an existing design the designer might want to know what would happen to a part in his design if he considered another manufacturing technology. The system will then infer the functional faces belonging to the original part and match these with manufacturing features of the domain of the new manufacturing technology. By reasoning with the manufacturing methods up front the system will then generate an alternative. Figure 8 presents an example where the end-user asks for an alternative for the rear fork of a recumbent bicycle made by means of milling.

³ <http://www.opm.wb.utwente.nl/>

‘[Insert figure 8]’

The link between the features and the manufacturing methods as well as the reasoning that results in a newly generated design is defined in an object builder module by a system user. Figure 9 shows the alternative proposed by the system.

‘[Insert figure 9]’

As can be seen from this alternative, it does not yet come up with optimum designs, since it has not considered generating a high-stiffness, low-weight, rear fork. It only produced a manufacturable fork that can fulfil its kinematics function using the same functional surfaces as the original component and which can be produced by means of milling. However, in principle, additional criteria like stiffness, weight could have been defined in the object builder as well. More details on the What-if system can be found in (Salomons et al. 2000)

4.1.2. A CAPP system

The Computer Aided Process Planning system ‘PROPEL’ is developed in the Soils, Solids and Structures Laboratory⁴ of Grenoble -France-.

PROPEL is a CAPP system developed in Open CASCADE environment. Open CASCADE⁵ is a CAD/CAM environment that supports the development of applications based on geometric features. The process planning design is an activity that planned actions (machining operations) while allocating and sharing resources (tools, fixturing, and machine tools). Depending on a complete or a partial model of the part, the software recognises geometrical and technological faces of the part that are able to realise a good locating and clamping for machining. Then it calculates the process plan defining simultaneously machining operations groups and fixture selections.

In PROPEL features are defined interactively by the user. From the part definition the software gives several possibilities to the user concerning the fixturing and the process planning. Those results depend on machining features and resources. PROPEL also gives machining information to the process planner. These information is

⁴ <http://www.3s.hmg.inpg.fr/>

relative to machining difficulties induced by a lack of adapted fixtures or by quality constraints that is not easy to realise. Those machining difficulty information are provided by the software using quality and stability indicators.

From a geometrically complete or incomplete model, a part is described with machining features using the GUI. Then, PROPEL recognises⁶ the geometrically and technologically available surfaces that can achieve the fixturing for machining. It automatically defines the process planning that groups several machining operations using the fixturing solutions. Finally PROPEL uses some gauges to indicate the machining capacity. This capacity relates if the process planning solution can reach the needed surface tolerance.

With all the information provided by PROPEL, it is possible to detail the product definition and thus to improve the manufacturability (Brissaud et al. 1998).

‘[Insert figure 10]’

4.2. Detailed design: the analysis

4.2.1. Starting data for machining analysis

In order to make the machining evaluation of a part; both CAD systems need starting information. This information is extracted from the minimal functional data generated during the initial design phase. Obviously the needed data are different for the two systems.

4.2.1.1. The What-if system

The What-if system is based on a *SolidWorks*TM geometrical model of the studied part. Unfortunately in the initial design phase CoDeMo only provides the definition of functional surfaces but not a geometrically complete model. Two solutions are then proposed to make the connection between CoDeMo and the What-if system (cf. Figure 11). The first way is to use CoDeMo GUI and to edit each component of the product geometric decomposition. Then the designer must re-create the model in *SolidWorks*TM. By the way he must, first of all, define what is called the basic shape of the part. The second way of exchanging the data is to create straight in CoDeMo the basic shape of the part.

⁵ <http://www.opencascade.org>. Open CASCADE is developed by Matra Datavis ion society.

Thus the coherent geometric model is created in CoDeMo. The designer has just to transfer the geometric data via an IGES or STEP format.

'[Insert figure 11]'

4.2.1.2. The CAPP system

As opposed to the What-if system, the CAPP system can recover isolated parts of the geometry. In other words, it can start only with the functional surfaces. The starting data have to be the machined surfaces defined as a shape plus their quality parameters (roughness, Tolerances, etc.). In fact this departure information is very close to the data stored in the *Frame* and *Geometric* views.

The connection between CoDeMo and the CAPP system is made exchanging departure data via a text file. The format has been chosen as simple as possible but could be changed if it is necessary. Indeed we have totally defined the API (Application Interface Programming). The text file as presented on figure 12 contains all the parameters of every feature stored in the *Frame* and *Geometric* views.

'[Insert figure 12]'

4.2.2. Analysis as Knowledge integration in design

After the starting information has been recovered from the shared database, the real integration of machining knowledge can be realised. This section deals with the use of knowledge in rules based systems. Those systems are thereby able to make an analysis of parts from a machining point of view.

4.2.2.1. The What-if system

Once the geometric model is recovered in *SolidWorks™* the what-if system can be run to evaluate a part. The evaluation is made of three steps:

- the functional surfaces recognition,

⁶ The surface recognition is achieved using extra feature information (size, area, etc.) provided by the geometrical

- the basic shape definition,
- the part evaluation based on machining knowledge.

The functional surface recognition can be made automatically from the assembly model of the product. Otherwise the designer can also achieve it manually straight in the system. Each feature is associated to a manufacturing process.

The basic shape is afterwards chosen according to the list of basic shapes stored in the system. Actually a basic shape has already been defined taking out the data from CoDeMo (cf. Figure 11). The evaluation can then be made keeping this first shape or changing it. In this case it is possible to compare different results and to make a choice for the cheapest solution for example. Figure 13 presents two solutions for the grip casing shape. One is issued from a cylindrical basic shape, the other one from a prismatic basic shape. In both solutions the functional surfaces have not been changed and still be the surfaces extracted from the initial design phase.

[Insert figure 13]

Today, all the features for functional surfaces and for basic shapes are not all stored in the What-if system. That means that geometrical model and, above all, the manufacturing knowledge relative to those features do not exist. The designer has then to hardcode the methods relative to the knowledge. In this way he uses a feature builder (cf. 4.1.1.) based on *CommercePackTM*. This builder makes easier the definition of relations between the geometric features and manufacturing process parameters.

4.2.2.2. The CAPP system

The main idea of treating two examples of CAD system was to show how two different analysis approaches could provide different information in the synthesis phase. Whereas the What-if system has provided a basic shape and process information (cost, manufacturing time, etc.), the CAPP system provides more information on process planning (Process plan, Set up, machining operation, fixturing, etc.). Figure 14 presents the different steps of the CAPP system analysis.

model.

'[Insert figure 14]'

The CAPP system PROPEL is feature based. From a geometric model of the part, the user describes the studied part with machining features.

First the system calculates the fixturing solutions (locating and clamping). A fixturing feature is a combination of three locating features, with at least one locating-clamping feature, that can be materialised by a fixture. The first location of the combination must be a primary location and ensures the stability of the feature, the second one must be a secondary location that ensures the orientation of the feature and the third one must be a thirdly location as a stop point. A locating-clamping feature is a geometrical form and a set of specifications able to ensure a locating and a clamping of the part while machining. This feature encapsulates geometrical information (places of location and clamp) coupled with expert technological information that qualifies the ability to locate and clamp the part (location accessibility, location quality, locating-fixturing quality). Then all the information (machining features, fixturing features and workshop) is set to design the process plans.

Two steps are necessary to design the process plan (figure 14). During the first step, we design all the machining processes necessary to obtain each machining feature. For each machining feature, we have to define all resources (machine tools, fixtures, and tools). This analysis is based on intrinsic attributes of the machining feature and on formalised knowledge concerning the design of process plan.

During the second step, relations between machining features to constrain the structure of feasible process plans are checked. This operation consists in removing from all the actual calculated process plans those that do not give guaranties concerning the tolerances among features. It is a mean to identify machining difficulties. During this step, dependence links between machining operations, machine tools, and tools types are created (Paris et al. 2000). It is a mean to ensure the required quality, the machine accessibility and the resources availability.

4.3. Detailed design: the synthesis

At this stage of the design the integration is not complete. Indeed manufacturing data are still on the specific expert possession but are not already shared into the integrated design group. The last step is therefore to integrate them in the Co-operative Design Modeller (CoDeMo). This modeller will then propagate the data to every designer.

The data integration is divided in three steps:

- grip decomposition in the view (in the current paper the *machining* view),
- grip decomposition in the *Frame* and *Geometric* views,
- relation definition between new specific data and already existing ones.

4.3.1. *Machining* view decomposition

From the technologist breakdown (*Technologist* view) done during the initial design, the grip decomposition in the *machining* view defines the new data brought by the machining expert. As shown on figure 15 features that represent the new machining information is associated to a component of the product model. The specific breakdown (i.e. the *machining* breakdown) becomes step by step more accurate and finally manages all the needed machining data.

In CoDeMo database, machining features are relative to manufactured surfaces, to machining process and operations. The *Operation* feature can for example be described with a tool name, a feed speed, a cutting speed, a cost, etc. Nevertheless, new features and/or attributes can be easily added in the database via the GUI.

‘[Insert figure 15]’

4.3.2. *Frame* and *Geometric* views decompositions

In addition to the specific manufacturing data, the machining integration has provided new shapes. These shapes have also to be integrated in CoDeMo to be taken into account by every designer. According to the integrated design method, those shapes are defined as functional surfaces for specific designers (machining expert for example) and not any more as functional surface for the product structure as it was the case in the initial design phase. However they have also to be added in the *Frame* view that stores every functional surface and afterwards in the *Geometric* view that stores the nominal shape data (cf. figure 16). If the basic shape has already been added in those view (cf. 4.2.1) it has not to be added again. Nevertheless if another basic shape has been chosen (cf. 4.2.2) it has to be changed.

‘[Insert figure 16]’

4.3.3. Relations definition

In order to definitively end the integration, relations among shared data have to be created in CoDeMo. A relation has two effects on the design:

- to constrain the design problem by propagating the constraints,
- to show the conflicts that appear among different design point of views.

The constraint propagation is achieved according to the constraint feature associated to the relation. In fact, as for the component decomposition, after a relation between product parameters has been created, a constraint (equality, position, etc.) has to be chosen (cf. figure 17). The constraint propagation is therefore automatically realised.

‘[Insert figure 17]’

If the constraint cannot be propagated, in other words if it is already violated according to the parameters values that means that a conflict exists between those involved parameters. From an integration point of view, it exists a conflict between two or more designers that have created those data parameters. Those conflicts appear as an alarm and have to be resolved with designer's arrangements using the informal level of communication (cf. 5.2). CoDeMo has not the objective to automatically resolve the conflicts.

5. Conclusions and recommendations

5.1. Conclusions

Concurrent engineering is nowadays used in improving the design process in industry. A lot of definitions of concurrent engineering exist but this paper focuses on Integrated Design. The main objectives of integrated design are to reduce the product development time, to increase product quality, and to reduce cost.

This paper presents an integrated design method based on an initial and a detailed design phase. The concept of knowledge integration is then applied toward a design by least commitments. The conclusion for knowledge integration is good enough. Technological information integrated in the initial design phase brings the minimal functional surfaces of the product. Afterwards, extra information is integrated during the detailed design by specific experts that bring new knowledge on the product (manufacturing, structure analysing, etc.). Every product datum is then well managed in the product model in order to be identified in case of modification or re-design. However, the knowledge integration strongly depends on each expert's analysis model. The fewer information is needed by those models the sooner expert's knowledge can be integrated in the product definition.

Software prototypes have also been developed and presented in this paper. Such tools are totally necessary to prove that the concepts of the integrated design method can be computer-supported. A Co-operative Design Modeller (CoDeMo) is used to manage the product models during both initial and detailed design phases. Two computer-aided process planning systems (a What-if system and a CAPP system) have been tested to integrate machining knowledge during the product design. The research presented in this paper concludes that some computing problems still remain to obtain a real integrated design environment. Some solutions are expressed in the following recommendations. The already-computed developments done in CoDeMo using IlogVision library allow the creation of geometric incomplete form as required in the integrated design method. However they do not provide a coherent form modelling that would be very interesting to experts' analysis as FEM, etc. Another problem remains in exchanging data between CoDeMO, the What-if system and the CAPP system. The exchanges are currently done manually and as consequence slow down the design process.

The last conclusion deals with the idea of connecting and comparing different software (a What-if and a CAPP system). These tools were a priori conceptually different, indeed the first system is normally use in a re-design context where as the second one is used in design context. Nevertheless the results show that both concepts are actually complementary and useful to create different kinds of data. The What-if system provides alternatives manufacturing solutions where as the CAPP system focus on the process planning.

5.2. Recommendations

As presented in this paper, part of the specific designer's analysis is based on a geometrically complete model and unfortunately CoDeMo does not provide such a model but geometrically partial model. Currently the designers have to create the geometrically complete model in their own geometric modeller from the geometric data stored in CoDeMo. This exchange of data would be improved using a real geometric modeller in CoDeMo. Open CASCADE libraries seem to be a good solution for that issue. Such a modeller would allow exporting data with a special format as STEP or IGES. The exported data can be then imported for the designer's analysis as FEA, etc.

During the knowledge integration, a lot of analyses are based on a geometric model. However those analyses required more specific information than the only form data. For example a CAPP system as presented in part 4.1.2 needs the functional tolerances of the analysed part. So far the studies and the best results on exchanging data have been obtained using for geometric data with IGES or STEP-AP203. It would be interesting to investigate STEP architecture in order to create exchange format for specific applications as CAPP system, structure analysis, etc. Some works have already been done for example with STEP- AP213 for machining but it is not yet computed in current CAD systems.

Part 4.3.3 of the paper presents how relation and constraints are set between several data of the product. The constraints are then propagated using a solver. If a constraint is violated an alarm advises the designers that a conflicts appears. CoDeMo does not aim at resolving automatically the conflicts. On the contrary, an arrangement must be found. An informal network of communication could be developed. This network would aim at supplying the traditional conversation (talks, sketches, etc.) in a design group where everybody is physically present. Some

tools already exist on the market as *Microsoft NetMeeting*TM and have to be studied. Other studies try to define what have to be the new CAD systems taking into account social aspects (Boujut et al. 2000).

REFERENCES

- Aldanondo M., Rougié S., Veron M., 2000, Expert configurator for concurrent engineering Caméléon software and model. *Journal of Intelligent Manufacturing*, **11**, 127-134.
- Beitz W., Küttner K.H., 1994, *Handbook of Mechanical Engineering* (springler verlag).
- Boujut J.F., Blanco E., 2000, Intermediary objects as a means to foster co-operation in engineering design. *Proceedings of the Fourth International Conference on the Design of Co-operative Systems, Sophia-Antipolis (France)*, 23-26 May.
- Brissaud D., Garro O., 1996, An approach to Concurrent Engineering Using Distributed Design. *CERA journal*, **4**(3), 303-311.
- Brissaud D., Paris H., 1998, Simultaneous design and process planning by indicators. *Proceedings of BASYS'98, Prague, August*.
- Gerbaud L., Lechevalier Ch., Bolopion A., Bignon J., 1998, Automatisation de la modélisation à topologie variable des convertisseurs statiques. *The European Physical Journal - Applied Physics*, **2**(2), 235-252.
- Mäntylä M., Opas J., Puhakka J., 1989, Generative process planning of prismatic parts by feature relaxation. *Advanced of Design Automation*, presented at the ASME Design Technical Conference – 15th Design Automation Conference, Montreal, 49-60.
- Paris H., Brissaud D., 2000, Modelling for process planning: the links between process planning entities. *Robotics and Computer Integrated Manufacturing*, **16**(4), 259-266.
- Roucoules L., Tichkiewitch S., 2000, CoDE: a Co-operative Design Environment. A new generation of CAD systems, *CERA journal*, Vol.8, n°4, pp 263-280.
- Salomons O.W., Popma M., Houten F.J.A.M. van, 2000, What-If synthesis support tools as a new paradigm in CAD. *Proceedings of TMCE 2000 conference, Delft, April*, pp. 169-182.
- Sohlenius G., 1992, Concurrent Engineering. *Annals of the CIRP*, **41**(2), 645-655.
- Tichkiewitch S., 1996, Specification on integrated design methodology using a multiple-views product model. *ESDA proceedings of the 1996 System Design and Analysis Conference (ASME)*, **80**, pp. 101-108.

Tichkiewitch S., Roucoules L., 1999, Innovative design methodology. Integration of process Knowledge into Design Support (Kluwer Academic Publishers). ISBN 0-7923-5655-1, pp. 79-88.

Tollenaere M., Belloy Ph., Tichkiewitch S., 1995, "A part description model for the preliminary design", Advanced CAD/CAM Systems - State-of-the-art and future trends in feature technology, pp 129-143, Chapman & Hall, Ed. Soenen.

Tollenaere M., Constant D., 1997, Linking conceptual and embodiment design of mechanical systems. Proceedings of the 10th International Conference on Engineering Design (ICED'97), Tampere, August 19-21.

Troussier N., Pourroy F., Tollenaere M., Trebucq B., 1999, A model to represent mechanical calculation process in an integrated design context. Proceedings of the 11th International Conference on Engineering Design (ICED'99), Munich, August 25-27.

LIST OF FIGURE CAPTIONS

Figure 1: New approach in product design method.

Figure 2: The initial and detailed design phases.

Figure 3: Representations of the product model for integration and the functional product model.

Figure 4: Concept of knowledge integration via a formal and an informal exchange of data.

Figure 5: Grip technological decomposition: a preview of functions and technologies.

Figure 6: Functional surfaces in the functional product model and in the *Frame* view.

Figure 7: Minimal definition of the geometry as a result in the initial design phase.

Figure 8: Existing design of the rear fork of a bicycle (sheet metal welding process).

Figure 9: Alternative design proposed by the What-if system for the rear fork of a bicycle (milling process).

Figure 10: Process planning definition using a CAPP system.

Figure 11: Two means in exchanging data from CoDeMo to the What-if system.

Figure 12: Exchanging data from CoDeMo to the CAPP system.

Figure 13: Manufacturing analysis steps using the What-if synthesis system.

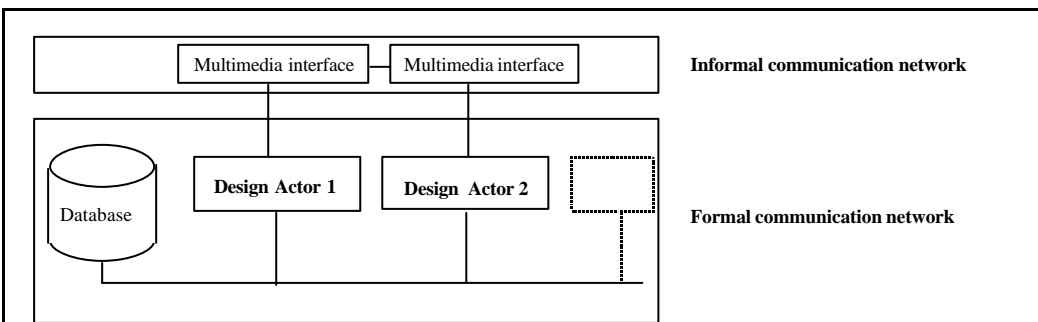
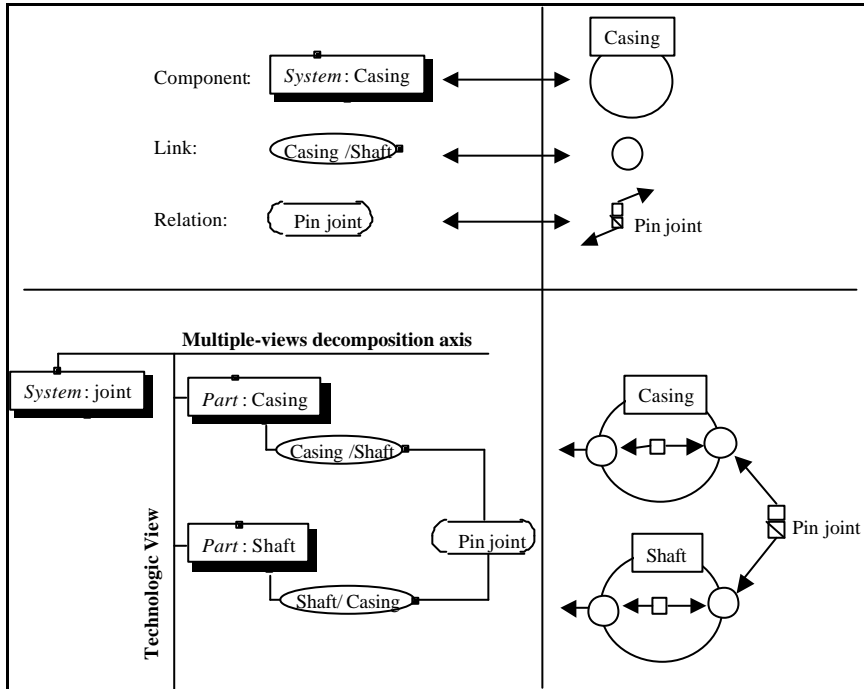
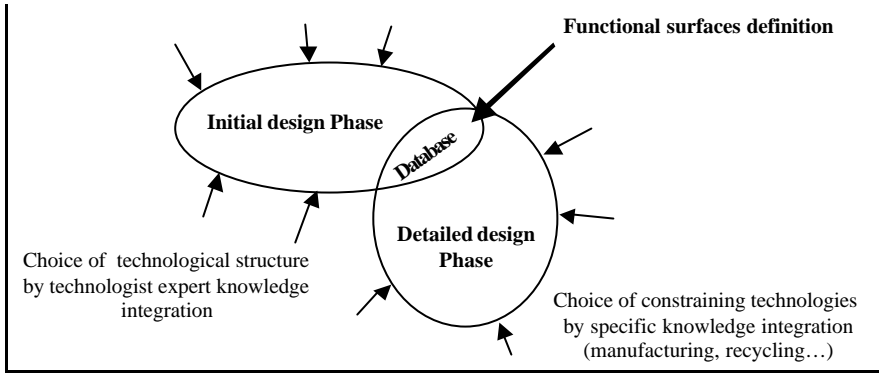
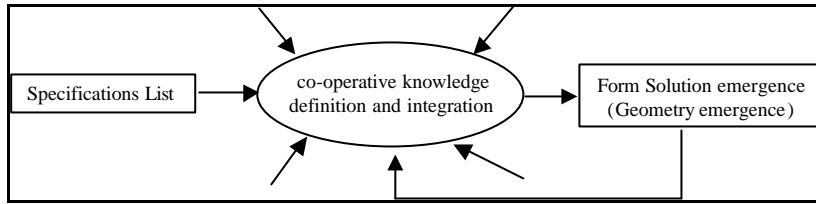
Figure 14: Steps of the CAPP system analysis from the functional surfaces of a part.

Figure 15: *Machining* view decomposition of the grip casing for integration.

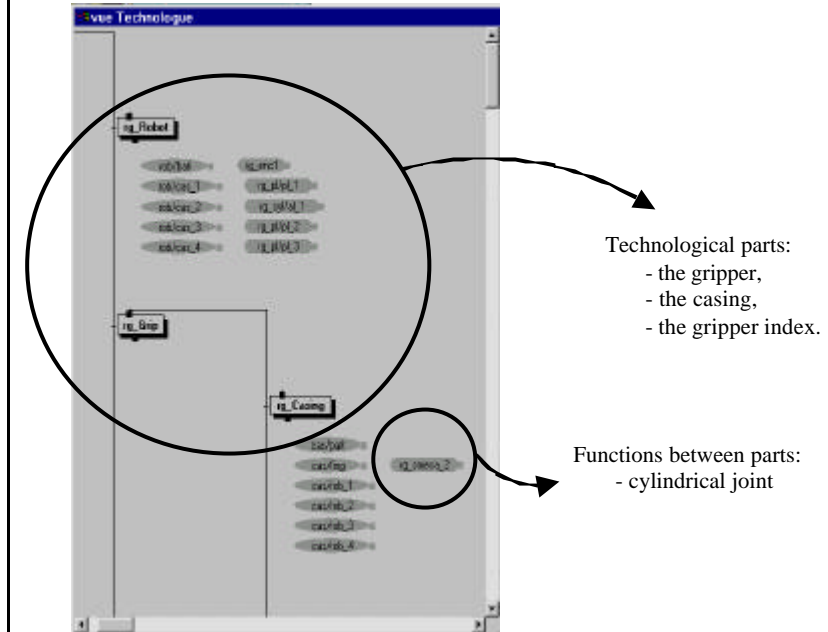
Figure 16: Addition of specific expert's functional surfaces in the *Frame* and *Geometric* views.

Figure 17: Relations and constraints definition.

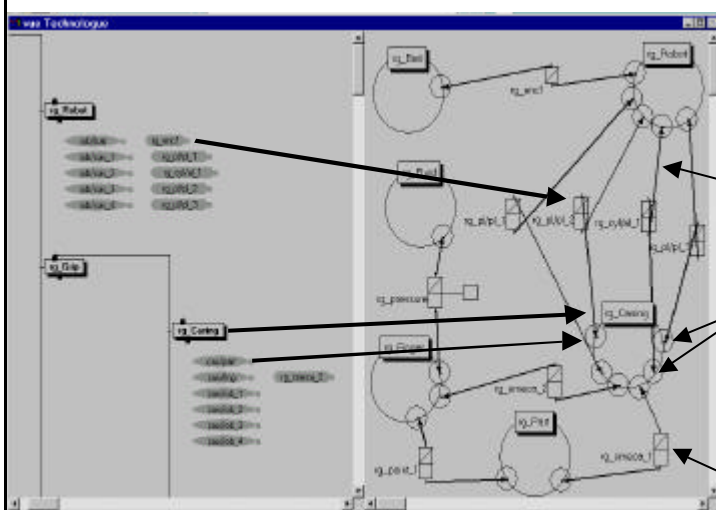
FIGURES



Technological decomposition in the *Technologist* view



Technological decomposition in the *Technologist* view



Product model decomposition in the *Frame* view



Product model for integration

Functional product model

