PERIODICA POLYTECHNICA SER. MECH. ENG. VOL. 39, NO. 2, PP. 81–113 (1995)

IMPLEMENTATION AND UNIFORM MANAGEMENT OF MODELLING ENTITIES IN A MASSIVELY FEATURE-OBJECT ORIENTED ADVANCED CAD ENVIRONMENT

Imre HORVÁTH, Péter KULCSÁR, Zsolt GÁBOR, Zoltán BAGOLY,
Antal HORVÁTH and Vilmos THERNESZ

Institute of Machine Design
Technical University of Budapest
H–1521 Budapest, Hungary

Received: Oct. 8, 1994

Abstract

Today we are spectators of the transition process in computer aided design from traditional geometry based on design systems to advanced computer-based engineering systems. The key is the feature technology that allows both integrating and managing modelling entities in a coherent way. Feature technology is developing rapidly. New research topics and contexts are emerging from time to time. This paper introduces concept, design and technological feature-objects to support operational, structural and morphological modelling of mechanical products. First, the feature-centred approaches to conceptual design are summarized and evaluated. Then an implementation of concept feature-objects and the methodology for using them is presented. The strength of concept feature-objects is in their morphology inclusive nature. They appear as parametrized three-dimensional skeletons providing geometrical representations for the modelled engineering conceptions. A concept feature-object models the physical ports, contact surfaces related to ports, bones between ports, DOF of ports, relevant physical parameters, scientific and empirical descriptions of intentional transformations and environmental effects. Concept feature-objects are related to design feature-objects that, in turn, are constructed of a relevant set of technological feature-entities. Concept feature-objects refer to the configurable and parametrized design feature objects through an indexing mechanism. The conceptions have been tested during the programming and further development of the authors' PRODES system.

Keywords: advanced CAD, feature technology, concept features, design features, technological features, workbench architecture.

Introduction

Traditional CAD systems cannot address those stages of mechanical product design process in that the shape is not known in detail. Nonetheless, some design processes, e. g. assembly design, kinematic simulation, behavioural analysis, would benefit from the functional description of the product model. Although conceptual design can, in principle, develop combined behavioural and morphological models, conceptual modelling is taken as a taboo from the aspect of integration with CAD — mainly because of its concomitant fuzziness. In a conceptual design subsystem, potential solution elements can be characterised by their functions, libraries of physical principles can be related to these functions, libraries of physical embodiments of principles can be maintained and, last but not least, parameters of conceptual entities can be related directly to their geometric counterparts (SHARPE – OH, 1994).

Application of feature paradigm seems to be advantageous in conceptual modelling. However, most of the available feature-based modelling systems consist of a traditional solid modeller and a feature definition shell (SHAH, 1992). Generally, the shape of form features is generated as a macro of elementary geometric entities. Among others, VOELCKER observed that one of the major gaps in the understanding of designs was the lack of means for modelling mechanical functions in a manner that links function to form (VOELCKER, 1988). Traditional feature oriented modelling techniques are not appropriate to cover the requirements of conceptual modelling (SALOMONS, 1994). Probably this is the reason why geometric modelling and physical modelling are disjoint activities in recent CAD systems (RUDE, 1991). In current modelling practice, physical information is tied to the geometric model as textual, attributive or symbolic information in the data base. Data management of this form usually restricts the user to define other than geometry centred feature classes.

The authors have presented a massively object oriented management of feature entities. The specific methodology, that they adapted, is feature-object composition (*Fig. 1*). Implementation of this methodology in an advanced CAD environment also assumes a specific system architecture, that was called workbench. The components of the feature-object-based workbench modeller can be seen in *Fig. 2*. The object oriented management of features, the feature composition methodology and the workbench architecture proved to be essential to implement integrated conceptual and morphological modelling. These have been used in the development of the authors' PRODES mechanical engineering design system. Concepts and implementation details can be found in (HORVÁTH – DOROZSMAI – THERNESZ, 1994), (HORVÁTH – KULCSÁR – THERNESZ, 1994).

A definitive aim of mechanical conceptual design is to allow users to develop design alternatives in the form of virtual or design prototypes. It is also useful to analyze the behaviour of the designed prototypes as early as possible to avoid unexpected and unforeseen malfunctioning. The primary source of malfunctions are non trivial dynamic interactions between components of the system and the time dependent variations in the state of the components of the system. Simulations made in the early stage of design can be based on virtual prototypes that approximate the final manifestation

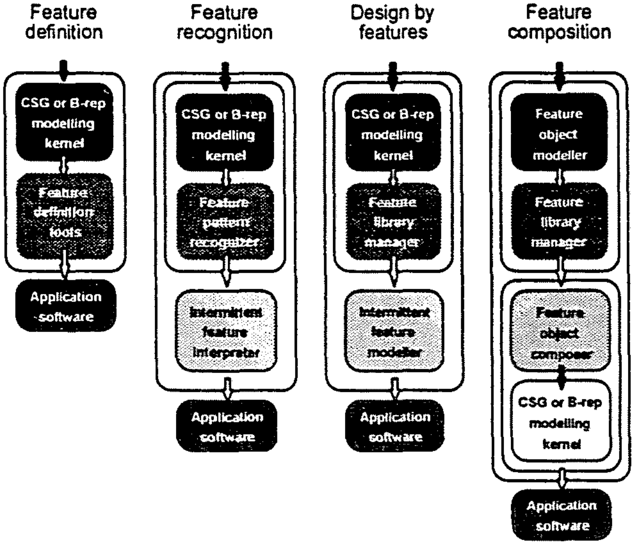


Fig. 1. Comparison of four feature management techniques

of the product. In order to be able to make a thorough investigation, the parameters defining the operation, geometry and the physical attributes should be available in quantitative and/or qualitative forms. Of course, complete behavioural analyses based on the refined geometric model, can provide more reliable information. But simulations with premature models help avoiding unexpected behaviours.

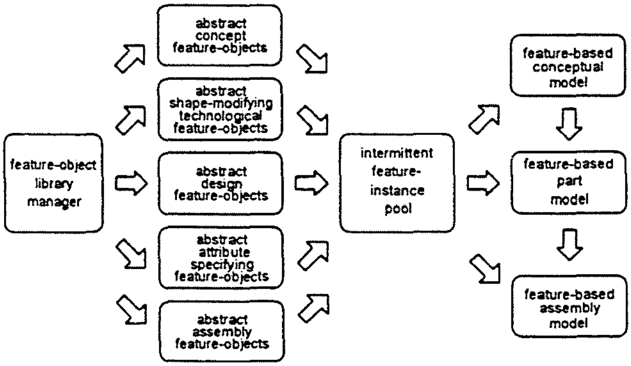


Fig. 2. Components of the workbench modeler

The concept feature-object approach elaborated by the authors is significant since it makes possible to extend feature technology to the early phase of design. Compared with the other approaches presented in Chapter 1, it goes one step further in morphology-inclusive modelling and it provides an object oriented framework for integration with geometric modelling (HORVÁTH – THERNESZ, 1995). To generate geometric models, organ structures are mapped onto a given configuration of design feature-objects. In this object oriented implementation, a design feature-object contains a simple/compound primary shape, and one or more modifying shapes that can be used to model the part or assembly from a geometrical point of view. Explicit description of functionality facilitates selecting entities from the design feature-object libraries. The library of both feature classes can be extended by programming the formerly not existing feature-objects in a high level design language. Later research activities are oriented to develop a quasi-automated general methodology for conversion of concept feature-objects into a topology of design feature-objects.

1. Other Approaches to Conceptual Modelling

There have been several proposals presented for computer support of conceptual design. Of course, it could not be our aim to review all of them. Hereinafter can be found an outline of those that have been judged to represent illustrative and defining approaches. All authors cited below have started out of the fact that both synthesis and analysis deal with the interaction of form and functionality (operation). They proposed methods to integrate the traditionally isolated conceptual, structural modelling and, to some extent, geometric modelling. Different in these approaches are the tools and methods which the authors developed to achieve their aims.

In his early work PAYNTER introduced a network type description of physical systems that was called bond graph (PAYNTER, 1961). The bond graph methodology was initially used for behavioural analysis of systems. Bond graphs, however, can be used for both design and analysis of energy transformations of mechanical systems (THOMA, 1990). Recently, several researchers have used bond graphs as modelling tools for conceptual design problems. This representation tool is advantageous to cover multiple energy domains. In a bond graph representation symbolic features are included that represent idealised physical principles. They form the nodes of a bond graph and they have different number of ports. On the basis of the number of the bonds that are connected to the node, the node can be 1-port, 2-port, and n-port. The behaviour of the nodes and the energy flow are described by principle equations and casual relationships. The mechanical system is characterised by the generalised quantities of efforts and flows on the bonds, connecting the ports. The quantities depend on the physical phenomena, that is the base of the energy transformation. This is important because bond graphs are not able to show the material transformation process which accompanies the changes of the material geometry. A drawback of the use of bond graphs is that they neglect most of the morphological aspects of modelling. Unfortunately, bond graph representations are too abstract to expect their broad acceptance in industrial environment.

In the conceptual design stage, designers need to build initial models that qualitatively represent the principle ideas about the operation of the mechanical product (KIRIYAMA – TOMIYAMA – YOSHIKAWA, 1990). TOMIYAMA et al. introduced a symbolic representation of concepts in the form of physical features that describes the physical phenomena and mechanical elements (KIRIYAMA – TOMIYAMA – YOSHIKAWA, 1992). Physical elements are related through scientific laws and causality. The use of physical features is indirectly motivated by the development of a modelling framework called metamodel. The aim of metamodelling is to relate various aspect models through an intentional model that is based on the qualitative physics and process theory (FORBUS, 1984). First, the designer chooses the appropriate physical features from the libraries and combines them to get the requested structure and operation. It is the task of a qualitative reasoner to find out if the primary model composed of physical features performs the desired operation. The primary model is used as the input for qualitative reasoning to produce the metamodel that correlates various aspect models. Dependency among the formulated concepts of the background theories includes physical laws, empirics, definitions and causalities. They are evaluated when simulation of the operating mechanical system is done. The problem of this methodology is that, in order to be able to reason about the system's behaviour qualitatively, dependencies among the concepts of all physical phenomena that the system knows should be described explicitly. Since physical features are represented by icons or schemes, their morphological fit cannot be directly reasoned out. Furthermore, optimization of physical parameters is difficult since consistent quantitative descriptions are not available for the object.

Having recognised the restrictions of recent CAD systems, HENDERSON and TAYLOR approached to the problem of computer-based conceptual design by introducing a meta-physical product model structure that includes product definition units (PDU) for need, function, physical principle, embodiment and artefact implementation (HENDERSON, – TAYLOR, 1993). Context, PDUs and alternatives are the three primary aspects of a meta-physical model. The authors generalized the feature concept in order to be able to tie the meta-physical product model to physical artefacts.

Besides material features they proposed to use energy and information features. Energy features are of importance in kinematics, fluid mechanics, thermodynamics and electro-mechanics. Information features are important in control theory, signal processing and information processing. The authors demonstrated the applicability of their methodology in redesigning a common mechanical clamp. During embodiment their conceptual entities are replaced by the physical equivalents that are defined based on material form features. It is difficult to reason out how straightforward the application of this symbolism and composition is if the design is not known at all at the beginning of the design process.

PALMER and SHAPIRO proposed a method based on the algebraic topological properties of mechanical objects (PALMER – SHAPIRO, 1993). They decomposed and recomposed physical objects into cells, complexes, chains and operations on them. Such abstract models can support specification, refinement and synthesis of engineering designs. Physical objects are interpreted as sets of quantities distributed in space and time. It implies the ability to characterize and distinguish the value of quantities in different regions of three dimensional spaces. The authors utilized that a cell complex can be decomposed into regions of possible functionality. Chains were used to represent the relations between these regions, and to associate distributed physical quantities with these regions. If the physical elements used to define the chain model represent the behaviour of some particular domain, the chain model both satisfies the design specification and has a physical realization. Obviously, this formal representation is advantageous from the point of view of interrelating various aspects of model generation. Application of the methodology may prove to be difficult if no formal specification exists for the design. Furthermore, it cannot be reasoned out how the methodology could be implemented in a really user friendly way.

SCHULTE et al. applied functional features in searching for solution principles and their proper combinations (SCHULTE – WEBER – STARK, 1990). Components of the product are described in terms of physical effects and, geometrically, of the primary functional faces derived from the operational processes. Simple functional features can represent design functions like energy transportation, storage of information, alteration of speed, torque transmission and so on. Surface pairs or groups, included in a functional feature, automatically result in the separate components of the modelled product due to the assignment of material vectors. However, functional features in themselves are not enough to define an operable design. In order to bridge the gap, earlier ROTH introduced subtle auxiliary function entities (ROTH, 1982). Nevertheless, functionality and morphology are only loosely connected. The morphological characteristics and attributes can be recognised by natural means and can be investigated by scientific methods. The functions, however, are not so definite, they cannot be recognised by scientific means. Consequently, for the sake of higher level computer support or automation of conceptual design, we must re-evaluate the role of functions and functional diagramming in design. It does not mean that hierarchical function descriptions are useless in the early phases of design but there are quite a few other representation schemes that can be taken into consideration simultaneously. The representations provided by functional features are not unambiguous. Functional features are not powerful enough to cover the morphological deviations of the feature instances or the embedding environment. Furthermore, in the case of a new design, it is difficult to find the primary surfaces without the prior synthesis of the physical processes.

Conceptual designs are often to be developed from first principles which goes together with exploring quite a lot of alternative schemes that are physically feasible before going into the details of the best candidate. BRACEWELL and SHARPE have developed a scheme oriented methodology for conceptual design and embodiment (BRACEWELL, – SHARPE, 1994). In their SchemeBuilder system the developers used design context diagrams. The schemes are generated based on this new information structure that covers a set of stored functional embodiments, which may be of either of two forms: means or principles. This information representation allows for the progressive refinement and development of a design from required functions expressed in an abstract way to their eventual chosen physical realization. First, the required functions are looked up in the functional embodiment knowledge base, and then, the system generates all possible distinct complete schemes by combining alternative functional embodiments specified within it. To be able to execute mapping, the system should have links joining components and required subfunctions within the means, and required subfunctions within the principle. The functional embodiment knowledge base is a tree structure of functions, in which energy flow types and data carrier types are progressively defined. Non-hierarchical links help in reusing any function that has already been embodied and to provide more than one function to embody with. This representation scheme is advantageous from the point of view of trade-offs of moving solutions either way in the mechatronic domain. Unfortunately, schemes generated this way do not contain any knowledge about the spatial arrangement of the physical components unless it has been specifically included into the definition of functions. It is, however, practically impossible to cover all possible relations.

2. Formal Definition of Concept Feature-Objects

In mechanical products, there are two basic groups of factors that determine possible functionality and operation: one is the physical effects, the other is the geometry. The physical effects influence manifestation of the object, and the geometry, at the same time, determines its materialisation. Since the morphology and the energy transformation processes are strongly related, the two aspects cannot be separated from each other in conceptual modelling. As it was shown earlier, the problem of morphological descriptions of conceptual modelling entities is somewhat disregarded in the current conceptual modelling methods. Therefore, as a practice oriented alternative we have proposed concept feature-objects (CFO) — (HORVÁTH – DOROZSMAI – THERNESZ, 1994). CFOs capture and model conceptions that are known and used by mechanical engineers during the design process. Concept feature-objects outline the organs (functional entities) of mechanical products that are based on some physical energy transformations. Roughly speaking, they represent both the minimum geometry that is required to consider geometrical aspects and the regularized physical phenomena that form the basis of operation. This way, the information content of a CFO can be formally represented as:

$$\mathrm{CFO} = (\{M\}, \{F\}), \tag{1}$$

where $\{M\}$ is a set of data representing the initial geometry, and $\{F\}$ is a set of data describing the behaviour. From a programming point of view, CFOs are objects that represent the physical reality of an artefact or its particulars. Covering morphological aspects, the concept feature-object-based modelling methodology can adapt to the evolution of the objects being designed. The adaptation assumes the capability of direct mapping of and/or conversions between aspect models.

An initial geometry M can be formally defined as:

$$M = (\{P\}, \{L\}, \{D\}, \{B\}, \{C\}), \tag{2}$$

where: $\{P\}$ means the port definitions, $\{L\}$ means the locations for ports, $\{D\}$ means the degrees of freedom at ports, $\{B\}$ means the geometric entities representing connections amongst ports, and $\{C\}$ means the contact surfaces. The operation oriented part of the modelling of mechanical objects requires the existence of initial geometry, but it also requires the representation of physical principles, operation parameters, and the formal description of scientific relationships. The ultimate aim of operational modelling is evaluation of the inherent functioning and the response of objects for the environmental effects. To this end, both qualitative reasoning and quantitative analysis are to be applied. The quantitative analysis and simulation require that any CFOs should have the following functional description:

$$F = (\{I\}, \{O\}, \{S\}, \{Q\}, \{T\}), \tag{3}$$

where: $\{I\}$ means parameters of input quantities on ports, $\{O\}$ means parameters of output quantities on ports, $\{S\}$ means parameters of substantial (characteristic) quantities of CFOs, $\{Q\}$ means expressions describing contact mechanisms on ports, and $\{T\}$ means expressions describing energy conversions made by CFOs.

A complete geometric representation covers the topological, metric, and attributive information. From a topological point of view, any object is of a specific topological genus. The boundary of a class of objects can be mapped directly onto a sphere. The objects pertaining to the other class can be mapped onto a simple torus or an n-fold torus. This information is important in order to construct the geometric representations for the CFOs with different topological genus. The topological genus influences how the objects can be joined at their input and output ports. In general, the objects that are homeomorphic to a sphere may join to each other as a linear sequence, a tree, or a network through the ports. This class of objects is called branching CFOs. The objects that are homeomorphic to a torus may have an alternative form of joining. They may surround or enclose other objects, therefore, they are called enfolding CFOs. Sometimes specific connections of more than one object result in an enfolding configuration. The enfolding concept feature-objects can be coupled directly or through an arbitrary number of branching type feature-objects. The introduced two possibilities of joining should be reflected by the geometry and the arrangement of the ports for all modelling entities. Graphical representations for branching and enfolding concept feature-objects can be seen in *Fig. 3*.

The concept feature-object-based modelling methodology assumes that mechanical objects can be disintegrated into operable constituents. According to the mechanical engineering terminology, these constituents can be distinguished as parts, groups, units, or subassemblies. They are in specific connection with each other. Those domains that determine the geometrical details of physical connections and, moreover, determine the energy transferring sub-processes are called ports. For the sake of synthesis and analysis, ports themselves should be treated as individual entities. Depending on their role in the energy transfer, ports can be in-ports, mid-ports and out-ports. Mid-ports are needed to describe those situations in which neither external energy inputs nor outputs can be identified. Mid-

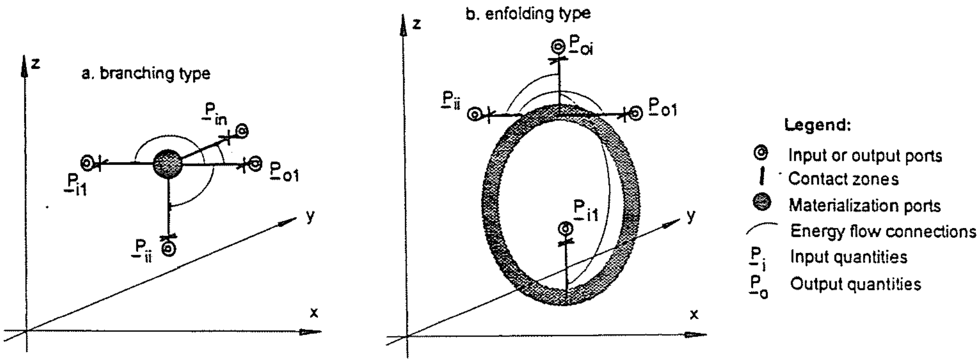


Fig. 3. Graphical representations of concept feature-objects

ports are quite often the materialization centres of mechanical parts. The sum of energy, transferred through the ports, must be the sum of the energy spread over the physical domains in a given moment in different forms.

3. Realization of Concept Feature-Objects

From a programming point of view, predefined concept feature-objects may exist in abstract, concrete, and instance forms respectively (HORVÁTH, – KULCSÁR, 1993). An **abstract concept feature-object** (ACFO) is a complex modelling tool that has an undetermined number of components (i. e. in-ports, out-ports, mid-ports, bones and contact surfaces). Thus, an abstract object is programmed with a default set of eligible modelling entities that can be specified as required. No operational parameters are defined for an ACFO. From this available set of entities, **concrete concept feature-objects** (CCFO) can be derived. These appear as skeletons together with their operational parameters. The components of the concrete feature-objects and their interconnections have already been determined — this is why they can have a fixed collection of functional parameters. Note that concrete feature-objects can also be defined directly through an interactive modelling process. In this state of their existence, the rudimentary dimensional and attributive parameter values can still be modified. Through the assignment of values we get an **instance concept feature-object** (ICFO). Being a pure data representation, it can be modified no longer.

Implementation of geometric modelling with these families of concept feature-objects is a stepwise mapping that allows systematic composition. The relevant ACFO, that is like an executable program, is selected from the library first and then activated. The coded components of an ACFO provide the means for geometric and functional parametrization with constraint management. They have parametrized interface surfaces that facilitate joining them to other components. Through the predefined interface parameters it is possible to merge individual parameter sets of the skeletons into the global parameter network of the organ system. The energy transfer and/or conversions on the skeletons and at the interfacing ports are described by the operational parameters and physical equations. Assignment of the specific values to geometric and functional variables leads to a data base representation of the feature entity and the organ system (HORVÁTH – KULCSÁR, 1993).

4. Creation of Initial Geometries

Application of concept feature-objects in conceptual design of mechanical engineering products presumes interactive use of the conceptual modelling software environment. In a massively feature-object-based product modelling environment, having accomplished this two-phase conceptual modelling, the designer can initiate the embodiment of the product with design and technological feature-objects. It was indicated earlier that an initial geometry was limited to capture those metric features of the product that were necessary for preliminary functional analysis and morphological evaluations. The minimum geometry varies for different objects. Therefore, the method and tools applied have to be flexible and, at the same time, comprehensive enough to cover the needs of various design cases. Furthermore, they have to be in harmony with the modelling methods and entities offered by present commercialised CAD/CAE and FEM systems. Domains of a solid geometry that have significance from a functional point of view, can be satisfactorily represented and, therefore, substituted by low level geometric entities, e. g. surface patches and wires. Thus, contact surfaces are described by surface patches and the energy transferring domains are idealized as wires. This results in a wire-frame-like model, but in our concept feature-object-based modelling its components have specific notations and graphical representation, respectively.

The ultimate aim of conceptual modelling is to generate the organ structure of the mechanical product. An organ structure represents the functional components, operational connections of components and the structural topology. The components are instances of concept feature-

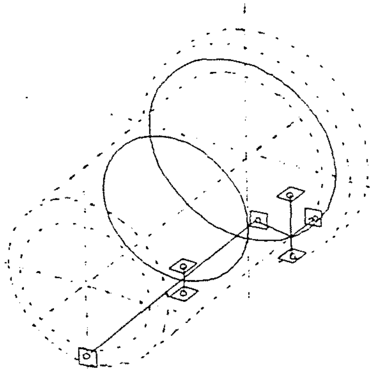


Fig. 4. Skeleton of the bushing

objects. The initial geometry belonging to these components of the organ structure is called the skeleton. Thus an organ is modelled jointly by a skeleton geometry and a physical description. The bushing shown in *Fig. 4* consists of two components since the original part can be separated into two functional units, namely the bush and the shoulder.

The elements of a skeleton are denoted as bones. From a functional point of view, bones specify physical locations of ports in the metric modelling space. Eventually, bones are for connecting the ports according to the logic of the presumed energy transfer and morphological changes. Bones are brought together to define the possible ways and directions of energy flows. In a branching skeleton, bones may form a chain, star or tree. For enfolding skeletons there is at least one bone that forms a loop. Note that the skeletons included in the organ structure are not necessarily physical parts. Depending on the application, a skeleton may represent a group of parts, an individual part, or a domain of a part.

Ports are to be defined by specifying their three dimensional Descartian system of coordinates and kinematic degrees of freedom at connections of ports. The specification has to cover both macro- and micro-geometric respects. From a shape modelling point of view, ports are specific regions of the boundary of the modelled product. Thus they can be identical to a given surface, a part of it, or may even encompass a given configuration of surfaces. If the requested analysis makes it important, contact surfaces can be modelled geometrically as analytical or parametrical surfaces. If the flow of energy through a mechanical part or unit can be identified, identification of in-ports and out-ports is trivial. Mid-ports can be regarded as nodes connecting the bones. Generally, mid-ports are those material centers of mechanical parts where the presumed energy flow ramifies. Thus, a mechanical part may have one or more mid-ports. Functional quantities are always unchanged for a given part geometry.

In order to model the whole organ structure morphologically, skeletons are to be connected. Only those ports of the skeletons which have the same parameters can be merged. Modelling of initial geometry thus includes the following steps: (*a*) definition of ports, (*b*) locating ports in the model-space, (*c*) arrangement of bones, (*d*) specification of degrees of freedom on ports, (*e*) connecting skeletons to form organ structure, and (*f*) decomposition of the organ structure into manufacturable and assemblable physical parts. Initial geometric modelling is to be executed in the given order. An example for the graphical representation of an initial geometry of a Cardanic joint subassembly can be seen in *Fig. 5*. This is a window captured from the PRODES system.

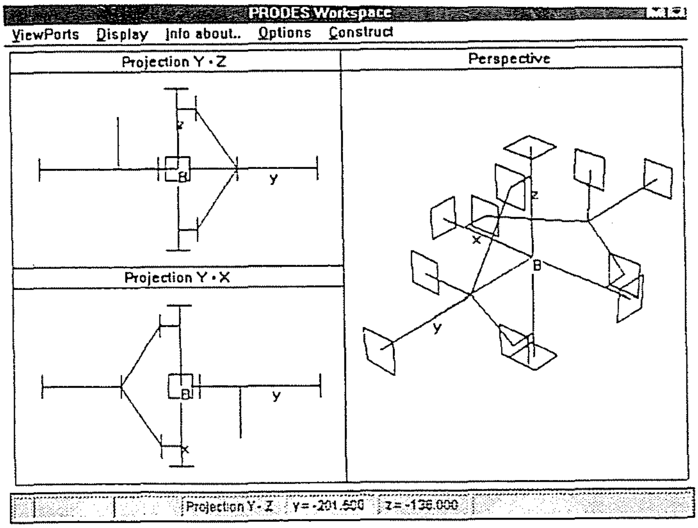


Fig. 5. Graphical representation of the initial geometry of a Cardanic joint

For methodological reasons, a design activity agenda is built into the interactive conceptual design subsystem. Obviously, steps can be repeated in order to modify the results of any previous specification. Because reconsiderations and modifications may be requested in the interactive conceptual modelling process, graphical representation can assist in recognizing morphological inconsistencies. In order to avoid mismatches between ports characterized by different parameters, an advance specification of input and output parameters is expedient.

Until now, the initial geometry for the organ structure has been defined as a static one. It is, however, very rare that no variation of the initial geometry takes place during the now supposed operation. Geometric changes of the initial geometry should be simulated. The observable behaviour of the mechanical system can only be modelled by taking operational changes into consideration. Eventually, together with modelling of the environmental effects, it provides a basis for behavioural simulation. There are several forms of geometric changes. They are consequences of variations in effects and loading during the operation, that will be discussed later. One group of modifications is caused by static operation. These, among others, are the following: (*a*) deformations due to loads that influence operation (e. g. buckling deflection), (*b*) initial distortions due to pretightening (e. g. setting springs), (*e*) extensions due to heat (e. g. longitudinal elongations). Another group is kinematic or dynamic modifications. These intended alterations come from: (*a*) positional variations (e. g. revolving excenter cam), (*b*) locational variations (e. g. in hydraulic cylinders), (*c*) geometrical reconfigurations (e. g. in arretating clutches), (*d*) functional repositioning (e. g. in elastohydrodynamic slide bearings), and (*e*) meshed connections (e. g. threads). The third group is functional variations that include both static and dynamic (kinematic) alterations. Furthermore, those that have their origin in some stochastic effects of the environment. Consider, for instance, alterations due to (*a*) shocking, (*b*) wear and (*c*) impact.

The last step of initial geometry definition is allocation of components into manufacturable (assemblable) physical parts. It is an implementation oriented decomposition of the organ structure. The reason is that the skeletons included in the organ structure do not necessarily result in optimally embodied parts. Sometimes combining the skeletons into complexes, or the opposite, separating them into less complex fragments may be advantageous from a functional, manufacturing, and economic point of view. This further processing can be made taking into consideration the principles of design for manufacture and assembly (DFMA). The process of embodiment is rather intuitive, even if it is executed with design features, and it supposes considerable manufacturing experience. There are three specific rules that are to be kept in mind in order to get parts with optimum final geometries: (*a*) Composition or separation may not devaluate functionality; (*b*) There must be a compromise between the number of parts and the complexity of parts from manufacture, assembly, maintenance and reparability points of view; (*c*) Secondary design aspects, e. g.

economy, aesthetics, reliability, environment detection cannot be treated as secondary questions to develop competitive products.

5. Quantitative Performance Analysis

Early performance evaluation is indisputably necessary for improving the technical merit of products. There are two approaches to this analysis: (*a*) by qualitative reasoning and (*b*) quantitative evaluation. The ideal would be to have a hybrid approach, however, it is not readily available yet. Qualitative reasoning is oriented to reason with process characteristics, but it has nothing to do with optimization of operational parameters. Quantitative evaluation covers numerical modelling of the product and the environment, and is oriented towards investigations of the operating states. A quantitative analysis needs a consistent specification of functional parameters and (non)algebraic expressions relating them.

A class of parts of mechanical engineering products needs to be evaluated for static performance only. The other class of parts requires evaluation from a dynamic (kinematic) and a time dependent functional point of view. The most comprehensive form of performance evaluation is time-inclusive behavioural simulation that requires modelling of the product as a system and the environment together. However, performance modelling and behavioural simulation should be separated and should be executed distinctly, since they require significantly different mathematical methods and tools.

Whichever form of evaluation is accomplished we have to consider varying morphological descriptions. This aspect has been discussed earlier in detail. In the case of behavioural simulation both static and kinematic geometry modifications should be taken into account. During the simulation it is advantageous to have the parametric descriptions of initial geometries of skeletons included in the organ structure and the contact surfaces at ports. In fact, even performance evaluation cannot be made without providing facilities for dynamic management of skeletons' geometries. Nonetheless, performance evaluation requires a complete description of the physical phenomena and their interactions.

In the course of static performance analyses, numeric values have to be defined or computed for port variables. The skeletons are to be further concretized by assigning attribute variables. In order to make clear-cut distinction between interfacing and attribute variables, *Fig. 6* presents an example of a V-belt pulley. Here, input energy variables are circumferential force from the key and angular velocity, output energy variables are force in the V-belt and tangential velocity of the V-belt. Attribute variables are

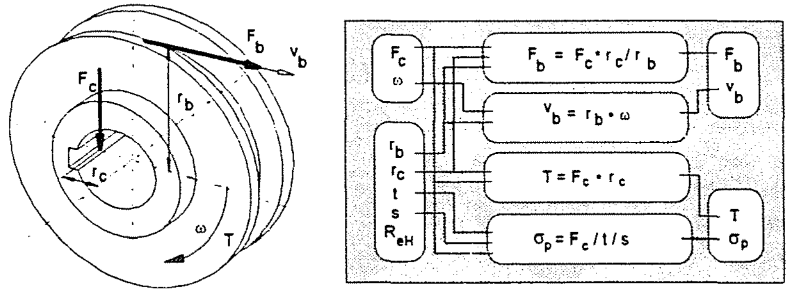
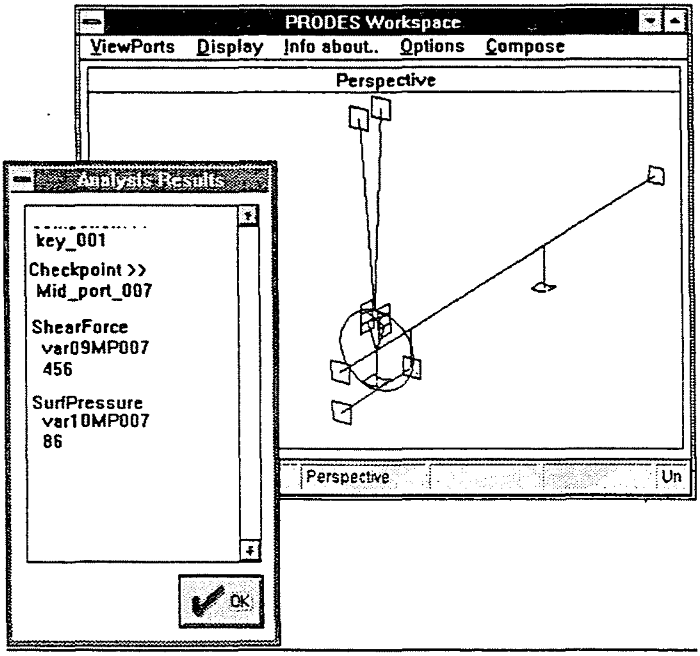


Fig. 6. Example for operational parameters and expressions of a V-belt pulley

those that describe the geometry, material and other physical characteristics of the pulley. These are the hub bore diameter, the distance of the key face center from the axis of rotation, the distance of the V-belt contact circle from the axis of rotation, and so on. There are, however, some other of parameters of different type that have not been mentioned up till now. They can be called latent parameters. These also need to be defined, but their values are needed not to be given prior to or during the simulations. The reason for this is that they can be computed and used in other calculations. For a V-belt pulley these variables are, for instance, shear stress due to torsion and accumulated strain energy. In other cases potential energy and energy of movement are latent variables.

The interfacing (input and output) variables for ports are defined in the geometry definition phase. In the overwhelming majority of cases it is true that interfacing parameters are closely related to primary forms of energy (e. g. mechanical, thermal, electrical), whereas latent parameters are related to secondary forms of energy (e. g. strain, potential, damping). Since it is practically impossible to specify in advance all parameters that might be ever needed in behavioural simulation, contextually named variables cannot be joined to parameters. In this case, variable names can be specified as identifiers to support users. For each variable, a memory address and space are attached through memory pointers. Computing the performance states is done by evaluation of explicit mathematical expressions. By mathematical expression we mean simple equations, sets of linear or nonlinear equations, differential equations and even straightforward numeric algorithms. In some cases we also need inequality expressions or recursive algorithms. Having specified the relevant interfacing and attribute parameters, we can construct the functional expressions by an equation editor. The concept of an algebraic expression editor is well known from,



```
key_001
Checkpoint >>
 Mid_port_007
ShearForce
 var09MP007
 456
SurfPressure
 var10MP007
 86
```

Fig. 7. The initial geometry and analysis results

for instance, spread-sheet applications. Solutions for the functional expressions are sought for by the equation solver program of the PRODES system. Note that nowadays similar software tools are already commercialized. For a shaft-key-V-belt pulley torque transferring subassembly some results of a static performance analysis can be seen in *Fig. 7.* The results in the child-window have been received by interrogating the mid-port defined in the center of the key.

6. Time-Dependent Simulations

The time-dependent behaviour of an organ structure can be evaluated by incremental simulation. This happens in four consecutive steps: (*a*) setting or modifying the initial geometry, (*b*) setting or modifying the environmental effects, (*c*) computing and evaluating discrete operational states of the organ structure, and (*d*) representing the operating state graphically.

Physical phenomena can be made explicit by constructing time-dependent technical expressions that control the energy transformations and transfers. For the complete description of processes, domains of legal parameter values and constraints on the operational parameters are to be specified as well. Simulation assumes time-dependent representation of the organ structure and the included components. Therefore, positional and operational states of the organ structure have to be described with a view to time variables. Depending on the logic of operation, several time variables are to be taken into consideration. Assigning discrete time values to time variables and computing the relevant states result in an incremental simulation. Encompassing possible variations of the effects coming from or caused by the environment lends itself to a behavioural simulation based on the organ structure. For behavioural simulation, descriptive equations should be arranged to form numerical procedures. Furthermore, constraint management, truth maintenance and conflict resolution may be required to solve complex cases. Causal relationships more or less reflect the logic of operation. Therefore, to produce reliable results, causalities must be strongly taken into consideration.

Resulting from the former theoretical considerations, the simulation phase covers the following activities: (*a*) specification of attribute and latent parameters (variables) for the skeletons, (*b*) description of contact mechanisms on the ports with expressions relevant for functions, (*c*) definition of functional relationships and ordering the numerical expressions based on causalities, (*d*) inclusion of time variables into functional relationships (if pertinent), (*e*) modelling the environmental effects and their variations, (*f*) operational and/or behavioural simulation and evaluation of the organ structure, (*g*) executing combinatorial, functional and morphological modifications to optimize the conceptual model of the design.

For time-dependent analysis of the constructed organ systems, the conceptual modeller of the PRODES system adapted timed neural type Petri nets with explicit transition representation (ZARGHAM – TYMAN, 1985). This approach has been found advantageous in representing both the structure and behaviour of mechanical organ systems. It should be mentioned that traditional Petri nets, however, lack at least two things that are important either in structural or operational modelling. First, they do not show initial modularity which is advantageous for the discrete description and modelling of organs included. Second, they do not consider time that is important for description of process flow and logistics. Application of neural Petri net model is a solution for both issues. After some abstractions, the material, energy and information transfer and conversion of concept feature entities can be represented by a specific neural Petri net. Connecting the inputs and outputs of the partial nets results in a configuration that corresponds to the completed organ system.

Formal definition of the neural Petri net (NPN) used in the simulation component of the PRODES system is as follows:

$$NPN = (P, T, A, S, F, D, N, M, V, W, Q, C),$$

where

P is a finite non-empty set of distinctly labelled places ($P1$, $P2$, ... Pn),

T is a finite non-empty set of distinctly labelled transitions ($T1$, $T2$, ... Tn),

A is a relation that is represented by an arc either from a place to a transition, or from a transition to a place,

S is a finite non-empty set of initial places that are starting points of the material, energy or information processing on a skeleton,

F is a finite non-empty set of final places in which the output quantities of a skeleton appear,

D is a real number which indicates the duration of a physical effect (token) in a place,

N is a real number which indicates the (minimum) threshold value of tokens required to fire a transition,

M is a real number which indicates the (maximum) limit value of tokens allowed to appear during a transition,

V is a set of equations that calculates the total token value in a place at every unit of time,

W is a set of functions that calculates the changes of the parameters that characterize the places,

Q is a function that associates an initial state qualifier (colour) with each output arc of a transition,

C is a set of colours.

Each NPN constitutes a segment of the complete NPN of an organ system. This computer internal representation allows us to construct NPNs for organ systems automatically based on the skeleton structure and to evaluate the whole system interactively. In formal evaluation model the set of places represents the set of geometrically defined ports, transitions are physical processes to and from a port. Each transition may have more than one input places that correspond to the possible multiple interactions of mechanical parts. Similarly, a transition may have more than one output places. A relation can be one of forward pointing, backward pointing and back referencing. Threshold values are needed to express that certain effects simply cannot be exerted if the physical quantities (e. g. hoisting force, temperature) are below a value, and limit values express that certain quantities (e. g. stress, deflection) must not exceed a given value. Qualifiers are needed to provide extra qualitative information on transitions. During incremental simulations of the behaviour, the assigned value of lifetime D of the effect can be decremented by one. If the effect is getting stronger or weaker in time, the qualifier tells how it takes place and the limit and the threshold values, respectively, can be checked for. Colouring allows us to express information, for instance, on the assumed initial state of the system by considering values, sign of the values (positive or negative), or intended directions.

7. Definition and Representation of Design Feature-Objects

One of the aims of the feature technology is to enable designers to compose mechanical designs from features that embody functional and geometrical properties and convey explicit information for manufacturing, assembly and measurement. The use of feature-objects assumes specific modelling methodology. This methodology is the feature-object composition that produces the geometric model of the product as an aggregation of the predefined design feature-objects (DFO). In this methodology, design feature-objects are the tools of capturing functional and morphological information and they transfer it to the product data base. Since both functional and morphological information is strongly product- and part-dependent, all embracing definition of the DFOs in the practice is not feasible. There is no obstacle, however, to define DFOs in advance, if the scope and content of the application are sufficiently known or predictable. Our PRODES system, which is intended to be a prototype of a completely feature-object-based advanced CAD system, makes use of this possibility at the generation of the DFO libraries. Taking into consideration the necessity of extending the libraries, the capability of generating new modelling entities has been provided. Designing with highly alterable DFOs does not restrict the realisation of creativity. A possible, but previously not existed arrangement of design feature-objects may result in innovative designs.

Unfortunately, the definition of feature-objects cannot be exhaustive even for a given application field, since form features are also formed when the part itself is embodied with design feature-objects. Note that these form features are interesting from the point of view of manufacturing, rather than functional and/or morphological design. It implies that the undefined form features should be related to one or more relevant technological feature-object. There exits a fairly large number of DFOs that come out with weak engagement to the application. These entities are mostly simple standard parts or their stereotype segments. Most of the simple DFOs have direct correspondence to a technological feature-object. A complex DFO, in general, can be decomposed into two or more shape modifier technological feature-object superposed on each other. In certain cases the appearance of a DFO is governed by styling considerations, but connections to manufacturing are also evident. Anyhow, the feature-object composition methodology provides the means to relate the initially non-defined form features to shape modifying technological feature-objects.

Considering the facts mentioned above, we may conclude that design feature-objects are to be specified according to the actual demands of the specific application. They can be constructed by considering previously developed designs. In order to be consistent with the included set of conceptual feature-objects, one or more DFO has to be defined to each CFO. The more significant the overlapping between a CFO and a DFO from functional point of view, the easier is the embodiment. In the PRODES system there has been a two-way referencing among CFOs and DFOs implemented by the use of library pointers.

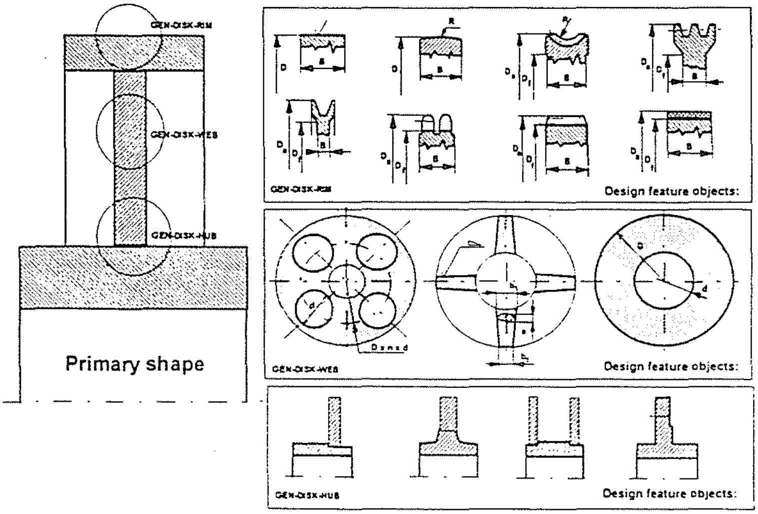


Fig. 8. Example of a complex abstract design feature-object

From the aspect of the information content and the purpose of formation, a design feature-object can exist in three forms: abstract object, concrete object and instance object, respectively. An **abstract design feature-object** (ADFO) is a specific shape model with varying structural topology. From the implementation point of view, an abstract object is a collection of a simple or compound primary shape (shape carrier), and one or more modifying shapes (shape modifiers), by which the designer can alter the geometry of the shape carrier. An example for a compound design feature-object is shown in *Fig. 8.* The modifying shapes, in themselves, represent simple design feature-entities. By the use of these feature-entities the designer can effectively modify the primary shape to provide the needed functionality. This solution needs less time than the traditional way of feature manipulation and is really powerful if a comprehensive library of design feature-objects belonging to the given application domain is available. From programming point of view, abstract design feature-objects are again executable programs that communicate with the user and the geometric modelling workbench of the PRODES system. ADFOs represent dynamic designs in which their primary and secondary shapes can be combined systematically. Furthermore, the set of the included entities can be extended or modified. Thus, the complexity of the programmable ADFOs is not limited.

By selecting the relevant modifying shapes and merging them with the primary shapes, the designer specifies an object of fixed descriptive topology. This object is called **concrete design feature-object** (CDFO). Besides the topology, the geometry of a CDFO is also specified, but the geometry is parametrized. Dimensions and attributes can be modified. Having specified the size parameter values and the attributes, the designer can transfer a CDFO into a special B-rep data structure. This provides an **instance design feature-object** (IDFO). Being a pure data model, an IDFO can be modified no longer. Introduction of the CDFOs is reasonable from modelling point of view, since in this state of model forming the components are only formally combined but not merged on B-rep data level. Instead, a preliminary aggregation of components, say design prototype, is generated. The history of the model forming process is recorded in a binary tree and the B-rep data segments, describing the components, are stored in the memory together with some pieces of information on their origin. If the designer wants to modify something, it can be more easily done with the prototype than with the final model. This modelling concept provides high level of flexibility for an advanced CAD system.

Abstract design feature-objects themselves, however, should be well designed in order to facilitate their comfortable use for designers. The number of the modifying shapes must be large enough to provide the required flexibility in modelling. Parametrization of the modelling entities has to be consistent and the parameters clearly arranged. In general, the designer uses the primary shape and the modifying feature-entities included in a given (programmed) ADFO. In certain cases, in order to cope with the versatility of the designs, it might be necessary to use a modifying shape that has been defined in another ADFO. Thus the modifying shapes of any abstract feature-objects should be activatable individually. Therefore, in the programmed abstract feature-objects the components are independent of each other. Very often the modifying entities are concrete technological feature entities.

Another question is the unification of the modelling entities in the model space of the workbench modeller. The entities can be combined by the method that is known from traditional CSG modellers. These systems, in general, work on closed and regular objects. We have to face some difficulties, however, if we want to adapt this technique for the feature entities. The problem originates in that the modifying features are not geometrically complete objects. They are certain aggregation of faces that forms an open shell. Therefore, the intersection computation should be face oriented rather than solid oriented. It requires extra computation to determine validity of the intermediate geometries resulted by all individual steps of the combination and closedness of the final part model. In some irregular cases, even additional transition surfaces are to be generated to ensure closedness.

A class of parts of mechanical engineering products needs to be evaluated for static performance only. The other class of parts requires evaluation from a dynamic (kinematic) and a time-dependent functional point of views. The most comprehensive form of performance evaluation is time-inclusive behavioural simulation that requires modelling of the product as a system and the environment together. However, performance modelling and behavioural simulation should be separated and should be executed distinctly, since they require significantly different mathematical methods and tools.

Technological features or manufacturing features have been defined in several ways earlier. According to the terminology of the PRODES system, technological features are those entities that are needed to describe the final geometrical state of the product supposing certain fabrication processes. The specific taxonomy of the available technological feature-objects is shown in *Fig. 9* and *11.* There are two general classes of TFOs: one is the class of shape modifiers and the other class is the attribute specifiers. Shape modifier TFOs are supposed to produce a given change relative to the original shape of the DFO (*Fig. 10*). The modification is determined by both the intent of application and the manufacturing facilities that can be used to execute the required manufacturing operation. The attribute specifier TFOs attach attributes that appear when a specific manufacturing operation is supposed to have been applied to the model or its components (*Fig. 11*). It is important that both technological feature-object classes refer to the final state of the parts.

Definition and implementation of shape modifier technological feature-objects are consistent with those of design feature-objects. Their generic form is **an abstract technological feature-object** (ATFO). By the use of an ATFO, designers may specify strictly manufacturing aspects of the ADFOs. In the programmed ATFO, a type specification of a pri-

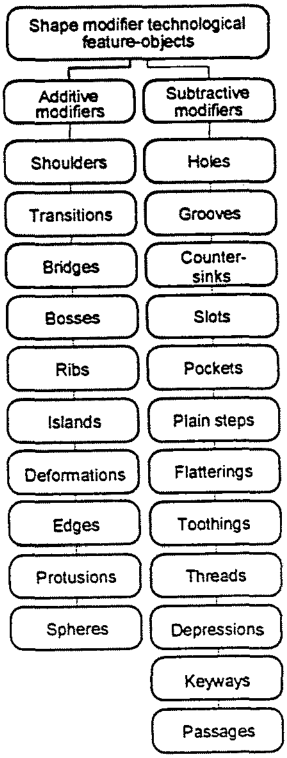


Fig. 9. Shape modifying TFOs

mary shape and several modifier shapes can be found (*Fig. 10*). In general, TFOs inherit their primary shapes. This shape may be the primary shape of the abstract design feature-object, or one of the modifying shapes. In the PRODES system the referencing of DFOs to TFOs is implemented through pointers that are handled by the library managers. It is possible to build certain shape modifier TFOs onto each other. This way a hierarchy of the TFOs can be implemented.

The components included in an abstract technological feature-object are parametrized with respect to the geometry and the machining operations. The geometrical parameters describe the shape modification, resulting from the execution of the assumed machining operation. The other set of data specifies the applicable machine tools or forming equipment, and the relevant tools, based on the dimension values specified for the geometry. These data provide explicit information for the manufacturing process planning.

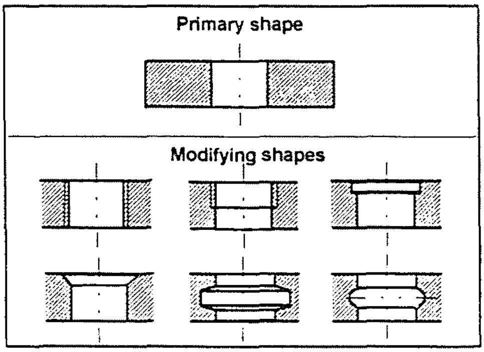


Fig. 10. Example for a TFO for shape modification

A programmed abstract feature-object for shape modification offers a choice of additive or subtractive modelling entities. By selecting and combining the appropriate shape modifiers, the designer is to specify the relevant parameter values. This way a **concrete technological feature-object** (CTFO) can be generated. After having specified the numerical values and the text-like attributes, the system converts the information package into a B-rep data structure. This produces again an **instance technological feature-object** (ITFO). The data structures of the technological feature-objects are organised similarly to that of abstract and concrete design features. However, they contain both numeric and qualitative data for the production equipment and operations.

The attribute specifier technological feature-objects support the designer in joining the specification of the required attributes to certain modelling entities. The attribute specifier TFOs may relate to the global manufacturing process (cutting, forging, casting, bending) of the part or to the specific processes and treatments (welding, hardening, polishing, knurl) applied on the part(s) concerned. The information conveyed by the attribute specifier TFOs may regard to a face of the part (e. g. surface roughness, size and shape tolerances, surface hardness), to a face group of the part (e. g. position tolerances, heat treatment specifications), to the part itself (material, assembly specification, patterns) and to a group of parts (e. g. assembly specification, functional specification). In contrast to shape modifier TFOs, that are executable programs, attribute specifier TFOs are implemented as operators. They can be invoked in the geometric modeller workbench of the PRODES system.

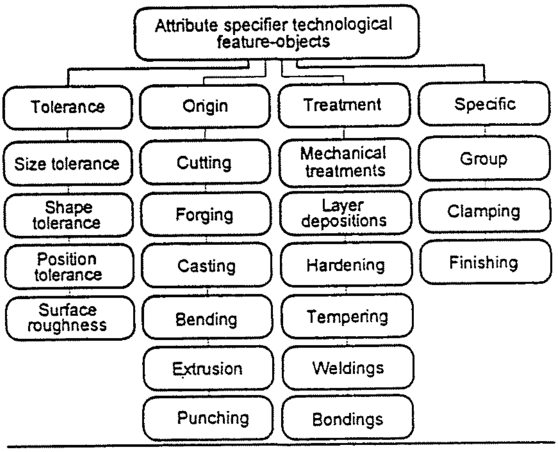


Fig. 11. Attributes by technological feature-objects

8. Coherent Management of Feature Entities

From a methodological point of view, the feature-object-based modelling process in principle can be divided into six phases that have different orientations. During the first phase, concept feature-objects themselves are to be derived and instances generated. Then the initial geometry of the organ system for the whole product is to be generated. The initial geometry considers spatial locations, geometrical extensions, external connections and intended global shape. The third phase focuses on quantitative characterization of the mechanical product based on the available initial geometry. It also specifies the physical effects on the components and allows studying and simulations of the behaviour of the whole organ system or its constituents. The fourth phase is the mapping of the decomposed organ system onto relevant abstract design feature-objects. In this phase, the system searches for potential solutions in the design feature-object libraries. The user is to select the best matching cases. During the fifth phase, the designer produces those domains of the parts that cannot be covered by features by traditional geometric modelling. These domains can also be defined as features and stored in library. The last phase is for generating rendered images of the part or assembly.

As it can be concluded from the interpretation given above, the morphological design can be supported by the preceding use of concept feature-objects to a great extent. Their arrangement projects to the possible configuration of the design feature-objects and influences the possible embodiment of parts. In fact, the intentions of the designer, the morphological consistency and the optimizing constraints determine the applicable DFOs. In an advanced CAD system, conceptual feature-objects should have one or more equivalents among the abstract design feature-objects. The design based on any organ system is feasible only if all of the included CFOs have been put together properly, i. e. their parameter values have been selected and determined with the view to the existing physical constraints of the available design feature-object. In connection with the conceptual feature-objects the most important consideration is that they are assumed to be mapable onto one or more abstract (or concrete) design feature-objects directly. This mapping lends itself to the integration of operational, structural and morphological syntheses. With the extension of the feature-object paradigm for conceptual and geometry design, the mapping process, that is rather intuitive otherwise, can be formalised and effectively supported by a computer.

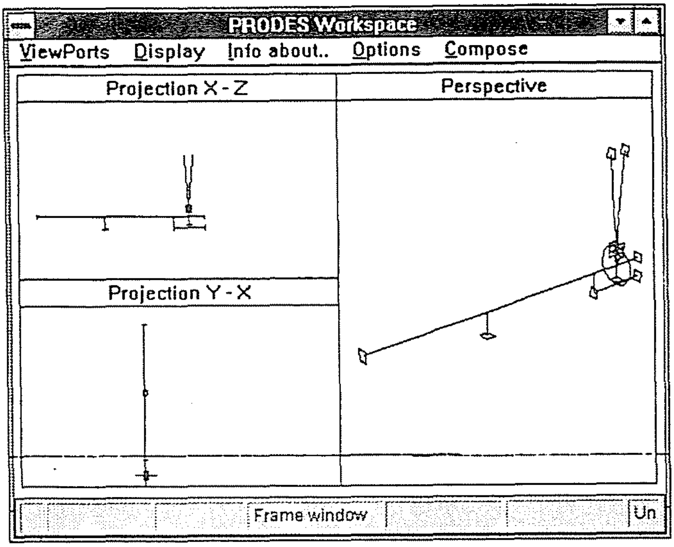


Fig. 12. Composition with concept feature entities

Fig. 12 gives an example of an initial geometry for a simple mechanical assembly. It is composed from the skeletons of a shaft-end, a key and a V-belt pulley. Here all skeletons have been generated by using an abstract concept feature-object. This approach is most advantageous if the results of the preliminary morphological modelling and behavioural simulations can be directly downloaded to a geometric modelling environment in which the embodiment is also based on design feature-objects. Nevertheless, the organ structures developed by the conceptual modelling methodology facilitate geometric modelling of part and assembly even with traditional CSG or B-rep systems. The unified approaches and consistent methodologies, however, can make massively feature-object-based advance CAD systems superior to conventional CAD systems.

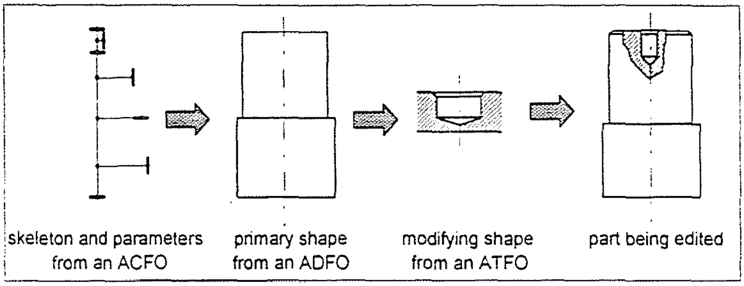
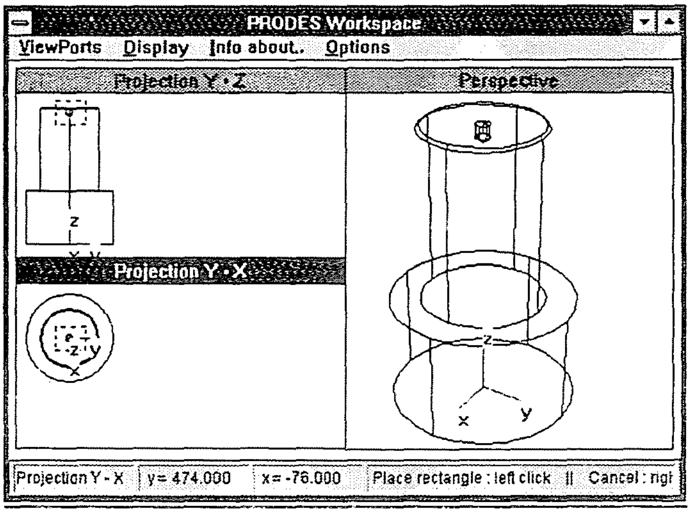


Fig. 13. Mapping with feature entities of various aspects

Fig. 13 presents the steps of mapping the skeleton of the shaft-end with centre hole onto applicable design and technological feature-objects. The geometric model can be seen in *Fig. 14*. This example is rather simple and, therefore, self-explanatory. In the more complex cases, however, there are five important issues to be solved at converting organ structures into morphological complete models. They are as follows:

a) *Domains of Different Structural Levels*

In general, an organ system comprises functional units of different structural levels. For instance, an electric motor or a hydraulic cylinder are often embedded into an organ structure that is made of part-level or segment-level entities. At the beginning of mapping the organ system into a morphological model, the task is to recognize the domains of different structural levels that have been built into the organ system, in order to be able to find the matching embodiments. It is to be done based on the cognitive knowledge of the user.



Projection Y - X | y= 474.000 | x= -76.000 | Place rectangle : left click || Cancel : rigt

Fig. 14. Geometric model by feature composition

b) Multiple Allocation Alternatives

Different domains of the organ system can be allocated into parts in several alternative ways. The intuitive decomposition, that is neither unique nor unambiguous, greatly influences the applicable set of design feature entities. If they have been defined accordingly, certain C.F.O.s can be related to D.F.O.s directly. Due to the use of shape modifying feature-entities, this is not always the case. Consequently, when a decomposition is made, the designer should be aware of what sort of design feature entities are available in design and technological feature-object libraries.

c) Overlapping of Design Feature Entities

Very often, more than one design feature entity provides the same subset of functions, which results in a kind of overlapping from the point of view of embodiment. It is in connection with part formation, which is influenced by design for manufacturability and assemblability considerations. Notwithstanding that the organ structure reflects continuous effort and flow parameter conversions, this overlapping may result in a situation that is to be resolved by user interaction.

d) Design Constraint Management

The concept feature-objects and the design feature-objects can be related directly only in their abstract forms. On the other hand, the organ system contains specific instances of A.C.F.O.s. The matching needs design constraint management both in symbolic and algebraic form. Library managers of D.F.O.s must be able not only to look up, but to administer the origin of the instances in order to provide facilities for re-evaluation of parameter values for morphological fitting. Design constraints would form a constraint network, which is practically impossible to construct, except simple cases, since the modelling entities included in the design cannot be foreseen.

e) Combination of Non-regularized Modelling Entities

The design feature geometries are combined based on set theoretical approach known from CSG modelling, but in this case unifying operators are applied to faces and groups of faces. This methodology requires modelling tools that are somewhat different to those known from commercialised CAD systems.

Considering the facts mentioned above, we may conclude that a practical solution is to implement the feature-object oriented design methodology in an interactive modelling environment. Providing an interactive methodology is a pragmatic approach, but it can be integrated with present CAD systems and it may be expected to be accepted by industrial users. In the PRODES system designers use high level modelling entities both in conceptual and morphological design. It is advantageous from the point of view of early synthesis and simulation, and assembly oriented geometric design.

There are works in progress in order to integrate assembly features to support functional reasoning with assembly models. The novelty of our approach lies in the parametrized description of assembly connections. Eight types of assembly features have been described formally: slide, roll, rotate, fit, against, contact, tilt and fixed. A feature description covers the shape of the contact surface patches, the possible degrees of freedom in the neighbourhood of the contact point, the physical effects acting at the contact point (static and dynamic forces and moments, impulse). Every functional description refers to the local system of co-ordinates of the part or unit. Having the geometrical models of components, the answer to the assumed contact effects can be concluded. Assembly of features has parametric description which, in turn, can be advantageously utilised for the modification of the parametrized geometric model.

9. Summary and Conclusions

Our paper presented an interactive platform to conceptual and morphological modelling. The feature oriented conceptual designer software is a proof-of-ideas subsystem that has been included into the PRODES host system. It has been implemented on an ACERFRAME workstation-like PC under WINDOWS 3 operating environment in C++ object oriented programming language. It was set out to provide an easy-to-use and user friendly interface based on window management. Windows with input and check out facilities and with graphical presentation fields were necessary in order to achieve the required highly interactive environment.
Our conclusions are as follows:

- Feature oriented conceptual modelling technique allows two-level reasoning, thus both design language based circumscriptions and morphology inclusive model generations can be considered in conceptual modelling systems.
- Morphological design can be supported by the preceding use of a concept feature-object-based preliminary design and simulation.
- The development of an interactive conceptual modelling subsystem is a pragmatic approach, but it is the most straightforward approach to extending the capabilities of present geometry and assembly oriented CAD systems.

Acknowledgement

The authors are also very grateful to the other members of the PRODES team: Károly Dorozsmai, Dr. László Horváth and Tamás Major.

References

BRACEWELL, R. – SHARPE, J.: A Computer Aided Methodology for the Development of Conceptual Schemes for Mixed Energy-Transforming and Real-Time Information Systems; *Computer Aided Conceptual Design, 1994 Lancaster International Workshop on Engineering Design CACD '94,* ed. by Sharpe, J., Oh, V., Lancaster University EDC, 1994, pp. 79–94.

FORBUS, K. D.: Qualitative Process Theory; *Artificial Intelligence,* Vol. 24, No. 3, 1984, pp. 85–168.

HENDERSON, M. R. – TAYLOR, L. E.: A Meta-Model for Mechanical Products Based Upon the Mechanical Design Process; *Research in Engineering Design,* Vol. 5, No. x, 1993, pp. 140–160.

HORVÁTH, I. – DOROZSMAI, K. – THERNESZ, V.: A Feature-Object-Based Practical Methodology for Integration of Conceptual and Morphological Design, *Proceedings of the 1994 Lancaster International Workshop on Engineering Design CACD '94,* ed. by Sharpe, J., Oh, V., Lancaster University EDC, 1994, pp. 131–149.

HORVÁTH, I. – KULCSÁR, P. – THERNESZ, V.: A Uniform Approach to Handling of Feature-Objects in an Advanced CAD System; in Advances in Design Automation, DE-Vol. 69–1, ed. by Gilmore, B. J., Hoeltzel, D. A., Dutta, D., Eschenaurer, H. A., ASME, New York, 1994, pp. 547–562.

HORVÁTH, I. – KULCSÁR, P.: An Object Oriented Modelling Methodology for the ICAD Development; in Preprints of the *JSME-ASME Joint Workshop on Design '93,* ed. by Fukuda, S., Tomiyama, T., Tokyo, 1993, pp. 52–57.

HORVÁTH, I. – THERNESZ, V.: Integration of Design and Process Planning Based on Abstract Feature-Objects; in *Proceedings of the International Symposium on Manufacturing Science and Technology for the 21st Century MST '94,* ed. by Zhaoying, Z., International Academic Publishers, Beijing, 1994, pp. 424–429.

HORVÁTH, I. – THERNESZ, V.: Morphology-Inclusive Conceptual Modelling with Feature-Objects; Submitted for *Computer Graphics and CAD Conference,* Wuhan, China, October 1995, (to appear).

HUMMEL, K. E. – BROWN, C. W.: The Role of Features in the Implementation of Concurrent Product and Process Design; in *Proceedings of Symposium on Concurrent Product and Process Design,* ASME WAM, San Francisco, December 11–15, 1989, pp. 1–8.

KIRIYAMA, T. – TOMIYAMA, T. – YOSHIKAWA, H.: Qualitative Reasoning and Conceptual Design with Physical Features; in *Proceedings of the 4th International Workshop on Qualitative Physics,* Lugano, 1990, pp. 153–160.

KIRIYAMA, T. – TOMIYAMA, T. – YOSHIKAWA, H.: Building a Physical Feature Database for Integrated Modelling in Design; in Working Paper for the *Sixth International Workshop on Qualitative Reasoning about Physical Systems,* Edinburgh, 1992, pp. 124–138.

KUTTIG, D.: Potential and Limits of Functional Modelling in the CAD Process; *Research in Engineering Design,* Vol. 5, No. 1, pp. 40–48.

PALMER, S. R. – SHAPIRO, V.: Chain Models of Physical Behaviour for Engineering Analysis and Design; *Research in Engineering Design,* Vol. 5, No. 3, 1993, pp. 161–184.

PAYNTER, H. M.: Analysis and Design of Engineering Systems; MIT Press, Cambridge, 1961.

ROTH, K. H..: Neue Modelle zur rechnerunterstützten Synthese mechanischer Konstruktionen; *Konstruktion,* Band 31, Heft 7, S. 283–289.

ROTH, K. H..: Konstruieren mit Konstruktionskatalogen; Springer-Verlag, Berlin, 1982.

RUDE, S.: Rechnerunterstützte Gestaltsfindung auf der Basis eines integrierten Produktmodells, *VDI Fortschrittsberichte,* Reihe 20, Nr. 52, VDI Verlag, Düsseldorf, 1991.

SALOMONS, O.: Computer Support in the Design of Mechanical Products; Gegevens Koninklijke Bibliotheek, Den Haag, 1994.

SCHULTE, M. – WEBER, C. – STARK, R.: Functional Features for Design in Mechanical Engineering, *Computers in Industry,* Vol. 23, pp. 15–24.

SHAH, J. J.: Feature Transformations between Application-Specific Feature Spaces; *Computer-Aided Engineering Journal,* Vol. 5, No. 6, 1989, pp. 247–255.

SHAH, J. J.: Features in Design and Manufacturing; Intelligent Design and Manufacturing, ed. by Kusiak, A., John Wiley and Sons, New York, 1992, pp. 39–71.

SHARPE, J. – OH, V. (eds.): Computer Aided Conceptual Design, *1994 Lancaster International Workshop on Engineering Design CACD '94,* Lancaster University EDC, 1994.

TALEB-BENDIAB, A.: ConceptDesigner: A Knowledge-Based System for Conceptual Engineering Design; in *Proceedings of ICED 93,* Vol. 3, ed. by Roozenburg, N. F., Edition Heurista, 1993, pp. 1303–1310.

THOMA, J. U.: Simulation by Bondgraphs, Springer Verlag, Berlin, 1990.

TOMIYAMA, T.: Systematization and Representation of Design Knowledge for Integrated CAD Systems; in Information als Produktionsfaktor, ed. by Görke, W., Rininsland, H., Syrbe, M., Springer-Verlag, Berlin, 1992, pp. 49–63.

TOMIYAMA, T.: Design Support Systems; *International Journal of Japan Society of Precision Engineers,* Vol. 27, No. 4, 1993, pp. 287–290.

VOELCKER, H. B.: Modeling in the Design Process; in Design and Analysis of Integrated Manufacturing Systems, ed. by Compton, W. D., National Academy Press, Washington, DC, 1988.

WELCH, R. V. – DIXON, J. R.: Conceptual Design of Mechanical Systems; in Design Theory and Methodology, DE–31, ASME. New York, 1991, pp. 61–68.

WELCH, R. V. – DIXON, J. R.: Guiding Conceptual Design Through Behavioral Reasoning; *Research in Engineering Design,* Vol. 6, No. 3, 1994, pp. 169–188.

XUE, D. – TAKEDA, H. – KIRIYAMA, T. – TOMIYAMA, T.: An Intelligent Integrated Interactive CAD – A Preliminary Report; in Intelligent Computer Aided Design, ed. by: Brown, D. C., Waldron, M., Yoshikawa, H., Elsevier Science Publishers, Amsterdam, 1992, pp. 163–187.

XUE, D. – DONG, Z.: Developing a Quantitative Intelligent System for Implementing Concurrent Engineering Design; *Journal of Intelligent Manufacturing,* Vol. 5, 1994, pp. 251–267.

ZARGHAM, M. R. – TYMAN, M.: Neural Petri Nets; in *International Workshop on Time Petri Nets,* ed. by Marsan, M. A., Balbo, G., Trivedi, K., IEEE, Computer Society Press, Silver Spring, 1985, pp. 72–77.