INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 03, STOCKHOLM, AUGUST 19-21, 2003

NUCLEUS-BASED PRODUCT CONCEPTUALIZATION: PRINCIPLES AND FORMALIZATION

I. Horváth, W.F. van der Vegte

Abstract

Focusing on computer support of conceptual design, the paper introduces the nucleus concept as a vehicle for the development of a novel modelling front-end that can be transitively integrated with commercial CAD systems as well as to extend the functional profile of these systems. The nucleus-based modelling focuses on the relations between the function carriers, rather than on their formation and properties. Relations are governed by the intended operations a product is supposed to provide as well as by the constraints that are imposed on its behaviours. Nucleus-based modelling is dedicated to conceptual design and simulation, but it can also be employed in detail designs and simulation. In conceptual design, artefactual design concepts are decomposed to interrelated nuclei. A nucleus is a purposeful coupling of at most two effect carrier objects, whose geometry is described by half-spaces, a set of non-conflicting physical effects acting on the carrier objects, and situations. The time history of the relationships implies elementary processes that are the basis of behavioural simulation. To represent the geometric and morphological aspects, various modelling techniques such as functional skeleton modelling, vague discrete interval modelling and fuzzy shape modelling can be adapted.

Keywords: modelling entity, conceptual design, design concepts, nucleus based representation, skeleton models

1. Introduction: Has everything been solved in computer-aided modelling?

Advertisements of the developers of computer aided design and engineering systems suggest that everything what is of importance for the practical designers is supported by the commercialised systems. Textbooks talk about the wide range of design functions that are supported by computer software, but they also criticize human-computer collaboration and interactions that were defined by the former level of computer science and technologies rather than by the intrinsic nature of design. Research papers claim that there is much more ahead than behind, since thinking about the actual needs of designers has just recently started and there are plenty things out there to understand and to support. Experienced design scientists recognize the great advancements achieved so far, but also tend to see an ocean of opportunities for further extension of the current functionalities and the way of using design support systems [1].

Computer aided design systems have been developing through three generations which are reflected in, among other things, the evolution of the modelling entities from (i) 2D/3D curve and symbolic entities, through (ii) volumetric, boundary or surface primitives, to (iii) predefined or user-defined solid and/or freeform surface features (Figure 1) [2]. The opportunities

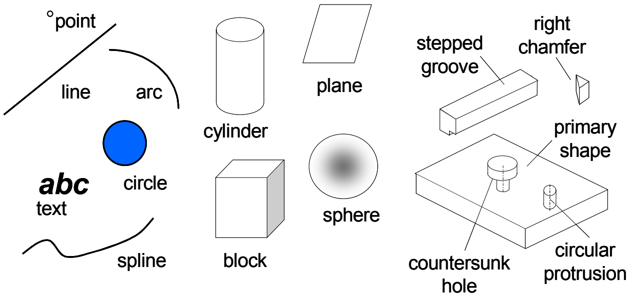

Figure 1 Evolution of modelling entities in CAD systems from low-level drawing entities through medium-level geometric entities to high-level feature entities

offered by the current feature-based design systems are incomparable to the opportunities provided by the early modelling systems [3]. Amplifying the remarkable push of the existing and emerging enabling technologies as well as the much weaker, but existing pull of the requirements originating in the practical applications, research in design has become a real engine behind the progress. Researchers are thinking not only about the design methods and techniques, but also about the real needs of designers and design processes in terms of computer based support. And this new thinking triggers, as many times earlier, a debate about the ultimate goal of computer internal modelling. The expansion of CAD/E systems to conceptual design introduces problems in terms of the modelling entities. When we take into account the modelling approaches that follow the mental processes and the thinking of designers, and, in addition, reflect the way the majority of designers would prefer to enjoy computer support, the current solutions are far form being optimal. Just consider, whatever it involves, computer support of conceptual design. The overwhelming majority of the currently used systems have been developed to support detailed design of parts and assemblies, and downstream application oriented modelling with geometry in the centre, to enable analyses and simulations. Research systems offer specific approaches to specific problems of conceptual design based on dedicated theories, but they are typically not connected to, and difficult to integrate with, the above mentioned systems due to the high level of abstractions in the models. In spite of the fact that many researchers believe it is totally in line with the nature of conceptual design, other solutions can also be thought of. Actually, this is the primary objective of this paper. With computer aided conceptual design in the centre, we sketch up a new way of thinking about modelling, which lends itself to a more evocative formation to models, following the way of thinking of designers.

2. Recognized requirements and the roots of limitations

The next generation of CACD/CAD systems is expected to provide both in breadth and in depth support to product development. In breadth support regards all possible or necessary aspects of the product life cycle, in depth support concerns the information and knowledge content of the model as it evolves, from definition of product concepts to realization. This requested increase in the capabilities of CAD systems assumes more knowledge-intensive modelling entities to be shared in modelling, analysis and simulation. Our research has concentrated on this latter aspect. The major issue for our research has been the description and testing of a modelling entity that is more powerful in conceptual design than any one of the known modelling entities. We drew a distinction between advanced product modelling and conceptual product modelling. The former one defines the requested functionality of products, determines the applicable and compatible physical principles, defines alternative components and structures, and derives generic (initial) geometries for a product. Conceptual design works

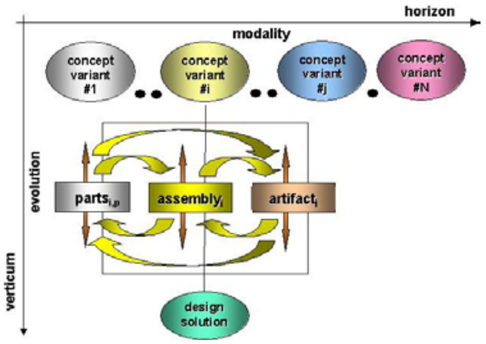

Figure 2 Managing multiple concept variants on part, assembly and system level

with design concepts that are typically abstract, incomplete and vague. Detail design is for a comprehensive specification of the geometric features and mechanical attributes of the parts and the assembly. Whilst early behavioural simulations provide information about the expected behaviour mainly by qualitative reasoning, advanced behavioural simulations are to qualitatively investigate the behaviour of a product and of the components of it in both the space and time domains.

On the level of functional and methodical requirements, we envisage CACD systems (a) used simultaneously by more than one designer at geographically distributed locations; (b) offering multi-modal input facilities such as verbal communication, hand motion, or digitalized input; (c) having the capabilities to handle incompleteness, vagueness and impreciseness of models and information; (d) managing multiple concept variants on part, assembly and system level (Figure 2); (e) being able to provide fast simulations of the physical behaviour of the product during conceptual design, involving the related humans and the environment; and (f) supporting in-process physical modelling and early prototyping. The CACD systems fulfilling these requirements will operate as front ends of the conventional CAD/E systems, facilitating detail design and numerical analysis of parts, assembly design and behavioural simulation of products. The roots of limitations of present CAD systems originate in the following facts. Present application feature definitions are (a) targeted to detail design, (b) based on geometric definitions at bottom (Figure 3), (c) restricted in terms of expressing behaviour, (d) lacking of high-level semantic associations. Apparently, (a) it is difficult, if not impossible, to define all features for multiple applications, (b) inter-views or inter–aspects management of features suffers from combinatorial complexity, (c) specification of features to support conceptualisation has got lost in a dead way, and (d) mapping of feature classes among each other is theoreti-

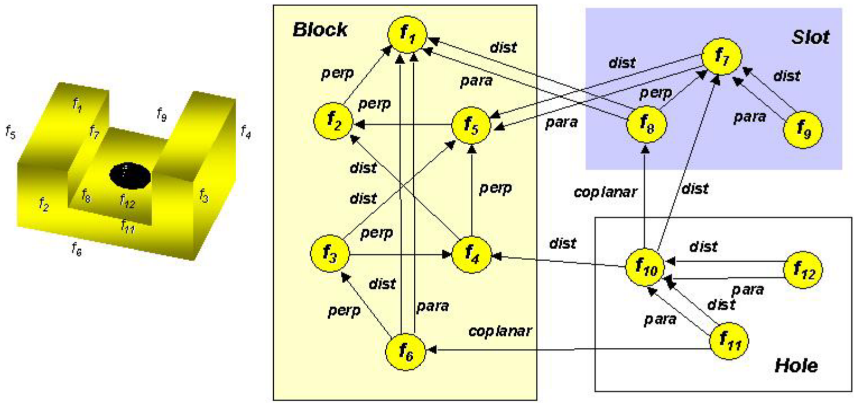

Figure 3 Defining a mechanical part as a combination of a base shape (block) and two parametric features (slot and hole)

cally not supported.

Application feature modelling is the current paradigm for detailed geometry, assembly and manufacturing modelling as well as for downstream activities. Introduction of the seminal concept of features in the mid-eighties has sparked an avalanche of scientific research and technology development work devoted to explore the theoretical fundamentals and develop feature-based design technologies. Since the notion of feature is more cognitive than factual, the supporting theories are typically weak. The kernel of conventional feature entities is some region of a part's geometry that carries a specific meaning for the designer. Typically, the representation of the feature geometry is extended with quasi-semantic information as well as with means for validity and consistency maintenance. Feature libraries have been defined for particular modelling views and various mapping techniques are used to create associations among multiple views. The definitions of features are orientated to downstream applications rather than to conceptualisation. Regrettably, typical feature definitions do not have the potential of an integral and evolutionary modelling of geometry, structure, processes and behaviour, and unable to handle uncertainty, incompleteness and ambiguity that are typical characteristics of conceptual models of artefacts. The major shortcoming with respect to behavioural simulation is that feature technology is confined to handling permanence rather than changes. Practically each natural and artefactual system is of a transitory nature that manifests in observable behaviour that is realized by the interactions of function carriers of different mechanical components. Conventional feature representations are application dependent and intend to capture morphological aspects rather than the semantics of functions and the manifestations of operation/behaviour.

Being aware of the potential of feature technology, the objective of our research has been to find possible answers to questions such as: What modelling entity concept comes in product modelling when the feature paradigm is exhausted? What information and/or knowledge have to be conveyed by these entities in order to be able to support conceptual modelling/simulation and detail modelling/simulation equally well? In this paper we propose the nucleus theory as a basis of next generation product modelling, explaining the innovative concept and showing that it results in a family of modelling entities that forms a superset of current feature entities and dramatically extends the functionality. The major difference relative to feature-based modelling is that the notion of geometric entities as fundamental building blocks to abandoned in favour to relations that actually govern the formation of geometry. In [4], an application-oriented discussion is included together with an application case study that demonstrates the advantages and directs the attention to the yet unexplored opportunities.

3. The concept of nucleus: What can it solve?

It is presumed that any new modelling entities should support feature-based design and processing, i.e., it has to support feature technology in general. In addition, the introduction of some new modelling entities should lead to knowledge-intensive conceptual models offering new functionalities for the designers to conceptualise products. We hypothesized that a new modelling entity has to focus on design concepts that are intuitively or systematically generated by the designers and to make it possible to represent their elements and entirety. It implies the need for a deeper understanding of the nature of design concepts and the possible ways of formalization without destroying creative power. It is especially important with respect to the inherent intuitiveness, incompleteness and uncertainty of design concepts and the heuristic nature of conceptualisation. Obviously, the modelling entities have to be of a very high level (or complex) to be capable to incorporate sufficient amount of knowledge for concurrent modelling of components, assemblies and systems. It amounts to saying that the current systems are somewhat limited in these capabilities.

We developed the nucleus theory as a foundational theory of a new product modelling methodology, and studied the feasibility and applicability. Obviously, this novel approach to conceptual and detailed modelling of products introduces new concepts, notions, terms and words that need to be defined, explained and put into context. Below we explain the fundamental concepts and clarify the specific notions. We had investigated various engineering products and found that they all can ultimately be decomposed to a purposeful composition of physically coupled pairs. Any physically coupled pair can be abstracted as a composition of - typically two - interacting objects and multiple physical relations between the objects that may appear in various situations. Actually, this abstract construct gave the idea of the nucleus, which is understood as a generic modelling pattern that can be specialized to describe the constituents of a design concept or its entirety. From a programming point of view, nucleus is a complex data and relation structure that covers geometric, structural, morphological, material and physical aspects. From a modelling point of view, this is the lowest level entity that carries both morphological and functional information to applications through the embedded structure of objects, relations and conditions.

As mentioned above, our intention has been to represent design concepts by a purposeful set and configuration of nuclei. With symbolic terms, we formalized a design concept as $DC = \{O, \phi, S, C, A, D, P\}$, where $O = \{(o_i, o_j)\}$ set of pairs of objects, A = attributes of objects, ϕ = physical relations, P = parameters describing the relations, S = situation in space and time, D = descriptors of situation, C = constraints on attributes, parameters and descriptors. Design concepts can be decomposed but not beyond any limit. If the objects, relations and situations are missing, the abstraction becomes meaningless. Actually, this is another reason to call the $N = \{O, \phi, S\}$ triplet the nucleus of a design concept (Figure 4). A semantics driven decomposition of design concepts results in nuclei that represent ultimate constituents. Representation of a most elementary design concept requires at least one nucleus. Compound design concepts however need a purposeful composition of finite number of nuclei. A situation arranges the objects in a set of relations, or, in other words, creates a given structure of elementary processes described by the mathematical formulas. A situated nucleus lends itself to computable behaviour, that is, to temporal changes in the parameter values as governed by the mathematical formulas and constraints.

The objects incorporated in a nucleus are metric entities, which are characterized for their shape and volume. The shape of the objects is represented by half spaces (HS). Actually, a region of these infinite half spaces is used in model building. The finite regions correspond to the natural surface patches of a mechanical part of a product, and lend themselves to effect carrying surface patches. Some of the effect carrying patches will be in contact with surface patches of other mechanical parts. The surface patches are positioned in the model by reference points and may have multiple other reference points for the physical relations assigned to them. For the reason that the geometry of these surface patches is always defined by the ge-

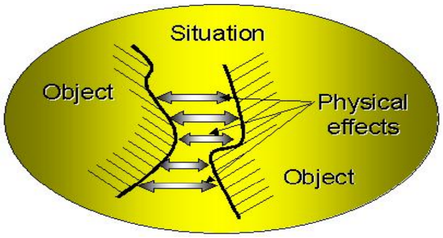

Figure 4 Ontological conceptualisation of a nucleus

ometry of the describing half spaces, in the further discussion we replace the abstract objects in a nucleus with half spaces. Thus, $N = (HS_n, HS_c, \phi, S)$, where HS_n is called a native half space, HS_c is called a complement half space, and ϕ and S are as above. A half space indicates the material domain of an object. Native half space is the term used to identify those half spaces that jointly define the boundary of a mechanical part. Complement half spaces are half spaces defining the boundary of other mechanical parts that are in logical, geometric, positional or physical relation with some native half spaces of a particular mechanical part. Our interpretation allows an object to exist in the nucleus without half space definition. In this case the object is logically identified, but geometrically not specified. This is a substantial assumption that enables incomplete modelling in conceptual design. If the half spaces included in a nucleus are geometrically specified, explicit and implicit analytic surface patches, finite parametric surface patches, or finite discrete point or particle clouds can be used as representations. From the aspects of physical modelling, arbitrary number of relations can be specified between the pairs of half spaces. For a nucleus to operate, at least one half space must be geometrically specified, but, in this case, only reflexive physical relations can be assigned. Represented by half spaces, the objects acting as 'environment' must have at least one reflexive relation to result is a non-limitless system.

The physical relations imply processes that boil down to the behaviour of a nucleus, or a design concept. Actually, the time-dependent changes described by the physical relations will lend themselves to some observable operation, or behaviour, of a nucleus, B, in some situations: $B(N) = \Gamma \{S_k (o_i \phi_{ij} o_j)\}$, where $o_i, o_j \in O$, ϕ_{ij} and S_k are as above, and Γ is a behaviour generator function, which takes into consideration the interaction of various nuclei and the influences on each other's behaviour. The introduction of Γ is necessary, since the observable operation of a modelled design concept, DC, is an aggregation of the elementary operations of the nuclei. For the reason that all nuclei might interact in a composition, this aggregation can be represented as a Descartian product rather than as a Boolean union of the observable elementary operations, that is, $B(DC) = B(N_i) \times B(N_j)$, or $B(DC) = \Pi (B(N_i), B(N_j))$, where Π denotes a mathematical product. The arrangement of situations, or in other words, the operation and interaction of the nuclei, are governed by so called scenarios. A scenario, Σ, prescribes a sequence of situations, in which the observable operation delivered by a nucleus or a configuration of nuclei incorporated in a design concept happens. That is, $\Sigma = \cup (S_k)$. With these, the behaviour of a DC is: $B(DC) = \Gamma (\Sigma \{ N_i \})$, or, on the level of relations, $B(DC) = \Gamma (\cup (S_k (o_i \phi_{ij} o_j)))$. Specification of the physical relations includes definition of the parameters, the mathematical formulas (equations and rules) that relate the parameters to each other, and the constraints and value domains. Thus, a nucleus is a primitive system in itself, since its data structure contains all pieces of information that is needed to simulate its behaviour. Based on the above terminology, we call our approach a nucleus-based conceptual modelling of engineering products. At this point we might revisit our previous observation, namely, that engineering products can be modelled in terms of physically coupled pairs (PCP) [5]. We may say that a PCP is a concrete manifestation of a nucleus, which is able to operate in situations. Examples for such PCP are shown in Figure 5.

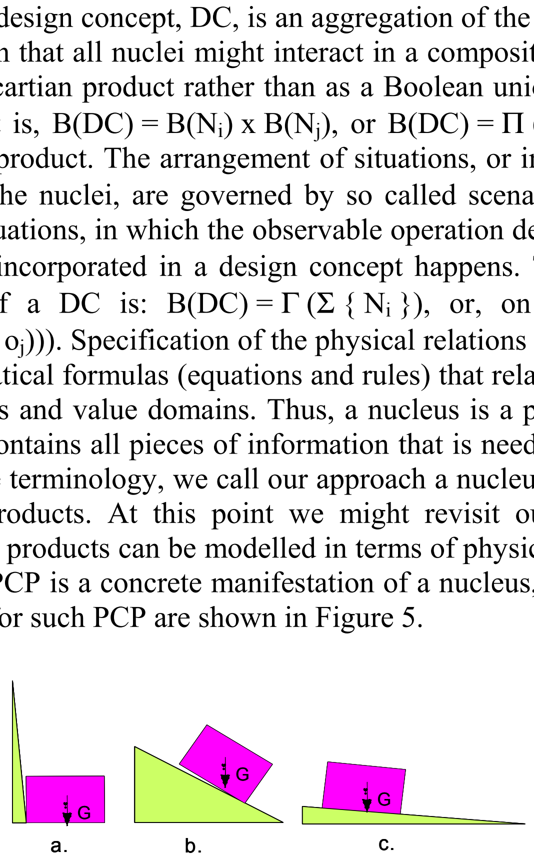

Figure 5 Examples for physically coupled pairs in different situations due to the different arrangement of the objects and the manifestation of physical effects

4. From relations to mechanical parts, assemblies and systems

In simple words, relations express the ways in which objects can stand with regard to one another or themselves. Let O be a set of objects and ϕ a set of relations. The domain of ϕ is the set of objects $o_1 \dots o_n \in O$ for which there is at least one o_i such that $\phi_i \in \phi$ holds. The converse domain of ϕ is the set of entities $o_1 \dots o_n \in O$ for which there is at least one o_j such that $\phi_i \in \phi$ holds. The logical sum of the domain and the converse domain is the field of relations ϕ. A universal relation contains both o_i and o_j as arguments. A universal relation is symmetric if $o_1\ \phi\ o_2$ and $o_2\ \phi\ o_1$ hold. A set of reflexive relations contains o_i as argument such that $o_i\ \phi\ o_i$. The square of a set of relations ϕ is $\phi \mid \phi$. A set of relations is transitive if each relation contains its square, that is, if $o_i\ \phi\ o_j$ and $o_j\ \phi\ o_k$ hold, then $o_i\ \phi\ o_k$. Relations can be seen as special sort of objects that connect other objects but are numerically distinct and ontologically independent from the connected objects. If o_i stands in relation ϕ to o_j, but neither its identity nor its nature depends upon o_j, the relation is external. If the opposite is true, then ϕ is internal.

The need for the explicit and effective handling of relations of mechanical parts has appeared more than 10 years ago with the intensification of the efforts to develop powerful assembly modelling systems. In such a system the specification of relations between the parts is at least as important as, or even more important than, the description of the parts. For an all-embracing handling of relations, the complexity presented in Figure 2 has to be taken into account. It opens up two dimensions of thinking about relations. The first one is the context of the relations; the second is the kind of relations. Various types of relations can be considered in various contexts. As contexts of specification of relations we identified mechanical part, assembly and system design (Figure 6). A mechanical part level relation exists in between pairs of native half spaces; therefore, it is called internal relation. If it brings two close neighbour (intersecting) half spaces in spatial relationship, then it is called direct internal relation. If it concerns two far neighbour half spaces, then it is an indirect internal relation. A mechanical assembly relation exists between one-one native half spaces of two mechanical parts, which represent a native-complement construct. The assembly relations are called external relations, and based on the analogy of internal relations, they can also be direct (in contact) or indirect (not in direct contact). Finally, system level relations describe interactions with elements of the nuclei representing the physical environments. System level relations offer themselves to the representation of, for example, product-user-environment configurations, as it will be shown in [4].

The type of relations depends on the semantics of the relations. We introduced (a) ontological, (b) connectivity, (c) morphological, (d) positional, and (e) physical relations [6]. An ontological relation indicates the existence of an object or any higher-level construct; therefore, it is

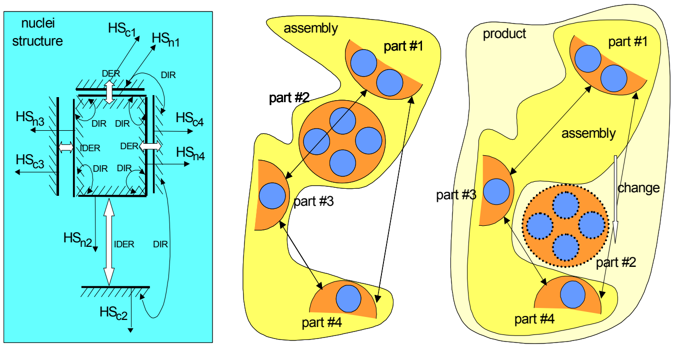

Figure 6 Relations on mechanical part, assembly and system levels

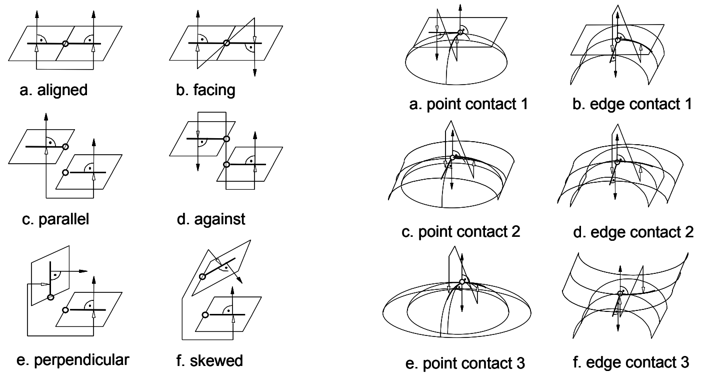

Figure 7 Associating morphological relations between pairs of planes and of single/double-curved surfaces

reflexive. Connectivity relations define the topography of relations between objects and, as discussed above, they are used to define either mechanical parts or assemblies. Reflexive morphological relations define the geometry of the half space describing the metric of an object. Associating morphological relations define the relationship between two half spaces of different objects. A set of associating morphological relations is shown in Figure 7. Positional relations specify the rotations and translations between the half spaces of a nucleus or any two higher-level constructs. Finally, physical relations formulate physics-based relationships between half spaces of a nucleus to transfer physical effects. They can be reflexive (such as mass) or non-reflexive (such as a force). The relations are described by means of parameters and mathematical formula. The geometric aspect and the effect aspect are brought into synergy through reference points or spots.

Based on the nucleus concept, a conceptual modelling system is able to know about and manage a complementing object when a native object is defined. The system is also able to automatically apply all default relations for any pair of objects and to let the designer activate only the necessary ones. Based on activating an internal relationship, the system can be aware of the fact that a mechanical part is being formed, and activating an external relationship means that an assembly is generated. The system can not only monitor these steps of conceptualisation, but also can control the processes and check for validity, completeness and consistence. In system programming, the nucleus concept lends itself to the internal modelling scheme of a CACD system. In fact, it is observable only in the prevailing modelling methodology that focuses on the relations and handling the changes in the relations of objects in various situations. Activation of a nucleus offers a generic modelling entity for the designer that can be further specified according to the design concepts to be applied to solve the design problem. Should a nucleus be activated, the designer is given a set of relations that are specified in terms of attributes, parameters and descriptors. In principle, infinite number of relations can be specified between two objects, but in practice only those will be instantiated that are important for a given modelling or simulation task [7].

Parameters representing flow quantities and cross quantities are referred to specific points on the half spaces, which are called ports. In the case of an incomplete part or assembly model, indication of the integrity is a remarkable problem. As a simple solution, fictitious connection lines are generated and visualized between the reference points of the half spaces being in internal positional relations. This leads us to a physically based skeleton model, which is one of the alternative realizations of the nucleus concept as a practical modelling methodology (Figure 8). Naturally, designers do not face these abstract concepts and terms when they are using a nucleus-based system in conceptual design. The design concepts are expressed in terms of

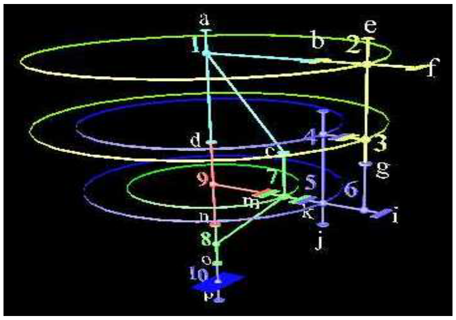

Figure 8 A physically based skeleton model of a ball pen

an arrangement of nuclei, e.g., in application features, which are represented as functionally related surface patches in given situations. A nucleus can be placed into different situations, which means instantiation of the interacting processes in different forms. Not only complex design concepts, but also design features can be defined in the same manner and used to express design concepts in a semantics-intensive way. Solid mechanics offers the means to treat the four main observable phenomena: motion, collision, deformation and fracture. Phenomena relating thermodynamics, fluid dynamics, gas dynamics, and so forth can also be considered in relations. It is a fact however that there exists no single predictive model that is capable to incorporate all phenomena and interrelated changes, not even theoretically.

5. Some conclusions on the merits

The proposed nucleus concept offers a relation oriented modelling, rather than entity centred modelling. It places the pairs of objects into a multitude of relations, which are not restricted to be in the same aspect or context. By doing so, it mimics the working of the human mind as it builds associations between neutral entities in a creative conceptualisation. It also tries to resolve the known problem of linking different views or jumping between aspects. By making the entity relationships more explicit and knowledge intensive, a nucleus-based conceptual design system converts the paradigm of ‘doing what you know’ to the paradigm of ‘knowing what you’re doing’. It allows the designers to describe design concepts as an aggregation of nuclei, to define and use application features, to construct mechanical parts and assemblies, and to investigate the physical behaviour of all these constructs based on space- and time-dependent evaluation of the specified relations. It involves validity management, consistency management and multi-view management.

An obvious advantage of the nucleus concept is that it does not force the designer to define the part geometries first. He may alternate between structure, component and system definition, leaving the geometry to appear as a by-product of the conceptualisation process. This methodology has been found advantageous in an evolutionary, multi-resolution artefact modelling. It assures compatibility with the existing systems since application features can also be defined based on the nucleus concept. A nucleus based conceptual design systems can work as a front end of a detail design system, integrating modelling and simulation functionality. Thus application of the nucleus concept means a significant step towards a truly knowledge-intensive artefactual system modelling and simulation.

The nucleus concept vindicates that models can be incomplete on mechanical part, assembly and system levels. Models can gradually be extended and refined as knowledge becomes available for the designed product. Extension and refinement may take place in terms of the morphological and physical relations. This way, the evolving model that integrates both artefact representation and process representation adapts to the progress of conceptualisation. This model is referred to as a multi-resolution model. A designer may start to solve the conceptualisation problem by making effort to define components, to describe the assembly as a structure, or to specify physical effects on system level, and can swap between these activities. The system supporting conceptual design is supposed to take care of the interrelations between these levels. A component is defined as a purposefully arranged set of nuclei. The definition can be complete or incomplete. The nuclei give the opportunity to specify assembly relations as well. Functional relations between nuclei can represent the history of physical effects together with the changes in a product.

Current research deals with extensional relations only, and considers them as n-ary relations that can be traced back to dyadic relations. The used prepositional functions do not extend to intentional relations. Note that we still face some sort of ‘metaphysical’ limitations in terms of being able to define any ideal modelling entity for the reason that an exact scientific understanding related to the following issues is still missing: (a) mapping requirements onto a system of functions or potential operations, (b) mapping target functions to first principles and physical processes, (c) mapping functions or structures to forms and embodiments, (d) deriving structures from first principles and physical phenomena, and (e) identification of the necessary constituents from physical processes.

References

[1] Horváth, I., “Shifting Paradigms of Computer Aided Design”, Delft University Press, Delft, 1998, pp. 1-36.

[2] McMahon, C., Browne, J., “CAD/CAM – Principles, Practice and Manufacturing Management”, Addison-Wesley, Harlow, 1998.

[3] Noort, A., 2002, “Multiple-view feature modeling with model adjustment”, Ph.D. Thesis, Delft University of Technology, Delft, pp. 1-138.

[4] van der Vegte, W.F. and Horváth, I., “Nucleus-based product conceptualization: application in designing for use”, Proceedings of ICED '03, Stockholm, 2003.

[6] Roth, K.-H., “Konstruiren mit Konstruktionskatalogen”, Springer Verlag, Berlin, 1982.

[7] Horváth, I., Kuczogi, Gy., Vergeest, J., “Development and application of design concept ontologies for contextual conceptualization”, in Proceedings of 1998 ASME DETC, Atlanta, DETC98/CIE-5701, 1998, pp. 1-16.

[8] Kitamura, Y., Sano, T., Namba, K., Mizoguchi, R., “A Functional Concept Ontology and Its Application to Automatic Identification of Functional Structures”, Advanced Engineering Informatics (former Artificial Intelligence in Engineering), Vol. 16, No. 2, 2002, pp. 145-163.

For more information please contact:

Professor Dr. Imre Horváth, Section of Computer Aided Design and Engineering
Faculty of Industrial Design Engineering, Delft University of Technology
Landbergstraat 15, 2628 CE Delft, The Netherlands
Phone: 00+31 15 278 3520, Fax: 00+31 15 278 1839
E-mail: i.horvath@io.tudelft.nl