Nucleus-Based Conceptual Design

Imre Horváth

Delft University of Technology, i.horvath@io.tudelft.nl

ABSTRACT

This paper presents the fundamentals and the instruments of a new approach to computer support of conceptual design. The main assumption is that a product can be described in conceptual design in terms of reflexive and bijective relations. A new modeling entity, called nucleus has been introduced to enable the explicit handling of relations. A nucleus comprises two objects and a set of relations between them according to a particular situation. Conceptual models are represented by specific compositions of nuclei of various levels such as component and assembly. The objects included in nuclei are discrete particles and connected by a given set of relations. To build a conceptual model the designer should generate the necessary nuclei by instantiating a generic one at multiple levels and by arranging these entities in a structure. The paper presents the categories of relations, and explains the concept of nucleus-based behavioral modeling and database management.

Keywords: conceptual design, relation centered representation, nucleus-based modeling

1. INTRODUCTION

Conceptual design is in a paradoxical situation. On the one hand, it is getting more and more attention in the industry due to the recognition that the most important decisions on functions, principles, structure, materials, operations, manufacturing, use, recycling and costs are made in this phase of product development [1]. In fact, it has been recognized that conceptual design needs at least as extensive computer support as detail (geometry and structure) design, simulation, manufacturing preparation and process control [2]. On the other hand, the theoretical understanding of conceptual design is still in its infancy, the methodological support of problem solving is behind the expectations, and the current computational approaches are rather specialized, disjointed and lacking coherence. Computer aided conceptual design (CACD) is far from a mature industrial technology [3].

There are two major reasons of this situation. First, conceptual design largely depends on human cognitive capabilities such as (a) conjectures, (b) hypothesizing, (c) ideation, (d) abstraction, (e) generalization, (f) creativity, and (g) analysis. The best computational approach to conceptual design would be one based on a high fidelity modeling of the human mind. Accommodation of design intent, physical behavior and causal explanation in concept models is the challenge the computer aided design research community has to face. Second, conceptual design is a complex, highly unstructured and knowledge-intensive process. It involves (a) information aggregation, (b) idea generation, (c) externalization of the ideas in the form of design concepts, (d) creative composition of elementary design concepts, (e) reasoning about the evolving product (artifactual system), (f) elaboration of abstract and/or concrete models and representation, and (g) assessment of the conceived behavior by preliminary assessment. The information processed in conceptual design is typically abstract, uncertain, incomplete, multiform, qualitative, fragmented and evolving. Actually, the influence and complexity are why conceptual design needs an effective computer support, but also they are what make advanced computer support a non-trivial task.

Various theories, methods, tools and systems have been proposed, which can be categorized as either intuitive approaches (relying on the thinking, experiences and skills of designers), or algorithmic approaches (based on computational techniques and agents) [6]. The currently available computer-mediated modeling techniques range over initial functional models, shape models, structure models and behavior models (Fig. 1). At the same there exists no all-embracing universal solution. Various studies have shown that an all-inclusive substitution of the human functions by computational means does not make sense. At the same time, there is a strong need for new ingenious approaches that open up

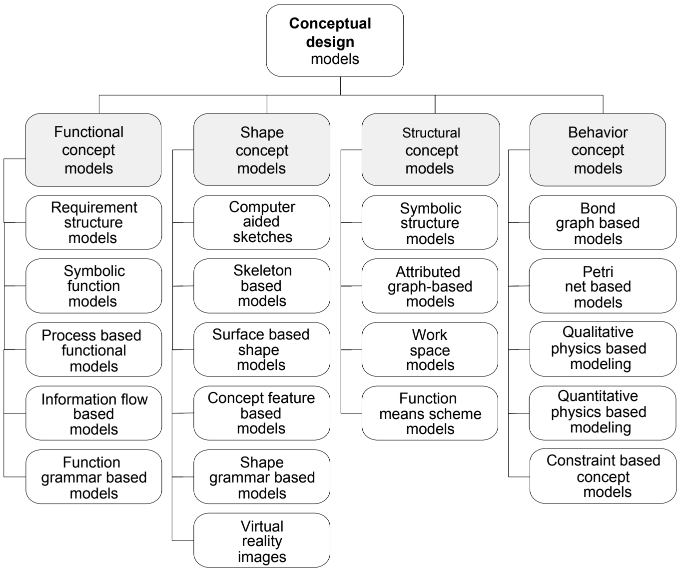

Fig. 1. A taxonomy of conceptual models

new opportunities in knowledge intensive conceptual modeling and simulation of products.

The goal of this paper is to contribute with a novel concept to the implementation of computer aided conceptual design (CACD). In view of the facts that (a) natural things and phenomena are interrelated in artifacts, (b) relations, rather than objects or entities, are the elementary structures to which physical systems can be reduced, (c) relations established in design determine the manifestation of artifacts and the occurrence of phenomena, and (d) in the process of conceptual design, what tend to remain invariable are the relations among objects, and not the objects themselves, we propose a relation-oriented approach to modeling and simulation of products in conceptual design. We hypothesized that relations can be used as a basis of modeling entities; likewise geometric objects have been used in computer-aided design (CAD) [4]. Based on this reasoning, a novel relation oriented modeling entity has been introduced. This modeling entity can be used as an instrument of a comprehensive methodology for early artifact modeling and behavioral simulation. It is very straightforward for designers to use since it represents a cognitive pattern, can be instantiated in multiple contexts, and enables an explicit handling of relations.

Session 2 further explains the idea and presents some formal definitions that underpin the relation oriented conceptual design methodology. Section 3 discuses the categories of relations and also presents the interpretation of relations. Section 4 deals with the issues related to the representation of conceptual artifacts on multiple structural levels. Section 5 presents the principle of a scenario driven control of behavioral simulation. Finally, Section 6 presents the supporting data scheme and knowledge management.

2. FUNDAMENTALS OF NUCLEUS BASED MODELING

The things that we can identify in the physical world surrounding us can typically be described in terms of an infinite number of relations. The relations may have to do something either with the manifestation of the things or with the behavior of them. This relation oriented view can also be applied to the conceptual design of artifacts in order to represent them in an intentional way, i.e., in terms of connections of the included objects under various circumstances.

Relations can be used to describe not only the properties of artifacts, but also their behavior. Consider the following example. A ball (object 1) put on (relation 1) a horizontal board (object 2) will stay on the board (relation 2) while it is kept horizontal (situation 1). The relations will change if the arrangement of the objects is changed, that is, if the situation changes. Namely, the ball (object 1) will roll down (relation 3) from the board (object 2) if it is sufficiently slanted (situation 2). And the ball (object 1) will fall down (relation 4) from the board (object 2) when its center of mass (relation 5) gets beyond (relation 6) the last contact point (relation 7) on the board (situation 3).

To implement a comprehensive relation oriented modeling we first have to take into consideration the issues of morphological representation of artifacts. In order to be able to apply relations to represent the geometry, the structure and the behavior of an artifact, we have to give up the conventional continuous representations and to introduce some sort of discrete representation that facilitate the explicit handling of

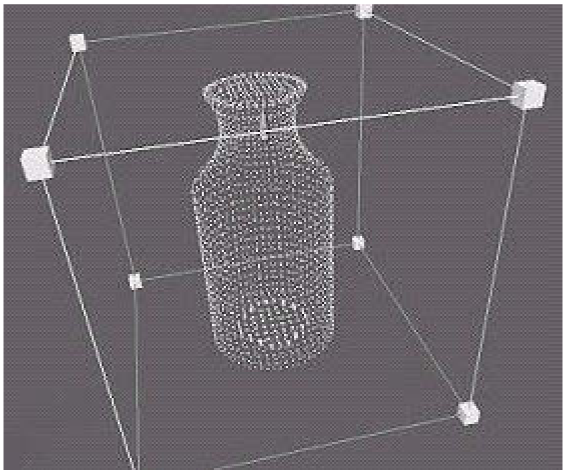

Fig. 2. Particle-based discrete model of a product

relations [5]. For this reason, we used clouds of discrete elementary objects, called particles, to represent artifacts from a morphological point of view (Fig. 2). By definition, particles are of finite and regular spatial geometry and have physical properties. A particle is positioned in the space E^3 by its reference vector that is a localized vector showing to endpoint of the vector from the origin of a global reference frame. Depending on the location of the particle in the artifact, we can identify boundary particles and volume particles. A boundary particle resides in the natural surface of an artifact and a volume particle is inside the boundary of the artifact. Hence, any artifact can be modeled as an arrangement of volume particles and boundary particles.

The particles may be involved in various relations. The relations can be either reflexive or bijective. Though, in a general case, the number of relations is infinite, a finite set of relations is usually sufficient for modeling the behavior of a concept product in a specific situation. If the situation is changing, the validity of relations is also changing. Based on this reasoning, we can consider an arrangement of two particles connected by a set of relations in a given situation as a modeling entity. Formally, $N = \{\Pi_i, \Pi_j, \Phi_k, S\}$, where Π_I and Π_j are two connected particle, Φ_k is the set of relations, and S is the situation. We took this construct as a modeling entity and called it *nucleus*.

A nucleus has a dual nature. Seen from our mental world, it is a meaningful logical unit of our reasoning that is associated with an elementary knowledge construct. From a modeling 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. Seen from the physical world, a nucleus is as an abstraction of elementary (constituting) parts of human design concepts.

Based on its dual nature, a composition of nuclei allows the designers to represent both design concepts and artifacts as a purposeful composition of specific instances of nuclei (Fig. 3). The composition can be both partial and complete, involving nuclei that are instantiated at multiple levels of complexity such as entity, component, subassembly and assembly. The valid changes that can be computed based on the mathematical formulation of relations are the basis of the simulation of the behavior of artifacts of the above complexity levels. The constraints applied to the relations give the conditions of the behavior in the considered situations. Since the nucleus is able to represent only logically defined particles, incomplete conceptual models can be defined and made complete as the knowledge about the artifact increases. It means different instantiation of a nucleus in the process of conceptual modeling in terms of the particles, the set of relations, and the situation.

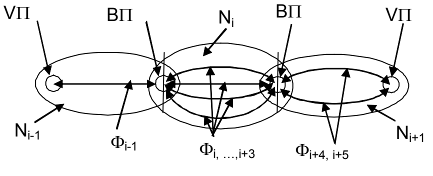

Fig. 3. Nucleus-based representation

As far as numerical processing of relations is concerned, nucleus-based modeling has a lot common with parametric/constraints-based modeling and quantitative modeling physical processes [7]. The relations are described in terms of parameters, equations and constraints, which together form a constraint network [8]. This network is typically represented as a constraint graph. The nodes of such a graph correspond to parameters and the edges of the graph correspond to the constraints. In addition to geometric parameterization, methods for topological and structural parameterization have been developed. Various types of constraints relevant in geometric modeling have been discussed in [9]. The concept of constraints-based designed has been extended to the area of layout and assembly modeling [10].

The idea of using constraint-driven approaches is not new in conceptual design. The notion of concept feature-objects has been introduced long time ago to represent energy transformation processes and describing the minimum geometry of mechanical products with morphological constraints [11]. The objective of this research was to allow a morphology-inclusive generation of conceptual models in addition to a design language-based representation. In their pilot implementation, organ structures of mechanical products were modeled as physically-based skeletons.

The issues of using formal language and grammar concepts to generate products have been investigated in [12]. A generic formalism is proposed as a common core for the application of formal grammars in various areas of engineering design. The mathematical definitions of relations for the nucleus-based modeling and the relations management have been presented in [13]. The applicability of the nucleus-based modeling in use process modeling and forecasting was investigated in [14].

3. CATEGORIES AND FORMALIZATION OF RELATIONS

Relations are special sort of ‘objects’ that connect particles but they are ontologically independent and functionally distinct from them. In fact, they are existential, manifestation and behavioral associations, dependencies and interactions between humans, artifacts and environments. Relations can be classified based on arity, that is, by the number of particles they involve. Considering the fact that any $M \leftrightarrow N$ relation can in principle be decomposed to M number of $1 \leftrightarrow N$ relations, which can further be decomposed to N number of $1 \leftrightarrow 1$ relations, only unary $(1 \leftrightarrow \quad)$ and binary $(1 \leftrightarrow 1)$ relations are represented.

Since unary relations concern one object only, they are reflexive Binary relations are defined between two objects. It follows from the above argumentation that relations can be assigned if and only if at least one particle has been defined. If one particle is defined, then reflexive relations can only be assigned. Particles acting as 'environment' must have at least one reflexive relation to result in a non-limitless system. Specification of the relations includes definition of the parameters, the mathematical formulas over the parameters, the constraints, and the value domains. Relations can be different from the aspect of semantics.

As it was mentioned earlier, we identified two general categories of relations: unary and binary. Unary relations are (a) existence, (b) reference, and (c) substance relations. Groups of binary relations are (a) connectivity, (b) positional, (c) morphological, (d) kinematical, (e) deformational, (f) kinetic, (g) physical effect, and (h) physical field relations. Note that some of these relations can be defined on each modeling level (i.e. connectivity). The others are specific to a given level (i.e. surface normal vector which can only be defined on particle level). The groups of the relations also depend on each other, since they use the information captured in their parent relation group. For instance, morphological relation of two objects provides information about the degrees of freedom for kinematical relations.

If we take into consideration the type of particles that are connected in a nucleus, we can differentiate internal positional relations and external positional relations. In the first case, both particles belong to the same component. If the distance between the reference points of these particles is zero, then we are talking about direct internal relation, otherwise, about indirect internal relation. If the two particles belong to two different components, then we talk about external relation. If the distance between the reference points of these particles is zero, then we are talking about direct external relation, otherwise, about indirect external relation. The above mentioned types of relations imply a kind of hierarchical structure, and therefore, an inheritance of the parameters and attributes.

As an example, the formal representation of nucleus defining one existence relation, $\square^{E}$, contains (a) a unique identifier of the relation, (b) reference to the identifier of a particle, (c) an empty list of relations that can be defined for the particle, (d) the level on which the nucleus is defined, and (e) the type of existence (i.e. logical, or metric, or both). As a consequence, when a particle is generated, a nucleus is also generated. A reference relation assigns a reference point on a component, assembly or a system as a reference of other relations. A substance relation assigns a set of substantial properties to a reference point. For the time being, six substance relations have been considered (a) kind of material, (b) mass, (c) center of gravity, (d) local curvatures, (f) inertia, and (g) surface normal vector.

Connectivity relation expresses that a particle belongs to an artifact, assumed it is at least logically defined. However, it is a bijective relation, since if a particle is connected to another, it is true vice versa. Different connectivity relations can be specified for the $B\Pi$ or $V\Pi$ particles. Connectivity relations can be internal (i.e., within a component) and external (i.e., between two components, assemblies and systems). If Π_i stand in a connectivity relation with Π_j, but neither its identity nor its nature depends upon Π_j, then the connectivity relation is external. If the opposite is true, then the connectivity relation is internal.

Positional relations quantitatively or qualitatively specify the position, rotation and placement of particles relative to each other. Actually, a position relation defines a metric distance between the reference points of two particles, or between two positions of the same particle. Rotation relation defines a metric angle between two pairs of coplanar reference points. A placement relation is a special composition of position relations to specify the relative position of two general point clouds, or higher level structures by eliminating all kinematical degrees of freedom.

A morphological relation operates with two substance relations, namely, it simultaneously manipulates the surface normal vectors and the local curvatures at the reference points of two boundary particles in order to set the requested morphological characteristics for the surfaces belonging to the concerned particles. Morphological relations depend on (a) the orientation of the surface normal vectors (i.e. parallel, perpendicular, and skewed), (b) the direction of the surface normal vectors (i.e. identical or opposing), (c) the positional relation (direct or indirect), and (d) the type of contact (point contact, edge contact and face contact).

Kinematical relations describe relative motions of two particles or higher level structures through the degrees of freedom between them. They also regulate the extents or limits of relative motions. Translate, spin, revolve and roll need to be defined as basic kinematical relations.

Deformation relation is an internal relation that is normally defined between two particles or two particle clouds within one component. It specifies the strain and stress caused by a given displacement of the reference vector of a particle with respect to the reference vector of another particle. Strain-stress tensors are used to represent the elastic, visco-elastic, elasto-plastic and plastic deformation of various materials. In the case of plastic deformation, the tensor is decomposed to elastic and plastic components. The plastic component of the stress tensor is calculated by introducing yield functions of hardening/softening of plastic material.

Kinetic relations can be used to define the motion of components caused by forces, or collisions between two components. Physical effect relations describe mechanical effects such as friction and adhesion due to mechanical forces. Finally, physical field relations are oriented to only one particle (i.e., the source of the effect is not important from a design point of view, only the target). This is the case with fields such as gravity, sunlight, and heat conduction.

4. REPRESENTATION OF ARTIFACTS ON VARIOUS STRUCTURAL LEVELS

It was mentioned earlier that a particle manifest either as a boundary particle, or as a volume particle in a nucleus. Thus, a nucleus containing boundary particles can be formally described as $N = (B\Pi_i, B\Pi_j, \Phi_{ij}, S_k)$, where $B\Pi_i$ and $B\Pi_j$ are the concerned boundary particles. Boundary particles are characterized for their reference point(s), contact surface, surface normal vector and volume. The contact surface of a BΠ is represented by a half space indicating the material domain of the particle. Volume particles (VΠ) are characterized for their reference point(s) and volume.

A nucleus establishing relations between a boundary particle and a volume particle can be formally represented as $N = (B\Pi_i, V\Pi_j, \Phi_{ij}, S_k)$, where BΠi is as above, and $V\Pi_j$ is a volume particle. A nucleus can represent the relationships between one-one particle of two artifacts, that is, $N = (B\Pi_{p,i}, B\Pi_{q,j}, \Phi_{ij}, Sk)$, where $B\Pi_{p,i}$ is a boundary particle of artifact p and $B\Pi_{q,j}$ is a boundary particle of artifact q. $B\Pi_{p,i}$ is called a native boundary particle of artifact p, and a complementing boundary particle of artifact q.

An advantage of the nucleus concept is that it allows us to define particles only logically defined at the beginning of artifact conceptualization, when there no idea about the geometry, morphology and structure yet. These particles can be redefined later to be metric objects with geometric and structural properties. Another advantage is that a nucleus can stand for a particle cloud (a natural surface), a mechanical component (a structured set of surfaces), an assembly (an aggregate of components), and a system (an assembly with functional changes) as well as for an interlinked pair of particles. When a nucleus represents higher level constructs such as a component, an assembly or a system, the particles are only logically defined. If there is a need to manipulate these constructs on the lowest level, that is, on the level of concrete particles, extra nuclei have to be added in which the particles are represented as metric entities. This mechanism lends itself to a kind of multi-resolution modeling.

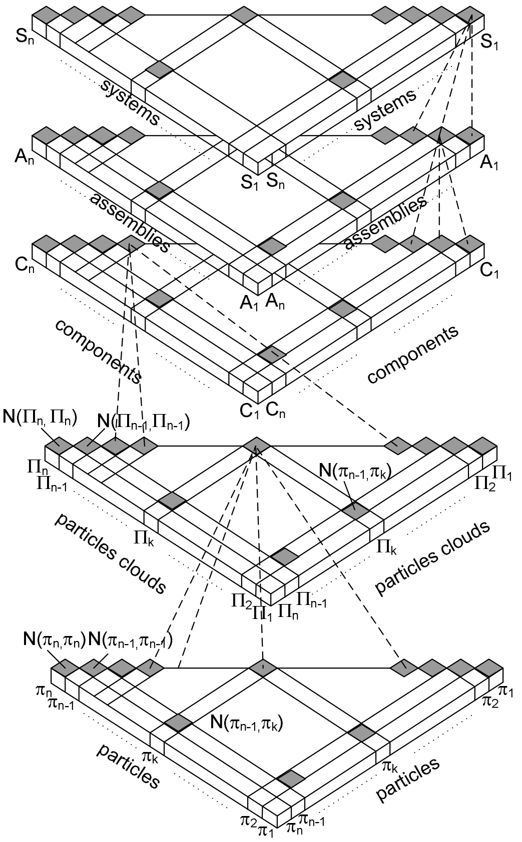

Fig. 4. Scheme of representing a hierarchy of nuclei

To handle the above mentioned five levels of artifacts in the database, nuclei are stored in five layers of triangular matrices (Fig. 4). In theory, n·n-n/2 non-reflexive and n reflexive nuclei can be defined. Hence, each matrix contains n·n+n/2 elements, where n is the number of particles being in relation in a given level. The reflexive relations are stored in the main diagonal of the matrix. Relations between the nuclei of different levels are defined by “has a” connectedness relations.

5. BEHAVIOURAL SIMULATION AND CONTROL OF SIMULATION

Using a finite set of particles we can generate a discrete representation of an artifact. This discrete model makes it possible for us to specify relations between two boundary particles, a boundary particle and a volume particle, and two volume particles of the same or different artifacts, keeping in mind the constraints originating in the physical reality. We can simulate the behavior of a nucleus or an artifact based on the mathematical formulation of relations. The process incorporates the computation of the status of each nucleus in all specified situations. The behavior of the investigated nucleus or artifact is derived from the time-dependent changes. That is, each physical relation implies a set of elementary processes that add up in the computable behavior.

The behavior, Θ, of a nucleus in a given situation can be specified as:

$$\Theta(N) = \Gamma \{S_k (\Pi_i, \Phi_{ij} \Pi j)\},$$

where ($\Pi_i, \Pi_j \in \Pi, \Phi_{ij}$ and S_k are as earlier, and Γ is a generic behavior generator function, which takes into consideration the interaction of various nuclei and the influences on each other's behavior. The introduction of Γ to handle the interactions of nuclei is necessary, since the observable behavior of a modeled design concept is the aggregation of the elementary behaviors of the nuclei. Since all nuclei might interact in a composition, this aggregation should be represented as a Descartian product rather than as a Boolean union of the observable elementary operations:

$$\Theta(CN) = \Theta(N_i) \times \Theta(N_j), \text{ or}$$

$$\Theta(CN) = \mathbf{\Pi} (\Theta(N_i), \Theta(N_j))$$

where CN is a composition of nuclei, and $\mathbf{\Pi}$ denotes a mathematical product.

For behavioral modeling and simulation, arbitrary number of physical relations can be specified between pairs of particles. Our goal is however not only to be able to simulate the behavior of an artifact, but also to be able to control the simulation of the behavior. For this purpose we need to introduce other concepts which belong to a scenario-based simulation. In general a scenario is a logically arranged sequence of activities or happenings. In order to name a complex structure of situations of nuclei (□), the sets of initial conditions (IC), boundary conditions (BC), and procedural conditions (PC), the notion of setting has been introduced. In our case, a scenario arranges and operates on settings. A situation is an arrangement of the particles and the relations in between them, taking into consideration the external circumstances. In a simulation process, the situations are changed and the changes for each nucleus are computed based on the specified relations and constraints.

A scenario, Σ, prescribes a sequence of situations, in which the interactions of the nuclei incorporated in a design concept happen. That is,

$$\Sigma = \cup (S_k)$$

With these, the behavior on the level of nuclei is $\Theta(N) = \Gamma (\Sigma \{ Ni \})$,

or, on the level of relations is:

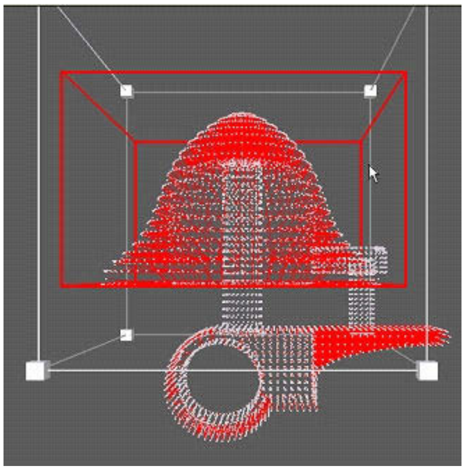

Fig. 5. Simulation model based on particles and relations

$$\Theta(CN) = \Gamma (\cup (Sk (\Pi_i, \Phi_{ij} \Pi j)))$$

The behavior of an artifact is simulated in a particular setting by alternative scenarios, which the designer might want to realize or avoid. The composition of all alternative scenarios forms a scenario tree. From a computational point of view, scenario tree is a branching graph showing the alternative logical structure of □ and parameter values of IC, BC, and PC. The significance of a scenario-based behavioral simulation is in that it enables us to model any logically and procedurally possible series of happenings in various settings. In terms of the simulation it means that it can to some extent be open-ended (as far as it is made possible by the tree of settings). The investigation of alternative scenarios means processing of a selected subset of branches in the scenario tree. An exhaustive simulation means investigation of the scenario tree as a whole. To run a simulation first an artifact model has to be developed, and the requested relation between the particles specified (Fig. 5). For the numerical simulations the initial states should be specified in the scenarios, from which the simulation algorithms can calculate the course of physical processes.

6. DATABASE SCHEME AND MANAGEMENT

From a programming point of view, the nucleus is a complex data and relation structure that includes geometric, structural, morphological, material and physical aspects. In the development of the database scheme of nucleus-based modeling we wanted to profit from the fact that relations can be arranged in a hierarchical structure according to their content (meaning). The developed database scheme, shown in Fig. 6, favors to the application of object oriented

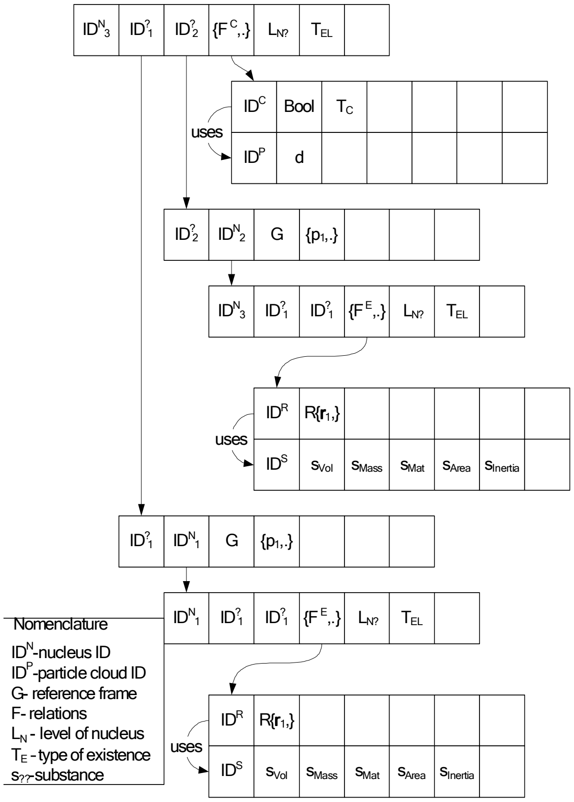

Fig. 6. External database scheme of nucleus-based modeling

programming of the nucleus-based conceptual design system. On the lowest level of the database scheme, nuclei are defined as couplings of two particles, either boundary or volume. Figure 6 shows the data fields directly included in the data structure of a nucleus. The access to the coupled particles is implemented through pointers to the data fields. The geometric data are specified through the substance relation to the existing nuclei of the concerned artifact (mechanical part). It means that the reference vectors of the particles, the surface normal vectors, the mass and material attributes can be specified using reflexive relations. The couplings between distinct artifacts are described by binary relations, which are stored in a common list of relations.

The basic laws of mechanics or physics are incorporated in the nuclei as default relations. This makes it possible for designers to run simulations without having to specify all the relations between the modeling entities individually and without having to derive the equations in question. This is unlike, for instance, constraint-based modeling, where the user has to define all the relations in the model. Nucleus-based modeling allows conformity between the visual feedback from the modeling and simulation system, and the physical appearance of the intended artifact.

7. CONCLUSIONS

The fundamentals and the means of a nucleus-based modeling methodology have been presented as a vehicle for the development of novel front-end systems for conceptual modeling and simulation. The nucleus-based modeling focuses on the relations, rather than on the geometry of artifacts. Nucleus-based modeling can also be employed in detail designs and simulation. In conceptual design, design concepts of artifacts are decomposed to interrelated nuclei. A nucleus is a purposeful coupling of two particles through a set of non-conflicting physical relations. The time history of the relations implies elementary processes that are the basis of the behavioral simulation. Nucleus-based modeling can easily deal with multi-level representations without the need to fully specify every level. It can represent complex components (e.g., supplied subassemblies, such as an electromotor) as a 'black box' without the need to model its internals. It can handle vague shapes, since the geometry of the particles can be vaguely defined.

8. REFERENCES

[1] French, M. J., *Conceptual Design for Engineers*, Design Council, London, 1985.

[2] Hsu, W., Liu, B., Conceptual Design: Issues and Challenges, *Computer-Aided Design*, Vol. 32, No. 14, 2000, pp. 849–850.

[3] Horváth, I., Conceptual Design: Inside and Outside, in *Proceedings of the 2nd International Seminar and Workshop on Engineering Design in Integrated Product Development - EDIProD 2000*, Rohatynski, R. ed., UZG, Zielona Gora, Vol. 1, 2000, pp. 63-72.

[4] Horváth, I. and W. F. van der Vegte, Nucleus-Based Product Conceptualization: Principles and Formalization, in *Proceedings of ICED '03, Stockholm*, August 19-21, 2003, pp. 1-10.

[5] Rusák, Z., *Vague Discrete Interval Modeling for Product Conceptualization in Collaborative Virtual Design Environments*, Ph.D. Thesis, Millpress, Rotterdam, 2003.

[6] Gero, J. S., Computational Models of Creative Design Processes, in *Artificial Intelligence and Creativity*, ed. by Dartnell, T., 1994, pp. 269-281.

[7] Mahoney, D.P., 2000, Multiphysics analysis, Computer Graphics World, 23 (6), pp. 44-46, 50, 52.

[8] Alber, R. and Rudolph, S., Constraint-Based Conceptual Design And Automated Sensitivity Analysis For Airship Concept Studies, in *Proceedings of Deutscher Luft- und Raumfahrtkongress 2002*, Stuttgart, September 23-26, 2002, DGLR Jahrbuch 2002, Band 3.

[9] Klein, R., The Role of Constraints in Geometric Modelling, in *Geometric Constraints Solving and Applications*, ed. by Bruderlin, B. and Roller, D., Springer-Verlag, Berlin, 2001, pp. 3-23.

[10] Anantha, R., Kramer, G. A. and Crawford, R. H., Assembly Modelling By Geometric Constraint Satisfaction, *Computer-Aided Design*, Vol. 28, No. 9, 1996, pp. 707-722.

[11] Horváth, I., Kuczogi, Gy. and Vergeest, J. S. M., Development and Application of Design Concept Ontologies for Contextual Conceptualization, in *Proceedings of 1998 ASME Design Engineering Technical Conferences DETC'98*, September 13-16, 1998, Atlanta, Georgia, CD-ROM: DETC98/CIE-5701, ASME, New York.

[12] Alber, R., Rudolph, S. and Kröplin, B., On A Grammar-Based Design Language That Supports Automated Design Generation And Creativity, in *Proceedings IFIP WG5.2 Workshop on Knowledge Intensive CAD (KIC-5)*, Malta, Malta, July 23-25, 2000.

[13] Rusák, Z., Horváth, I. and van der Vegte, V., First Steps Towards An All Embracing Relations-Based Modeling, in *Proceedings of ASME 2004 Design Engineering Technical Conferences and Computers and Information in Engineering Conference DETC'04*, September 28 – October 2, 2004, Salt LakeCity, Utah, CD-ROM DETC2004/CIE-57735, pp. 1-12.

[14] Van der Vegte, W.F. and Horváth, I., Consideration and Modeling of Use Processes in Computer-Aided Conceptual Design, *Transactions of the SDPS - Journal of Integrated Design & Process Science*, Vol. 6, No. 2, 2002, pp. 25-59.