Towards a Collaborative Function-to-Form Mapping

Denis Pallez, Parisa Ghodous, Christel Dartigues

Towards a Collaborative Function-to-Form Mapping

D. Pallez
University of Lyon1, PRISMa, France

P. Ghodous & C. Dartigues
University of Lyon1, LIGIM, France

ABSTRACT: The technological advances carried out these last years in the field of products development led the researchers to elaborate approaches that reduce the cost and time of product development, enhance the quality of product and help the designers to be more creative. Currently, computer aided systems have concentrated on the capture and representation of geometrical shape and technical information as opposed to provide supports for product design in the earlier stages of design process. The aim of this paper is to present a methodology to aid designers during the first stages of design. The standard models using the STEP standard are proposed in order to ensure co-operative and collaborative works during the first stages of design. Then, we apply our methodology in the case of a bottle design by considering multiple designers' viewpoints.

1 INTRODUCTION

The technological advances carried out these last years in the field of products development led the researchers to elaborate the approaches that reduce the cost and time of product development, enhance the quality of product and help the designers to be more creative. These objectives are difficult to obtain due to a large number of phases, which should be carried out during the product development and the large number of experts of different disciplines that are involved. Concurrent Engineering is considered as one of the key concepts that enable companies to reach these objectives (Ghodous et al. 2000a).

Concurrent engineering is a wide field of research. Researchers who are interested in concurrent engineering work on different aspects such as (Prasad 1997):

- Philosophical aspect, which deals with the boundaries of the responsibility and the authority, culture and organization management,
- Methodological aspect, which deals with system thinking, approaches to system complexity, systems engineering, product realization taxonomy and system integration,
- Conceptual aspect, which deals with concurrency and simultaneity, modes of concurrency and co-operation, work flow mapping and
- Virtual aspect, which deals with capturing life cycle intent and information modeling.

Our work is situated on the methodological and virtual aspects. In product design, each expert works with his own applications, handles his own data, has knowledge and constraints, which are specific to his domain, and has his own point of view on the product (Ghodous et al. 2000b). This heterogeneity implies many problems, in particular information exchange and sharing with the other experts. It is thus necessary to make it possible to each expert to represent adequately its data while facilitating integration and communication of their data with the other experts. In this context, our objective is to develop a concurrent and collaborative system allowing all experts to participate on product development as soon as possible and help them to work together (Dartigues et al. 2000). In the next, we firstly present the function to form mapping approach as a conceptual design technique. Then, we explain why it is more realistic to consider this methodology in a collaborative context and we present enhanced version of the function to form mapping approach presented in (Gardan et al. 1999a, Gardan et al. 1999b, Pallez et al. 2001). In addition, we present the data models that are necessary to achieve a collaborative and conceptual design aided system based on this methodology. Due to reasons of normalization, these models are described using the EXPRESS-G formalism (ISO 1994). Finally, we present how the collaborative function-to-form mapping could be used in a water bottle design.

2 THE FUNCTION TO FORM MAPPING AS A TECHNIQUE FOR CONCEPTUAL DESIGN

The aim of any design is to develop the product

as earlier as possible. In a non–routine design, it is delicate and extremely complex to obtain the best products answering customer's specifications. Indeed, a great number of different experts of different disciplines participate in product design. The difficulty lies in the fact that those persons have to collaborate. Shape is today the main representation of a product, even though the current trend is to remove geometry from its central position in order to add high-level information (Gorti et al. 1996, Szykman et al. 2000, Szykman et al. 2001). Therefore, the function to form mapping appears to be one of the most important activities of the design process and up till now happened manually. This activity is very important both for choices that are made and for the amount of work that it represents.

That is why we have the desire to assist designers in the first stages of product design. The objective is not to construct automatically the shape but both to automate a certain number of heavy and tiresome tasks, and assist designers during the first stages of design. In best case, this assistance makes the designers' stimulation possible by presenting them solutions that they had not thought before.

We suppose that the first information we have at our disposal are included in the customer's specifications, which are expressed in natural language (Figure 1a). Due to the functional decomposition (Figure 1b) eventually using functions of making, maintaining, prevention, control (Keuneke 1991) and allowing (McDowell et al. 1996), behaviors of the product could be identified (Figure 1c) in order to respect the philosophy of FBS (Ranta et al. 1996, Tomiyama et al. 1993, Umeda et al. 1996). Next, with the help of the same philosophy, the structure of the product can be obtained. This previous model can be identified as a set of product's components linked by assembly relation (Figure 1d). Actions achieved by designers in the traditional FBS methodology (represented by Figure 1a to Figure 1d) are currently manual. Assisting designers during those phases is difficult due to the fact that manipulated information are mainly expressed in natural language. That is why we attempt to formalize some design information in order to partially automate the shape mapping (Figure 1e to Figure 1g).

In some cases, we have noticed that a function can also be decomposed into a set of constraints on *physical parameters*, called *intermediate specifications* (Figure 1e). Only such design cases will be considered in next sections. Decomposition that consists in translating functions into constraints on physical parameters can correspond to the "function → structure" reasoning techniques identified in (Hsu et al. 1998). *Physical parameters* are defined as quantifiable and measurable entities referring to the physical world. For instance, in the case of a box design, aesthetics functions may be converted as the following: the ratio of length to height of the box approaches the gold number $\left((\sqrt{5}+1)/2\right)$. The corresponding constraints would be length = $\left((\sqrt{5}+1)/2\right)$ /height. In a larger extent, the handling of a water bottle by a human being, which is a function, may be converted into constraints on the weight, the compactness and so on, which are parameters. Physical parameters used in the intermediate specifications are called *intermediate parameters* (Gardan et al. 1999a, Gardan et al. 1999b). They are of a rather high level and are not necessarily related to geometry. Like parameters and specifications, constraints established in the intermediate specifications are called *intermediate constraints*. In (Gardan et al. 1999b), we have defined an intermediate constraint by a quadruple <*IP*, *R*, *Exp*, *W*> where *IP* is an intermediate parameter, *R* is a relation among {<, >, =, ≠} that must be considered as fuzzy relation, *Exp* is an arithmetic expression and *W* is the relative weight of the intermediate constraint in comparison with the other constraints of the intermediate specifications. For the moment, only one designer gives the weight for all the intermediate constraints.

Once a great number of intermediate constraints are manually deduced from the functional decomposition obtained by the FBS methodology, we propose

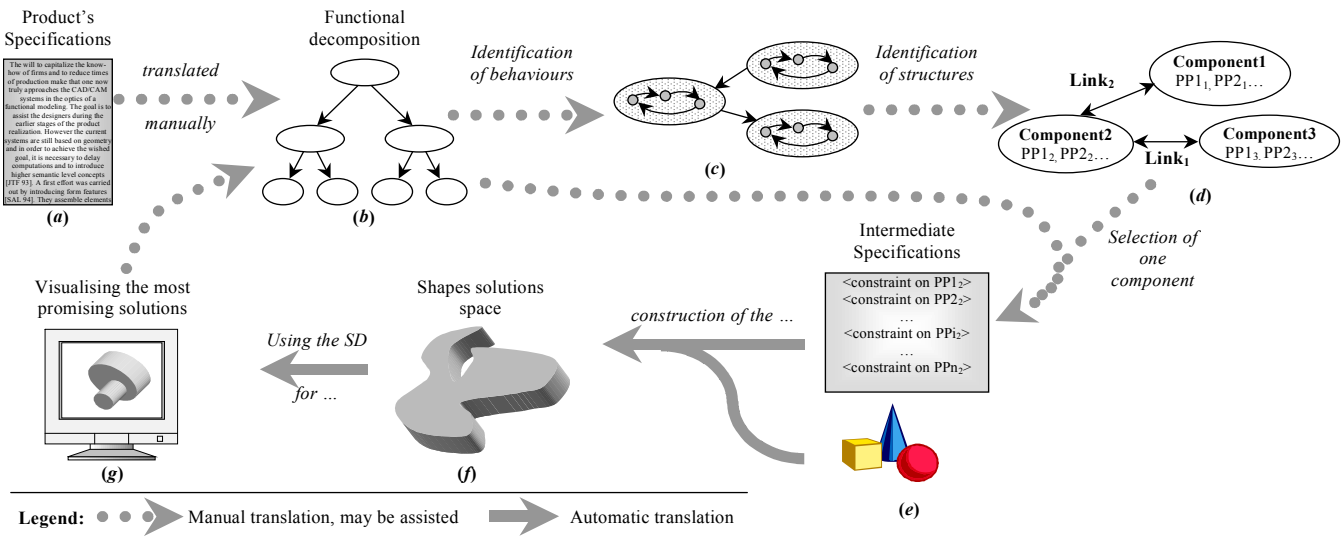

Figure 1. Overall architecture of the system

solution shapes to designers that achieved those intermediate constraints. As those constraints are deduced from functions, solution shapes satisfy functions.

In order to reduce the complexity, the function-to-form mapping is first applied on non-assembly components. The product's shape may be obtained by using the function-to-form mapping for each component of the product structure's and by assembling shapes of each component. Some assembly problems will still remain and it will be necessary to define a new kind of intermediate constraints that permit to take into account assembly problems. This axis of research is not treated in this paper. As a consequence, we assume afterwards that a component is a rigid, finite and homogenous solid that contains no other components.

So, our methodology consists in (Gardan et al. 1999a) proposing a set of solutions, called *solutions space* (Figure 1f). Each shape of the solutions space is deduced and constructed from:

- intermediate constraints;
- a library of parameterized shapes;
- algorithms that translate intermediate constraints into constraints on parameters which depend directly on the parameterized shape chosen in the library. Those constraints are called *terminal constraints* in order to distinguish them from intermediate constraints. Parameters that define a parameterized shape and used in terminal constraints are called *terminal parameters*. For instance, the radius of a sphere, the width, the length and the height of a box are terminal parameters. One intermediate constraint can generate several terminal constraints. As a consequence, we have to compute the intersection between the all generated terminal constraints into intervals of variation for each terminal parameter (Gardan et al. 1999a). For instance, if an intermediate constraint generates a terminal constraint such as "length greater than 5" and another intermediate constraint generates another terminal constraint such as "length less than 10", the system needs to compute those terminal constraints in order to obtain values that are possible for each terminal parameters. Then, previous algorithms must be seen as knowledge of the design domain because they represent expert's rules.

Once one knows algorithms that translate intermediate specifications into constraints on terminal parameters, a parameterized shape can be chosen in the library and the computation of variation intervals generates solution shapes. Next to this, one can consider that several shapes could be proposed to the user in so far as the design problem is underconstraint. So, we must detect the shapes, which best satisfy the intermediate constraints (Figure 1g). To do that, we have defined a Satisfaction Degree $SD_{ic}(s)$ of an intermediate constraint *ic* for a given shape *s* as a real number in the [0,1] interval (Gardan et al. 1999a). This number expresses the quality with which *s* satisfies the intermediate constraint *ic*. If $SD_{ic}(s)$ is near zero, the constraint is badly satisfied, if it is near one, the constraint is well satisfied.

One has to consider the function-to-form mapping as a means of generating solutions shapes for one component that may give ideas to the designer. We applied this methodology within the framework of a filling system for foundry mould design (Gardan et al. 2001). We validated the methodology by taking the specificity of the foundry (trade features) into account. We showed that this approach was applicable in this case and that we obtained quickly results better than those based on an expert's knowledge.

In fact, the method presented in this section (§2) was firstly thought for only one designer. Nonetheless, in real products design, it will be difficult for one designer to apply this methodology due to the large number of constraints to manage.

A more suitable methodology would be to distinguish between intermediate constraints from a design domain and intermediate constraints from another design domain. As a consequence, we treat in the next section the way to apply our function-to-form mapping in a collaborative context with the aim of reducing productivity time but also with the aim of managing a lot of expert's points of view.

3 METHODOLOGY APPLIED IN A COLLABORATIVE CONTEXT

As it seems to be unrealistic to design the product shape only by one designer due to a large number of design domain that take part in product design, we propose to put the previous methodology in a collaborative context.

First of all, the different stages that consist in translating product's specifications (Figure 1a) into intermediate specifications (Figure 1e) and corresponding to the FBS methodology are unchanged except the fact that there is at least one expert per design domain. Consequently, we can group the intermediate constraints based on the same design domain together. This is the starting point of our collaborative methodology.

We have already discussed about collaborative function to form mapping in (Pallez et al. 2001) where the methodology consisted in generating a solutions space for each design domain and then merging solutions spaces into one using a suitable technique that has to be defined (Figure 2). The proposed steps were to find the most promising solutions of the merged shapes solutions space and to show them to designers. As it is utopian to consider that the solution could be found in one time, the possibility to change or to expand the functional decomposition is

given to designers. As a consequence, information (behaviors, structure and so on…) deduced from functions also may have changed. Consequently, designers have to modify information deduced from functions and to apply all the steps again and again until to find a shape solution that satisfies all the design domains.

The methodology represented by the diagram of the Figure 2 is possible only if an expert of a design domain is capable of generating shapes solutions from intermediate constraints. That is to say, the expert is capable of defining its own primitive shape library. However, one can find some design domains where experts cannot generate any solutions as regards geometry. Nevertheless, those design domains might be linked with the shape of the design product even if the experts are not able to generate shapes solutions. That is why we have to extend our collaborative methodology in a more general context; i.e. the methodology has to be applicable in a lot of design domains, even in design domains in which experts do not have shapes library.

By this way, we present in next paragraphs an improvement of our collaborative methodology presented in (Pallez et al. 2001). The following methodology has to be used when expert of a design domain do not have a shape library.

With previous regards, the collaborative function to form mapping should be the following:

1 First of all, each design domain has to define its own intermediate specifications (IS) for only one component of the product to design. The intermediate constraints are deduced from the functional decomposition of the design product regardless of the other design domains participating in the design process. As a consequence, in this step, intermediate specifications of a domain will use only physical parameters of the domain. By this way, a designer of a domain can be considered as an expert of this domain in contrary of the methodology presented in §2 in which only one designer have to manage with physical parameters coming from different design domains.

2 Next, as we are convinced that there exist relationships between constraints from one design domain to another, this step consists in establishing those constraint's relations. There is several ways to achieve this. The first one is the traditional and well-known manner: experts of all design domains involved in the design process have to find manually links between all the intermediate constraints. The next manner that could be considered is a semi–automatic one: during the design process, intermediate constraints are mainly deduced from the functional decomposition. One can construct the following rule: if two constraints from two different domains are deduced from the same function, then one can consider that they are related. However, this rule is not sufficient because relations may exist between two constraints deduced from two different functions.

Now, consider for instance the design of a water bottle in order to illustrate a relation between two constraints from different design domain: a constraint on the transparency (intermediate parameter manipulated by a chemist) may be related to a constraint on the thickness (intermediate parameter manipulated by a manufacturer). The nature of relations may be mathematical or other kind of relations.

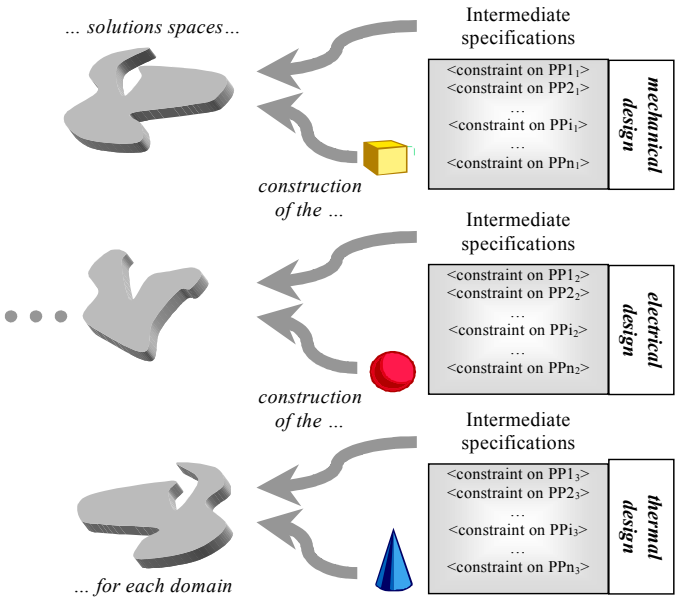

Figure 2. Collaborative methodology proposed in (Pallez et al. 2001)

3 This step corresponds to the solutions space generation. It represents the more critical step of the function to form mapping because product's needs are not directly linked with product's shape. In fact, a shape satisfies many functions (represented by the intermediate constraints in our methodology) and conversely a function could be satisfied by many shapes. However, in order to ensure a mapping in concrete application, some assumptions are made to simplify this methodology. We attempted to give an answer to this problem in (Gardan et al. 2000) by considering different approaches such as stochastic or morphing ones. Nevertheless, for the moment, we assume that only experts of design domain who have a shape library are in charge of proposing solutions by applying the methodology presented in §2.

4 Once shapes solutions are generated, experts of design domain who do not have a library must react to the proposed solutions by participating in the selection of the most promising solutions. So, in this part, experts from all design domains participate in selecting solutions. This selection uses the Satisfaction Degree (Gardan et al. 1999a and presented before) in order to evaluate each solution for all intermediate constraints from all de-

sign domains.

5 As it is inconceivable that a promising solution could be find after the first try of shapes solution generation and selection, we consider that intermediate specifications have to be more refined in order to propose most promising solutions to designers. As a consequence, designers have to modify their intermediate specifications in collaboration with designers from other design domains in so far as some constraints are related to other constraints from other design domains. So, in this step, designers may modify the weights of constraints and/or add other intermediate constraints and/or add other parameterized shape in their library. The function-to-form mapping process starts again from the second step until a promising solution satisfies all the designers involved in the product design.

The methodology presented above must be seen as a progressive and iterative method in which designers are assisted by a computer application that proposes solution shapes. As experts have to exchange data in order to ensure the collaborative function–to–form mapping, we discuss in the next section standard models that capture concepts related to this methodology.

4 PROPOSED MODEL FOR REPRESENTATION OF EXPERT VIEWPOINT

The analysis of current works on product modeling shows that current single fixed representations are inadequate to model the various concepts present in multidisciplinary product development situation. Consequently, the dynamic representation of multiple views of a product based on functional contexts seems to be necessary.

Depending on the view taken, certain properties and descriptions of the object become relevant. A comprehensive model of a product must be able to built depending on the particular need.

Consider the example of the mechanical part. The mechanical design model of this product is different of electrical engineer's model or thermal model. Any model should allow a dynamic evolution and must be capable of accommodating multiple concepts unambiguously and consistently so that the elements could not be duplicated. Any inconsistencies between the various models have to be discovered and corrected. This process may go through several iterations. The result is a set of models, one per consulting discipline, where, although each set represents the product using a different point of view, the comprehensive representation is consistent. There is no attempt to integrate the various sets into one set.

The basic description of a product differs from one viewer to another. Each view may represent a product with different elements and different composition hierarchies. No one model contains a full comprehensive description of the product but each model should be consistent regarding to the object being described. Different descriptions of the same elements and different subsets of these descriptions in different models exist.

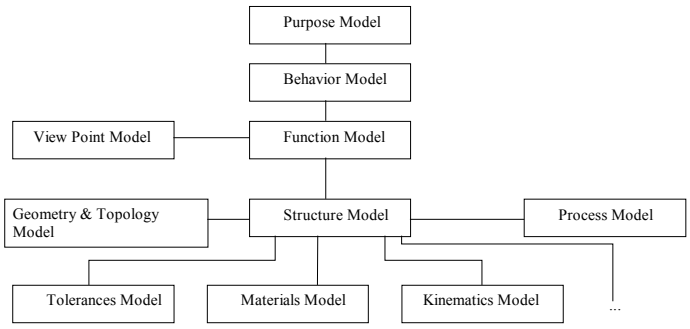

Figure 3. Some necessary models and their relationship

Some models used for product description are shown in Figure 3. We detail the model of intermediate specifications detailed in §2.

Viewpoint model represents the information about a product from a particular view and the relationships between the different viewpoints. Several functions are related to each viewpoint.

Purpose explains why an object does what it does and it is related to the human socio-cultural environment concept. The purpose model represents the purpose and the relationship between the purposes.

Function is what an object does. Functional model represents function and the different relationships between the functions.

Behavior is how the object does what is does. Behavior model represents the behavior, the relationships between the behaviors and the relationships with functions.

Structure is what the object is. We use the STEP standard to represent the concepts related to product data (the structure of product and its relations with other models such as shape, materials, tolerances, etc). The definition of a product in the STEP product data model is any physical object, which is produced by either natural or manufacturing processes. Any part or assembly that contributes to a product is also considered to be a product. A car is a product while its wheels and engine assemblies are considered as other products. Furthermore, each of these products can be further decomposed into smaller components or products.

The details of these models are described in (Ghodous et al. 2000b).

In this paper, due to the function form mapping technique, we have introduced the intermediate specifications model. We present the detail of this model in the following.

We represent the intermediate specifications concepts by EXPRESS-G formalism (Figure 4).

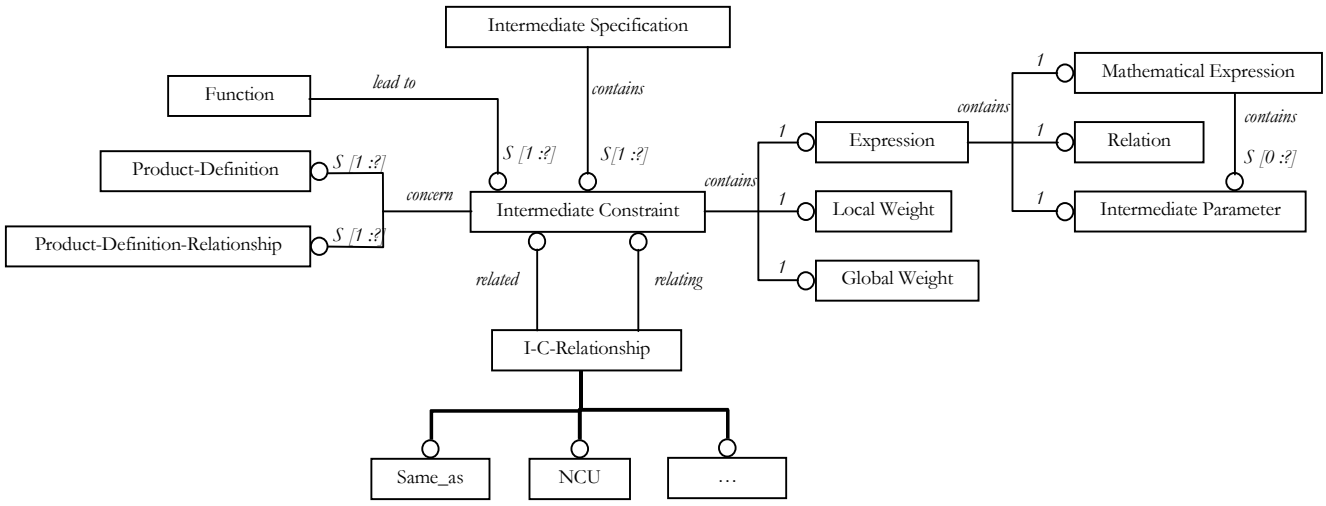

Figure 4. Intermediate specifications model expressed in an EXPRESS–G format

EXPRESS-G is a graphical language developed by ISO 10303 STEP (ISO 1994). The EXPRESS-G basic notations used in figures include *entities* (rectangles); *super-type/subtype relationships* (thick solid lines); *required attributes* (normal lines); relationship for *optional* attributes (dashed lines). Additionally, the direction of an attribute is symbolized by an open circle, where the circle represents the "many" side of a "one to many" relationship.

In Figure 4, the *intermediate specifications* are a set of different *intermediate constraints*. The intermediate constraints are elaborated from functions that characterize the product. As functions relate to a given view point (mechanical view point, electrical view point, thermal view point and so on), the intermediate constraints also relate to this same viewpoint. On the other hand, each intermediate constraint is defined on a component, i.e.: is related to the structure of the product (entities *Product-Definition* and *Product-Definition-Relationship*).

The intermediate constraints are expressed in the following way: first of all, an intermediate constraint described by an expression. This expression contains an intermediate parameter (which can be the volume, the weight or any engineering parameter), a relation (<, >, =, ≠, etc.) and a mathematical expression, which can possibly include other intermediate parameters. With each intermediate constraint are associated two weights: a local weight and a total weight. These weights are used to give a list of priority of the constraints that must be carried out for the expert on a precise field (*local weight*) and for all of the experts (*total weight*).

Lastly, due to the fact that each expert handles his own whole of constraints, there are relations between them (*IC-Relationship*). These relations are of two types. We define the Same_as relation, which expresses the fact that two experts handle the same constraint. Their expression and their total weight will be then identical, but their local weight might be different. We also define the NCU relation (*Next Constraint Usage*), which means that an intermediate constraint can be compose of one or more other constraints.

5 CASE STUDY: A WATER BOTTLE EXAMPLE

In this section, we present the way to use our methodology and our models in a real case, which is the design of a water bottle.

First of all, we have to define information obtained manually during the first stages of design, that is the elaboration of functions, behaviors, structure of components and intermediate specifications (Figure 1a–e). The considered functions are: to hold one liter and a half of drinking water (1), not to be heavy (2), to be handled by a human-being (3), to be stable (4), to be compressible (5), to be packaged (6), to stand a force of 50 Newton (7), to be impervious to natural daylight (8), to be fine-looking (9), to be emptied (10), to be filled (11). All those functions have to be refined in order to obtain a functional decomposition; however, the eleven functions are enough to understand our methodology in the next. Secondly, the design process recommends extracting behaviors from functions. As we did not express how to model behaviors, we will leave this step out in so far as it permits to deduce components of the design product which are in our case the body and the cap of the bottle. In this section, we only consider the shape design of the bottle's body in so far as we assume that we design the shape of the product component by component without regarding assembly problems (see §2). The next step concerns precisely the elaboration of intermediate specifications. As in all product design, we have to take into consideration several viewpoints. In the case of a water bottle design, we consider three design domains: an expert on materials who may be a chemist, an expert on geometry and a manufacturer. It is obvious that it is not an exhaustive list of design domains. Figure 5 represents what could be the inter-

EXPERT ON MATERIALS	EXPERT ON GEOMETRY	MANUFACTURER
1. *Granularity* ≥ 10 µm; 5; 10; **F1**	1. *Height* ≥ 2 × *Width*; 5; 10; **F1, F2, F3**	1. *Volume* ≤ 1 m³; 5; 7; **F5**
2. *Density* ≤ 1; 3; 6; **F2, F3**	2. *Width* ≥ 4 cm; 6; 10; **F1, F2, F3**	2. *Volume* ≥ 10 cm³; 2; 15; **F5**
3. *Boiling Point* ≤ 50°C; 5; 10; **F1, F8**	3. *Width* ≤ 10cm; 5; 8; **F1, F2, F3**	3. *Weight* ≤ 2 kg; 1; 10; **F6**
4. *Thickness* ≤ 2 mm; 8; 12; **F2, F5**	4. *Volume* ≤ 5 dm³; 5; 3; **F3**	
5. *Thickness* ≥ 1 mm; 1; 3; **F7**	5. *Height_centre_of_gravity* ≤ $\frac{Height}{2}$; 5; 10; **F4**	
6. *Illumination_inside* ≤ 100 lux; 5; 10; **F8**		

F1: to hold one liter and a half of drinking water,
F2: not to be heavy,
F3: to be handled by a human-being,
F4: to be stable,
F5: to be compressible,
F6: to be packaged,
F7: to stand a force of 50 Newton,
F8: to be impervious to natural daylight,
F9: to be fine-looking,
F10: to be emptied,
F11: to be filled

Figure 5. Intermediate specifications and considered functions for a water bottle

mediate specifications in the three concerned domains, that is to say results of the first step of our methodology (see §3). The intermediate constraints are expressed regarding concepts used in Figure 4. For instance, consider the intermediate constraint "*Height* ≥ 2 × *Width*; 5; 10; F1, F2, F3": '*Height*' represents an intermediate parameter for the expert on geometry; '≥' represents a relation in the expression; '2 × *Width*' represents a mathematical expression in which another intermediate parameter is used; '5' represents the local weight of the intermediate constraints; '10' represents the global weight of the same constraint; finally, 'F1, F2, F3' represent functions from which the intermediate constraints are deduced.

The second step of our methodology is the comparison of the intermediate constraints to themselves in order to identify relationships that could exist: for instance, there is a link between constraint 4 of the expert on geometry and constraints 1 and 2 of the manufacturer (Figure 5). In one hand, a 'Same_Relation_As' relation can be used between constraints 4 and constraints 1 because the same mathematical relation (≤) is used on the same intermediate parameter in the two constraints; on the other hand, a 'Same_Parameter_As' relation can be used between constraints 4 and 2. In addition, a 'Mathematical' relation could be used to link constraints 4 of the expert on geometry and constraints 2 of the materials' expert.

The third step of proposed methodology consists in using parameterized shapes library and experts knowledge to generate shapes that satisfy the intermediate constraints. At first, only the expert on geometry can generate a shape solution in so far as he is the only one who owns a shape library (Figure 6a). In our example, we consider that he could generate three kinds of solutions (Figure 6b): a hollow sphere, a hollow cylinder and a hollow box.

The next step allows the expert on materials and the manufacturer rejecting solutions represented by a sphere or a box because they are not stable and fine-looking enough.

Then, the last step allows the expert on geometry to modify his intermediate specifications by adding the constraints "*Nb_Contact_Point_Plane* ≥ 3; 5; 10; **F4**", which means the number of contacts between a point of the shape and a plane is greater than 3 to ensure the stability of the bottle (function 4) (Figure 6c). Moreover, all the experts are agree with the fact that the solutions are not handled enough. So the expert on geometry is proposed to add new parameters to the primitive shape of the library. Next to this, the step 3 of the methodology is applied. The possible resulting solutions of previous modifications are represented in (Figure 6d). It is understandable that the sphere could have been rejected because it is not really stable (function 4) and it is possible that the hollow box has been rejected because its dissatisfaction with too many intermediate constraints. By modifying and adding a certain number of constraints and by applying our methodology (Figure 6e), we may assume that the collaboration of all the experts during the design process will converge on good solutions that may be improved by experts' experiences in order to find the solution (Figure 6f).

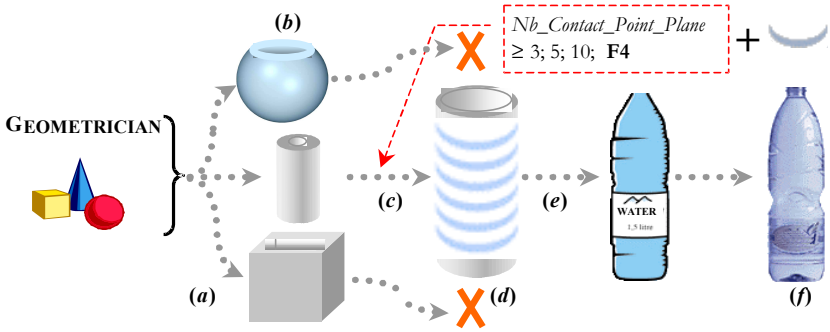

Figure 6. How the bottle shape is obtain using the methodology?

6 CONCLUSION AND FUTURE WORKS

The analysis of current works on product modeling shows that current single fixed representations are inadequate to model the various concepts present in multidisciplinary product development situation. Consequently, the dynamic representation of multiple views of a product based on functional contexts seems to be necessary. In this paper, we have studied the problem of representation of experts' multiple-view in a collaborative conceptual design environment. We have considered the function to form mapping approach as a conceptual design method and we have presented the way to apply this methodology in a collaborative context by considering a real example: a water bottle design with three different design domains.

Our future works are numerous. First of all, in short-term, it is necessary to identify all the possible relations between to constraints in order to allow more precise communication between experts of different design domains. Then, it is important to study how to maintain the consistency of models. We have defined some coherence rules between different models. Each expert at any time can define his model and may collaborate with the other models. As a consequence, when one model is manipulated, corresponding effects will be made automatically in the other. In that case, future works will focus on the definition and formalization of these rules to improve proposed multiple-view model.

In long term, future works are related to geometric reasoning: it could be interesting for the experts to define intermediate constraints using other kind of relation. For instance, instead of using well-known mathematical relations ($\leq$, $\geq$, $=$, $\neq$), the expert on geometry would like to use a "look like" relation ($\equiv$) in order to introduce new experiences on shapes. Once it will be done, it will be very interesting for designers, and especially for experts who have a shape library (expert on geometry), to combine shapes contained in the library so as to be more creative.

7 ACKNOWLEDGMENTS

We thank Christian Minich and Estelle Perrin from the CMCAO team of the university of Reims (France) who took part in the elaboration of the case study.

8 REFERENCES

Dartigues C., P. Ghodous, M. T. Martinez, and D. Vandorpe, (2000), "Modélisation collaborative des activités de conception", in *Proceedings of Conférence Internationale sur la CFAO, la Simulation et les Nouvelles Technologies de Conception et de Fabrication (MICAD'00)*, Paris, France, pp. 231-240.

Gardan Y., C. Minich, and D. Pallez, (1999a), "On shape to specifications adequacy", in *Proceedings of IEEE Proceedings of Information Visualisation (IV)*, D. Dornfeld (eds), Kluwer academic, London, Great Britain, July 14-17, pp. 315-320.

Gardan Y., C. Minich, D. Pallez, and E. Perrin, (1999b), "From Functions to Shapes", in *Proceedings of Third International conference on Engineering Design and Automation (EDA)*, Vancouver, Canada, August 1-4, pp. 287-294.

Gardan Y., C. Minich, D. Pallez, and E. Perrin, (2000), "Towards a specifications-to-shape translation tool", in *Proceedings of Third International Symposium on Tools and Methods of Competitive Engineering (TMCE)*, Delft, The Netherlands, April 18-21.

Gardan Y., Y. Lanuel, D. Pallez, and F. Vexo, (2001), "A methodology for a function-to-shape translation tool in foundry", *Computers in Industry,* vol. 44 (2) : 117-130.

Ghodous P., D. Vandorpe, (2000a), Editors of Advances in Concurrent Engineering '2000, Technomics Publishing, USA, ISBN 1-58716-033-1.

Ghodous P., M. T. Martinez, D. Vandorpe, (2000b), "Collaborative and standard design and manufacturing model", *International Journal of Computer Applications in Technology*.

Gorti S., Sriram, D. Ram, (1996), "From Symbol to Form: A Framework for Conceptual Design", *Computer Aided Design*, Vol. 28, No. 11.

Hsu W., Woon I., (1998), "Current research in the conceptual design of mechanical products", *Computer Aided Design*, Vol. 30, No 5.

ISO 10303-11, STEP Product Data Representation and Exchange (1994) `Part 11, Description Methods: The EXPRESS Language Reference Manual', International Organization for Standardization, Subcommittee 4, NIST.

Keuneke A. M., (1991), "Device Representation. The significance of functional knowledge", *IEEE Expert Intelligent Systems & Their Applications*, 6(2), pp. 22-25, April.

McDowell J., T. Lenz, J. Sticklen, and M. Hawley, (1996), "Conceptual design for polymer composite assemblies", *AI System Support for Conceptual Design, J. Sharpe, ed., Springer-Verlag* , pp. 377-389.

Pallez D., C. Dartigues, P. Ghodous, M. Martinez, (2001), "Data Architecture for Collaborative Conceptual Design", in *Proceedings of the 8th IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*, Antibes–Juan les Pins, France, October 15-18.

Prasad B., (1997), *Concurrent Engineering Fundamentals*, Vol I, II, Prentice Hall , USA.

Ranta M., M. Mäntylä, Y. Umeda, T. Tomiyama, (1996), "Integration of Functional and Feature-based product modelling – the IMS/GNOSIS experience", *Computer Aided Design*, 28(5), pp. 371-381.

Szykman S., R. D. Sriram, C. Bochenek, J. W. Racz and J. Senfaute, (2000), "Design Repositories: Engineering Design's new Knowledge Base", *IEEE Intelligent Systems, May/June*, pp. 2-9.

Szykman S., R. D. Sriram, W. C. Regli, (2001), "The role of knowledge in next-generation product development systems", in *ASME Journal of Computation and Information Science in Engineering*, 1(1).

Tomiyama T., Y. Umeda, and H. Yoshikawa, (1993), "A CAD for functional Design", *Annals of CIRP'93*, pp. 143-146.

Umeda Y., M. Ishii, M. Yoshioka, Y. Shimomura, and T. Tomiyama, (1996), "Supporting conceptual design based on the function-behaviour-state modeller", *Artificial Intelligence for Engineering Design*, Analysis and Manufacturing, 10, pp. 275-288.