“Top-down” design methodology utilizing abstraction of mechanical interfaces

Peter Goranov[1], *Desislava* Georgieva[1*], and *Elena* Todorova[1]

[1]FTMD, Faculty of Mechanical Engineering, TU-Sofia, Bulgaria

Abstract. CAD systems are a key design tool. As technology evolves, they provide greater functionality and solve tasks that are more complex. Consequently, the question of their effective use is becoming increasingly important. The present paper discusses the "top-down" design method. During the initial stages of a design process, the geometric description of the components is missing or it is not complete which calls into question the use of CAD. The present work proposes a “top-down” methodology that includes CAD in the early stages of the design process. The discussed methodology takes advantage of the object-oriented approach. There is originated a hierarchical scheme of objects, each of which inherits the methods and attributes of its higher class. This allows design information to spread throughout the whole design process as well as its integration. With an implementation of this model, CAD models of an assembly and its parts can be created on the basis of the information available when the product concept is being developed. The geometric description recorded in the CAD model elaborates as the design project evolves. In this way, the formation of the CAD model goes hand in hand with the clarification of the product design.

1 Introduction

CAD systems are the key tool that mechanicians use. With the development of technology, they provide more functionality and are capable to solve tasks that are more complex. Thus, the question of their effective use is becoming increasingly important.

Two basic ways to create the geometric model of an assembly commonly referred to as “design methods”, can be defined:

- “Bottom-up” design.
- “Top-down” design.

The "bottom-up" design assumes that the 3D geometric models of the parts are known and they just need to be assembled. As an opposite, “top-down” design corresponds to the natural manner that new products are created. During the initial stages of the design process, the geometric description of the components is unknown or not complete. This calls into question the use of CAD, because they require the existence of 3D geometric models.

In the present work a “top-down” methodology is proposed, which involves utilization of CAD in the early stages of the design process. This allows product data from the all stages of the creative process to be integrated, which means information that is contrived during the initial stages of the design process distributes to its next stages. In this way, the productivity of CAD is increased and the design intention is secured up to the final stage of embodiment design.

2 Literature review on CAD design methodology

According to [1] the CAD model should contain the design intention. This means that it should be retained notwithstanding subsequent model changes. Modern CAD systems do not ensure compliance with all explicitly introduced design constraints.

In [2] is discussed how CAD systems are employed and possibilities for increasing their efficiency are explored. A method that centralizes engineering information in CAD models, based on skeleton modelling, is advanced. Two types of skeletons are defined: master and slave. The components are positioned in the assembly relative to the base skeleton. The purpose of slave skeletons is to recalculate the assembly model when dimensions of components are changed.

According to [3] the relationships between components are not explicitly given in a CAD model. An algorithm for extracting these data in a form of assembly constraints is developed that includes the type of constraints and associated features. This information is

[*] Corresponding author: desyst@abv.bg

subsequently used to generate assembly sequences automatically.

Similar issues are addressed in [4]. The structure of CAD assembly model is being considered as a set of hierarchical links between the main assembly and its subassemblies or parts. Problems, addressed in mentioned work, are solved by extracting this information from the assembly model.

In [5] a method for complexity management, which performs a hierarchical decomposition of the system and its modules as well as their interfaces are developed. The decomposition takes place at the following levels: user requirements, system specification as subsystems and components, physical and logical configurations that realize the desired functions, analysis of system performance in terms of the intended functions.

Usually, simplification of a CAD model includes minor geometric features such as chamfers and fillets with respect to a single component. [6] proposes a method of model simplifying, which preserves the geometric features that are used for positioning and the associated assembly constraints.

Data related to conceptual design is usually limited to a specification of the requirements and schemas of the system architecture. This creates problems in the information flow where information streams are broken or parts of information are re-entered. These issues are addressed in [7], where a development environment is discussed that integrates conceptual design data into CAD/PDM applications.

Research in [8] aims at increasing CAD performance by reducing extra drudgery workload. This is achieved by creating high-level CAD templates that are linked to the "top-down" methodology. As additional results here is pointed out the possibility of creating precise models at the conceptual stage and their maintenance during the whole design process.

The effectiveness of CAD is also seen in [9], but in terms of modifying and reusing 3D models. A methodology for creating geometric models, where the geometric elements are independent from one another and thus allows easy modifications, is proposed.

The creation of an assembly succeeds the design of the parts, so the information of the assembly is not processed in the same quantity and quality. Therefore, [10] suggests an assembly-oriented design. This approach makes it possible to manage inter-assembly relationships employing the "top-down" design principle. The information is created at the high levels of the structure and then conveyed to the lower levels.

The existence of large number of various scientific researches in the field in consideration means that there are many unresolved problems related to geometric CAD design and opportunities to improve CAD effectiveness.

3 Description of the problem

The "top-down" design method is considered as essential when a new product is created. A number of commercial CAD systems, such as [11], offer support for this approach. The basic functionality includes: working with component skeletons, component envelopes, assembly layout as conceptual sketches from which 3D models can be subsequently created.

These solutions typically deal with separate aspects of various tasks such as space distribution between components, positioning of the main components in the assembly, and kinematic analysis. Here are discussed problems related only to embodiment design and the proposed methodology attempts to cover all components of the product and to offer a complete solution regarding the creation of a 3D assembly model with an application of the "top-down" approach.

The discussed methodology is based on the object-oriented approach. A hierarchical scheme of objects, each of which inherits the methods and attributes of its higher class, is created. This allows distribution and integration of design data at all stages of the design process. The abstraction levels of the geometric interfaces that are discussed are:

- Item list.
- Extended structural scheme.
- Skeleton of parts.
- Geometric features.

4 Object-oriented model

4.1 Object-oriented model of an assembly

An object-oriented model of an assembly at the abstract level is proposed in [12]. This model is shown in Figure 1 in UML notations [13]. The model contains the classes as follows:

- The class "Assembly" expresses a CAD assembly

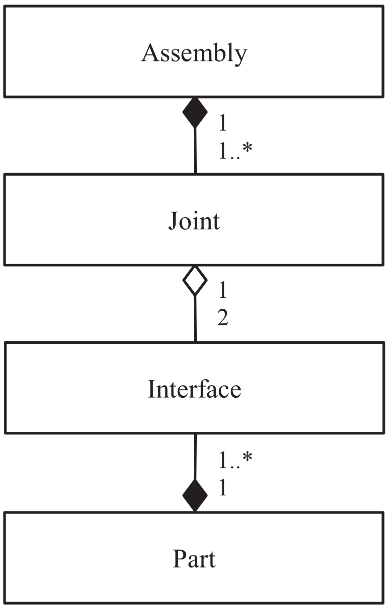


Fig. 1. Object-oriented model of assembly – abstract level.

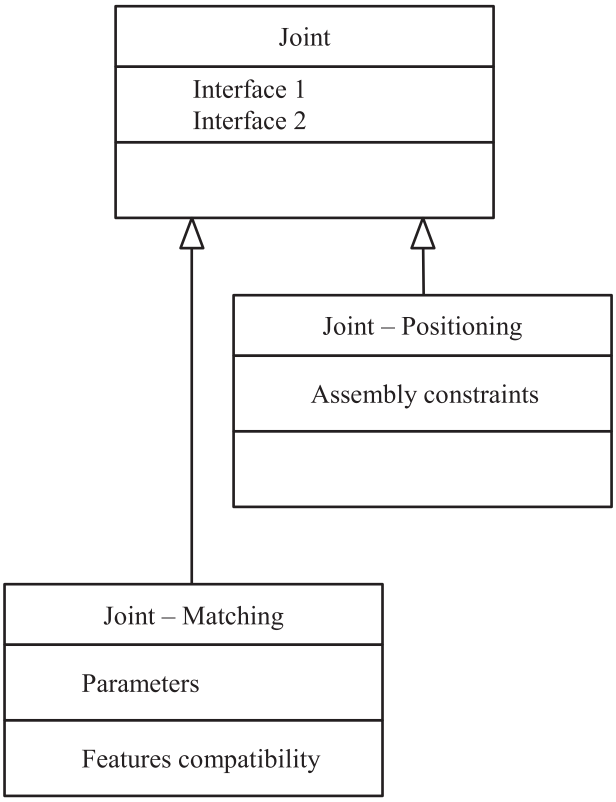


Fig. 2. Specialisation of class "Joint".

model and it comprises all connections between its parts. From the point of view of CAD systems, the assembly model (a file of an assembly model) contains only a description of how the parts are oriented towards each other. The complete geometric description of the parts is contained in their 3D models (files of part models). It is assumed that an assembly has to contain at least one joint and the number of joints is not limited.

- The class "Joint" contains a full description of how the details are coupled. The parts are coupled by their mechanical interfaces. It is assumed that each joint involves exactly two interfaces that belong to different parts.
- The class "Interface" holds a description of the mechanical interfaces of a part. Mechanical interfaces include the geometric features that involve the part in joints. In order to be a member of an assembly the part has to have at least one interface. One part can participate in many joints, so the number of its interfaces is not limited.
- The class "Part" contains the geometric description of a part. From CAD point of view when an assembly model is created only mechanical interfaces of parts matter. The rest of the complete geometric description of parts formally is not related to the assembly so their geometry can be completed independently.

The discussed assembly model is abstract and provides only a basic idea of the logical structure of an assembly. In order for this model to be used for practical application, it is necessary the above-mentioned classes to be concretised.

4.2 Specialisation of class "Joint"

Parts are included in an assembly by creating joints. The joints work for both mutual orientation of parts and transfer of mechanical forces. In the assembly model shown in Figure 1, the parts are considered at the highest level of abstraction and that is their name. At this level, the assembly is represented by its structural scheme, which sets out the logical relationships between the constituents. These relationships can be considered as the highest level of abstraction. The specialization of these relationships (Class "Joint") is illustrated in Figure 2.

4.2.1 Specialisation on level "Positioning"

The positioning of parts in an assembly can be done in different ways. According to one of them parts are located in an assembly whereby their coordinate systems become fixed. Thus, a model of an assembly can be created if complete geometric description of its components is not available, but actually, no mechanical relationships (joints) are created. This way, important features are lost such as, for example, automatic adaptation of the assembly to changes in components dimensions, it is not possible to perform kinematics analysis, etc.

Due to the considerations mentioned, it is expedient to position parts through geometrical constraints that are imposed – in Figure 2 this is the class "Joint – Positioning". This class contains the assembly constraints that are related to the particular joint.

4.2.2 Specialisation on level "Matching"

Formally creating a model of an assembly is related to the correct positioning of parts relative to one another. From engineering point of view, this is not enough. In addition, the geometric parameters of the mechanical interfaces of the jointed parts must be consistent so that the product can be assembled and work properly.

The mutual conformity between mechanical interfaces includes both compatibility between geometrical features and their dimensional equality. In Figure 2, this mutual conformity is achieved through the class "Joint – Matching". This class appends information about the parameters of the joint, which must be the same for the two interfaces. It also checks whether the geometric features that make up the two interfaces can form a joint.

It should be noted that the compatibility of the interfaces can be done independently of the positioning of parts. The information on the imposed geometric constraints is not necessary in the case, therefore, the class "Joint – Matching" is a direct successor of the abstract class "Joint".

4.3 Specialisation of class "Interface"

The existence of a connection in the structural scheme means that the both of two linked parts must provide mechanical interfaces that can form a joint. The implementation of these joints requires a specialisation of logical relationships in the structural scheme – Figure 3.

4.3.1 Specialisation on level "Positioning"

The class "Interface – Positioning" should provide the opportunity for an object of class "Joint – Positioning" to be created, i.e. it has to provide geometric features on which assembly constraints may be imposed.

It is assumed that at this level the complete geometry is not known. Geometric features are developed that allow the functionality of the parts to be analysed. These features generally form the skeleton of parts. Therefore, the class "Joint – Positioning" adds information about the skeletons of parts that are represented by reference geometry – planes and axes. Thus, the logical links in the structural scheme are specialised as physical links that withdraw the degrees of freedom.

4.3.2 Specialisation on level "Matching"

Synchronisation of mechanical interfaces involves solving the following tasks:
- Matching the shape of features.
- Matching the parameters (dimensions) of features.
- Coordinate the location of features.

Mechanical interfaces should therefore be considered at their complete geometry. Since the rest of the part geometry does not need to be known, the interfaces are alienated from the other geometric features.

The class "Interface – Matching" (Figure 3) appends information about the geometric features that are constituents of the interface design as well as their parameters. In order to create a joint, it is not just enough to match the interfaces, but they have to be appropriately positioned. Therefore, the set of parameters also includes those that specify the location of the geometric features, which is relative to the reference features that form a joint of level "Positioning" (class "Interface – Positioning").

4.4 Application and analysis of the object-oriented model under consideration

4.4.1 Application

The object-oriented model offers different levels of abstraction of an assembly and its components. This corresponds to the development of the concept of products during the design process. The vague notion of the product gradually becomes more precise to its full geometric description. According to the model under discussion, three main levels of specification can be distinguished:

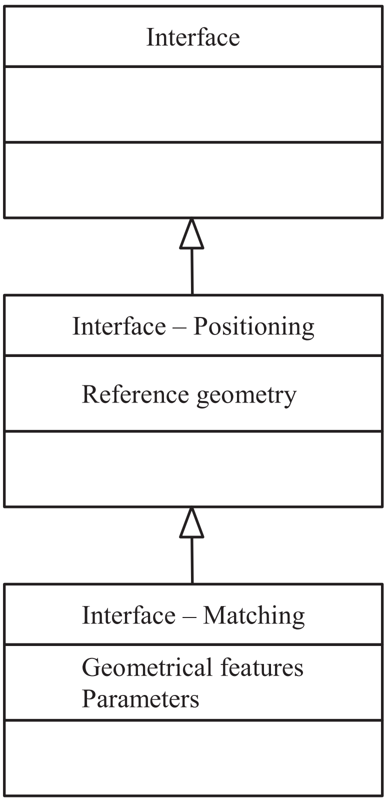


Fig. 3. Specialisation of class "Interface".

- Abstraction level – geometric information is missing or individual ideas are available in the form of conceptual sketches. Parts are presented with their names and the documentation is the structural scheme of products.
- Positioning – a model of the assembled unit, in which the parts are positioned by imposing assembly constraints, is created. There is also no complete geometric description at this level, and the parts are represented by their skeletons, which contain reference planes and axes. Assembly constraints are imposed on parts skeletons.
- Matching – parameters of mechanical interfaces are mutually accorded. In order to ensure this, the mechanical interfaces must be fully defined as a set of geometric features. The layout of mechanical interfaces is determined by the skeleton of parts.

Based on the discussed object-oriented model, CAD assembly model can be created in the following sequence:
- The structural scheme is developed.
- Depending on the functional purpose each logical connection in the structural scheme is represented by a set of reference geometric elements – planes and axes.
- The skeletons of parts are created.
- The assembly model is created by imposing a "mate" constraint on the relevant elements of the skeletons of parts.

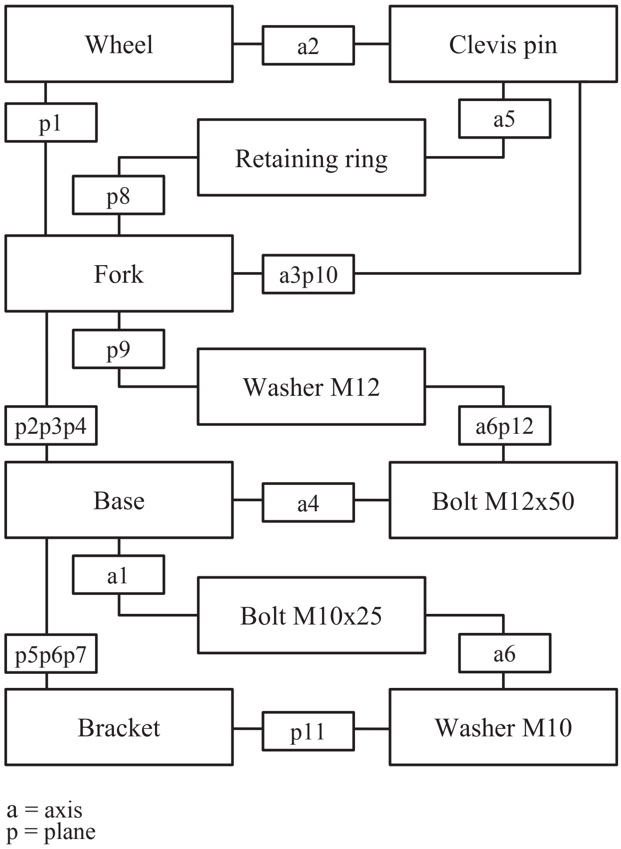


Fig. 4. Extended structural scheme of "Guide pulley" with designated mechanical interfaces on level "Positioning".

- A complete geometric description of the parts interfaces is developed and associated with their skeletons.
- A complete geometric description of all parts is developed.

4.4.2 Analysis

Standard parts and gears have a significant share of the components of mechanical assemblies. However, they are not paid a special attention by the current version of the discussed object-oriented model.

In principle, standardized components can be dealt with as any other original components. In this case, the library of standardized components has to be supplemented with definitions of the class "Interfaces". It should be pointed out that the dimensions of the standardized components can only have predefined values that require a particular logic to match the parameters.

Gear trains can also be included in the model under consideration but for this purpose they should be presented as a single component (subassembly). Full support of gear trains (including kinematics study) requires geometric constraints of the type "Motion" to be imposed, which the present model is not provide with.

Regardless of the possibility of standardized components and gear trains to be included in the model under consideration, their representation is insufficient. In order to ensure complete support of these components it is necessary special objects in the model to be defined, which will be subject to future research.

5 Implementation with an example

The succession of developing a CAD model of assembly "Guide pulley" is discussed here as an example of the application of the theory under discussion. The two tasks – positioning and concordance are discussed separately.

5.1 Positioning of parts in the assembly model

In general, the structural scheme of a product incorporates only its main components. Therefore, the concept of "extended structural scheme" is introduced in [14], allowing a complete model of the product to be created at the conceptual stage of the design process.

The extended structural scheme contains all parts and geometric relationships between them. In terms of CAD, a geometric relationship means an imposed assembly constraint. These constraints are associated with geometric items that belong to different parts. It should be noted that in the initial stages of the design process there is no actual geometry and the parts are represented by their skeletons. Skeletons are a set of reference planes and axes. The reference features can be associated with either real surfaces on which the parts are oriented and fixed, as well as virtual features such as planes and axes of symmetry.

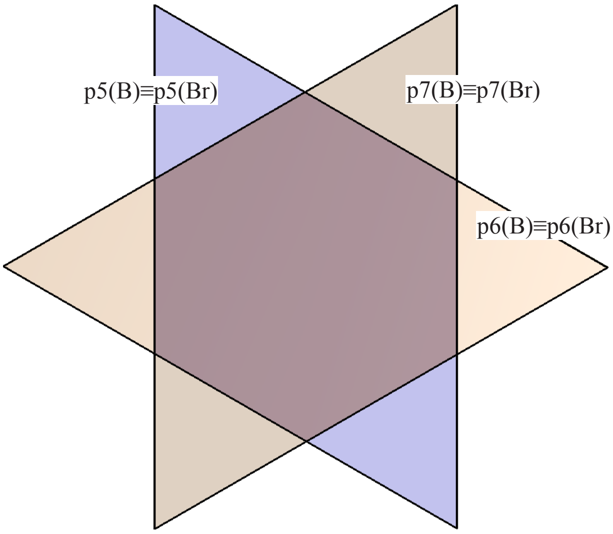


Fig. 5. Assembly model of "Guide pulley" (that includes only parts "Bracket" and "Base") where skeletons of parts are generated and assembly constraints are imposed.

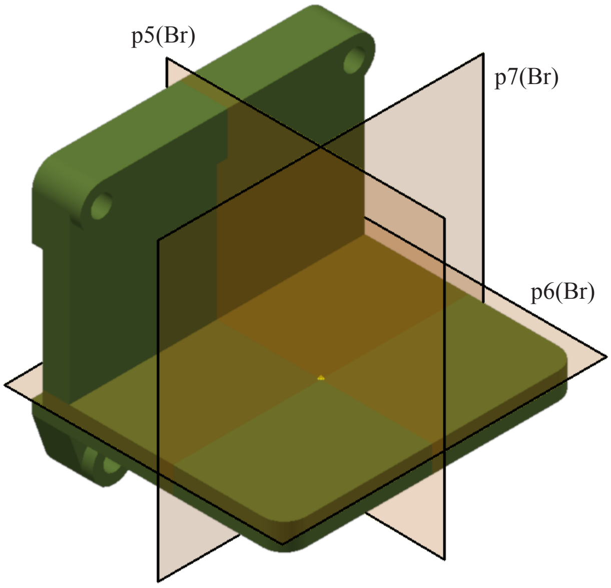


Fig. 6. Model of part “Bracket” where the reference planes, which position part “Base”, are redefined properly.

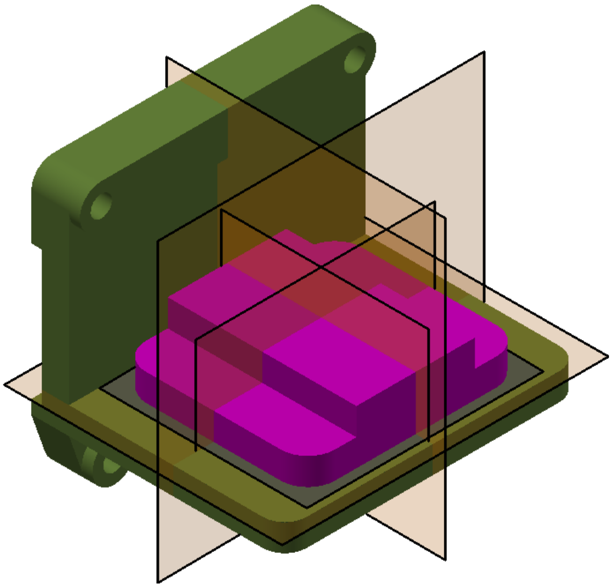


Fig. 8. The assembly model of Figure 5 as soon as the models of parts have been created.

Figure 4 shows the extended structural scheme of the mentioned assembly "Guide Pulley". A connection in the structural scheme means the withdrawal of degrees of freedom. As is well known, a solid body in space has 6 degrees of freedom. Each of these connections may take between 1 and 6 degrees of freedom. How, in a particular case, the degrees of freedom are taken depends on the functional purpose of parts and this is a decision of the designer.

The extended structural scheme of Figure 4 includes the reference geometric features with which the parts are positioned. The sets of reference geometric features represent the geometric interfaces of parts at the level “Positioning”.

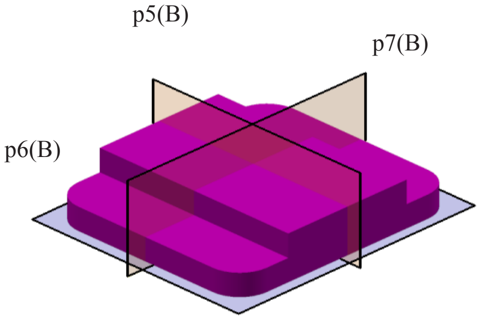


Fig. 7. Model of part “Base”.

For each connection in the structural scheme according to the geometric interfaces mentioned above, a pair of reference geometric features is generated. They are either planes or axes and they are ingredients of the skeletons of the two related parts. In the assembly a “Mate” constraint is imposed on each of these pairs of reference geometric features.

The assembly model generated from the information of the structural scheme is shown in Figure 5. It contains all the parts, each of which is given with its own skeleton. The assembly model only contains a list of parts and imposed assembly constraints, which is exactly the information shown in Figure 5.

Initially, the reference geometric features are in their default position, usually at the beginning of the coordinate system. Once the final geometric model of the part has been created, reference planes or axes, where the other parts are positioned, are redefined in order to occupy the anticipated location. This is done for the part “Bracket” in Figure 6. The geometric model of the part "Base" is appropriately designed with respect to the reference planes on which it is oriented – Figure 7.

When an assembly model is created by imposing geometric constraints, it automatically adapts to a change in parts geometry. Thus, after the actual geometry of the parts is developed, the assembly model on Figure 5 is automatically recalculated in the proper model, which is shown in Figure 8.

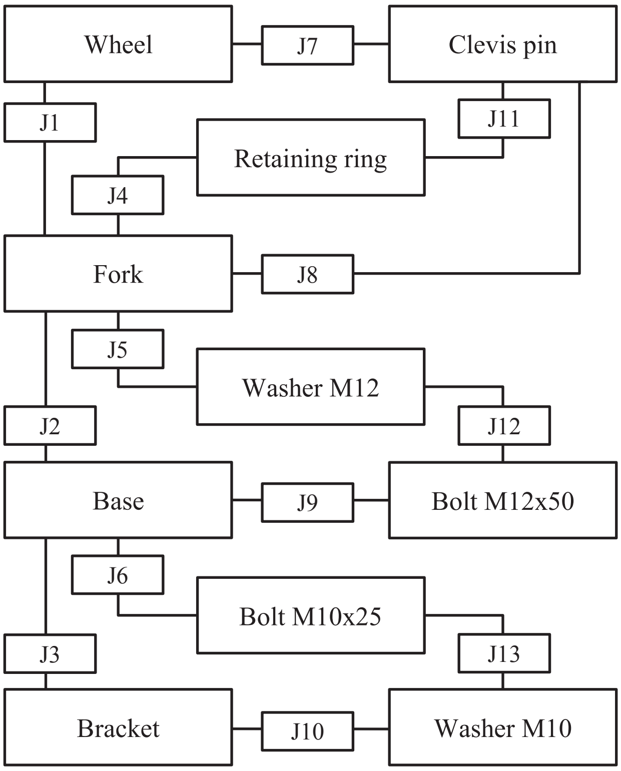


Fig. 9. Extended structural scheme of "Guide pulley" where joints are designated.

5.2 Concordance of geometric interfaces

In order to coordinate the geometric parameters, it is necessary to represent the joints as objects of the class "Joint – Matching" and respectively, the interfaces as objects of the class "Interface – Matching". It should be pointed out that while it is not necessary to create objects of class "Joint – Positioning", the pre-formation of objects of class "Interfaces – Positioning" is mandatory because the class "Interfaces – Matching" is the successor of the class "Interfaces – Positioning". The design information for the objects "Joint – Matching" is recorded in the extended structural scheme as is shown in Figure 9.

As an example, Figure 10 shows the structure of a particular object "Joint" denoted in the structural scheme (Figure 9) as "J2". The object incorporates the information as follows:

- Pointers to objects of class "Interface – Matching" that create the joint in question.
- Parameters that include dimensions of geometric features and their location.
- A method that checks the compatibility of the associated interfaces (whether they can create a joint).

The same figure shows the structure of the both interfaces involved in the joint under consideration. It includes:

- The name of the part to which they belong.
- Pointer to its parent object of class "Interface – Positioning".
- The geometric features that constitute interface.
- Pointers to the parameters (dimensions and location) recorded in the associated joint.

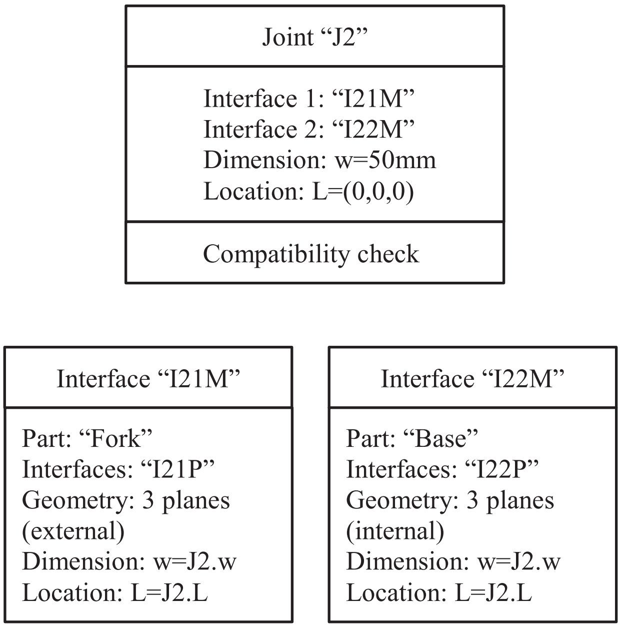
The use of pointers in the both interfaces, that refer to the parameters defined in the same object "Joint", guarantees the compliance of their parameters.

Conformity between geometric parameters of the two interfaces is not sufficient they can create a joint. It is also necessary that they be in a suitable place in the space of the assembly. For this purpose, the location of the geometric interfaces is determined by the reference features (ingredient of part skeletons) which orientate the two parts between which the joint under consideration is formed.

It is assumed during the conceptual design the type of joints between parts, which are necessary to provide the intended functions of the assembly, is specified. For each of these joints the geometric interfaces of the parts are developed. As a final result objects of class "Joint" and class "Interface", that contain the data shown in Figure 10, are created. The presence of these data allows mechanical interfaces in models of parts to be created formally.

As illustrated in Figure 11 and Figure 12, the geometric models of the part "Fork" and the part "Base" with mechanical interfaces generated are shown. These models can serve as a basis for building the complete geometric description of the parts.

Figure 13 shows how the assembly looks like when the parts are presented with their geometric interfaces.


Joint "J2"

Interface 1: "I21M"
Interface 2: "I22M"
Dimension: w=50mm
Location: L=(0,0,0)

Compatibility check

Interface "I21M"

Part: "Fork"
Interfaces: "I21P"
Geometry: 3 planes (external)
Dimension: w=J2.w
Location: L=J2.L

Interface "I22M"

Part: "Base"
Interfaces: "I22P"
Geometry: 3 planes (internal)
Dimension: w=J2.w
Location: L=J2.L

Fig. 10. Object Joint "J2" and associated Interfaces "I21" and "I22".

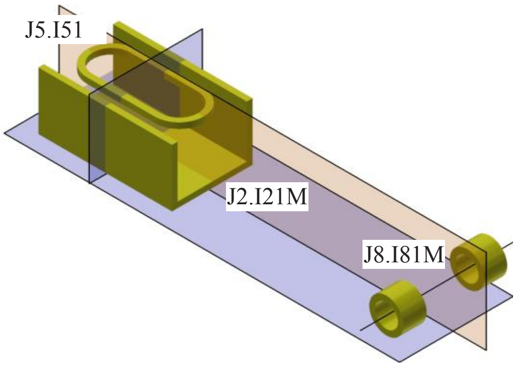


Fig. 11. The model of part "Fork" where mechanical interfaces are materialized (objects of class "Interface – Matching").

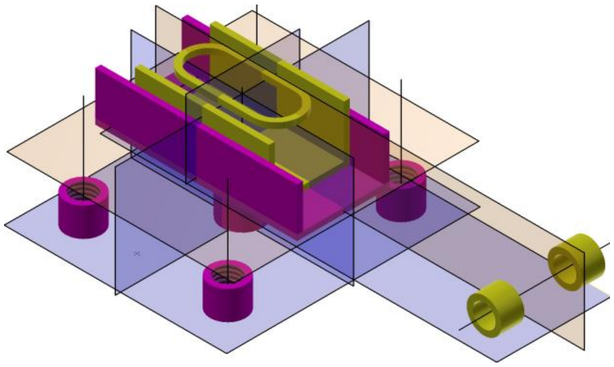


Fig. 13. Assembly model of "Guide pulley" (that includes only parts "Bracket" and "Base") where mechanical interfaces are materialized.

The ways in which a match of geometric parameters is achieved depends on the CAD system used. For example, Autodesk Inventor [11], which allows the geometrical parameters to be saved in an MS Excel spreadsheet and subsequently to be linked with geometric models, may be mentioned.

6 Conclusion

An object-oriented assembly model is discussed which defines the geometric interfaces at different abstraction levels. These abstraction levels correspond to the different stages of a design process. The implementation of this model allows the information that is available when the concept of a product is being developed to be introduced in the CAD models of an assembly and its parts. With the progress of the design project the geometric description recorded in the CAD model, grows bigger. In this way, the creation of the CAD model goes hand in hand with the clarification of the design of the product.

The discussed approach allows a great degree of independence at building of the geometric description. Assembly constraints are imposed on reference geometric features that are ingredients of parts skeletons. This provides freedom in creating the actual geometry because it does not affect the position of the parts. Complementing skeletons of parts with geometric representation of mechanical interfaces actually includes all the essential constraints that secure the function of parts – the parts can be assembled and work properly.

The pointed out independence in creating the

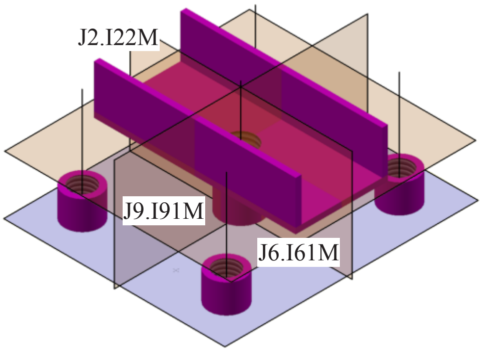


Fig. 12. The model of part "Base" where mechanical interfaces are materialized (objects of class "Interface – Matching").

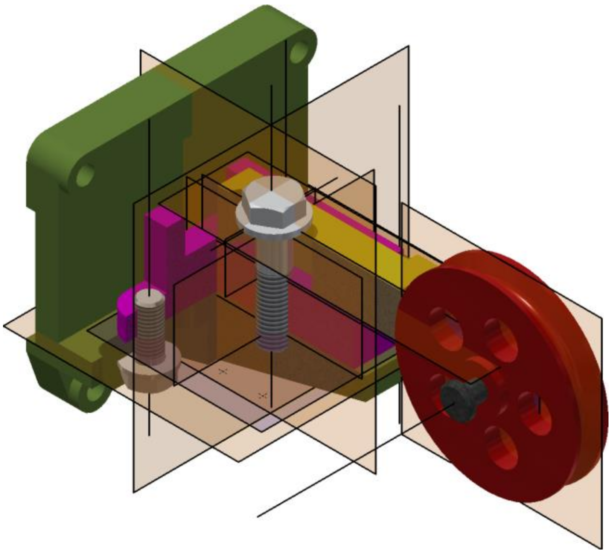


Fig. 14. The assembly model (section view) of Figure 5 after the accrual geometric description of parts have been created with the parts skeletons visible.

geometric description is important in large scale or cooperative design projects where many professionals are involved. When the concept of the product has been developed, the final geometric description can be shaped simultaneously and independently of different designers. Besides, the integrity of the assembly model is automatically ensured.

The easy modification management can be considered as another asset. Geometric parameters of the part interfaces can be controlled centrally by an electronic version of the extended structural scheme. These parameters can be easily changed, however the model consistency is retained. The structural scheme allows the links between the parts to be traced and hence the spread of modifications to be determined.

As a special case of modification management, it is possible to replace easily one part with another. This is essential when configurable products are created. The condition of replacing one part with another is the new part has to have the same skeleton and the same mechanical interfaces.

Finally, it will be pointed out that the object-oriented models of the joints and the mechanical interfaces create formal definitions that allow a high degree of automation. Once the structure of a product has been created and joints and mechanical interfaces are being developed, it is possible part skeletons, an assembly model and geometric descriptions of interfaces to be generated automatically.

The authors would like to thank the Research and Development Sector at the Technical University of Sofia for the financial support.

References

1. J. Otey, P. Company, *Proceedings of the ASEE Annual Conference & Exposition*, ID #9556 (2014)
2. Y. Bodein, B. Rose, E. Caillaud, *CAD*, **45**, 1198 (2013)
3. M. Trigui, R. BenHadj, N. Aifaoui, Int J Adv Manuf Technol, **79**, 1465 (2015)
4. I. Belhadj, M. Trigui, A. Benamara, Int J Adv Manuf Technol, **87**, 2829 (2016)
5. T. Habib, H. Komoto, Mechatronics, **24**, 788 (2014)
6. J. Yu, H.J. Xiao, Zhang, H. Cheng, B. Xin, Int J Adv Manuf Technol, **68**, 2335 (2013)
7. V. Torres, J. Ríos, A. Vizán, J. Pérez, *AIP Conf. Proc.*, **1431**, 22 (2012)
8. K. Amadori, M. Tarkian, J. Ölvander, P. Krus, Adv Eng Inf, **26**, 180 (2012)
9. J. Camba, M. Contero, P. Company, CAD, **74**, 18 (2016
10. M. Vielhaber, H. Burr, T. Deubel, C. Weber, S. Haasis, *Proceedings of DESIGN 2004,* 539 (2004)
11. http://help.autodesk.com/view/INVNTOR/2019/ENU/
12. D. Georgieva, *Scientific proceedings of the Scientific-technical union of mechanical engineering "INDUSTRY 4.0"*, **27**, 26 (2016)
13. https://en.wikipedia.org/wiki/Unified_Modeling_Language
14. E. Todorova, *Scientific notifications*, **10/93,** 309 (2006) (in Bulgarian)