Proceedings of the ASME 2010 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2010
August 15-18, 2010, Montréal, Canada

DETC2010/28805

ENRICHING ASSEMBLY CAD MODELS WITH FUNCTIONAL AND MECHANICAL INFORMATIONS TO EASE CAE

Gilles Foucault
Jean-Claude Léon
Laboratoire G-SCOP (Grenoble-Sciences pour la Conception, l'Optimisation et la Production).
INPG - UJF - UMR CNRS 5272.
46 Avenue Félix Viallet
38031 GRENOBLE Cedex 1, France
Gilles.Foucault@ujf-grenoble.fr
Jean-Claude.Leon@grenoble-inp.fr

ABSTRACT

Assembly models can be regarded as a kernel for product development processes where they can efficiently contribute to many product simulation behaviors. Assembly models are often containing 3D B-Rep CAD models, possibly with geometric constraints between the components and bill of materials. However, these models are often difficult to process for simulations because algorithms often face a very large diversity of configurations. One origin of such difficulties can be found in companies' practice where components may be represented differently from one company to another and their interfaces as well. In any case, interfaces between components are not explicit, which leads to tedious model processing tasks. This paper illustrates preparation of assembly models to ease CAE through an analysis of company practices, showing that a concept of conventional representations is an important starting point to efficient treatments of assemblies. In addition, it is described how interfaces and conventional representations can be combined to derive functional and mechanical information from geometric models of components. Illustrations of the proposed approach is given throughout the paper using various standard components.

1 Introduction

Assembly models are widespread and key elements in product development processes where they often form reference data for companies. Design reviews strongly rely on assembly models to assess design solutions, modifications. However, when it comes to simulations, assembly models become complex to process to produce the desired simulation models whether structural behavior, immersive simulations, etc., are targeted. Indeed, geometric models of assemblies reduce, most of the time, to a collection of component models whose connections have to be transformed to meet the simulation model requirements. Interactive operators are the basis for these transformations, which leads to tedious tasks. Analyzing the content of digital mock ups as encountered in industry is a starting point to structure assembly models so that they can incorporate technological / functional data that can be processed to generate more efficiently the simulation models.

Form features and features have been set up extensively to structure component models and help processing them at various stages of a product development process [1]. Often, the user has to add external information to define features or form features, which is efficient but may increase the overall model processing time because feature parameters must be specified interactively. Generally speaking, adding technological data to components or assemblies is efficient [1], [2] and small amounts of data may be enough to significantly reduce the combinatorial complexity of assembly processing. Following this approach, a key issue appears that would be to identify the core data (technological, mechanical, ...) from which feature parameters, attached to components or assemblies, can be derived using technological, mechanical, ..., properties.

Combining geometric models with reasoning processes proved also to be efficient to enrich geometric models of assemblies [3], which can be combined with the position of components to define some kinematical linkages and start structuring assembly models.

Mereotopology techniques presented in [4] is an ontology-based reasoning engine that aims at identifying joint properties (mated parts, joint type, ...) from complete joint design information in assemblies. The required joint design informations must contain all fastener attributes, their functionnal surface (screw head, cap, ...) to be complete. It is not mentioned how these data are obtained from a digital mock-up since they are not generated during the digital mock-up generation. Adding these informations interactively would be tedious and not applicable to product development processes in industry. Further work based on real data available in digital mock-ups and additional reasoning rules could be an approach to process assemblies where joint design information is incomplete as currently a possible orientation.

Transforming an assembly model as defined in a digital mock-up, i.e. a set B-Rep volumes, into a cell model or, more generally, a non-manifold model is current practice to meet the requirements for structural behavior simulations or Computer Aided Engineering (CAE) needs [5], [6]. Similarly, assembly model transformation needs have been identified to configure them for immersive simulation operations with haptics [7], [8]. Collision detections in pin-hole assembly configurations are among the cases where facetted models create undesired haptic behavior when pin and hole diameters are too close to each other, thus requiring assembly model transformations to obtain acceptable simulation results.

Here, the paper deals particularly about how raw geometric representations of assemblies can be used to infer automatically functional or technological data of assembly features. It can be regarded as a combination of the identification process of core data combined with a reasoning process to set up functional, technological data attached to components and assemblies, as mentioned previously. The paper is organized as follows:

- Section 2 describes the initial parameters and data present in representations for assembly models, including industrial practices and facts observed for assembly representations,
- Section 3 presents needs for various CAE applications,
- Section 4 presents the conventional interface concept aiming at enriching the geometric representation with high-level informations enabling automatic creation of CAE informations,
- In the last section, the proposed approach is applied to an assembly model: conventional interfaces are set up to identify screws and infer their function from their geometric representation in the assembly.

2 An analysis of assembly representations in companies

2.1 CAD functionalities for assembly representation

CAD modelers are used in industry to specify an assembly from a set of entities:

- The component, representing the geometry of an elementary part of the product,
- The location and orientation of components,
- Attributes associated to each components like their name, color, ...

A *component* can be either:

- An elementary component, considered as a non-decomposable one.
- A subassembly, i.e. a set of components grouped through a criterion. This criterion is defined by the designer or prescribed by company methodologies. It is implicit because it is not available as an attribute of a sub-assembly. As an example, components can be grouped together so that they can be handled simultaneously and turned upside down without separating: this criterion is based on kinematic properties of the component set. In addition, it should be mentioned that implicit criteria may lead to non uniform assembly configurations, i.e. in the same assembly, the criterion used for a subassembly may differ from the one assigned to another subassembly.

The set of components is represented as a tree structure where leaves belong to elementary components, and nodes are subassemblies, and the root is the whole product. In the current CAD software, this tree is unique for a given product in a given product file, even though this product structure may need to be reorganized to meet the requirements of different product development process stages [9]. This highlights the fact that the tree structure needs not be unique and multiple criteria take place for an assembly. PDM systems offer possibilities to derive several tree structures for a given product, which requires specific software development for setting up the data structures and the procedures to store and process these tree structures. Even though tree structures are the only category of data structure available to describe an assembly, it does not mean that users can correctly describe an assembly structure under any type of criterion, e.g. if the decomposition criterion is a kinematic criterion and the assembly is a mechanism, the product structure is a loop, which cannot be described with a tree structure.

The geometric model of components used is of B-Rep type, based on canonical surfaces (plane, cone, torus, cylinder, sphere) and NURBS for free-form surfaces.

The location of each component can be specified by a set of geometric constraints between user selected surfaces of different components lying in the same subassembly, or by a translation and a rotation of each component's local frame in the reference frame of its subassembly or of the assembly. The latest solution is commonly used in aeronautics and in the automotive industry to represent large and complex products. Often, the user selected surfaces are functional surfaces but this is up to the user and may differ from one component to another in the same assembly model. Nevertheless, none of the two possible methods rely exclusively on the areas of contact between two components, while this information represents intrinsically the mechanical linkage between the functional surfaces of two components. It must be noticed also that using geometric constraints can become very difficult to handle when assembly modifications are performed. Effectively, when a component modification takes place several geometric constraints may become invalid. Thus, it can be tedious, time consuming and difficult for a user to set up an updated set of constraints when an assembly has several hundreds of components.

Here, the term *"semantic informations"* designates technological informations, kinetic properties, and functions of components. Indeed, it is used here as a generic designation for non geometric information.

E-BOM (engineering bill-of-materials) reflects the product as designed by engineers, with the list of components and their part number, material, and designation. However, E-BOM data provides few semantic information on components, and its access and format are not standardized. There is no guarantee of consistency in the component designations because it reduces to a user-defined character string. Hence, functionally identical components, though slightly different in shape, can be assigned different designations, which shows how the consistency of an assembly with respect to a criterion uniquely relies on the designer's analysis.

As a synthesis of the above analysis, assembly models currently described in a CAD context suffer from possible inconsistencies regarding the functional description of assemblies. It is therefore difficult to set up algorithms that could efficiently process assemblies for inserting functional, technological, mechanical, ... information enriching these models.

2.2 Annotations for the representation of assembly features

Annotations are largely used in 2D technical drawings, which remain a universal language of engineering: it is a formal and precise way of communicating informations about the function, shape, size, features, and precision of physical components. It should be observed also that the concept of annotation stated above distinguishes, somehow, the geometric data forming a technical drawing from non geometric data forming the annotations.

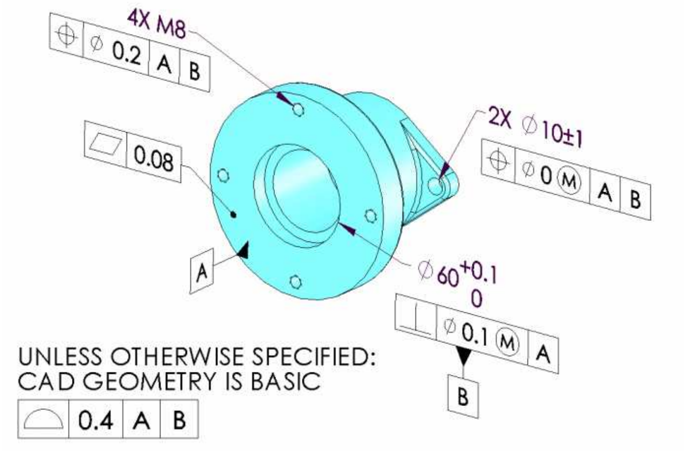


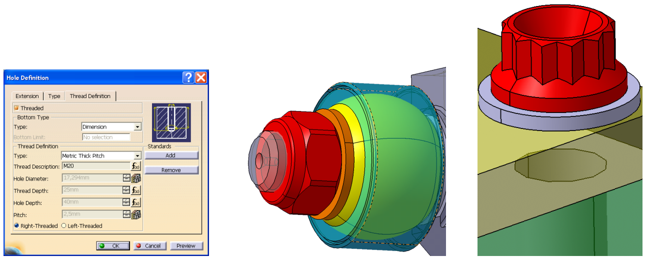
FIGURE 1. Example of CAD model following ASME Y14.41-2003 GD&T annotations and dimensions conventions.

Nowadays, technical drawings tend to be replaced by 3D models. ASME Y14.41-2003 and ISO 16792:2006 standards have been set up to provide a uniform representation of annotations and dimensions and other elements of manufacturing information into 3D CAD models. As shown in figure 1, recent CAD software provide tools for 3D annotations including functional tolerancing, enabling engineers to specify the same informations as in 2D while conforming to the ISO and ASME standards. Here again, we can observe that the concept of annotation refer to information, graphic entities that differ from the geometric model to which they are attached.

2.3 Other 3D representations of assembly features

We observe that other methods exist for the representation of assembly features in 3D CAD models. For example, threads can be represented using the following principles:

- **form features**: recent CAD modelers enable the representation of threads as features in the feature-tree of the CAD part, the thread feature parameters usually include its description (e.g. M20), its depth, its pitch. Such form features are equivalent to 3D semantic annotations (see figure 2(a)),
- **geometric convention**: threads can be represented as an interference between a cylindrical hole and a shaft: the hole representing the small diameter of the internal thread and the shaft the major diameter of the external thread. This interference representation in 3D can be viewed as an extension of technical drawing conventions for threads in a context of



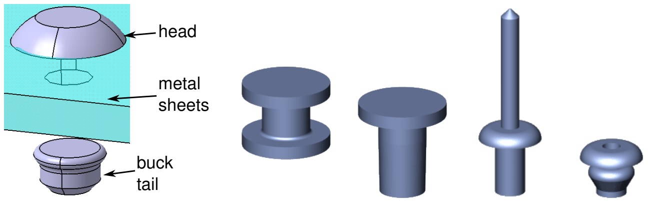
☐ Threaded

Bottom Type
Type: Dimension
Bottom Limit: No selection

Thread Definition
Type: Metric Thick Pitch
Thread Description: M20
Hole Diameter: 17,294mm
Thread Depth: 25mm
Hole Depth: 40mm
Pitch: 2,5mm
● Right-Threaded ○ Left-Threaded

(a) (b) (c)

FIGURE 2. (a) hole feature parameters (b) geometric interference representing the fitting of the threaded shaft into a nut (c) single head representing the complete head screw, and regular hole representing a threaded hole.



(a) (b) (c)

FIGURE 3. (a) representation of the head and the buck-tail of a rivet found in an EADS DMU model, (b) normal and bucked representation of a DIN 7338 rivet found in traceparts, (c) normal and bucked representation of an ISO 16585 rivet found in traceparts.

3D models. It should be noticed also that 3D models differ compared to technical drawing conventions in the fact that there is nothing equivalent to the difference of segment thickness used in technical drawings (see figure 2(b)),

- **partial geometric representation**: the screw can be partially represented, for example some complex assembly models represent screw heads only, hiding the threaded shaft because it is not considered as critical for the purpose of the digital mock up (see figure 2(c)).

According to our knowledge, there is no clear rule assigned to the use of representation of threads in 3D models.

2.4 Excel sheets representation of components interactions

Simulation engineers often create a partial description of an assembly in a set of excel sheets. As an example, in [10], an assembly description dedicated to structural analysis of complex assemblies consisting of two excel sheets is presented: the first one being the list of components with attributes (material, dimensions, name), and the second one being the list of interactions between components (rivets, bolts, lug area, strut area). If this information is sufficient for structural simulations, the main issue is how this information can be derived from a 3D digital mock-up of a product.

2.5 The lack of conventional 3D simplified representation

No uniform 3D simplified representation exist to represent major and minor diameter of threads, whereas technical drawing standards specify bold and thin lines representation of these features. In 2D drawings, the following conventional representations are set up:

- the major diameter of *external threads* uses bold lines, the minor diameter is then shown in thin lines. The interface between the thread and its adjacent smooth cylinder is indicated by a bold line,
- the major diameter of *internal threads* is shown with a thin line, and the minor diameter is then shown with bold lines. The interface between the thread and the smooth cylinder of the hole is indicated by a bold line.

As 2D technical drawings conventions specify that bold lines represent physical surfaces of the product, the normalized representation of two fitting thread features results in an interference area between the internal thread and the external one indicating the effective contact area between threads. This way, the sole 2D geometry contains major informations about the thread. This should be the case for 3D representations to enable an explicit representation of components with the threads they can incorporate.

2.6 Industrial Practices for assembly representations in 3D

In the framework of a product development process, we observed that only geometric conventions can be exploited efficiently to infer semantic information of components based on the following facts:

- annotations of 3D models can be lost when exchanging a model, either a component or an assembly, between CAD and CAE applications. Indeed, current CAD software do not support feature-tree and 3D annotation data exchange in STEP neutral format,
- CAE applications require semantic informations at early stages of design, when 3D annotations have not yet been created,
- excel sheet description of components is a tedious and time consuming task that is performed only when engineers require an input data file for CAE applications. In addition, this practice characterizes the lack of integration between digital mock-up originated from CAD and CAE.

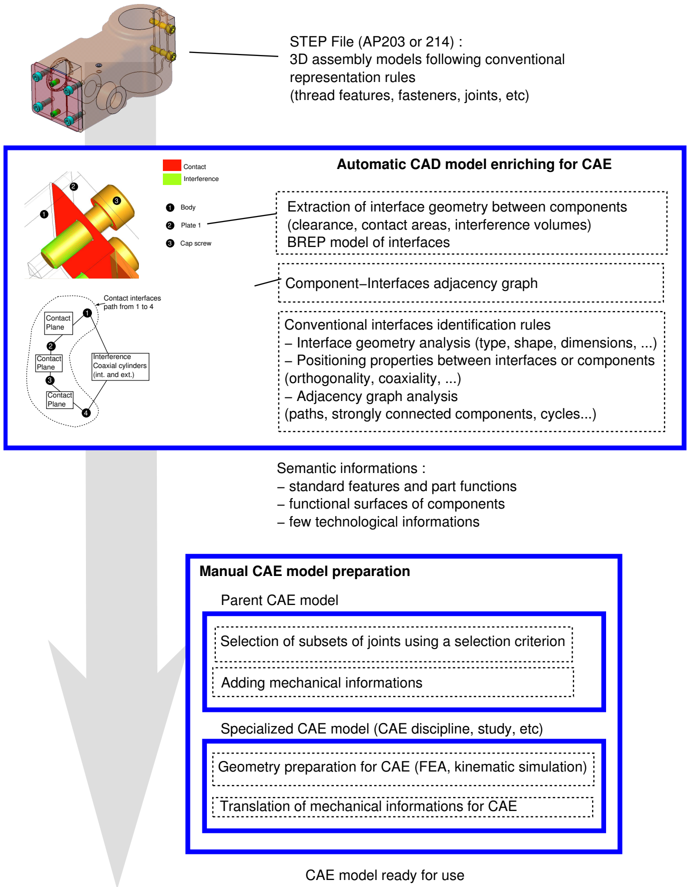


FIGURE 4. Data flow and task diagram of the proposed CAE model preparation approach

On the contrary, geometric conventions are usually followed by designers to represent the semantic of components in assemblies, while the data exchange of geometric representations is currently well supported in CAD and CAE applications through the STEP AP214 and AP203 formats.

The geometric conventions are usually specific to a company or even an engineering department. Our approach to infer semantic informations from geometric conventions is based on two principles:

- designers accept to follow geometric representation rules specific to a standard (ASME, ISO) and to their company,
- a basic set of identification criteria can be adapted to a specific context.

If representation rules help bringing consistency in assembly models, they are still not strongly able to bring a real consistency.

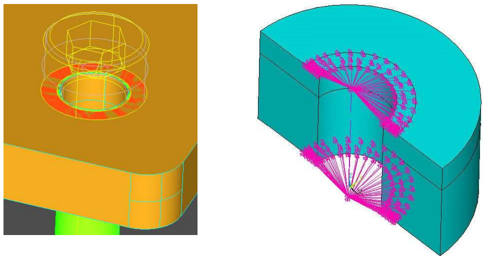


FIGURE 5. Bolt spider model based on a CAD model with an explicit representation of the head contact. [11]

3 Information requirements for CAE applications

3.1 FEA (Finite Element Analysis) of products

The structural simulation of mechanical assemblies is a complex problem because the generation of their numerical simulation based on 3D CAD models takes too much time to fit into current product development processes as soon as the number of joint – whatever joint technology is addressed (bolted joints, rivets, bonded, or welded joints, ...) – reaches a couple of tens of joints.

Recent CAD models preparation for FEA studies have concentrated on the context of standalone components rather than connected sets of components.

Most joint connections simulation models require an explicit representation of the mating part contact area (see figure 5). Indeed, the finite elements are refined and respect the geometry of contact areas, and special elements can be connected to these nodes to join the finite element model of the thread.

To prepare joints FEM models, CAD software rely intensively on interactivity with the user to cut surfaces of components in order to define common areas between them and then to connect the trimmed surfaces obtained, which is largely time consuming.

As a result, FEA preparation of assembly models requires mainly the following informations:

- the semantic informations of joints: e.g. screw, nut, rivets, informations associated to components geometric models,
- the explicit geometric representation of the contact area between joints and mating components,
- material, screw pretention, and other mechanical informations that should be given by the user,

Figure 4 describes the CAD model semantic enrichment and geometric transformations to provide both a "generic" model semantically enriched and specialized ones prepared for each CAE study needs.

3.2 Assembly and Disassembly simulation, Kinematic models

Léon in [2] shown that the function and fitting information of components can be used to reduce the combinatorial complexity of disassembly simulation. By creating automatically associations of components, invalid sequences can be rejected as early as possible in the sequence generation algorithm.

Kinematic models also require the relative mobility between components, which can be inferred from:

- interfaces between components, e.g. a cylindrical fit mobility can be inferred from a coaxial contact between two cylindrical areas of components,
- function of components, e.g. anchorage mobility between a stack of components connected by a bolt,

4 Enriching assembly CAD models with technological and mechanical informations to ease CAE

Section 2.1 underlined that CAD modelers enable only the geometric description of components and products, a tree structure, and a set of attributes. As a result, CAD modelers do not provide high-level assembly informations that are required for CAE domains listed below :

- assembly-disassembly simulation,
- structural analysis of products,
- kinematic simulation,
- search criteria matching components function and features of assembly models.

The above statement derives from the fact that the model consistency cannot be enforced with the current software tools.

4.1 Concept of 'interface' between components

An 'interface' between two components is the set of intrinsic properties of two interacting components including:

- the **interaction type** between two geometric representations of localized components, i.e. clearance, interference, or contact,
- the **localization of the interacting surfaces** on each component, described by an exact B-Rep model based on canonical surfaces (planes, cylinders, spheres, cones), and canonical curves (straight lines, circles, ellipses).
- the **semantic informations attached to interacting surfaces**, expressed as text, physical quantities and symbols (conventional designation, GD&T annotations),

4.2 Concept of 'conventional interface'

When the representation used for one specific component interaction is dominant – in a domain, a company, or one engineering department – then this representation is qualified as a 'conventional interface'. In this paper, we assume that there exist a bijection between the set of 'conventional interface' and the set of 'functions', i.e. each conventional interface matches only one function.

Conventional interfaces enable to set up identification criteria of components' function based on interface informations and semantic informations.

4.3 Enrichment of the CAD assembly model

Rules associated to conventional interfaces enable to enrich the assembly model with semantic informations:

- the **designation** of functional surfaces, i.e. buck tail of a rivet, support surface of a screw head,
- the **function** of components, i.e. cap screw, set screw, stop screw, washer, stop ring,
- the **mobility** between components, i.e. anchorage, pin joint, cylindrical fit, prismatic fit, helical fit, support plane, spherical fit, punctual, linear annular fit, rectilinear fit, cylindrical fit unidirectional, helical fit unidirectional, bilateral plane, n arbitrary plane.

4.4 Component-interface adjacency graph

The component-interface adjacency graph represents the topology of assemblies as following:

- each node represents a component,
- each arc represents an elementary interface linking two components,

The component-interface adjacency graph expressing topological links between components has many advantages:

- graph operators simplify the definition of criteria matching assembly features: for instance, a stack of components is represented by a path where each arc is a contact interface. Another example is a subassembly which is represented as a connected set of nodes connected by arcs having no relative mobility.
- graph operators enables assembly topology editing operators: for instance, the adjacency graph of rigid body subassemblies is obtained by contracting arcs having no relative mobility, assuming deformable components have been already processed. Each node of the resulting graph represents a solid subassembly, while each arc represents an interface having at least one degree of freedom. After such transformations, graph nodes are representing kinematic equivalence classes, which is a first step toward the extraction of kinematic models from assembly models.

Here, the adjacency graph can be regarded as an example of model, intrinsic to an assembly, that can be derived from the analysis of CAD assembly models and stands as a first step toward the generation of kinematic simulations derived from assembly models.

4.5 Requirements for a simplified representation of components interactions

A coherent 3D representation of interactions between components is required to infer conventional interfaces and technical informations characterizing functional surfaces of fasteners.

The 3D representation of features should remain simple: an exact representation of thread ridges respecting the flank, the crest, the root would result in complex B-Rep representations involving NURBS surfaces. Moreover, exact representation would not facilitate the automatic identification of threads.

3D representations should represent functional surfaces of features.

This conventional representation should enable a uniform simplified representation of some features like threads, rivets, etc.

- The interaction domain belonging to two components fitted together through their internal and external threads should be represented by an interference,
- Initially, the relative positions of components are known and coincide with an assembled position of a normal operating configuration.

Currently, it is also assumed that the assembly model used as input contains no error of conventional representation, i.e. all the interfaces conform to the conventional representations defined and these representations are used in places where it is consistent with the corresponding function of a component.

5 Examples of conventional interfaces codification

In this section, conventional interfaces codification is used to identify automatically components that match different screw functions. To this end, an example is used to illustrate how the conventional interfaces are defined and related to different functions. The example shows also the impact of the previous concepts on the enrichment of an assembly model and the component functions that can be derived from the analysis of these interfaces.

5.1 Overall presentation of the assembly

The assembly studied is a precision manual press (Courtesy Janesville Tool and Manufacturing Corp.). Subassemblies are defined by connected sets of components having mobility-less linkage. This kinematic decomposition is represented by an interface adjacency graph in figure 6:

- The '*Lever + sprocket wheel*' subassembly is composed of the lever, the sprocket wheel, an axis joining the lever to the wheel, and fixture components,
- The *pressing rack* subassembly is composed of the rack, the button featuring the pressing area, and a cap screw joining them,
- The *displaceable press support* subassembly is composed of a machined body sliding vertically on the column and supporting the sliding rack and the axis of the lever. Other components are mounted on the machined body to enable a prismatic-fit bi-directional mobility of the rack,
- The *column and foot* subassembly is composed of a foot on which the column is fixed with set screws.

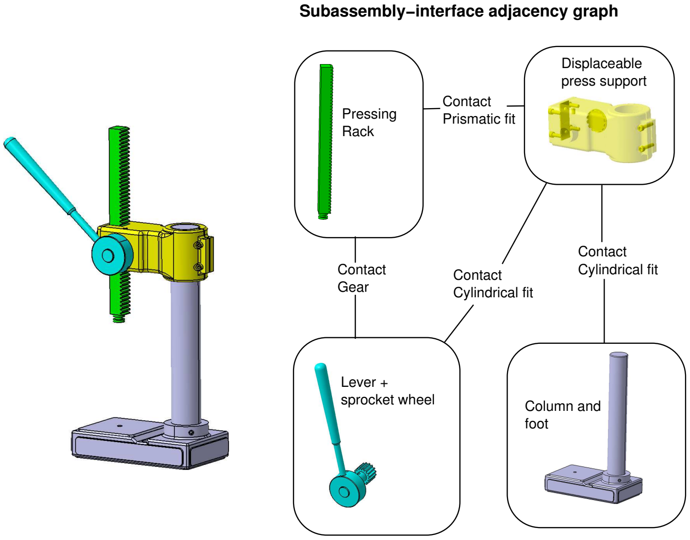


FIGURE 6. Subassemblies composing the precision manual press (Courtesy Janesville Tool and Manufacturing corp.).

These subassemblies are set up using a structural decomposition criterion, commonly used in companies. Here, the transformation of this decomposition in connection with the graph stated at section 4.4 is out of the scope of the present paper. Here, the focus is set on the component level only.

5.2 CAD model interfaces generation phase

A demonstrator based on OpenCascade [12] CAD kernel has been designed to create an explicit CAD representation of interfaces between components. The current demonstrator identifies interfering surfaces, contact surfaces, and creates a CAD model representing each interface, i.e.:

- each interference between two components A and B is represented as a volume, which is obtained by cutting surfaces of each component A or B with the other B or A, respectively (represented by green surfaces in figure 7). This operator is equivalent to a regularized Boolean intersection,

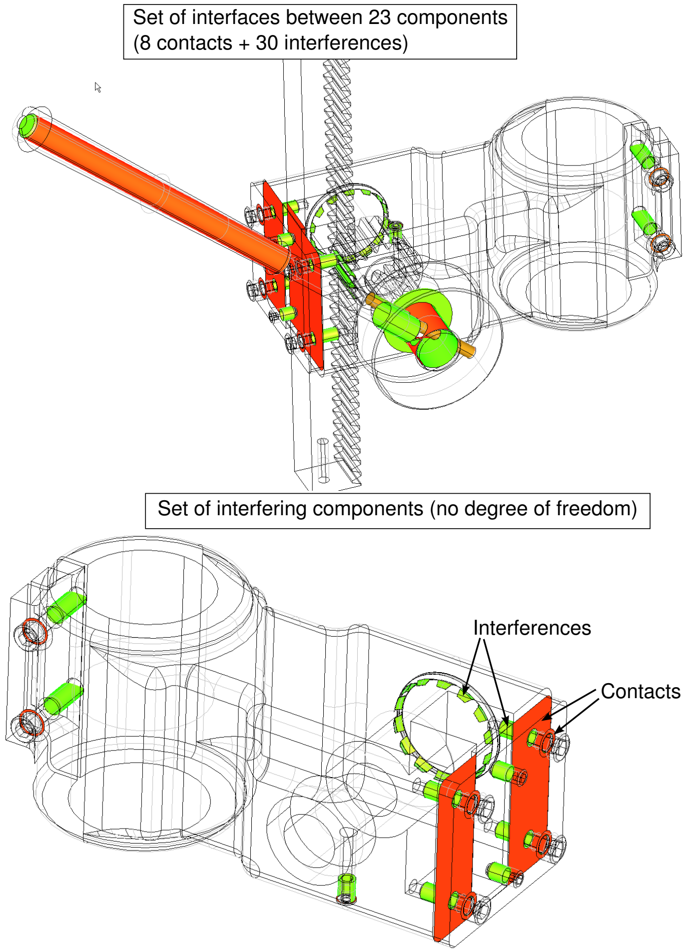


FIGURE 7. CAD model enriched with the explicit representation of interfaces.

- each contact is represented as a shell, which is obtained by cutting surfaces of each component A or B with the edges of the contact area of the other component ∂B or ∂A, respectively (represented by red surfaces in figure 7).

This phase does not create a new topology of each component with respect to the interface boundaries, which is required for FEA models preparation. Subsequent treatments, aim at incorporating the result of the above operators into the boundary decomposition of each component A and B [6]. This is not described here since the focus is the properties of the interfaces and their contribution to component functions. Then, component functions combined with the component shape will be processed to derive new component models suited for FEA simulations. This is part of future work.

5.3 Thread linkage

Figures 6, 7, and 8 illustrate the assembly digital mock-up considered. In this assembly, contacts between one external thread being fit into one internal thread are characterized by the following conventional interface: *an interference where interfering surfaces feature a pair of coaxial cylinders with opposite normals*. This forms a tubular volume whose extremities can be defined by planar or conical surfaces depending on the type of bounding surfaces: planar face participating to a contact area between components or chamfer at the end of a screw thread.

A thread linkage has also the following properties. It must be tubular with a circular section because all threads lie cylindrical surfaces to describe helices. The interference formed in the threaded area of a screw, a nut or a component has a large radial thickness compared to other possible interferences describing component interactions. Indeed, if components are snug fitted into each other and if the corresponding difference of diameter between the shaft and the hole is effective in their 3D model, then the corresponding radial distance is much thinner than the height of a thread. Whether plastic or metallic, a large difference of diameters between snug fitted components is not possible because of the very high stresses that would be generated in the components when assembled.

Assuming other conventional interfaces characterizing assembly configurations: welding, gluing, if cylindrical surfaces occur, they would be positioned with respect to each other either with a clearance or a contact area but there could not be any interference. This conventional representation of interfaces is not refined since its characteristics are far enough from those of a threaded area.

Therefore, assuming an assembly model where its components are consistently located with respect to each other, i.e. there is no assembly error from a technological point of view, the interferences described by threaded areas are the largest ones that can be found for cylindrical areas. Here, the interferences are characterized by the ratio of the interference thickness over the external diameter of this interference. As a consequence, thread linkages are uniquely defined over an assembly model and their identification rely on canonical surfaces that can be found in STEP files. They form a distinctive feature in an assembly model that can be used to characterize functions more easily than analyzing components independently of each other.

The above assertions, if not applicable for any type of product, are strong enough to be applicable to a large range of mechanical products. As stated in the previous sections, the current matter is related to conventional configurations, therefore, there may be products or companies where the proposed convention is not applicable. A new one may be derived: this is however an open question subjected to further analyses.

In the present assembly example, Fig. 7 shows that tubular interferences effectively define the thread linkages between components (screws and lever). The other interferences are not matching the thread linkage definition.

5.4 Cap screws

The function of any component featuring an external thread (x) linked to an internal thread (y) is a cap screw when:

- one component – or a set of adjacent components in contact with each other – is supported against (y) and (x). The normals of the corresponding functional surfaces of (x) and (y) are opposite (see figure 8).

Cap screws are identified by a set of $n \geq 2$ interfaces (see figure 8) using:

- *an* **interference** *typed interface between (x) and (y) components where interfering surfaces feature a pair of coaxial cylinders with opposite normals ((x) featuring the internal oriented cylinder, and (y) the external oriented cylinder),*
- *a* **path** *connecting (x) to (y) where each node of the path is a component, and each arc of the path is a contact interface. Normals of (x) and (y) contact surfaces are opposite.*

The above statement characterizes the function of a cap screw to assemble at least two components. Additional conditions can be stated regarding the contact areas of the components assembled because these areas are most often planar and orthogonal to the axis of the screw. This is a mechanical constraint for a screw to press components against each other and avoid sliding movements orthogonal to the axis of the screw. This justifies the orthogonality constraint to identify a first set of cap screw configurations. Deviation from this reference case is rare and limited to very small angles (around a couple of degrees). In any case, the extreme contact surfaces must be orthogonal to the screw axis since it is a functional requirement when the screw is rotated during the assembly/disassembly operations.

To conform to the conventional representation depicted in Fig. 8 cap screw must also contain a tubular-shaped clearance area that can be useful to enforce their characteristics compared to other assembly configurations.

Within the scope of the assembly example, four configurations only satisfy the above conditions, which effectively characterizes the four cap screws of this mechanism and differentiate them with regard to the other screws in this assembly.

5.5 Set screws

The function of any component featuring an external thread (x) linked to an internal thread (y) is a set screw when the tip of (x) bears on a mating component (w) through an interface that is of contact type (see figure 10).

The geometric properties of the tip interface can be:

- a *closed curve*, e.g. a circle when a cup shaped tip is adjacent to the planar surface of the mating component (see figure below),

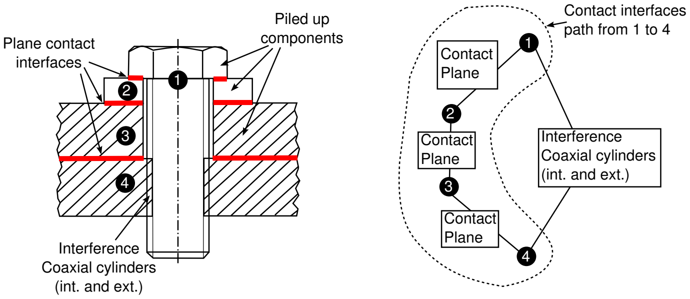


FIGURE 8. Cap screw inferred by an interference interface between internal and external coaxial cylinders and by a contact interface path from the head (1) to the internal thread (4).

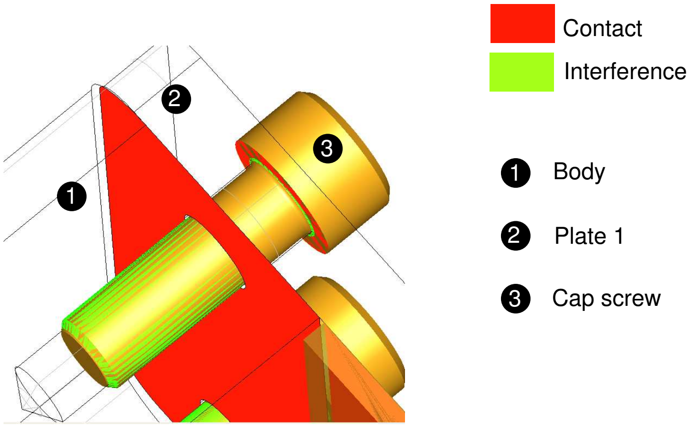


FIGURE 9. Cap screw derived thread conventional interface and superimposed interfaces.

- a *point*, e.g. when a – cone shaped, or dome shaped – tip of the screw set bears the mating component,
- a *surface*, e.g. a disc when a flat tip of the screw set bears the mating component.

In figure 10, the tip interface is a contact circle. The above conditions are clearly independent of the conditions of the cap screw definition. This independence is a strong property to identify efficiently the various configurations and distinguishes the components among each other as well as their functions. In the current example, only one configuration falls into this category, which uniquely identifies and characterizes the set screw.

It has to be noticed that the screw tip touching component (w) follows a conventional representation that can be discussed and modified if conflicting with others or if raising algorithms hardly robust.

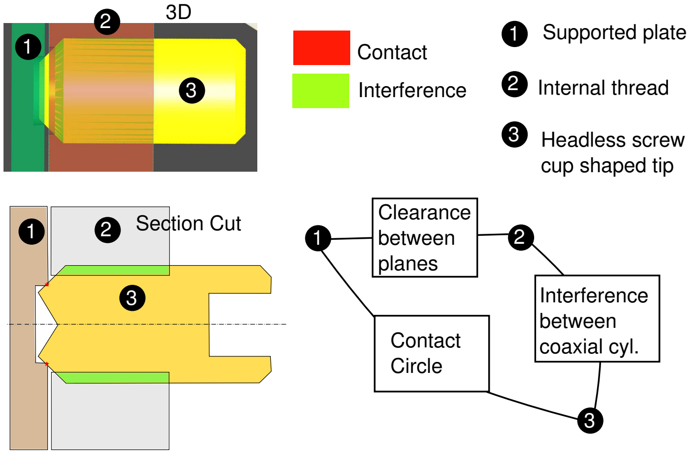


FIGURE 10. Set screw with a cup shaped tip, and headless.

5.6 Clamping cap screws

The displaceable support is linked against the vertical column by adherence between its bore and the column. The adherence is obtained by clamping the bore against the column using two cap screws as showed in figure 11.

The specific clamping function of this cap screw can be identified as follows:

- both internal/external thread interface and contact interface against the screw head belong to the same pair of components (see figure 11),
- the lack of stack of components between the head screw support surface and the thread indicates that the screw function involves a deformation that belongs to a clamping between the head support surface and the thread,
- representing the component in a deformed configuration (obtained by a mechanical behavior analysis) leads to an interference between the column and the displaceable support that indicates the clamping linkage between these two components.

The two first properties are clearly distinct from those of the cap screw and set screw, demonstrating the independence of the three conventional representations and showing that their functions can be derived from their geometetric models. Differences are of topological order, which is robust and clearly separates each configuration even though some interfaces may be hard to identify robustly. The clamping cap screw, because of its principle, must be coupled to a kinematic link, most often of type cylindrical fit or prismatic fit. This can help characterizing the function of the component where the clamping cap screw is located. The last property, though more complex to set up is a means to locate more precisely the area where the clamping effect takes place.

As a summary of the previous specification of interfaces and their combinations to characterize functions, different categories of screws can be identified and clearly distinguished, as confirmed on the example analyzed. In addition, it can be observed that the categories distinguish the lever from the screws because of its interference area combined with contact interfaces. The lever configuration could be further studied to characterize its function. Another strong observation holds in the fact that form features internal to components are not needed. Therefore, it appears to be more powerful to use interfaces to characterize form features of components than processing components independently of each other.

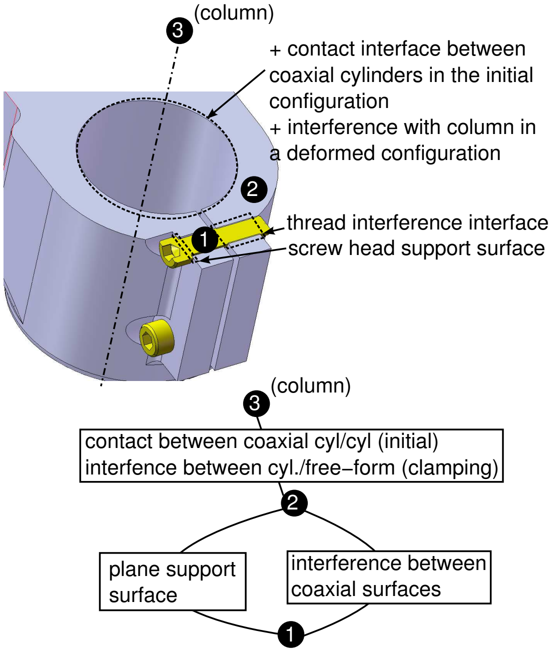


FIGURE 11. Clamping cap screw analysis.

6 Conclusion and perspectives

In this paper, an overview of problems arising when preparing 3D assembly models for CAE was presented. It has been shown that a first step to ease assembly models preparation relies on the enrichment of CAD models:

- identification of conventional simplified 3D representations for standard components and other assembly features, in specific companies,
- geometry preprocessing for the extraction of interfaces between components of assemblies,
- characterization of conventional interfaces based on the analysis of the geometric properties of interfaces and the component-interface adjacency graph.

It has been demonstrated how common conventional interfaces between components enable to uniquely identify thread linkages and hence, screws. Then, combining interfaces into a set of indenpendent properties led into an explicit decomposition of screws into families characterized by their functions. The resulting information can help validating E-BOM designations, improves the consistency of an assembly model and initiates a functional assembly description that can be for subsequent design tasks.

Current limitations of our approach are the following:

- it can be difficult to characterize components functions when their behavior relies on a shape deformation, while these components are represented in their initial shape configuration,
- the geometry preprocessing is subjected to the combinatorial complexity of components in assembly models. Then, processing large assemblies containing tens of thousands of joints must be based on an efficient interface localization process.

Our middle-term perspectives are:

- the extension of the conventional simplified representations and conventional interface characterization to a wide panel of:
    * joints: rivets, bonded, welded joints, etc,
    * bearings: bushings, rolling-elements bearings, plain bearings,
    * standard parts: gears, racks, ball bearings, strings,
- the study of a GPU-based approach to speed up the localization process prior to the identification of conventional interfaces.

As a summary, the conventional representation of interfaces extracted from a geometric model of assemblies proved to be very efficient to derive intrinsic functional properties that can enrich the assembly model and be subsequently processed.

REFERENCES

[1] Shah, J., and Mantyla, M., 1995. *Parametric and feature-based CAD/CAM concepts, techniques, and applications*. http://www.amazon.fr/Parametric-Feature-Based-Cad-Cam-Applications/dp/0471002143.

[2] Leon, J.-C., Rejneri, N., and Debarbouillé, G., Sep 9-12 2001. "Assembly/disassembly simulation early during a design process". In In Proceedings of 2001 ASME DETC-CIE Conference.

[3] Çiçeka, A., and Gülesin, M., 2007. "A part recognition based computer aided assembly system". *Computers in Industry,* ***58***(8-9), pp. 733 – 746.

[4] Kim, K.-Y., Yang, H., and Kim, D.-W. "Mereotopological assembly joint information representation for collaborative product design". *Robotics and Computer-Integrated Manufacturing*.

[5] Armstrong, C., Monaghan, D., Price, M., Ou, H., and Lamont, J., 2002. "Integrating cae concepts with cad geometry". *Engineering Computational Technology*, pp. 75–104. ISBN: 1874672172 Publisher: Civil Comp Limited.

[6] Hamri, O., Léon, J.-C., Giannini, F., Falcidieno, B., Poulat, A., and Fine, L., 2008. "Interfacing product views through a mixed shape representation. part 1: Data structures and operators". *International Journal on Interactive Design and Manufacturing,* ***2***(2), pp. 69–85. Springer Verlag.

[7] Iacob, R., Mitrouchev, P., and Léon, J.-C., 2008. "Contact identification for assembly/disassembly simulation with a haptic device". *The Visual Computer,* ***24***, pp. 973–979.

[8] Howard, B. M., and Vance, J. M., 2007. "Desktop haptic virtual assembly using physically-based modeling". *Virtual Reality,* ***11***(4), pp. 207–215.

[9] Drieux, G., Léon, J.-C., Guillaume, F., and Chevassus, N., 2009. "Processes to integrate design with downstream applications through product shapes adaptation". *IEEE Systems Journal,* ***3***(2), pp. 199–209.

[10] Gonze, A., and Verstuyft, J. "Associative sizing of aeronautical structures from catia v5 to samcef : applications to static and bird impact analyses". In 9th SAMTECH Users Conference.

[11] Montgomery, J., 2002. "Methods for modeling bolts in the bolted joint". In ANSYS User's Conference.

[12] OpenCascade, 2010. Open CASCADE Technology, 3D modeling & numerical simulation. http://www.opencascade.org/.