Collaborative Virtual Assembly of Spatially Dispersed and Heterogeneous 3D CAD Models

Posada, Jorge
VICOMTech, San Sebastian 20009 (Spain), Paseo Mikeletegi 57 jposada@vicomtech.es

Stork, André
Fraunhofer Institute for Computer Graphics (IGD), Darmstadt 64283 (Germany), Rundeturmstrasse 6, stork@igd.fhg.de

Abstract

In a typical Concurrent Engineering scenario, users can discuss currently their 3D CAD models in Collaborative CAD environments to a certain extent, but the valuable functionality of Virtual Assembly of dispersed 3D CAD parts steming from heterogeneous systems is not well addressed yet. Users should be able to examine their models in relation to others and verify that they match properly as a powerful decision-making tool independent of any specific CAD system. We present a novel user-driven approach approach, based on highly intuitive sequential steps, to assembly parts of 3D CAD models using a 2D input device which is completely integrated in an environment supporting geographically dispersed and heterogeneous CAD models. Users with different skills (even non-CAD users) are enabled to assemble 3D CAD models in this scenario. Our approach overcomes the inherent difficulty of assembling 3D CAD models from different systems by using basic attributes of the model instead specific assembly schemas.

1 Introduction

Collaborative CAD is becoming more and more a fundamental tool in the Concurrent Engineering practice. Currently, teams of designers and engineers located in dispersed geographic areas work collaboratively on complex projects involving 3D CAD models, not unfrequently modeled with different CAD systems. The workteams can nowadays use collaborative CAD applications to discuss and work cooperatively on their models to a certain extent. However, most of these applications have limitations handling 3D CAD models from different systems for the purpose of Virtual Assembly. These applications can be roughly classified into three categories:

- Applications supporting several input formats with restricted functionality (mainly visualization and restricted interrogation of the data).
- Applications supporting a wider range of functionality for one specific format.
- DMU Applications providing more advanced functionality such as collision and clearance analyis based on triangulated models.

In the first case the task of Virtual Assembly faces the problem of having different assembly schemas from the individual CAD systems. In the second case all the specifics of a particular system (including possibly assembling knowledge) can be only exploited at the expense of limited compatibility with other 3D CAD formats. In the third case, only the triangulated geometry is known and key model information (such as radii or axes for holes) is lost, limiting considerably a potential assembly procedure. Due to these reasons, Virtual Assembly of dispersed 3D CAD parts coming from heterogeneous systems is not available in most Collaborative CAD applications.

However, Virtual Assembly of such 3D CAD Models can offer clear advantages in the Collaborative CAD process. The users can verify if the different parts of a complex model do match as planned. They can discuss and compare their models in their final context and not as isolated parts. They can use it as a valuable tool helping in the iterative decision-making process of collaborative CAD.

Considering this, we developed a novel user-driven approach for assembling parts of 3D CAD models using a 2D input device. We specifically address the problem of assembling CAD parts coming from (possibly) different CAD systems in a virtual environment for the purpose of Design Review[1]. Therefore, no assumptions about a

---

[1] Optimal assembly paths or manufacturing processes are not addressed in this work since the focus is not the simulation of the physical assembly operation, but mainly the Design Review of the validity of assembly configurations

particular way of modeling for the parts (parametric, feature-based, etc) are made. In contrast to common approaches in individual CAD systems, the user doesn't need to abstract the assembly problem by means of auxiliary coordinate systems, analytical systems of constraints, or the use of predefined assembly features. He doesn't depend on still uncommon 3D input devices either. With the presented technique, users can enhance their collaborative work. The procedure can be summarized as follows. Successive relationships between parts based on topological attributes meaningful to the user (faces, edges/axes and vertices) can be defined. One part will be transformed accordingly to coincide with its counterpart. A corresponding reduction of the degrees of freedom (DOFs) for the parts is automatically introduced. In case of violating the existing restrictions a feedback message with the exact cause is displayed. In every intermediate step the parts are in a consistent state, represented by visual feedbacks showing their remaining DOFs. Both hints guide the user through the assembling process. An interactive and discretized positioning and orientation of the parts based on current DOFs is then allowed. It is possible to release constraints to gain back a DOF without undoing parts transformations

Our research has been recently integrated (patent pending) in a commercial environment for collaborative work on 3D CAD data from different sources as well as geographically dispersed models (CoCreate's OneSpace®). As a Computer-Supported Cooperative Work (CSCW) application for Concurrent Engineering in CAD, OneSpace® offers a collaborative platform to implement the research approach described in this paper. Shortly described, it is a "virtual conference room" (where among several important supporting functions) CAD models from the most important commercial systems can be discussed over the Internet. As the models are handled in a neutral-CAD format, the specifics about the assembling process in each CAD system are not available. Our approach has proven to be easily integrable and is now fully functional in the last version of OneSpace®.

2 Related Work

Initially the problem of assembly of CAD parts was focused mainly on the underlying modeling and solution of certain assembly constraints. Extensions to feature-based design, as well as reasoning based on geometric constraints, have been studied for assembly purposes.

The feature-based approaches work on an Assembly Modelling concept, where semantic "Assembly Features" are introduced during the part modelling and used afterwards for assembly (see [8][7][11]). The user must explicitly set the assembly features in the modelling process; if he doesn't, the possibilities of the approach remain unused. For parts coming from different CAD systems is not a practical solution.

The approaches based on geometric constraints include among others the work of Kramer/Anantha [5][1], Fa/Fernando [2] and Veltkamp [10]. Kramer studies the solution of geometric constraint systems for assembly and prefers a DOF analysis instead of instable numerical iterations. He does not address intuitive interaction and visualization techniques, nor the release at will of a given constraint. Fa's "allowable motion" detects possible matching topological entities for a part while it is moved and restricts correspondingly its possible movements on the fly. However, the user can not introduce the constraints at will with a clear assembly goal, but depends on the automatic assignment of the system. No groups can be simultaneously assigned either. The user doesn't have an intuitive perception of remaining DOFs for a part. Other common constraint-based approaches require a complete specification of the constraints before evaluating an assembly possibility, therefore demanding a high level of abstraction and are not very user-friendly.

More recently, the focus has moved from the analytical underlying models to the user-computer interaction needed to accomplish the assembly. Virtual Reality techniques have shown to be very useful in this area [6][4][3]. Rosen [6] already integrates some visualization and interaction aspects for assembly and disassembly. Jayaram [4] defines Virtual Assembly (VA) as the use of computer tools to assist assembly-engineering decisions without the physical realization and presents the prototype system VADE for specific assembly scenarios. Advanced I/O devices are introduced to the assembly context, giving interesting possibilities for the future, when those devices are widespread enough. VADE can not be used to assemble parts from different modelers since it relies on specific modeling information from the CAD system.

Sun [9] independently developed an interactive approach for Virtual Assembly that seems to be in some sense close to the spirit of our work, unfortunately not described in depth. He follows a merging approach based on attribute graphs and simplifies a complex set of constraints in terms of simple rotational/translational operations in a sequential way. He also provides interactivity of parts in the virtual environment and doesn't require complete predefinition of constraints for evaluation. However his work is based on 3D input devices and doesn't provide feedback mechanisms for user guidance in the process.

It can be said that the main lack of existing approaches for assembly of CAD parts in virtual environments is that they are not applicable for wide industrial use. Reasons include the use of input/output devices still uncommon in the standard workplace (HMD, 3D Mouse, Virtual Table, trackers, etc.), assumptions on specific ways to model the independent parts, need of complete boundary definition before analysis or difficulties in the integration with normal GUI's for CAD systems.

3 Process Overview

As a response to some of the problems described, our approach allows the user the intuitive and interactive Visual Assembly of CAD parts from different models by means of a 2D input device. Only 2 mouse clicks are needed to introduce a constraint. The constraint effects are visualized immediately. Visual feedbacks represent at all times the remaining DOFs. Errors and constraint violations are reported with their cause. Complete assembly configuration is typically reached after only 3 relationships (see Figure 1).

The procedure presents among others the following advantages:

- It is applicable for current workplace configurations.
- No assumption on a specific part modeling technique is made. Therefore CAD models from different sources are supported. The method just relies on vertices, i.e. their positions, edge/axis information, i.e. origin and direction, and face/plane definitions, i.e. origin and normal. The current implementation is limited to straight edges/axes and planar faces. Although this might appear as a major limitation, one has to note that most assembly situations in mechanical CAD can be represented by these entites and relations between them. Moreover, this fundamental information can be extracted from any CAD model.
- The user doesn't need to be familiar with CAD modeling itself. Intuitive assembly positioning is allowed instead of abstract and complicated assembly schemes.
- The assembly constraints introduced are clearly goal-oriented, in contrast with "random" constraints [2].
- The user is guided through the assembly process.
- 2D Interactive and discretized 3D manipulation of the parts is possible at all times, taking into account current DOFs.
- It works with the exact mathematical representation of the parts, not with their tessellated approximation.
- The sequential approach followed is highly intuitive.
- It's suitable for collaborative assembly sessions.

The process description is given below.

3.1 Process Description

3.1.1 Principle

The basic principle behind our approach is the incremental constraining of parts by means of simple relationships between them. The relationships associate topological attributes (faces, edges/axes and vertices) from one part to another. There is a finite set of possible relations, manageable with a case based approach instead of an analytical geometric constraint solution. The cases used are Vertex-Vertex (VV), Edge-Axis / Edge-Edge / Axis-Edge (EE), and Face-Face (FF). Notice that we manage edges and axes indistinguishably.

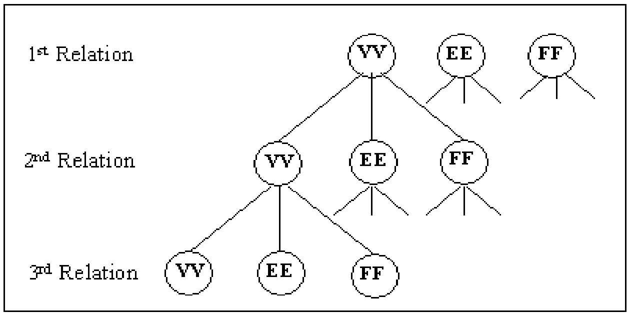


Figure 1. Constraint relationships (partial view)

In Figure 1 a partial view of all possible cases is shown.

At one stage, the user introduces a relationship simply by clicking on the respective topological attributes of two different parts. If the relationship is valid, the following immediate effects occur on the parts: positioning of the first selected part (hence called "moved part"), update of visual feedbacks and DOF reduction of both. The role of "moved part" can change from relation to relation. After a maximum 3 relations applied to a part it reaches a full constrained status (zero DOFs).

Originally the parts have 6 DOF: 3 for translation and 3 for rotation. Only a small set of possible states can be reached by introducing the relations described above (see Table 1)

Table 1. Possible resulting combinations of DOF for a part after a relationship (VV,EE,FF) is made

DOF Rotation	3	1	1	1	0	0
DOF ranslation	0	2	1	0	1	0
e.g. after	vv	ff	ee	Vv,ee	ff,ee	ee,ee

For each combination of DOFs a small subset of possible relationships (FF, EE, VV) can be introduced. Depending on the geometric configurations of the moved and fixed parts, the reduction of DOFs is handled differently for each DOF combination. In this way, about 20 initial situations from the 7 valid DOF combinations are defined (the extra one is 3,3).

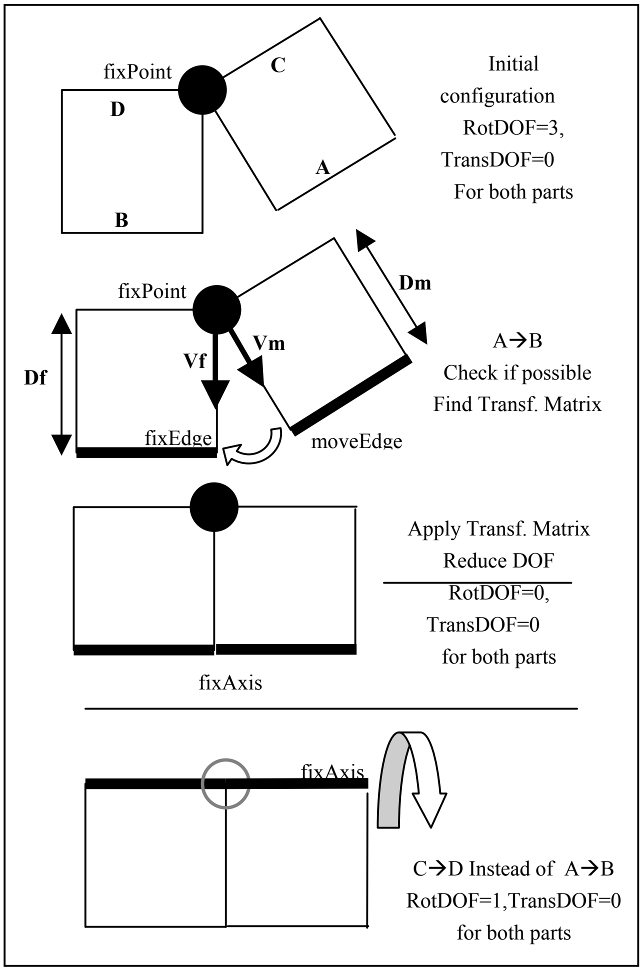


Figure 2. Introducing a Relationship

When the user tries to introduce a relationship, the procedure first looks for some geometric transformation of the "moved part" that can fulfill the desired relationship without violating the existing restrictions of DOFs. The remaining DOFs are adjusted accordingly.

Finally, a visual feedback will be displayed on both parts representing the new status of DOFs.

At this point it is possible to interact with the parts (rotate/translate respecting current DOFs) or introduce a new relation.

3.1.2 Example of relationship procedure

In the Figure 2 a schematic example of the process of introducing a relationship is given. The initial configuration depicts two 3D boxes (depth not shown) sharing a common point from a previous relationship. Both parts still have all 3 rotational DOFs but no translational DOFs remain.

The user picks with the 2D input device first the edge A and then the edge B. The edge A is therefore the "moving edge". The picking actions are interpreted as the user wanting a relation EE between the parts. The system detects that by means of a rotation of the "moved part" the desired EE relation can be exactly fulfilled, since the distance between fixPoint and fixEdge (Df) is equal to the distance between fixPoint and moveEdge (Dm). Although not relevant for this example, it is important to mention that we use heuristics to predict the desired assembly configuration: if *coincidence* (it is, exact matching) can not be achieved, *parallelism* would be tried, as explained later.

A calculation of the appropriate transformation matrix is then made. In this case it is a pure rotation matrix, defined by:

Rotation Axis = crossProduct (Vm, Vf)
Rotation Angle = angle (Vm, Vf)
Rotation Point = fixPoint
With
Vm = moveEdge.nearest (fixPoint) – fixPoint , and
Vf = fixEdge.nearest (fixPoint) – fixPoint.

The matrix transformation is then applied to "movePart" and the DOFs reduced accordingly. In the example, both parts finish with no available DOFs.

Now, let us assume that Df is different from Dm. Then, the two selected entities cannot be made coincident. Instead, our procedure checks if it is possible to make them parallel. If so, a parallel constraint is established automatically, if not, the assignment is rejected telling the user exactly why it was not possible to match the features. In the example it would be possible to make A and B *parallel* if Df and Dm were different, since the moved part has all rotation degrees of freedom. Similar effects are achieved in other situations for faces/planes with the automatic *parallelism* assignment from heuristics.

Let´s return to the the example in Figure 2. Had the user clicked the edges C and D (instead of A and B) another result would have been obtained, as shown in the last part of figure 2. A rotation DOF would allow in that case to rotate any of the parts around fixAxis.

Sometimes the user wants to establish a *parallel* constraint although *coincidence* can be achieved. Since our heuristic always tries to establish a coincident constraint – if possible -, the user has the possibility to input a distance value afterwards to change a *coincident* into a *parallel* constraint. If the user wants to change the distance between to parallel entities he can always come back and enter a new value.

3.1.3 Preselection

The described procedure can be applied not only to individual parts, but also to groups of parts. A set of parts, no matter if they are geometrically connected or not, can be preselected before the relations are introduced, and be handled afterwards as a single unit ("move" or "fixed") during the process.

3.1.4 Visual Feedbacks

At any stage visual feedbacks represent very intuitively the available DOFs for the parts.

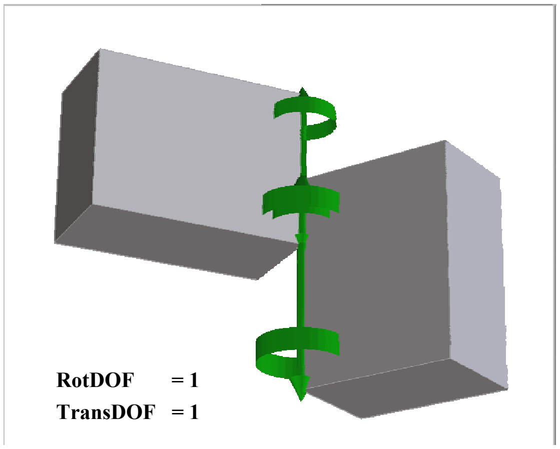


Figure 3.1 Visual Feedback : Both parts can rotate and translate about the common axis.

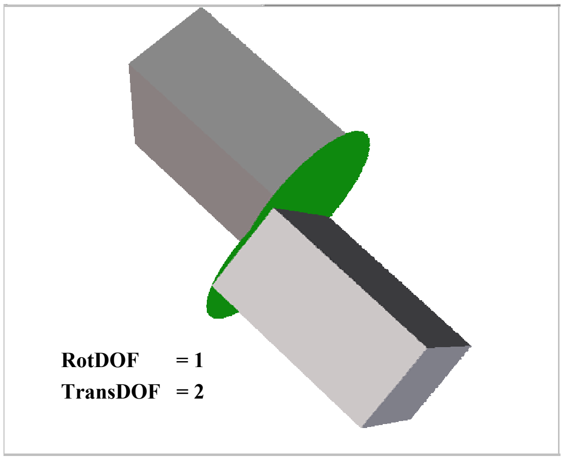


Figure 3.2 Visual Feedback: Parts can rotate around common face normal and translate in the common plane.

The feedbacks are updated automatically after a new relationship is introduced. They are attached to the parts and moved with them during interaction. Figures 3.1 and 3.2 shows two feedbacks as example.

3.1.5 Message Feedbacks

The user is constantly assisted through messages during the assembly process. For example, in case of violation of existing restrictions a message with the exact cause is given (In Figure 2, if the user tries A→D the message returned would be "edge distances to fixPoint are different").

3.1.6 Discretized Interaction

The user can interact with the parts at any time. He only needs to drag on a part with the mouse to get an intuitive displacement that takes into account the current DOFs. The button pressed determines if translation or rotation is desired. The movements are discretized in units configurable by the user to achieve precise assembly configurations. The user sees the exact units of translation/rotation movement at any time.

Different mechanisms were used in order to give the user an intuitive manipulation of the parts in the 3D environment with a 2D input. The most important mechanism is the intelligent use of *projectors.* Projectors are based on the intersection of the 3D ray of the projected 2D position of the mouse on 3D auxiliary geometries (invisible to the user). Those geometries are automatically generated and include cylinders, spheres and planes related with the visual feedbacks introduced in 3.1.4.

Switching between different projectors during interaction occurs automatically based on mouse movements and camera changes for optimum response to user expectations.

3.2 Reverse / Release Constraints / Offset

Some optional mechanisms enhance the basic procedure. The *reverse* option inverts the orientation of a part about an axis/edge or a normal after an assembly step, in the few cases where the heuristics used to predict the right orientation don't give the desired result. As heuristic we use minimum transformation, i.e. if we have to rotate an object to achieve coincidence between entites, we prefer the rotation with the minimum angle.

With the *release constraints* option the user can gain back DOFs from the last assembly step without undoing parts transformations. This is useful in several contexts, such as providing a reasonable initial position for the discrete movements that can be done interactively with the parts after an assembly step.

In the standard way to assign faces/edges to each other the *coincidence* between them is obtained. In some cases *parallelism* is desired instead. The *offset* option allows to assign an offset distance to the last assembly operation, provided that the direction for the offset is not ambiguous in the current configuration.

With the help of these additional options complex assembly operations can be achieved.

3.3 Assembly Example

An example of matching CAD components of a CD ROM unit assembly model is shown in figure 4 (see next page).

The parts come from different CAD sytems. In the first step, a user in the Collaborative CAD session clicks on the bolt axis and then on the hole axis. Automatically the first part is oriented to make both axis colinear. An intuitive feedback indicating one translational DOF along the common axis and one rotational DOF about the same axis is immediately shown. At this point, the part can be interactively moved along or around the axis in discrete steps.

In the next step, the user clicks on the axes of the second bolt and the second hole. Both axes are hence aligned, taking into account the available DOF after the first step. Now both bolts are aligned with their respective holes. The remaining translational DOF is shown with a corresponding visual feedback. If the axes could not be matched (e.g. if the distance between the holes differs from the distance between the bolts), a feedback message indicating the reason for the failed assembly step would have been be displayed.

In the third and last step the user clicks on the two planar faces that will match to finish the assembly. The part is then completely constrained and the assembly is finished. Notice that only six intuitive mouse clicks are needed to accomplish the assembly.

4 Conclusions and Future Work

The Virtual Assembly of spatially dispersed and heterogeneous 3D CAD Models is a very useful tool for Collaborative CAD in the Concurrent Engineering practice. Most current applications do not address this topic well. We present a functional, general approach for interactive virtual assembly based on 2D input that overcomes many of such limitations and can handle such dispersed and heterogeneous 3D CAD models, based on sequential model-dependent constraints and the corresponding reduction of DOFs. Future extensions can support for a wider set of matching topologies and further functionality such as collision detection during interaction.

5 Acknowledgements

We want to thank CoCreate Software Inc. for their financial support for this research and sharing their valuable experience in CAD user needs with us. The work described in this paper is already fully functional in the product OneSpace from CoCreate.

References

[1] Anantha, R., Kramer, G.A., Crawford, R.H.: “Assembly modelling by geometric constraint satisfaction, Computer-Aided Design, Vol. 28, No. 9, 1996.

[2] Fa, M., Fernando, T.: Interactive Constraint-based Solid Modelling using Allowable Motion. Proceedings of the 2nd ACM Solid Modelling ’93, Montreal, Canada, May, 1993.

[3] Gupta, R. et al. Prototyping and Design for Assembly analysis using Multimodal virtual environments. Computer-Aided Design. Vol.29 No. 8 pp. 585-597, 1997.

[4] Jayaram, S. et al. Virtual Assembly using Virtual Reality Techniques. Computer-Aided Design. Vol. 29, No. 8, pp. 575-584. 1997

[5] Kramer, G. A Geometric Constraint Engine. In Constraint Based Reasoning, MIT Press, pp. 327-360. 1994.

[6] Rosen, D. et al. Virtual Prototyping for Product Demanufacture and Service using a Virtual Design Studio Approach. ASME Computers in Engineering Conference, pp.951-958, 1995.

[7] Shah, J.J. and Rogers, M.T.: Assembly Modeling as an Extension of Feature-Based Design, Res. Engineering Design, Vol. 5, pp. 218-237, 1993.

[8] Shah, J.J. and Tadepalli, R.: Feature-based Assembly Modeling. Comput. Engineering. Vol. 1, 1992.

[9] Sun, Hanqiu et al. Interactive Task Planning in Virtual Assembly. ACM VRST99 Proceedings, 1999.

[10] Veltkamp, R. A Quantum Approach to Geometric Constraint Satisfaction. In Object-Oriented Programming for Graphics (Laffra et al), Springer Verlag. pp. 54-70, 1995.

[11] Whitney, D. E.: The Potential for Assembly Modeling in Product Development and Manufacturing. MIT 1996.

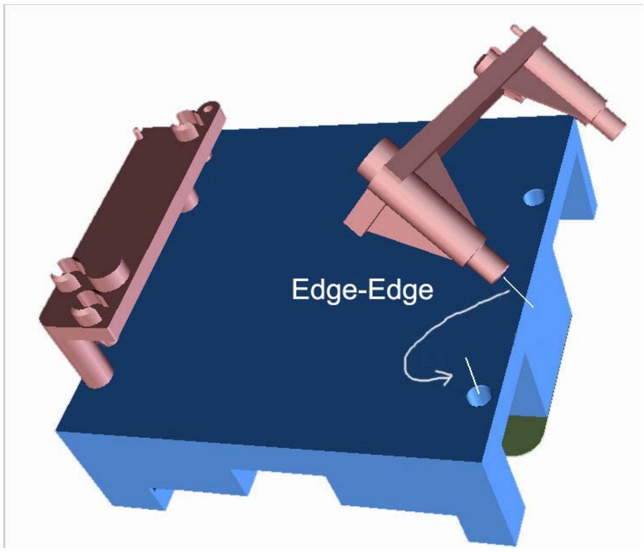


Figure 4.1 . Assembly Example: First Assignment (EE)

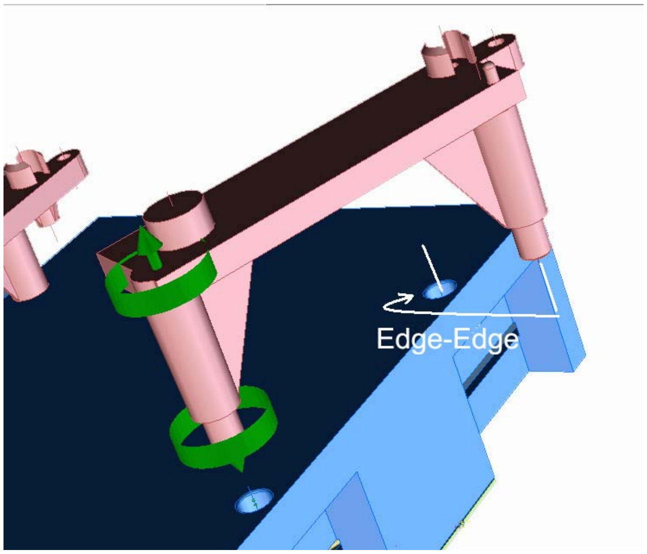
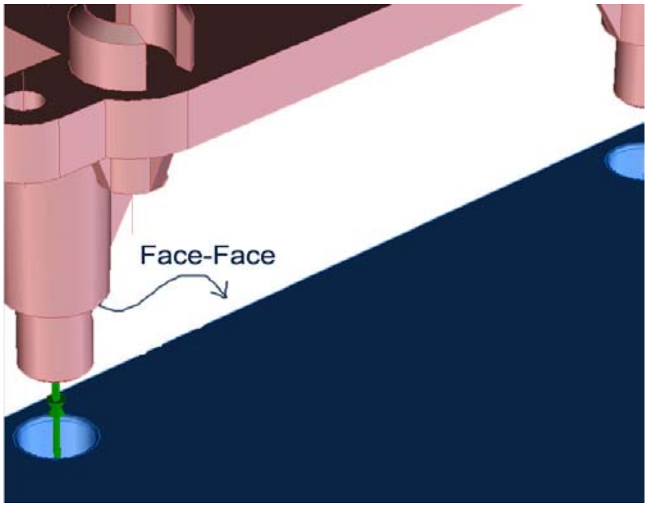


Figure 4.2 . Assembly Example: Second Assignment (EE)



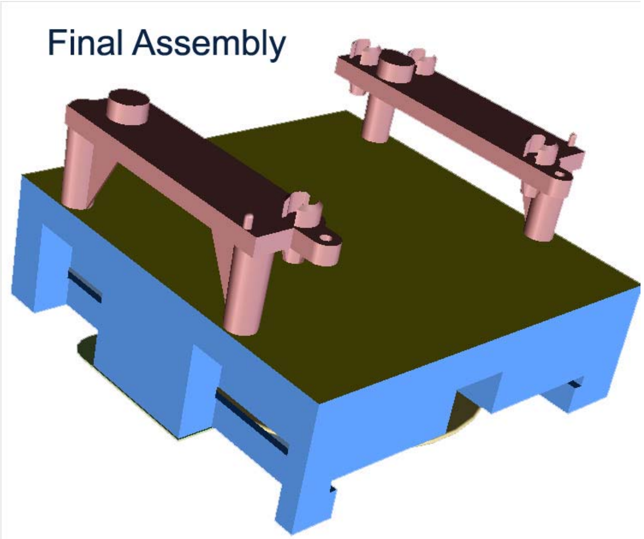


Figure 4.4 . Assembly Example: Assembly Finished