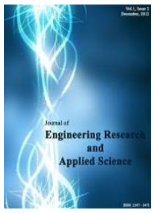
Journal of Engineering Research and Applied Science
Available at www.journaleras.com
Volume 7 (1), June 2018, pp 765-777
ISSN 2147-3471 © 2018



An implementation of proactive assembly design process to compare its efficiency

O.Kayabasi[1,a]

[1]Duzce University, Biyomedical Engineering, Duzce, Turkey.

Accepted 14 March 2018

Abstract

The study is related to the development of CAD software designed for assembly-oriented environment. Although component based approach was used for many CAD software, assembly oriented design was proven that it was much more convenient with respect to errors occurred during product development process. Because assembly oriented design introduces proactive design for assembly in addition to the assembly sequence construction scheme. Although there are many studies regarding automated feature recognition and geometric information extraction, information is given by designer manually in this study. This study deals with the effects of assembly-oriented design on the design efficiency. The computer system defined by Swift and Jared are repeated in this work to compare the results with the component based design method.

Keywords: experts systems; design for assembly (DFA).

1. Introduction

Design for X (DFX) is an emerging philosophy that design decisions of products and processes are improved simultaneously by examining their interrelationships. It is an umbrella term for a suite of effective design techniques in product development. Examples include design for manufacturability, design for assemblability, design for inspectibility, design for reliability, design for serviceability, design for recyclability, etc. Among them, design for assembly (DFA) has been most widely applied in industries with most impressive achievements.

Since the prevalence of two well-known DFA tools Boothroyd–Dewhurst DFA[1] and Lucas DFA [2] in industries, significant developments have been attempted in several directions. Syan and Menon [3] and Huang [4] provide more comprehensive discussions on DFX development. One is to exploit recent development in computing such as Internet and multimedia to facilitate the provision and utilization of DFX techniques.

Design for Assembly (DFA) is an important manufacturing tool that can substantially reduce the costs attributable to assembly. Besides cost reduction, DFA generates additional benefits such as higher quality, increased reliability, and shorter manufacturing time. A major effort was made to develop DFA methodologies during the eighties. Since their emergence, two different approaches have been investigated. The first method focuses on the evaluation of each elementary part of a product whereas the second considers the product as a whole [5]. Design for assembly starts at the early stage of product development and plays very important role for product preparation time, product quality and therefore the cost of product. Although the total product cost can be controlled with efficient design for assembly current design systems do not considered this fact. They concentrate on the product components separately. Therefore conventional software must be improved to consider the assembly problems and feature-based assembly in order to increase manufacturability of components [6].

Design for assembly techniques were used in almost every design process for many industries. These techniques are able to reduce the unnecessary parts reduce time requirements and therefore reduce cost in a certain extent [7]. However there are some disadvantages of the component-based design for assembly.

These techniques are time consuming and are applied at the end of the design. Therefore any improvement on any component needs a feedback to the early stages of design. Since the assembly stage is considered after the components are designed, it is not possible to see overall assembly requirements [8]. Refinement of design for assembly will help to the

[a] Corresponding author;
Phone: +90-380-542-1036, Email: oguzkayabasi@duzce.edu.tr

designer to design the component in terms of easy manufacturability and easy assembly.

With the help of new design and information systems proactive design for assembly method was introduced [9]. Method was not a new idea rather than was an understanding of assembly oriented design. First of all assembly sequences were studied [10]. Suitable assembly sequences were evaluated to use in DFA [11]. The automation of design for assembly evaluation process needed frameworks to work in a link with CAD systems [12]. In order to gain the benefits of concurrent design of components assembly design, DFA and redesign must be combined [13].This effort needed one more thing which was methodology to perform these actions during designing stage.

The assembly relations are already defined in CAD systems as schematic representation [14]. Graphic representation of hierarchies between parts is important in this study also .Otherwise the information transfer between steps is not easily understood for designer.

In this study the effect of this proactive assembly oriented design method was studied. Therefore an experiment was conducted. The assembly of a RoboClean (vacuum cleaner) was created by component-based approach and by proactive DFA method. In order to perform these simulations, all necessary information is provided manually. The relevant information is geometric data, orientation data and constraint data. Corresponding tables are created and they are used in DFA process. Details of the experiment (RoboClean) are given in section 4.1.

The time consumed for the same assembly both for assembly oriented and component oriented methods were obtained. The errors and their costs were estimated. In order to obtain these results a code was developed. This code was similar to the work done by Swift and Jared [15]. The other works has shown the use of the methodology for generating assembly sequences in parallel with the design definition. This methodology has been shown to successfully overcome many of the issues with current sequence generation approaches; it might become more effective with the addition of further validation and evolution criteria and use of functional representation to enable methodology to be applied earlier in the design process [16-19].

1.1. Geometric requirements for DFA

Basic DFA requirements are always used for manual assembly analyses. However, feature recognition processes are only needed for automated assembly operations. Table 1 lists geometric requirements for the DFA methodology used.

Table 1. Geometric Information Required for DFA [20]

Rotational part	**Non Rotational part**
Basic Information	
Dimension L, D	Dimension A.B.C
	Axis symmetry X, Y or Z
Weight	
α, β symmetries	
Feature Information	
Steps **Grooves** **Cavities**	

Dimensions *L*, *D* for rotational parts and *A*, *B*, *C* for non-rotational ones, with the symmetry axes *X*, *Y*, and *Z* are computed from a part's minimal bounding box.

$\beta(\alpha)$ Symmetry is the angle through which a part must be rotated around its insertion axis (an axis perpendicular to its insertion axis) in order to repeat its orientation. The insertion axis is defined manually in *DFA* system.

Boothroyd–Dewhurst [1] dedicates specific Tables for defining the orientation of parts in automated assembly operations. Parts are oriented and distributed by bowl feeders. These bowls perform mechanical selections that define, step-by-step, the orientation of a part.

For a non-rotational part, 2D features in one or more planar projections, i.e. projections along *X*, *Y, Z* axes of the bounding box respectively named *Proj X*, *Proj Y*, *Proj Z*, could be used to partially or completely define the part orientation. 2D steps and 2D groove features are of interest when a part is progressing in a straight-line movement. But 2D features alone might not be adequate to orient a part. For example, if the component shape is a cube with a blind hole on one of its faces, no 2D feature could be detected in a 2D projection. The only feature that can be used to orient the part is its blind hole, i.e. a 3D feature. For a rotational part the Boothroyd–Dewhurst's Table refers to αsymmetry and βsymmetric (2D and 3D) features to orient the part relative to its end faces. The former Table exploits β symmetry and β asymmetric features to orient the part around its *Z* axis. In the B–D's DFA technique, operation time for automatic manipulation for a prismatic part or a cylindrical part is evaluated in terms of ease of handling using three Tables.

2. Assembly planning

Assembly planning process in many different industries uses the experience of product, assembly equipment in conjunction with geometric data from cad system to create assembly sequences [20]. The aim is to create them automatically by means of software. Many automated assembly sequencing systems use very complex algorithms to generate hard and soft constraints for feasible sequences. In this study the procedure followed for creating assembly sequencing started with the description of assembly structure.

2.1 Structure

To start the assembly design, product structure and their functions decided on at the conceptual design stage are necessary. It means some documentation for the component and subassembly in which it would be placed must be clarified. This information will help to the designer to understand the parallelism in subassemblies. Without the product structure and their functions it is not possible to understand the possible assembly configurations.

This information provided at conceptual design stage is an input data for assembly oriented design environment. Components with its functions and its subassembly positions were defined in assembly structure system.

2.2 Sequencing

This part of process is an interactively sequence developing issue. Partitioned subassembly definition detailed in the structure part is controlled in terms of manufacturability and ease of assembly. The single sequence or whole sequencing of assembly can be controlled at any time [21].
The main error positions in sequencing are the integral of link between sequences. By means of hard and soft constraints the errors encountered were minimized [22]. So it is possible to obtain a good sequence first time with fewer errors. However designer's choice is not this sequencing. It is provided by means of proactive design for assembly analysis.

2.2.1 Proactive design for assembly

Proactive design for assembly is a deciding mechanism. This mechanism concerns the three criteria, which are;

- Component group control
- Component Structure control
- Component detailed design control

After designer inserts a part in a subassembly, proactive DFA is used. Proactive DFA checks these three control mechanisms. If any of them guides for certain assembly constraints then the program advises these standard assembly methods. If the designers choice is different then the programs advice then the sub assembly is pointed out as expensive assembly color (red).

Component Group Control;
This control mechanism checks the uniqueness of the component inserted. If it finds any similarities or relationships with other components already inserted it advice standardizations or rationalizations for them. It searches the possibility of creating component family, standard assembly sequence or etc.
The aim of mechanism is arranged to advice standard parts and features reducing variability in manufacturing. This is provided by a link to the high level DFA principles such as minimizing number of parts in assembly, or using standard parts if it is possible.

Component Structure Control;
In this control mechanism the functional requirements of the component is checked systematically in terms of relative motion and materials, and replacing a group of components with a single piece. The correct information for the whole assembly was only obtained at the end of sequencing because there was no other way. However, by means

of direct interaction with the hierarchy and assembly sequencing diagram it is possible to get these information now. Therefore it is possible to see how successful the assembly during designing.
Since it is possible to check the success of the action performed online. The cost analyses of the assembly as a whole or for single operations can be performed also.

Component Detailed Design Control;
The success of DFA analysis depends on the understanding of the capabilities of manufacturing processes and adoption of different materials or manufacturing processes. For these reason detailed technical CAD data is necessary. If such a data is available then the different materials or manufacturing processes can be advised. Component attributes, estimated size and complexity determines many actions in assembly design.

DFA Proactive Evaluation;
In the light of these three control mechanisms it is possible to see many problems earlier. Traditional DFA methods only concerns the component cost estimates. They do not deal with the assembly problems etc. Decision on the process selection for simple volume and size check is possible during assembly sequencing.

3. Assembly-oriented design requirements

In the computer program designed for this study there are four sections, CAD model, assembly planning, assembly representation, and component interactions. To perform assembly sequencing expert knowledge representation and geometric inference data are necessary. For these reasons expert system technology and geometric reasoning were used.

3.1 Geometry definitions for components

In order to add a part into assembly important parameters of the part related to the geometry, such as shape complexity, symmetry, cross sectional area, degrees of freedom, insertion trajectories etc, are defined by means of material handling and insertion tables. In these tables, standardized material handing methods and insertion types together with degrees of freedom are defined with certain numbers. For each part one number is given for material handling and another number is also given for insertion type. The multiplication of these two numbers is used for the expert system evaluations. However in the determination of DFA criteria the most important factor is the existence of symmetry.

3.1.1 Part definition

The parts in the assembly are defined in such a way that DFA requirements are solved. Data structures used in *DFA*, especially feature classes, are described in this section. As shown in Figure 3, the part class aggregates technological characteristics represented by sets (i.e. the material), component assembly characteristics (i.e. mating with other parts), and geometric characteristics (i.e. symmetries, form features). Appending product assembly characteristics such as the rate per hour of an assembly operator, allows the part class to be extended to support a product. Form features are the most complex geometric characteristics used in *DFA*. Two classes were created to handle 2D and 3D features. The minimum information contained in each of them is as follows:

- The faceted B-Rep model of the feature's underlying geometry,
- The identified feature type, (protrusion, cutout, fillet etc.)
- The measured feature size, (height, base radius or equivalent base radius)
- The feature axis. (Insertion axes by vector)

Figure 1 highlights the two CAD representations used in *DFA*. CAD systems are currently used to describe a part by a parametric based B-Rep solid model in order to enable efficient product data management. In *DFA*, geometric and topological entities of such a model are described using the STEP AP203 standard. Downstream engineering applications however, such as finite element analysis, may use other representations of the part. A faceted B-Rep model generated by a tessellation process on the parametric based original model is used to solve the DFA requirement problem. The tessellation is based on a Delaunay method and built according to a user-specified precision.

Figure 1. Highlights the two CAD representations used in *DFA*. CAD systems are currently used to describe a part by a parametric based B-Rep solid model in order to enable efficient product data management. In *DFA*, geometric and topological entities of such a model are described using the STEP AP203 standard. Downstream engineering applications however, such as finite element analysis, may use other representations of the part. A faceted

B-Rep model generated by a tessellation process on the parametric based original model is used to solve the DFA requirement problem. The tessellation is based on a Delaunay method and built according to a user-specified precision.

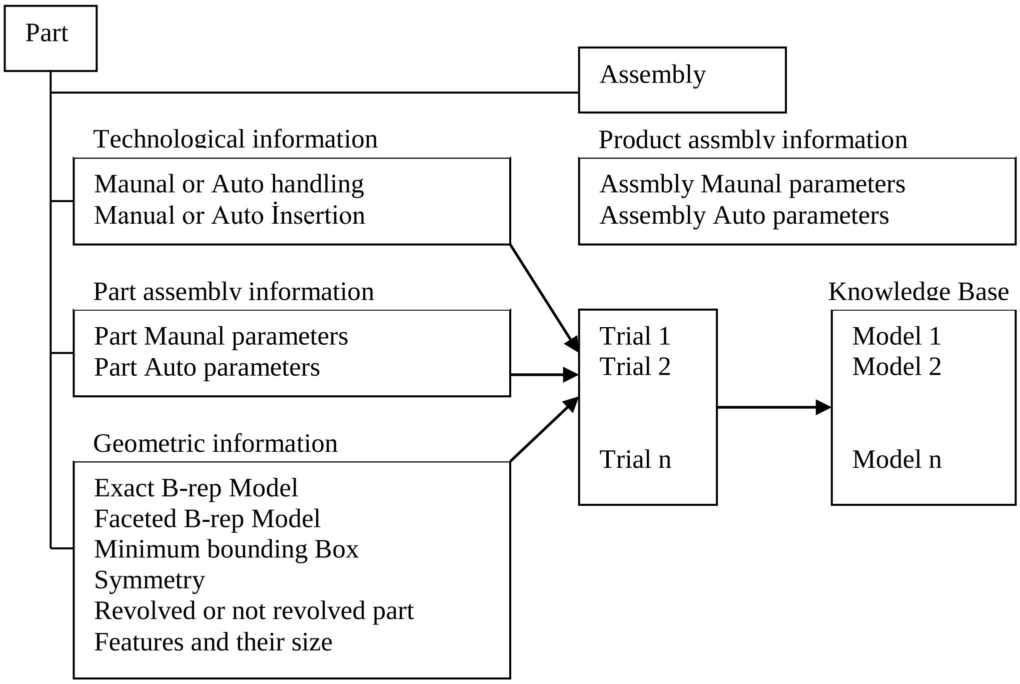


Figure 1. Part and Product data structures used in DFA Research Experiment.

3.1.2 Geometric information

Bounding Box and axis of rotation;

Although basic geometric requirements seem easy to find, dimensions and symmetries depend on the 3D coordinate system attached to the part. When using a 3D modeler, designers usually create parts in a global 3D coordinate system which is predefined for a given product or sub-product by the product architect. In the study the bounding box of each part and whether or not revolved parts are given by the designer as information.

Symmetry;

In DFA methodology, α and β symmetry properties are required to estimate the orientation and insertion part efficiency in the assembled product and to correctly orient the parts for automated assembly operations. Partial symmetry detection processes [23] can be useful for roughly orienting a part but are not convenient to automated insertion operations since the part must be well oriented. The method used here is based on existing works [24], and the complete rotational α and βsymmetry properties are detected. A part is considered to be α or β symmetric if an angle Φ exists (Φ=360° / n with $n \geq 2$) around an appropriate axis Δ such that the intersection volume between the part before and after rotation of Φ around Δ is null (or close to zero). For the DFA orientation approach, α and βsymmetry properties are sought for each part around the three axes of its associated coordinate system previously defined.

Feature Shape and Size;

All type of features on each part is defined in a table manually. Although some new works are performed for feature recognition for 2D and 3D, since it is out of the scope of this work and to difficult to use in the project period, the manual information tables are preferred. But the general rules of feature definition are used in this study also;

Rule 1. If there is only one face:

• If the face type is planar, then the feature type is Feat3d_Step.
• If the face is cylindrical or other, the feature type is Feat3d_Step or Feat3d_Groove depending on the location of the material.

Rule 2. If there is one face of which all edges are constrained then the type is Feat3d_Step (boss) or Feat3d_Cavity (blind hole or pocket) depending on the location of the material.

Rule 3. If each face has two constrained edges then the type is Feat3d_Cavity (hole or pocket through

all).
Rule 4. If the proportion of constrained edges (number of constrained edges/total number of edges) in each face is ≥0.5 then the type is Feat3d_Step (notch).
Rule 5. If the proportion of free edges (number of free edges / total number of edges) in each face is ≥0.5 then the type is Feat3d_Groove.

3.2 Expert system

As detailed data is not always available, any decision making process during the early design stages, must cope with indeterminacy. It is here that expert systems incorporating heuristics and expert knowledge with reasoning ability are best able to give guidance and suggestions on how to create an easy to assembly and manufacture product. There are many aspects in this environment which can benefit from expert system support, such as in the generation of the assembly sequences. It is here that the so called Expert Assembler can advise and help with which part(s) to start with, the Starting Component Advisor and which part(s) you should choose next, the Next Component Advisor.

The Starting Component Advisor provides suggestions on which part or parts should not be used as the base component(s). It is then easy to deduce which part(s) can be used as a starting component. Based on various attributes, a number of rules have been extracted from case studies and knowledge engineering exercises with experts in industry. [25]

The first rule is to find the part having only child relation.
Second rule is to find one child and one parent relations, i.e., intermediate parts are eliminated.
Third rule is finding heaviest parts. The heavier parts must be at the bottom of the machine.
These sequencing rules are useful for first and next part advisor algorithms.

The Next Component Advisor highlights the best possible next part to add to the assembly. It provides suggestions based on component group information in the assembly structure, the assembly strategy preferred by designer (such as bottom up, inside out), and some general rules extracted from case studies and industrial experts.

Both advisors are transparent to the user and are continually updated as more information becomes available. The reasoning behind the advisors is accessible and secondary to the user's decision making. The advisors are working with incomplete data as only the designer is aware of data unavailable to the system.

3.2.1. Orientation

This section discusses the definition of an optimal part orientation as it relates to DFA methodology. Boothroyd–Dewhurst uses *OE*, the orientation efficiency, and *FC,* the relative cost of the bowl feeder, to characterize part orientation. The following approach selects features maximizing the objective function *F*obj=OE/FC, analyzing 2D and 3D features orientation capabilities. The orientation is optimal in terms of *OE* and *FC* values. In order to define the part orientation, the algorithm used [20] evaluates the inner symmetries and searches for symmetric features for each 2D feature in a 2D projection and for each 3D feature in the part. Six symmetry attributes (*sym X*, *sym Y*, *sym Z, has sym X*, *has sym Y, and has sym Z*) are therefore added to the 2D and 3D feature data structures, as well as the feature usefulness. Symmetry *sym Z* is not of interest for 2D features. According to the given definitions, no 2D feature can be *sym Z* symmetry. Only features of type 'hole' or 'cavity' could have been *sym Z* symmetry but they are not used in the approach due to their insignificance in the DFA methodology. Furthermore, only 'hole' or 'cavity' features could have been both *sym X* and *sym Y* inner symmetries. Table 2 gives an indication of *F*2*dik* capabilities, the *k*th feature of the projection *i*, where $i \in \{X, Y, Z\}$, depending on the number of its inner symmetries and the number of its symmetric features.

In Table 2, a case 2 feature, having one inner symmetry and one symmetric feature, is a feature that has one symmetric feature around *Z* (i.e. *G*2*di*), and hence, this feature is case 2

When a feature is quoted as useless (case 4 in Table 2), a specific case is taken into consideration: if the 2D bounding box of the 2D projection is squared and if this feature is a groove, then it is useful and is case 2 in Table 2, as it has a G2d symmetric feature (i.e. its attribute *has_symZ* is true).

Lastly, if *F*2*dik* is case 2 in Table 2, a 3D characteristic axis, named Axis3D, is defined as the axis around which the part can be 180°-rotated so the face representing the projection *i* stays the same before and after this rotation. Table 3 provides the *F*2*dik*Axis3*D* value, which is one of the 3D coordinate systems of the minimal bounding box, corresponding to a symmetry axis (inner symmetry and/or symmetric feature) of *F*2*dik.*

Table 2. Orientation capabilities for a 2D feature [20].

No	# of Own Symmetries	# of Symmetric Features	Properties
1	0	0	Orients the part
2	0 1 1	1 0 1	This feature removes two rotation axes for the part
3	0.1 1 1	2 0 1	This feature and one of these symmetric features orient the part
4	0.1	3	This feature is useless
5	2	0.1, 2 or 3	Impossible

Table 3 Connection between 2D and 3D coordinate systems

Projection name	2D symmetry axis	3D axis
Proj X	X2d Y2d G2d	Y' Z' X'
Proj Y	X2d Y2d G2d	X' Z' Y'
Proj Z	X2d Y2d G2d	X' Y' Z'

3D feature orientation capabilities. Table 4 presents F3di capabilities, depending on the number of its inner symmetries and the number of its symmetric features.

Table 4. Orientation capabilities for a 3D feature [20]

No	# of Own Symmetries	# of Symmetric Features	Properties
1	0	0	Orients the part
2	0 1	1 0	This feature removes two rotation axes for the part
3	0.1	2	This feature and one of these symmetric features orient the part
4	0.1 1 2,3	3 1 0	This feature is useless except for cubic part
5	1 2,3	2,3 1,2,3	Impossible

4. The code environment

Instead of creating new CAD modeler in the code, the solid and assembly modeling ability of mechanical desktop version 6 software was employed. The four-layer model information was supported by access and expert system support was provided by Clips. The STEP AP203 format of assembly and IGES format of solids are transferred to the code. There are three different windows in the environment;

a. Hierarchy and model creation window; this windows is designed for creating hierarchy of the assembly consisting of individual part, and sub assemblies. Mechanical desktop 6 is connected to this window to perform these actions.

b. Assembly sequence creating window; this window is connected to the access database in which STEP format of assembly designed from mechanical desktop exists. Assembly definition is read and the expert rules created by Clips are used to perform sequencing actions in the current window. When the sequencing actions are finished the assembly definition is rewritten to access database. The new step format is read by mechanical desktop on the first window and then action is completed.

c. Comparison window; this window is used to compare the existing assembly sequences with the sequences obtained from the environment. By means of this window it is possible to obtain the efficiency of assembly oriented design. Efficiency is obtained by the ratio of time necessary for assembly divided by assembly time of product already manufactured. Flow chart of the code is given in following figure 2.

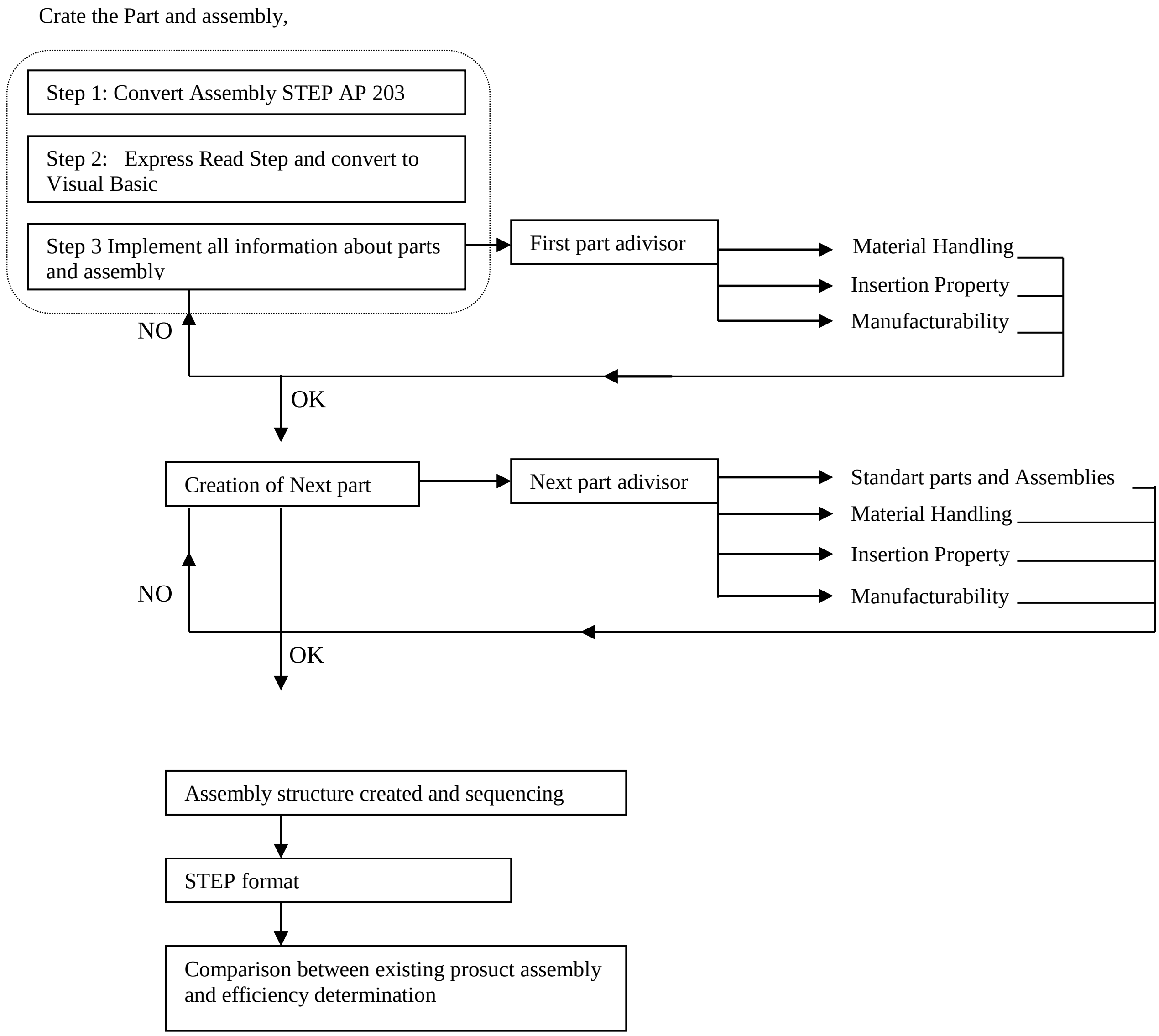


Figure 2. Flow chart of the code generated.

Adding a new part on the assembly needs the definition of insertion action itself. The type of insertion and additional equipments are also defined to make the proactive DFA evaluation efficient. The joining process and their types are also defined in this stage. In the environment there is a dialog box to define these joining and insertion actions. The interface of environment is given in figure 3. An already designed and manufactured product is used for this example. The assembly sequences of the existing product are input to the system for comparison purposes. The assembly structure is partially defined in the structure definition part and then designer start to add components and create suitable assembly sequences. When the designer adds a new part into the assembly, the suitability of part to the system is checked.

The similarities of added components or subassemblies are determined and distinctions between them are input to the assembly. Addition of a new part or subassembly is simultaneously represented in the Hierarchy and model creation window but Assembly sequence creating window is waiting for the necessary actions. When the sequencing is completed the part and its sequence appears also in this window.

Code also lets designer to create sequences manually if it is necessary. This sometimes is helpful because of the fact that all components may not yet be added or created.

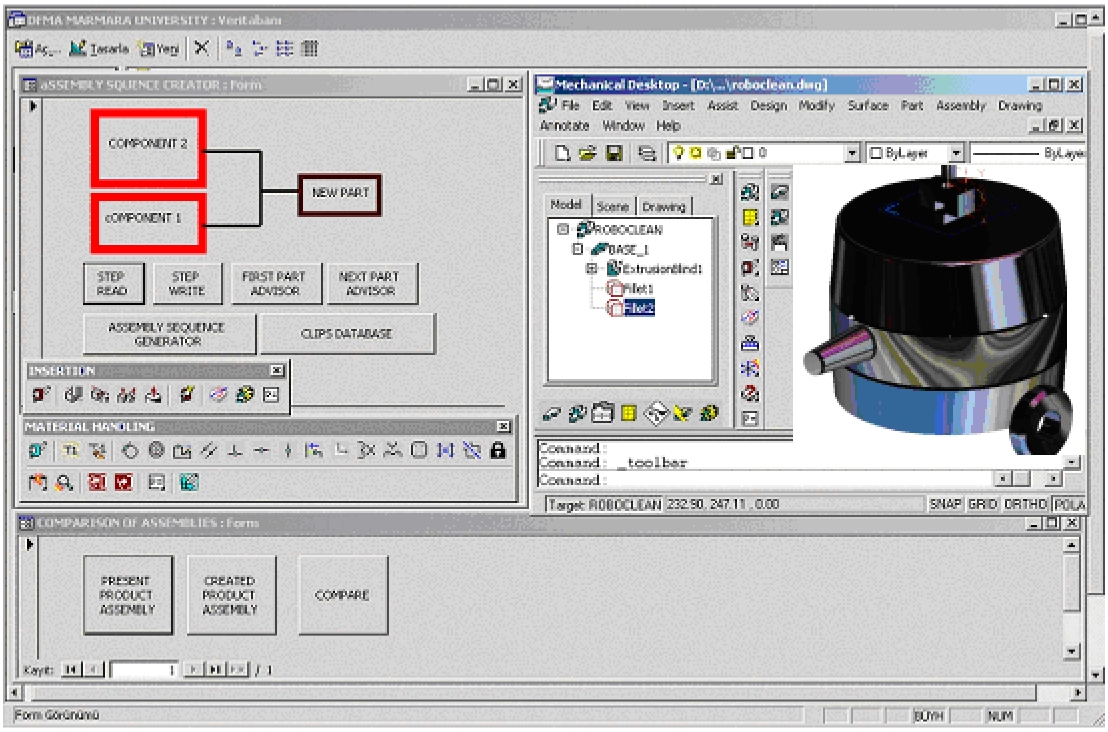


Figure 3 Interface of Code Environment

The accuracy of DFA process depends on the data given to code related to component. The attributes of component and geometric reasoning together with CAD data are necessary but not obligatory. The lack of these data does not stop the designer to perform sequencing. However information of component determines the sensitivity of the output.

The addition of a new component starts to DFM process in which the attributes of component together with its CAD data are taken into account to find the manufacturability of the studied part. The size and shape complexity of the component play important role in the decisions. The cost of the component determines its strength for selection. The cheaper, the component, the more preferable to use in assembly.

Starting part advisor is not used only for addition of the first part but it functions through out the process. It tries to find the best choice for the starting part at every new parts addition. It also searches to find the best starting part for subassemblies also.

Next component advisor also works for full time designing process. The reason for this the advise of the code for new part may be changed after addition of a new part or any update for attributes of previously inserted part will effect the assembly completely. Therefore proactive DFA code is dynamic program not stationary.

4.1 A case study; roboclean

The description of research experiment including details of methodology, data and results is outlined for RoboClean vacuum cleaner below. This machine is already a product market and its assembly structure and number of parts are known. Total number of parts in the model is 83 and assembly time is 4 hour. In the study the parts are defined according to the system explained in Figure 1.

The efficiency of assembly-oriented design is studied with respect to component-based approach. The efficiency is measured by the reduction of parts and the reduction of assembly time. The assembly of RoboClean, which is a brand for vacuum house cleaning machine was done and therefore all necessary data is available. This assembly and the data is input to the system. The assembly was drawn in CAD modeler and the attributes of parts and assembly, were input the access database manually for each part.

Step 1; After creating a new part the assembly is saved in STEP format.

Step 2; This STEP file is read by Express Program, which is able to convert the STEP format to visual basic code. In Visual Basic Code, each component of assembly is separately defined. In addition to this, the constraints between parts are also defined.

Step 3; After evaluation of assembly definition in visual basic environment, the next step is to define the geometric and handling information. This information are manually created for each part in Access database defining handling capabilities and manufacturability and geometric information such as bounding box, symmetry, features etc., The Table 5 define geometric attributes of the part shown in Figure 4, such as feature definitions.

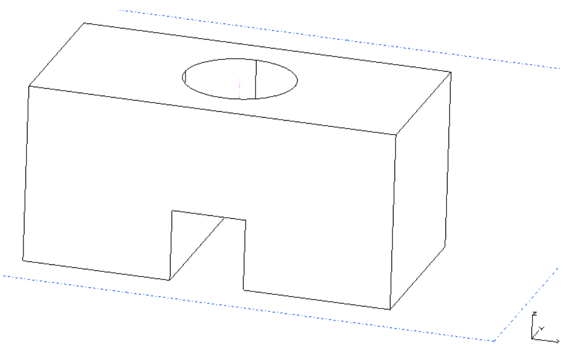


Figure 4. The fixture part used in RoboClean between Vacuum Pipe and Upper Body.

This part has two features the first one is at top surface and feature type is cavity with screw. The second one is at bottom surface and feature type is groove. In part description Table 5, the feature coordinates are given as feature position (x, y, z coordinates) and feature depth (x,-x, y, -y, z,-z).

Table 5. Part definition containing all feature information.

PARTDEF : Tablo

Part	X1	X2	Y1	Y2	Z1	Z2	Feature No 1	Type	R 1	L 1	A1	B1	C1	Feature No 2	Type 2	A2
1	0	100	0	50	0	50	1	CAVITY	1	20	30			2	GROOVE	40

Kayıt: 1 / 1

The next table defines the handling properties and the insertion constraints and hierarchies. The parent and child parts are also defined in this Table 6. Fixture part is used to connect and the Upper Body part and Vacuum Pipe as shown in Figure 5.

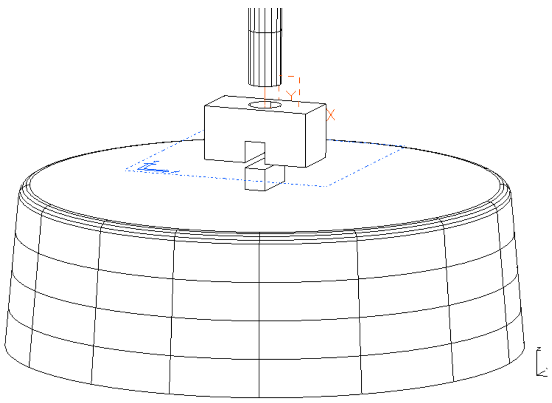


Figure 5. Assembling Parts in RoboClean.

Table 6. Handling and Insertion Information.

Handling : Tablo

Part	Ins Axeis	Ins Surf Type	Ins Surf No	Handling	Join Op	Join Part Type	Join Part No	Parent part	Chield Part
1	Z	RECTANGLE	1	MANUAL	SCREW	Standart	5	2	4

Kayıt: 1 / 1

The manufacturing process and their attributes are given in Table7.

Total number of parts in RoboClean is 68. The general shape is shown in Figure 6.

Table 7. Manufacturing information about parts in assembly.

manufacturing : Tablo

Part	Material	Weight	Material Price	Manuf Proces	Production Ti	Labor Cos
1	Plastic	185 gr	25 Cent	Machining	20 Minutes	8 $

Kayıt: 2 / 2

Figure 6 RoboClean.

First Part Advisor;
Visual Basic environment send the gathered information the first part advisor. The first part advisor checks the child parent relations and tries to find the part having only child relation. Second criteria are the number of features on each part. First part advisor put the parts in sequence according to number of features on each part. One parent and one child relation parts are defined as intermediate parts. This rule dramatically reduces the number of parts must be judged critically. Usually most of the parts are intermediate parts. The other rules help to first part advisor to perform its function easily. This procedure is repeated for each new part and the sequencing is obtained. Partially drawn assembly was the starting point in assembly-oriented design. Proactive DFA system was used and a new assembly was created. The differences between old and new designs are encountered;

The main difference is that the number of parts decreased in the new design. The change in numbers is not to much because of the old design is also good design because it has been modified many times. Although these modifications, two water tank produced with the same materials were merged to unit part in the new design. Due to these merging operations fixing parts were also reduced in the ratio of ½ approximately. Total number of parts changed from 83 to 68. The assembly drawing time was given as 4 hours for the old one and in new design this time was reduced to 3 hours 21 minutes. So number of part efficiency is 6.9 % ((86-68)/86x 100). The assembly time efficiency is 16.25 % ((240-201)/240x 100), as minutes although efficiency seems as 6.9 %, these values do not take account of modification times. In the assembly oriented DFA system at one trial these improvements has been provided.

Although this example is given as the evidence of efficiency of DFA system, it may not be so exact. Many tests must be done and the capability and the knowledge-based features of environment must be improved.

5. Conclusion

The software designed in this study is an implementation of the previously described proactive DFA code environment. In this code the comparison of the sequence generated is compared with the original assembly. Instead of creating a new CAD kernel environment, the link between commercially available CAD systems is integrated to the environment. This automatically reduces the developing time dramatically. By means of the proactive code early precautions can be taken by means of assembly sequences provided. In addition to this, the experience of assembly designers can be improved by comparisons between sequences of already finished products and the new proactive sequences.

In the case study the efficiency of number of parts is realistic and easily compared with the traditional assembly technique. But the efficiency of assembly time is not so realistic because of the assembly time of proactive DFA system depends on the handling times provided by designer whereas assembly time of traditional system is gathered from the manufacturer, therefore it valid.

In this paper, optimally manufacturability and easy to assembly designs are described by means of proactive DFA process. In this process assembly sequence generation is defined manually together with geometric reasoning and expert advisor system. Although an example provides the success of such environments to reduce time and error, further work on such codes is obligatory for increasing functionality of expert advisor and information content for the components. Since the measurability of information for CAD data for a component or features of it is not defined yet, this type studies must continue.

The new features of such codes must be knowledge based. Environment must learn the functions of product and the realistic sequences of assembly after completing assembly and apply this knowledge to similar designs.

References

[1] G. Boothroyd, P. Dewhurst, W. Knight, Product Design for Manufacture and Assembly, Marcel Dekker, 1994.

[2] K.G. Swift, Design for Assembly Handbook, Salford University Industrial Centre, UK, 1981.

[3] C.S. Syan, U. Menon (Eds.), Concurrent Engineering: Concepts, Implementation and Practice, Chapman and Hall, London, 1994.

[4] G.Q. Huang (Ed.), Design for X—Concurrent Engineering Imperatives, Chapman and Hall, London, 1996.

[5] Tollenaere M "Conception de Produits Mechaniques Hermes ‘ ISBN 2- 86601-694-7, 1998.

[6] C J Barnes,G F Dalgleish,G E M Jared, H Mei,K G Swift," Assembly Oriented Design", ISATP 99, Porto, Portugal, pp 45-50, July 1999.

[7] S. Miyakawa, T. Ohashi, „The hitachi assemblability evaluation method",Proceeding on International conference on product design for assembly, 1986.

[8] R. Bäßler, T. Schmaus, Procedure for Assembly-oriented Product Design. 2nd Int Conf on Product Design for Manufacture & Assembly, Rhode Island,1987.

[9] T. De Fazio, S. Rhee, D. Whitney, A Design-specific Approach to Design for Assembly (DFA) for Complex Mechanical Assemblies. Proc. IEEE Int. Symposium on Assembly and Task Planning, 152-157, 1997.

[10] R.Li, C.Hwang. A Framework For Automatic DFA System Development. Computers In Industrial Engineering 22(4):403-413, 1992.

[11] D Polfreman, "An Investigation Into the Evaluation Of Assembly Sequences", MSc Thesis, Cranfield University, 1996.

[12] R Sturges, M Kilani, "Towards an Integrated Design For an Assembly Evaluation and Reasoning System", Computer Aided Design, 1992, 24(2):67-79.

[13] T.De Fazio,D.Whitney. Simplified Generation of All Mechanical Assembly Sequences. IEEE Journal of Robotics and Automation 3(6): 640 - 658, 1987.

[14] A.Sanderson, "L.Homem de Mello. And/Or Graph Representation Of Assembly Plans", IEEE Transactions on Robotics and Automation. 6(2):188 - 199, 1990.

[15] Tate SJ, Jared GEM, Brown NJ, Swift KG. An introduction to the Designers' Sandpit. DETC/DFM-14014. Proceedings of the Fifth ASME Design for Manufacturing Conference, Baltimore, Maryland; September 2000.

[16] C J Barnes, G. F. Dalgleish, G. E. M. Jared, K. G. Swift, "Decision support for sequence generation in assembly oriented design environment", Robotics and Computer-Integrated Manufacturing Volume 20, Issue 4. August 2004, Pages 289-300

[17] C J Barnes, G. F. Dalgleish, G. E. M. Jared, K. G. Swift, S.J. Tate "Computer support for proactive DFA", Proceedings of Second International Symposium on Tools and Methods for Concurrent Engineering, pp 313-

320, Manchester, UK, 1998.

[18] C J Barnes, G. E. M. Jared, K. G. Swift, ”A pragmatic approach to interactive assembly sequence evaluation”, Proceedings of the I MECH E Part B Journal of Engineering Manufacture, 1 April 2003, vol.217, no. 4, pp. 541-550(10)

[19] N.J. Brown, K.G. Swift, J.D. Booker ”Joining process selection in support of a proactive design for assembly”, Proceedings of the I MECH E Part B Journal of Engineering Manufacture, 1 October 2002, vol.216, no. 10, pp. 1311-1324(14)

[20] O. Coma, C.Mascle, and P.Veron, “Geometric and form feature recognition tools applied to a dsign for assembly methodologis”, Computer aid design, vol. 35, Issue 13, pp.1193-1210, 2003.

[21] T Abell, GP Amblard, DF Baldwin, TL De Fazio, M-C M Lui, DE Whitney, Computer Aids For Finding, Representing, Choosing Amongst,, and Evaluating The Assembly Sequences of Mechanical Products, Computer-Aided Mechanical Assembly Planning, Kluwer Academic Publishers, USA,, 383-436, 1991.

[22] M.Santochi, G.Dini. Computer Aided Planning Of Assembly Operations : The Selection Of Assembly Sequences. Robotics and Computer Integrated Manufacturing, 9(6):439 - 446, 1992.

[23] Tate S, Jared G, Swift K. Detection of symmetry and primary axes in support of proactive design for assembly. Proceedings of the Fifth Symposium on Solid Modeling and Applications, Ann Arbor, USA; 1999. p. 151–8.

[24] R.R. Martin and D. Dutta, Tools for asymmetry rectification in shape design. J Syst Engng 6 (1996), pp. 98–112.

[25] G.Jared, M.Limage, I.Sherrin, K.Swift, Geometric Reasoning and Design For Manufacture. Computer Aided Design, 26(7) : 528 - 536, 1994.