An Intelligent Computer-Aided Assembly Process Planning Methodology for Mechanical Parts

Dillip Kumar Biswal and Dibakar Bandopadhya*

Abstract: **An intelligent computer-aided assembly process planning (CAAPP) methodology has been developed that generates an automatic assembly sequence for mechanical parts. The proposed method is relied on knowledge based expert system. It can generate all feasible assembly sequences for complete assembling of a mechanical product from a given set of disassembled parts. A scheme for representation of the input data on the various constituent parts of the assembly is developed along with a set of knowledge based rules. Both techniques are developed for selecting the most suitable base part automatically taking into account of the various constraints. The method can generate all feasible sequences for assembling the various components together starting from a base part to the constituent part to the finished assembly. The proposed methodology is flexible, readily adaptable to the demanding assembling environments and also proves to be less time consuming.**

Keywords: Assembly Sequence Planning, Knowledge Based Expert System, Assembly Constraints, Base Part

I. INTRODUCTION

Assembling of mechanical parts is one of the key manufacturing activities to be found in almost every industrial sector. Yet, it is one of the most complicated and time consuming activities accounting for a large proportion of the manufacturing cost. As a result, there is much interest in reducing the cost of assembly activities. One way of achieving this is to improve the assembly process planning to determine the sequences in which parts of subassemblies are arranged together. Assembly process planning requires significant amount of expertise and experiential knowledge. This demand that the traditional product assembly sequence should be planned manually by an expert process planner with necessary skill and experience.

Dibakar Bandopadhya, Faculty of Department of Mechanical Engineering, IIT Guwahati, Assam, India. Email: dibakarb@iitg.ernet.in

However, manual process planning is highly laborious and time consuming activity that requires considering numerous constraints and large number of potential assembly sequences, particularly where complex assembly process is involved. Thus, essentially there is much interest in developing systematic and computer-aided assembly process planning techniques to automate the assembly sequence processes. Assembly process involves integration of components or parts sequentially to create a product to operate satisfactorily for a predefined function **[1]**. Generally assembly sequence planning consists of three main steps: assembly modeling, generation of all feasible assembly sequences, and optimization of assembly sequences **[2]**. The data for assembly sequence generation can be provided by the user or can be directly extracted from the CAD model of the product. Generally a typical product can have a large number of possible assembly sequences and can rise enormously with the increase in number of components **[3]**. However, due to assembly constraint the number is reduced significantly.

In the past, various methods have been proposed by the researchers to generate assembly sequences. A question-and–answer procedure was developed, where questions are answered with a precedence relation between two components, to generate the assembly sequence **[3]**. Where, a liaison sequence graph (LSG) is used to represent the assembly sequence. Concept of AND/OR graph for compact representation of all feasible assembly sequences was also proposed **[4]**. An algorithm was developed based on case-based reasoning (CBR), to generate assembly sequence procedures **[5]**. Adjacency matrix methods were also used in Cartesian coordinate system to generate the assembly planning sequence **[6]**. Further, algorithm was developed and implemented using a LISP program to generate all feasible assembly sequence by considering the precedence relationship and combining the various subassembly operations **[7]**. A methodology was proposed to generate all feasible assembly sequences along three directions (X,Y,Z) in Cartesian coordinates system utilizing contact and translational functions **[8-9]**. Where, an assembly sequence table (AST) was developed to represent assembly sequence of the fit. A connector-based relation model (CBRM) was developed to generate a feasible assembly sequence **[10],** further two stages **[11]** and three stages **[12]** decision support procedure were also proposed to generate the parts subassembly sequences. Further, a mathematical model was developed to determine the optimum assembly sequence from the AND/OR graph of a mechanical product **[13]**. A web-based 3D assembly sequence system was developed as framework for customers to assist their design in decision making **[14]**. Nowadays there is an increased application of artificial intelligence (AI) technique such as expert systems, neural networks, genetic algorithms etc., in the design and planning of products and assemblies **[15]**. To reduce the search space, heuristic rules and genetic algorithms (GAs) have also been used **[16]** to find out the feasible assembly sequence of a product. Other procedures, such as using Ant Colony **[17-18]**, Neural Networks **[19-22]**, and knowledge-based expert systems **[23-24]** were developed as a tool to generate optimum or sub-optimum assembly sequences.

It is important to choose appropriate part as the base part for initializing the sequence as this affects the remainder of the sequence. An inappropriate part can lead to inefficient or even infeasible sequence. The base part is the foundation of the assembly sequence planning as proper selection of a base part can reduce the number sequence generations **[1]**. Assumptions and various criteria have been adopted by researches in the past for selection of the base part **[1, 11, 25-27]**. Further, it is difficult to develop an exact algorithm for assembly sequence planning as the exact information for assembly sequence planning cannot be expressed quantitatively **[27]**.

The main objective of the work is to develop an intelligent assembly planner capable of generating assembly sequences automatically. The assembly planner is purely relied on a knowledge based expert system. In order to determine the feasible assembly sequences, the contact and degree of freedom or disassemble information among the parts are used. Contact relation gives the information about the contact between two parts along the three principal directions. Disassemble relation gives the information about the direction of disassemblies between two parts in three principal directions. This information provides by the user. Assembly constraints such as contact relations, mobility and stability features are used to generate the feasible assembly sequences. Initially the expert system chooses a suitable base part using the rules and then chooses the next part to be assembled with the base part. Subsequently, a feasible assembly sequence is generated for the given product satisfying all the constraints. The methodology for generating all the feasible assembly sequences for a given product is presented systematically, followed by illustrations of assembling results of two mechanical product and discussion with examples and finally the conclusions.

II. PROPOSED METHODOLOGY

Expert systems are integrated with computer programs that are shown to be a powerful tool for solving many real-domain problems for specific knowledge based systems **[28]**. Particularly, in this work an expert system approach has been proposed for generating automatic assembly sequence planner for mechanical parts. For this, a rule based expert system shell CLIPS (an acronym for C language integrated production system) **[29]** is developed and has been implemented on a PC. CLIPS is based on forward-chaining strategy and rule-based production system language.

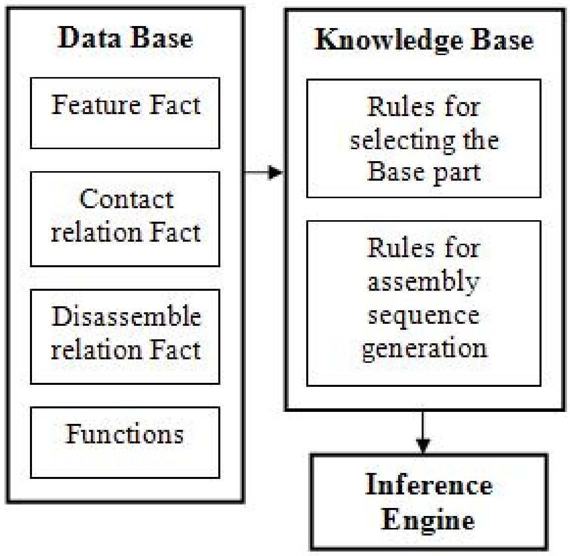

Figure 1. Overall structure of the expert system module based on assembly sequence planning

Figure 1 shows the schematic diagram of the proposed assembly sequence planning methodology with its modules that consists of a database, a rule driven knowledge base and an inference engine. The database of the expert system contains the data files, stores all the information about the product for all feasible assembly sequence generation. The data files contain features of the part i.e., the contact relations and the relative degrees of freedom in translation between pair of components along three principal directions. This information provides by the user constitutes the input to the expert system. The input information includes properties like mass and other attributes such as dimensions, tolerance with other features, etc. In addition, this also include various functions used for performing certain calculations e.g., function to compare the largest mass for different parts, function to determine the total number of mating links for different parts of a product etc.

The concept of template has been used for representation of the input data to the CLIPS expert system shell. Template consists of a list of named fields called slots use to store various attributes and information **[30]**. For example, the input data on a feature may be entered in the format of a template as shown in **Template 1**. Similarly another example has been shown in **Template 2** and is used to define the format for representation of the input data for contact relations, such as, the direction and whether the two parts are in contact or not in three principal directions. The input data on a typical feature may be entered as shown in **Template 3**.

```
(deftemplate MAIN: component
(slot number (type INTEGER)(default ?NONE))
(slot name (type SYMBOL)(allowed-symbols
base_part BRACKET BLOCK WASHER BOLT))
(slot length (type NUMBER))
(slot width (type NUMBER))
(slot radius (type NUMBER))
```

Template 1: Features template showing the representation of various attributes

```
(deftemplate MAIN:: mating-links
(multislot numbers (type INTEGER)(default ?NONE))
(multislot name (type SYMBOL)(allowed-symbols base_part BRACKET BLOCK WASHER BOLT))
(multislot axis_dir (type SYMBOL)(allowed-symbols POS_X NEG_X POS_Y NEG_Y POS_Z NEG_Z)(default ?NONE))))
```

Template 2: Format for feeding contact relation information

```
(deffacts MAIN::component_list
(component (number 3)(name WASHER) (mass 1.00)(radius 20))
(component (number 1)(name BRACKET) (mass 3.00)(length 150)(width 150))
(component (number 4)(name BOLT)(mass 0.890)(radius 15))
```

Template 3: Format for representation of the input data

The input data are saved as data file with an extension .clp and when requires, are loaded into the expert system shell during the time of execution. Alternatively, it may also be directly entered manually by the user through a user interface. In addition, the database also comprises of functions used for performing certain calculations, such as finding the largest mass value of a component with respect to other parts. Sometimes finding the total number of mating links of a component with respect to other components may be in demand and as given in **Functions 1and 2**.

```
(defglobal ?*largest-mass* = 0
?*component-having-largest-mass* = 0)
(deffunction MAIN::largest-mass(?f1)
(bind ?m1(fact-slot-value ?f1 mass))
    (if(> ?*largest-mass* ?m1)
        then (bind ?*largest-mass* ?*largest-mass*)
        else (bind ?*largest-mass* ?m1))
(return ?*largest-mass*))
```

Function 1. Function to find out the largest mass from the various mating parts

The function calculates and compares all the components mass value and return the value of the largest mass. To calculate the total number of mating links, a function of the form can be used and expressed as:

```
(deffunction MAIN:: total-number-mating-links (?f1 ?f2 ?f3)
(bind ?l1 (length$(fact-slot-value ?f1 axis_dir)))
(bind ?l2 (length$(fact-slot-value ?f2 axis_dir)))
(bind ?l3 (length$(fact-slot-value ?f3 axis_dir)))
(bind ?l4 (+ ?l1 ?l2 ?l3))
(return ?l4))
```

Function 2: Function to calculate the total number of mating links of a mechanical product

The 'function 2' used to obtain the number of mating contacts for components with each other, and returns the value of total number of contacts for each component. The knowledge base of the expert system consists of assembly sequence planning in the form of rules. A set of rules have been devised for selecting the base part and for assembly sequence generation between a pair of parts taking into account of the assembly precedence constraints. A rule for selecting the largest mass component may be of the following type.

```
(defrule MAIN: largest-mass "return the largest mass"
(declare (salience 100))
  ?f1 <- (component)
=>
  (largest-mass ?f1)
 (component-having-largest-mass?f1))
```

Rule 1: Rule for selecting the component having largest mass among the various parts

This rule selects the component having the largest mass compared to the other parts of the given product. Mating link is calculated using data contact function. The connectivity between the pair of components of the assembly can be expressed in terms of contact functions. Contact function ensures the presence or absence of contact between the pair of components. This can be obtained by finding the movement of components along the Cartesian coordinates only, e.g. +X, +Y, +Z, -X, -Y, -Z directions. The rule for obtaining the total number of contacts in each component is given as,

```
(defrule MAIN:: total-number-mating-links "return
        the total number of mating links"
      (declare (salience 100))
      ?f1 <- (mating-links (numbers ?n1 ?n2))
      ?f2 <- (mating-links (numbers ?n1 ?n3))
      ?f3 <- (mating-links (numbers ?n1 ?n4))

(test (and(<> ?n2 ?n3)(<> ?n3 ?n2)(<> ?n2 ?n4)(<>
?n3 ?n4)))
  =>
 (assert (number-of-mating-links (number?n1) (value
(total-number-mating-links ?f1 ?f2  ?f3))))
(retract ?f1 ?f2 ?f3))
```

Rule 2: Rule for asserting the total number of mating links in each component.

In order to identify the base part from a given set of components automatically, a set of rules have been devised based on heuristics and expert knowledge. A host of constraints are considered such as heaviest mass, number of contacts between parts for selecting the base part. Some heuristics rules for selecting the base part are given below.

i. A part that is large and heavy in relation to the other parts is selected as the base-part
ii. The part which shares the most mating links with the others is taken as the base part
iii. The non fastener part is also considered as the base part

```
(defruleMAIN:: base-part-selection-1"return the
component number with the largest mass"
?f1 <- (component (number? number))
(not (component (name base_part)))
=>
(if (= ?number ?*component-having- largest-mass*)
then (duplicate? f1 (name base_part)))))
```

Rule 3: Rule for selection of the base part during assembly

The above rule selects the large and heaviest component as best fit for the base-part. Once the base-part for the assembly operation is selected, the other parts are considered successively for the assembly operation. Assigning the base-part as the first step of the assembly sequence, other components are assembled together one-by-one satisfying all the constraints till the last component is assembled. The following rules are devised that illustrate an example of assembling between two components.

```
(defrule MAIN::precedence_constraint_between_two_part
(declare (salience 70))
(component (number ?n1)(name base_part))
(component (number ?n2))
(and (mating-links (numbers ?n1 ?n2)(axis_dir
$??axis_dir1 $?))
(mating-links (numbers?n1?n2)(axis_dir$?NEG_Y$?)))
(test (and(<> ?n1 ?n2)(<> ?n2 ?n1)))
=>
(assert (subassembly ?n1 ?n2)))
```

Rule 4: Rule for assembling two components in an assembly sequence

The above rule performs the following tasks,

- First selects the base-part as the first component to be assembled with the other components
- Identifies the second component next to be assembled that satisfies the contact relation constraint and stability constraint due to gravity
- The part that satisfies the above constraints chooses as the second component and completes the subassembly with the first component

The following rule gives an example of assembling between more than two components.

```
(defrule
MAIN::precedence_constraint_more_than_two_part
(declare (salience 40))
(subassembly ?n1 ?n2)
(component (number ?n3))
(or(and(mating-links(numbers ?n1 ?n3)(axis_dir $?
?axis_dir1 $?))
(mating-links (numbers ?n1 ?n3)(axis_dir $? NEG_Y
$?)))
(and (mating-links (numbers ?n2 ?n3)(axis_dir $?
?axis_dir1 $?))
(mating-links (numbers ?n2 ?n3)(axis_dir $? NEG_Y
$?))))
(and (disassemble-dir (numbers ?n1 ?n3)(axis_dir $?
?axis_dir2 $?))
(disassemble-dir(numbers ?n2 ?n3)(axis_dir $?
?axis_dir2 $?)))
(test (and(<> ?n1 ?n2 ?n3)(<> ?n1 ?n3 ?n2)(<> ?n2
?n1 ?n3)(<> ?n2 ?n3 ?n1)(<> ?n3 ?n1 ?n2)(<> ?n3
?n2 ?n1)))
=>
(assert (subassembly ?n1 ?n2 ?n3)))
```

Rule 5: Rule for assembling more than two components

The above stated rule performs the following tasks:

- First, it takes into account of the beforehand subassembly results between the two components
- Identifies the third component to be assembled that satisfies the contact relation, disassemble relation constraint, and stability constraint due to gravity
- The third component completes the subassembly with the earlier sub-assembled parts
- The process continues till the last component is assembled

III. ASSEMBLY SEQUENCE GENERATION

Three types of constraints are taken into consideration for generation of feasible assembly sequence. These include connectivity constraint, mobility constraint and stability constraint due to gravity. Connectivity constraint specifies the contact relation in terms of an assembly operation. Mobility constraint expresses the disassemble relation between components whereas, stability constraint stand for the instability of the assembly for non-contact between the two parts along the direction of gravity. **Figure 2** shows a typical five components assembly product represented in three-dimensional coordinate system. The contact between the component 1 and 2 is represented as, $C_{12} = (C_1, C_2, C_3, C_4, C_5, C_6)$, a (1x 6) binary function and can be expressed as,

$$C_i = (0,1) \tag{1}$$

where, $i = 1,2,3, \dots ,6$
$C_i = 1$, refers that a contact exists in the direction of i, i.e. part 2 is in contact with the part 1 along the direction i. $C_i = 0$ indicates the absence of contact along the direction i. Similarly for disassemble relations, $D_{12} = (D_1, D_2, D_3, D_4, D_5, D_6)$, a (1x6) binary function represents the degree of freedom motion between part 1 and 2 and can be expressed as,

$$D_i = (0,1) \tag{2}$$

where, $i = 1,2, \dots ,6$. Further, $D_i = 1$, indicate that part 2 has the freedom of translational motion with respect to the part 1 along the direction i while, $D_i = 0$ refers freedom of motion does not exist along the direction i. The feasibility of two component subassemblies can be verified from the contact relation between those two parts. For any pair of components, at least one contact relation should exist between the two components out of three principal directions to make the pair a feasible subassembly. Let us investigate the case of subassembly between the pair (1, 2) and (2, 5). The contact relation between the pair is given in **Table 1**.

Table 1. Contact relations between component (1, 2) and (2, 5) for the part shown in **Figure 2**

Pair	C_1	C_2	C_3	C_4	C_5	C_6
(1,2)	0	0	0	1	1	0
(2,5)	0	0	0	0	0	0

As shown in the **Table 1**, two entries out of six directions are 1, thus, subassembly between the pair (1, 2) is possible. Mathematically, this contact relation can be expressed as,

$$(C_1 \vee C_2 \vee C_3 \vee C_4 \vee C_5 \vee C_6) = 1 \quad (3)$$

While, no contact relations does exist between components (2, 5) in any direction, i.e.

$$(C_1 \vee C_2 \vee C_3 \vee C_4 \vee C_5 \vee C_6) = 0 \quad (4)$$

The feasibility of subassemblies with more than two components can be verified from both contact and disassemble relationships among them. If the resultant truth-value of contact relation is equal to 1, implies that there is a contact between the parts. Further, if the resultant truth-value of disassemble relation becomes 1; indicate that assembly operation is possible. For clarification, say the component (5) is to be assembled with the subassembly (1, 2), the contact and disassemble relation between the component (1, 5) and (2, 5) is given in the **Table 2 and 3**.

Table 2. Contact relations between components (1, 5) and (2, 5) for the part as shown in **Figure 2**

Pair	C_1	C_2	C_3	C_4	C_5	C_6
(1,5)	0	1	0	0	0	0
(2,5)	0	0	0	0	0	0
C	0	1	0	0	0	0

$$C = C_i(1,5) \square C_i(2,5) \quad (5)$$

where, $i = 1,2, \dots ,6$. Thus, there is a contact between pair (1, 2) and the part (5) as,

$$(C_1 \vee C_2 \vee C_3 \vee C_4 \vee C_5 \vee C_6) = 1 \quad (6)$$

Table 3. Disassemble relations between component (1, 5) and (2, 5) for the part as shown in **Figure 2**

Pair	D_1	D_2	D_3	D_4	D_5	D_6
(1,5)	1	0	1	1	1	1
(2,5)	1	0	1	1	1	1
TT	1	0	1	1	1	1

$$TT = D_i(1,5) \square D_i(2,5) \quad (7)$$

where, $i = 1,2, \dots ,6$. The component (5) can be assembled, if at least one entry of TT is 1, i.e.,

$$TT_1 \vee TT_2 \vee TT_3 \vee TT_4 \vee TT_5 \vee TT_6 = 1 \quad (8)$$

In the above truth table, it is found that five entries of TT has a value 1, this implies that the component (5) enjoys five collision free disassemble direction with respect to subassembly (1, 2). This further entails that the subassembly between (1, 2, 5) is possible and the component (5) can be assembled in any five collisions free disassemble direction. Further, taking into account of the third constraint i.e., stability due to gravity, it is observed that, between subassembly (1,5), there is a collision free disassemble direction along the gravity. This is further understood that along '–Y' direction the pair can be disassembled subjected to a condition that there is nothing to support it. Thus, even the subassembly is made between the components (1,5); the pair turns out to be an unstable one. Thus subassembly of pair (1,5) alone is not feasible unless some kind of supporting mechanism holds the object (5).

Thus in summary, for generation of the assembly sequence, the expert systems first find out the base part. A feasible subassembly is obtained then between a pair consists of two parts where primary component is the base part. In the subsequent steps, the assembly sequence systematically generates all feasible assembly sequences with the base part satisfying various constraints. Finally, the parts sequence completes the full assembly of the product from base to the last part of the series. The assembly sequence is generated automatically using the proposed rules that select the parts orderly. The assembly sequence building that explained in section 2 and 3 is shown in **Table 4**.

IV. ILLUSTRATION: RESULTS & DISCUSSIONS

Two examples of assembling the mechanical product are presented in **Figure 2** and **Figure 3** and are used to demonstrate the proposed technique.

4.1. Example 1: Application of Rules

The mechanical object as shown in the Figure 2 consists of 5 components, component (1) is a bracket, component (2) , (3), and (4) are block, washer and nut respectively while the component (5) is a bolt. All the information such as mass, contact relations, disassemble relations are presented in the format explained earlier and are stored in data file with extension .clp. These data files are loaded in the CLIPS environment along with knowledge based rules for solving the assembly problem. On execution of the expert system program, all feasible assembly sequence of the product is generated automatically. Table 4 summarises the results of the output generated by the expert system assembly planner for the mechanical connecter as shown in **Figure 2**. By assigning more priority to the heaviest mass as the base part, expert system chooses item (1) as the base part. Finally, the expert system generates two assembly sequence i.e. sequence (**1-3-2-4-5**) and sequence (**1-2-3-4-5**) for assembling of the product. For both the cases, expert system chooses part 1 as the base part, while parts 3, 2, 4, 5 in order to complete the first sequence. However, for the other sequence, expert system chooses part 2, prior to the part 3 to complete the assembly. This indicates that part 2 or part 3 any one of them could be chosen after selecting the base part (1) to complete the assembly. The following notations are chosen for the axis representation of assembly sequence:

$$+X = 1, +Y = 2, +Z = 3, -X = 4, -Y = 5, -Z = 6$$

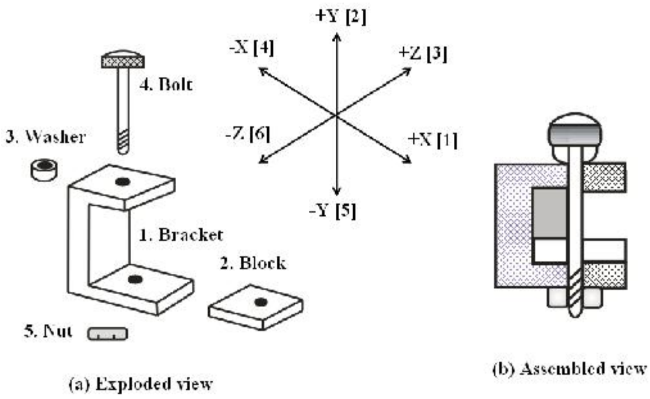

Figure 2. An example of assembly of a mechanical connector in Cartesian coordinate system

Table 4. Feasible subassembly and assembly sequence generation for the mechanical product as shown in Figure 2

4.2. Example 2

The ball-point pen assembly, as shown in **Figure** 3 has six components. The product consists of body (1), cap (2), button (3), head (4), tube (5) and the final component (6) is ink. Even though, component (6) is a liquid, it has been considered a solid proponent for implementation of the proposed technique. Thus, in addition to the previous rules few more rules are devised further for assembling of the product.

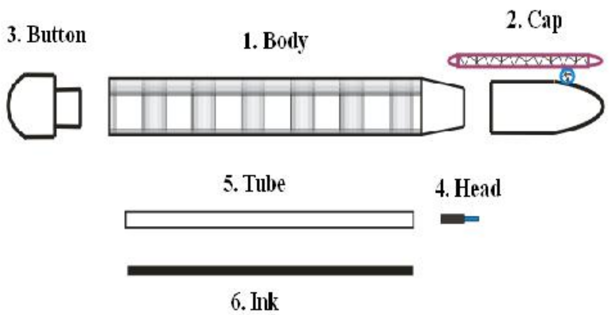

Figure 3. Exploded view of a ball-point pen assembly

Rule 6: An encapsulate product i.e. consists of both exterior and interior body parts, the inner component(s) should be assembled first

Rule 7: For clearance fit and press fit assembly, press fit assembly should be done first

Applying the rule 6 to the above example, it is found that component (4), (5) and (6) are the inner parts of the ball-point pen setup. Therefore, these components should be assembled first prior to the other components. On execution of the expert system and assigning more priority to the total number of mating links, it is found that expert system chooses the component (4) to be assembled first. **Table 5** shows the assembly and subassembly sequence for the inner components and **Table 6** represents the final subassembly and assembly sequence of the ball-point pen as shown in **Figure 3**.

Table 5. Feasible subassembly and assembly sequence generation for the inner components as shown in Figure 3

Level	Task	Parts
1	Subassembly	(4 5)
2	Subassembly	(4 5 6)

Applying the rule 7, it is found that component (3) assembled before the component (2), as button of the body is a tight-fit compared to cap and the body assembly. Thus final assembly of the product comes out to be, head-tube-ink-body-button and finally the cap.

Level	Task	Parts
1	Subassembly	(1 2)
	Subassembly	(1 4)
	Subassembly	(1 3)
2	Subassembly	(1 3 2)
	Subassembly	(1 3 4)
	Subassembly	(1 2 4)
	Subassembly	(1 2 3)
3	Subassembly	(1 2 3 4)
	Subassembly	(1 3 2 4)
4	Assembly	(1 3 2 4 5)
	Assembly	(1 2 3 4 5)

Table 6. Feasible subassembly and assembly sequence generation for the pen as shown in Figure 3

Level	Task	Parts
1	Subassembly	(4 5)
2	Subassembly	(4 5 6)
3	Subassembly	(4 5 6 1)
4	Subassembly	(4 5 6 1 3)
5	Assembly	(4 5 6 1 3 2)

In summary, the proposed methodology adopted a pure expert system approach to solve the assembly sequence planning problem. The description of the heuristic rules have been organised into different modules of the expert system, such as the database, the knowledge base and the inference engine. The modular nature of the expert systems imparts added flexibility to the proposed approach. For the assembly sequence as discussed in **Table 4** and **6**, expert system took only 2.390s and 6.485s on a Core 2 Duo; 3.00GHz PC with 1GB RAM to generate the above outputs. Any modification in the proposed assembly planning technique can be done by simply modifying the rules in the knowledge base of the expert system. Also a new knowledge can be easily acquired from the expert system by adding new rules to its knowledge base. The expert system program developed by authors can be integrated readily with other modules of CAAPP system such as feature extraction from CAD files for mating relations, disassemble relations and mass properties.

V. Conclusion

A knowledge based expert system methodology for assembly sequence planning of mechanical items has been proposed. The proposed method is capable of generating the assembly sequence of a given product automatically. Two illustrative examples have been discussed using the proposed methodology and it is found that assembly sequence planning can be accomplished automatically by investing very limited amount of time. The developed expert system is more flexible as compared to the combined expert system and therefore readily adaptable to changing design and manufacturing environments. Further, there is scope for optimization of the assembly sequence planning based on various criteria. The proposed methodology and expert system could be much useful in automobile industry as well.

REFERENCES

[1] Chen W C, Tai P H, Deng W J and Hsieh L F (2008), “A three –stage integration approach for assembly sequence planning using neural networks”, Expert Systems with Applications, vol.34, pp. 1777-1786.

[2] Zhao J and Masood S (1999), “An Intelligent Computer-Aided Assembly Process Planning System”, International Journal of Advanced Manufacturing Technology, Vol.15, pp. 332–337.

[3] De Fazio T L and Whitney D E (1987), “Simplified generation of all mechanical assembly sequences”, IEEE Journal of Robotics and Automation, Vol. RA-3 No.6, pp. 640–658.

[4] Homen de Mello L S and Sanderson A C (1991), “A correct and complete algorithm for the generation of mechanical assembly sequences”, IEEE Transaction on Robotics and Automation, Vol.7, No.2, pp. 228–240.

[5] Pu P (1992) “An assembly sequence generation algorithm using case-based search techniques”, Proceeding of the 1992 IEEE International Conference on Robotics and Automation, Nice, France, pp. 2425-2430.

[6] Dini G and Santoshi M (1992), “Automated sequencing subassembly detection in assembly planning”, Annal CIRP, Vol.41, pp.1-4.

[7] Ben-Arich D, and Kramer B (1994), “Computer-aided process planning for assembly: generation of assembly operation sequences”, International Journal of Production Research, Vol.33, pp.643-656.

[8] Gottipolu R B and Ghosh K (1997), “Representation and selection of assembly sequences in computer-aided assembly process planning”, International Journal of Production Research, Vol. 35, No. 12, pp. 3447 – 3465.

[9] Gottipolu R B and Ghosh K (2003), “A simplified and efficient representation for evaluation and selection of assembly sequences”, Computers in Industry, Vol. 50, pp.251–264.

[10] Yin Z P, Ding H, Li H X, and Xiong Y L (2003), “ A connector-based hierarchical approach to assembly sequence planning for mechanical assemblies” Computer-Aided Design, Vol.35,pp.37-56.

[11] Barnes C J, Jared G E M and Swift KG (2004), “Decision support for sequence generation in an assembly oriented design environment”, International Journal of Robotics and Computer-Integrated Manufacturing, Vol. 20, pp.289-300.

[12] Chen R S, Lu K Y, and Tai P H (2004), “Optimizing assembly planning through a three-stage integrated approach”, Production Economics, Vol.88, pp.243-256.

[13] Lambert A J D (2006), “Generation of assembly graphs by systematic analysis of assembly structures” European Journal of Operation Research, Vol.168, pp.932-951.

[14] Chen C F, Soo M K, and Yu K M (2006), “Customer-driven collaborative product assembler of internet-based commerce”, Concurrent Engineering, Research and Applications, Vol.14, pp.99-109.

[15] Zha X F (1998), “Integrated intelligent design and assembly planning: A Survey”, International Journal of Advanced Manufacturing Technology, Vol.14, pp.664-685.

[16] Marian R M, Luong L H S and Abhary K (2006), “A genetic algorithm for the optimization of assembly sequences”, Computers & Industrial Engineering, Vol.50, pp. 503–527.

[17] Failli F and Dini G (2000), “Ant colony systems in assembly planning: a new approach to sequence detection and optimization”, Proceedings of the 2nd CIRP International Seminar on Intelligent Computation in Manufacturing Engineering, pp.227–232.

[18] Wang J E, Liu J H and Zhong Y F (2005), “A novel ant colony algorithm for assembly sequence planning”, International Journal of Advanced Manufacturing Technology, Vol. 25, pp.1137–1143.

[19] Hong D S and Cho H S (1995), “A neural-network-based computational scheme for generating optimized robotic assembly sequences”, International Journal of Engineering Application of Artifical Intelligence, Vol.8, pp. 129-145.

[20] Huang H H, Wang M H and Johnson M R (2000), “Disassembly sequence generation using a neural network approach”, Journal of Manufacturing Systems, Vol. 19, pp. 73-82.

[21] Chen W-C, Tai P-H, Deng W-J, Hsieh L-F (2008), “ A three-stage integrated approach for assembly sequence planning using neural networks” Expert Systems with Applications, Vol.34,pp.1777-1786.

[22] Chen W-C, Hsu Y-Y, , Hsieh L-F, Tai P-H (2009), " A systematic optimization approach for assembly sequence planning using Taguchi method, DOE, and BPNN" Expert Systems with Applications, DOI:10.1016/j.eswa.2009.05.098.

[23] Huang Y F and Lee C S G (1991), "A Framework of Knowledge-based Assembly Planning", Proceedings of the IEEE International Conference on Robotics and Automation, Sacramento, California.

[24] Mazouz A K, Souilah A and Talbi M (1991), "Design of an expert system for generating optimal assembly sequence", Computer-Aided Engineering Journal, December, pp. 225-259.

[25] Ye N and Urzi D A (1996), "Heuristic rules and strategies of assembly planning: experiment and implications in the design of assembly decision support system", International Journal of Production Research, Vol. 34, No. 8, pp. 2211-2228.

[26] Chandra M, Sarin S and Thomas P E (1995), "Computer aided assembly sequencing: Die set application", 17th International conference on computers and industrial engineering, vol.29, No.1-4, pp.687-691.

[27] Deshmukh A, Paul Yung J and Wang H Pin (Ben) (1993), "Automated generation of assembly sequence based on geometric and functional reasoning", Journal of Intelligent Manufacturing, Vol.4, pp. 269-284.

[28] Chen Z (1999), "Knowledge acquisition assisted by CLIPS programming", Engineering Applications of Artificial Intelligence, Vol.12, pp. 379-787.

[29] Giarratano J C (1998) Expert Systems: Principles and Programming, P W S Publishers Co. Boston, USA.

[30] Hazarika M , Deb S and Biswal D K (2008), "An Expert System Based Computer-Aided Process Planning Methodology for Setup Planning of Machined Prismatic Parts", Competitive Manufacturing-Proceeding of the 2nd International and 23rd AIMTDR Conference,IITMadras,Chennai,India,pp.1049-1054.

Dibakar Bandopadhya is currently working as Assistant Professor, Department of Mechanical Engineering, In Indian Institute of Technology Guwahati. He did his PhD from Indian Institute of Technology Kanpur and he was involved with an international project (INDO-ITALIAN, N.ER/6), where research on Development of Force Reflecting Surgical Robot Using Micro Fingertip Sensor was undertaken.

Dillip Kumar Biswal is currently working as Associate Professor, Department of Mechanical Engineering, In Eastern Academy of Science and Technology (East), Dist:-Khurda, Bhubneswar, Odisha, Pin: 754001. He did his PhD from Indian Institute of Technology Guwahati, Assam.