International Journal of Industrial Engineering, 15(1), 92-103, 2008.

Design of An Artificial Neural Network for Assembly Sequence Planning System

Cem Sinanoğlu

Erciyes University, Engineering Faculty
Department of Mechanical Engineering,
38039 Kayseri, Turkey
Email: {Cem Sinanoğlu, csinan@erciyes.edu.tr}

The problem of assembly sequence generation is complex and has proven to be difficult to solve. Various method have been used in attempting to solve the problem, including mathematical modeling and search techniques. This paper presents an investigation for analyzing assembly sequences of assembly systems with a proposed neural network predictor. The proposed neural network has three layers with recurrent structure. A fast learning algorithm Backpropagation (BP) algorithm is employed for updating the weight parameters of the proposed network.

Significance: An ordered and sequenced connection process of parts existing in a system is known as assembly. Assembly sequence planning is the choice of the most optimum one. Although much research has been conducted in the field, there are still some shortcomings related to the sequence planning of large system. The developed assembly sequence planning system is graph based and automatic. By optimization, the best assembly sequences are determined. Due to parallel structure and fast learning with small errors, the neural network can be employed as an algorithm for this kind of assembly systems.

Keywords: Assembly, Binary vector representation, Sequence planning, Neural nets, Contact matrix

(Received 29 July 2005; Accepted in revised form 26 March 2007)

1. INTRODUCTION

An assembly can be defined as the overall function after joining individual parts, each of which has an independent function. It is possible to divide an assembly into various subassemblies depending on its complexity level. Although intensive research efforts have been made in the field of assembly sequence planning, there are still some problems to be solved. For example, there exist some shortcomings to support full automatic assembly sequence planning or to obtain assembly plans for large systems.

In an automated assembly system, a product is completed through a series of assembly operations according to a predefined sequence. Partially assembled components are automatically transported from one machine to another machine until the final product is organized. Yee and Ventura (1999) describe a method for finding optimal assembly sequences in an assembly system. All feasible assembly sequences can be represented by an AND/OR graph which provides all feasible geometric configurations and relationships among the components of a product. The AND/OR graph is converted into a Petri net which can be formulated as a linear program with the objective of minimizing the total assembly time or cost. A dynamic programming algorithm is developed to find the optimal sequences from the Petri net since too much computational effort is required to obtain the optimal solution of the linear program by utilizing the simplex method and interior point methods. The algorithm has the computational complexity of O (nm), where n and m denote the number of assembly operations and base components, respectively, and it is more efficient than the former methods. Three assembly products are provided to validate the algorithm.

Simaria and Vilarinho (2001) studied simple assembly line balancing problem with parallel workstations. This work presents a new mathematical programming model for the simple assembly line balancing problem with parallel workstations that allows the user to control the way workstations are 'parallelised'. As in the conventional procedures the user can limit the number of replicas allowed for each workstation, but in this novel approach the user can also define a minimum task time to trigger the replication of the workstations. Another important characteristic of the model is that it simultaneously minimizes the number of workstation and smoothes the workload among them. Due to the model complexity a simulated annealing approach is proposed to solve it.

Nanthavanij and Yenradee (2001) studied the selection of appropriate degree of automation of component insertion method for printed circuit board (PCB) assembly was studied. Components can be inserted into printed circuit boards (PCBs) using a manual, semi-automated, or automated method. The degree of automation of component insertion affects required capital investment, process capacity, product quality, etc. This paper discusses a systematic procedure for selecting appropriate degree of automation of component insertion method for PCB assembly. Initially four mutually exclusive alternatives ranging from manual to fully automated insertion are defined. Fourteen decision criteria are nest defined and grouped into four categories, namely, cost, insertion operation, product, and cognitive activity. These criteria are compared using a pair wise comparison technique to determine their relative importance towards component insertion. The results are synthesized to obtain global priorities of alternatives, from which the most appropriate component insertion method can be selected.

In addition to these, another related research issue is that parts existing in the assembly should have geometry for easiness in assembling (Boothroyd, 1994). In this respect, product design should directly influence the assembly sequence planning that imposes the final product. Moreover, the selection of a special assembly sequence should have influences over assembly processes. These processes are the selection of assembly equipment, the design of tools and fixtures, construction of subassemblies, and assembly time and cost (Laperriere and Marghy, 1994). Assembly sequence planning plays a very important role in the manufacture of many mechanical products consisting of non-standard parts. The research in this field tries to get insight into the relations among assembly sequence planning, design and assembled products (Seow and Devanathon, 1993)**.** The main purpose here is to obtain a number of possible assembly sequence plans.

An exact assembly sequence plan consists of assembly, operations, existing assembly techniques and some details of relations between parts (Garrod and Everett, 1990)**.** Some researchers apply assembly sequence planning in reverse order. Recent researches on assembly sequence planning have established some important issues related to concurrent engineering analyses (Laperriere and Marghy, 1994)**.** These issues are the representation of a product to be assembled, the generation of assembly sequence plans and the determination of precedence constraints, the representation of resulting assembly sequence plans and the selection of the optimum assembly sequence plan(s).

Since many assembly sequences share common subsequences, attempts have been made to create more compact representations that can encompass all assembly sequences. Therefore, the works in this field are graph-based approaches that represent all feasible assembly sequences. The basic approach to find the most suitable assembly sequences is to represent the assembly system in a space where it is possible to explore all different assembly sequences. Thus, some criteria may be used to obtain these sequences. Then optimum assembly sequence can be selected by the application of some other criteria. These criteria may include the number of tool changes, part replacement and clamping on the jigs, concurrency in operations, reliable subassemblies, etc**.**

In this article, an assembly sequence planning system, which determines assembly sequences of any product, is explained. This planning system is used to graph based approach in the representation of product and assembly sequence plans. The neural network structure is very suitable for this kind of problem. The network is capable of predict the assembly plans of the assembly system. The network has parallel structure and fast learning capacity.

2. SYSTEM DESCRIPTION

An assembly planning system is used to assembly's connection graph for the representation product. Parts and relations among these parts are represented by this graph. Contact relations between parts are supplied by scanning module. Scanning module scans various assembly views of any product whose assembly view is produced by Unigraphics in the computer environment and determines to contact and intersection relations between parts (Sinanoglu and Börklü, 2004). These relations are formed as a matrix. The system constitutes assembly's connection graph of product to be assembled, by applying Boolean operators on the elements of the contact matrices according to certain rules. Moreover, intersection relations are also accumulated in a form of intersection matrices for determination of geometric feasibility of assembly states later.

In the assembly planning system, binary vectors represent assembly states. Therefore, all binary vector representations, whether corresponding to assembly states or not, are produced by the system. By evaluating assembly's connection graph and binary vector representation simultaneously with the scanning module, vector representations corresponding to assembly states are determined. Some of the assembly states cannot take part in a feasible assembly sequence. The determination of the assembly states not corresponding to feasible assembly sequence is achieved with the analysis module. The analysis module controls all assembly states according to stability, subassembly and geometric feasibility constraints. Boolean operators on the elements of intersection matrices do determination of geometric feasibility. The feasible assembly states and assembly sequences are represented by a directed graph. Assembly states supplying constraints are settled down in the nodes of directed graph hierarchically by the system. Any path from the root node to terminal node in the directed graph corresponds to feasible assembly sequence. The optimum one is selected from the feasible assembly sequences with optimization module. This system can determine the least suitable sequence among the feasible sequences. Moreover, the neural network approach has been employed for analyzing feasible assembly sequences and optimum assembly sequence for sample product. Due to parallel learning structure of the network, the proposed neural network has superior performance to analyze these systems.

3. ARTIFICAL NEURAL NETWORKS

Artificial Neural Networks (ANNs) are non-linear mapping systems with a structure loosely based on principles observed in biological nervous systems. A network is specialized different functions by varying the connection topology and the values of the connecting weights. Usually, the processing units have responses like (Canbulut et al., 2004)

$$y = f\left(\sum_i u_i\right) \quad \cdots \quad (1)$$

where u_i are the output signals of hidden layer to output layer, $f(.)$ is a simple non-linear function such as the sigmoid, or logistic function.

In some cases the choice of training method can have a substantial effect on the speed and accuracy of training. The best choice is dependent on the problem, and usually trial-and-error is needed to determine the best method. In this study, logistic function and backpropagation learning algorithm are employed to train the proposed ANNs. Backpropagation is a minimization process that starts from the output and backwardly spreads the errors. The weights are updated as follows;

$$\Delta w_{ij}(t) = -\eta \frac{\partial E(t)}{\partial w_{ij}(t)} + \alpha \Delta w_{ij}(t-1) \quad \cdots \quad (2)$$

where η is the learning rate, and α is the momentum term. Online Back propagation updates the weights after each pattern is presented to the network.

In this study, the logistic function is used to hidden layers and output layers. Linear function is taken for input layer. Logistic function is as follows;

$$y = f(x) = \frac{1}{1+e^{-x}} \quad \cdots \quad (3)$$

The linear function is;

$$y = f(x) = x \quad \cdots \quad (4)$$

Training and structural parameters of the network are given in Table 1.

Table 1 Structural and training parameters of the neural predictor

Neural Network	η	μ	n_I	n_H	n_O	N	AF
	0. 1	0	1	10	4	500000	Logistic

4. REPRESENTATION OF A PRODUCT

In order to model the assembly system, the assembly's connection graph whose nodes represent assembling parts and whose edges represent connections between parts is used. Some researchers used assembly's connection graph in order to model assembly systems. But, these researchers assume that whenever two parts are assembled, all the contacts in between are established. In the assembly planning system developed, this assumption has rejected by forming the assembly's connection graph in a different way. For the current analyses, it is assumed that exactly two subassemblies are joined at each assembly task, and after those parts have been put together, they remain together until the end of the assembly process.

Due to this assumption, an assembly can be represented by a simple undirected graph $\langle P, C\rangle$, in which $P = \{p_1, p_2, ..., p_N\}$ is the set of nodes, and $C = \{c_1, c_2, ..., c_L\}$ is the set of edges. Each node in P corresponds to a part in the assembly, and there is one edge in C connecting every pair of nodes whose corresponding parts have at least one surface contact. In order to explain the developed approach for modeling assembly systems used for this research, we will take a sample assembly shown in side view in Figure 1.

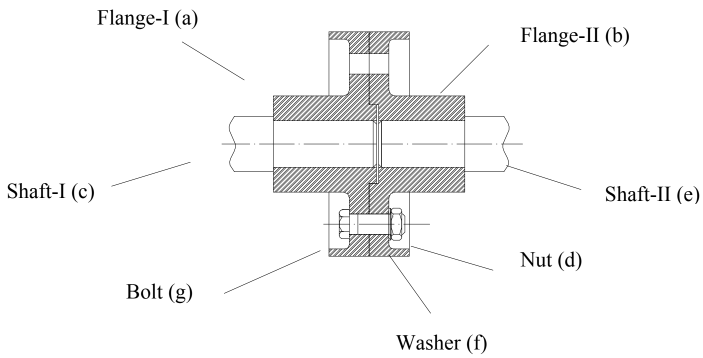

Figure 1 The coupling assembly system

The sample assembly is a coupling consisting of seven components. These are: Flange-I, Flange-II, Shaft-I, Nut, Shaft-II, Washer and Bolt. These parts are represented respectively by the symbols of a, b, c, d, e, f and g sets. For this particular situation, the connection graph of the assembly has the set of the nodes as $P = \{a, b, c, d, e, f, g\}$ and the set of the connections as $C = \{c_1, c_2, c_4, c_5, c_6, c_7, c_8, c_9\}$. The contact matrices are used to determine whether there are connections between parts in the assembly state. These matrices are represented by a contact condition between a pair of parts as an $\{A, B\}$. The elements of these matrices consist of "$Boolean$" values of "$true(1)$" or "$false(0)$". For the construction of contact matrices, the first part is taken as a reference part. Then whether this part has a contact relation in any i axis directions with other parts is examined. That relation, if any, is defined as "$true$"; otherwise that is defined as "$false$". The row and column element values of contact matrices in the definition of six main coordinate axis directions are contact relations between the parts that constitute the coupling assembly. To determine these relations, the assembly's parts are assigned to rows and columns of the contact matrices. Contact matrices are square matrices and their dimensions are 7x7 for the coupling assembly. Some example contact matrices of the coupling assembly system can be determined as follows;

$$B_{-x} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_{-y} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \quad B_{-z} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

The double side contact relation concept has been suggested for the construction of the assembly's connection graph. For instance, in the manner of $\{a, b\}$ ordered pair of parts, it is sufficient to determine contacts related in the ordered direction to indicate contact in any direction. For this reason, an "$\vee(or)$" operator is applied to these parts. It is also necessary to have contact in any direction for inversely ordered pairs of part in the assembly's connection graph. If these values are "1" for every ordered pair of parts, then there should be edges between corresponding nodes of the assembly's connection graph. Therefore, an "$\vee(or)$" operator is applied to this ordered pair of parts. Table 2 shows contact relations regarding $\{a, b\}$ and $\{b, a\}$ pairs of parts.

Table 2 Contact relations of $\{a, b\}$ and $\{b, a\}$ pairs of part

$c_1 : (a, b)$	x	y	z	$-x$	$-y$	$-z$	"$\vee(or)$"$\Rightarrow$	"$\wedge(and)$"$\Downarrow$
a/b	0	1	1	1	1	1	1	1
b/a	1	1	1	0	1	1	1	1
								1

In this table, $\{a,b\}$ ordered pair of parts meets to at least one contact condition in the related direction of $(0 \vee 1 \vee 1 \vee 1 \vee 1 \vee 1 = "1")$. $\{b,a\}$ pair of parts also meets to at least one contact in the related direction of $(1 \vee 1 \vee 1 \vee 0 \vee 1 \vee 1 = "1")$. An $"\wedge(and)"$ operator is applied to obtain these values. Therefore, after applying $"\wedge(and)"$ logical operator, a $"1"$ value is obtained. Therefore, the connection between these parts is represented as an edge in the graph of connections.

Figure 2 shows the coupling graph of connections, which has seven nodes and nine edges (connections). For example, there are not any contacts between the Flange-I and Shaft-II, nut, washer. Therefore, for instance, the graph of connections does not include an edge connecting the nodes corresponding to the Flange-I and the nut.

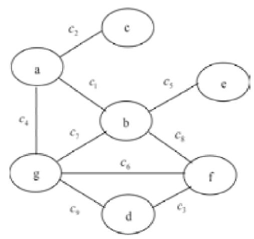

Figure 2 The graph of connections for coupling assembly

This assembly's connection graph determines an assembly model of the coupling system. It can be used as an input by an assembly sequence planning system.

5. ASSEMBLY SEQUENCES GENERATION AND REPRESENTATION

In the developed approach an L-dimensional binary vector can represent a state of the process $(x = \{x_1, x_2, ..., x_L\})$. The elements of these vectors define the connections between assembly components. Assembly sequences are obtained with enumeration of the vectors, which correspond to assembly states.

Based upon the establishment of the connections, the elements of these vectors may have the values of either $"1"$ or $"0"$ at any particular state of assembly task. For example, the i^{th} component x_i would have a value of $true$ if the i^{th} connection were established in that state. Otherwise, it would have a value of $false$. Not all binary vector representations correspond to an assembly state. In order to determine assembly states, the established connections in binary vectors and the assembly's connection graph are evaluated together.

There are nine edges in the example assembly connection graph. Consequently, there are nine elements of vector and 9-dimensional binary vector can represent that $[c_1 c_2 c_3 c_4 c_5 c_6 c_7 c_8 c_9]$. For instance, the binary vector can represent the initial state of the assembly process for the product shown in Figure 2 $[000000000]$ whereas the final state can be represented by $[111111111]$. In the assembly sequence planning system developed, first of all binary vector representations must be produced. Whether these binary vector representations constitute assembly states or not is determined by using assembly's connection graph. In the determination of the assembly states, a new approach employing Boolean operators has been developed.

For example, if the first task of the assembly process is the joining of the Shaft-I to the Shaft-II, the second state of the assembly process can be represented by $[100000000]$. This vector can be represented by $\{\{a,b\},\{c\},\{d\},\{e\},\{f\},\{g\}\}$. Figure 3 shows these connections. It can be seen from Figure 3 that it is not necessary to establish any connection so that c_1 connection between part $\{a\}$ and $\{b\}$ can be established. Therefore, $[100000000]$ vector is an assembly state. Vectors with only one established connection form an assembly state.

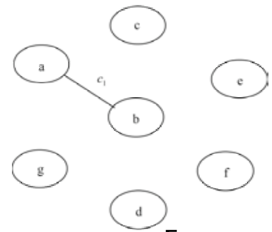

Figure 3 c_1 connection in $[100000000]$ vector

After the determination of the assembly states, the determination of the feasibility of these assembly states for any assembly sequence is required. Assembly states not correspond to feasible assembly sequences must be eliminated by some assembly constraints. These constraints are as follows;

Subassembly Constraint: This constraint defines the feasibility of a subassembly of a set of partitions to established connections in assembly states. In order to form a subassembly set of partitions, it is not a set of partition that contains a pair of part not in a contact relation in the assembly's connection graph. For instance, there is no connection between the Shaft-I (c) and Shaft-II (e) for the coupling system. Because of that it does not meet the subassembly constraint set partitions containing $\{c, e\}$ set of partition.

Stability Constraint: A subassembly is said to be stable, if assembling parts maintains relative position and they do not break contact during the assembly operation. It is assumed that whenever an assembly state meets a subassembly constraint, it also meets stability condition.

Geometric Feasibility Constraint: In the developed assembly sequence planning system, the geometric feasibility of the assembly states are determined with the intersection matrices and applying Boolean operators on these matrices. The elements of intersection matrices take into consideration interference conditions during the joining parts. In the determination of geometric feasibility, $"\wedge"$ and $"\vee"$ logical operators are applied to elements of intersection matrices. This operation must utilise established connections, that is, joining pairs of parts. In describing intersection matrix elements, interference is taken into consideration, which occurs while the reference part is moving together with another part in the related axis direction. If there is interference during this transformation motion, intersection matrix elements are $"0"$ if not, they are defined as $"1"$. Interference relations are located in rows and columns of the intersection matrices. Therefore, intersection matrices are formed. Some of these matrices are as follows;

$$A_{-x} = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \quad A_{-y} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \quad A_{-z} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

In order to determine whether assembly states with only one established connection are geometrically feasible or not, it is necessary to apply Cartesian products between ordered pairs of parts which represent established connections and parts which are not in this ordered pair of parts. In this situation, different intersection tables are obtained and these tables are used to check geometric feasibility. For instance, the geometric feasibility of $[000000101]$ is determined as follows;

In this assembly state, the connections of c_7 and c_9 have been established. The geometric feasibility of $[000000101]$ can be determined to applying the Cartesian product between the (b, g) and (d, g) pairs. The Cartesian product between them is given as follows; $(b, d).(b, g).(g, d)$. Table 3 shows the interference of these ordered pairs of parts.

The result of $"\wedge"$ and $"\vee"$ logical operators is "0". This result explains that the $[000000101]$ assembly state is geometrically unfeasible. Similar operations must be applied to other assembly states of the coupling system. Therefore, feasible assembly sequences of coupling system can be determined. But, if any assembly state is geometrically unfeasible in any assembly sequence, this sequence will be geometrically unfeasible. Three hundred and seventy three feasible assembly sequences for the seven-part coupling system have been determined.

Table 3 The geometric feasibility of $[000000101]$

$c_7:(b,g), c_9:(d,g)$	x	y	z	$-x$	$-y$	$-z$	
b/d	1	0	0	0	0	0	
b/g	0	0	0	1	0	0	
g/d	1	0	0	0	0	0	
$(b/d \wedge b/g \wedge g/d)\Downarrow$	0	0	0	0	0	0	$"\vee(or)"\Rightarrow$
							0

6. ASSEMBLY SEQUENCE OPTIMIZATION

Developed assembly planning system can be determined to find the optimum assembly sequence. In this section, an optimization approach is explained for the developed assembly sequence planning system. For this purpose, the coupling assembly system is taken as an example. It has been obtained from feasible assembly sequences in previous sections. In order to optimize the assembly sequence, two criteria are developed, weight and the subassembly's degree of freedom. First certain costs are assigned to edges of directed graph depend on these criteria, and then the total cost of each path from root node to terminal is calculated the minimum cost sequence is selected as an optimum one. This optimization also enables determination of the least suitable sequences from the feasible sequences.

6.1 Optimization of Weight Criterion

In order to determine the optimum assembly sequence, all assembly states in an assembly sequence must be taken into consideration. The heaviest and bulkiest part is selected as a base part and then the assembly sequence continues from heavy to light parts. The parts with the least volume, i.e. connective parts, like bolts and nuts must be assembled last. The weights and volumes of parts were calculated automatically with a CAD program (Unigraphics). Therefore, determination of the costs of assembly states is necessary to obtain a optimum feasible assembly sequence. After that these costs are used as a reference to different assembly states. Calculated weight costs of assembly states in the assembly sequence are compared with reference weights. The difference of weight is multiplied by unit weight value (100). The weights of parts of the coupling system are as follows; a: 0.3945 kg, b: 0.345 kg, c: 0.25 kg, d: 0.25 kg, e: 0.0165 kg, f: 0.0093 kg, g: 0.0042 kg

Using the weight criteria, the total established connection weights of each assembly state in optimum assembly sequence could be determined. The total weight of assembly states in the optimum sequence according to weight criteria can be defined as;

$$Ow_m = \sum_{i=1}^{n}(W) \text{ (Required Weight)} \quad \ldots \quad (5)$$

where W is the weight of assembly states, (n) is the number of the established connections and (m) is the order of assembly states. Ow_m is the required weight of assembly states in the optimum assembly sequence. Each connection weight for the coupling system is as follows:

Connection c_1 between part $\{a\}$ and part $\{b\}$ has been established. The number of established connections is $n=1$. Some of the weight of established connection is:

$$Ow_1 = \sum_{i=1}^{n=1} W = W(c_1) = 0.7395kg,\ Ow_3 = \sum_{i=1}^{n=3} W = 1.7455kg,\ Ow_6 = \sum_{i=1}^{n=9} W = 3.3747kg,\ Ow_4 = \sum_{i=1}^{n=5} W = 2.4934kg$$

In order to determine the optimum assembly sequence, the weights of all assembly states in assembly sequences are calculated. This weight is expressed as;

$$Cw_m = \sum_{i=1}^{n}(W) \text{ (Calculated Weight)} \quad \ldots \quad (6)$$

where W is the weight of assembly states, (n) is the number of the established connections and (m) is the ordered of assembly states. After that, (Ow_m) is used as a reference for different assembly states. Calculated weights of assembly states in assembly sequences are compared with reference weights. This weight difference (Dw) is multiplied by unit weight value $(Uwv = 100)$. The result is the weight costs of assembly states (Wc).

For example, in the assembly state of $[100000000]$ the calculation of weight costs of two different assembly states in the feasible assembly sequence are given as follows;

It has been used to establish c_1 connection between part $\{a\}$ and part $\{b\}$. The number of the established connections is $n = 1$. The weight of the established connection (Calculated Weight) is:

$$Cw_1 = \sum_{i=1}^{n=1}(W) = W(c_1) = 0.7395kg \quad \ldots \quad (7)$$

In the first assembly state, the necessary weight is $Ow_1 = 0.7395kg$. Therefore, The difference of weight Dw is as follows;

$$Dw = Ow_1 - Cw_1 = 0.7395 - 0.7395 = 0kg \quad \ldots \quad (8)$$

If Dw is multiplied by the unit weight value $(Uwv = 100)$, the weight cost of $[100000000]$ will be calculated as follows;

$$Wc = Dw.Uwv = 0.100 = 0 \quad \ldots \quad (9)$$

As a result, the weight cost of $[100000000]$ is "0". Table 4 shows weight costs of some examples of feasible assembly sequences for the coupling system.

Table 4 The Wc of some example feasible sequences for coupling system (AS-assembly States)

No	*AS there is no established connections*	*AS established one connection*	W_c	AS established two connections	W_c	*AS established three connections*	W_c	*AS established all connections*	*total* W_c *costs*
1	$[000000000]$	$[010000000]$	12	$[011000000]$	75	$[011000100]$	98	$[111111111]$	185
2	$[000000000]$	$[010000000]$	12	$[011000000]$	75	$[011100000]$	96	$[111111111]$	297
3	$[000000000]$	$[010000000]$	12	$[011000000]$	75	$[011010000]$	75	$[111111111]$	162
4	$[000000000]$	$[010000000]$	12	$[010010000]$	14	$[010110000]$	35	$[111111111]$	61
5	$[000000000]$	$[010000000]$	12	$[010010000]$	14	$[011010000]$	75	$[111111111]$	101

As seen in Table 4, the optimum feasible assembly sequence is,

$[000000000][010000000][010010000][010110000][111111111]$

The total weight cost is "61". Some of the sample feasible assembly sequences and the changes of the weight costs related with these assembly sequences are shown in figure 4 (Case 1). From the figures, the proposed neural network exactly follows the desired results (feasible assembly sequences-FAS and optimum assembly sequence) of the assembly system.

The error convergence graph of the case 1 is depicted in Figure 5 during the training of the network. As can be seen from the figures, the error is suddenly reducing to small values. Small epochs can be employed for case 1 (51200 epoch).

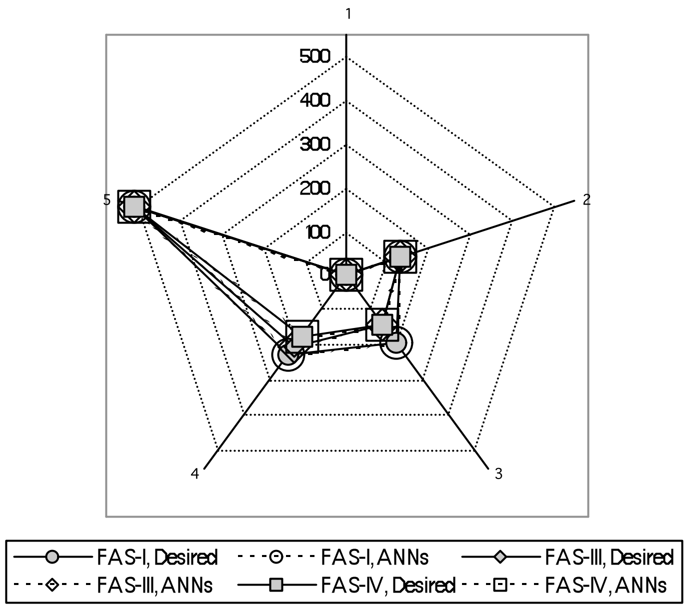

Figure 4 The sample feasible assembly sequences for the coupling assembly system according to weight criterion

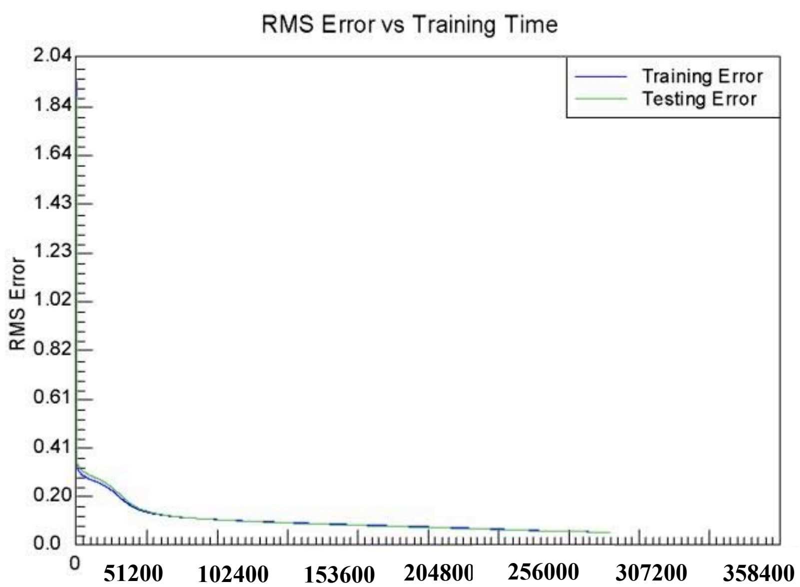

Figure 5 The error convergence graph of the case 1

6.2. Optimization of Subassembly Degree of Freedom Criterion

The subassembly degree of freedom criterion is based on the selection of parts with low degrees of freedom. So degree of freedom between the subassembly parts is low, the assembly of these parts can be done more easily. It is a unit cost (unit degree of freedom value, $(Udof)$ also used for this criterion. It is "25" and this criterion is more important than the other. Therefore, it is selected as the lower unit cost according to weight criterion, and so that total cost of assembly sequences can be reduced. It determines degree of interference for pairs of parts connections established along the six main directions of the Cartesian coordinate system. The total degree of freedom $(Tdof)$ for pairs of parts is the product's unit cost. Therefore, in the directed graph costs of degree of freedom according to this criterion are calculated as the degree of freedom for each path from initial node to terminal node. As a result, the minimum cost of the assembly sequence can be selected as an optimum with respect to the degree of freedom criterion.

For instance, the calculation of the degree of freedom costs of two different assembly states in the feasible assembly sequence for the coupling system is given as follows;

- ***Assembly State of*** $[100000000]$

In this assembly state, connection c_1 between part $\{a\}$ and part $\{b\}$ has been established. The degree of freedom of this pair of parts is shown in Table 5.

Table 5 Degree of freedom between parts $\{a\}$ and $\{b\}$

$c_1:(a,b)$	x	y	z	$-x$	$-y$	$-z$
a/b	1	0	0	0	0	0
b/a	0	0	0	1	0	0

The total degree of freedom for $[100000000]$ is $Tdof = 2$. If this value is multiplied by the $Udof = 25$ unit freedom cost, the result will be "50". Therefore, the degree of freedom cost $DOFc$ for $[100000000]$ assembly state is "50". Table 6 shows the degree of freedom costs for some example feasible assembly sequences for the coupling system.

Table 6 The $DOFc$ of some example feasible sequences for coupling system (AS-Assembly States)

No	*AS there is no established connections*	*AS established one connection*	DOF_c	*AS* established two *connections*	DOF_c	*AS established three connections*	DOF_c	*AS established all connections*	*total* DOF_c *costs*
1	$[000000000]$	$[001000000]$	250	$[011000000]$	300	$[011010000]$	350	$[111111111]$	900
2	$[000000000]$	$[001000000]$	250	$[001010000]$	300	$[011010000]$	350	$[111111111]$	900
3	$[000000000]$	$[010000000]$	50	$[010010000]$	100	$[010110000]$	150	$[111111111]$	300
4	$[000000000]$	$[010000000]$	50	$[010010000]$	100	$[011010000]$	350	$[111111111]$	500

$[000000000]$ and $[111111111]$ are the same for all assembly sequences. Therefore, these states can be neglected when calculating the total cost of feasible assembly sequences. Figure 6 (Case 2) shows some of the feasible assembly sequences in table 6. The results in the ANNs and test data targets are matched.

The optimum weight cost is "61" for the coupling system. The optimum assembly sequence has a cost of "300" according to the subassembly degree of freedom criterion. These assembly sequences are as follows:

$[000000000][010000000][010010000][010110000][111111111][000000000][010000000][010010000][010010100][111111111]$
$[000000000][001000000][000110000][010110000][111111111][000000000][000010000][010010000][010110000][111111111]$
$[000000000][000010000][010010000][010010100][111111111][000000000][000010000][010010000][010010010][111111111]$
$[000000000][000010000][000110000][010110000][111111111][000000000][000010000][000010100][010010100][111111111]$
$[000000000][000000100][000010100][010010100][111111111][000000000][000000010][010000010][010010010][111111111]$
$[000000000][000000010][010000010][010000011][111111111]$

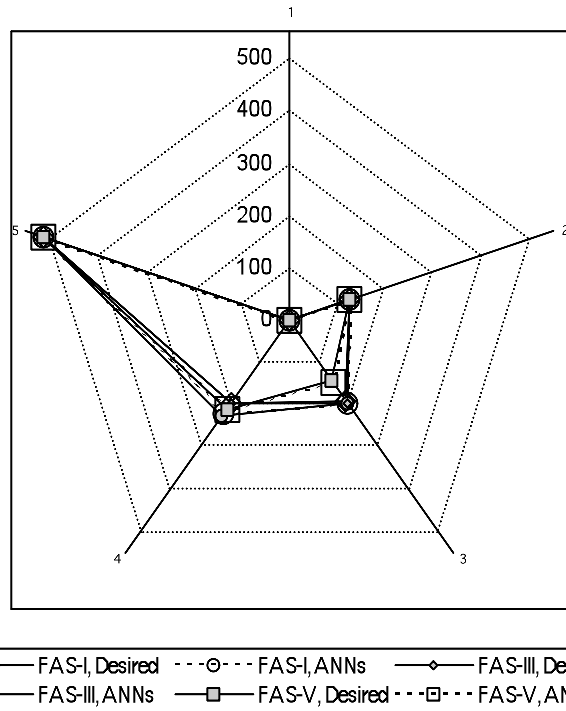

Figure 6 The sample feasible assembly sequences for the coupling assembly system according to subassembly degree of freedom criterion

In the optimization approach, both optimization criteria are minimal and so that assembly sequence is optimum. The optimum assembly sequence for coupling system is $(Wc = 61, DOFc = 300)$

$[000000000][010000000][010010000][010110000][11111111]$

Set representation for this sequence is as follows:

$[000000000] \Rightarrow \{\{a\},\{b\},\{c\},\{d\},\{e\},\{f\},\{g\}\}, [010000000] \Rightarrow \{\{a,c\},\{b\},\{d\},\{e\},\{f\},\{g\}\}, [010010000] = [110010000] \Rightarrow \{\{a,c\},\{b,e\},\{d\},\{f\}\} = \{\{a,b,c,e\},\{d\},\{f\},\{g\}\}$
$[010110000] = [110110100] \Rightarrow \{\{a,b,c,e,g\},\{d\},\{f\}\}, [11111111] = \{\{a,b,c,d,e,f,g\}\}$

In this representation, $[010010000]$ is the same as $[110010000]$. Because, in the $[010010000]$, $c_2(\{a,c\})$ and $c_5(\{b,e\})$ connections have been established. So, a cartesian product should be applied to elements of sets to determine $(a,c)(b,e) \Rightarrow (a,b),(a,e),(c,b),(c,e)$. $\{a,e\}$, $\{c,b\}$ and $\{c,e\}$ sets not corresponding to a connection (Figure 2). But, the set of $\{a,b\}$ corresponds to c_1 connection. Therefore, both vectors have the same meaning as far as sets are concerned. To determine optimum assembly sequence, $[010010000]$ vector is considered. Because, the weight cost of this vector is lower than the other. In this optimum assembly sequence for the coupling assembly, the Flange-I is joined to the shaft-I in the first assembly state, the Flange-II and the shaft-II are added in second assembly state. In the third assembly state, the bolt is used to fix this subassembly, and then the washer and nut is joined in the last subassembly. The total cost of feasible assembly sequence for any product is expressed as a cost function fc;

$$Wc = [(Ow_1 + Ow_2 + ... + Ow_k) - (Cw_1 + Cw_2 + ... + Cw_k)] x Uwv = \left[\left(\sum_{i=1}^{l} Ow_k\right) - \left(\sum_{i=1}^{l} Cw_k\right)\right] \times Uwv \quad ... \quad (10)$$

$$DOFc = Tdof\ xUdof \quad ... \quad (11)$$

$$fc = Wc + DOFc \quad ... \quad (12)$$

(l) is the number of assembly states in the feasible assembly sequence for any product.

7. CONCLUSIONS

In this paper, an assembly sequence planning system, based on binary vector representations, is developed. The neural network approach has been employed for analyzing optimum assembly sequence for assembly systems. The input to the assembly system is the assembly's connection graph that represents parts and relations between these parts. The output to the system is the optimum assembly sequence. In the constitution of assembly's connection graph, a different approach employing contact matrices and Boolean operators has been used. Moreover, the neural network approach is used in the determination of optimum assembly sequence. The inputs to the networks are the collection of assembly sequence data. This data is used to train the network using the Back propagation (BP) algorithm.

The proposed neural network model outperforms the available assembly sequence-planning model in predicting the optimum assembly sequence for mechanical parts. Due to the parallel structure and fast learning of neural network, this kind of algorithm will be utilized to model another types of assembly systems. In the proposed neural approach, the Back propagation algorithm is used. Various training algorithms can be employed.

The simulation results suggest that the neural predictor would be used as a predictor for possible practical applications on modeling assembly sequence planning system. This paper discusses a new modeling scheme known as artificial neural networks. The neural network approach has been employed for analyzing feasible assembly sequences and optimum assembly sequence for assembly systems.

8. REFERENCES

1. Boothroyd, G. (1994). Product Design for Manufacture and Assembly. Computer Aided Design, 26(7): 505-509.
2. Canbulut, F., Sinanoğlu, C. and Yıldırım, Ş. (2004). Analysis of Effects of Sizes of Orifice and Pockets on the Rigidity of Hydrostatic Bearing Using Neural Network Predictor System. KSME International Journal, 18(3): 432-442.
3. Garrod, W. and Everett, L.J. (1990). A.S.A.P.: Automated Sequential Assembly Planner. Computers in Engineering ASME, 1: 139-150.
4. Laperriere, L. and El Marghy, A. (1994). Assembly Sequence Planning for Simultaneous Engineering. International Journal Advance Manufacturing Technology, 9: 231-244.
5. Nanthavanij, S. and Yenradee, P. (2001). Selecting Appropriate Degree of Automation of Component Insertion Method for Printed Circuit Board (PCB) Assembly. International Journal of Industrial Engineering-Theory Applications and Practice, 8(1): 16-27.
6. Seow, K.T. and Devanathon, R. (1993). Temporal Logic Programming for Assembly Sequence Planning. Artificial Intelligence in Engineering, 8: 253-263.
7. Simaria, AS. And Vilarinho, PM. (2001). The Simple Assembly Line Balancing Problem With Parallel Workstations-A Simulated Annealing Approach. International Journal of Industrial Engineering-Theory Applications and Practice, 8(3): 230-240.
8. Sinanoğlu, C. and Börklü, H.R. (2004). An Approach to Determine Geometric Feasibility to Assembly States by Intersection Matrices in Assembly Sequence Planning. Journal of Intelligent Manufacturing, 15: 543-559.
9. Yee, S.T. and Ventura, JA. (1999). A Dynamic Algorithm to Determine Optimal Assembly Sequences Using Petri Nets. International Journal of Industrial Engineering-Theory Applications and Practice, 6(1): 27-37.

BIOGRAPHICAL SKETCH

Dr. Sinanoğlu received his Dip. Eng. Degree from Erciyes University, Engineering Faculty Department of Mechanical Engineering. He received his MSc Degree in Assembly Modelling from Erciyes University, The Institute of Natural Sciences and Technology in 1996 and PhD Degree in Assembly Sequence Planning from the same Institute in 2001. His research interests include; Assembly Modelling, Assembly Planning, Artificial Neural Networks and Tribology. He has authored or co-authored over 40 refereed journal and conference proceeding papers in above areas.