Contents lists available at GrowingScience

International Journal of Industrial Engineering Computations

homepage: www.GrowingScience.com/ijiec

Hybridized genetic-immune based strategy to obtain optimal feasible assembly sequences

Bala Murali Gunji[a], B. B. V. L. Deepak[a*], M V A Raju Bahubalendruni[b] and Bibhuti Bhusan Biswal[b]

[a]Department of Industrial Design, National Institute of Technology- Rourkela,769008, India
[b]Department of mechanical engineering, GMR Institute of Technology –Andrapradesh, 532127, India

CHRONICLE

Article history:
Received October 2 2016
Received in Revised Format
October 28 2016
Accepted December 2 2016
Available online
December 2 2016

Keywords:
Assembly sequence planning
Artificial immune system
Genetic algorithm
Assembly automation
Feasible assembly sequence
Assembly automation

ABSTRACT

An appropriate sequence of assembly operations increases the productivity and enhances product quality there by decrease the overall cost and manufacturing lead time. Achieving such assembly sequence is a complex combinatorial optimization problem with huge search space and multiple assembly qualifying criteria. The purpose of the current research work is to develop an intelligent strategy to obtain an optimal assembly sequence subjected to the assembly predicates. This paper presents a novel hybrid artificial intelligent technique, which executes Artificial Immune System (AIS) in combination with the Genetic Algorithm (GA) to find out an optimal feasible assembly sequence from the possible assembly sequence. Two immune models are introduced in the current research work: (1) Bone marrow model for generating possible assembly sequence and reduce the system redundancy and (2) Negative selection model for obtaining feasible assembly sequence. Later, these two models are integrated with GA in order to obtain an optimal assembly sequence. The proposed AIS-GA algorithm aims at enhancing the performance of AIS by incorporating GA as a local search strategy to achieve global optimum solution for assemblies with large number of parts. The proposed algorithm is implemented on a mechanical assembly composed of eleven parts joined by several connectors. The method is found to be successful in achieving global optimum solution with less computational time compared to traditional artificial intelligent techniques.

1. Introduction

Industrial robots are widely used in manufacturing applications for material handling and assembly operations. Typical assembly process consumes major stake of overall manufacturing cost, an appropriate assembly sequence can reduce the assembly time and effort and thus offers high productivity. Achieving an optimal assembly sequence is a major problem faced by the manufacturers as there exists millions of possible combinations. Many traditional methods have been proposed by the researchers for optimization of assembly sequence with large search space and high computational time (Linn & Liu, 1999). Computer

* Corresponding author
E-mail: bbv@nitrkl.ac.in (B.B.V.L. Deepak)

doi: 10.5267/j.ijiec.2016.12.004

aided design (CAD) integrated knowledge based methods have been implemented to extract the predicate criteria such as liaison data (Bahubalendruni & Biswal, 2014a, 2014b; Bahubalendruni & Biswal, 2015a) mechanical feasibility (Bahubalendruni et al*.,* 2015b, 2015c), stability (Bahubalendruni & Biswal, 2015d), and geometrical feasibility (Bahubalendruni et al*.,* 2016 & Bahubalendruni et al*.,* 2016) to solve optimum assembly sequence planning.

Knowledge based methods solve Assembly Sequence Planning (ASP) accurately but these methods take more computational time due to their huge search space utilization (Dong et al*.,* 2007). To overcome this problem researchers implemented several Artificial Intelligent (AI) techniques to achieve optimum assembly sequence. In most AI techniques, the input supplied to the process is generated manually which is again time consuming and needs accuracy without error. Due to human intervention while supplying inputs, these techniques have the drawback of high redundancy and premature convergence during the optimization of assembly sequence. To validate the quality of an assembly sequence, it must be tested for assembly predicates. Besides liaison and geometrical feasibility predicates, stability and mechanical feasibility are two more essential assembly predicates to yield the appropriate results (Zha et al*.,* 1998). Limitations of AI based optimal ASP methods in terms of assembly predicate consideration, premature convergence and high computational time are listed in Table 1.

Table 1
Assembly predicate considerations with AI in the cited research literature

Reference	Algorithm	Objective function (minimization of)	Assembly predicate criteria	Premature convergence/ Local Optima	Computational time
(Hsu et al., 2011)	KBM	Computational time	C	C	NC
(Kashkoush & ElMaraghy, 2015)		Computational time	C	C	NC
(Chen & Liu, 2001)	GA	Computational time	C	C	NC
(Chen et al., 2002)		Computational time	C	C	NC
(Sinanoglu & Riza Börklü, 2005)	NN	Computational time	C	C	NC
(Bahubalendruni et al., 2015)	PSO	Computational time	C	C	NC
(Wang & Liu, 2010)		Computational time	C	C	C
(Nayak et al., 2015)	SA	Computational time	C	C	NC
(Lee & Gemmill, 2001)		Assembly cost	C	C	NC
(Bahubalendruni et al., 2016)	AIS	Computational time	C	C	NC
(Biswal et al., 2103)		Assembly cost	C	C	NC
(Chang et al., 2009)		Computational time	C	C	C
(AkpıNar et al., 2013)	HA	Computational time	C	C	NC
(Xing & Wang, 2012)		Computational time	C	C	C
(Shan et al., 2009)		Computational time	C	C	C

PSO-Particle Swarm Optimization, NN-Neural Network, SA-Simulating Annealing, KBM-Knowledge Based Method, GA-Genetic Algorithm, HA- Hybrid Algorithm.
Note: C- considered, NC- Not considered.

Even though AI with Computer Aided Design (CAD) integrated algorithms have been developed, but these have failed to achieve the global optimum solution for ASP. To solve many of the engineering problems, GA and AIS have been implemented successfully. The current research work is focused on the hybridization of these algorithms to achieve the global optimization solution of ASP. In this paper, a hybrid AIS-GA based algorithm along with CAD integration for extraction of liaison data, stability feasibility, mechanical feasibility and geometrical feasibility is explained to obtain an Optimal Feasible Assembly Sequence (OFAS) with less computational time.

2. Assembly Information Extraction

Generally, performing the required input data for generating the algorithm to optimize the assembly manually is a time consuming process. This section details about the methods to extract liaison matrix, interference matrices, stability matrix and mechanical feasibility matrix from CAD environment. Computer Aided Three-dimensional Interactive Application Version-5 (CATIA V5R17) is used as CAD tool and programming is done in Visual Basic (VB scripting) to extract the outcomes. A transmission assembly consists of eleven parts is considered in this investigation is as shown in the Fig.1. The six possible directions of part movements during assembly process can be represented as: x+ → 1, x- → 2, y+ → 3, y- → 4, z+ → 5 and z- → 6.

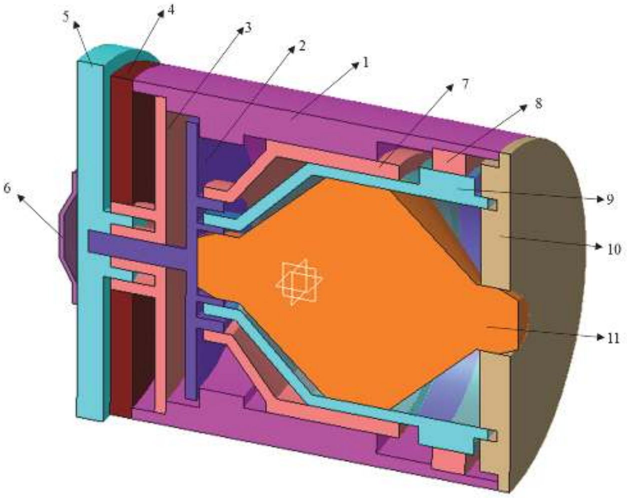

Fig. 1. Cut section view of transmission assembly (De Fazio and Whitney, 1987)

2.1. Liaison Matrix Extraction

Liaison data gives the information about all possible mating surfaces of an assembly. Two weights 1 &0 are assigned for mating and not mating parts respectively. Since in the current investigation an eleven-part assembly structure has been considered, a matrix with 11-by-11 is extracted from the CAD tool: CATIA-V5R17. The obtained liaison matrix for the considered eleven-part mechanical assembly structure is as follows:

$$\text{Liaisoan matrix} = \begin{array}{c|ccccccccccc|} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\ \hline 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 2 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 3 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 6 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 7 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 8 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 9 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 10 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 11 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{array}$$

2.2. Interference Matrix Extraction

The interference matrices give the information about all possible interferences of the components during the assembly. Two weights 0 &1 are assigned for the interference and not interference respectively. Part movements during the assembly process are of six directions, six interference matrices corresponding to each direction will be obtained. Since in the current investigation an eleven-part assembly structure has been considered, a matrix with 11-by-11 is extracted from the CAD tool.

The obtained interference matrices for the considered eleven-part mechanical assembly structure in X+ and X- directions are as follows:

$$\begin{array}{c|ccccccccccc}
X+ & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\
\hline
1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
3 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
4 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
5 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
6 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
7 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
8 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
9 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
11 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0
\end{array}
\qquad
\begin{array}{c|ccccccccccc}
X- & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\
\hline
1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
3 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
4 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
5 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
6 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
7 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
8 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
9 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
11 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0
\end{array}$$

The obtained interference matrices for the considered eleven-part mechanical assembly structure in Y+ and Y- directions are as follows:

$$\begin{array}{c|ccccccccccc}
Y+ & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\
\hline
1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
2 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
3 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
4 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
6 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
7 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
8 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\
9 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\
10 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
11 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0
\end{array}
\qquad
\begin{array}{c|ccccccccccc}
Y- & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
3 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
4 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
5 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
6 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
9 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
10 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\
11 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}$$

The obtained interference matrices for the considered eleven-part mechanical assembly structure in Z+ and Z- directions are as follows:

$$\begin{array}{c|ccccccccccc}
Z+ & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\
\hline
1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
3 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
4 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
5 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
6 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
7 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
8 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
9 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
11 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0
\end{array}
\qquad
\begin{array}{c|ccccccccccc}
Z- & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\
\hline
1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
3 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
4 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
5 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
6 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
7 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
8 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
9 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
11 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0
\end{array}$$

2.3 Stability Matrix Extraction

Stability matrix gives the information about the stability of the parts during assembly whether the parts are incomplete stable, partial stable or complete stable. In this study three weights, 0, 1 & 2 have been allotted for incomplete stable, partial stable and complete stable respectively. Since in the current investigation an eleven-part assembly structure has been considered, a matrix with 11-by-11is extracted from the CAD tool-CATIA (V5R17).

Combined stability matrix:

$$\begin{array}{c|ccccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\ \hline 1 & 0 & 0 & 2 & 2 & 0 & 0 & 1 & 0 & 0 & 1 & 2 \\ 2 & 0 & 0 & 1 & 0 & 2 & 0 & 1 & 1 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 0 & 2 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 6 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 7 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 8 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 9 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 10 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 11 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{array}$$

2.4. Mechanical Feasibility Extraction

Mechanical feasibility matrix gives the information about, whether two parts can be joined by a physical connector in the presence of already existed part without any interference. The order of the matrix is represented by n-by-n-by-n. The third dimension represents, whether the part represented in it offers any interference to place hard connections between parts represented in first two dimensions. The mechanical feasibility matrix for 11-part assembly is shown in the Fig. 2.

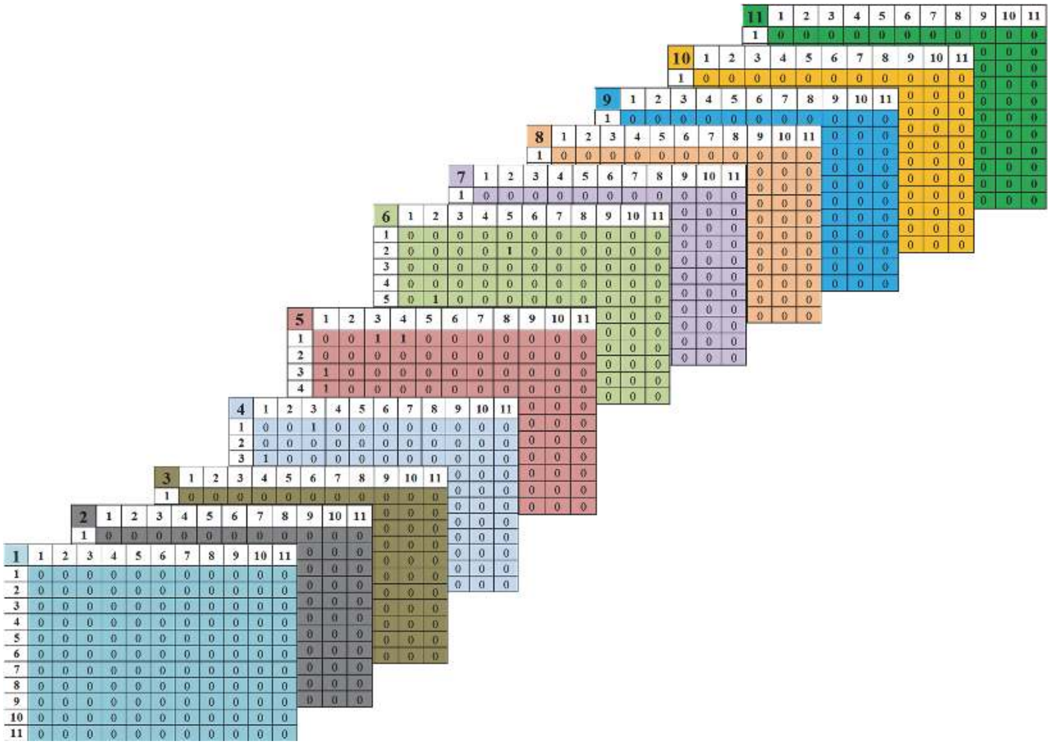

Fig. 2. Mechanical feasibility matrix for 11-part assembly

In the Fig. 2. it is clearly shown that parts (1&3) cannot be joint with a physical connector in the presence of part 4. Similarly, parts (1&3) and (1&4) can’t be joint with physical connectors in the presence of part 5 and also parts (2&5) can’t be joint with physical connector in the presence of part 6.

3. Fitness Function Evaluation

Once an assembly sequence qualifies liaison, stability and geometrical feasibility, the sequence is treated to be a valid sequence which will be further sent for fitness value calculation. Therefore a fitness function (m) is to be developed which should satisfy the following three criteria in the assembly tasks.

i). Assembly operations should associate with less number of end-effectors/tool changes (n_{tc}). An end effector is used to hold an object to perform assembly task. Since an assembled product consists of various mechanical parts there is a requirement of using several end effectors to hold the specific object. It is required to minimize the number of changes of end effector during assembly process because changing an end effector requires lot of time.

$$\Rightarrow m_1 = w_{tc} * n_{tc} \tag{1}$$

Where (w_t) is the weighing parameter is corresponds to end-effectors/tool change

ii). Assembly operations should associate with less number of direction changes (n_{dc}).It is prefers to perform the assembly operation for all parts in a single direction. Changing the direction frequently leads to the time consuming.

$$\Rightarrow m_2 = w_{dc} * n_{dc} \tag{2}$$

Where$(w_{dc)}$is the weighing parameter is corresponds to directional changes

iii). Assembly operations should associates with less number of part movements (n_{pm}). During assembly operation parts to be aligned in correct position for quicker assembly because parts are hold by the gripper, which leads to time and energy consumption.

$$\Rightarrow m_3 = w_{pm} * n_{pm} \tag{3}$$

Where$(w_{pm)}$is the weighing parameter is corresponds to part movements

By combining the Eq. (1), Eq. (2) and Eq. (3) the final form of the fitness will be as follows

$$m = m_{1+}m_2 + m_3$$

$$m = w_{tc} * n_{tc} + w_{dc} * n_{dc} + w_{pm} * n_{pm} \tag{4}$$

By observing the equation (4), the fitness value of an optimal assembly sequence must be low. In the meanwhile, the assembly sequence should satisfy the basic predicate tests. Therefore, it is required to apply an AI based optimization technique in order to minimize the total manufacturing time and cost

The proposed method is implemented for various combinations of weights aiming minimum gripper change, minimum assembly direction and minimum part movements. However, the selection of weights is significantly dependent on the industrial engineer’s decision based on the assembly tool facilities and time to change grippers.

4. Artificial Intelligent Techniques in Assembly Sequence

Several AI techniques are presented to achieve the optimized assembly sequence. But these techniques which are using are not sufficient to get the optimum feasible sequence because of huge search time and manual data extraction. Instead of manual data extraction, CAD platform has been used here while subjected to four basic criteria. This research work has been carried out by combining the AIS-GA algorithms with integration of CAD to achieve global optimal solution while reducing the system redundancy and computational time.

4.4. Artificial Immune System

The interest in studying the immune system is increasing over the last few years. Computer scientists, engineers, mathematicians, philosophers and other researchers are particularly interested in the capabilities of this system, whose complexity is comparable to that of the brain (Deepak & Parhi 2016). Many properties of the immune system are of great interest for computer scientists and engineers:

- *Uniqueness*: each individual possesses its own immune system.
- *Recognition of Foreigners*: the antigens are recognized and eliminated from body.
- *Anomaly Detection:* the immune system can detect and react to pathogens that the body has never encountered before.
- *Distributed Detection:* the cells are distributed all over the body and are not to subject any centralized control.
- *Noise Tolerance:* the system is flexible since the recognition of the antigens is not required.
- *Reinforcement Learning and Memory:* future responses to the same pathogens are faster and stronger since the immune system can "learn" the structures of pathogens.

Because of its special features, several models have been introduced to solve various engineering problems (Deepak & Parhi 2013). The following immune inspired algorithms are used in this investigation:

- *Bone Marrow Models:* For generating possible assembly sequences.
- *Negative Selection Model:* To reduce the system redundancy and predicate check.

4.4.1. Bone marrow model for Assembly Sequence Generation

In this model two gene libraries are considered with five-bit memory in order to generate an antibody (assembly sequence). First gene library i.e. library-1 corresponds to the part allocation and second gene library i.e. library-2 corresponds to the memory allocation of the part. The population selection is done randomly by the AIS algorithm shown in the Fig. 3. Once an allocation is filled in the either of the library, that specific part/memory allocation will not be repeated during the current assembly sequence generation. For example, if part A is selected from library-1 and memory allocation-1 from library-2, then part A will be in allocation-1 and these two allocations will not be repeated.

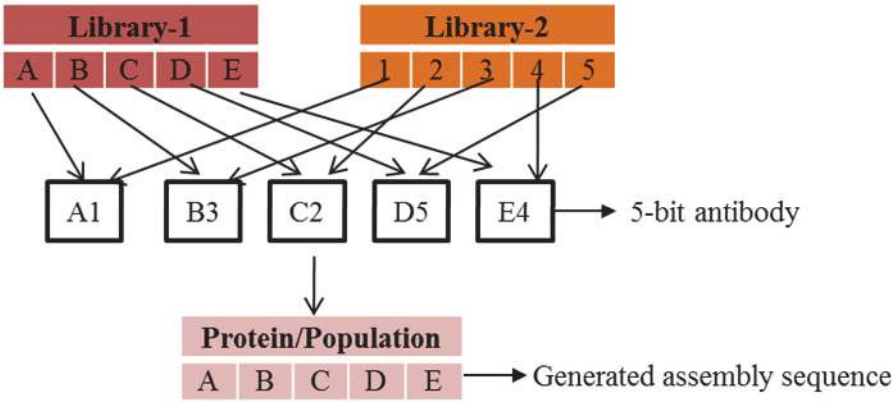

Fig. 3. Selection of population using AIS method

In order to avoid the system redundancy / generation of same antibody (assembly sequence) string current anti body string is subjected to reinforcement learning.

4.4.2. Negative selection algorithm

The negative selection algorithm is applied for the stored anti body string (assembly sequence) generated from the bone marrow model is tested with the predicated criteria. If this criterion is satisfying, then this model stores the current assembly sequence as a feasible assembly sequence which shown in the Fig.4. Later the stored feasible assembly sequence will be evaluated for its optimal criteria.

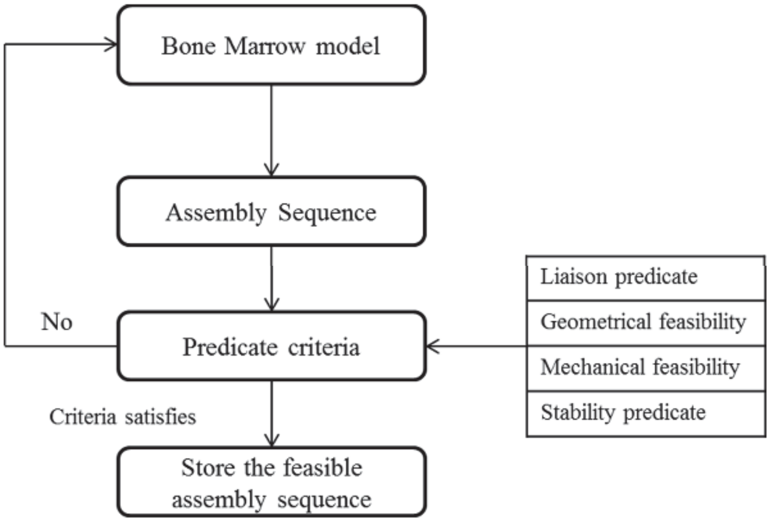

Fig. 4. Negative selection algorithm

4.5. Genetic Algorithm

Genetic algorithm is one of the artificial intelligent technique which uses the heretic search to minimize the process of natural selection. This is generally used for the search problems to generate the useful solution for optimization which is inspired by the inspired by natural evolution techniques, such as inheritance, mutation, selection, and crossover. In this algorithm population initialization is from negative selection algorithm. For the generated population fitness value is to be calculated as discussed section-3. For the evaluated population apply mutation and cross over genetic operators to generate the off-springs as shown in the Fig. 5 & 6.

Mutation: Mutation is swapping of two randomly chosen parts. A typical example is shown in the Fig. 4 for the mutation where C3 and D4 are chosen randomly in the parent, after swapping it generates the off spring.

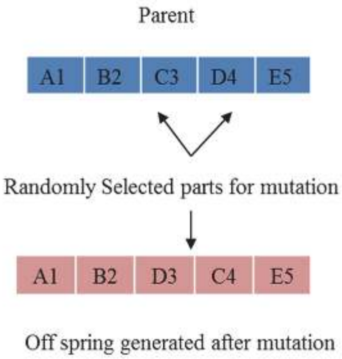
Parent
A1
B2
C3
D4
E5
Randomly Selected parts for mutation
A1
B2
D3
C4
E5
Off spring generated after mutation

Fig. 5. Mutation for off-spring generation

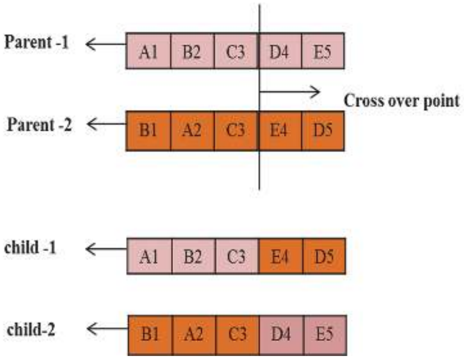

Fig. 6. Cross over for generating off-spring

Cross over: Cross over is selecting a common point of two parents so that two new off -springs will be generated. A typical example for crossover implementation is shown in Fig. 5, where a common point is selected for the parent-1& parent-2. After cross over two off-springs are generated having characteristics of both parents. Flow chart for the implementation of GA is explained in detail with the flow chart as shown in the Fig. 7.

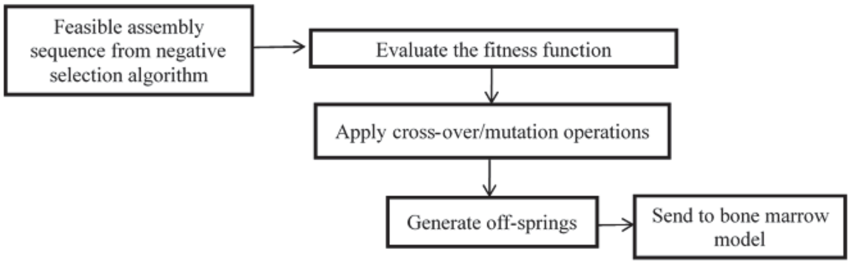
Feasible assembly sequence from negative selection algorithm
Evaluate the fitness function
Apply cross-over/mutation operations
Generate off-springs
Send to bone marrow model

Fig. 7. Flow chart of GA

4.6. Hybrid AIS-GA Architecture

The Hybrid AIS-GA based algorithm for generating the optimal feasible assembly sequence is explained in detail with the steps as below:

Step 1. Generate the population (assembly sequence) using bone marrow model.
Step 2. Check the predicate criteria (Liaison data, Stability, Geometric feasibility and Mechanical feasibility) for the generated assembly sequence through negative selection model.
Step 3. Send the feasible assembly sequence to GA model to evaluate its fitness value.
Step 4. Apply mutation and cross over genetic operators for producing the off-springs (new assembly sequences).
Step 5. Send the generated off-springs to the bone marrow model for its storage.
Step 6. Continue step 2 to step 5 until it reaches optimal criteria.

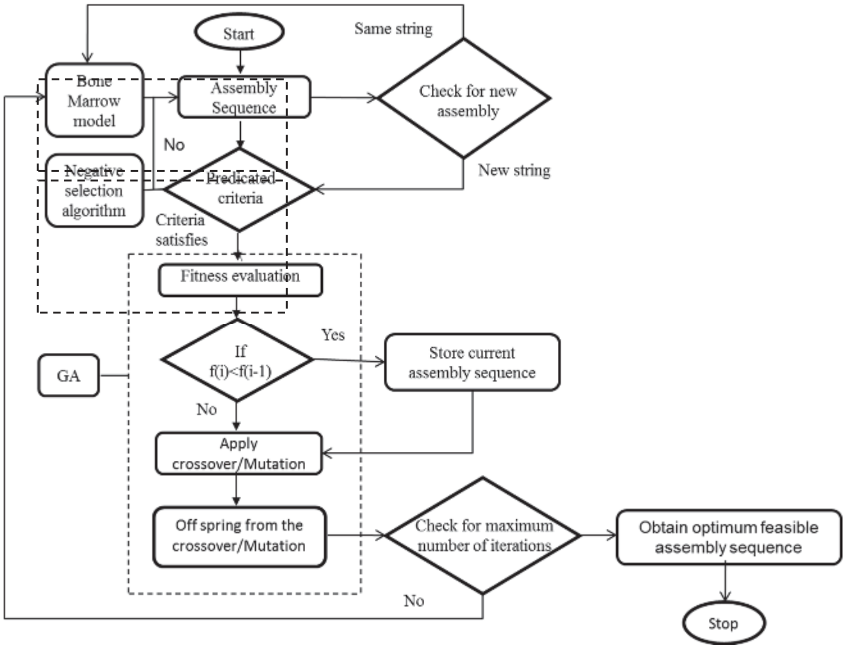

Fig. 8. Flow chart of the AIS-GA algorithm

5. Results & Discussion

The developed methodology for obtaining an optimum feasible assembly sequence has been implemented for a transmission assembly. This product consists of 11 parts (parts labelled with 1 to 11) as shown in Fig. 1 and the possible sequences are 11-factorial. Hybrid AIS-GA model which is discussed in section 4.3 produces the feasible assembly sequence planning. From the 11! Number of possible sequences, AIS-GA produced 1808 feasible sequences as listed in the supplementary material. For these feasible sequences optimization criteria to be applied with respect to end effector changes, directional changes and part movements.

The developed algorithm has been implemented to a standard transmission assembly as considered by Wang et al. (2005) and Smith (2004). Classical genetic algorithm has been implemented by Smith (2004) and ant colony optimization has been implemented by Wang et al. (2005) to produce OFAS while considering the directional changes as the objective function. From the past experience, it is found that the obtained OFAS is 11-9-8-7-10-1-2-3-4-5-6. However, the past strategies are dealt with only considering the directional changes during the assembly process. Whereas in the current research work, all criteria such as directional changes, gripper changes and energy of the parts are considered for obtaining the optimal feasible assembly sequence. More over the results are compared with the GA and AIS, which take less time for obtaining the optimal feasible assembly sequence as shown in the table 4 &table 5 respectively. Three weighing parameters corresponding to the end effectors (w_{tc}), directional changes (w_{dc}) and part movements (w_{pm}) are to be decided. In this investigation following two cases are considered for investing the weighing parameters

Case 1: Performing assembly operation by considering equal priority to tool changes, directional changes and part movements (i.e. $w_{tc} = w_{dc} = w_{pm} = 0.33$)

Case 2: Performing assembly operation by considering full priority to end-effectors/tool changes (i.e. $w_{tc} = 1, w_{dc} = 0, w_{pm} = 0$)

According to the nature of the assembly process the weights for each parameter can be decided by the user. In this paper, analysis has been performed for the above two cases of industrial settings.

Case 1: Performing assembly operation while considering equal priority to tool changes, directional changes and part movements (i.e. $w_{tc} = w_{dc} = w_{pm} = 0.33$)

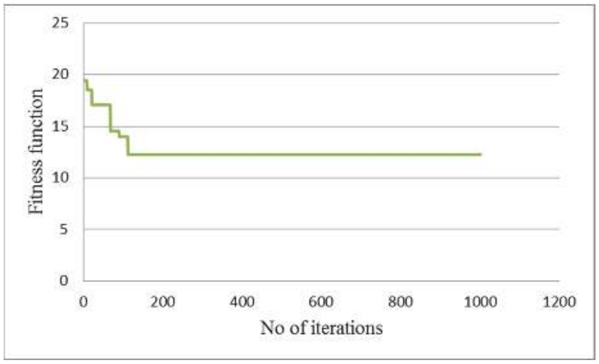

Fig. 9. Convergence of fitness over number of iterations

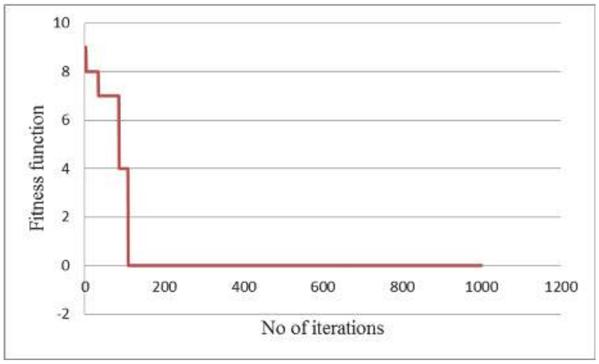

Fig. 10. Convergence of fitness over number of iterations

From the Fig. 9, it is noticed that the model is converged at 112th iteration with the fitness value 7.676 and the optimized assembly sequence is listed in the Table 2.

Table 2
Optimal feasible assembly sequence for Case-1

Iteration number	Assembly sequence	Assembly direction	Gripper changes	Fitness value
112	1-7-8-10-9-11-2-3-4-5-6	3-3-3-3-3-3-4-4-4-4-4	1-7-8-10-9-11-2-3-4-5-6	7.676

Case 2: Performing assembly operation while considering full priority to end-effectors/tool changes (i.e. $w_{tc} = 1, w_{dc} = 0, w_{pm} = 0$)

It is observed from Fig. 10 that the model is converged at 110th iteration having the fitness value ‘0’ and the optimized assembly sequence is listed in the table 3.

Table 3
Optimal feasible assembly sequences for Case-2

Iteration number	Assembly sequence	Assembly direction	Gripper changes	Fitness value
110	11-9-8-7-10-1-2-3-4-5-6	4-4-4-4-4-4-4-4-4-4-4	11-9-8-7-10-1-2-3-4-5-6	0

5.4. Comparison Among GA, AIS and AIS-GA

For the comparison purpose this analysis is focused only on assembly directional changes. Therefore the weights of fitness function are $w_{dc} = 1,\ w_{tc} = 0, w_{pm} = 0$. Here the analysis is carried out for GA, AIS and AIS-GA up to 1000 iterations. The fitness value of ‘1’ achieved by GA, AIS and AIS-GA at 414th, 479th and 101st iteration respectively. Individual GA and AIS algorithms couldn’t generate the optimal sequence with fitness value ‘0’ for 1000 iterations. Whereas, the hybrid AIS-GA model achieved the optimal feasible assembly sequence with ‘0’ fitness value at 110th iteration. A detail analysis results are illustrated in table 4 and Fig.11.

Table 4
Optimum assembly sequences for the assembly directional changes

Algorithm	No. of iterations	Time taken to generate OFAS (Sec)	Generated OFAS	Assembly direction of OFAS	No. of directional changes
GA	Up to 500	12.4	10-8-9-11-7-1-2-3-4-5-6	3-3-3-3-4-4-4-4-4-4-4	1
	Up to 1000	18.6	10-8-9-11-7-1-2-3-4-5-6	3-3-3-3-4-4-4-4-4-4-4	1
AIS	Up to 500	13.5	10-8-9-11-7-1-2-3-4-5-6	3-3-3-3-4-4-4-4-4-4-4	1
	Up to 1000	19.5	10-8-9-11-7-1-2-3-4-5-6	3-3-3-3-4-4-4-4-4-4-4	1
Hybrid (AIS-GA)	Up to 500	8.6	11-9-8-7-10-1-2-3-4-5-6	4-4-4-4-4-4-4-4-4-4-4	0
	Up to 1000	10.6	11-9-8-10-7-1-2-3-4-5-6	4-4-4-4-4-4-4-4-4-4-4	0

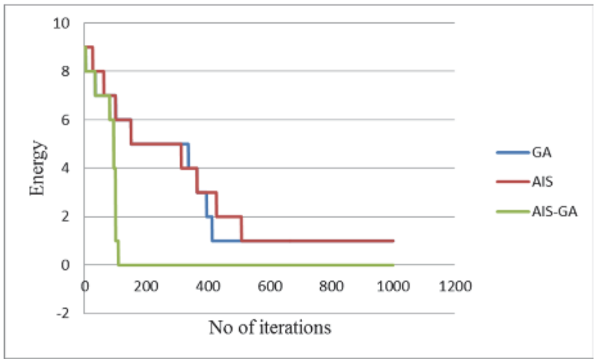

Fig. 11. A graph for developed AI techniques with directional changes with respect to number of iterations

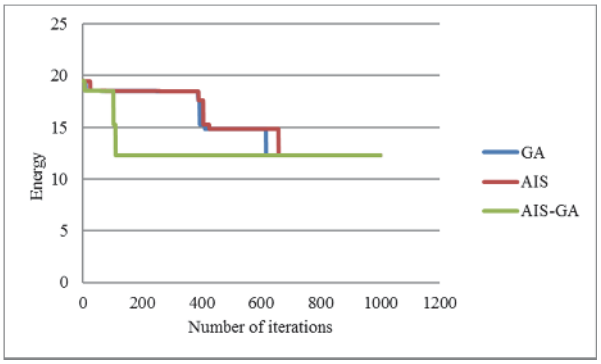

Fig. 12. A graph for different AI techniques with energy and Number of iterations

For the comparison purpose, this analysis is focused only on energy i.e. part movements. Therefore the weights of fitness function are $w_{dc} = 0,\ w_{tc} = 0, w_{pm} = 1$. Here the analysis is carried out for GA, AIS and AIS-GA up to 1000 iterations. The minimum fitness value of ‘12.262’ achieved by GA, AIS and AIS-GA at 616th, 658th and 110th iteration respectively. A detailed analysis results are illustrated in table 5 and Fig. 12.

Table 5
Optimum assembly sequences for the energy

Algorithm	No. of iterations	Time taken to generate OFAS (Sec)	Generated OFAS	Assembly direction of OFAS	Fitness value
GA	Up to 500	13.1	10-8-9-7-1-11-2-3-4-5-6	3-3-3-4-4-3-4-4-4-4-4	14.810
	Up to 1000	19.2	10-8-9-11-7-1-2-3-4-5-6	3-3-3-3-4-4-4-4-4-4-4	12.262
AIS	Up to 500	14.4	10-8-9-7-11-1-2-3-4-5-6	3-3-3-4-3-4-4-4-4-4-4	12.302
	Up to 1000	20.1	10-8-9-11-7-1-2-3-4-5-6	3-3-3-3-4-4-4-4-4-4-4	12.262
Hybrid(AIS-GA)	Up to 500	8.8	11-9-8-7-10-1-2-3-4-5-6	4-4-4-4-4-4-4-4-4-4-4	12.062
	Up to 1000	11.1	11-9-8-10-7-1-2-3-4-5-6	4-4-4-4-4-4-4-4-4-4-4	12.062

In the current study, two cases: $w_{tc} = w_{dc} = w_{pm} = 0.33$; $w_{tc} = 1, w_{dc} = 0, w_{pm} = 0$ are considered for understanding purpose. But the selection of these weights depends on the user/industrialist requirement.

6. Conclusion

Hybrid algorithm with CAD assistance has been introduced in this research work to obtain optimal feasible assembly sequence for a given product. In the present investigation, two separate algorithms namely GA and AIS are combined to generate OFAS. The developed methodology is found successful in succeeding all sets of valid assembly sequences for different configurations of assemblies in less computational time. Unlike the traditional optimization algorithms as mentioned in introduction section, the hybrid model does not generate redundant assembly sequence and thus saves lots of computational time and drives toward global optima. Moreover, the proposed hybrid algorithm has the flexibility for user defined weights: selection for gripper changes, directional changes and part movements unlike the existed artificial intelligent techniques. As a future work the developed hybrid algorithm can be applied effectively for the identification of sub-assembly to make more flexible for parallel assembly systems.

References

AkpıNar, S., Bayhan, G. M., & Baykasoglu, A. (2013). Hybridizing ant colony optimization via genetic algorithm for mixed-model assembly line balancing problem with sequence dependent setup times between tasks. *Applied Soft Computing, 13*(1), 574-589.

Bahubalendruni, M. V. A., & Biswal, B. B. (2014a). An algorithm to test feasibility predicate for robotic assemblies. *Trends in Mechanical Engineering & Technology, 4*(2), 11-16.

Bahubalendruni, M. R., & Biswal, B. B. (2014b). Computer aid for automatic liaisons extraction from cad based robotic assembly. *In Intelligent Systems and Control (ISCO), 2014 IEEE 8th International Conference on (pp. 42-45)*. IEEE.

Bahubalendruni, M. R., & Biswal, B. B. (2015a). A novel concatenation method for generating optimal robotic assembly sequences. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 0954406215623813.

Bahubalendruni, M. R., Biswal, B. B., Kumar, M., & Nayak, R. (2015b). Influence of assembly predicate consideration on optimal assembly sequence generation. *Assembly Automation, 35*(4), 309-316.

Bahubalendruni, M. R., & Biswal, B. B. (2015c). A review on assembly sequence generation and its automation. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 0954406215584633.

Bahubalendruni, M. R., & Biswal, B. B. (2015d). An intelligent method to test feasibility predicate for robotic assembly sequence generation. *In Intelligent Computing, Communication and Devices* (pp. 277-283). Springer India.

Bahubalendruni, M. R., & Biswal, B. B. (2016). Liaison concatenation–A method to obtain feasible assembly sequences from 3D-CAD product. *Sadhana, 41*(1), 67-74.

Bahubalendruni, M. R., Biswal, B. B., Kumar, M., & Deepak, B. B. V. L. (2016). A Note on Mechanical Feasibility Predicate for Robotic Assembly Sequence Generation. In *CAD/CAM, Robotics and Factories of the Future*(pp. 397-404). Springer India.

Bahubalendruni, M. R., Deepak, B. B. V. L., & Biswal, B. B. (2016). An advanced immune based strategy to obtain an optimal feasible assembly sequence. *Assembly Automation, 36*(2), 127-137.

Bahubalendruni, M. V. A., Biswal, B. B., & BB, V. (2015). Optimal Robotic Assembly Sequence generation using Particle Swarm Optimization. *Journal of Automation and Control Engineering, 4*(2), 89-95

Biswal, B. B., Deepak, B. B., & Rao, Y. (2013). Optimization of robotic assembly sequences using immune based technique. *Journal of Manufacturing Technology Management, 24*(3), 384-396.

Chang, C. C., Tseng, H. E., & Meng, L. P. (2009). Artificial immune systems for assembly sequence planning exploration. *Engineering Applications of Artificial Intelligence, 22*(8), 1218-1232.

Chen, S. F., & Liu, Y. J. (2001). An adaptive genetic assembly-sequence planner. *International Journal of Computer Integrated Manufacturing, 14*(5), 489-500.

Chen, R. S., Lu, K. Y., & Yu, S. C. (2002). A hybrid genetic algorithm approach on multi-objective of assembly planning problem. *Engineering Applications of Artificial Intelligence, 15*(5), 447-457.

De Fazio, T., & Whitney, D. (1987). Simplified generation of all mechanical assembly sequences. *IEEE Journal on Robotics and Automation*, *3*(6), 640-658.

Deepak, B. B. V. L., & Parhi, D. R. (2016). Control of an automated mobile manipulator using artificial immune system. *Journal of Experimental & Theoretical Artificial Intelligence, 28*(1-2), 417-439.

Deepak, B. B. V. L., & Parhi, D. (2013). Intelligent adaptive immune-based motion planner of a mobile robot in cluttered environment. *Intelligent Service Robotics, 6*(3), 155-162.

Dong, T., Tong, R., Zhang, L., & Dong, J. (2007). A knowledge-based approach to assembly sequence planning. *The International Journal of Advanced Manufacturing Technology, 32*(11-12), 1232-1244.

Hsu, Y. Y., Tai, P. H., Wang, M. W., & Chen, W. C. (2011). A knowledge-based engineering system for assembly sequence planning. *The International Journal of Advanced Manufacturing Technology, 55*(5-8), 763-782.

Kashkoush, M., & ElMaraghy, H. (2015). Knowledge-based model for constructing master assembly sequence. *Journal of Manufacturing Systems*, *34*, 43-52.

Lee, H. R., & Gemmill, D. D. (2001). Improved methods of assembly sequence determination for automatic assembly systems. *European Journal of Operational Research, 131*(3), 611-621.

Linn, R. J., & Liu, H. (1999). An automatic assembly liaison extraction method and assembly liaison model. *IIE transactions, 31*(4), 353-363.

Nayak, R., Bahubalendruni, M. R., Biswal, B. B., & Kumar, M. (2015, September). Comparison of liaison concatenation method with simulated annealing for assembly sequence generation problems. *In Next Generation Computing Technologies (NGCT), 2015 1st International Conference on (pp. 531-535). IEEE.*

Shan, H., Zhou, S., & Sun, Z. (2009). Research on assembly sequence planning based on genetic simulated annealing algorithm and ant colony optimization algorithm. *Assembly Automation, 29*(3), 249-256.

Sinanoglu, C., & Riza Börklü, H. (2005). An assembly sequence-planning system for mechanical parts using neural network. *Assembly Automation, 25*(1), 38-52.

Smith, S. S. F. (2004). Using multiple genetic operators to reduce premature convergence in genetic assembly planning. *Computers in Industry, 54*(1), 35-49.

Wang, Y., & Liu, J. H. (2010). Chaotic particle swarm optimization for assembly sequence planning. *Robotics and Computer-Integrated Manufacturing, 26*(2), 212-222.

Wang, J. F., Liu, J. H., & Zhong, Y. F. (2005). A novel ant colony algorithm for assembly sequence planning. *The International Journal of Advanced Manufacturing Technology, 25*(11-12), 1137-1143.

Xing, Y., & Wang, Y. (2012). Assembly sequence planning based on a hybrid particle swarm optimisation and genetic algorithm. *International Journal of Production Research, 50*(24), 7303-7312.

Zha, X. F., Lim, S. Y., & Fok, S. C. (1998). Integrated knowledge-based assembly sequence planning. *The International Journal of Advanced Manufacturing Technology, 14*(1), 50-64.