International Journal of Advance Research in Science and Engineering
Volume No. 10, Issue No. 11, November 2021
www.ijarse.com
IJARSE
ISSN 2319 - 8354

Assembly sequence planning using non-traditional Optimization methods (soft computingapproaches)

Shudhanshu Kumar

BIT SINDRI

Abstract:

Generating Assembly Sequence Planning describing the seriesof components in the course of muster is defined to upgrading/downgrading of the productin the course of its use by means of the customers. Performing muster in an optimized manner isresidences of the product and the constraints set by means of the muster system. Inthis paper highest quality muster series may be attained via implyingGenetic Algorithm. A genetic set of rules that generates and evaluates musterplans is proposed. This paper info thesoft computing approaches with a example i.e.,muster plan of Fixture assembly using genetic set of rules for producing highest quality assembly and also it info a brief review on research in assembly sequence plan(ASP) that employing soft computing approaches for past few years.The studies shows that three soft computingalgorithms that frequently used to solve ASP and ALB are Genetic Algorithm, Ant ColonyOptimisation and Particle Swarm Optimisation.

Key Words: muster planning, optimization problem, genetic set of rules, soft computing approaches.

1 INTRODUCTION:

Assembly making plans (or muster making plans) is to discover and collection the subassembly components of a muster collectively to shape the assembled product. Assembly collection (or an muster plan) specifies which units of components can shape subassemblies and the order wherein the components are becoming a member of collectively to shape subassembly. Assembly making plans is a combinatorial hassle. The quantity of muster plans is a factorial characteristic of the quantity of additives withinthe product. The choice of muster plans has a bearing impact at the very last product.

The desire of the muster collection wherein components or subassemblies are prepare within the mechanical muster of a product can substantially have an effect on the performance of the muster process. Hence an green muster plan, significantly determines lead-time, manufacturing cost, and, thus, capability product success. The muster-making plans hassle includes the identity, choice and sequencing of muster operations said as their results at the components. The identity of muster operations typically ends in the set of all viable muster plans.

In product development, current global market continuously gives pressure tomanufacturer to compete with competitors from all over the world. Manufacturer needs tospeed up the time to market and at the same time minimise the manufacturing cost toensure that their products remain competitive. Assembly is considered one of

the importantprocesses in manufacturing. It consumes up to 50% of total production time and accountfor more than 20% of total manufacturing cost.Research in assembly optimisation can help manufacturer to speed up assembly processand reduce assembly cost. Research in assembly optimisation can becategorised based on which product development and production phases is being studied(Fig. 1).In product conception and design stage, the aim of assembly optimisation is to reduce theassembly costs by applying design for assembly (DFA) approach in product design. Besidesreducing cost, DFA may also bring about additional benefits in terms of increased quality,reliability and shorter manufacturing time. The approach shortens the product cycle andensures a smoother transition from prototype to production.

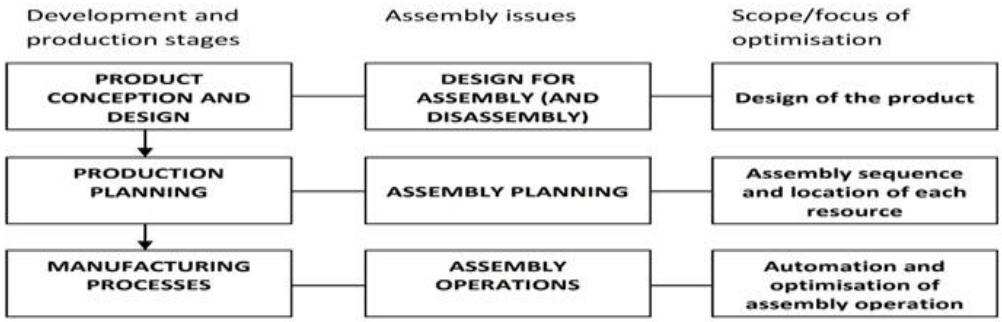


Fig. 1 Assembly related issues in different product development stages

These methods are mainly divided into two categories i.e. Traditional and Non Traditional. A brief introduction of these techniques is given below.

- **Traditional methods:**

These techniques are slow and guarantee of global convergence as long as problems are small. Traditional Techniques are also called as Optimization Techniques. They are Mathematical programming like Linear programming, Integer programming, Dynamic programming, Transportation etc.Enumerate Procedure Decomposition like Lagrangian Relaxation.

Non traditional methods:

These methods are very fast but they do not guarantee for optimal solutions. Non Traditional Techniques are also called as Approximation Methods. They involve

1. Constructive Methods like Priority dispatch rules, composite dispatching rules.
2. Insertion Algorithms like Bottleneck based heuristics, Shifting Bottleneck Procedure.
3. Evolutionary Programs like Genetic Algorithm, Particle Swarm Optimization.
4. Local Search Techniques like Ants Colony Optimization, Simulated Annealing, adaptive Search, Tabu Search, Problem Space methods

5. Iterative Methods like Artificial Intelligence Techniques, Artificial Neural Network, Heuristics Procedure, Beam-Search, and Hybrid Techniques.

2 Literature Review:

The muster of merchandise is the very last step withinside the production technique. For maximum merchandise, meeting is carried out manually. The labour value for meeting varies among 50% and 75% of the entire labour value for production the product. Optimization of the muster technique is consequently very important. One started to realise that the simplest manner to enhance the muster technique is to optimize the product layout for muster. The researcher DeFazio1987 provided And/OR Graph, for a whole illustration of all feasible muster plans through decomposition primarily based totally on cut-set approach. They dealt and confirmed how to plot restore sequences the usage of and/or graph illustration of muster plans. Two standards have been provided for the choice of muster plans. DeLit2001 defined maximizing the ability of sequencing the muster obligations and minimizing the muster time via parallel execution of the project which supplied a foundation to lessen the make span. They reviewed on representations of mechanical muster sequences. The trouble of producing the assembly sequences from the geometry of a intention muster become elaborated through DeFazio1987. The improvement of a machine for the automated popularity of assembly and disassembly capabilities for the era of sequences is presently in a studies phase. Srinivasan1999 statistics on assembly and disassembly capabilities and associated sources are retrieved from the CAD and Resource Database so as to plot operation collection. Tseng2011 provided a closed-loop muster collection making plans version through integrating assembly and disassembly collection making plans models.

3 Objectives Of assembly sequence planning:

The sequencing is made to meet specific objectives. The objectives are decided upon the situation, market demands, company's demands of rate of production. There are two types for the assembly sequencing

Objectives :

To minimise idle time and maximise utilisation in assembly has alsobeen utilized in ASP optimisation. The idle time in every workstation is described as thedistinction among processing time and allowed cycle time in assembly line . In the meantime, the assembly utilisation degree has been applied in type of ways.

4 Overview of Non-Traditional Algorithms:

An muster making plans primarily based totally on genetic set of rules become addressed via way of means of HomemdeMello1990, incorporating sure standards to evaluate the excellent of viable assembly sequences, like minimizing the orientation adjustments, the gripper adjustments etc. More recently, planners paintings mechanically from a CAD version and different non-geometric information. An most suitable muster plan is now sought, decided on from the set of all viable muster plans.

4.1 Ant Colony Algorithm:

Like genetic algorithms, ant algorithms draw their concept from biology; this time mimicking the manner ants forage for meals. It may be determined that ants are capable of locate the shortest course to a meals supply

through speaking with one another. They do that through depositing a pheromone path that may be detected through different ants that take the equal course. In the mid-1990 this concept become advanced into a technique that allowed huge seek areas to be explored. It become in the beginning implemented to the travelling salesman trouble and has latterly been implemented to many different trouble domains, consisting of automobile routing, quadratic venture and facility layout. Wang2005 proposed a unique ant colony set of rules for meeting series Planning. Zhang2008 introduce a summation rule to update the unique pheromone â€˜drop and evaporates€™ updating rule. In pheromone summation updating rule, the first-rate path is decided through summation of overall pheromone that dropped with out thinking about evaporation factor. Chandrasekharan2004 offered unique ant colony set of rules for float keep scheduling specifically max-min ant system(MMAS).

Any ant algorithm must specify the following elements:

I. construction of solutions,
II. heuristic information,
III. pheromone updating rule,
IV. selection probability,
V. local search algorithm, and
VI. stopping criterion.

4.2 Memetic Algorithm:

Genetic algorithms (GAs) are adaptive search and optimization algorithms that mimic the principles of natural genetics. GAs is very different from traditional search and optimization methods used in engineering design problems. Because of their simplicity, ease of operation, minimal requirements and global perspective, GAs has been successfully used in Memetic Algorithms. Both genetic algorithms and ant algorithms are examples of populace primarily based totally techniques in which some of answers are maintained and there's a few trade among the numerous individuals of that populace in an try to produce higher excellent people within the subsequent era. Other seek techniques (including hill climbing, simulated annealing and tabu seek) handiest preserve a unmarried answer at anyone time and try to enhance on that answer in small (normally) steps. The concept at the back of memetic algorithms is that a populace primarily based totally method seek mechanism is blended with a neighbourhood seek method in order that individuals of the populace may be taken to neighbourhood optima earlier than the subsequent era is produced. HongGuang2010 used a memetic set of rules to remedy meeting series making plans. Interference matrix is used to pick out the possible series and the meeting direction. In this paper guided neighbourhood seek technique employed.

4.3 Artificial Immune System:

Like a Genetic Algorithm (GA), Artificial Immune System (AIS) is likewise populace primarily based totally and the most beneficial answer is acquired through the evolution of the populace. In AIS, the issues to be solved are appeared as antigens. In general, three sorts of measurements are used to assess the antibodies, namely, health for the excellent, affinity for similarity among antibodies and awareness for populace variety. AIS is computational intelligence paradigm stimulated through the organic immune gadget, which has observed

software in sample reputation, scheduling, control, machine-learning, and statistics gadget security. To enforce the optimization thoughts addressed previously, immune operations along with immune choice, clonal choice and inoculation are introduced. AIS are found out through the subsequent steps: (1) reputation of antigens; (2) era of preliminary antibodies; (three) assessment of antibodies; (4) proliferation and suppression of antibodies; (5) era of latest antibodies; (6) development of antibodies. Steps three-6 can be iterated till convergence standards are satisfied.

4.4.1 Assembly sequence planning

ASP is one of important component in assembly planning. ASP refers to a task for whichplanners, on the basis of their particular heuristics in assembling all the components of aproduct, arrange a specific assembly sequence according to the product design description.ASP is an NP-hard combinatorial problem, where the solution space is excessivelyincreased when thenumber of component increased. Consider a product with sixcomponents that can be assembled in any sequences. In this case, the number of possiblesolution for this product is given by s = 6! which is equal to 720 solutions. When the number ofcomponent increased to seven, the possible solutions for the products excessively increasedto 7! = 5,040. Additionally, in a real assembly problem, there are some constraints that need tobe considered when generating assembly sequences.

4.4.2 ASP constraints

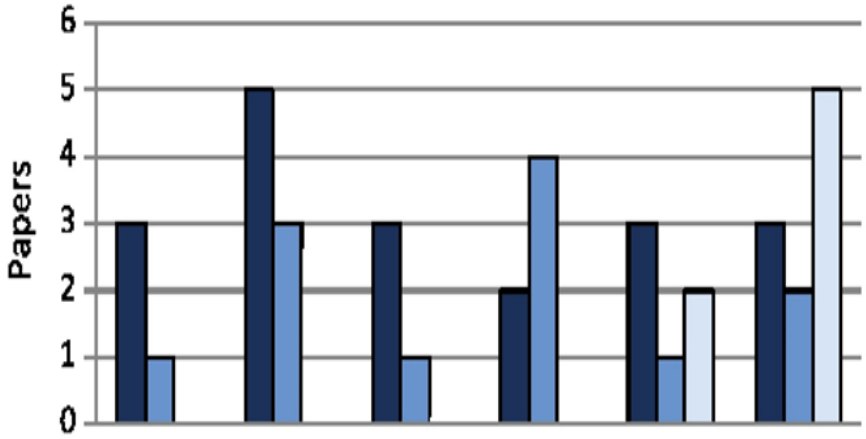
According to, there are two types of constraints in assembly, which are ‘absoluteconstraints’ and ‘optimisation constraints. The absolute constraints refer to constraints that, ifviolated, lead to infeasible assembly sequence. Meanwhile, the optimisation constraints arethe constraints that lead to lower quality of assembly sequences when violated.

4.4.3 Discussions and study potentials:

This paper studied research in ASP that used soft computing approaches for thepast 10 years. From this study, the previous research patterns and trends were identified.the trend of single- and multi-objective usage in ASP and optimisation for 2005 until 2015. The trend shows research papers that used single objectivewere fluctuated from 2005 to 2015. Meanwhile, similar trend was also found in number ofpapers that used multi-objective optimisation for the first 5 years. However, this trend waschanged for the second half of this period. The number of papers that used multi-objectiveoptimisation was started to grow from 2010 until 2015. The multi-objective optimisation wasmany researchers because of complexity of the problem and closer to the realassembly application.

In terms of optimisation algorithms usage, application of GA in ASPpapersbetween 2010 and 2015 is quite stable with an average of three papers per year. In terms of optimisation algorithms usage, application of GA in ASPpapersbetween 2010 and 2015 is quite stable with an average of three papers per year (Fig. 2).For the same period, the ACO usage in the cited research papers fluctuates. Meanwhile, thePSO algorithm was first implemented in ASP research in 2014. The number ofpapers that applied PSO algorithm had shown rapid

progress with two papers in 2014 andfive papers in 2015. In 2015, papers using PSO in ASP and optimisation was outnumbered.



(Figure no.02)

2010201120122013201420152015

(years)

GA

ACO

PSO

5 PROBLEM DESCRIPTION:

The paintings is targeted on deciding on an most beneficial or near-most beneficial meeting plan. The geometric constraints imposed through product are used to put together liaison graph. The genetic set of rules is used to expose a manner to generate a possible muster plans primarily based totally on priority matrix. The Assumptions concerned on this paintings contains of the subsequent: Exactly components or subassemblies are joined at every time, on every occasion components are joined forming a subassembly, all contacts among components in that subassembly are established, the feasibility of becoming a member of subassemblies is unbiased of the way the ones sub-assemblies had been built.

6 RESEARCH METHODOLOGY:

The meeting-making plans hassle for brand new product must be accomplished primarily based totally at the geometric constraints among components and subassemblies or a whole. Here the studies is cantered at the hassle of selecting the most beneficial or near-most beneficial meeting plan, i.e. one of the meeting bushes primarily based totally on minimization of make span. The meeting plans are evaluated primarily based totally

on make span, which incorporates operational complexity and stages of freedom. The priority constraints imposed through liaison graph are used to generate possible plans that constitute a fixed of preliminary populace. A genetic set of rules is designed to generate a possible meeting plans primarily based totally in this priority matrix. The priority family members are derived from the restrictions imposed on a aspect whilst placed withinthe very last meeting. A aspect may be both absolutely confined or partly confined, whilst the aspect is disassembled alongside as a minimum one mating direction. For a aspect this is absolutely confined, priority family members exist such that the aspect must be in vicinity earlier than a few different set of components. Next, pick out the variety of viable assembly, which represents muster assignment. With this, assemble a priority matrix; on the whole constitute a mechanism or schema for producing preliminary populace for the Genetic Algorithm. As a result, greater latest meeting making plans studies tries to apply synthetic intelligence techniques (in particular neural community and genetic algorithms) to broaden a technique which could take care of complicated systems and more than one constraints (with an inexpensive computation time),and accordingly convey muster making plans toward real-international software. Assembly planners that use genetic algorithms use a directed stochastic seek method to lessen the variety of answers taken into consideration. As a result, muster planners that use genetic algorithms can normally locate tremendous answers greater fast than muster planners that use graph looking strategies.

6.1 AND / OR Graph:

Each assignment can also additionally have numerous direct predecessors and won't start execution till all its predecessors are complete. This conventional version falls quick in describing many real-time programs encountered in practice. In those programs, a assignment can also additionally end up prepared for execution whilst a few however now no longer all of its direct predecessors are complete. Such responsibilities are known as OR responsibilities. The ensuing assignment gadget, containing both (AND) and (OR) responsibilities, is stated to have AND/OR priority constraints. Precedence constraints may be given through AND/OR graph or priority matrix. Based on priority constraints possible sequences may be generated. This can be the enter to GA. But Lazzerini2000 evolved most beneficial disassembly procedure plans from AND/OR relationships the use of a hierarchical Genetic Algorithm.

6.2 Methods of ASP:

For instance variety of meeting responsibilities to devise is moderate, starting from a hundred to two hundred for a standard car plant. Thus it's far nearly conceivable to remedy the hassle in an inexpensive quantity of time for the long-time period strategic making plans, including ASP (Assembly series making plans). Two strategies are taken into consideration for locating the series: 1. List all the possible meeting sequences, and examine of those sequences to get the nice series (handiest for seven or fewer components). 2. A dynamic programming method the use of the recursive formulation for meeting series.

6.2.1 Genetic algorithm:

It is a search heuristic that is inspired by Charles Darwin’s theory of natural evolution. This algorithm reflects the process of natural selection where the fittest individuals are selected for reproduction inorder to produce offspring

of the next generation. Genetic algorithms (GAs) are adaptive seek anoptimization algorithms that mimic the concepts of herbal genetics. GAs may be very unique from conventional seek and optimization strategies utilized in engineering layout issues. Because in their simplicity, ease of operation, minimum necessities and worldwide perspective, GAs has been efficiently utilized in a huge type of hassle domains.

The main characteristics of a genetic algorithm are as follows:

I. The genetic algorithm works with a coding of the parameter set, not the parameters themselves.
II. The genetic algorithm initiates its search from a population of points, not a single point.
III. The genetic algorithm uses payoff information, not derivatives.
IV. The genetic algorithm uses probabilistic transition rules, not deterministic ones.

In actual fact, a GA is a set of techniques which when common enable solutions to particular problems. To accomplish the objectives, the GA produces consecutive population alternate solutions until a solution is obtained which yields acceptable results. With in the generation of each successive population, improvements in the quality of the individual solutions are increased. In this manner, a GA can rapidly transfer to a fruitful result without having to inspect all likely solution to the problem. The procedure used is cantered on the vital processes that regulate the growth of biological organisms, namely, natural selection and reproduction. These two processes together improve an organism's capacity to persist within its atmosphere in the following manner:

I. Usual selection governs which organisms will have the chance to reproduce and persist within a population.
II. Reproduction involves genes from two discrete individuals uniting to form offspring that take over the persistence features of their parents. These algorithms pursue to start the manner in which are useful genes reproduce themselves from end toned consecutive populations and in future subsidize to the steady ability of an organism to stay.

6.2.2 Notion of natural selection:

The process of natural selection starts with the selection of fittest individuals from a population. They produce offspring which inherit the characteristics of the parents and will be added to the next generation. If parents have better fitness, their offspring will be better than parents and have a better chance at surviving. This process keeps on iterating and at the end, a generation with the fittest individuals will be found.

This notion can be applied for a search problem. We consider a set of solutions for a problem and select the set of best ones out of them.

Five phases are considered in a genetic algorithm.

Initial population
Fitness function
Selection
Crossover
Mutation

6.2.2.1 Initial Population

The process begins with a set of individuals which is called a **Population**. Each individual is a solution to the problem you want to solve.An individual is characterized by a set of parameters (variables) known as **Genes**.

Genes are joined into a string to form a **Chromosome** (solution).In a genetic algorithm, the set of genes of an individual is represented using a string, in terms of an alphabet. Usually, binary values are used (string of 1s and 0s). We say that we encode the genes in a chromosome.

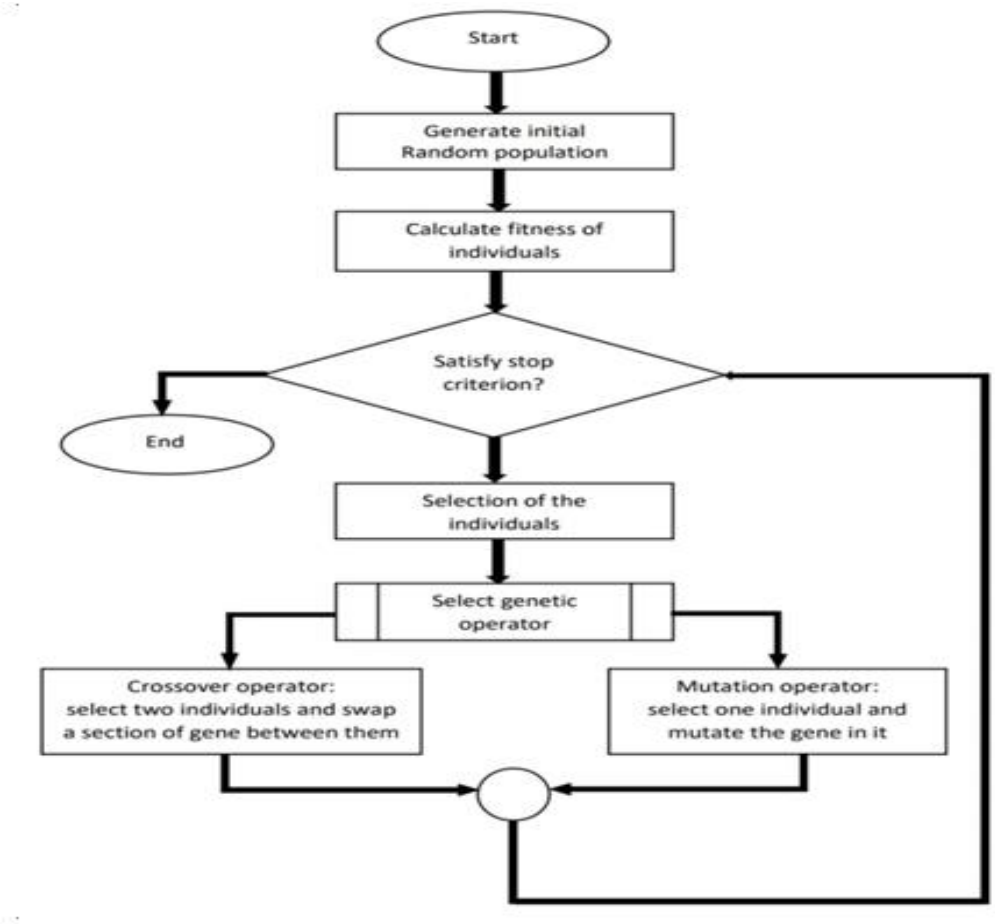
6.2.2.2 Fitness Function

The **fitness function** determines how fit an individual is (the ability of an individual to compete with other individuals). It gives a **fitness score** to each individual. The probability that an individual will be selected for reproduction is based on its fitness score.

6.2.2.3 Selection

The idea of **selection** phase is to select the fittest individuals and let them pass their genes to the next generation. Two pairs of individuals (**parents**) are selected based on their fitness scores. Individuals with high fitness have more chance to be selected for reproduction.

6.2.3GA flowchart:



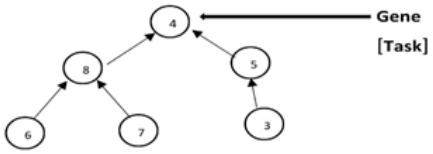
(Figure no.3)

6.2.4 Coding for Chromosome:

An assembly plan can be represented by an assembly tree. Hence the algorithm encodes the assembly plans by a parenthesis string that describes the assembly tree. The structure of chromosome, which stands for the form of

the solution, is determined to different problems. In this paper, the usage of decimal numbers is towards the genetic coding of chromosome. The chromosome length equals the number of function parts; the numeral stands for the part number, the location of the numeral indicates the assembly sequence of parts.

$$P_i = \frac{F_i}{\sum^{n}_{i} F_j}$$



Where Fi is the fitness for the string i in the population, expressed as f(x).

(1)Figure 1: Gene structure

$$F_i = \text{summation}(i,j) \in \{(i,j) | i < j\} C_{ij} \qquad (2)$$

With respect to : Assembly sequence

Subject to. :Precedence constraints

Where Pi is probability of the string i being selected. n is number of individuals in the population The values of Cij depend solely on the relative positions of the two tasks i and j, not the tasks in between, nor their absolute positions in the assembly sequence. For example, a chromosomes is "1, 2, 3, 4, 5, 6, 7, 8, 9", that means the assembly sequence is 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 → 9.

"→" denotes the precedence order of the part. Probability of the string can be calculated by using equation .1. The chromosomes which represent the assembly plans are checked for feasibility from the precedence relation of and/or graph before undergoing genetic operation. In the Fig. 1.

, genes 4, 8,6,7,5, and 3 are required to complete final assembly. All the chromosomes have the same length. It means that the number of levels is same for building a complete final product, such that in each level one task will be completed.

6.2.5 Crossover Method:

The basic operator for generation new chromosomes in the GA is that of crossover. Like its counterpart in nature, a crossover generates new individuals that have some parts of both parents' genetic material. In this paper, multipoint crossover is used. For multi-point crossover, two crossover positions are generated, then the section between these two points are exchanged between the two parents to produce two new offspring. This process is illustrated as follows.

Parent1 = (1 2|3 4 5|6 7 8)

Parent2 = (8 7|5 6 4|3 2 1)

Where "|" denotes cross over point. After the cross over operator, the offspring as follows.

Offspring1 = (1 2|5 6 4|3 7 8)

Offspring2 = (8 7|3 4 5|6 2 1)

There may be the same genes in non-crossover section as them in crossover section. Then these genes are randomly replaced by missed genes in no crossover section.

6.2.6 Mutation Method:

In natural evolution, mutation is a random process, where one allele of a gene is replaced by another to produce a new genetic structure. In GA, mutation is randomly applied with low probability, typically in the range 0.001 and 0.1, and modifies elements in the chromosomes. Here we choose two genes and exchange their positions. Results and discussion To validate the proposed methodology a simple product is considered. Thus, the proposed genetic algorithm is applied for the Fixture assembly is discussed below. Bill of material and the exploded view of fixture assembly is shown in Fig. 2

6.2.7 Pseudo code:

START

Generate the initial population

Compute fitness

REPEAT

Selection

Crossover

Mutation

Compute fitness

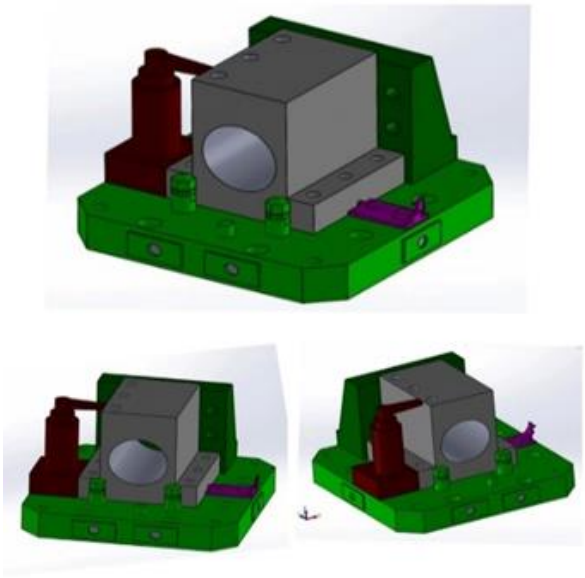
UNTIL population has converged

STOP

6.2.8 GA Input Parameters:

I. The crossover rate is set at0.55;
II. The mutation rate0.1;
III. The population size 60;
IV. and the maximum number of generations 300.

6.2.9 GA Output:



Items no.	Part's name	Quantity
01	Work piece	1
02	Support	1
03	Locator pin sub-assembly	2
04	Locator pin sub-assembly	2
05	Locator pin sub-assembly	2
06	Strap clam	1
07	Base clamp	1
08	Plunger(base clamp)	1
09	Link(base clamp)	2
10	Knob(base clamp)	1
11	Lever(base clamp)	1
12	Clamp	1
13	Pin	3
14	Fixture base	1

(BOM and 3D view of fixture assembly) : figure no.4

As a result, the fitness is Processing time f(x) = 7.9seconds;

The optimized sequence is1->8->->18->6->20->2->12->9->->11->16->10->->19->13->17->21->15->3->14->4->7->5->->1.

7 CONCLUSION:

The muster plans are evaluated based on make span, which includes operational complexity and degrees of freedom. A purposely-developed genetic operator is used to perform crossbreed and mutation function. Thus a Genetic Algorithm is proposed to solve an assembly sequence problem, a much more difficult problem than other sequencing problems. The algorithm responds quickly to produce a set of good assembly plans, but entails difficult in generating all the valid ones. The proposed methodology can be adapted to assemblies where computational time is significant. The main aim of this Paper is to realize the non-traditional optimization techniques in assembly sequencing problem. Visual studio program (ASP.cs) was developed for the optimal assembly sequence. Future work will be focus on the application of this methodology to complex products with a large number of parts. We are currently working on a methodology to identify the best way of performing disassembly for the products with large number of parts so as to satisfy some optimization criteria, e.g., minimization of the cost of material.

A key feature of these metaheuristics procedures is their ability to escape from local optima and perform a robust search of feasible region. This paper introduces the most prominent types of non-conventional type algorithms or metaheuristics. In addition, it may employ intensification and diversification strategies based on

long-term memory to focus the search on promising continuous. The following are the advantages of non-traditional techniques over the traditional techniques:

I. The non-traditional techniques yield a global optimal solution.
II. The techniques use a population of points during search.
III. Initial populations are generated randomly which enable to explore the search space.
IV. The techniques efficiently explore the new combinations with available knowledge to find a new generation.
V. The objective functions are used rather than their derivatives

Reference

[1] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," Science, vol. 220, no. 4598, pp. 671–680, 1983.

[2] D. E. Goldberg, Genetic Algorithms in Search, Optimizationand Machine Learning, Addison-Wesley, Reading, Mass, USA,1989.

[3] R. M. Marian, L.H.S. Luong, K. Abhary, A genetic algorithm for the optimisation of assembly sequences, Computers & Industrial Engineering, 50, pp. 503–527, 2006.

[4] Y.K. Choi, D.M. Lee, Y.B. Cho, An approach to multi-criteria assembly sequence planning using genetic algorithms, Int. J. of Adv. Manuf. Tech. 42, pp. 180-188, 2009.

[5] A. Yasin, N. Puteh, R. Daud, M. Omar, S.L.S. Abdullah, Product assembly sequence optimisation based on genetic algorithm, Int. J. on Comp. Sci. and Eng. 29, pp. 3065-3070, 2010.

[6] D.S. Hong and H.S. Cho, A genetic-algorithm based approach to the generation of robotic assembly sequence, Control Engineering Practice, 7, pp. 151-159, 1999.

[7] D. Whitley, An executable model of a simple genetic algorithm. Foundations of genetic algorithms, 2 (15-19), pp. 45-62, 2014.

[8] R.M. Marian, L.H.S. Luong, K. Abhary, Assembly sequence planning and optimisation using genetic algorithms Part I. Automatic generation of feasible assembly sequences, Applied Soft Computing, 2/3F, pp. 223–253, 2003.