Post-print versions published by authors.
DOI: 10.1115/1.4039767
Journal of Manufacturing Science and Engineering JULY 2018, Vol. 140 / 071015-1

A Multistage Approach to the Selective Assembly of Components without Dimensional Distribution Assumptions

Abolfazl Rezaei Aderiani *
Department of Industrial and Materials Science
Chalmers University of Technology
SE-412 96 Gothenburg, Sweden
Email: aderiani@chalmers.se

Kristina Wärmefjord
Department of Industrial and Materials Science
Chalmers University of Technology
SE-412 96 Gothenburg, Sweden
Email:kristina.warmefjord@chalmers.se

Rikard Söderberg
Department of Industrial and Materials Science
Chalmers University of Technology
SE-412 96 Gothenburg, Sweden
Email: rikard.soderberg@chalmers.se

Selective assembly is a means of obtaining higher quality product assemblies by using relatively low-quality components. Components are selected and classified according to their dimensions and then assembled. Past research has often focused on components that have normal dimensional distributions to try to find assemblies with minimal variation and surplus parts. This paper presents a multistage approach to selective assembly for all distributions of components and with no surplus, thus offering less variation compared to similar approaches. The problem is divided into different stages and a genetic algorithm is used to find the best combination of groups of parts in each stage. This approach is applied to two available cases from the literature. The results show improvement of up to 20 percent in variation compared to past approaches.

1 Introduction

To survive on the market, products need to be of high quality and relatively low on cost. Most products involve some assemblies and subassemblies which are, in turn, dependent on the quality of the mating parts. Thus if product quality is to be improved, higher quality mating parts must be manufactured. This usually means tighter tolerances; a method of quality improvement that can make the production process more expensive. Selective assembly makes higher quality assemblies possible whilst using (relatively speaking) lower quality parts. Parts of equivalent dimensions are selected from each component being assembled (in this paper the word 'components' refers to the elements of an assembly and 'parts' refers to produced parts of that element for mass production). However, there are some problems inherent in this technique; a major issue is mismatched or surplus parts if the number of parts in some groups does not fit into their mating group.

Bearing and engine industries have been using this technique since before 1950 to get the power of making tight fits between parts. However, increased competitiveness in manufacturing industry and increased availability of inspection data, in line with trends such as Industry 4.0, open up for a broader use of selective assembly. Mass-production can be replaced with mass-customization and an adoption of the production processes to incoming material is suggested by Soderberg et al [1], where a digital twin is used for geometry assurance. Selective assembly supports this vision.

1.1 Selective assembly

The first step of selective assembly is to divide manufactured parts into groups, based on their measured dimensions. This is because, in large scale, it is not practical to do the matching for every part. The next step is to find the best matching groups so that the variation of the target dimension for all produced assemblies is minimum.

Suppose a product which involves two components (A and B) will be assembled. If the target for the number of products is 1000, obviously 1000 individual parts from each component should be manufactured. After production of these individual parts, dimensions of the parts that affect the target dimension of the assembly will be measured. Then, the individual parts will be divided into some groups (for example 6 groups) based on their dimensions. This grouping, shown in Figure 1, is done based on the assumption that dimensions of produced parts follow the normal distribution.

*Corresponding author.

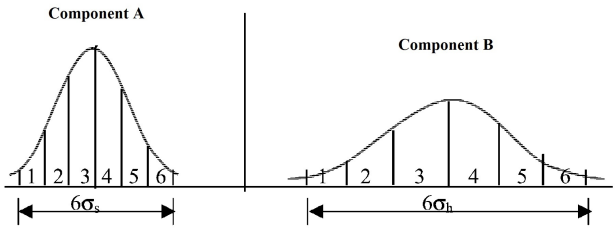

Fig. 1: Grouping of parts based on their dimensions

Then each group of Component A should have a match group in Component B. But, the problem is that the number of parts in those groups are not necessarily equal. Therefore, if, for instance, the first group of Component A is selected to be assembled with the third group of Component B, they may not have the same number of parts. Therefore, some parts of one group cannot be used for assembly. These parts are called surplus parts and this problem is called mismatching.

Research into the area of selective assembly began by Mansor [2], who presented a specific procedure for selective assembly to reduce surplus parts for shaft and hole problems. The procedure divided the problem into three categories based on the design, process and workshop tolerances of each part and developed a solution for each category. Desmond [3] presented a case study and investigated selective assembly for this problem. They introduced a parameter called the Grading Factor (GF); the improved assembly is quantified, with a lower GF signifying a higher quality assembly. This factor is then obtained for different numbers of selective groups. The borders of groups are defined to obtain minimum GF. Pugh [4] presented a method of selective assembly for components of different variance in normal distribution by calculating the limits for components with major variance so that those produced at extremes of the distribution would be discarded during assembly. A new method of grouping was presented by Fang et al [5] to minimise the surplus parts (mismatching). This paper deals solely with shaft and hole problems. Although the number of surplus parts decreased with this method, some surplus parts remained. Another article by Fang [6] developed a new quantitative criterion for predicting the matchability of parts in selective assembly. An algorithm was then produced, to calculate manufacturing parameters and achieve the desired matchability. Chan and Linn [7] presented a new grouping method for shaft and hole problems based on equal probabilities of groups. Cumulative normal distribution was used for making groups with equal probabilities. Mease et al [8] developed a new method of binning based on minimising the absolute and square-loss functions. This method has a significantly lower loss compared to heuristic methods, such as equal-width and equal-area binning.

1.2 Optimisation algorithms in selective assembly

A genetic algorithm (GA) is used extensively to find the optimum combination of groups in selective assembly. GA is designed based on a class of evolutionary algorithms using techniques inspired by Darwinian evolutionary theory, such as inheritance, mutation, natural selection and crossover. In GA, a population of candidate solutions is created randomly and evolves toward better solutions in each iteration. Each candidate solution is called a chromosome or phenotype. These chromosomes can be represented in strings of binary, integer or real numbers. Ponnambalm [9] presented a method of coding for selective assembly using GA. Their objective was to minimise the deviation of assemblies by selecting the optimum combination of groups in an assembly. In their utilized coding procedure, a chromosome is divided into substrings equal to the number of components. Each substring has a number of genes equal to the number of groups in each component. Therefore, the length of a chromosome is equal to the number of components multiplied by the number of groups and each gene represents a group. Consider, for example, the assembly has 3 components, and individual parts of each component are divided into 6 groups. Then, in a random chromosome like this: 256143 134526 561243, the first six numbers are component A substring, the second six genes represent the group numbers in component B and the last six integers show the group numbers in component C.. Translation of this chromosome into combinations of groups will be (A2 B1 C5), (A5 B3 C6), (A6 B4 C1), (A1 B5 C2), (A4 B2 C4) and (A3 B6 C3). To create the first generation, three random permutations of 1-6 are generated for each chromosome. Crossover only occurs within substrings.

Kumar et al [10] used GA to find the optimum combination of groups and thus minimise surplus parts. After finding the best combination, surplus parts are sorted in ascending order and divided into three groups. GA is then used again to assemble the parts. Asha et al [11] used optimisation of clearance variation to solve piston and cylinder problems. This assembly has four clearances and so the problem is one of multi-objective optimisation. For this reason, a non-dominated sorting genetic algorithm (NSGA) was used to find the optimum solutions. Kanan et al [12] carried out selective assembly in three stages. Each stage uses a pair of groups comprising a minimum number of parts. These groups are then removed in the next stage of optimisation and the optimum combination of the remaining groups is again obtained using GA. However, this is based on the assumption of normal distribution of parts. Kumar et al [13] used selective assembly in a three-stage strategy to optimise group combinations using a particle-swarm algorithm.

In another paper [14], these authors presented a new method of optimising parts with skewed normal distribution using GA; the result was zero surplus parts. As already mentioned, the common method is to consider the length of chromosome as the product of the number of components and the number of groups. However, in their methodology, the length of each chromosome was given as $N \times (2n-1)$ genes (where n = number of groups and N = number of components). Hence, the length of each substring is $2n-1$ instead of n. Therefore, each group number could be repeated in a substring more than once. Each combination of group numbers results in an assembly set. However, number of assem-

blies that can be produced by that combination is equal as the number of parts in the group that contains minimum number of parts. Therefore, the remaining parts in other groups are surplus. The algorithm repeats these group numbers in the remaining genes so that those remaining parts could be used and finally after the last gene of a chromosome all parts in all groups are consumed. The main problem of this methodology is that most of genes are forced to take specific group numbers instead of letting GA to find optimum combinations. Using this strategy, Wang et al [15], optimised the group combinations for three components. There was no assumption on dimensional distribution of parts in their case study.

Another strategy used by Raj et al [16] is to find the best combination of all parts instead of dividing them into some groups. This method should result in less variation of assemblies, however, the length of chromosomes is related to the number of parts. Hence, when the number of parts of each component increases, the algorithm may not converge at all. In another research, Xu [17] have done selective assembly with an application for hard disk drives. In their research they have utilized two strategies to reduce the variation and the surplus parts, the "discarding theorem", which defines the threshold condition for discarding the most inferior parts, and the "bining theorem", which defines how to match the remaining parts. The presented strategy is limited to assemblies with only two components and components with a normal distribution of parts.

1.3 Scope of the paper

This paper presents a new methodology for optimising group combinations and selective assembly, which will result in zero surplus parts. Dimensional distribution of the mating parts is not restricted to exact normal distribution. Based on the literature, only one method has this capability (as presented by Kumar et all [14] and Wang et al [15]). However, since that methodology finds the optimum combinations of groups in just one stage, it forces some genes to take group numbers that have surplus parts in each new chromosome. Therefore, some combinations are selected without optimization process. The method presented in this paper uses different stages of optimization which makes it possible to achieve lower variation across all assemblies. When comparing the results of previous papers with those from the algorithm developed in this paper, a considerable improvement of up to 20 percent in variation of the final assemblies is evident.

This paper consists of seven sections. Section 2 outlines the problem and presents two sample cases used to evaluate the results. Section 3 describes the methodology of the algorithm used. Section 4, entitled Genetic algorithm, discusses the coding procedure and genetic operators in detail. Section 5 applies the algorithm to the sample cases and presents the results. Finally, there is a discussion of the results in section 6 and a summary in section 7.

Table 1: Number of parts in each group of Sample case 1

Component	Number of parts in groups					
	1	2	3	4	5	6
A	12	67	260	370	256	35
B	5	111	448	331	98	7

2 Problem outline and sample cases

To be able to compare the results from this paper with those from [14] and [15], the same assemblies as given in their papers will be examined here. The utilized sample in [14] is a shaft and hole example and the case in [15] is an assembly of three gears. The two cases are described below.

2.1 Sample case 1

The first sample case is a shaft and hole assembly presented by Kumar et al [14]. The tolerances of each component are given in Figure 2.

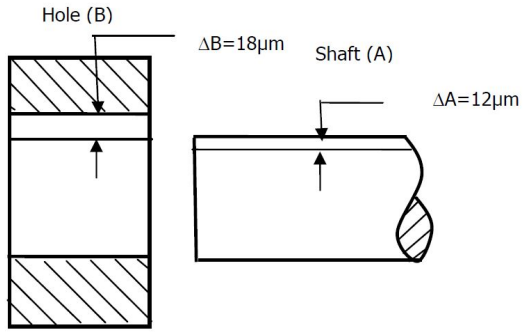

Fig. 2: Sample case 1 [14]

The diameter of shaft is $35^{+0.000}_{-0.012}$ and $35^{+0.018}_{-0.000}$ for the hole. Therefore, considering interchangeable assembly of shafts and holes, the maximum clearance range will be 30 μm and the objective is to reduce this clearance range. In this case, the goal is to assemble 1000 shaft and holes. The exact dimension of each part is created considering 0.2 skewness in normal distribution dimensions of the mating parts. They have also considered six equal groups for selective assembly. Considering the tolerances of each part, the range of dimensions in each group for the shaft is 2 μm and for the hole 3 μm. All parts are distributed among groups based on their dimensions. Number of parts for each component in each group is shown in Table 1.

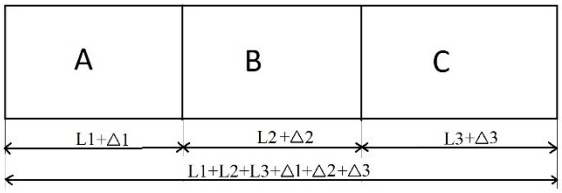

Fig. 3: Linear assembly of sample case 2

2.2 Sample case 2

This case is a gearbox assembly that is utilized by [15] and it is a linear assembly consisting of three components. The tolerance range of each component and their part distributions all differ. This assembly is shown in Figure 3. In this case, when carrying out an interchangeable assembly, maximum and minimum tolerance limits are obtained from Equations 1 and 2.

$$T_{max} = \Delta 1_{max} + \Delta 2_{max} + \Delta 3_{max} \tag{1}$$

$$T_{min} = \Delta 1_{min} + \Delta 2_{min} + \Delta 3_{min} \tag{2}$$

If the lower tolerance limit of each component is considered to be zero and the upper limit is 12, 15 and 18 μm for $\Delta 1$, $\Delta 2$ and $\Delta 3$ respectively, then:

$$T_{max} = 12 + 15 + 18 = 45 \mu m$$

$$T_{min} = 0$$

The dimensions of the components are $10^{+0.012}_{-0.000}$, $12^{+0.015}_{-0.000}$ and $14^{+0.018}_{-0.000}$ for components A, B and C respectively based on the work of Wang et al [15]. Therefore, the dimension of the assembly will be $36^{+0.045}_{-0.000}$ when an interchangeable assembly is used. Thus, the variation range of the assemblies will be 0.045 mm and the problem is that this range of variation is too high. The goal is to decrease this variation as much as possible, using selective assembly. Parts of each component are divided into six groups based on their dimensions (measured from produced parts). Considering the tolerance of each component and number of groups, the width of each group is 2 μm, 2.5 μm and 3 μm for components A, B and C respectively. The number of parts in each group and the total number of assemblies are shown in Table 3 and Table 2, respectively.

Table 2: Input variables for the problem [15]

Variable	Input
Total number of parts for each component	1000
Number of components	3
Number of groups	6
Tolerance of component A	12 μm
Tolerance of component B	15 μm
Tolerance of component C	18 μm

Table 3: Number of parts in each group

Component	Number of parts in groups					
	1	2	3	4	5	6
A	9	50	175	375	256	135
B	10	111	438	321	108	12
C	12	67	220	390	236	75

3 Methodology

The methodology developed in this paper for implementing selective assembly is illustrated using mathematical formulations. These formulas are applied to the second sample case to make them more clear. However, the procedure is general and can be applied to all kinds of selective assembly problems. The proposed methodology is also applied to the first case and the results from this are presented in the next section. The flowchart of the presented methodology is also illustrated at Figure 4.

The methodology is based on several stages of combinatorial optimisation. Firstly, the optimum combination of all groups of all components is calculated so that the variation of assemblies is at a minimum. Considering the sample case 2, the optimum combination is calculated as this: 132456 364152 624513, which results in a reduction of the dimensional variation to the range of 0.0095 - 0.0275. This optimum combination is calculated using GA, described in Section 4. Following Figure 4, the start of the algorithm is shown by Begin block. In this Figure, i represents the number of each stage and since in the first stage there is not a problem for finding the best combination, this stage does not need the first 5 steps of the algorithm. The number of assemblies that could be produced at this stage (using this combination of groups) can be calculated using Equation 3:

$$NA_i = ng \times m_i \tag{3}$$

Here NA_i represents the number of assemblies that could be produced by the optimum combination which is calculated in stage i; ng shows the number of groups and m_i is the number of parts in the group with the minimum number of parts among all groups of all components which can be obtained

Table 4: Number of parts in each group after first stage

Component	Number of parts in groups					
	1	2	3	4	5	6
A	0	41	166	366	247	126
B	**1**	102	429	312	99	3
C	3	58	211	381	227	66

using Equation 4:

$$m_i = min(A^i_j, B^i_q, C^i_r) \tag{4}$$

The variables are:
A^i_j the number of parts from component A that fall in the j^{th} group of this component in stage i, $j = 1,2,3,\ldots 6$
B^i_q the number of parts from component B that fall in the q^{th} group of this component in stage i, $q = 1,2,3,\ldots 6$
C^i_r the number of parts from component C that fall in the r^{th} group of this component in stage i, $r = 1,2,3,\ldots 6$
Therefore, considering the case sample, number of assemblies that could be produced by the optimum combination of stage 1 is:
$m_1 = min(A^1_j, B^1_q, C^1_r) = A^1_1 = 9$
$NA_1 = 6 \times 9 = 54$
It means that the calculated combination of groups in stage 1 can be applied to make 54 assemblies. The number of parts used for making these assemblies must be subtracted from all groups of all components. Therefore, the new number of parts in different groups are calculated from Equations 5, 6 and 7 for component A, B and C respectively. This step is shown by step 7 at the presented flowchart at Figure 4.

$$A^i_j = A^i_j - m_i \tag{5}$$

$$B^i_q = B^i_q - m_i \tag{6}$$

$$C^i_r = C^i_r - m_i \tag{7}$$

Using Equations 5, 6 and 7 new numbers of parts in different groups for the case sample are calculated and shown in Table 4. After that, there are no parts in group 1 of component A. The stage 1 on of selective assembly is then finished.

To be able to continue finding the best combinations, we need to load the empty groups with parts from other groups. Obviously, if parts from other groups are supposed to load a group, they must be from groups of the same component. The groups with maximum number of parts are considered for this and they are shown by *MA*, *MB*, and *MC* for component A, B and C, respectively. These values can be calculated from Equation 8, 9 and 10 (steps 1 and 2 of stage 2 at Figure 4).

$$MA_i = max(A^i_j) \tag{8}$$

$$MB_i = max(B^i_q) \tag{9}$$

$$MC_i = max(C^i_r) \tag{10}$$

Each empty group in stage i will be loaded as m_i. Thus, m_i multiplied by the number of empty groups for each component must be subtracted from M_i of that component. This is shown in Equation 11, 12 and 13 for components A, B and C, respectively (step 3 of stage 2 at Figure 4).

$$MA_i = MA_i - p_i m_i \tag{11}$$

$$MB_i = MB_i - q_i m_i \tag{12}$$

$$MC_i = MC_i - r_i m_i \tag{13}$$

Where:
p_i represents the number of groups which contain no parts from component A in stage i.
q_i represents the number of groups which contain no parts from component B in stage i.
r_i represents the number of groups which contain no parts from component C in stage i.
For the sample case in stage 2, only component A has a group containing no parts. Hence, $p_2 = 1$, $q_2 = 0$ and $r_2 = 0$. Using Equation 4, m_2 is:
$m_2 = min(A^2_j, B^2_q, C^2_r) = B^2_1 = 1$
From Equation 8, MA_2 can be calculated:
$MA_2 = max(A^2_j) = A^2_4 = 366$
Then, using Equation 11 the new amount of MA_2 is obtained:
$A^2_4 = A^2_4 - p_2 m_2 = 366 - 1 = 365$
Afterwards, all empty groups will load as much as the amount of m_i:(step 4 of stage 2 at Figure 4).
$A^2_1 = m_2 = 1$
The new number of parts in the different groups are shown in Table 5. The numbers for the group that has been loaded, and the group doing the loading, are shown in bold.

Now, the next optimum combination of groups can be calculated using an optimization algorithm. But, instead of group number 1 from component A, group number 4 must be considered for optimization (step 5 of stage 2 at Figure 4). In other words, there is no gene with number 1 in the first substring of chromosomes anymore. Instead, there are two

Table 5: Number of parts in each group after loading the first zero group

Component	Number of parts in groups					
	1	2	3	4	5	6
A	**1**	41	166	**365**	247	126
B	1	102	429	312	99	3
C	3	58	211	381	227	66

genes with number 4 in that substring. The optimum combination of this stage is calculated as: 465423 641325 125463 (step 6 of stage 2 at Figure 4). The number of assemblies that can be assembled using this combination are calculated from Equation 3:

$NA_2 = 6 \times 1 = 6$

By subtracting the amount of m_i from all groups (step 7 at Figure 4) stage 2 is finished now, and the next stage can be done using the same procedure.

This procedure continues to the point when at least the result of one of Equations 11, 12 or 13 is equal or less than zero. The best combination of the remaining parts is then calculated to obtain the minimum dimensional variation.

4 Genetic algorithm

The optimum combination of groups at each stage can be obtained using different optimisation algorithms. A genetic algorithm (GA) is designed based on a class of evolutionary optimisation algorithms that use techniques inspired by Darwinian evolutionary theory: inheritance, mutation, natural selection and crossover. GA is used in this paper to obtain the optimum combination of groups in each stage of the proposed methodology. The coding procedure is the same as in previous papers utilising GA, which is explained in Section 1. Details of the utilized algorithm are discussed in this section.

4.1 Selection

The fitnesses of chromosomes are calculated based on an exponential scaling factor. This function is shown in Equation 14.

$$fitness(i) = e^{(-kfit(i))} \tag{14}$$

where:

$k = 0.05$

The objective function ($fit(i)$) is the dimensional variation of assemblies in chromosome i. This function can be obtained using Equation 15.

$$fit(i) = max(T^n_{max}) - min(T^n_{min}); \quad n = 1,2,3,\ldots 6 \tag{15}$$

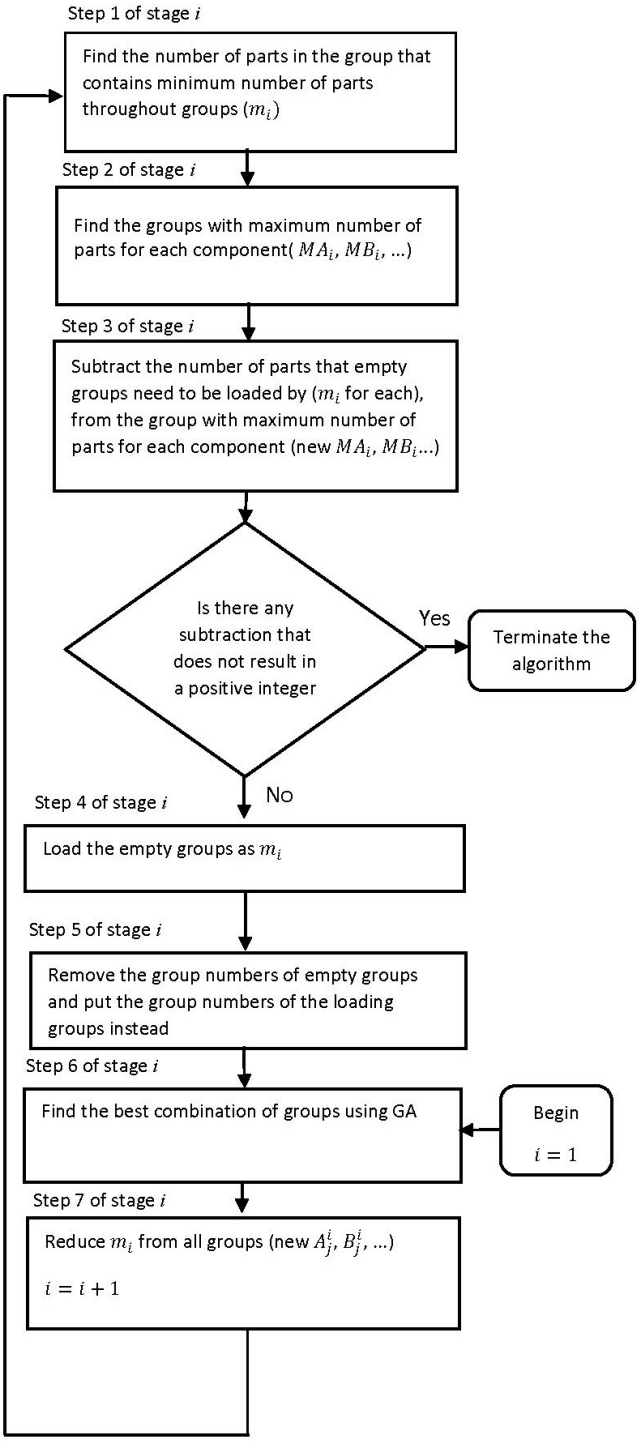

Fig. 4: Flow chart of the presented methodology

Since the tolerance ranges for each group of components A, B and C are 2, 2.5 and 3 respectively, the maximum and minimum tolerance limits of each combination can be calculated from Equations 16 and 17.

$$T^n_{max} = 2(N^n_A) + 2.5(N^n_B) + 3(N^n_C); \quad n = 1,2,3,\ldots 6 \tag{16}$$

$$T^n_{min} = 2(N^n_A - 1) + 2.5(N^n_B - 1) + 3(N^n_C - 1) \quad (17)$$

Where, N^n_j is the group number of component j in the n^{th} combination.

Consider a chromosome with 465423 641325 125463 combinations as an example. Using Equation 16 and 17, the maximum and minimum tolerance limits are calculated as follows.

$T^1_{max} = 2(N^1_A) + 2.5(N^1_B) + 3(N^1_C) = 2(4) + 2.5(6) + 3(1) = 26$

$T^1_{min} = 2(N^1_A - 1) + 2.5(N^1_B - 1) + 3(N^1_C - 1) = 2(3) + 2.5(5) + 0 = 18.5$

Doing this procedure for $n = 2, 3, 4, 5$ and 6, six T_{max} and six T_{min} are obtained as follows:

$T^2_{max} = 28$, $T^3_{max} = 27.5$, $T^4_{max} = 27.5$, $T^5_{max} = 27$ and $T^6_{max} = 27.5$

$T^2_{min} = 20.5$, $T^3_{min} = 20$, $T^4_{min} = 20$, $T^5_{min} = 19.5$ and $T^6_{min} = 20$

Using Equation 15, the fitness of this chromosome is calculated.

$fit(i) = 28 - 18.5 = 9.5$

Roulette wheel selection, see [18], is used to select chromosomes based on their fitness. In this kind of selection, fitness is used to associate a probability with each chromosome. If the fitness of chromosome i is considered as fi, the probability of its selection is calculated using Equation 18.

$$p_i = \frac{f_i}{\Sigma^N_{(j=1)} f_j} \quad (18)$$

where N is the total number of individuals in a population.

4.2 Crossover operation

There are some different methods of doing crossover in GA for sequence optimisations. However, there is a main difference between this problem and those sequence optimisation problems. In this problem, after the first stage, the sequence of numbers is not complete. Some numbers are removed from the complete sequence and some other numbers are repeated instead of them. Because of that, it is not possible to apply those existing crossover operations that are based on a pair of parents on this problem. Therefore, the crossover operation is carried out for just one parent (instead of two) and only between substrings. If a chromosome is selected for the crossover operation, the child will be produced by creating a random integer less than the substrings' length. The first substring is then split from the gene whose order is same as that of the random number. These two substring sections then swap their order. This procedure is repeated in other substrings as well. For example, consider a chromosome selected for crossover as 256143 134526 561243. Since the length of each substring is six, the random number will be 1-5. If that number is 3, the first substring after crossover will be **143**256.

4.3 Mutation

The common mutation operation in this kind of optimisation is to change the location of the gene selected for mutation with its next gene. This procedure is used in this paper. For instance, for chromosome 256143 134526 561243, if only the second gene is selected for mutation, the chromosome after mutation will be 2**65**143 134526 561243. The genes whose location has changed are shown in bold.

4.4 Convergence criteria

The convergence criterion considered here is that if the best chromosome in the whole population does not change after a specific number of iterations, then the algorithm stops. This number has been calculated by trial and error as 150 iterations.

5 Results

GA inputs such as crossover and mutation rates are considered as 0.6 and 0.05 respectively based on previous research [14], and a population size of 50 is considered for the optimization. Results of the optimisation for each sample case are presented in this section.

5.1 Sample case 1

Based on the presented methodology, stages of selective assembly continue up to creating at least one zero or minus number form Equations 11, 12 or 13 (the new amount of parts in the groups with maximum number of parts). This stopping criterion occurs in stage number 7 for the sample cases 1. There are 334 assemblies left after 7 stages of optimisation that are assembled in another stage. The details of each stage of optimisation for this sample case are given in Table 6. The second column of the table shows the group numbers that each chromosome can take for the optimization. The first row of these numbers are group numbers of shafts and the second row represents the group numbers of holes. These numbers are two complete sequences of one to six for the first stage. But for other stages, some numbers are removed from the complete sequences and some other numbers are repeated instead. For instance, in the second stage, there is not group number one for the holes, and there are two group number three instead. The third column represents the number of parts in each group of each component after empty groups are loaded (Step 6 at Figure 4). The next column depicts the optimal combination of groups in that stage. Number of assemblies that can be produced with that optimal combination is presented in the fifth column of this table. This number is calculated using Equation 3. The last column also indicates the minimum and maximum tolerance limit of assemblies that are produced by the optimal combination of that stage. The minimum and maximum tolerance limits among all stages (that are used to calculate the final variation of assemblies) are shown in bold.

The summation of assemblies up to stage 7 is 666. After stage 7, there are 110, 79 and 145 parts left from groups 1, 2 and 3 of the shaft respectively. In addition, 176 and 158 parts

Table 6: Details of each stage of optimisation for the sample case 1

Stage number	Group numbers in each gene considered for optimisation	Number of parts in each group before optimisation	Optimum combination	Number of Assemblies	Minimum and maximum tolerance limits of the optimum combination
	1 2 3 4 5 6	12 67 260 370 256 35	245136 542631	30	20
1	1 2 3 4 5 6	5 111 448 331 98 7			**10**
	1 2 3 4 5 6	7 62 255 365 251 30	162534 625343	12	20
2	3 2 3 4 5 6	2 106 441 326 93 2			12
	1 2 3 4 5 6	5 60 253 363 249 28	423165 343523	30	19
3	3 2 3 4 5 3	5 104 429 324 91 5			10
	4 2 3 4 5 6	23 55 248 335 244 23	463425 324353	138	19
4	3 2 3 4 5 3	23 99 378 319 86 23			12
	4 2 3 4 5 4	32 32 225 248 221 32	432544 345233	192	19
5	3 2 3 4 5 3	32 76 291 296 63 32			11
	4 4 3 4 5 4	31 31 193 123 189 31	443544 445234	186	**21**
6	4 2 3 4 5 4	31 44 259 202 31 31			11
	3 3 3 4 5 3	13 13 123 92 158 13	335343 432333	78	18
7	3 2 3 4 3 3	13 13 189 171 13 13			10
	0 0 3 4 5 0	0 0 110 79 145 0	5344 3434	333	20
8	0 0 3 4 0 0	0 0 176 158 0 0			12

from groups 3 and 4 of the holes are also left. These parts can be assembled so that A5 and B3 make 145 assemblies, A3 and B4 make 110 assemblies, and A4 with the remaining parts of B3 and B4 make the 79 remaining assemblies. The maximum and minimum tolerance limits of these two assembly sets will be 20 and 12 respectively, based on Equations 16 and 17. This is shown as stage 8 in Table 6.

Considering the total number of assemblies in all stages, 1000 assemblies are produced, therefore, no surplus part is remains. The dimensional variation will be the difference between the maximum and minimum tolerance limits of all stages. The maximum tolerance limit will be produced in stages 6 and equals 21. The minimum tolerance limit will be produced from stage 1, 3, and 7, and is 10. Therefore, the maximum dimensional variation is 11. Using the previous method, the dimensional variation is 13 [14]. Hence, an improvement of 15 percent in the maximum variation is gained by using this method, compared with the previous method for the sample case 1.

5.2 Sample case 2

The stopping criterion occurs in stage number 12 for the second sample case. After 12 stages of optimisation the total number of assemblies left is 40. These are assembled at another stage. The total number of stages is thus 13. Table 7 shows the details of each stage of optimisation for this case. This table is structured same as Table 6. However, since the assembly in this sample case consists of three components, there are three rows in the second and third column for each stage. The first, second and third row of these columns represent components A, B, and C respectively. The optimum combination of groups that are shown in the fourth column is also consist of three substrings. The first, second and third section represent components A, B, and C in the optimal combination, respectively. Number of assemblies in each stage is shown in the fifth column. Summation of these numbers for all stages is 1000. Therefore, all parts are used for the assemblies and there is no surplus part.

After stage 12, the remaining parts are 15 parts from group 3 and 25 parts from group 5 (component A), 25 parts from group 3 and 15 parts from group 4 (component B) and 40 parts from group 4 (component C). These remaining parts can then be assembled so that 25 parts of A5, B3 and C4 make 25 assemblies and 15 parts of A3, B4, and C4 make 15 assemblies. The maximum and minimum tolerance limits of these two assembly sets will be 29.5 and 20.5 respectively, based on Equations 16 and 17. The maximum tolerance limit will be produced in stages 9 and 10 and equals 32.5. The minimum tolerance limit will be produced from stage 1 and is 18. Therefore, the maximum dimensional variation is 14.5. Using the previous method, [15] have obtained the dimensional variation of 17.5 . Hence, an improvement of 20 percent in the maximum variation is gained by using this method for the sample case 2.

6 Discussion

Selective assembly in previous research has generally dealt with components which have normal dimensional distribution of parts. In practice, however, the produced parts might not always follow the normal distribution. The method presented here makes no assumptions regarding the normal distribution of parts and it results in zero surplus parts. The application of this approach using other optimisation algorithms could be assessed in future research. In addition, ap-

Table 7: Details of each stage of optimisation for sample case 2

Stage number	Group numbers in each gene considered for optimisation	Number of parts in each group before optimisation	Optimum combination	Number of assemblies	Minimum and maximum tolerance limits of the optimum combination
1	1 2 3 4 5 6 1 2 3 4 5 6 1 2 3 4 5 6	9 50 175 375 256 135 10 111 438 321 108 12 12 67 220 390 236 75	132456 364152 624513	54	27.5 **18**
2	4 2 3 4 5 6 1 2 3 4 5 6 1 2 3 4 5 6	1 41 166 365 247 126 1 102 429 312 99 3 3 58 211 381 227 66	465423 641325 125463	6	28 18.5
3	4 2 3 4 5 6 3 2 3 4 5 6 1 2 3 4 5 6	2 40 165 362 246 125 2 101 426 311 98 2 2 57 210 380 226 65	256443 235643 641235	12	29.5 19.5
4	4 2 3 4 5 6 3 2 3 4 5 3 4 2 3 4 5 6	38 38 163 322 244 123 38 99 348 309 96 38 38 55 208 340 224 65	623544 334523 364254	228	29.5 20
5	4 4 3 4 5 6 3 2 3 4 5 3 4 2 3 4 5 6	17 17 125 250 206 85 17 61 276 271 58 17 17 17 170 285 186 25	462435 333245 436542	102	29.5 20
6	4 4 3 4 5 6 3 2 3 4 5 3 4 4 3 4 5 6	8 8 108 217 189 68 8 44 243 254 41 8 8 8 153 252 169 8	546443 435332 624513	48	30.5 20
7	4 4 3 4 5 6 4 2 3 4 5 4 4 4 3 4 5 4	33 33 100 143 181 60 33 36 235 180 33 33 33 33 145 145 161 33	446354 432335 454643	198	31.5 21.5
8	5 5 3 4 5 6 3 2 3 4 3 3 5 5 3 4 5 5	3 3 67 110 142 27 3 3 193 147 3 3 3 3 112 112 119 3	344456 534442 454434	18	30.5 21.5
9	5 5 3 4 5 6 3 3 3 4 3 3 5 5 3 4 5 5	24 24 64 107 91 24 24 24 94 144 24 24 24 24 109 109 44 24	535645 332343 555345	144	**32.5** 21
10	4 4 3 4 5 4 4 4 3 4 4 4 3 3 3 4 5 3	20 20 40 23 67 20 20 20 70 40 20 20 20 20 25 85 20 20	536455 433333 354555	120	**32.5** 21
11	5 5 3 4 5 5 3 3 3 4 3 3 4 4 3 4 4 4	3 3 20 3 38 3 3 3 38 20 3 3 3 3 5 53 3 3	445443 444443 333435	18	30 19.5
12	5 5 3 5 5 5 3 3 3 4 3 3 4 4 3 4 4 4	2 2 17 2 27 2 2 2 27 17 2 2 2 2 2 42 2 2	435555 343333 443444	12	29.5 19
13	0 0 3 0 5 0 0 0 3 4 0 0 0 0 0 4 0 0	0 0 15 0 25 0 0 0 25 15 0 0 0 0 0 40 0 0	53 34 44	40	29.5 20.5

plying this method for problems that have more than one dimension that variation of it should be minimum can be investigated in future. A MATLAB code has been developed for this algorithm. The simulation time for all stages using the methodology presented here is 155 seconds, using a Core i7 CPU and 16 MB of RAM for the sample case 2 and 65 seconds for the sample case 1. This figure may improve if other optimisation algorithms or improved genetic operations are used. The other area of discussion is the method of grouping. The method of grouping used here is one of the equal distances for each group. An equal probability of parts distribution for grouping could be investigated. In term of the number of groups, the higher the number of groups, the lower the variation in results obtained [4]. However, this increases production costs as well as calculation time.

7 Summary

This paper presented a new approach to selective assembly of components with an arbitrary distribution of part dimensions and it results in zero surplus parts. Based on this approach, selective assembly is carried out in different stages. Each stage consumed a group of parts, taken from the group with the most parts of the same component. Combinatorial optimization of group numbers in each stage was done using genetic algorithm. Comparison of the results of the new method with the one presented by [14] and [15] showed 15 and 20 percent improvement in maximum variations for the first and second cases, respectively, and no surplus parts.

Acknowledgements

This work was carried out in collaboration within Wingquist Laboratory and the Area of Advance Production at Chalmers within the project Smart Assembly 4.0, financed by The Swedish Foundation for Strategic Research. Their support is gratefully acknowledged.

References

[1] Söderberg, R., Wärmefjord, K., Carlson, J. S., and Lindkvist, L., 2017. "Toward a digital twin for real-time geometry assurance in individualized production". *In CIRP Annals,* **66**, pp. 137–140.

[2] Mansor, E., 1961. "Selective assembly-its analysis and applications". *International Journal of Production Research,* **1**(1), pp. 13–24.

[3] Desmond, D., and Setty, C., 1962. "Simplification of selective assembly". *International Journal of Production Research,* **1**(3), pp. 3–18.

[4] Pugh, G. A., 1992. "Selective assembly with components of dissimilar variance". *Computer and Industrial Engineering,* **23**(1–4), pp. 487–491.

[5] Fang, X., and Zhang, Y., 1995. "A new algorithm for minimizing the surplus parts in selective assembly". *Computer and Industrial Engineering,* **28**(2), pp. 341–350.

[6] Fang, X., and Zhang, Y., 1996. "Assuring the matchable degree in selective assembly via a predictive model based on set theory and probability method". *Journal of Manufacturing Science Engineering,* **118**, pp. 252–258.

[7] Chan, K., and Linn, R., 1999. "A grouping method for selective assembly of parts of dissimilar distributions". *Journal of Quality Engineering,* **11**(2), pp. 221–234.

[8] Mease, D. N., Vijayan, N., and Sudjivnto, A., 2004. "Selective assembly in manufacturing: statistical issues and optimal binning strategies". *Technometrics,* **46**(2), pp. 165–175.

[9] Ponnambalam, S., Sankar, S. S., Sriram, S., and Gurumarimuthu, M., 2006. "Parallel populations genetic algorithm for minimizing assembly variation in selective assembly". *Proc. International Conference on Automation Science and Engineering, Shanghai, China*.

[10] Kumar, M., Kannan, S., and Jayabalan, V., 2007. "A new algorithm for minimizing surplus parts in selective assembly by using genetic algorithm". *International Journal of Production Research,* **45**(20), pp. 4793–4822.

[11] Asha, A., Kannan, S., and Jayabalan, V., 2008. "Optimization of clearance variation in selective assembly for components with multiple characteristics". *International Journal of Advanced Manufacturing Technology,* **38**, pp. 1026–1044.

[12] Kannan, S., Asha, A., and Jayabalan, V., 2005. "A new method in selective assembly to minimize clearance variation for a radial assembly using genetic algorithm". *Journal of Quality Engineering,* **17**(4), pp. 595–607.

[13] Kumar, M., Sivasubramanian, R., and Jayabalan, V., 2009. "Particle swarm optimization for minimizing assembly variation in selective assembly". *International Journal of Advanced Manufacturing Technology,* **42**, pp. 793–803.

[14] Kumar, M., Sivasubramanian, R., and Jayabalan, V., 2009. "A new method in selective assembly for components with skewed distributions". *International Journal of Productivity and Quality Management,* **4**, pp. 569–589.

[15] Wang, W., and D Li, C. J., 2009. "Minimizing assembly variation in selective assembly for complex assemblies using genetic algorithm". *Second International Conference of Mechanic Automation and Control Engineering (MACE)), Hohhot, China*.

[16] Raj, M. V., Sankar, S. S., and Ponnambalam, S. G., 2011. "Genetic algorithm to optimize manufacturing system efficiency in batch selective assembly". *Int J Adv Manuf Technol,* **57**, pp. 795–810.

[17] Xu, H. Y., Kuo, S. H., and Tsai, J. W. H., 2014. "A selective assembly strategy to improve the components utilization rate with an application to hard disk drives". *Int J Adv Manuf Technol,* **75**, pp. 247–255.

[18] Bäck, T., 1996. *Evolutionary Algorithms in Theory and Practice*. Oxford University Press.