# UNIVERSITY OF BIRMINGHAM

## University of Birmingham

## Research at Birmingham

# Intelligent selective disassembly planning based on disassemblability characteristics of product components

Parsa, Soran; Saadat, Mozafar

*Document Version*
Publisher's PDF, also known as Version of record

[Link to publication on Research at Birmingham portal](#)

# Intelligent selective disassembly planning based on disassemblability characteristics of product components

Soran Parsa[1] · Mozafar Saadat[1]

## Abstract

Many studies have used different optimisation methods to find a near-optimal solution by optimising the disassembly operations sequence. These studies have used disassembly operation time as the main optimisation parameter, and other parameters such as direction change or tool change are converted to time scale. In order to determine accurate operation time, a product needs to be completely disassembled, noting that the same EOL products can be in a different condition and result in different operation time. In this work, new optimisation parameters based on the disassemblability and components demand are defined. These include Disassembly Handling Index (DHI), Disassembly Operation Index (DOI) and Disassembly Demand Index (DDI). In order to consider the operation time and other costs, Disassembly Cost Index (DCI) is further defined. Genetic algorithm optimisation method was employed to optimise the process sequence. Here, the most demanded components with the easiest disassembly operations are disassembled first without requiring to disassemble the unwanted components and avoid complicated operations. Two case studies were analysed to determine the effectiveness and compatibility of this method. The result shows 13% and 10% improvement in overall disassembly time for the case studies.

**Keywords** Remanufacturing · Disassembly sequence planning · Multi-objective planning · Intelligent optimisation · Genetic algorithm · Disassemblability

## 1 Introduction

High environmental pollution and low renewable material resources are just a few disadvantages of the traditional manufacturing industry [1]. New manufacturing and remanufacturing approaches such as cloud manufacturing and remanufacturing and also smart manufacturing and remanufacturing have shown good ability as the future in the manufacturing industry [2]. These new approaches can employ all new manufacturing resources and abilities [3]. Mishandling of the end-of-life (EOL) products in the traditional manufacturing industry for many years has led to the wasting of the resources and severe environmental problems [4]. Remanufacturing of the EOL products deals with both environmental and economic aspects of the EOL products by considering reusing them in an optimal way [5]. The first

consideration of each remanufacturing process is disassembly which mostly has been carried out manually due to its complexity.

Although it looks like that disassembly process planning is the reverse of the assembly process in manufacturing, there are significant differences especially in the purpose of disassembly process which is to retrieve demand parts. This requires unique and different approaches. Disassembly planning has to deal with necessary and unnecessary parts at the same time, which makes disassembly planning much more complicated than assembly planning in practice [6].

It can be seen that majority of disassembly planning studies can be categorised into three classes: (a) to increase the value of the parts that need to be disposed and (b) to minimise the cost of retrieving some specific parts and (c) efficient full disassembly sequence of the operations of a product [6].

Gupta and Taleb [7] focused on increasing the value of retrieved parts through minimising the disassembly costs in comparison with the value of disassembled parts. In order to optimise the disassembly operations sequences, Lambert introduced a mathematical model using AND/OR graph. Although Lambert's model efficiently finds optimum

✉ Soran Parsa
soran.parsa@gmail.com

[1] Department of Mechanical Engineering, School of Engineering, University of Birmingham, Birmingham, UK

sequences, it does not consider reusable part demand which is applicable in remanufacturing companies [8].

In a remanufacturing field in which disassembly operations sequences and costs have to be optimised, Barba et al. [9] investigated a model using lot-sizing and studied the effect of it on the costs. Veerakamolmal et al. [10] introduced a structure known as the disassembly tree (DT) to demonstrate the relationships and order of disassembling. The main advantage of this method is that the reusable parts can be demanded at any point and make it applicable in the remanufacturing field, although the size of DT increases dramatically by increasing the parts.

The optimization of the disassembly planning especially where the number of parts rises is categorised as NP-hard problems. One of the most effective tools to optimise this kind of problem is the genetic algorithm (GA) that was proposed by Kongar and Gupta [11] to optimise disassembly planning. GA is a heuristic approach to solve the problem quickly and effectively and as it is easy to adapt this technique with rather little mathematics it is getting more and more popular in this field. McGovern et al. [12] studied this method to balance the disassembly lines. Parsa and Saadat [13] investigated automated disassembly using the genetic algorithm and proposed a model for robotic disassembly sequence optimisation. Other optimisation methods such as particle swarm optimisation algorithm were used to solve multi-objective optimisation problems [14].

Although disassembly requests are expanding and disassembly planning techniques are getting more efficient, because of the environmental regulations and the increasing amount of the products require to be disassembled, manual disassembly is getting more difficult and inefficient [15]. Therefore, robotized disassembly processes are in the focus of new researches and are getting more and more essential in the disassembly industry. Studies on robotized disassembly process for electronic products started at the early 1990s. First industrial application in this field was a robot assistant for telephone disassembling [16]. Furthermore, in this area, Torres et al. [17] proposed a robotized disassembly cell that can handle non-destructive disassembly with some degree of automatization. Pomares et al. [18] followed their work and proposed an object-oriented model. This model was required in order to a disassembly process. Gil et al. [19] used co-operative robots to develop a flexible multi-sensorial system in an autonomous disassembly process. Torres et al. [20] followed this work and proposed a task planner using decision trees.

Current demand for more effective disassembly strategies makes full disassembly uneconomical [21–24]. To find an optimal solution for planning problems, selective disassembly which aims to disassemble a product partly to retrieve demanded components is receiving more attraction in current studies. ElSayed et al. [25] investigated an intelligent automated disassembly cell which disassembled products online and selectively. They modelled an online GA (genetic algorithm)

for selective disassembly to optimise the disassembly sequences. A selective disassembly planning method for waste electrical and electronic equipment (WEEE) was proposed by Li et al. They develop a selective disassembly planning method based on particle swarm optimisation with customisable decision-making. They applied this model on WEEE to maximise the economic profit and reduce environmental problems [26].

In this work, first new parameters and objectives for selective disassembly planning are introduced. Then genetic algorithm using new parameters and objectives is employed to find the optimum solution. Finally, the proposed method is tested on an automotive case study to verify its effectiveness and results are discussed.

## 2 Methodology

### 2.1 Introducing new parameters for intelligent selective disassembly planning

A full disassembly plan allows a product to be fully disassembled. However, in realistic industry problems, a product does not require to be fully disassembled and it can be inefficient. Full disassembly plan disassembles all the individual parts of the product regardless of the disassembly process requirement. Selective disassembly planning considers the disassembly aims and objectives and makes it efficient and practical. The total number of required operations in full disassembly plan is $n$, and the number of required operations in selective disassembly is $m$ which $n \geq m$. In selective disassembly, some parts are indicated as target parts to be disassembled and the disassembly process will be continued until the target parts are disassembled.

Since the early 1990s, in order to solve the disassembly sequence problems, researchers have started to use intelligent heuristic methods and the majority of studies are going in this direction. The mainly accepted methodology among the professional is the "graph model + solving method". However, some differences among these methods are noticed, the main idea is the same. Based on the "graph model + solving method", disassembly planning problems can be divided into three sub-problems: (a) product disassembly modelling, (b) sequence generating, (c) disassembly sequence optimization [27]. Most of the researches on disassembly sequence optimization have focused on cost and time of the disassembly operations as the main parameters of disassembly optimization. Using times and costs of disassembly operations as the main optimization parameters leads to inaccurate and unrealistic results. First, the majority of the studies estimate the time of a disassembly operation as measuring actual times of operations required to disassemble product completely. Also, the same EOL products have a different condition which leads

to a different time for the same operation. Secondly, the same disassembly operation in different sequences of operations has different disassembly time and measuring or estimating all this time can be problematic.

Therefore, in this work, Disassembly Handling Index (DHI) and Disassembly Operation Index (DOI) are introduced in order to consider the difficulty and feasibility of the disassembly operations as the main optimization parameters instead of the time and cost. The main advantage of this method is that the disassembly operations can be evaluated easily and quickly without the need to disassemble the EOL product. Also, this evaluation can be done for the same EOL product with different condition individually. Furthermore, Disassembly Demand Index (DDI) is defined to prioritise the demand for each component. This parameter indicates the level of demand for each component in the product. In addition to these parameters, the Disassembly Cost Index (DCI) is defined to include costs and times of disassembly operations. In a sequence of the disassembly process, the operation for component $i$ is shown as $Op(i)$ and the position of the $Op(i)$ in the sequence is shown as $Pos(Op(i))$.

### 2.1.1 Disassembly handling index

The first parameter for a new approach to the intelligent disassembly planning is the Disassembly Handling Index (DHI). Disassembly Handling Index is found by analysing the part's shape, size, weight and orientation, part handling difficulty and where it fits into the product. In order to analyse each part's disassembly handling, a table that categorises the product parts is defined, which is shown in Table 1 [28]. This categorisation is based on the part's geometric characteristics and also the handling of the parts during a disassembly operation. DHI analyses each part of the product using three parameters: A, size of the part; B, weight of the part and, C, shape of the part. As can be seen from Table 1, each characteristics of a part allocated with a score. A higher score means

that the part is more difficult to be handled during a disassembly operation. First, Part Handling Index (PHI) for each part $P(i)$ is defined as below:

$$PHI(P(i)) = A + B + C \qquad (1)$$

where A, B and C can be found using Table 1, for example, for $P(1)$, a difficult to be grasped part (A = 4), light (B = 2) and symmetric (C = 0.8), PHI(P(1)) = 6.8.

Then DHI for a set of operations during the disassembly process is calculated as below:

$$DHI = \sum_{i=1}^{n} PHI(P(i))/Pos(P(i)) \qquad (2)$$

where $Pos(P(i))$ is the position of the part in the disassembly process sequence. For example, if $P(1)$ with PHI(P(1)) is disassembled with the third operation, it needs to be divided by 3. A smaller DHI indicates that the parts that are easier to be handled are disassembled first and ensures that the unnecessary parts with high PHI will not be disassembled. Dividing PHI(P(i)) by Pos(P(i)) allows the algorithm to arrange the disassembly process in a way that the parts with smaller PHI be disassembled earlier. For example, consider component {P1, P2, P3} with PHI {2, 4, 6} respectively. Now, assume two possible disassembly sequences as seq1 = {P2, P1, P3} and seq2 = {P3, P2, P1}. For seq1, DHI = (4/1 + 2/2 + 6/3) = 7 and, for seq2, DHI = (6/1 + 4/2 + 2/3) = 8.67. It can be seen that DHI1 is smaller than DHI2 therefore component with higher PHI, i.e. P3, is at the end of the sequence, while for seq2 with higher DHI, P3 is at the beginning of the sequence.

### 2.1.2 Disassembly operation index

Disassembly Operation Index (DOI) analyses the difficulty of each operation in the disassembly process. In order to calculate the DOI, main disassembly operations were categorised into four categories: A, disassembly force; B, requirement of tools for disassembly; C, accessibility of joints/grooves and,

**Table 1** Disassembly handling categories and scores

| A. Component size<br>One of the following: | | C. Component shape<br>One of the following: | |
|---|---|---|---|
| Easily grasped | 2 | Light and symmetric | 0.8 |
| Moderately difficult to grasp | 3.5 | Light and semi-symmetric | 1.2 |
| Difficult to grasp | 4 | Light and asymmetric | 1.4 |
| B. Component weight<br>One of the following: | | Moderately heavy, symmetric | 2 |
| | | Moderately heavy, semi-symmetric | 2.2 |
| | | Moderately heavy, asymmetric | 2.4 |
| Light (< 7.5 lb) | 2 | Heavy and symmetric | 4.4 |
| Moderately heavy (< 17.5 lb) | 2.5 | Heavy and semi-symmetric | 4.6 |
| Very heavy (< 27.5 lb) | 3 | Heavy and asymmetric | 5 |

D, positioning. Each category has several sub-categories to which a score is given which can be seen in Table 2 [28]. A higher score indicates that an operation is more difficult to be carried out in the disassembly process. Disassembly Operation Index (DOI) for each operation Op($i$) is computed as below:

$$DOI(Op(i)) = A + B + C + D \qquad (3)$$

where A, B, C and D can be found using Table 2. For instance Op(1), a pull operations with hand with moderate effort (A = 1), with common tool requirement (B = 2), on a plane surface (C = 1) and symmetry with high accuracy requirement (D = 5) has DOI(Op(1)) = 9.

Disassembly Process Index (DPI) for all operations in a selective disassembly process is computed as below:

$$DPI = \sum_{i=1}^{n} DOI(OP(i))/Pos(Op(i)) \qquad (4)$$

where Pos(Op($i$)) is the position of the operation in the disassembly process sequence. For example, if Op(1) with DOI(Op(1)) is the third operation in the sequence, it needs to be divided by 3. A smaller DPI is beneficial by which the operations with lower DOI is carried out first and ensures that the unnecessary and difficult operations will not be carried out. Dividing DOI(Op($i$)) by Pos(Op($i$)) allows the algorithm that arranges the disassembly process in a way that the operations with smaller DOI be started earlier.

### 2.1.3 Disassembly demand index

In selective disassembly planning, the goal is to first disassemble specifically targeted components without disassembling the product completely. In this work, Disassembly Demand Index (DDI) is defined to optimise the disassembly process in a way that the most demanded components disassembled first without disassembling unwanted components. Therefore, the level of demand of each component is categorised in four levels, i.e. low, medium, high and very high which can be represented quantitatively by (5, 3, 1, 0) for (low, medium, high and very high) respectively. DDI can be calculated using the following equation:

$$DDI = \sum_{i=1}^{n} LD(i)/Pos(i) \qquad (5)$$

where LD($i$) is the level of demand of component $i$ and Pos($i$) is the position of component $i$ in the disassembly sequence. For example, for a sequence of operations of (3, 1, 4, 2) and LD of (5, 3, 0, 2) respectively, DDI = 5/3 + 3/1 + 0/4 + 2/2 = 5.67.

**Table 2** Disassembly operation categories and scores

| A. Disassembly force One of the following: | | | C. Accessibility One of the following: | | |
|---|---|---|---|---|---|
| Push/pull operations with hand, straight line motion without exertion of pressure | Little effort required | 0.5 | Dimensions | Shallow and broad fastener recesses | 1 |
| | Moderate effort required | 1 | | Deep and narrow fastener recesses | 1.6 |
| | Large amount of effort required | 3 | | Very deep and very narrow fastener recesses | 2 |
| Twisting and push/pull operations with hand, straight line and twisting motion without pressure | Little effort required | 1 | Location | On plane surface | 1 |
| | Moderate effort required | 2 | | On angular surface | 1.6 |
| | Large amount of effort required | 4 | | In a slot | 2 |
| Inter-surface friction and/or wedging, Straight line motion with exertion of pressure | Little effort required | 2.5 | D. Positioning One of the following | | |
| | Moderate effort required | 3 | Level of accuracy required to position the tool- Symmetry | No accuracy required | 1.2 |
| | Large amount of effort required | 5 | | Some accuracy required | 2 |
| Inter-surface friction and/or wedging, Straight line and twisting motions with exertion of pressure | Little effort required | 3 | | High accuracy required | 5 |
| | Moderate effort required | 3.5 | Level of accuracy required to position the tool-Asymmetry | No accuracy required | 1.6 |
| | Large amount of effort required | 5.5 | | Some accuracy required | 2.5 |
| Material stiffness, Twisting motions with pressure exertion | Little effort required | 3 | | High accuracy required | 5.5 |
| | Moderate effort required | 4.5 | | | |
| | Large amount of effort required | 6.5 | | | |
| B. Requirement of tools for disassembly One of the following | | | | | |
| Exertion of force | No tools required | 1 | Exertion of torque | No tools required | 1 |
| | Common tools required | 2 | | Common tools required | 2 |
| | Specialised tools required | 3 | | Specialised tools required | 3 |

**a** BorgWarner turbocharger  **b** Turbocharger CAD model  **c** Turbocharger exploded view
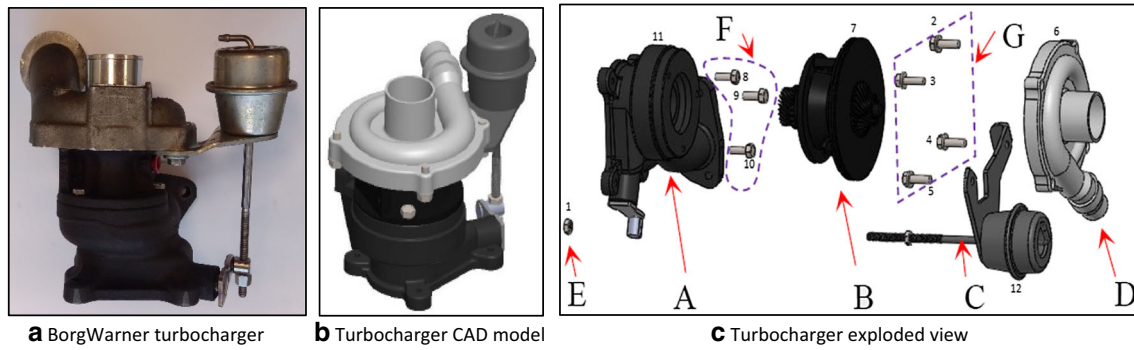
**Fig. 1  a** BorgWarner turbocharger. **b** Turbocharger CAD model. **c** Turbocharger exploded view

### 2.1.4 Disassembly cost index

Disassembly Cost Index (DCI) is defined to analyse each operation's cost to disassemble a specific part of the product in a disassembly process in term of time. It can be used to determine the cost of each operation to allow the operations with lower cost be carried out earlier. DCI can include different operation costs which all converted in a time unit. In this research, DCI parameters are as below:

Operation time:

The basic time requires to disassemble part $P(i)$ using operation $Op(i)$ is shown as $OT(Op(i))$. As the CDI needs to have the same effect as the other parameters in optimisation algorithm, the operation times are normalised as below:

$$OT_N(Op(i)) = \frac{OT(Op(i)) - OT_{min}}{OT_{max} - OT_{min}} \tag{6}$$

where $OT_N(Op(i))$ is the normalised time for each operation and $OT_{max}$ and $OT_{min}$ represent the maximum and minimum operation times of all possible disassembly operations respectively.

Tool change:

The other parameter of DCI is tool change. In this work, tool changing for an operation is penalised in a time unit.

$$TC\Big(Op(i) = \begin{array}{l} 0 \text{ (s) if tool changed} \\ 1 \text{ (s) if tool not changed} \end{array} \tag{7}$$

where $s$ represents a time unit in second.

Disposal cost:

Finally, the last factor is the disposal cost of a part after it is disassembled. If after a disassembly process a part of the product is not reusable and it costs to be disposed of, the operation will be penalised in time unit:

$$DC(Op(i)) = \begin{array}{l} 0 \text{ (s) if part is reusable} \\ 1 \text{ (s) if part is not reusable} \end{array} \tag{8}$$

**Table 3**  The properties and required disassembly tasks for all individual components

| Number | Component | Disassembly task | Disassembly tool | Reusable | Disassembly point (x,y,z) (mm) | Mass (g) | Demand |
|--------|-----------|------------------|------------------|----------|-------------------------------|----------|--------|
| 1 | Nut | Unscrewing | Spanner | No | (175, 75, 0) | 4 | Low |
| 2 | Bolt 1 | Unscrewing | Spanner | No | (50, 100, 100) | 7.5 | Low |
| 3 | Bolt 2 | Unscrewing | Spanner | No | (50, 30, 100) | 7.5 | Low |
| 4 | Bolt 3 | Unscrewing | Spanner | No | (100, 30, 100) | 7.5 | Low |
| 5 | Bolt 4 | Unscrewing | Spanner | No | (100, 100, 100) | 7.5 | Low |
| 6 | Turbine housing | Removing | Hammer | Yes | (80, 80, 105) | 1770 | Very high |
| 7 | Cartridge | Removing | Hammer | Yes | (80, 80, 70) | 1174 | High |
| 8 | Bolt 5 | Unscrewing | Spanner | No | (60, 60, 50) | 5 | Low |
| 9 | Bolt 6 | Unscrewing | Spanner | No | (65, 70, 50) | 5 | Low |
| 10 | Bolt 7 | Unscrewing | Spanner | No | (70, 65, 50) | 5 | Low |
| 11 | Compressor housing | Removing | No tool | Yes | (80, 80, 50) | 480 | Very high |
| 12 | Electronic actuator | Removing | No tool | Yes | (140, 50, 105) | 260 | Medium |
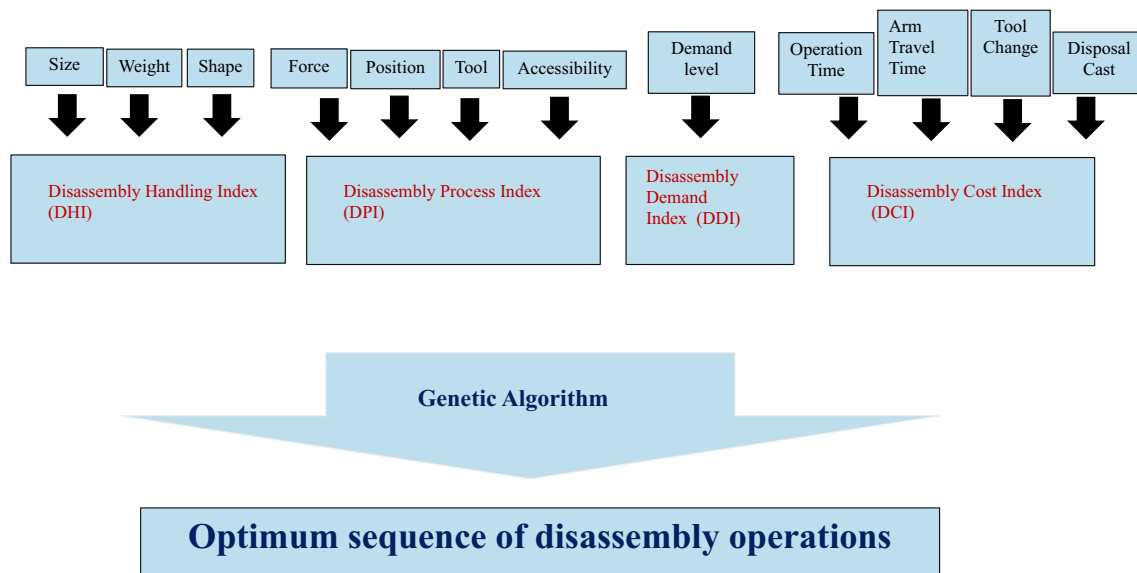
Fig. 2 Flow chart of the proposed model

Now total DCI for disassembly process can be calculated:

$$
\begin{aligned}
\mathrm{DCI} = &\sum_{i=1}^{n} \mathrm{OT_N}(\mathrm{Op}(i))/\mathrm{Pos}(\mathrm{Op}(i)) \\
&+ \sum_{i=1}^{n} \mathrm{TC}(\mathrm{Op}(i))/\mathrm{Pos}(\mathrm{Op}(i)) \\
&+ \sum_{i=1}^{n} \mathrm{DC}(\mathrm{Op}(i))/\mathrm{Pos}(\mathrm{Op}(i))
\end{aligned} \tag{9}
$$

## 2.2 Improved genetic algorithm optimisation

### 2.2.1 Disassembly representation using a hybrid graph model

Graph-based methods such as AND/OR which was introduced by Lambert and Homem are widely used to represent disassembly sequences and precedencies space; however, the number of nodes rises dramatically by the rising number of the components. For example, there will be 16,383 nodes in the AND/OR graph if the product consists of 14 components [29]. In this work, a hybrid graph method is used to represent disassembly sequences and precedencies space [30]. The hybrid graph method describes the topological structure of the product in the form of a graph. It defines the relationships of the constraints between the components of the product using a four-tuple, G = {V; Ef; Efc; Ec}. In this four-tuple nodes set V = {v1, v2, v3, .., vn} defines a minimum disassembly component unit (part or subassembly) where $n$ is the number of units. Ef = {ef1, .. efi} represents the contact constraints between two components and is shown using an undirected solid line. Efc = {efc1, .., efcj} is disassembly contacted constraints and precedence between two components and is shown using a directed solid line. The direction of the line defines the disassembly precedence between the components. Finally, Ec = {ec1, …, eck} which is represented by a directed dashed line defines the disassembly precedence constraints for two non-connected components.

## 2.2.2 Disassembly feasibility and constraint matrices

The relation between product components and disassembly constraints are represented mathematically by two matrices:

1- Components relation matrix detonated by Cr which represents the relationship between product components

$$
C_r = \left\{ cr_{ij} \right\}_{n \times n} = \begin{bmatrix} cr_{11} & cr_{12} & \cdots & cr_{1n} \\ cr_{21} & cr_{22} & \cdots & cr_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ cr_{n1} & cr_{n2} & \cdots & cr_{nn} \end{bmatrix} \tag{10}
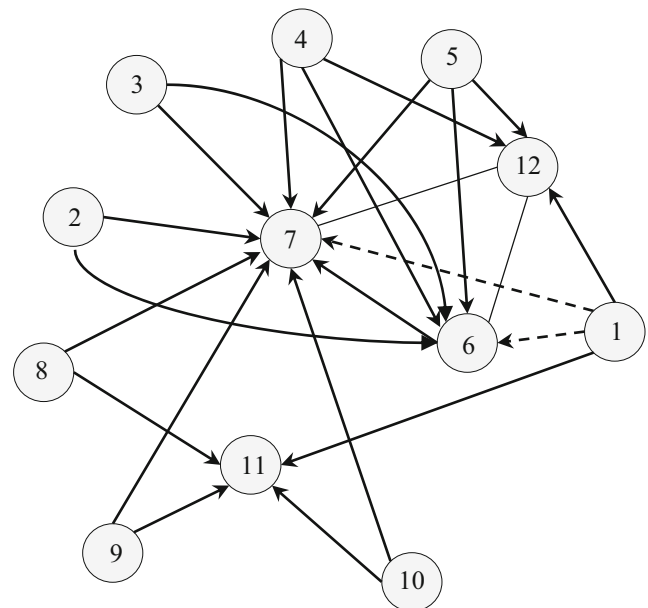$$



Fig. 3 Hybrid graph model for turbocharger

**Table 4** Disassembly handling analysis and disassembly operation analysis for turbocharger

| Number | Component | Disassembly handling analysis | | | Disassembly operation analysis | | | | Demand |
|---|---|---|---|---|---|---|---|---|---|
| | | Size | Weight | Shape | Force | Tools requirement | Accessibility | Positioning | |
| 1 | Nut | 2 | 2 | 0.8 | 4 | 2 | 1 | 2 | 5 |
| 2 | Bolt 1 | 2 | 2 | 0.8 | 4 | 2 | 1 | 2 | 5 |
| 3 | Bolt 2 | 2 | 2 | 0.8 | 4 | 2 | 1 | 2 | 5 |
| 4 | Bolt 3 | 2 | 2 | 0.8 | 4 | 2 | 1 | 2 | 5 |
| 5 | Bolt 4 | 2 | 2 | 0.8 | 4 | 2 | 1 | 2 | 5 |
| 6 | Turbine housing | 4 | 2 | 1.4 | 1 | 3 | 1 | 1.2 | 0 |
| 7 | Cartridge | 3.5 | 2 | 1.2 | 3 | 3 | 1.6 | 1.2 | 1 |
| 8 | Bolt 5 | 2 | 2 | 0.8 | 4 | 2 | 2 | 5 | 5 |
| 9 | Bolt 6 | 2 | 2 | 0.8 | 4 | 2 | 2 | 5 | 5 |
| 10 | Bolt 7 | 2 | 2 | 0.8 | 4 | 2 | 2 | 5 | 5 |
| 11 | Compressor housing | 3.5 | 2 | 1.2 | 1 | 3 | 1 | 1.2 | 0 |
| 12 | Electronic actuator | 2 | 2 | 1.2 | 1 | 3 | 1 | 1.2 | 3 |

where, $i,j = 1, 2, 3, \ldots, n$ and

$$cr_{ij} = \begin{cases} 1 & if \ (v_i, v_j) \in E_f \\ 2 & if \ (v_i, v_j) \ or \ (v_j, v_i) \in E_{fc} \\ 0 & else \end{cases} \quad (11)$$

2- Disassembly constraints matrix, Dc, represents the disassembly constraints mathematically

$$D_c = \left\{dc_{ij}\right\}_{n \times n} = \begin{bmatrix} dc_{11} & dc_{12} & \cdots & dc_{1n} \\ dc_{21} & dc_{22} & \cdots & dc_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ dc_{n1} & dc_{n2} & \cdots & dc_{nn} \end{bmatrix} \quad (12)$$

where $i,j = 1, 2, 3, \ldots, n$ and

$$dc_{ij} = \begin{cases} 1 & if \ (v_i, v_j) \in E_c \\ 2 & if \ (v_i, v_j) \ or \ (v_j, v_i) \in E_{fc} \\ 0 & else \end{cases} \quad (13)$$

This matrix provides geometrical constraints and disassembly precedencies to determine whether a part can be disassembled without restriction. Unit vj can be disassembled if:

$$\sum_{i=1}^{n} dc_{ij} = 0 \ and \ \sum_{i=1}^{n} cr_{ij} > 0 \quad (14)$$

### 2.2.3 GA parameters and operators

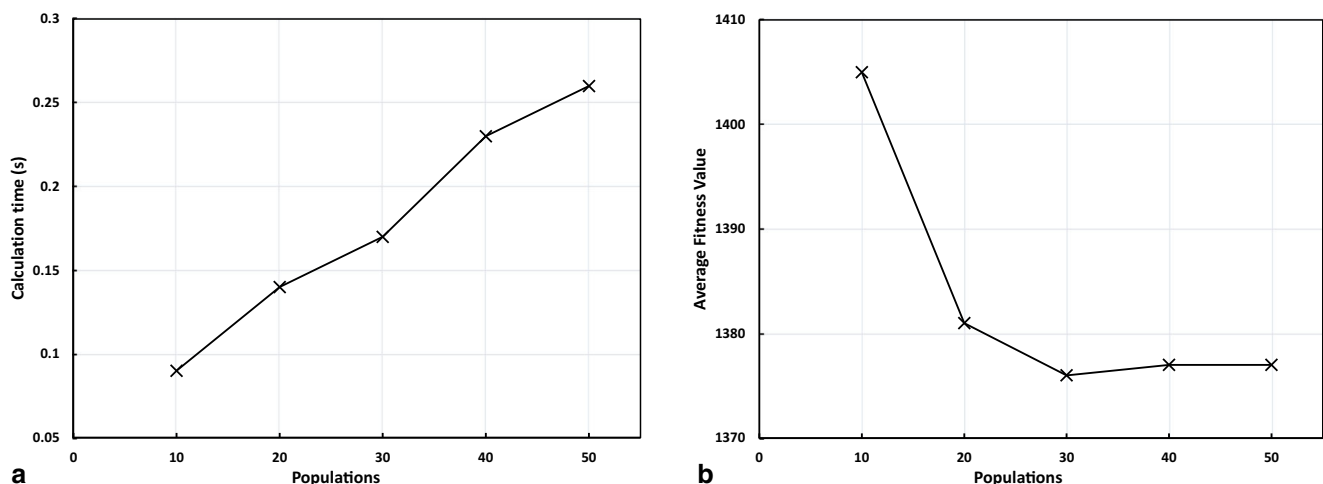Several studies have investigated different optimization methods in order to find optimum disassembly sequences



**Fig. 4** **a** Calculation time under different population. **b** Average fitness values under different population number
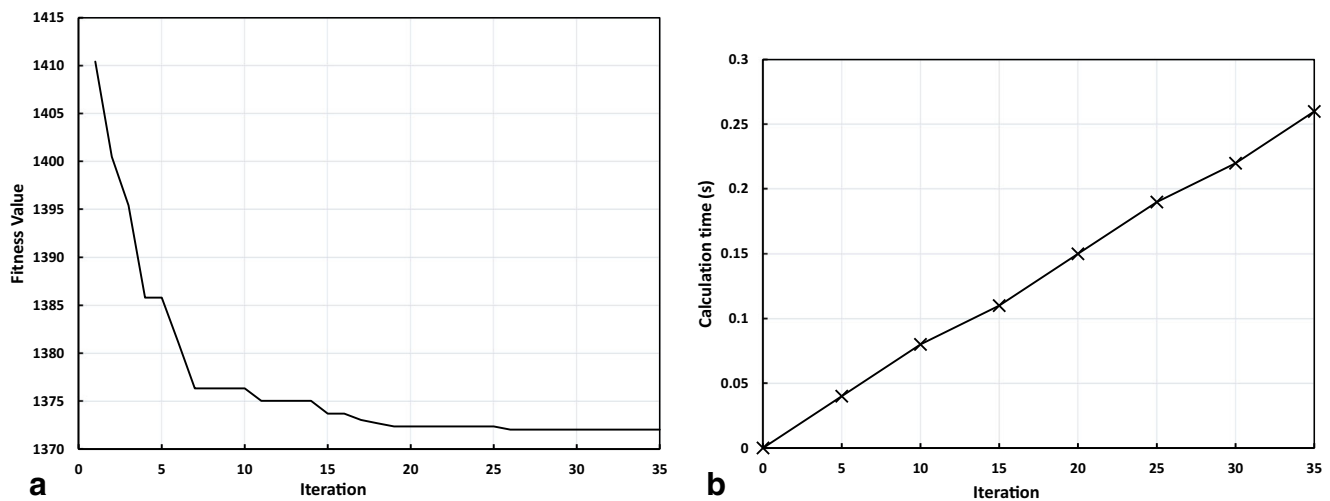
**Fig. 5** **a** Average fitness values under different generations. **b** Running time under different generation number

such as ant colony optimization (ACO), simulated annealing and genetic algorithm (GA). Studies show that GA is the most successful technique and is widely used [27]. Genetic algorithm is a nature-inspired method based on Darwin's natural selection of evolution. It has been used for optimising the constrained and unconstrained problem. GA repeatedly modifies and generates a new population of possible solutions as a new generation. In order to produce a new generation, better solutions of the current population combined together and mutate to produce offspring. The aim is that the characteristics of the better solutions pass to the next generation and population evolve toward an optimum solution. GA has been employed as a powerful optimisation tool in a variety of subjects and researches [31–33]. Basic GA approach was modified by several researchers to improve results. According to Kongar and Gupta who are the pioneer of GA, the method got its idea from the evolution theory and can be simplified based on that [11]. To initiate the algorithm, a set of the possible solutions which was called population is selected in which any member can be encoded as a chromosome. A

chromosome is identified using a combination of several different characters. Different characteristics of product and disassembly can be encoded in each chromosome. Then chromosomes are given scores. These scores are based on a fitness function. The fitness function depends on the disassembly parameters which in this work are DHI, DOI, DDI and DCI. To identify a chromosome with an optimum score, the new population needs to be generated iteratively in each step in which mutation can happen. Also, the crossover may happen in which two different chromosomes can mate and a child is produced. Depending on the disassembly aim and objectives, the disassembly parameters in the objective function can be customised by using different weights. In the following, these parameters are defined:

– Chromosome representation

In order to represent the disassembly solutions and parameters, the disassembly sequences are encoded in chromosomes. A chromosome is a string of genes that occupy specific
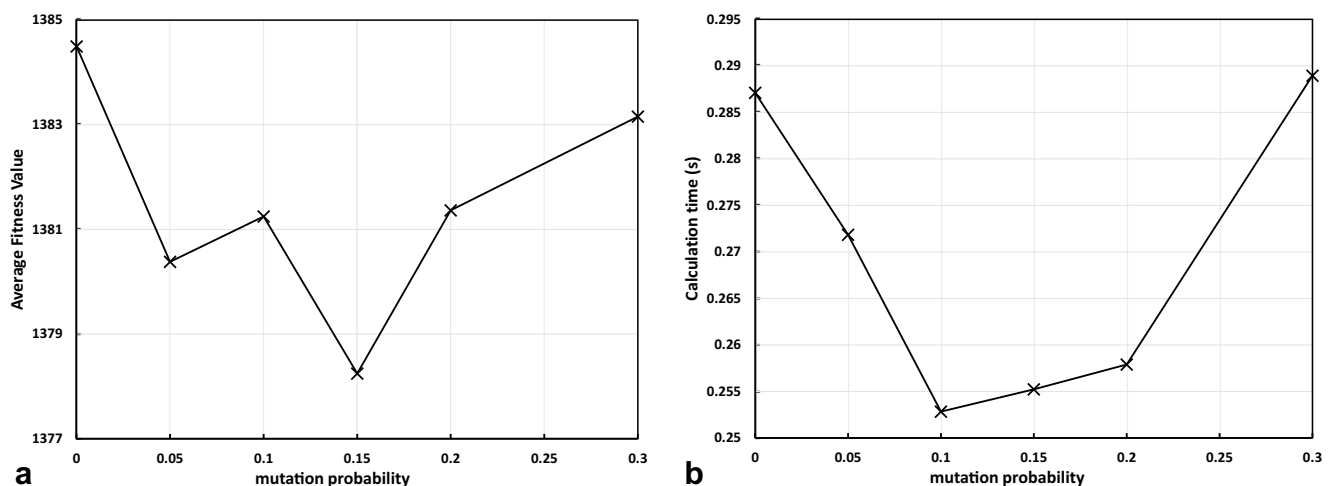


**Fig. 6** **a** Average fitness values under different probability. **b** Running time under different probability

**Table 5** Final optimum disassembly process sequence

| Optimum solution | $F$(ch,gn) |
| --- | --- |
| 1 4 5 12 2 3 6 8 9 10 7 11 | 1362.0333 |

**Table 6** Optimum disassembly process sequence generated by the conventional time-based method

| Optimum solution | $F$(ch,gn) |
| --- | --- |
| 8 9 10 4 3 5 2 1 6 12 11 7 | 1372.0333 |

locations in a chromosome. Parameters of the disassembly operations are encoded in each chromosome as genes. Each chromosome indicates a disassembly sequence and its operations characteristics. A combination of numbers and other characters are used to encode solutions and parameters. For example, if the disassembly process has five variables, they will be codified in a chromosome form composed of five equal section in which each parameters of operations are represented respectively. In this research, each chromosome consists of five string; Sequence, Disassembly Handling Index (DHI), Disassembly Operation Index (DOI), Disassembly Demand Index (DDI) and Disassembly Cost Index (DCI).

– Objective function

An objective function was defined to evaluate the chromosomes and find their fitness level. This function depends on the disassembly process parameters. In this work, these parameters are Disassembly Handling Index (DHI), Disassembly Operation Index (DOI), Disassembly Demand Index (DDI) and Disassembly Cost Index (DCI) which previously were defined. Therefore, the objective function is calculated as follows:

$$f(\text{ch}, \text{gn}) = \alpha \text{DHI} + \beta \text{DOI} + \gamma \text{DDI} + \varepsilon \text{DCI} \quad (15)$$

where $f$(ch, gn) is the fitness value of the ch[th] chromosome in the gn[th] generation and $\alpha$, $\beta$, $\gamma$ and $\varepsilon$ are the user-defined weights for disassembly process parameters which are dependent on the aims and objectives of the disassembly. In this research, the objective of the GA is to minimise the fitness function by minimising DHI, DOI, DDI and DCI of each chromosome.

– Initial population

In order to initiate the optimisation algorithm, a series of randomly selected chromosomes are considered as initial population. The number of chromosomes in the initial population (ncr) is defined depending on the disassembly process characteristics. Higher ncr can result in higher numerical calculation time, while smaller ncr can result in wrong solutions [27]. All constraints and other relationships based on structure graph (feasible solutions) must be satisfied among these chromosomes. In this work, ncr was examined and the optimum ncr subject to optimum numerical calculation time and a better solution was selected.

– Chromosome selection

Roulette wheel technique was employed to select chromosomes of each generation as parents to generate a new population. In the roulette wheel technique, each chromosome is assigned a probability of selection based on its fitness level. Probability of a chromosome to be selected is calculated by Eq. (16).

$$P_i = \frac{1}{f_i \sum_{j=1}^{n} \frac{1}{f_j}} \quad (16)$$

where $f_i$ is the fitness value of individual chromosome in the population. This technique is used to assure that the chromosomes with the lower fitness level are selected. However, some chromosomes with higher fitness value can be selected as parents of a new generation.

– Crossover
– Crossover operator:

In order to generate a new population, crossover operator must be applied to the most efficient chromosomes of the previous generation. In this study, the precedence preservative crossover (PPX) was employed in order to generate new populations. In this methodology, two chromosomes are selected as parents and considered as parent 1 and parent 2. The algorithm starts with generating randomly selected masks which contain 1 and 2 and have the same length as the first section of the chromosomes. These masks impose the order of each child in a new generation. Then an empty offspring is initialized, and crossover operation based on the relative mask is applied and the offspring chromosome is filled with new genes. The mask specifies which parent should be considered to select the gene and after the selected gene will be removed from both parents. This algorithm is repeated until both parents are emptied and a new child is generated. As an example, consider the first two chromosomes of the initial population as parent 1 and parent 2 respectively. Two randomly selected masks which contain 1 and 2 to generate child 1 and child 2 are as follows:
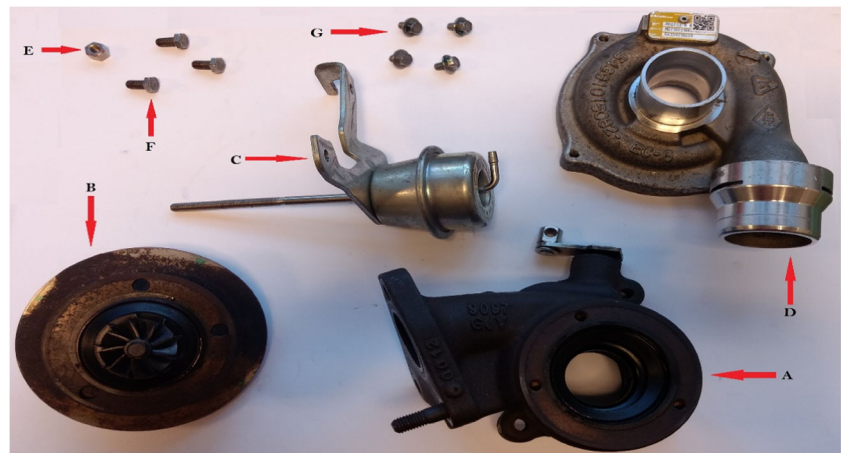
Mask 1: 1 2 1 2 2 2 1
Mask 2: 2 1 1 2 2 1 2
Using (PPX) method to generate new population, the chromosomes of child 1 and child 2 are as follows:
Child 1: 1 7 5 3 6 2 4
Child 2: 7 1 5 3 6 4 2

– Mutation

**Fig. 7** Disassembled components of the turbocharge



After a new population is generated using crossover operation, the chromosomes are subjected to mutation. Mutation is not a dominant operator in genetic algorithm method, and just a small number of the chromosomes are subjected to mutation. A number of chromosomes are selected randomly, and some genes are exchanged. The mutation happens with a defined mutation probability Pm and in such a way that all relations and constraints are preserved. The mutation operation preserves the diversity of the chromosomes and makes sure that the new solution candidates are explored. In this study, the swap technique was used as the mutation operator and different probabilities were investigated. If a chromosome is selected, its genes will be subjected to mutation otherwise its properties will be preserved, and it remains unchanged.

– Termination conditions

Two conditions are set in order to terminate GA calculations. If one of these conditions is met, GA will be terminated. The first condition is that if the number of produced generations exceeds a maximum value GA will be terminated (in this work 40). The second condition is that if the difference between the average fitness of the new generation and the fitness of the previous generation is smaller than a pre-defined number, i.e., the solutions are remaining constant.

**Table 7** Disassembly operations and process times for turbocharger

| Disassembly operations and process times based on the proposed method | | | | | | | Disassembly operations and process times based on the conventional method | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Disassembled part | Trial 1 (s) | Trial 2 (s) | Trial 3 (s) | Trial 4 (s) | Trial 5 (s) | Mean time(s) | Disassembled part | Trial 1 (s) | Trial 2 (s) | Trial 3 (s) | Trial 4 (s) | Trial 5 (s) | Mean time(s) |
| 1-Nut | 8 | 8 | 7 | 9 | 7 | 7.8 | 8-Bolt 5 | 26 | 25 | 30 | 24 | 26 | 26.2 |
| 4-Bolt 3 | 20 | 17 | 18 | 21 | 22 | 19.6 | 9-Bolt 6 | 24 | 22 | 26 | 26 | 23 | 24.2 |
| 5-Bolt 4 | 22 | 27 | 25 | 24 | 20 | 23.6 | 10-Bolt 7 | 26 | 22 | 21 | 30 | 28 | 25.4 |
| 12-Electric Actu. | 7 | 5 | 5 | 8 | 8 | 6.6 | 4-Bolt 3 | 24 | 30 | 29 | 20 | 21 | 24.8 |
| 2-Bolt 1 | 20 | 26 | 23 | 22 | 23 | 22.8 | 3-Bolt 2 | 22 | 23 | 25 | 20 | 25 | 23 |
| 3-Bolt 2 | 24 | 20 | 22 | 23 | 19 | 21.6 | 5-Bolt 4 | 28 | 28 | 35 | 35 | 25 | 30.2 |
| 6-Turbine hous. | 10 | 9 | 8 | 11 | 8 | 9.2 | 2-Bolt 1 | 23 | 20 | 23 | 26 | 24 | 23.2 |
| 8-Bolt 5 | 20 | 21 | 23 | 25 | 20 | 21.8 | 1-Nut | 8 | 8 | 7 | 7 | 9 | 7.8 |
| 9-Bolt 6 | 22 | 18 | 22 | 25 | 20 | 21.4 | 6-Turbine hous. | 10 | 11 | 11 | 10 | 9 | 10.2 |
| 10-Bolt 7 | 23 | 19 | 24 | 26 | 27 | 23.8 | 12-Electric Actu. | 7 | 5 | 9 | 7 | 8 | 7.2 |
| 7-Cartridge | 5 | 6 | 7 | 6 | 8 | 6.4 | 11-Comp. Hous. | 9 | 10 | 9 | 10 | 11 | 9.8 |
| 11-Comp. Hous. | 9 | 7 | 8 | 9 | 8 | 8.2 | 7-Cartridge | 7 | 8 | 8 | 5 | 6 | 6.8 |
| Total process time (s) | 190 | 183 | 192 | 209 | 190 | 192.8 | Total process time (s) | 214 | 212 | 233 | 220 | 215 | 218.8 |
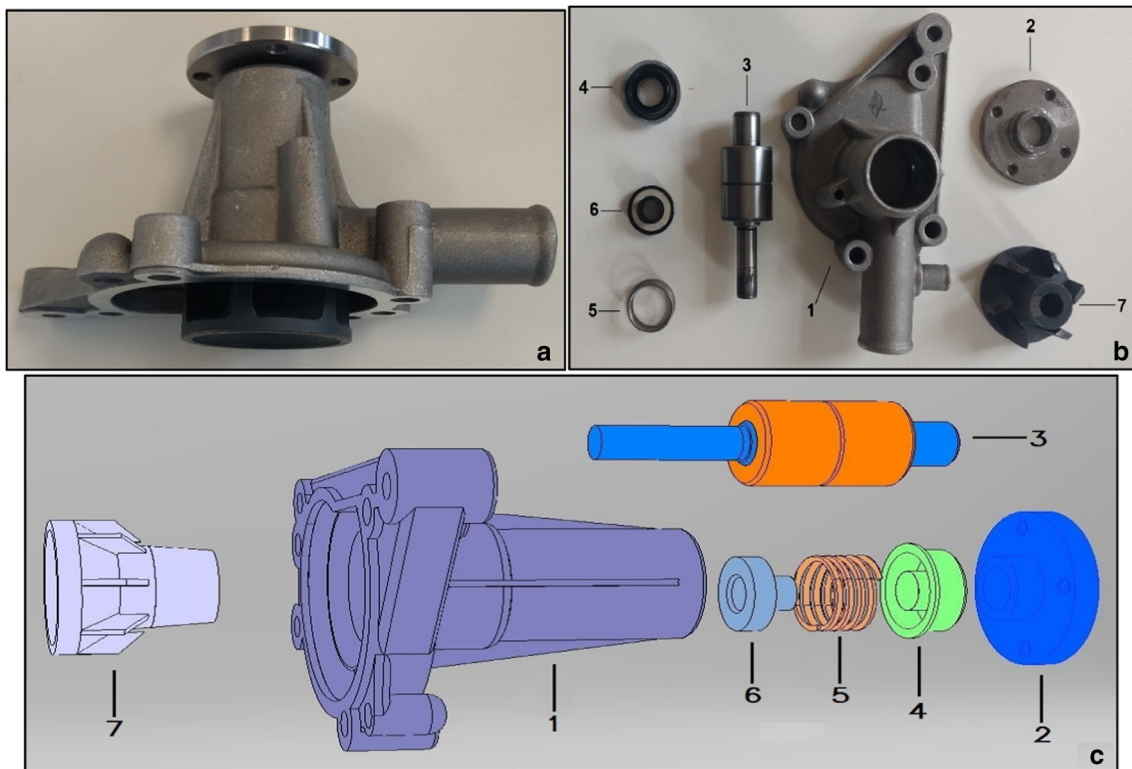
**Fig. 8.** **a** Water pump. **b** Disassembled components. **c** Exploded view

# 3 Case study and performance analysis

## 3.1 Case study 1: turbocharger

### 3.1.1 Background

As the first case study, a turbocharger supplied by Reco Turbo Ltd. is selected to verify the proposed method. It is made by BorgWarner and is used in different cars such as Renault, Nissan and Dacia. Figure 1a, b, c show the turbocharger, CAD model and exploded drawing. As it can be seen, the turbocharger made of 12 individual components which can be categorised into 7 different types: A, B, C, D, E, F and G. Components such as A and C can be disassembled further, but in this work, they assumed as individual components that do not require further

disassembly. The properties and required disassembly tasks for all individual components are shown in Table 3. The main disassembly operations to disassemble the product completely are unscrewing and removing.

It is assumed that the whole disassembly process is carried out manually. The penalty time for a tool change is calculated by Eq. (6). In order to disassemble this product completely, one spanner and one hammer are required. Also, the disposal cost penalty is described by Eq. (7). Figure 2 shows a general flow chart of the proposed model.

### 3.1.2 Performance analysis

Proposed genetic algorithm method was programmed on MATLAB© (version 8.5.0 (R2015a)). Then it was run on a computer with Intel® Core™ i5 6500 CPU at 3.2 GHz and

**Table 8** The properties and required disassembly tasks for all individual components

| Number | Component | Disassembly task | Disassembly tool | Reusable | Disassembly point (x,y,z) (mm) | Mass (g) | Demand |
|---|---|---|---|---|---|---|---|
| 1 | Crankcase cover | Removing | Manual | Yes | (40, 40, 45) | 1021 | Very high |
| 2 | Crankcase | Removing | Manual press | Yes | (23, 23, 5) | 129 | Medium |
| 3 | Bearings and shaft | Pulling out | Manual press | Yes | (30, 30, 45) | 152 | High |
| 4 | Spring chamber | Removing | Manual press | No | (35, 35, 900) | 16 | Medium |
| 5 | Spring | Removing | Manual | No | (30, 30, 90) | 9 | Low |
| 6 | Ring | Pulling out | Manual | No | (35, 35, 95) | 35 | Medium |
| 7 | Rotor | Pulling out | Manual press | Yes | (50, 50, 100) | 250 | Very high |

**Table 9** Disassembly handling analysis and disassembly operation analysis for turbocharger

| Number | Component | Disassembly handling analysis | | | Disassembly operation analysis | | | | Demand |
|---|---|---|---|---|---|---|---|---|---|
| | | Size | Weight | Shape | Force | Tools requirement | Accessibility | Positioning | |
| 1 | Crankcase cover | 2 | 2 | 0.8 | 0.5 | 1 | 1 | 1.2 | 0 |
| 2 | Crankcase | 2 | 2 | 1.2 | 3 | 3 | 1 | 1.2 | 3 |
| 3 | Bearings and shaft | 2 | 2 | 0.8 | 3 | 3 | 2 | 2 | 1 |
| 4 | Spring chamber | 2 | 2 | 0.8 | 3 | 1 | 2 | 1.2 | 3 |
| 5 | Spring | 3.5 | 2 | 1.2 | 1 | 1 | 2 | 2 | 3 |
| 6 | Ring | 2 | 2 | 0.8 | 0.5 | 1 | 2 | 1.2 | 3 |
| 7 | Rotor | 2 | 2 | 0.8 | 3 | 3 | 1 | 2 | 0 |

8.00 GB RAM. In this section, the first convergence capability of the proposed method is discussed and the performance of this method for different iteration and generation is investigated. Then, the optimum sequence of the operations based on this method is obtained and discussed. Finally, in order to validate this method, the turbocharger is disassembled based on the sequences proposed by this method, conventional time-based genetic algorithm and total disassembly time compared.

In order to represent the product disassembly proceedings and constraints mathematically, the hybrid graph method is used. Figure 3 shows the hybrid graph model for turbocharger. It is constructed according to the rules described in Section 2. Based on this model, relation matrix Cr and constraint matrix Dc are as follows:

$$
Cr = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 \\
0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 2 \\
0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 2 \\
0 & 2 & 2 & 2 & 2 & 0 & 2 & 0 & 0 & 0 & 0 & 1 \\
0 & 2 & 2 & 2 & 2 & 2 & 0 & 2 & 2 & 2 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 2 & 0 & 0 \\
2 & 0 & 0 & 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

$$
Dc = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 2 & 2 \\
0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 2 \\
0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 2 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

Table 4 shows the disassembly handling analysis, disassembly operation analysis and demand for the product which are calculated using Tables 1, 2 and 3. For each component, size, weight and shape were considered to find PHI using Eq. (1) and Table 4. Then DHI was calculated using Eq. (2). In order to analyse the disassembly operation for each component, force, requirement of tool, accessibility and positioning were considered. The values for these properties are shown in Table 4 which were used in Eq. (3) to calculate DOI. Then, DPI was calculated using Eq. 4. Finally, based on the demand level in Table 3, the demand for each component was determined

In this work, it is assumed that the basic operation time to disassemble a component is constant and does not depend on the position of the operation in the sequence. Different initial population numbers (ncr) were examined in order to determine the optimum convergence time and more realistic solutions. Figure 4a, b show the calculating time and average fitness values under different population number respectively. It can be seen that higher ncr result in higher calculating time; however, it leads to lower average fitness values. It can be seen after ncr = 30 the average fitness value remains constant; however, the calculating time rises with increasing ncr. Therefore, ncr = 30 was selected as the optimum population number for this
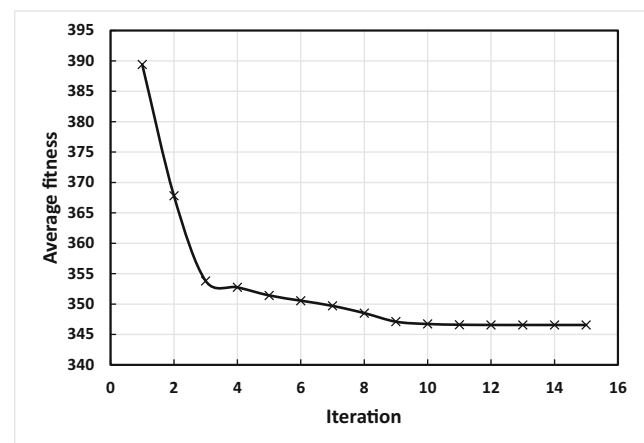


**Fig. 9** Average fitness values under different generations

**Table 10** Final optimum disassembly process sequence

| Optimum solution | $F$(ch,gn) |
| --- | --- |
| 2 1 7 6 3 5 4 | 34.65 |

**Table 11** Optimum sequence generated by the time-based method

| Optimum solution | $F$(ch,gn) |
| --- | --- |
| 2 1 7 3 6 4 5 | 42.22 |

study. Figure 5a, b show the average fitness values and calculating time under a different number of generations.

The effects of different mutation probabilities on the proposed method characteristics were studied. Average fitness value trends and calculation times for different probabilities are shown in Fig. 6a, b respectively. It can be seen that mutation probability has a significant effect on both fitness value and calculating time. Selecting a very small or very high mutation probability leads to higher calculating time and higher average fitness function. Therefore, choosing an optimum mutation probability in order to achieve an optimum solution is essential. Analysing these results show that the mutation probability from 0.1 to 0.15 gives best results, which minimise both calculating time and fitness value.

The optimum disassembly process sequence based on the proposed model and its fitness value is shown in Table 5. This optimised solution was achieved in 0.25 s using the proposed method.

In order to compare the proposed method with conventional methods, the turbocharger was disassembled manually based on the sequences generated by this research and conventional GA method and operations times and the total disassembly process time were measured. To generate a sequence based on the conventional GA, the method proposed by Parsa and Saadat [13] was employed. Based on their method, the main parameter for GA objective function to find an optimum solution is basic operations times which should be estimated to initiate the algorithm. Also, as in this research the method for disassembly is manual, the algorithm used at [13] was modified to remove the automation parameter, i.e., time travel of the robot arm. The sequence generated using this method can be seen in Table 6.

Then, the turbocharger was disassembled 5 times to minimise the error and mean times for operations and total disassembly process calculated. The disassembled turbocharger can be seen in Fig. 7. Also, the turbocharger was disassembled 5 times based on the sequence generated by the conventional GA method and mean times calculated. The measured and calculated times are shown in Table 7.

Although the main advantage of the proposed method in this research is to avoid initial time estimation and generate more realistic sequences of disassembly operations based on the disassemblability of the products, it can be seen that the overall disassembly time was improved by 13%. This improvement can be due to more realistic objective parameters and avoiding estimating operation time which reduce the errors. It should be noticed that the main objective of the conventional time–based method is to minimise the overall

disassembly time; however, the proposed method in this research considers other objectives such as demand and disassembly times alongside the main objective which is disassemblability.

### 3.2 Case study 2: water pump

The second case study is a water pump model GMP187. Figure 8a, b, c show the water pump, exploded drawing and disassembled components of the water pump respectively. The water pump made of 7 sub-assemblies. Table 8 shows the properties and required disassembly tasks for all individual components.

As can be seen, the disassembly operations are carried out manually and just on the tool is required. Furthermore, demand for each component, disassembly tasks and required tool are detailed in Table 8. The same method as the case study 1 is employed to represent the product disassembly proceedings and constraint, therefore, presenting the graph and matrices are not repeated for this case study.

The disassembly handling analysis, disassembly operation analysis and demand for the product were calculated using Tables 1, 2 and 8 which presented in Table 9. It can be seen that components 1 and 7 have the highest demand. It is assumed that the basic operation time to disassemble a component is constant and does not depend on the position of the operation in the sequence.

GA parameters and operators were set up to find an optimum operation sequence based on the DHI, DOI, DDI and DCI. For the case study, ncr = 20, as the component number is lower than the first case study. This helps to reduce computational cost. Analysing the results of the first case study showed that mutation probability from 0.1 to 0.15 archives the best results, therefore, mutation probability for this case study was set at 0.15.

Figure 9 shows average fitness values under a different number of generations. It shows that at generation 10 the algorithm reaches its minimum fitness value and running it further does not improve the results. The optimum disassembly process sequence based on the proposed model and its fitness value is shown in Table 10. This optimised solution was achieved in 0.16 s using the proposed method.

The same method as the first case study was employed to generate a time-based disassembly sequence which is shown in Table 11. Then the water pump was disassembled manually based on the two different disassembly sequence. In order to minimise the error, the water pump was disassembled 5 times based on each sequence and average times were calculated

**Table 12** Disassembly operations and process times for water pump

| Disassembly operations and process times based on the proposed method | | | | | | | Disassembly operations and process times based on the conventional method | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Disassembled part | Trial 1 (s) | Trial 2 (s) | Trial 3 (s) | Trial 4 (s) | Trial 5 (s) | Mean time(s) | Disassembled part | Trial 1 (s) | Trial 2 (s) | Trial 3 (s) | Trial 4 (s) | Trial 5 (s) | Mean time(s) |
| 2-Crankcase | 18 | 19 | 20 | 17 | 17 | 18.2 | 2-Crankcase | 18 | 18 | 16 | 20 | 17 | 17.8 |
| 1-Crankcase cover | 5 | 5 | 6 | 6 | 4 | 5.2 | 1-Crankcase cover | 4 | 5 | 6 | 5 | 6 | 5.2 |
| 7-Rotor | 15 | 14 | 16 | 17 | 14 | 15.2 | 7-Rotor | 12 | 15 | 17 | 16 | 15 | 15 |
| 6- Ring | 5 | 4 | 4 | 5 | 7 | 5 | 3-Bearings and shaft | 14 | 14 | 13 | 15 | 14 | 14 |
| 3-Bearings and shaft | 11 | 13 | 12 | 11 | 9 | 11.2 | 6- Ring | 8 | 8 | 10 | 7 | 7 | 8 |
| 5-Spring | 8 | 8 | 7 | 8 | 7 | 7.6 | 5-Spring | 8 | 9 | 10 | 7 | 10 | 8.8 |
| 4-Spring chamber | 10 | 12 | 12 | 10 | 9 | 10.6 | 4-Spring chamber | 12 | 12 | 11 | 11 | 14 | 12 |
| Total process time (s) | 72 | 75 | 77 | 74 | 67 | 73 | Total process time (s) | 76 | 81 | 83 | 81 | 83 | 80.8 |

which can be seen in Table 12. It can be seen the overall disassembly time was improved by 10% which is slightly less than the first case study. This can be due to lower components for this case study than the first one.

## 4 Conclusion

In this paper, a new method was introduced to solve disassembly sequence planning problems. The majority of studies have focused on time as the main parameter to find an optimum solution and other parameters such as tool change requirement and disposal casts are converted in a time unit. These studies estimate the time required to disassemble a component of a product as it is difficult to determine the time accurately. This is due to the fact that the product should be disassembled completely in order to measure the time. Also, the same EOL products can have different conditions, which cause different disassembly times. Furthermore, the same disassembly operation in a different sequence order can have a different disassembly time. In this work, DHI, DOI and DDI were introduced as optimisation parameters to analyse handling, disassembly difficulty and demand of a product. Additionally, DCI was introduced to consider disassembly time and other disassembly costs. The hybrid graph method was used to represent the mathematical model of the product and disassembly constraints. A genetic algorithm was then employed to search the possible sequences and find a near-optimum solution. Finally, a turbocharger as an industrial product was selected to test and verify the proposed method. The results showed the effectiveness and compatibility of the method. One of the solutions with the minimum fitness value was presented as the optimum sequence for this case study. In

this sequence, all disassembly constraints are met. In order to compare the effectiveness of this method with conventional time-based GA methods, the turbocharger was manually disassembled based on the sequences generated by the proposed method and conventional method. Disassembly operations and process times were measured and compared. The results showed 13% and 10% improvement in disassembly time for case studies 1 and 2 respectively. Further improvement of this method can be obtained by improving the disassemblability categorisation and scoring.

## References

1. Tao F, Cheng JF, Cheng Y, Gu SX, Zheng TY, Yang H (2017) SDMSim: a manufacturing service supply–demand matching simulator under cloud environment. Robot Comput Integr Manuf 45: 34–46. https://doi.org/10.1016/j.rcim.2016.07.001
2. Diallo C, Venkatadri U, Khatab A, Bhakthavatchalam S (2017) State of the art review of quality, reliability and maintenance issues in closed-loop supply chains with remanufacturing. Int J Prod Res 55(5):1277–1296. https://doi.org/10.1080/00207543.2016.1200152
3. Tao F, Zhao DM, Hu YF, Zhou ZD (2008) Resource service composition and its optimal-selection based on particle swarm

optimization in manufacturing grid system. IEEE Transactions on Industrial Informatics 4(4):315–327. https://doi.org/10.1109/TII.2008.2009533

4. Ren YP, Yu DY, Zhang CY, Tian GD, Meng LL, Zhou XQ (2017) An improved gravitational search algorithm for profit-oriented partial disassembly line balancing problem. Int J Prod Res. https://doi.org/10.1080/00207543.2017.1341066

5. Guide VDR (2000) Production planning and control for remanufacturing: industry practice and research needs. J Oper Manag 18(4):467–483. https://doi.org/10.1016/S0272-6963(00)00034-6

6. Jinmo S, Bongju J (2014) A heuristic for disassembly planning in remanufacturing system. Sci World J:1: 1–1:10

7. Gupta M, Taleb N (1994) Scheduling disassembly. Int J Prod Res 32(8):1857–1866

8. Lambert A (2001) Optimum disassembly sequence generation. Environmentally concious manufacturing; 4193: 56–67

9. Barba-Guti'errez Y, Adenso-D'ıaz B et al (2008) Lot-sizing in reverse MRP for scheduling disassembly. Int J Prod Econ 111(2): 741–751

10. Veerakamolmal P. and Gupta M. (1998) Design of an integrated component recovery system. Proceedings of the IEEE International Symposium on Electronics and the Environment; (9): 264–269

11. Kongar E, Gupta M (2006) Disassembly sequencing using genetic algorithm. Int J Adv Manuf Technol 30:497–506

12. McGovern S, Gupta M (2007) Balancing method and genetic algorithm for disassembly line balancing. Eur J Oper Res 179(3):692–708

13. Parsa S, Saadat M (2018) Intelligent planning using genetic algorithm for automated disassembly. Proceeding of 16th International Conference on Manufacturing Research ICMR2018. doi:https://doi.org/10.3233/978-1-61,499-902-7-189

14. Tao Y, Meng K et al (2018) Joint decision-making on automated disassembly system scheme selection and recovery route assignment using multi-objective meta-heuristic algorithm. Int J Prod Res. https://doi.org/10.1080/00207543.2018.1461274

15. Kopacek P, Kopacek B (2006) Intelligent, flexible disassembly. Int J Adv Manuf Technol 30(9):554–560

16. Hohm K, Hofstede M, et al. (2000) Robot assisted disassembly of electronic devices. Proceedings of the 2000 IEEE

17. Torres F, Gil P et al (2004) Automatic PC disassembly for component recovery. Int J Adv Manuf Technol 23:39–46

18. Pomares J, Puente S et al (2004) Virtual disassembly of products based on geometric models. Comput Ind 55:1–14

19. Gil P, Pomares J et al (2007) Flexible multi-sensorial system for automatic disassembly using cooperative robots. International Journal of Computer Integrated Manufacturing (IJCIM) 20:757–772

20. Torres F, Puente S et al (2009) Automatic cooperative disassembly robotic system: task planner to distribute tasks among robots. Control Eng Pract 17:112–121

21. Ziqiang Z, Guohong D et al (2015) Research of partial destructive based selective disassembly sequence planning. The Open Mechanical Engineering Journal 9:605–612

22. Smith S, Hung P-Y (2015) A novel selective parallel disassembly planning method for green design. J Eng Des 26:283–301

23. Kara S, Pornprasitpol P (2006) Selective disassembly sequencing: a methodology for the disassembly of end-of-life products. Ann CIRP 55:1

24. CHUNG C, PENG Q (2006) Evolutionary sequence planning for selective disassembly in de-manufacturing. Int J Comput Integr Manuf 19(3):278–286

25. ElSayed A, Kongar E et al (2012) A robotic-driven disassembly sequence generator for end-of-life electronic products. J Intell Robot Syst 68:43–52

26. Li WD, Xia K et al (2013) Selective disassembly planning for waste electrical and electronic equipment with case studies on liquid crystal displays. Robot Comput Integr Manuf 29:248–260

27. Agrawal D, Phani T et al. (2013) Automated disassembly sequence planning and optimization. Proceedings of the 2013 Industrial and Systems Engineering Research Conference

28. AnoopDesai AM (2003) Evaluation of disassemblability to enable design for disassembly in mass production. Int J Ind Ergon 32:265–281

29. Li JR, Khoo LP, Tor SB (2002) A novel representation scheme for disassembly sequence planning. Int J Adv Manuf Technol 20:621–630

30. Zhang XF, Zhang SY (2010) Product cooperative disassembly sequence planning based on branch-and-bound algorithm. Int J Adv Manuf Technol 51:1139–1147

31. Kheder M, Trigui M, Aifaoui N (2015) Disassembly sequence planning based on a genetic algorithm. Proceeding of the Institution of Mechanical Engineering, Part C: Journal of Mechanical Engineering Science. 229(12):2281–2290

32. Giudice F, Fargione G (2007) Disassembly planning of mechanical systems for service and recovery: a genetic algorithms based approach. J Intell Manuf 18:313–329

33. Tseng H, Chang C et al (2018) A Bloch-based genetic algorithm for disassembly sequence planning. Expert Syst Appl 96:492–505