

Optimizing Disassembly Line Balancing Problem Using Multi-objective ADAM Optimizer Algorithm

**Nadir A.Siddig, Zeqiang Zhang, Abdallah mokhtar,
Ahmed Abualnor**

Article history:

Received August 31, 2021

Received in revised form September 27, 2021

Accepted November 11, 2021

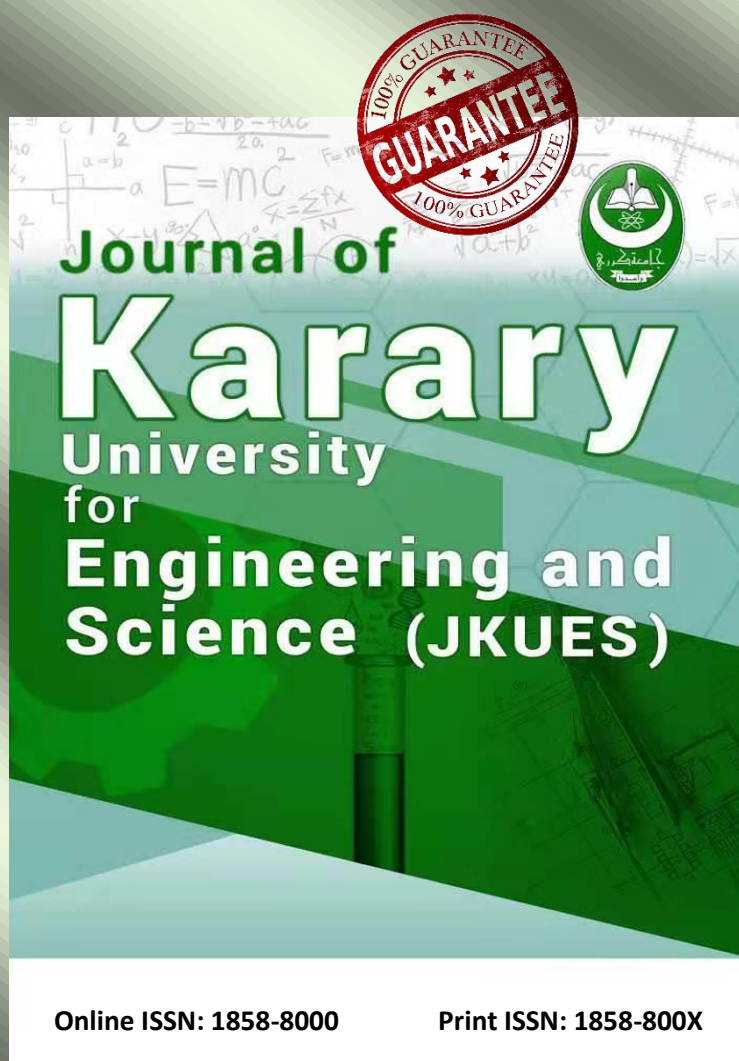
Available online November 22, 2021



**Journal of Karary University for
Engineering and Science (JKUES)**

Online ISSN: 1858-8000

Print ISSN: 1858-800X



Optimizing Disassembly Line Balancing Problem Using Multi-objective ADAM Optimizer Algorithm

Nadir A.Siddig^{*†}, Zeqiang Zhang^{*†}, Abdallah mokhtar[‡], Ahmed Abualnor[§]

^{*}School of Mechanical Engineering, Southwest Jiaotong University, Chengdu, 610031, China.

[†]Technology and Equipment of Rail Transit Operation and Maintenance Key Laboratory of Sichuan Province, Chengdu, China.

[‡]School of Mechanical Engineering, Sudan University of Science and technology, Khartoum, Sudan.

[§]School of Information Science and Technology, Southwest Jiaotong University, International Cooperation Research Center of China.

Abstract

Disassembly for the end of life products considered very important issues nowadays, to keep an environment is clean as much as possible and reduce pollution by reducing industrial waste, the disassembly line is to solve of this problem, but the disassembly line is face more problems such as smooth of work load in workstations, hazard parts in disassembly products, etc. To optimize these problems in disassembly line, we proposed a multi-objective ADAM optimizer algorithm to optimize and balance the objectives. An automatic disassembly line assumed to increase disassembly processes rather than manually. The mathematical model embedded in proposed algorithm simulated in default disassembly line, a pareto set used to detect for non inferior solutions in the algorithms, and the results indicated to improve the productivity of between 400% to 500%, reduced the energy consumption between 50% to 75%, reduced the number of workstations between 80% to 95%, and reduced cycle time between 70% to 85%.

Key words-Disassembly line balancing; Pareto set; ADAM optimizer; Multi-objective ADAM optimizer algorithm.

I. INTRODUCTION

To avoid diseases from touch the products, reducing difficulty of disassembly tasks, and stressful works for workers, it would be necessary to find other methods for disassembly to increase the productivity of disassembly and other objectives. Interdisciplinary research cooperation to solve the problems related to the engineering and science fields. There are many suggestions were studies the disassembly plans to enhance disassembly line works such as a parallel model of balancing a partial disassembly line with stochastic disassembly time, disassembly line estimates include the number of workstations. Gupta et al. [1] studied a variety of factors affecting the efficiency of disassembly lines, and proposed the Disassembly Line Balancing Problem (DLBP) for the first time. They considered to reduce idle time, disassemble high-demand parts as soon as possible, and disassemble hazards as soon as possible. The optimization objectives in the early researches used heuristic methods [2], [3] are the mainstay, such as greedy algorithm [4], 2-opt algorithm [5], a combinatorial optimization method [6], [3]. The principle of the heuristic method is intuitive and simple to operate. It can quickly solve a feasible solution to the problem, but the solution result is single and uncertain, and the quality of the solution completely depends on the heuristic rule. In order to improve the quality of the solution, a mathematical programming method that can obtain the optimal solution of the problem was proposed in [7], [8], but similar to the heuristic algorithm, only single objective is considered, and DLBP has been strictly proved by mathematics to be an NP-complete problem [9], with the increase of the problem scale, the solution space traversed by the mathematical programming method increases exponentially, and it is difficult to obtain

the optimal solution within the effective time. Therfor, the mathematical programming method is not suitable for solving large-scale problems. At present, intelligent algorithms are widely used in solving DLBP due to their good versatility and strong global optimization ability, including genetic algorithm [10], ant colony algorithm [3], [11], simulated annealing algorithm [12] and deep learning algorithm [12], etc. With the deepening of research [13] learned from the assembly line balance problem related to the sequence correlation between assembly tasks [13], and in the defined disassembly model, not only the physical constraints between parts, that is, the disassembly of parts, were considered. Priority relationship, the next step of disassembly can only be carried out after all the immediate tasks are completed; at the same time, it is also considered that there is no mutual influence between the parts of the disassembly priority relationship. Some parts that have not been disassembled will hinder the disassembly of other parts. This dependence on the constraints of the disassembly sequence is different from the constraints of the priority relationship. There is no mandatory requirement for the disassembly sequence of the sequence-related parts, and it usually takes more time to complete the disassembly task, which is called the sequence-related time. The basic DLBP ignores the sequence correlation time to simplify the solution model. In practical applications, the work station time may exceed the tact time, resulting in the production plan not being completed as scheduled. Therfor, the sequence correlation between parts needs to be taken into account in the solution model. The above-mentioned intelligent algorithms all use the dictionary order method or linear weighting a method to convert multi-objective problems into single-objective problems, and multiple decision-making objectives are often in

conflict with each other and their importance is difficult to determine. This method of processing is difficult to ensure that all objectives are consistent. Recently, robots bring to help in this field, robotic workstations considered more a capability to disassemble the end of life products (EOL) [14].

The remainder of this paper is organized as follows: [Section II](#) define the disassembly line problem (DLBP), [Section III](#) describes the problem methodology and mathematical model of the multi-objective ADAM optimizer algorithm. [Section V](#) presents computational and comparison results and discussion. [Section VI](#) gives a summary and conclusion of research.

II. STATEMENT OF THE PROBLEM

Waste products disassembly is a very important activity to keep an environment as well, now days there are many plans to improve disassembly lines, but many of them dependence on manual disassembly lines, disassembly lines have more advantage's side by side more disadvantages, an increase of the end of life (EOL) products have led to an accumulation of these issues which made the matter of disassembly hazard from direct touch parts when workers who work in these fields. To avoid diseases from touch the products, reducing difficulty of disassembly tasks, and stressful works for workers, it would be necessary to find other methods for disassembly to increase the productivity of disassembly and other objectives. Interdisciplinary research cooperation to solve the problems related to the engineering and science fields. There are many suggestions were studies the disassembly plans to enhance the disassembly line works such as a parallel model of balancing a partial disassembly line with stochastic disassembly time, disassembly line estimates include the number of workstations, smooth workload, dismantling profit, and with algorithm for genetic simulated annealing was proposed to optimize the model [15]. Another purposed a memetic algorithm to solve two side mixed model disassembly line method [16], also a sequence dependent partial disassembly Line Balancing problem model was proposed and a Multi-objective mathematical model was created to simultaneously reduce the number of workstations, total idle time, idle balance index, and the number of tools, with a pareto discrete Hummingbird algorithm to solve disassembly problem [17], a mathematically multi-objective model used to solve parallel line with two products need disassembly [18], [19], [20], meta heuristics became very importance technique to solve sophisticated problems [21], therefor disassembly line balancing problems (DLBP), such as idle time, sooth workload, the energy consumption, and so on, all these problems need to solve by meta heuristic to obtain optimum solutions.

III. METHODOLOGY

[Fig. 1](#) the traditional disassembly line is generally manual, the disassembly products are too low productivity, many of workers at workstations, and the workload is not smooth, our methodology refers to convert the manual disassembly line to the automatic disassembly line (see [Fig. 6](#)), to improve the proposed disassembly line, and balancing the problem of disassembly processes by using proposed algorithm, we assumed some details to design meta heuristic algorithm.

A. Assumptions

The waste products will face many uncertain factors during the disassembly process, such as differences in the structure of different types of scrap products, uncertainty in the quantity and quality of parts, and connection damage in some parts. In order to simplify the model, the following assumptions are made:

- 1) The waste product is a single species;
- 2) The quantity of scrap products is sufficient, which is suitable for assembly line production method;
- 3) The number of parts of the scrapped product is complete, and there is no loss or modification, etc.;
- 4) The parts are connected reliably and can be disassembled normally;
- 5) Use standardized disassembly operations;
- 6) Ignore the uncertain factors in the disassembly process, and the disassembly operation time is fixed;
- 7) Disassemble all parts of the waste product;
- 8) Each part corresponds to a disassembly task.
- 9) Each disassembly task can not be divided.

According of above assumptions for fuzzy disassembly products, we proposed new meta heuristics algorithm to determine optimum solutions of the disassembly line balancing problem.

B. Proposed algorithm

In this paper, the multi-objective ADAM optimizer is artificial intelligence algorithm to solve four objectives, the objectives mean the problems face the disassembly line, to balance these problems need to build up the population for the disassembly line based to the Multi-Objective ADAM Optimizer Algorithm (MOAOA) to balance solutions to evaluate of the disassembly solutions, this population would be typically suitable for solving discrete and continue optimization problems, and the decoding method is discredited as shown in an instant example contains 9 parts [Fig. 3](#), and the technique which is used to determine the sequence of disassembly parts at the line [Fig. 2](#), this example needs disassembling under four objectives with limit constraints such as sequence in workstations and other constrains in equations (21-32), and flowchart in [Fig. 7](#), to obtain an initial population for the allocation of the workstations and the assignment of works should be determined at the flowing:

Step 1: select the current assignment in order of the disassembly sequence from left to right, if the work is not completed, perform step 2; otherwise, step 5.

Step 2: calculate the actual disassembly time for the current assignment, including job time, adjustment time, and sequential correlation time. The calculation of adjustment time is based on the disassembly direction of each adjacent work in the disassembly sequence, and determines whether each work needs to adjust the operation, if the disassembly direction of a work is inconsistent with that of the previous work, the determination of the sequence priority should be combined with the disassembly sequence and the sequence priority matrix, and for two disassembly works with sequence correlation, determine the location of each work in the disassembly sequence a work has an impact on the execution of another work.

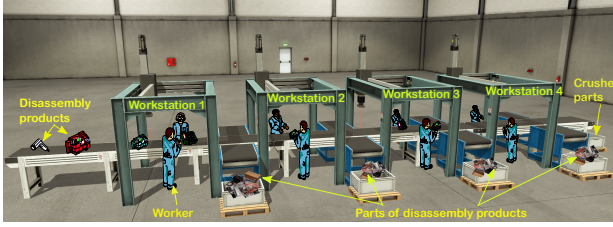
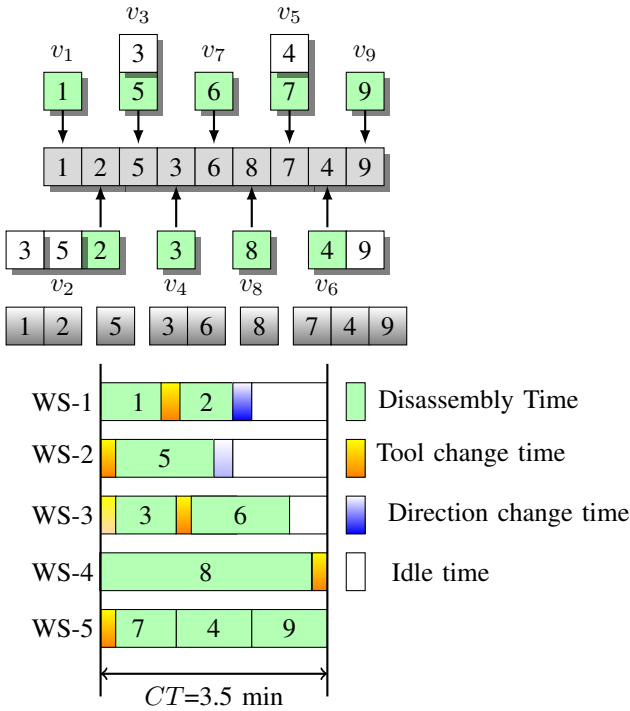


Fig. 1. The manual disassembly workstation with human efforts

Fig. 2. A schematic diagram of the priority selection of parts to remove from EOL product according to multi-objective functions. CT means the cycle time.

Step 3: determine whether the actual disassembly time of the current assignment work is not exactly the disperse time of the momentum workstation, provided that this is true, appoint the work to the current workstation, and the excess disseminate time of the workstation will be reduced. Perform step 1; otherwise, step 4.

Step 4: open a new workstation, reset the remaining allowable time to idle time, and perform step 3.

Step 5: output a location of all tasks in the disassembly sequence, and assigned workstations, the workstation time and whether or not to perform the adjustment operation according to four objectives, and the decoding program go ahead to the end.

Suppose N components need removing sequence, each component is defined as a disassembly work, according to the physical constraints between the components, the priority relation diagram of the components is drawn and the disassembly priority relation matrix is determined.

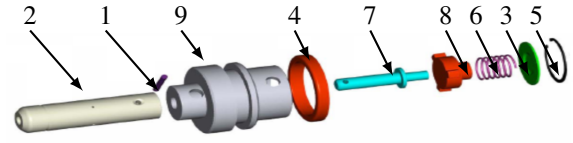


Fig. 3. Removed parts from the end of life product contain 9 parts according to scheme in Fig. 2.

$$A_{m \times n} = \begin{matrix} & c_{j=1} & c_{j=2} & \cdots & c_{j=n} \\ \begin{matrix} c_{i=1} \\ c_{i=2} \\ \vdots \\ c_{i=n} \end{matrix} & \begin{pmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,n} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m,1} & b_{m,2} & \cdots & b_{m,n} \end{pmatrix} \end{matrix}$$

where $A_{m \times n}$ the disassembly priority matrix ; c_i, c_j components numbered i, j ; n is a overall components; b_{ij} is the disassembly preference of components c_i and components c_j , if component c_i takes precedence over component c_j , $b_{ij} = 1$, otherwise $c_j, b_{ij} = 0$.

The number of the initial population in MOAOA have a certain effect on the convergence treatment and execution of the algorithm, in order to expand the search scope a group of one dimensional arrays with n length is randomly generated on the basis of satisfying the constraint of priority relationship, the n times are repeated and the initial population of n is generated, among of them satisfying the constraint of priority relationship, ensures that the random solution is in the feasible region besides the difference between these random solutions and a global search, steps for constructing a random solution are as follows:

Step 1: according to the priority relationship diagram, algorithm establishes the priority relationship matrix, initialize the disassembly sequence position K_1 .

Step 2: find out the work j that makes it work and add it to the candidate work V_k (see Fig.2).

Step 3: determine whether the candidate work set holds, if so, perform step 6; otherwise, perform step 4.

Step 4: randomly select a work J_1 in V_K to relegate to the K situation of the disassembly succession.

Step 5: rebuild the precedence matrix, constrain the assigned work j_1 , so that the work j_1 will not be a candidate again, and the work j_1 allocation does not impose constraints on other works. Therefore, release the constraints on the unassigned works from the assigned work J_1 , make K_1 , K_1 , and perform step 2.

Step 6: By outputting the disassembly sequence, the random solution constructor will finish.

C. Notation

This list contains mathematical model notations used in this paper.

f_1	Improve productivity
f_2	Minimize the energy consumption
f_3	Minimize the number of workstations
f_4	Minimize idle time
ω_p	The number of parallel lines
δ	An average coefficient of inter sectional imposition of productivity losses
k_{tec}	A parameter describing the degree of technical processing of the disassembly line
EC	Energy Consumption
E_i	Input energy for the disassembly operations
E_{le}	An energy loss for the disassembly operations
E_{lm}	An energy friction loss of mechanical transmission system
E_{dis}	An energy for the disassembly
E_{idle}	An energy used at idle time
E_{loss1}	An energy loss
t_c	Cycle time
t_k	The idle time
$t_{o.t}$	Lost time outside the disassembly cycle time due to technical reasons
t_n	An optional period of time to study the functionality of the disassembling equipment
θ_n	The average time required to reset and calibrate the disassembling equipment
θ_p	The total time required for disassembling equipment to perform its functions
u	Random variable an independent
ϵ	A decreasing ratio of energy cost
E_{tcij}	The energy consumption to change tool
P_{tc_i}	The power need to change tools

D. The pareto solution set

The sub-objective of the multi-objective problem may conflict with each other and the dimensions are inconsistent, which makes it impossible to evaluate the solution. In this paper, the pareto solution set is used to deal with multi-objective problems. the optimization objectives can be simplified to two objectives (see Fig.4), any two solutions X_1 and X_2 satisfy an equation 1, which means that the objective value f_1 of X_1 is better than the objective value f_2 of X_2 . It is said that X_1 dominates X_2 , the non-inferior solutions contained in the pareto optimal solution set are not dominated by any solution in the feasible decision space, and the set of objective functions is called the pareto optimal frontier (see Fig.5).

$$\begin{cases} \forall i \in \{1, 2, \dots, M\}, f_i(X_1) \leq f_i(X_2) \\ \exists j \in \{1, 2, \dots, M\}, f_j(X_1) < f_j(X_2) \end{cases} \quad (1)$$

IV. THE MATHEMATICAL MODEL OF THE PROPOSED MULTI-OBJECTIVE PROBLEM

A. The modeling of multi-objectives to balance the disassembly Line

1) *The model of productivity:* According to the method based on multi-objectives containing four objectives, a main

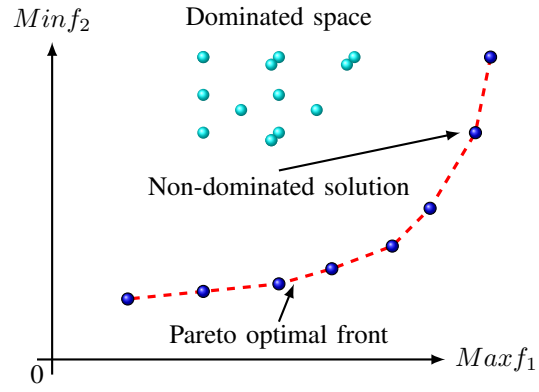


Fig. 4. Pareto-optimal front containing multi-objectives problem and the solution area combination between max and min

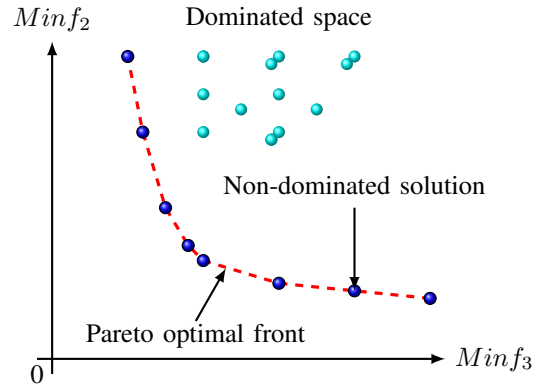


Fig. 5. Pareto-optimal front containing multi-objectives problem and the solution area combination between minimum objectives

objective Q_{max} aim to maximize the productivity of the automated disassembly line, to obtain this objective Q_{max} based to MOAOA relating to this following equation:

$$\text{maximize } f_1 = \sum_{w=1}^w Q \quad (2)$$

The automated disassembly line productivity Q (part/min) [19], and [22] with Limited buffer capacity, maximum productivity must be related to maximum buffer, and the location of the buffer near the workstation is considered in the serial line

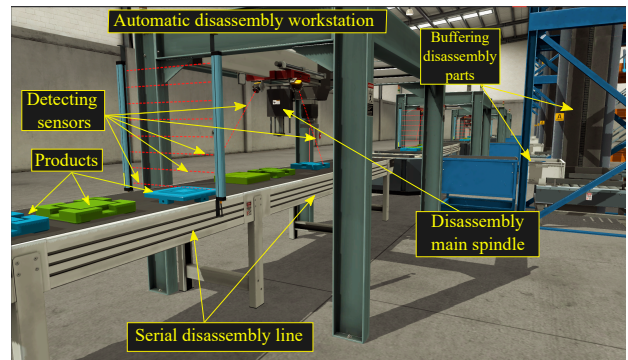


Fig. 6. the automatic disassembly workstation in the disassembly line without complete human efforts

[23] and, [24] refer to relationship between productivity and capacity of buffer, thus the main objective here is considering this relationship.

$$Q = \frac{w_p}{\frac{T_d}{w_q} + T_a + \frac{w_p w_q \sum T_i}{n} \left[1 + \frac{2\delta(1-\delta^{n-1})}{1-\delta} \right]} \quad (3)$$

Where w_p refers to workstation in parallel line, here $w_p = 1$; T_d components the disassembly in cycle time; T_a adjusting time (component/min); T_i idle time loss at single workstation (min/component); w_q is the number of sequential workstations in MOAOA.

The relationship between productivity Q_c and the number of workstations W is shown in the equation (4) with a special cases, where productivity increases at the beginning with the increase in the number of workstations and after that it decreases sharply, there is an ideal value for the number of workstations W , which achieve the greatest productivity Q_{max} , this ideal value can be obtained by taking the first derivative and equating it with zero in the equation (5)

$$Q_c = f(W) \quad (4)$$

$$\frac{dQ_w}{dw} = 0 \quad (5)$$

In the event of an unequal distribution of the works in workstations and having regular breakdowns and stops, productivity will always be less than theoretical value Q_w in equation (6).

$$Q_w = \frac{1}{\frac{t_{po}}{W} + t_{x.x} + \sum ci + Wt_c} \quad (6)$$

Where t_{po} is a time of removal parts, $t_{x.x}$ is time of non removal parts, $\sum ci$ total of losses outside of the disassembly tools, and t_c is losses of outside the disassembly tools for one mechanism on the disassembly line.

B. The model of Disassembling Energy Consumption of Workstations

1) *Energy consumption objective* : The second objective assignment refers to reduce the consumption of the energy would be used in the disassembly workstation, this objective aim to minimizing the power as long as the automatic workstation are working, and the model as following:

$$\text{minimize } f_2 = \sum_{w=1}^w EC(s) \quad (7)$$

Workstations and other equipment would be attaching in the disassembly line need enough energy to work, these energy can be determine in following:

1) Model of Start up:

In the stand by periods, disassembly machines getting ready to do the work, so that the value of the removing energy E_{dis} is zero; when the remover arm rise from a starting to require force, the energy of stepper motors or servo motors to obtain the consumption energy uses for the moving parts to remove the disassembly components. Therefore, the EC model as follows:

$$E_i(t) = E_{le}(t) + E_{lm}(t) + dE_m/dt + dE_k/dt \quad (8)$$

E_i is the input energy. E_{le} is the energy loss ; E_{lm} is the loss energy of shifting parts; E_m is the energy consumption of motor's magnetic field. E_k is the consumption energy of the mechanical system .

2) Energy consumption model for idle time

In an idle time, a disassembly EC E_{dis} , and the work done by the disassembly machines is constant, so the energy of the stepper motors (a magnetic field) ,and the utilization energy of the movable parts are almost consistent. Accordingly, the energy utilization of the idle time is as per following:

$$E_{idle}(n) = E_{loss1}(n) + E_{loss2}(n) \quad (9)$$

E_{idle} an energy in an idle time i_{th} of workstation, E_{loss1} is the energy loss of i_{th} workstation, and E_{loss2} is the loss energy of the mechanical transmission system in the main driving system

3) Disassembling Energy Consumption

In the disassembling periods, E_{dis} the mechanical transmission system of the workstation, the consumption energy of the moving tool arm is change heavily, and the amount of dE_m/dt and dE_k/dt are approximately equal to 0. So, the energy consumption is measured as follows:

$$E_i(t) = E_{le}(t) + E_{lm}(t) + E_{dis}(t) \quad (10)$$

4) Energy Consumption of Disassembling Tool Changing

As evidenced by the disassembly technique of adjacent the disassembly works arrangement, it may be important to change a tool, normal tool change time can be obtained by using fact strategy. Energy usage to change tool can be planned as follows:

$$E_{tcij} = P_{tc_i} \cdot (t_{std} + \delta pos_j \cdot t_{tc0}) \quad (11)$$

where P_{tc_i} is power of the tool change at the i_{th} workstation, t_{std} is the steady time of working tool, δpos_j is the moved the tool from the $(k-1)th$ part to that of the k -th part, and t_{tc0} is a time of the tool holder.

C. Model of workstations

The third objective assignment is determine to reduce the workstations, in actual terms as work for the limitation number of workstations

$$\text{Minimize } f_3 = \sum_{k=1}^k W_{optimum} \quad (12)$$

$$W_{min} \leq W_{optimum} \leq W_{max} \quad (13)$$

$$W_{optimum} = \sqrt{\frac{t_{po}}{t_c}} = \sqrt{\frac{t_{po}}{\omega_c \tau_B}} \quad (14)$$

D. Model of Idling time

Total losses time is one of the most important factors affecting productivity and work ability of the workstation. In this objective is assign to minimum of total idling time of all workstations, taking into consideration balance of workstations:

$$\min f_4 = \sum_{k=1}^k (W_k \cdot t_c - t_k)^2. \quad (15)$$

The total idle time can be calculated:

$$\sum t_k = \sum t_c + \sum t_{o.t} + \sum t_n \quad (16)$$

Total losses time is one of the most important factors affecting in productivity and work ability of the workstation, and this time is calculated by equation (17)

$$\sum t_k = \frac{\sum \theta_n}{z} = \frac{\theta_n}{\theta_p} \cdot T \quad (17)$$

Similarly can be computed another losses times by equations (18),(19),(20) respectively.

$$\sum t_c = \frac{\sum \theta_c}{\theta_p} \cdot T \quad (18)$$

$$\sum t_{o.t} = \frac{\sum \theta_{o.t}}{\theta_p} \cdot T \quad (19)$$

$$\sum t_n = \frac{\sum \theta_n}{\theta_p} \cdot T \quad (20)$$

E. Constraints:

Equations(21-24) ensures that all parts are disassembled and each the disassembly work is located in different positions in the disassembly sequence.

$$\sum_{i=1}^n \sum_{k=1}^n w_{ik} = \frac{(n-1) \cdot n}{2} \quad (21)$$

$$w_{ik} = 0, \forall i \quad (22)$$

$$w_{ik} + w_{ki} = 1, \forall i, \forall k, i \neq k \quad (23)$$

$$\sum_{j=1}^n w_{ik} \neq \sum_{k=1}^n w_{i'k}, \forall i, \neq i', i \neq i' \quad (24)$$

Equation (25) for the precedence constraint of the disassembly sequence, the position of a work in the disassembly sequence is less than the position of the work before the disassembly sequence was completed.

$$\sum_{k=1}^n w_{ik} > \sum_{k=1}^n w_{uk}, \forall i, \forall u \in \{u | a_{ui} = 1\} \quad (25)$$

$$p_i = \sum_{k=1}^n w_{ik} + 1, \forall i \quad (26)$$

$$T_j = \sum_{k=1}^n \left(t_k + \sum_{k=1}^n (s_{ik} \times w_{ik} + t_s \times z_k) \times v_{jk} \right), \forall j \quad (27)$$

$$T_j \leq T_c, \forall j \quad (28)$$

$$\sum_{j=1}^j v_{ij} = 1, \forall i \quad (29)$$

$$v_{ij} \leq \sum_{e=1}^j v_{1e}, \forall j, \forall i, \forall l \in \{v | p_l < p_i\} \quad (30)$$

$$z_j = 1, \forall j \in \{j | p_j \neq 1, r_j \neq r_v, v \in \{v | p_v = p_j - 1\}\} \quad (31)$$

$$v_{ik}, w_{ij}, z_j, S_k \in \{0, 1\} \forall i, \forall j, \forall k \quad (32)$$

The equation (26) determines the position of every work in the disassembly Progress. The equation (27) determine the working time of every workstation, including the work time of disassembling work. The equation (28) is a losses time constraint, indicating that working time doesn't surpass the idle time. The equation (29) is a work allocation constraint to ensure that all works are allocated, furthermore, the disassembly work must be relegated to a workstation. The equation (30) is the priority relationship constraint of work allocation, which indicates that the pre task in the disassembly sequence is assigned to the workstation first. The equation (31) determine whether the disassembly work is adjusted or not, only if the work is adjusted. The latter work needs to perform the adjustment operation before the disassembly operation begins and change the placement position of the obselete. The equation (32) defines the 0-1 variables of v_{ik} , w_{ij} , Z_j , and W_k .

to increase of labor productivity constitutes the main criterion to choose the optimal structural possibility for the production line, which allows to calculate the increase in the productivity of the line, change total costs and investment expenses, by reducing the number of workers on the line and the work period. In order to correctly determine the number of sectors, it is necessary to perform a comparison of the parameters of automated line and manual line consisting of the same workstations. It is possible to obtain sufficient accuracy during the calculation taking into account the following assumptions:

- 1) The workstations in the automated line are the same, loss times on all workstations are equal.
- 2) The storage and feeding operations between the processes in the manual line allow to avoid the mutual relationship between the workstations, while in the automated line with the rigid line, the failure of any work station leads to the suspension of the rest of the work stations.
- 3) The costs of the automation line are not related to the number of workstations. The rate of work productivity λ increase in the flow line compared to the automated line is determined by the equation (33)

$$\lambda = \frac{Q_2}{Q_1} = \epsilon \varphi \frac{K_{tec} + N_y(m_c + 1)}{k_{tec}\sigma + N_y(m_c\delta_\varphi + \frac{1}{\epsilon})} \quad (33)$$

Where Q_1 work productivity achieved by equipment and technology of the first type, and Q_2 work productivity achieved by equipment and technology of the second type, φ growth of workstations productivity, k_{tec} is a parameter describing the degree of technical processing of the disassembly line, N_y investment period by years, σ is the variation of the disassembly tools cost, ϵ is

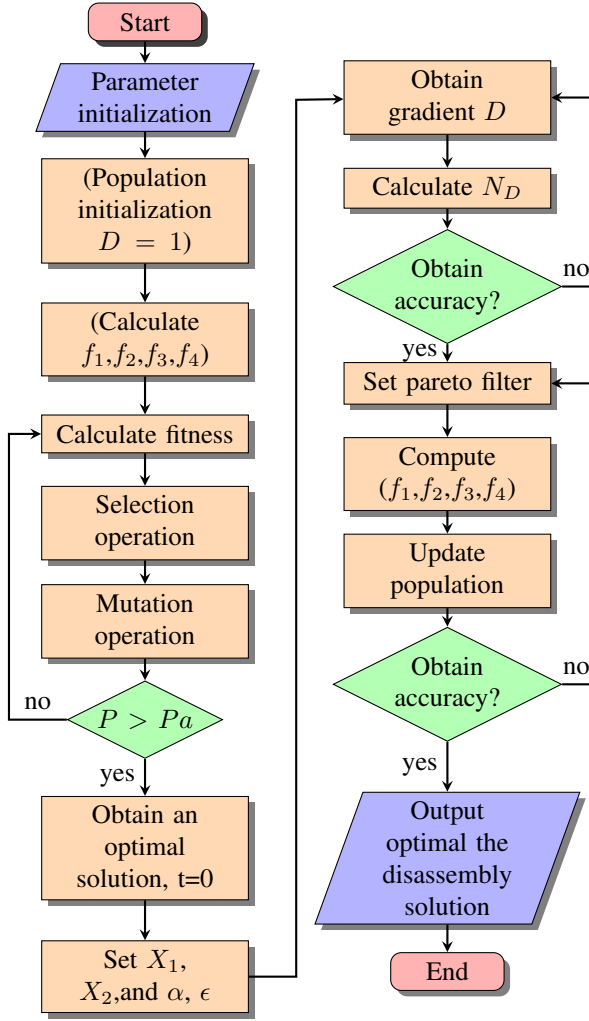


Fig. 7. Pareto multi-objective ADAM optimizer flow chart algorithm was used in this paper to find optimum solutions

decreasing ratio of energy cost, δ_{φ} is the variation of the disassembly energy for one product, m_c is energy cost factor.

The multi-objective ADAM optimizer algorithm is improved by joining to the pareto set to develop searching in solutions area, and to compute the present proposed method is shown in Fig. 7 and can be summarize in the following detail:

Step 1: set algorithm parameters, population size N , maximum iteration number Max_{gen} , probability Pa .

Step 2: process of constructing a random solution according to generates an initialized population with a scale of N .

Step 3: process of initialize the algorithm iteration counter $iteration = 1$.

Step 4: calculate the objectives assignment f_1, f_2, f_3 , and f_4 of the individual in the population, select the non-bad set by using the definition of the pareto optimal solution set.

Step 5: keep the non-bad set in the data set and update the bad solution according to MOAOA operation.

Step 6: generate random probability P to simulate the discovery probability of MOAOA, determine whether the random

probability P is greater than Pa , if it is established, carry out local search and update the neighbor location according to the operation of MOAOA; otherwise, directly execute step 7.

Step 7: calculate the objective assignment value f_1, f_2, f_3 , and f_4 , filter the non bad set, and update D in the data set.

Step 8: compute total non-bad sets of N_D : if $N_D > N_0$, create population of non-bad sets according to the formula, keep N_0 not the worst solutions with large total overcrowded distance in D , otherwise perform step 9 directly.

Step 9: update the population: recalculate the number of non-subordinate sets of data sets, calculate the total clustering distance of not the worst sets, choose N which is not the worst solution in the larger total cluster distance as a new population; otherwise, according to the random solution construction process.

Step 10: taking a decision whether the algorithm iteration counter $iteration$ reaches maximum number of iterations Max_{gen} , and if it is reaches iteration is established, let $iteration = iteration + 1$ and go to step 4; otherwise, outputting the non bad solution in the external data set to obtain an optimal disassembly scheme, and the algorithm is terminated.

V. RESULTS AND DISCUSSION

A. Numerical results

Our algorithm is developed and implemented in a computer with Microsoft windows 10. Intel Core (R) and (TM) i710510 CPU 2.30 GHz and 16 GB RAM (see TABLE II), the code verified in matlab R2019a, and python 3 in anaconda navigator, and the disassembly line productivity was simulated in FACTORY I/O. Considering the performance of the proposed algorithm and the other nine previous algorithms to compare and investigate of better solutions. The algorithm parameters mainly include the population size N , maximum number of iterations Max_{gen} , and the discovery probability Pa . The proposed MOAOA algorithm can also solve other DLBP without considering the adjustment of operating time. Therefore, in order to verify the performance of the proposed MOAOA algorithm, the test examples is shown in Fig. 8, and detailed in TABLE I, a sensing valve for Railway Truck, it consists from 23 parts, the direction of the disassembly, expressed in integers: $+x \rightarrow +1, -x \rightarrow -1, +y \rightarrow +2, -y \rightarrow -2, +z \rightarrow +3, -z \rightarrow -3$, and that refer to the disassembly directions. Aiming at the shortcomings of the previous the disassembly line balancing problem models, which only determines the task distribution, and cannot express the automatic disassembly, and does not consider the productivity caused by the change of the disassembly production and the energy consumption. The proposed MOAOA algorithm can also solve DLBP without considering the direction of parts during the disassembly processes. Other meta heuristics algorithms such as Artificial Bee Algorithm (ABC) [25], Ant Colony Optimization (ACO) [13], Genetic Algorithm (GA) [26], Particle Swarm Optimization (PSO) [27], RFD (River Formation Dynamics, RFD) [28], Simulated Annealing (SA) [29], Tabu Search (TS) [30], variable neighborhood search (VNS) [31], genetic algorithm with a variable neighborhood search (VNSGA) [32]. The

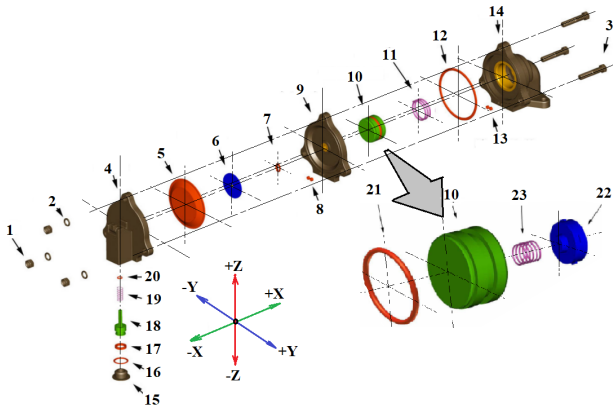


Fig. 8. The disassembly product example used to confirm proposed algorithm

algorithms above assume the importance of four objectives and uses the dictionary order method to optimize each objective in DLBP. This processing method has the following problems:

- 1) Different decision makers may have inconsistent perceptions of the importance of each objective;
- 2) The essence is still single-objective optimization, and only a single solution can be obtained once the algorithm runs.

The pareto filter for find non inferior solutions mentioned in the literature [33] to deal with multi-objective problems. It optimizes each objective at the same time without considering the importance of four objects. At the same time, it uses an improved crowding distance mechanism to solve the overflow problem of non-inferior solutions. Especially the problem of screening the boundary solutions of sub-objectives. Once the algorithm runs, multiple non-inferior solutions with different focuses can be obtained, providing decision-makers with multiple options.

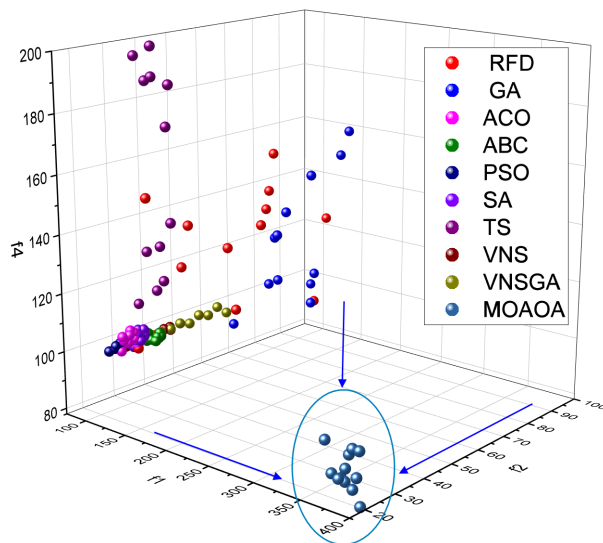


Fig. 9. The pareto optimal set to filter the solutions after balance objectives and compare the objectives 10 algorithms, MOAOA investigated the superior objectives.

TABLE I
THE PARTS SPECIFICATIONS OF THE DISASSEMBLY PRODUCT EXAMPLE A SENSING VALVE FOR RAILWAY TRUCK

SN	Name	Quantity	Directions
1.	Nut	3	-1
2.	Washer	3	-1
3.	Bolt	3	1
4.	Cap	1	-1
5.	Plate	1	-1
6.	Rod	1	-1
7.	Seal	1	-1
8.	gasket	2	-1
9.	Int. body	1	-1
10.	Valve body	1	1
11.	Spring	1	-1
12.	O seal	1	-1
13.	Gasket	2	-1
14.	Back body	1	1
15.	Back cover	1	-3
16.	O seal	1	-3
17.	Y seal	1	-3
18.	Piston	1	-3
19.	Spring	1	3
20.	Screw	1	3
21.	O seal	1	1
22.	Gland	1	-1
23.	Valve spring	1	1

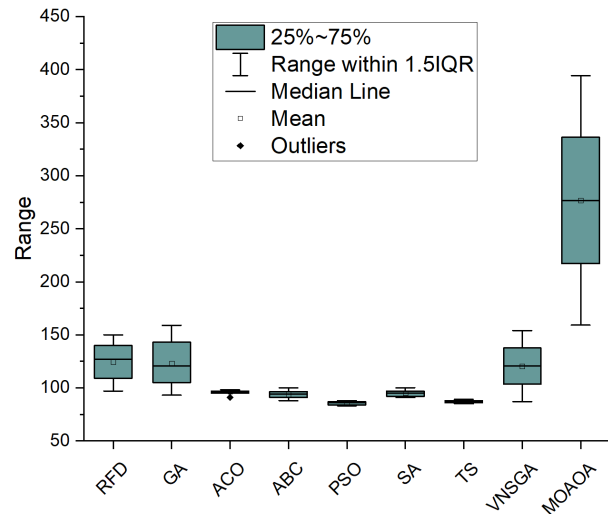


Fig. 10. The comparison between 9 algorithms, and MOAOA algorithm to show a range of the first objective (f_1) increasing productivity in the manual and the automatic disassembly lines.

B. Comparison results

TABLE II the analysis shows that MOAOA obtain maximum $f_1 = 394.28$ part/min, minimum $f_2 = 20.28$ kJ, minimum $f_3 = 3$ workstations need to complete the disassembly processes in the automatic disassembly line, minimum cycle time $f_4 = 78.09$ sec, and the work time of MOAOA in the computer 0.30 sec

TABLE III The sensitivity analysis was performed with the productivity in RFD, GA, ACO, ABC, PSO, SA, TS, VNS algorithms slopes are not significantly, because there are dispersion in the population solutions, and MOAOA Slope is

TABLE II
COMPARISON BETWEEN NINE PREVIOUS META HEURISTICS ALGORITHMS, AND PROPOSED ALGORITHMS (MOAOA)

Source	Algorithm	f_1	f_2	f_3	f_4	Time/s	Computer configuration	Program environment
literature [28]	RFD	97.00	33	5	96.05	3.62	Intel Core2 1.79GHz	MATLAB
literature [26]	GA	93.00	67	5	91.76	5.3	Intel Core2 1.79GHz	MATLAB
literature [13]	ACO	91.00	29	5	96.05	5.36	Intel Core2 1.79GHz	MATLAB
literature [25]	ABC	88.00	38	6	96.05	2.4	Intel Core2 1.79GHz	MATLAB
literature [27]	PSO	83.00	27	5	96.05	0.81	Intel Core2 1.79GHz	MATLAB
literature [29]	SA	91.00	33	5	96.05	-	Intel Core2 1.79GHz	MATLAB
literature [30]	TS	95.00	34	5	111.43	0.87	Intel Core2 1.79GHz	MATLAB
literature [31]	VNS	85.00	32	5	96.05	0.83	Intel Core i7-3770 3.40GHz	MATLAB
literature [32]	VNSGA	87.00	30	5	96.05	0.65	Intel Xeon E5-2650 2GHz	C++
This paper	MOAOA	394.28	20.28	3	78.09	0.30	Intel Core (R) i710510 CPU 2.30 GHz	MATLAB & python

TABLE III
THE COMPARISON OF THE DESCRIPTIVE STATISTICAL OF NON INFERIOR SOLUTIONS IN 10 ALGORITHMS TO DETERMINE THE SIGNIFICANCE OF ALGORITHMS

Algorithms	Sum of Squares	Mean Square	F Value	Prob > F	Intercept		Slope		Statistics Adj. R-Square
					Value	Standard Error	Value	Standard Error	
RFD	485.1405	485.14047	1.71328	0.1951	129.07726	4.12671	-0.13608	0.10397	0.01053
GA	449.0934	449.09341	1.08723	0.30089	127.78183	4.98415	-0.13093	0.12557	0.0013
ACO	1.6674	1.6674	0.99636	0.32184	96.01888	0.31725	0.00798	0.00799	-5.43E-05
ABC	0.49175	0.49175	0.04078	0.84058	93.74759	0.85155	0.00433	0.02145	-0.01452
PSO	5.09945	5.09945	1.67295	0.20037	85.1216	0.42816	0.01395	0.01079	0.00994
SA	7.67845	7.67845	0.92995	0.3384	94.33582	0.70468	0.01712	0.01775	-0.00105
TS	0.40497	0.40497	0.29106	0.59136	87.37094	0.28927	-0.00393	0.00729	-0.01069
VNS	3.02486	3.02486	2.9427	0.09096	87.1734	0.24864	0.01075	0.00626	0.02818
VNSGA	26197	26197	-	-	86	0	1	0	1
MOAOA	322749.7	322749.66	3.69E+30	0	397.79	7.26E-14	-3.51	1.83E-15	1

TABLE IV
WITH 100 NON INFERIOR SOLUTIONS COMPARED THE 10 ALGORITHMS FOR f_1

Algorithms	N	Mean	SD	Min	Max
"RFD"	100	124.3824	16.91685	97	150
"GA"	100	123.2647	20.33717	93	159
"ACO"	100	96.29412	1.2936	91	98
"ABC"	100	93.89706	3.44743	88	100
"PSO"	100	85.60294	1.75465	83	88
"SA"	100	94.92647	2.87198	91	100
"TS"	100	87.23529	1.17331	85	89
"VNS"	100	87.54412	1.02846	85	89
"VNSGA"	100	120.5	19.77372	87	154
"MOAOA"	100	276.695	69.40576	159.11	394.28

significantly, and the different from zero is too low, standard error was scaled with square root of reduced Chi-Square to measure superior of algorithms, which resulting to the improve of f_1 maximize productivity in the automatic disassembly line rather than the manual disassembly line, and the total trend in this study indicating to the productivity is increase in MOAOA algorithms and the Automatic Disassembly Line Balancing Problem (ADLPB).

The descriptive statistics in TABLE IV record the mean of $N = 100$ non inferior solutions population, standard deviation (SD), minimum (Min), maximum (Max) obtained by the 10 algorithms, the numbers the MOAOA productivity significantly refers to high mean, SD, min, and max when

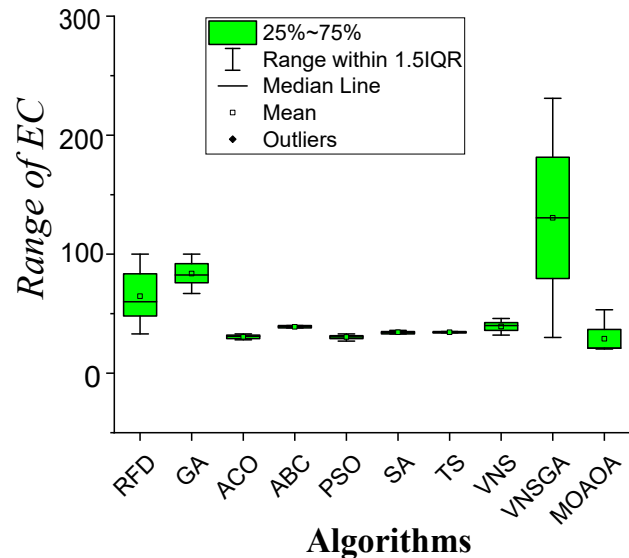


Fig. 11. The comparison between 9 algorithms, and MOAOA algorithm to show a range of the second objective (f_2) reducing the energy consumption.

compared with RFD, GA, ACO, ABC, PSO, SA, TS, VNS algorithms.

Fig. 9 show the non inferior solutions of all algorithms which filtered by the pareto optimum solutions set, the non inferior solutions of MOAOA indicates to superior solutions rather than other algorithms, MOAOA population solution obtained to maximum f_1 , minimum f_2 , and minimum f_4 .

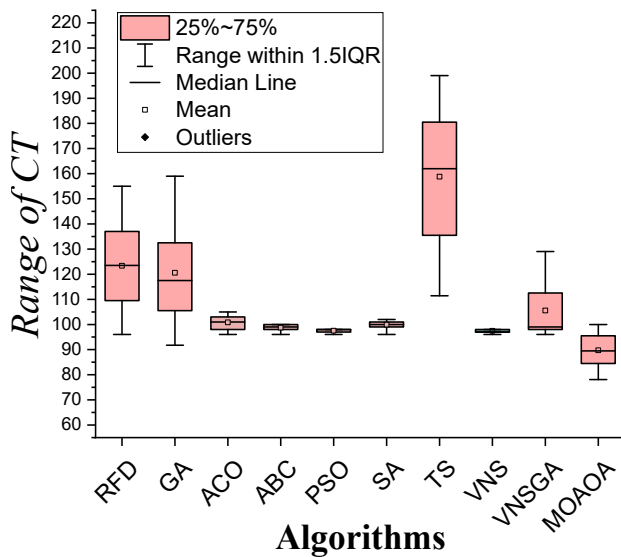


Fig. 12. The comparison between 9 algorithms, and MOAOA algorithm to determine the fourth objective (f_4) is the decreasing cycle time.

Fig. 10 show the range of f_1 between MOAOA and other algorithms, MOAOA investigated the wide range of solutions with high productivity f_1 , and mean and median in the mid of solutions population.

In addition, Fig. 11, of 10 algorithms record the minimum (Min) range of EC in MOAOA rather than other algorithms, but not wide range, mean and median of solutions population are not in the middle of box plot, and Fig. 12 the box plot of f_4 cycle time solutions population refers to minimum range of the disassembly processes time

VI. CONCLUSION

In this paper, we took advantage of artificial intelligence algorithms and we developed an new algorithm named multi-objective ADAM optimizer. This algorithm is established to optimize the productivity, energy consumption, the number of workstations, and cycle time, to balance all these objective functions it was necessary to assume that the disassembly line is automated. The descriptive statistics of the MOAOA algorithm, and the automatic disassembly line indicate the probability of converging to the true pareto frontier from 400 to 500% increase productivity, decrease the energy consumption between 50% to 75%, decrease the number of workstations between 50% to 70%, and decrease cycle time between 70% to 85%. Through the above comparative analysis, we can see that the MOAOA algorithm proposed in this paper is superior to the above nine algorithms. At the same time, compared with the deficiency of transforming the multi-objective problem. The results show that the proposed algorithm and model can provide guidance for decision makers to construct remanufacturing disassembly lines. All this leads to the protection of the environment in which we live from harmful pollution to the health of society.

ACKNOWLEDGMENTS

National Science Foundation of China supported this research by [Number's grant 51675450, 51205328]; Youth Es-

tablishment for Humanities and Social Sciences Ministry of Education of China [Number's grant 18YJC630255]; and the Sichuan Science and Technology Program [Number's grant 2019YFG0285].

REFERENCES

- [1] S. Gupta and A. Gungor, "Product recovery using a disassembly line: challenges and solution," in *Proceedings of the 2001 IEEE International Symposium on Electronics and the Environment. 2001 IEEE ISEE (Cat. No.01CH37190)*. IEEE. [Online]. Available: <https://doi.org/10.1109%2Fisee.2001.924499>
- [2] S. M. Gupta, E. Erbis, and S. M. McGovern, "Disassembly sequencing problem: a case study of a cell phone," S. M. Gupta, Ed., Philadelphia, PA, Dec. 2004, pp. 43–52. [Online]. Available: <http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=851852>
- [3] S. M. McGovern and S. M. Gupta, "Combinatorial optimization analysis of the unary NP-complete disassembly line balancing problem," *International Journal of Production Research*, vol. 45, no. 18-19, pp. 4485–4511, Sep. 2007. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/00207540701476281>
- [4] S. McGovern and S. Gupta, "Greedy algorithm for disassembly line scheduling," in *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme - System Security and Assurance (Cat. No.03CH37483)*, vol. 2. Washington, DC, USA: IEEE, 2003, pp. 1737–1744. [Online]. Available: <http://ieeexplore.ieee.org/document/1244663/>
- [5] S. M. McGovern and S. M. Gupta, "2-opt heuristic for the disassembly line balancing problem," S. M. Gupta, Ed., Providence, RI, Feb. 2004, pp. 71–84. [Online]. Available: <http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=833581>
- [6] —, "Combinatorial optimization methods for disassembly line balancing," S. M. Gupta, Ed., Philadelphia, PA, Dec. 2004, pp. 53–66. [Online]. Available: <http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=851853>
- [7] F. T. Altekin, L. Kandiller, and N. E. Ozdemirel, "Profit-oriented disassembly-line balancing," *International Journal of Production Research*, vol. 46, no. 10, pp. 2675–2693, May 2008. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/00207540601137207>
- [8] M. L. Bentaha, O. Battaa, and A. Dolgui, "Chance Constrained Programming Model for Stochastic ProfitOriented Disassembly Line Balancing in the Presence of Hazardous Parts," in *Advances in Production Management Systems. Sustainable Production and Service Supply Chains*, V. Prabh, M. Taisch, and D. Kiritsis, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, vol. 414, pp. 103–110, series Title: IFIP Advances in Information and Communication Technology. [Online]. Available: http://link.springer.com/10.1007/978-3-642-41266-0_13
- [9] S. McGovern and S. M. Gupta, *The disassembly line: balancing and modeling*. New York: McGraw-Hill, 2011, oCLC: 692373799. [Online]. Available: <http://accessengineeringlibrary.com/browse/disassembly-line-balancing-and-modeling>
- [10] S. M. McGovern and S. M. Gupta, "A balancing method and genetic algorithm for disassembly line balancing," *European Journal of Operational Research*, vol. 179, no. 3, pp. 692–708, Jun. 2007. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0377221705007307>
- [11] —, "Ant colony optimization for disassembly sequencing with multiple objectives," *The International Journal of Advanced Manufacturing Technology*, vol. 30, no. 5-6, pp. 481–496, Sep. 2006. [Online]. Available: <http://link.springer.com/10.1007/s00170-005-0037-6>
- [12] C. B. Kalayci, S. M. Gupta, and K. Nakashima, "A Simulated Annealing Algorithm for Balancing a Disassembly Line," in *Design for Innovative Value Towards a Sustainable Society*, M. Matsumoto, Y. Umeda, K. Masui, and S. Fukushima, Eds. Dordrecht: Springer Netherlands, 2012, pp. 714–719. [Online]. Available: http://link.springer.com/10.1007/978-94-007-3010-6_143
- [13] C. B. Kalayci and S. M. Gupta, "Ant colony optimization for sequencedependent disassembly line balancing problem," *Journal of Manufacturing Technology Management*, vol. 24, no. 3, pp. 413–427, Mar. 2013. [Online]. Available: <https://www.emerald.com/insight/content/doi/10.1108/17410381311318909/full/html>

- [14] Z. A. il, S. Mete, and F. Serin, "Robotic disassembly line balancing problem: A mathematical model and ant colony optimization approach," *Applied Mathematical Modelling*, vol. 86, pp. 335–348, Oct. 2020. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0307904X20302249>
- [15] K. Wang, X. Li, L. Gao, P. Li, and S. M. Gupta, "A genetic simulated annealing algorithm for parallel partial disassembly line balancing problem," *Applied Soft Computing*, vol. 107, p. 107404, Aug. 2021. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1568494621003276>
- [16] S. Mutlu and B. Gner, "A memetic algorithm for mixed-model two-sided disassembly line balancing problem," *Procedia CIRP*, vol. 98, pp. 67–72, 2021. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S2212827121000305>
- [17] T. Yin, Z. Zhang, and J. Jiang, "A Pareto-discrete hummingbird algorithm for partial sequence-dependent disassembly line balancing problem considering tool requirements," *Journal of Manufacturing Systems*, vol. 60, pp. 406–428, Jul. 2021. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0278612521001424>
- [18] L. Zhu, Z. Zhang, and C. Guan, "Multi-objective partial parallel disassembly line balancing problem using hybrid group neighbourhood search algorithm," *Journal of Manufacturing Systems*, vol. 56, pp. 252–269, jul 2020. [Online]. Available: <https://doi.org/10.1016%2Fj.jmsy.2020.06.013>
- [19] R. Usabmatov, A. M. Alwaise, and Z. M. Zain, "Productivity and optimization of section-based automated lines of parallel-serial structure with embedded buffers," *International Journal of Advanced Manufacturing Technology*, vol. 65, no. 5-8, pp. 651–655, 2013.
- [20] Z. Li and M. N. Janardhanan, "Modelling and solving profit-oriented U-shaped partial disassembly line balancing problem," *Expert Systems with Applications*, vol. 183, p. 115431, Nov. 2021. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0957417421008484>
- [21] K. Gao, Z. He, Y. Huang, P. Duan, and P. Suganthan, "A survey on meta-heuristics for solving disassembly line balancing, planning and scheduling problems in remanufacturing," *Swarm and Evolutionary Computation*, vol. 57, p. 100719, Sep. 2020. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S2210650220303722>
- [22] L. Volchkevich, "Automation of production processes," *Mashinostroenie, Moscow*, vol. 380, 2005.
- [23] L. Shi and S. Men, "Optimal buffer allocation in production lines," vol. 35, no. 1, pp. 1–10. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/074081703004431>
- [24] I. Sabuncuoglu, E. Erel, and Y. Gocgun, "Analysis of serial production lines: characterisation study and a new heuristic procedure for optimal buffer allocation," vol. 44, no. 13, pp. 2499–2523. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/00207540500465535>
- [25] C. B. Kalayci and S. M. Gupta, "Artificial bee colony algorithm for solving sequence-dependent disassembly line balancing problem," *Expert Systems with Applications*, vol. 40, no. 18, pp. 7231–7241, Dec. 2013. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0957417413004636>
- [26] C. Kalayci, "Solving sequence-dependent disassembly line balancing problem using a hybrid genetic algorithm," 01 2013.
- [27] C. B. Kalayci and S. M. Gupta, "A particle swarm optimization algorithm with neighborhood-based mutation for sequence-dependent disassembly line balancing problem," *The International Journal of Advanced Manufacturing Technology*, vol. 69, no. 1-4, pp. 197–209, Oct. 2013. [Online]. Available: <http://link.springer.com/10.1007/s00170-013-4990-1>
- [28] "River Formation Dynamics Approach for Sequence-Dependent Disassembly Line Balancing Problem," in *Reverse Supply Chains*, 0th ed., S. M. Gupta, Ed. CRC Press, Apr. 2016, pp. 307–330. [Online]. Available: <https://www.taylorfrancis.com/books/9781439899038/chapters/10.1201/b13749-16>
- [29] C. B. Kalayci and S. M. Gupta, "Simulated Annealing Algorithm for Solving Sequence-Dependent Disassembly Line Balancing Problem," *IFAC Proceedings Volumes*, vol. 46, no. 9, pp. 93–98, 2013. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1474667016342677>
- [30] —, "A tabu search algorithm for balancing a sequence-dependent disassembly line," *Production Planning & Control*, vol. 25, no. 2, pp. 149–160, Jan. 2014. [Online]. Available: <https://www.tandfonline.com/doi/full/10.1080/09537287.2013.782949>
- [31] C. B. Kalayci, O. Polat, and S. M. Gupta, "A variable neighbourhood search algorithm for disassembly lines," *Journal of Manufacturing Technology Management*, vol. 26, no. 2, pp. 182–194, mar 2015. [Online]. Available: <https://doi.org/10.1108%2Fjmtm-11-2013-0168>
- [32] —, "A hybrid genetic algorithm for sequence-dependent disassembly line balancing problem," *Annals of Operations Research*, vol. 242, no. 2, pp. 321–354, Jul. 2016. [Online]. Available: <http://link.springer.com/10.1007/s10479-014-1641-3>
- [33] L. P. Ding, Y. X. Feng, J. R. Tan, and Y. C. Gao, "A new multi-objective ant colony algorithm for solving the disassembly line balancing problem," *International Journal of Advanced Manufacturing Technology*, vol. 48, no. 5-8, pp. 761–771, 2010.



Nadir A. Siddig got his B.S. degree in mechanical engineering from Sudan University of Science and Technology, Khartoum, Sudan, in 2005, and M.S. degree in Production Engineering and Manufacturing from Sudan University of Science and Technology, in 2009, and is currently pursuing his Ph.D. degree from Southwest Jiaotong University, chengdu, China, School of mechanical Engineering.



Zeqiang Zhang is a professor and doctoral tutor of Southwest Jiaotong University. His current research interest is manufacturing systems and intelligent optimization



Abdallah mokhtar is an Assistant Professor in College of Engineering, Sudan University of Science and Technology, Mechanical School/ Power Department, got his B.S. degree in mechanical engineering from Sudan University of Science and Technology, Khartoum, Sudan, in 2004, M.S. degree in Mechanical Engineering from Sudan University of Science and Technology, in 2009, Ph.D. degree in Mechanical Engineering from Sudan University of Science and Technology, in 2016.



Ahmed Abualnor is received a B.Sc. in Information Technology and Information science from the University of Blue Nile, Sudan, in 2004, and M.Sc. degrees in Communication Engineering from Omdurman Islamic University, Sudan, in 2011. He is currently pursuing a Ph.D. degree in Information and Communication Engineering from the School of Information Science and Technology, Southwest Jiaotong University, Chengdu, China. He was a Lecturer with the Sinner University, Sudan. His current research interests include 5G network, energy-efficient

communication, Non-orthogonal Multiple Access (NOMA), millimeter-wave communications, cooperative communications, heterogeneous small cell networks, and resource allocation and optimization.