

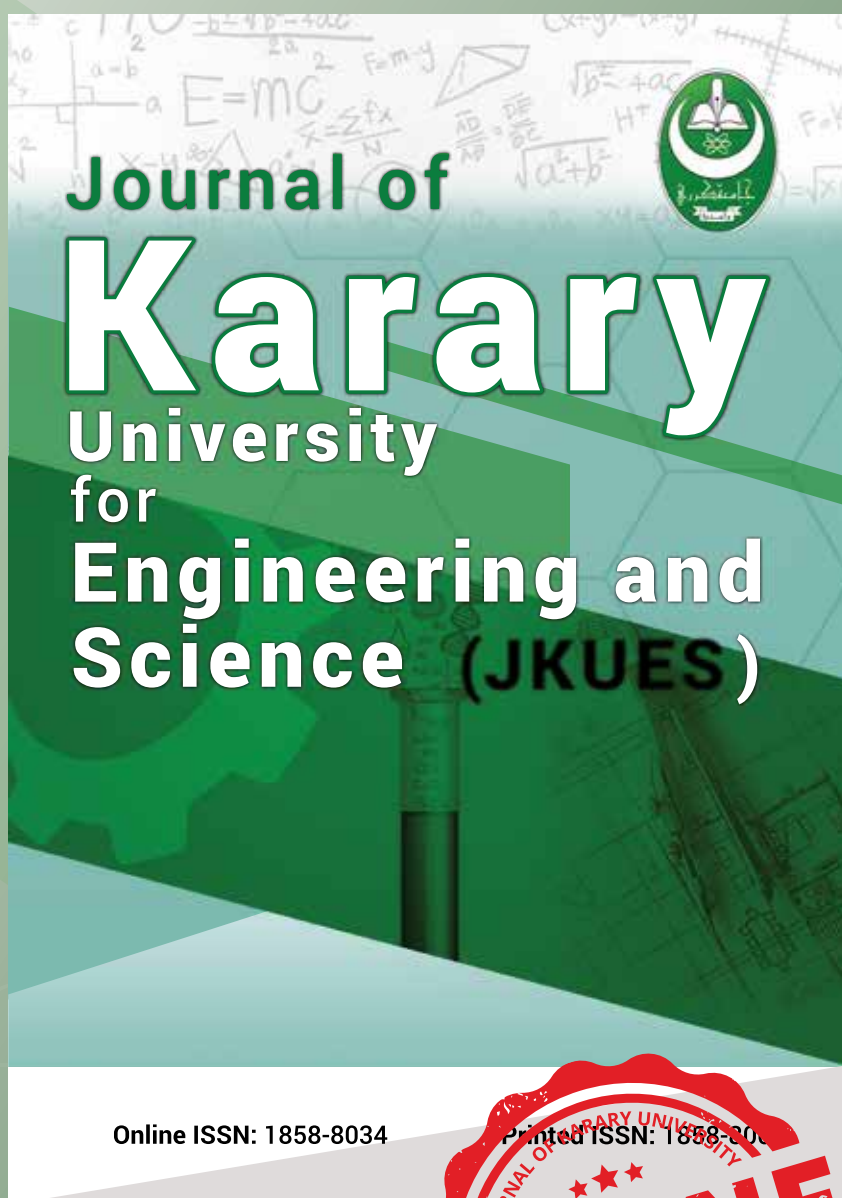
Hybrid Genetic-ADAM Optimizer to Improve Disassembly Line Balancing Problem

Nadir A.Siddig

Zeqiang Zhang

Abdallah mokhtar

Ahmed Abualnor



Online ISSN: 8000-1858

Print ISSN: 800-1858X

Online ISSN: 1858-8034

Printed ISSN: 1858-8034

Article history:

Received	21, December 2021
Received in revised form	11, February 2022
Accepted	16, February 2022
Available online	30, March 2022



Hybrid Genetic-ADAM Optimizer to Improve Disassembly Line Balancing Problem

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.

[†]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.

Email: nadir@my.swjtu.edu.cn

Abstract

Modern products are designed to meet the needs of customized and short-lived products, after a short period of time, these products become waste products, these waste products are accumulating at an exponential rate, resulting in environmental degradation as a result of pollution, in order to keep this environment clean, it is necessary to have disassembly lines for those waste products, so in this research a Hybrid Genetic ADAM optimizer Algorithm was proposed to facilitate the disassembly operations at work stations placed in the disassembly lines, the proposed algorithm used to solve multi-objective disassembly line balancing problem, and Pareto optimal solution was used to determine non inferior solutions from a population. The trend of results reveal 6.1 percent a reduction in disassembly workstations, reduction of at least 0.05 percent of the idle time, reducing to minimum run time 1.5 percent comparing proposed and other meta heuristics algorithms. Furthermore, the plans of disassembly data are created to be valuable regarding trace ability and disassembly improvement processes in the future.

Key words- *Hybrid Genetic ADAM optimizer Algorithm, multi objective disassembly line balancing problem, Pareto optimal solution.*

I. INTRODUCTION

Due to the great industrial progress these days, as well as the development of products day by day, the waste products are increasing exponentially, and this leads to the pollution of the environment surrounding human life, we need different plans to disassembly these waste products, these plans help in accelerating disassembling processes and balancing the necessary disassembly lines. Other methods of disassembly must be established to lower the risk of infection from contact with waste products, to make disassembly activities easier, and to reduce the amount of stressful labor for personnel. These ideas are embodied in the plans for disassembly workstations, with the primary purpose of any workstation or group of workstations in the disassembly line being to complete an annual disassembling timetable. In order to attain minimum idle time for each activity, researchers should focus on lowering the idle time used at these workstations as much as possible, in addition to using the overall time required to operate them. Assembling parts to make devices has become very easy these days, with large and small factories working in workshops or at home to assemble parts to make a variety of items. However, as a result of a variety of factors, such as poor design and short product life, these items become waste, resulting in a massive pile of obsolescence devices [1], Despite the fact that traditional disassembly plans have begun to balance this rising demand, traditional disassembly is insufficient to deal with the rapid outdated manufacturing of electronic and electric parts. Many suggestions for improving disassembly line work have been studied, such as [2], which used product self-disassembly technology to improve recovery

performance and reduce component damage, and [3], which purposed a method to improve labor productivity at assembly line with U shaped. On the other hand, a meta-heuristic methodology was improved to address the issue of adjusting the disassembling line based on arrangement [4], as well as the issue of adjusting the disassembling line based on degree of completion of ancillary work for loading and operating the workstations system with a parallel and semi-automated disassembly line [5]. In flexible processing, a Petri net method was used to forecast energy behavior, which incorporated parallel systems. [6], a few methods on planning the energy concept of mechanized lines have been offered in the study as an equipment for energy saving and reducing major adverse effects [7], as an apparatus for energy sparing and decrease significant inverse impacts, a few methodologies on planning the energy thought of mechanization lines have been proposed in the study [8], and a method of energy saving optimization that is chosen to choose the workstations on a line for picking up and viewing energy savings by an additional algorithm for analysis has been developed [9], disassembly for the recovery of deliberately significant materials from electric devices was proposed to solve a problem in disassembly lines, and an ant colony evaluation algorithm was improved to find the ideal disassembling progression to choose the sub assembling, and an ant colony evaluation algorithm was improved to find the ideal disassembling progression to choose the sub assembling [10], and [11] suggested a mathematical paradigm to limit power costs for arranging the creation of single machines during creation measures, to reduce energy consumption based on graphical restoration of constraints and improve the algorithm for optimizing ant colony modify was

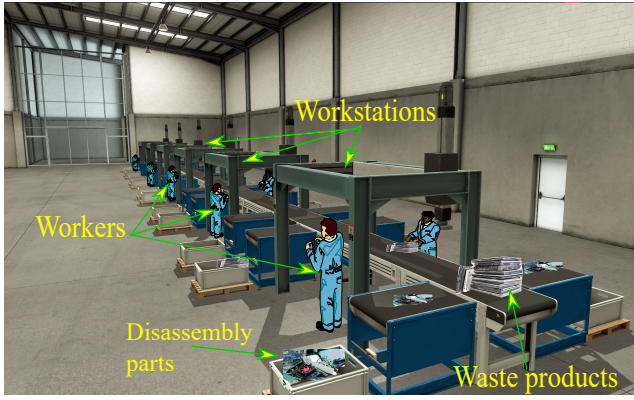


Fig. 1. Traditional disassembly line in one side workstations.

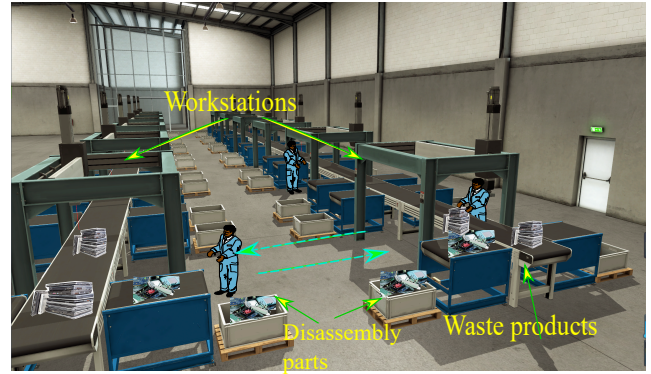


Fig. 2. Proposed an parallel disassembly, the workers can be move between opposed workstations, and working in one workstation when the other workstation is idle time.

used to find the global collection sequence determined by the selection and the proposed number of components [12], [13] suggested that the problem of balancing the uncertainly disassembling line with numerous purposes required for the distribution of disassembling tasks for the distributed assembly of disassembling workstations, and to solve the problem of balancing traditional collection and the use of Monte Carlo techniques to determine reinforcement training to compile a large-scale problem within a reasonable computational time to solve the DLBP [14]. In addition, an artificial swarm method was improved to handle the challenges of balancing traditional collection with multi-objective optimization and optimal Pareto solutions to solve the problem of disassembly [15], the disassembly of a mathematically multi-objective model utilized to solve a parallel line with two products is required. [16], to change production process, a two-phase heuristic model with mathematical method was proposed once more to solve the problems in DLBP [17], and a new technique was developed to produce a disassembly sequence for tasks that may be reused or re manufactured [18]. There is always a gap in the disassembling lines, and this gap is concentrated in lost time during disassembling operations, which may lead to a lack of knowledge of sufficient work stations, a lack of knowledge of how to disassemble steps without stopping, a lack of knowledge of disassembling costs and whether the process is profitable or not, and finally, a lack of knowledge of the best disassembling directions to reduce disassembling time.

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

II. METHODOLOGY

A. Problem description

The main objective of this paper is to assist in the resolution of problems associated with disassembly lines in general, including the danger of disassembly parts, the accumulation of required disassembly parts, not knowing the appropriate

number of workstations required for disassembly operations, and idle time during disassembly operations. A hybrid algorithm was proposed between the genetic algorithm (GA) and ADAM optimizer to reach the best solutions, which are global solutions and avoid local solutions, where the genetic algorithm elicits improved genetic population and global mutations from solutions, and ADAM optimizer is a meta heuristics algorithm using to automate elicitation of all solutions and filtering global solutions, to address a set of problems that were set in specific goals and set limitations, and to reach excellent solutions. We also proposed a Parallel Disassembly Line Balancing Problem (PDLBP)(see [Fig. 2](#)) instead of a traditional Disassembly Lines Balancing Problem (TDLBP) that are in one line (see [Fig. 3](#)), and in this way we can reduce the idle time in the operating stations.

In the literature, [19], the pareto filter is used to identify non inferior solutions to multi-objective issues. It maximizes each aim at the same time, disregarding the importance of the other four. Simultaneously, it solves the overflow problem of non-inferior solutions with an enhanced crowding distance technique. Specifically, the problem of screening sub-objective boundary solutions. Following the execution of the algorithm, many non-inferior solutions with various focuses can be found, giving decision-makers with a variety of possibilities.

III. PROBLEM DEFINITION AND MODEL

The disassembly line balance problem is a type of combinatorial optimization problem that has several objective functions and complex constraints. The disassembly sequence, disassembly task allocation, and the number of workstations opened must all be determined when completely disassembling waste products. Other decision-making variables include a demand index, a danger index, and other factors. Assume that used mechanical and electrical items are made up of n parts, and assign a disassembly task to each one.

A. Mathematical Models

The mathematical description of multi-objective disassembly line balance problem is as follows:

$$\text{Min } F = [f_1, f_2, f_3, f_4] \quad (1)$$

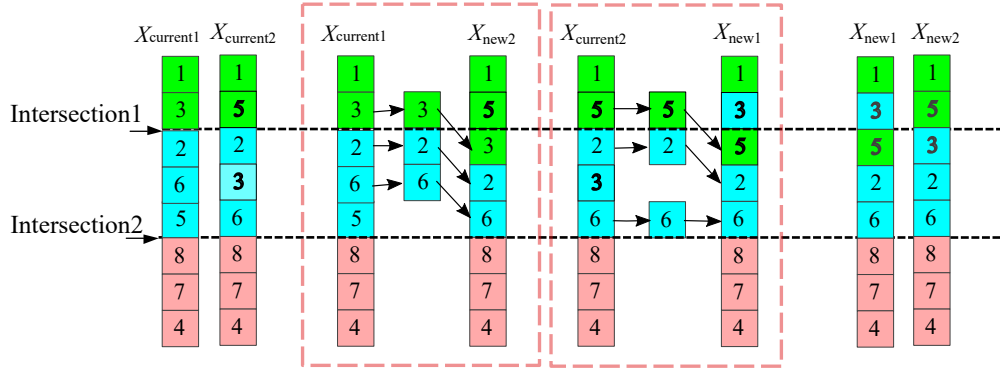


Fig. 3. Schematic diagram of GA computing, selection, and mutation

$$\min f_1 = K; \quad (2)$$

$$\min f_2 = \sum_{j=1}^K (T_C - T_{sj})^2; \quad (3)$$

$$\min f_3 = \sum_{j=1}^K T_C \cdot (\max \{W_i \mid i \in (i \mid x_{ij} = 1)\}); \quad (4)$$

$$\min f_4 = \sum_{i=2}^m R_i \quad (5)$$

The objective function: Eq. 1 represents four minimization objectives, Eq. 2 represents minimizing the number of workstations; Eq. 3 represents balancing the direction change of each workstation; Eq. 4 represents minimizing the cost of disassembly, and the disassembly cost per unit time of the workstation is defined as the disassembly task in the workstation; Eq. 5 represents to minimize the number of changes in the direction of disassembly. It needs to be pointed out that the change in the direction of the disassembly task in the parallel disassembly line is not the change in the direction of the workers task in the disassembly line.

Constraints

$$\left[\sum_{i=1}^m t_i / T_C \right] \leq K \leq n; \quad (6)$$

$$T_{sj} \leq T_C, \quad \forall j \in \{1, 2, \dots, K\} \quad (7)$$

$$\sum_{j=1}^K x_{ij} = 1, \quad \forall i \in \{1, 2, \dots, m\} \quad (8)$$

$$x_{aj} \leq \sum_{j=1}^K x_{ij}, \quad \forall (a, i) \in MP \quad (9)$$

Where: K is the number of workstations; T_C is the disassembly cycle time; T_{sj} is the operating time of the j -th workstation; m is the number of disassembly tasks; W_i is a disassembly cost of the i -th disassembly task; define r_i as the i -th disassembly task the direction of disassembly, X_{ij} is the relationship between the task and the workstation, if task

i is assigned to workstation j , then $X_{ij}=1$, otherwise $X_{ij}=0$; MP is order to disassemble the task priority relation set, (a, i) refer that task a has priority over task i .

$$r = \begin{cases} +x & +1 \\ -x & -1 \\ +y & +2 \\ -y & -2 \\ +z & +3 \\ -z & -3 \end{cases} \quad R_n = (r_1, \dots, r_j, \dots, r_n)$$

where r_j the possible disassembly direction of part j which indicated by integer; R_n the disassembly direction matrix of parts, if $r_i = r_i - 1$, Then $R_i = 0$, otherwise $R_i = 1$; t_i is the working time of the i -th disassembly task. Eq. 6 refers that the number of workstations is not less than the number of theoretical workstations and does not exceed the number of disassembly tasks; Eq. 7 refers that the operating time in each workstation does not exceed the production cycle; Eq. 8 refers that each task is Must be assigned to a workstation; Eq. 9 indicates that the disassembly sequence must satisfy the disassembly priority relationship.

IV. PARETO GA-ADAM ALGORITHM

The GA-ADAM algorithm and GA algorithm show good solution performance in solving aggregate optimization problems. In this paper, according to the characteristics of the disassembly line balancing problem, together with the idea of the Pareto solution set, the initial practical solution of ADAM genetic algorithm is improved, and the genetic process is created. ADAM Pareto genetic algorithm to solve the parallel multi-objective disassembly line equilibrium problem.

A. Building a practical solution

The use of ADAM genetic algorithm to solve the disassembly line equilibrium problem can be considered as a decent random gradient looking for the next disassembly task with a directive to update the weights of the iterative network based on training data. The path that ant passes through constitutes the sequence of disassembling the problem. The guideline in ADAM genetic algorithm is the predicted score for selecting the next decoding task. Due to the characteristics of the equilibrium problem of the disassembly line, the guideline

is the longest disassembling process time and the smallest disassembling cost difference, and the experimental function of the task η_l is

$$\eta_l = \frac{t_l}{T_C} + \left(1 - \frac{|W_l - W_i|}{\max_{m \in A_l} |W_m - W_i|}\right) \quad (10)$$

In this equation: A_l represents the feasible set of tasks for the current task l and i represents the i task completed. The probability of GA-ADAM algorithm from task i to task j is

$$p_{ij} = \begin{cases} Y_1 : i = \arg \max_{l \in A_j} \left(\left(\sum_{e=1}^j \tau_{le} \right)^\alpha \cdot \eta_l^\beta \right), \\ 0 \leq r \leq r_1 \\ Y_2 = \frac{\left(\sum_{e=1}^j \tau_{ie} \right)^\alpha \cdot \eta_i^\beta}{\sum_{s \in A_j} \left(\sum_{e=1}^j \tau_{se} \right)^\alpha \cdot \eta_s^\beta}, r_1 < r \leq r_2 \\ Y_3 : \text{random selection } i \in A_j, r > r_2 \end{cases} \quad (11)$$

In this equation: r is the random number between (0,1); r_1, r_2 is the defined parameter; gradient descent optimization for the task i to the subsequent task e ; A_j is the subsequent optional task set of the task i ; gradient descent optimization weight; heuristic factor weight. Y_1 indicates that if $r \leq r_1$, selects the maximum task j from the next task assignment; Y_2 indicates that if $r_1 < r \leq r_2$, selects the next task of random proportions in roulette selection; and Y_3 refers that if $r > r_2$, randomly selects a task from the feasible task set A_j . In order to ensure that the feasible solution meets the actual disassembly situation of the disassembly line, the choice of each task should consider the disassembly priority relationship constraints, the generation process of feasible solutions does not consider the influence of the disassembly removal, as the main parameter affecting the number of workstations, at the same time, fix the position of each task on both sides of the parallel disassembly line.

B. Filter for the Pareto solution set

Using the gradient descent optimization method and the sequential processing method, its essence is to transform multi-objective problem into a single-objective problem. The Pareto solution set idea can balance between multiple optimization objectives (see Fig. 4), and the solution results are more consistent with the actual situation of the problem and can be used as the solution idea of the problem. Given two feasible removal sequences X_1, X_2 , if X_1, X_2 are run across.

$$f_m(X_1) \leq f_m(X_2), \quad \forall m \in \{1, 2, 3, 4\} \quad (12)$$

$$f_m(X_1) < f_m(X_2), \quad \exists m \in \{1, 2, 3, 4\} \quad (13)$$

Call the X_1 Pareto dominates the X_2 , as $X_1 < X_2$. If the feasible solution X_* satisfies: $\neg \exists X$ so, the feasible solution X_* is called the Pareto optimal or solution that isn't inferior. The Pareto optimal solution set is made up of all Pareto optimal solutions, and the Pareto optimal frontier is made up of multi-objective function values of the disassembly sequence. Store the Pareto non-inferior solution in the data set, determine the dominant relationship between the new solution and the data set' non-inferior solution, leave the non-dominant solution in the data set while the dominant solution is removed, and complete the external file update.

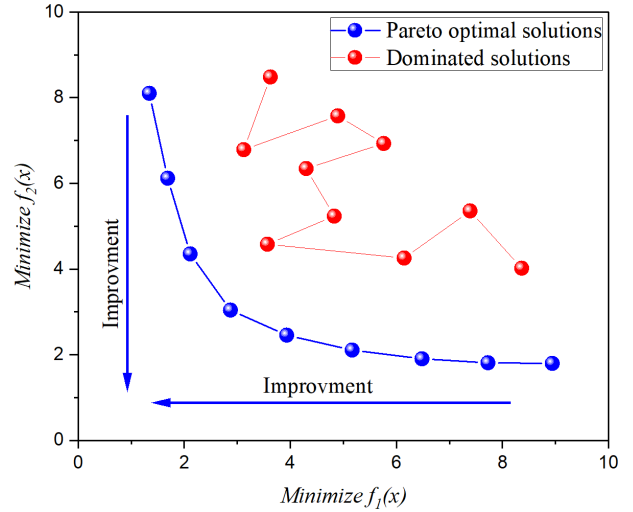


Fig. 4. The f_1 comparison between GA-ADAM and the other ten algorithms.

C. Selecting Operation

The fitness function is the objective function in multi-objective disassembly line balance problem. The most common form of selection is to choose based on one size of the fitness value; the probability of selection is $P_i = f_i / \sum_{i=1}^n f_i$, and in multi-objective problem, the selection probability cannot be determined, the gradient descent optimization algorithm's data set is not superior, and the higher the probability, the higher the fitness value, the non-inferior solutions of the external archives are optional high-quality individuals. As the next phase, two non-inferior solutions were chosen at random using the elite selection approach.

D. Interlace Operation

Fig. 3 cross $X_{current1}, X_{current2}$ the result of selecting action, with two points for the parent. The sequence before intersection 1 and the sequence after intersection 2 stay intact; the sequence between the individual points 1 and 2 is chosen at random as an intersection. $X_{current1}$ intersection 1 and intersection 2 is obtained by the parent $X_{current2}$ map, forming the new child individual X_{new1} , can the new child individual X_{new2} . The specific operation procedure is shown in Figure 3. Since the current removal task in the parallel disassembly line can be selected from the non-tight back task, the task in the crossover section may not meet the removal priority relationship before its tight back task, and you need to re select the intersection point.

E. Mutation Operation

try to make more use of crossing character. $X_{current}$ as a single-point insertion variation's parent. The variation point of the point task is determined by a random point in the parent sequence k_i , which can be positioned between the tasks closest to the work and the tight front and back tasks k_i . Maintain the sequences preceding and following the tight front task, randomly select a point in the insertion position for insertion, do the variation operation, and create a new individual. X_{new} .

TABLE I
DISASSEMBLY CHARACTERISTICS OF THE MECHANICAL SENSING VALVE.

Part	Name	Q_{ty}	r	T	P	c
1	Body valve	1	-1	-	2	0.78
2	Rear cover	1	+1	T1	-	0.69-0.98
3	Compressor piston	1	-1	T3	1	.3
4	Sliding rod	1	+1	-	9	0.9
5	Pressure springs	1	+1	T4	3	0.2
6	Restoring springs	1	+1	T5	5	0.1
7	Rubber ring	1	+1	T6	16	0.1
8	O rubber ring	1	+1	T6	2	0.1
9	Pin	1	-3	T7	3	0.5
10	O rubber ring	1	-1	T6	4	0.1
11	Connecting rod	1	-1	-	12	0.04
12	Sandwich valve	1	-1	T8	6	0.1
13	Sandwich spring	1	-1	T8	14	0.1
14	Gland	1	-1	T8	15	0.1
15	Snap ring	1	-3	T9	3	0.2
16	Snap ring	1	-3	T9	-	0.8
17	Sealant	1	+1	T6	1	0.3

You must re-select the variation point if you insert in the selection location or if task priority constraints are not met. In Fig. 3 depicts the variation process.

F. ADAM Optimizer

ADAM optimization is a stochastic gradient descent approach that uses Adaptive Moment (ADAM) estimation, the method is computationally efficient, has small memory needs, is robust for diagonal re scaling of gradients, and is well suited for big data/parameter problems, according to research of ADAM optimizer [20] this meta heuristics works in tensor flow navigator with high accuracy and low loss of data analysis. Initialization, stochastic gradient descent discretization Eq. 14 The decaying averages of past squared gradients, m_t and m_v , are computed in the following way:

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \end{aligned} \quad (14)$$

β_1 and β_2 are decay rates of average of gradients ($\beta = 0.9$ and $\beta = 0.999$), g_t is the exponentially decaying average of squared gradients.

G. Case Study

Mechanical sensor valve is shown in Fig. 5, and exploded views of the parts are shown in Fig. 6. The Mechanical sensor valve disassembly case contains a total of 17 disassembly tasks. table I shows the parts, the disassembly quantity (Q_{ty}), direction change (r), disassembly tool (T) of each disassembly task, proceed task (P), and task time (c) The cost per yuan. There are nine types of tools, with the numbers 19 representing a hand, an electric screwdriver. The mathematical models were implemented in PC with Microsoft windows 10, the PC with Intel Core (R) and (TM) i710510 CPU 2.30 GHZ and 16 GB RAM, the proposed algorithm is coded and verified in python 3 in anaconda navigator platform, FACTORY I/O, and Codesys were used to simulate results.

H. presumptive situation

In order to simplify the model, the following assumptions were made: different types of waste products have different structures, portions of the quantity and quality uncertainty, some parts of the joint damage, and so on.

- 1) The waste product is only one type;
- 2) The amount of end-of-life products is sufficient, allowing assembly line production to proceed;
- 3) The quantity of parts in end-of-life items is comprehensive, and no instances of loss or modification have been reported;
- 4) The components are securely attached and may be disassembly easily;
- 5) Disassembly operations should be standardized ; 6. Ignore the unknown factors in the disassembly process and set the disassembly time.
- 6) All components of waste products should be disassembled ;
- 7) A disassembly duty is assigned to each part;
- 8) Each disassembly task cannot be subdivided any further.

Multi-objective method are the major features of the proposed GA-ADAM. The proposed algorithm is created by integrating the characteristics of the problem in order to ensure that the outcomes are feasible. This article proposes a multi-objective technique based on Pareto dominance to assure the diversity of disassembly strategies. The GA-ADAM flowchart is represented in the diagram as shown in Fig. 7.

V. RESULTS AND DISCUSSION

Sensitivity analysis was performed with the aim of investigating and optimizing the overall process conditions of multi-objective disassembly problems.

A. Effect of GA-ADAM to global results

In this research, multi-objective Pareto line was diversified to obtain global results, and we actually obtained them as shown in Fig. 8, where it appears to us in a three-dimensional Pareto diagram. By comparing three algorithms, we see that the GA-ADAM obtained global solutions as there is a reduction of the three objectives f_2 , f_3 , and f_4 , but some solutions such as number 12, 11, 10, 9, 8, and 7 achieved relatively high amounts, but these solutions were taken directly without checking the confidence period, but the confidence period was taken into account and better results were obtained with a confidence period of 95% as shown in Fig. 9 we can note that all solutions have become more global than GA, and MOAOA.

table III According to analysis, GA-ADAM obtained a minimum $f_1 = 3$ workstation, as indicated in Fig. 12, minimum idle time $f_2 = 0.002$ sec as indicated in Fig. 13, minimum cost $f_4 = 0.1075$ yuan/sec as indicated in Fig. 15, and the work time of GA-ADAM in the computer 0.30sec to obtain results for 17 parts disassembly process for mechanical sensor valve as indicated This example completely disassembled 17 parts of the casing, springs and rubber parts, the optimization objectives is to minimize the average direction change rate $F_{direction}$, maximum load balance index F_{smooth} , minimize

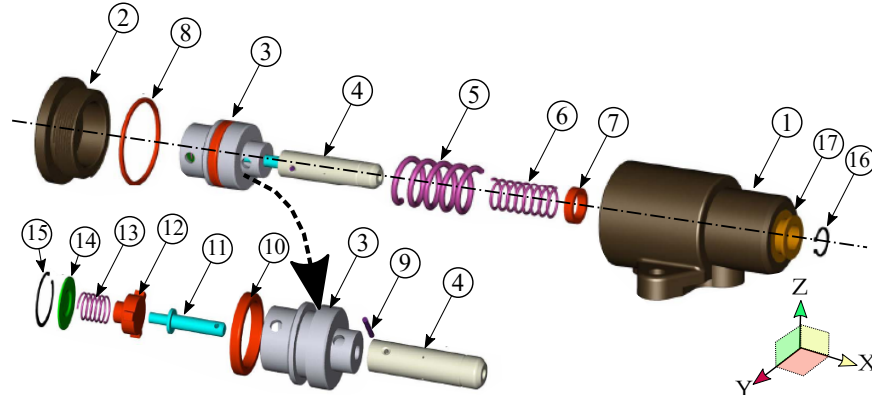


Fig. 5. The mechanical sensor valve is an example of a waste product with 17 elements that require disassembly.

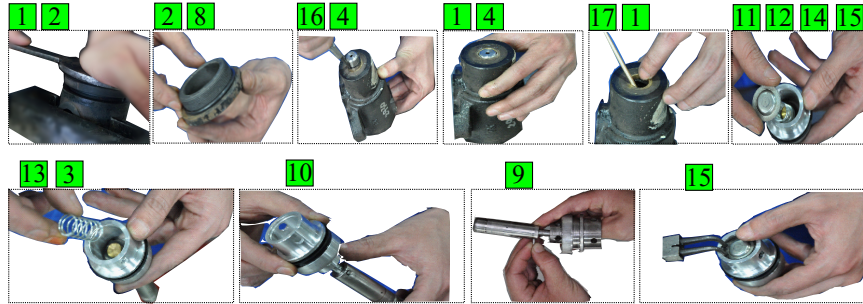


Fig. 6. The real parts of a waste product example are shown in Fig. 5, along with a sequence of tasks to complete disassembly processes.

the disassembly cost F_{cost} . The results of 6 equilibrium schemes obtained by GA-ADAM using multi-objective Pareto ADAM algorithm are listed in table II, using multi-objective Pareto GA-ADAM algorithm, whose parameter is set to: $N = 200$, $T = 200$, $N_0 = 8$, $\alpha = 1.5$, $\beta = 0.5$, $r1 = 0.6$, $r2 = 0.9$, $1 = 0.4$, $2 = 0.3$, $q = 10$, $P_c = 0.9$, $Pm = 0.1$, $T_C = 400sec$. A total of 6 Pareto solutions were obtained and all their equilibrium schemes are listed in table II in three workstations.

The algorithm results of this paper are compared with other meta heuristics algorithms such as Artificial Bee Algorithm (ABC) is optimization method based on honey bee swarm's intelligent foraging behavior, proposed by [24], Ant Colony Optimization (ACO)[23] is a probabilistic method for resolving computing issues that can be simplified to the search for good paths via graphs. Multi-agent methods inspired by the behavior of real ants are known as artificial ants. Biological ants' pheromone-based communication is frequently used as a model. Artificial ants and local search algorithms have been the preferred solution for a variety of optimization challenges requiring graphs. Genetic Algorithm (GA)[22], Particle Swarm Optimization (PSO)[25], RFD (River Formation Dynamics, RFD)[21], Simulated Annealing (SA)[26], Tabu Search (TS)[27], variable neighborhood search (VNS) [4], genetic algorithm with a variable neighborhood search (VNSGA) [28].

B. The effect of the proposed algorithm idling time index

The idle time adversely affects production and thus profit, which increases the costs of disassembling products, so we

tried in this research to reduce the idle time by proposing a PDLBP method to reduce the idle time so that the worker can work in the opposite station during the stoppage of the first station. According to Fig. 10 we can see the effect range of workstations between 4-11 workstations, and the total idle time in the PDLBP at the workstations can be reduced by GA-ADAM rather than GA, and MOAOA. In Fig. 11 we can see the comparison between two algorithms (GA-ADAM, and GA), and the proposed algorithm Beat the second algorithm.

C. The effect of the proposed algorithm on smoothing index

On F_{smooth} indicators, the proposed algorithm results range of 98.6%-99.8%, while the GA, and MOAOA results range of 85.9%-97.5%, the proposed algorithm has a higher balance rate, according to table II, the smoothing index range between 0.8594-0.9285, and can be considered if the minimum smoothing index is required, minimum $f_3 = 0.08875$ sec need to complete the disassembly processes in the parallel disassembly line as indicated in Fig. 14 f_3 in index of smooth work objective, the suggested algorithm achieves 4.9 %, which is comparable to previous algorithms.

D. The effect of the proposed algorithm on cost index

the disassembly cost range is $0.6987 - 0.715yuan/s$. If the cost is minimum, PDLBP can be selected; the number of disassembly direction changes is 17-20. If the minimum change of disassembly direction is required, PDLBP can be selected, these two indicators can also be taken into consideration: if the smoothing index and disassembly cost are given

TABLE II
RESULTS OF PDLBP AND TDLBP NON BAD RESULTS FILTRING BY PARETO GA-ADAM OPTIMIZER ALGORITHM

SN	Task assignment scheme	$F_{workstations}$	F_{idle}	F_{smooth}	F_{cost}
1	$\{2, 1, 8, 3\} \rightarrow \{9, 4, 10, 11, 12, 13, 14, 15\} \rightarrow \{16, 17, 5, 6, 7\}$	3	0.00279	0.8594	0.1050546
2	$\{2, 1, 8\} \rightarrow \{3, 9, 4, 10, 11, 12, 13, 14, 15\} \rightarrow \{16, 17, 5, 6, 7\}$	3	0.00579	0.9285	0.1059768
3	$\{2, 1\} \rightarrow \{3, 9, 4, 10, 11, 8, \} \rightarrow \{16, 12, 13, 14, 15, 17, 5, 6, 7\}$	3	0.00379	0.8632	0.1052412
4	$\{2, 1, 16, 17\} \rightarrow \{9, 4, 10, 15, 14, 13, 12, 11\} \rightarrow \{8, 3, 5, 6, 7\}$	3	0.04757	0.9681	0.1070357
5	$\{2, 1, 16, 17\} \rightarrow \{3, 9, 4, 10\} \rightarrow \{8, 5, 6, 7, 15, 14, 13, 12, 11\}$	3	0.01757	0.9747	0.1073388
6	$\{2, 1, 8, 3\} \rightarrow \{9, 4, 10, 15, 14, 13, 12, 11\} \rightarrow \{16, 17, 7, 5, 6\}$	3	0.01757	0.9451	0.1067416

TABLE III
COMPARISON BETWEEN 9 PRECEDING META HEURISTICS ALGORITHMS, AND PROPOSED ALGORITHM GA-ADAM)

Source	Algorithm	f_1	f_2	f_3	f_4	Time/s	Computer configuration	Program environment
literature [21]	RFD	4	0.0579	0.0285	0.150546	3.62	Intel Core2 1.79GHz	MATLAB
literature [22]	GA	4	0.1757	0.0178	0.152412	5.3	Intel Core2 1.79GHz	MATLAB
literature [23]	ACO	4	0.2935	0.0378	0.154278	5.36	Intel Core2 1.79GHz	MATLAB
literature [24]	ABC	4	0.4113	0.0373	0.156144	2.4	Intel Core2 1.79GHz	MATLAB
literature [25]	PSO	4	0.5291	0.042	0.15801	0.81	Intel Core2 1.79GHz	MATLAB
literature [26]	SA	4	0.6469	0.0466	0.159876	-	Intel Core2 1.79GHz	MATLAB
literature [27]	TS	6	0.4113	0.0378	0.154275	0.87	Intel Core2 1.79GHz	MATLAB
literature [4]	VNS	6	0.52455	0.159833	0.446154	0.83	Intel Core i7-3770 3.40GHz	MATLAB
literature [28]	VNSGA	6	0.9271	0.87496	0.76456	0.65	Intel Xeon E5-2650 2GHz	C++
literature [29]	MOAOA	4	0.0344	0.45656	0.34356	0.30	Intel Core (R) i710510 CPU 2.30 GHz	MATLAB & python
This paper	GA-ADAM	3	0.002	0.08875	0.10754	0.30	Intel Core (R) i710510 CPU 2.30 GHz	MATLAB & python

priority, solutions 2 or 6; if the smoothing index is maximum, and less changes of disassembly direction, solutions 6 or 5; if the requirements are low disassembly cost and the number of disassembly direction changes is less, solution 1 or 2 can be selected. The 6 balance solutions obtained in this paper can provide multiple choices for decision makers.

E. The effect of the proposed algorithm on running time index

in Fig. 16 and due to the rapid development of the computer, a tenth generation computer was used with i7 core, and the Python package is faster than Mat lab, so the program execution speed is the best in the least running time of the algorithm compared to the rest of the other algorithms, they were implemented in earlier times, so The proposed algorithm outperformed by 1.5 % equally with MOAOA.

VI. CONCLUSION

Through the hybrid genetic algorithm with Adam Optimizer, we can reach solutions to problems available in the disassembly lines of all kinds, and can fill a gap in the field of disassembly technology, which in turn leads to the improvement of disassembling lines by reducing work stations, reducing disassembling costs, and reducing changes in disassembling directions. The GA-ADAM algorithm combines the advantages of the genetic algorithm and the ADAM optimizer algorithm, which can prevent the algorithm from falling into the local optimum. Improve the heuristic rules of the genetic algorithm to make the assignment of tasks more in line with the actual production situation of the problem; evaluate the non-inferior solutions through the crowding distance and filter the non-inferior solutions to achieve elite retention; use the crowding degree to update the global information of the GA

it can balance the influence of each target on concentration of mutation in the population; treat non-inferior solutions as individuals of genetic manipulation to speed up the efficiency of genetic manipulation; the result of genetic manipulation is positively fed back to the concentration of gene on optimal solution, which ADAM Optimizer search optimal solutions, while speeding up the convergence speed of the algorithm. Use the proposed algorithm to verify the test problems of 17 disassembly tasks, change the linear disassembly line to parallel disassembly line, and compare it with multi-objective Pareto GA-ADAM algorithm in the linear disassembly line. The proposed algorithm The solution is obviously better than the linear disassembly line scheme, which shows that the proposed algorithm can effectively solve multi-objective disassembly line balancing problem.

The proposed algorithm has better results than other algorithms, and it reduces the disassembly workstation by 6.1%, reduce idle time to 0.05%, reduce cost to 3.9%, and reduce minimum run time 1.5 % comparing proposed and other meta heuristics algorithms. But the proposed algorithm needs to be developed more in the future to give a better result in terms of smooth work objective, as it achieved 4.9% not less than other algorithms.

DECLARATION OF INTEREST STATEMENT

The authors state that they have no possible consideration ambivalence, and no potential conflicts of interest.

ACKNOWLEDGMENTS

National Science Foundation of China supported this research by [Number's grant 51675450, 51205328]; Youth Establishment for Humanities and Social Sciences Ministry

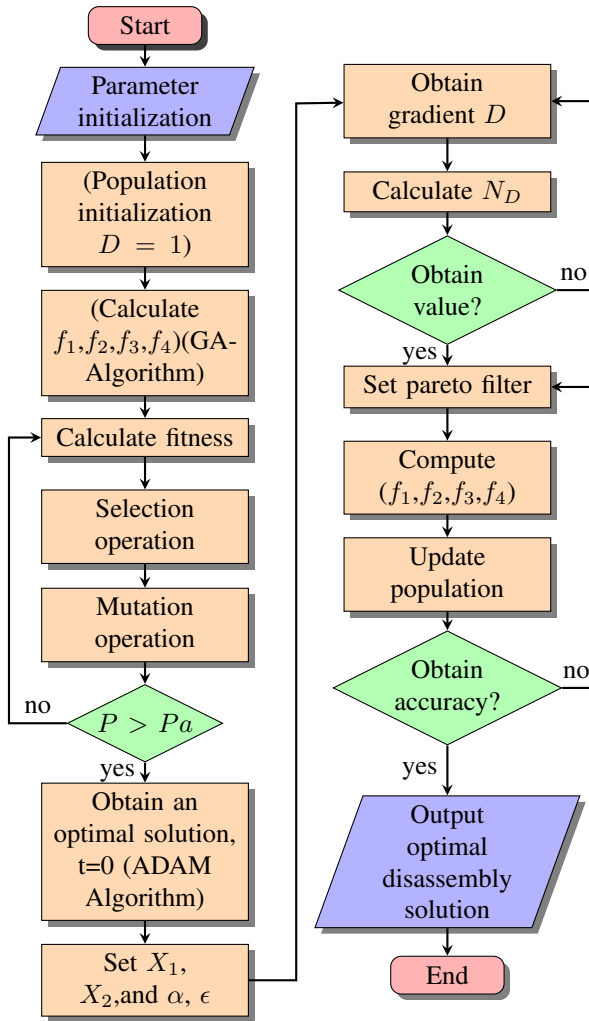


Fig. 7. The Pareto flow chart process based on GA-ADAM.

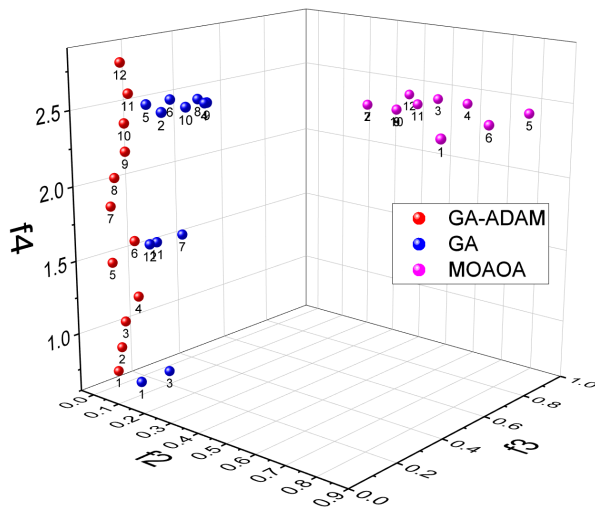


Fig. 8. The pareto optimal set to filter the solutions after balance objectives and compare the objectives three algorithms, GA-ADAM investigated the superior objectives.

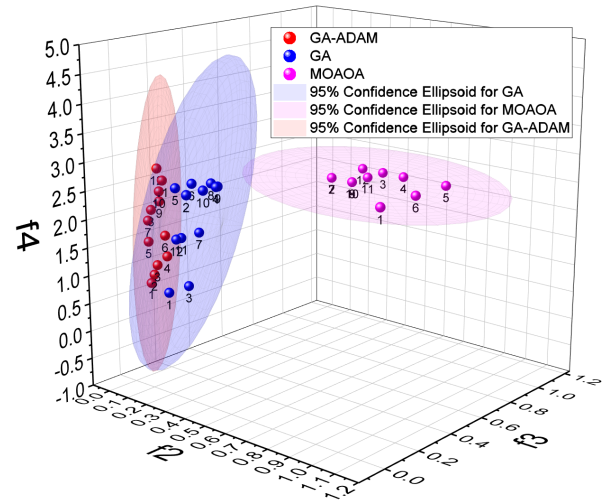


Fig. 9. The pareto optimal set to filter the solutions after balance objectives and compare the objectives three algorithms, GA-ADAM with 95% confidence interval three algorithms .

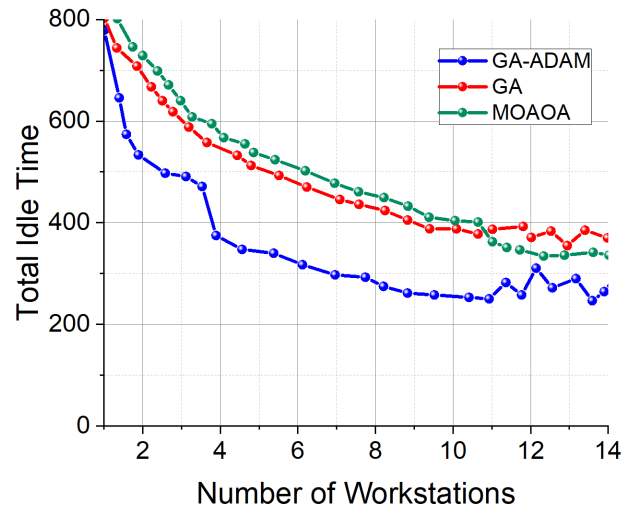


Fig. 10. The total idle time loss in the workstations.

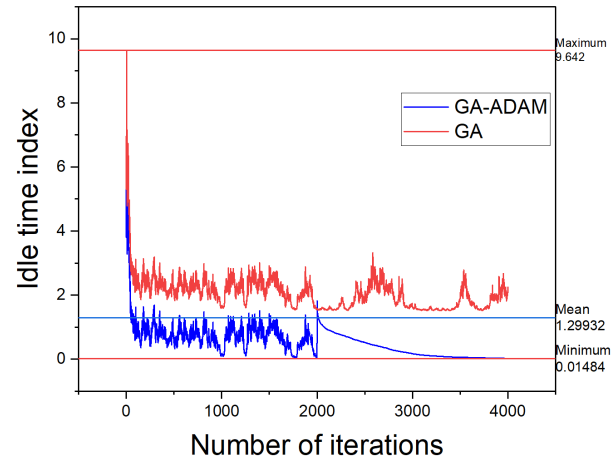


Fig. 11. The comparison between GA-ADAM and GA to reduce idle time index.

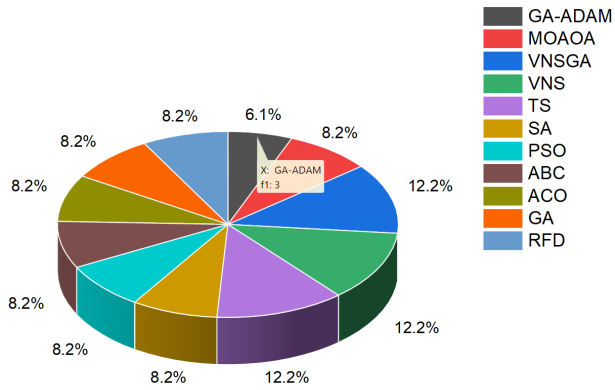


Fig. 12. The f_1 comparison between GA-ADAM and the other ten algorithms.

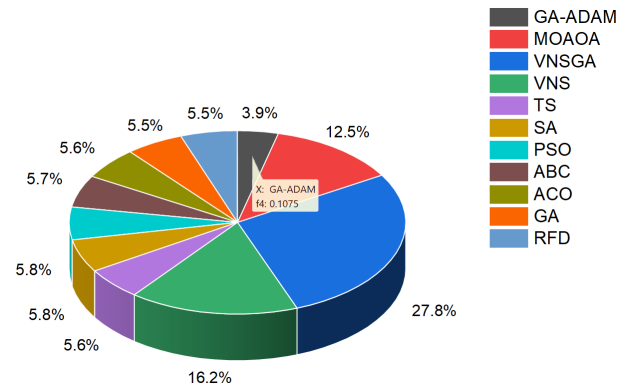


Fig. 15. The f_4 comparison between GA-ADAM and the other ten algorithms.

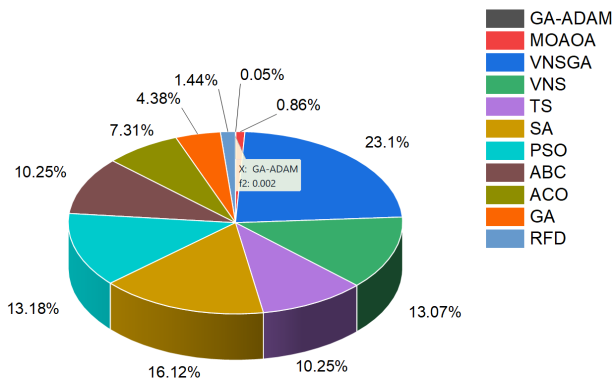


Fig. 13. The f_2 comparison between GA-ADAM and the other ten algorithms.

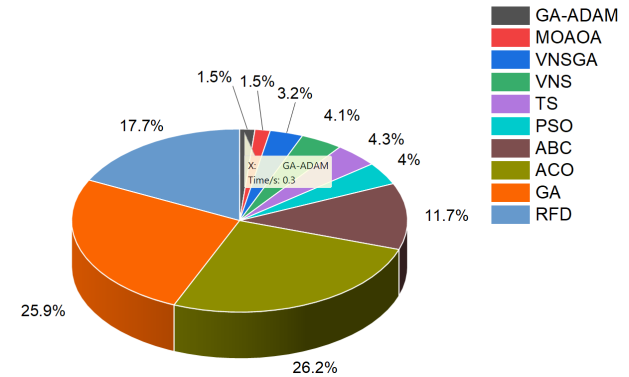


Fig. 16. The time taken to implement GA-ADAM and the other 10 algorithms in a computer is compared here.

of Education of China [Number's grant 18YJC630255]; and the Sichuan Science and Technology Program [Number's grant 2019YFG0285].

REFERENCES

- [1] H. Lödding, R. Riedel, K.-D. Thoben, G. Von Cieminski, and D. Kiritsis, "Advances in production management systems. the path to intelligent, collaborative and sustainable manufacturing: Ifip wg 5.7 international conference, apms 2017, hamburg, germany, september 3-7, 2017, proceedings, part ii," vol. 514, 2017.

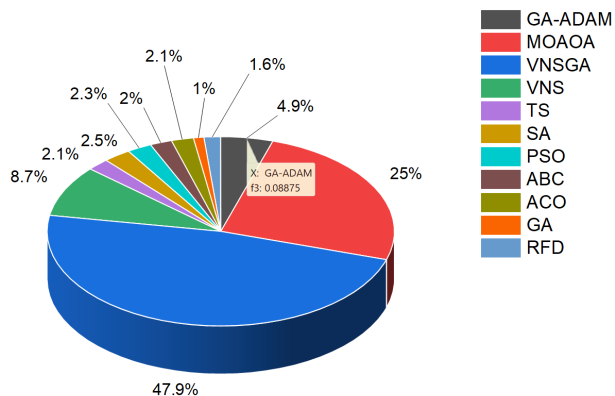


Fig. 14. The f_3 comparison between GA-ADAM and the other ten algorithms.

- [2] J. Chiodo and W. Ijomah, "Use of product self-disassembly technology to improve remanufacturing productivity," in *5th International Conference on Responsive Manufacturing - Green Manufacturing (ICRM 2010)*, IET, 2010.
- [3] G. R. Aase, J. R. Olson, and M. J. Schniederjans, "U-shaped assembly line layouts and their impact on labor productivity: An experimental study," *European Journal of Operational Research*, vol. 156, pp. 698–711, aug 2004.
- [4] C. B. Kalayci, O. Polat, and S. M. Gupta, "A variable neighbourhood search algorithm for disassembly lines," *Journal of Manufacturing Technology Management*, vol. 26, pp. 182–194, mar 2015.
- [5] G. Mouzon and M. B. Yildirim, "A framework to minimise total energy consumption and total tardiness on a single machine," *International Journal of Sustainable Engineering*, vol. 1, pp. 105–116, jun 2008.
- [6] H. Mokhtari and A. Hasani, "An energy-efficient multi-objective optimization for flexible job-shop scheduling problem," *Computers & Chemical Engineering*, vol. 104, pp. 339–352, sep 2017.
- [7] M. Dai, D. Tang, A. Giret, M. A. Salido, and W. Li, "Energy efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm," *Robotics and Computer-Integrated Manufacturing*, vol. 29, pp. 418–429, oct 2013.
- [8] J. Hanigofsky and W. Lackey, "Thermodynamic analysis of the chemical vapor deposition of hfb 2+ sic oxidation resistant composite coatings," *MATERIAL AND MANUFACTURING PROCESS*, vol. 6, no. 3, pp. 521–532, 1991.
- [9] Y. Gao, Q. Wang, Y. Feng, H. Zheng, B. Zheng, and J. Tan, "An energy-saving optimization method of dynamic scheduling for disassembly line," *Energies*, vol. 11, p. 1261, may 2018.
- [10] J. Li, M. Barwood, and S. Rahimifard, "Robotic disassembly for increased recovery of strategically important materials from electrical vehicles," *Robotics and Computer-Integrated Manufacturing*, vol. 50, pp. 203–212, apr 2018.
- [11] F. Shrouf, J. Ordieres-Meré, A. García-Sánchez, and M. Ortega-Mier, "Optimizing the production scheduling of a single machine to minimize

total energy consumption costs,” *Journal of Cleaner Production*, vol. 67, pp. 197–207, mar 2014.

- [12] B. Hu, Y. Feng, H. Zheng, and J. Tan, “Sequence planning for selective disassembly aiming at reducing energy consumption using a constraints relation graph and improved ant colony optimization algorithm,” *Energies*, vol. 11, p. 2106, aug 2018.
- [13] M. Seidi and S. Saghari, “The balancing of disassembly line of automobile engine using genetic algorithm (GA) in fuzzy environment,” *Industrial Engineering and Management Systems*, vol. 15, pp. 364–373, dec 2016.
- [14] E. Tuncel, A. Zeid, and S. Kamarthi, “Solving large scale disassembly line balancing problem with uncertainty using reinforcement learning,” *Journal of Intelligent Manufacturing*, vol. 25, pp. 647–659, dec 2012.
- [15] Z. Zhang and W. Cheng, “Improved heuristic procedure for mixed-model u-line balancing problem with fuzzy times,” in *Proceedings of China Modern Logistics Engineering*, pp. 395–406, Springer, 2015.
- [16] 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.
- [17] M. K. K. Habibi, O. Battaa, V.-D. Cung, and A. Dolgui, “An efficient two-phase iterative heuristic for collection-disassembly problem,” *Computers & Industrial Engineering*, vol. 110, pp. 505–514, aug 2017.
- [18] A. Gngr and S. M. Gupta, “Disassembly sequence plan generation using a branch-and-bound algorithm,” *International Journal of Production Research*, vol. 39, pp. 481–509, jan 2001.
- [19] 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.
- [20] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [21] “River Formation Dynamics Approach for Sequence-Dependent Disassembly Line Balancing Problem,” in *Reverse Supply Chains* (S. M. Gupta, ed.), pp. 307–330, CRC Press, 0 ed., Apr. 2016.
- [22] C. Kalayci, “Solving sequence-dependent disassembly line balancing problem using a hybrid genetic algorithm,” 01 2013.
- [23] C. B. Kalayci and S. M. Gupta, “Ant colony optimization for sequencedependent disassembly line balancing problem,” *Journal of Manufacturing Technology Management*, vol. 24, pp. 413–427, Mar. 2013.
- [24] 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, pp. 7231–7241, Dec. 2013.
- [25] 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, pp. 197–209, Oct. 2013.
- [26] 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.
- [27] C. B. Kalayci and S. M. Gupta, “A tabu search algorithm for balancing a sequence-dependent disassembly line,” *Production Planning & Control*, vol. 25, pp. 149–160, Jan. 2014.
- [28] C. B. Kalayci, O. Polat, and S. M. Gupta, “A hybrid genetic algorithm for sequence-dependent disassembly line balancing problem,” *Annals of Operations Research*, vol. 242, pp. 321–354, July 2016.
- [29] N. A. Siddig and Z. Zhang, “Optimizing Disassembly Line Balancing Problem Using Multi-objective ADAM Optimizer Algorithm,” p. 12.



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.



efficient communication, resource allocation and optimization.

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-



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.