RESEARCH PROGRESS IN MECHANICAL AND MANUFACTURING ENGINEERING VOL. 4 NO. 2
(2023) 91-99

RPMME

Homepage: http://penerbit.uthm.edu.my/periodicals/index.php/rpmme
e-ISSN : 2773 - 4765

Assembly Line Balancing Using Carnivorous Plant Algorithm

Eshmen Ryan Gilong[1], Salleh Ahmad Bareduan[1]*

[1]Faculty of Mechanical and Manufacturing Engineering,
Universiti Tun Hussein Onn Malaysia (UTHM), Parit Raja, 86400, MALAYSIA

* Corresponding Author Designation

DOI: https://doi.org/10.30880/rpmme.2023.04.02.011
Received 01 June 2023; Accepted 10 Nov 2023; Available online 15 December 2023

Abstract: Assembly line balancing (ALB) involves minimising workstations, eliminating bottlenecks, distributing workloads evenly, and maximising line efficiency. Finding a reliable solution for ALB problems, which are often complex, is crucial. Among the available approaches (exact solution, priority-based heuristics, and meta-heuristics), meta-heuristics specifically, have shown promise in tackling complex real-world problems. The chosen approach for this project is the meta-heuristics approach, specifically the carnivorous plant algorithm (CPA), which has shown promising results in solving mechanical engineering design problems, however, it has not been applied to ALB. This project aims to identify fundamental elements of CPA, implement fundamental elements of CPA to solve ALB problem and evaluate the effectiveness of CPA in solving ALB. The implementation of CPA to solve ALB begins with a thorough review of CPA to identify the fundamental elements of CPA. Subsequently, the identified fundamental elements of CPA are adjusted slightly to ensure the successful implementation of CPA in solving ALB problem. The evaluated performance of the result for ALB solution by using CPA is 0.725 and 5.196 for line efficiency and smoothness index, respectively. The performance of the CPA solution is then compared to the performance of the selected common priority rules solutions namely, largest candidate rule (LCR), smallest candidate rule (SCR), and ranked positional weight (RPW). The result of the comparison showed that the CPA solution is similar to the performance of LCR and RPW solution in terms of line efficiency at 0.725. However, when compared to SCR solution, it showed that CPA solution has 16.7% better performance. Additionally, in terms of smoothness index CPA solution emerged as the victor when compared against the selected priority rules by having 10.6% better performance than both LCR and RPW solution, and 52.8% better performance than SCR solution. According to these results, it is shown that ALB solution by using CPA performed better than the selected common priority rules.

Keywords: Assembly Line Balancing, Carnivorous Plant Algorithm, Heuristics

---

**Corresponding author: saleh@uthm.edu.my*

1. Introduction

An assembly line is a type of production system where individual units of work (e.g., stations) are arranged in a sequence and connected by some form of transportation, such as a conveyor belt [1]. Additionally, assembly line balancing (ALB) organizes tasks on an assembly line to achieve equal completion times, promoting a smooth workflow between stations [2]. This approach creates a balanced and efficient production process with minimal workstations and cycle time. The main approaches to solve ALB can be categorised in three categories which are exact method, heuristic – priority-based method, and heuristics – search-based method (i.e., meta-heuristics method) [3].

ALB is an obligatory process particularly in manufacturing and production aspect of industry nowadays. Thus, a reliable solution is needed to solve those ALB problems that often come with high complexity. The heuristic method is often considered superior to the exact solution method due to its ability to efficiently solve complex real-world problems with large dimensions and higher constraints [4]. The difference between heuristics and meta-heuristics is as follows: heuristics are problem-specific solutions, while meta-heuristics are problem-independent solutions that can tackle various real-world problems [4]. Therefore, of the three approaches to solve ALB problem, meta-heuristics approach (i.e., heuristics – search based) is chosen to be implemented in this study.

Recently, there is a new meta-heuristics algorithm that have been developed, which is the Carnivorous Plant Algorithm (CPA) [5]. CPA mimics carnivorous plants' survival tactics, preying on insects for sustenance and utilizing pollination for reproduction [5]. CPA demonstrated promising results in 18 mechanical engineering design problems and ranked first in classical optimisation benchmark functions, as well as in the Congress on Evolutionary Computation (CEC) 2017 test functions [5]. To the best of our knowledge, the application of CPA to solve ALB has yet to be studied in the literature, therefore, the aims of this study are to identify the suitable fundamental elements in CPA for solving assembly line balancing, implement the selected fundamental CPA elements in solving ALB, and evaluate the performance of the CPA solution against the common priority rules.

2. Materials and Methods

This section discusses the data collected, the selected ALB problem studied, the steps to solve ALB problem using the selected fundamental elements of CPA, the tool used, and the method of analysis for the performance of ALB solutions obtained.

2.1 Data Collected

Data for this project is collected through literature review. The collected data includes the number of task elements, task times, and precedence relationships. The precedence diagram collected, and its summary of data are presented in Figure 1 and **Error! Reference source not found.** respectively. Since the collected data did not specify the cycle time, a cycle time of 8 minutes is assumed for this study.

Table 1: Summary of data collected

Task	Task time (min)	Immediate predecessor
A	1	--
B	5	A
C	4	B
D	3	A
E	5	B
F	6	E
G	5	D

Note: The numbered formed task name is change as follows: 1=A, 2=B, 3=C, etc

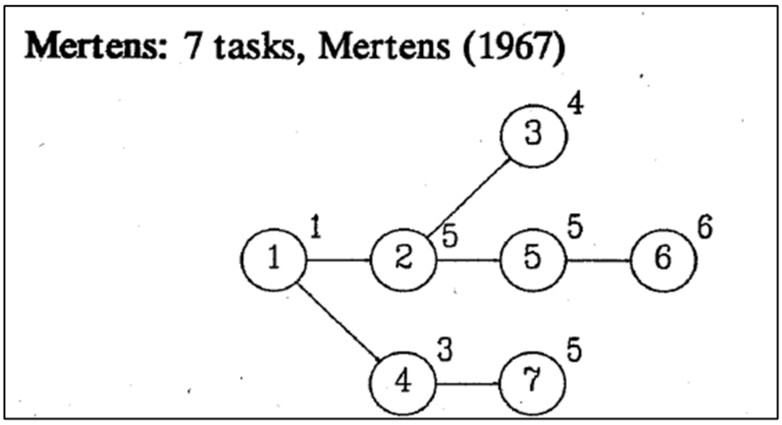


Figure 1: Precedence diagram collected [6]

2.2 Selected ALB problem

The project focuses on studying SALBP-1, a simple ALB problem aimed at minimising workstations within a fixed cycle time [1], [7], [8]. The study considers specific assumptions, which are as follows[1], [7], [8]:

- Production is focused on a single, consistent product.
- There are no alternatives in the way tasks are processed.
- The pace of the assembly line has a fixed, uniform cycle time to meet a specific output quantity.
- The assembly line is in a single, sequential flow and does not have any additional lines or parallel elements.
- The order in which tasks are completed is subject to prior restrictions.
- The time it takes to complete a task is known and exact.
- The only restriction when it comes to task assignments is the prerequisite constraints (i.e., precedence constraint).
- A single task cannot be divided and assigned among multiple workstations.
- All workstations are equipped in the same way with regards to tools and labour.

2.3 Selected Heuristics – Priority Based

The selected priority rules used in this study to solve the selected ALB problem is (i) longest processing time first (i.e., largest candidate rule (LCR)), (ii) shortest processing time first (smallest candidate rule (SCR)), and (iii) ranked position weight (RPW). The results obtained by these priority rules is then analysed and compared to the result obtained by CPA method.

2.3 Carnivorous Plant Algorithm

The chosen elements of CPA (i.e., grouping phase, growth phase, reproduction phase, and recombination phase) are reviewed and implemented to solve the selected ALB problem in this project. The detailed procedure of these elements implemented in this study is as follows:

Step 1: Initialisation and evaluation

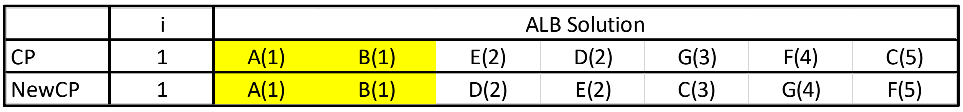
The CPA initialised the initial population of solution randomly using the Excel ALB solution function, then it will evaluate the fitness value of each of the solutions.

Step 2: Classification and grouping

The CPA arranged the ALB solution according to its corresponding fitness value. The best fitness value (i.e., highest, or lowest value depending on the objective function) is rearranged to the top rank and the rest of the ALB solution follows in descending order (i.e., best to worst fitness value).

Step 3: Growth process

In this step, a prey is randomly selected from each group and compared to a predefined attraction rate using a random number. This simulates the interaction between the prey and the carnivorous plant. If the generated number exceeds the attraction rate, the prey undergoes a growth process; otherwise, the carnivorous plant (CP) undergoes a growth process. The growth process involves creating a random new ALB solution while preserving the tasks in the first workstation. The growth process is as illustrated in Figure 2.

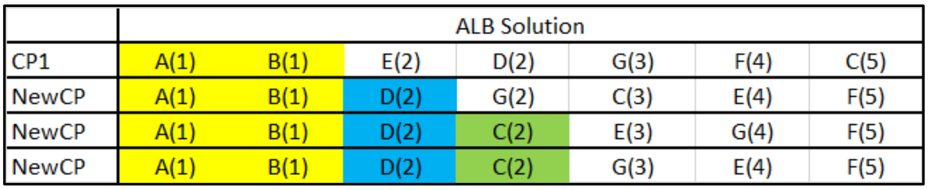


	i	ALB Solution						
CP	1	A(1)	B(1)	E(2)	D(2)	G(3)	F(4)	C(5)
NewCP	1	A(1)	B(1)	D(2)	E(2)	C(3)	G(4)	F(5)

Figure 2: Illustration of growth process

Step 4: Reproduction process

This step applies only to the first rank CP (CP from the first group). The reproduction process resembles the growth process. Initially, a random ALB solution is generated while keeping the same task at the first workstation. However, in subsequent iterations, the task is not maintained in terms of workstation (i.e., workstation 2, workstation 3, workstation 4, etc). Only the first task in the second workstation and the task(s) in the first workstation are preserved while a new ALB solution is generated. This dynamic continuation occurs throughout the iteration process. Refer to Figure 3 for a visual representation of the process.



	ALB Solution						
CP1	A(1)	B(1)	E(2)	D(2)	G(3)	F(4)	C(5)
NewCP	A(1)	B(1)	D(2)	G(2)	C(3)	E(4)	F(5)
NewCP	A(1)	B(1)	D(2)	C(2)	E(3)	G(4)	F(5)
NewCP	A(1)	B(1)	D(2)	C(2)	G(3)	E(4)	F(5)

Figure 3: Illustration of reproduction process

Step 5: Recombination process

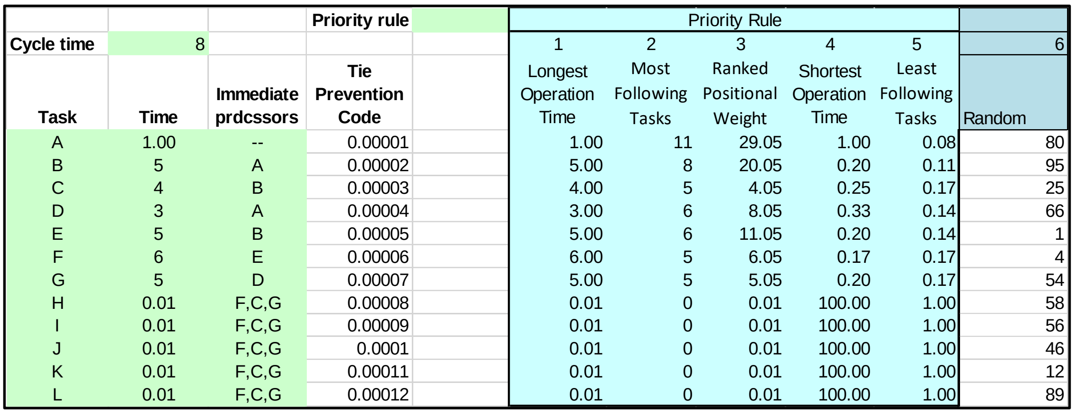
In this step, the newly generated ALB solutions were combined with the initial population. The combined solutions were then sorted by fitness value. The top two solutions were selected, and seven more solutions were randomly generated to complete the initial population for the next iteration if the termination condition is not met. This process enhances solution exploration and prevents getting stuck in local optima.

Step 6: Check termination condition

In this step, if the termination condition does not meet, the procedure will continue repeatedly from step 1 to step 6 until it is met. When the termination condition is met, CPA will present the global best ALB solution.

2.4 Tool Used

The Microsoft Excel Spreadsheets software is used to solve ALB using CPA and selected heuristics. It includes a function called ALB solution template (Figure 4) to obtain the solution. Additionally, Excel is utilized to generate charts for analysing the performance of ALB solutions between CPA and selected heuristics.



			Priority rule		Priority Rule					
Cycle time	8				1	2	3	4	5	6
Task	Time	Immediate prdcssors	Tie Prevention Code		Longest Operation Time	Most Following Tasks	Ranked Positional Weight	Shortest Operation Time	Least Following Tasks	Random
A	1.00	--	0.00001		1.00	11	29.05	1.00	0.08	80
B	5	A	0.00002		5.00	8	20.05	0.20	0.11	95
C	4	B	0.00003		4.00	5	4.05	0.25	0.17	25
D	3	A	0.00004		3.00	6	8.05	0.33	0.14	66
E	5	B	0.00005		5.00	6	11.05	0.20	0.14	1
F	6	E	0.00006		6.00	5	6.05	0.17	0.17	4
G	5	D	0.00007		5.00	5	5.05	0.20	0.17	54
H	0.01	F,C,G	0.00008		0.01	0	0.01	100.00	1.00	58
I	0.01	F,C,G	0.00009		0.01	0	0.01	100.00	1.00	56
J	0.01	F,C,G	0.0001		0.01	0	0.01	100.00	1.00	46
K	0.01	F,C,G	0.00011		0.01	0	0.01	100.00	1.00	12
L	0.01	F,C,G	0.00012		0.01	0	0.01	100.00	1.00	89

Figure 4: Function in Excel ALB solution template

2.5 Method of Analysis

The performance analysis of CPA is conducted by calculating its fitness and comparing it with the performance of selected priority rules. To achieve this, two fitness functions are carefully chosen which are line efficiency and smoothness index. The computation is carried out using Microsoft Excel Spreadsheets software, facilitating accurate and efficient analysis.

The equation for line efficiency (E) ex

$$E = \frac{\sum t}{CT \times W} \qquad \text{Eq. 1}$$

The

$$SI = \sqrt{\sum_{i=1} [(CT - T_i)^2]} \qquad \text{Eq. 2}$$

3. Results and Discussion

This

3.1

The

Table

Method	ALB							Line	Smoothness
CPA	A	B	C	D	G	E	F	0.725	5.196
LCR	A	B	E	D	F	G	C	0.725	5.745
SCR	A	D	B	C	E	G	F	0.604	7.937
RPW	A	B	E	D	F	G	C	0.725	5.745

Note:

Next,

Table

Station	Task	Cumulative	Idle
1	A	6	2
2	C	7	1
3	G	5	3
4	E	5	3
5	F	6	2
Total		29	11

Table

Station	Task	Cumulative	Idle
1	A	6	2
2	E	8	0
3	F	6	2
4	G	5	3
5	C	4	4
Total		29	11

Table

Station	Task	Cumulative	Idle
1	A	4	4
2	B	5	3
3	C	4	4
4	E	5	3
5	G	5	3
6	F	6	2
Total		29	19

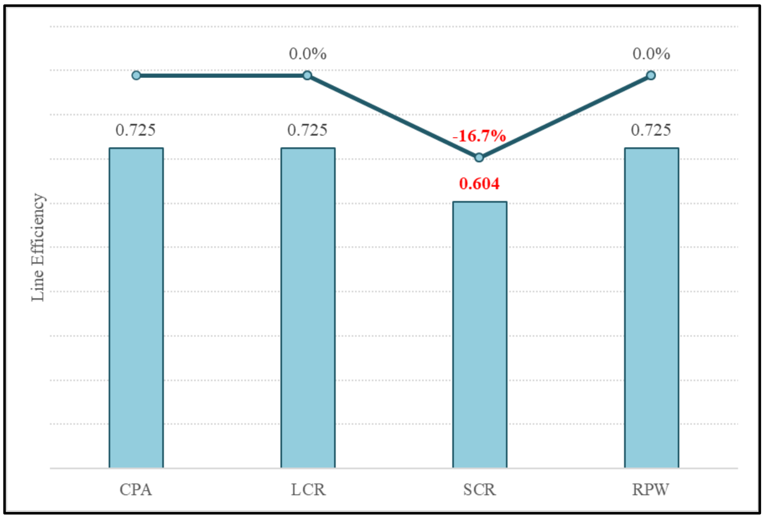
Table

Station	Task	Cumulative	Idle
1	A	6	2
2	E	8	0
3	F	6	2
4	G	5	3
5	C	4	4
Total		29	11

3.2

This

Figure



Figure

The difference in line efficiency performance between the CPA and SCR solutions primarily stems from the discrepancy in the number of workstations utilized by each method. Table 2 illustrates that CPA, LCR, and RPW employ five workstations, whereas SC

Figure 6 compares the smoothness index of the ALB solution implemented using CPA and the selected heuristics. The results indicate that the performance of the CPA solution surpasses that of the selected heuristics, as shown in Figure 6. Specifically, the CPA solution achieves a smoothness index value of 5.196, outperforming LCR and RPW by 10.6%. Additionally, the CPA solution demonstrates superior line efficiency compared to SCR, which only achieves a smoothness value of 7.937. In fact, the CPA solution exhibits an impressive 52.8% improvement in smoothness index over the SCR solution.

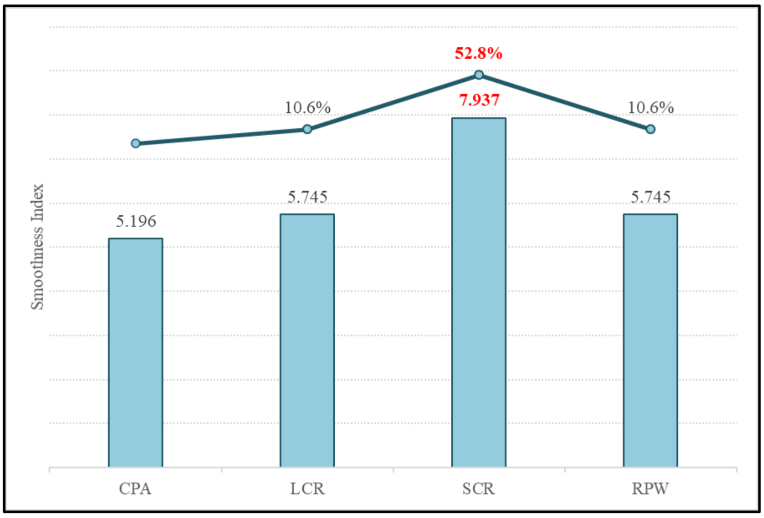


Figure 6: Chart of comparison of smoothness index between CPA and the selected heuristics

The significant difference in smoothness performance between the CPA and SCR solutions can be primarily attributed to the variance in their total idle time. Table 3 demonstrates that the CPA method has a total idle time of 11 minutes, while the SCR method has a total idle time of 19 minutes, as indicated in Table 5. However, the dissimilarity in smoothness performance between the CPA solution and the LCR and RPW solutions cannot be solely explained by the total idle time. This is evident from the identical values presented in Table 3, Table 4, and Table 6. The main factor lies in the longest idle time available within each workstation. As evidenced by Table 3, Table 4, and Table 6, the CPA method, as well as the LCR and RPW methods, offer the longest idle time of 3 minutes and 4 minutes, respectively.

4. Conclusion

After a comprehensive review of CPA, the appropriate fundamental elements for solving ALB were identified and categorized into four phases: grouping, growth, reproduction, and recombination. Following the four identified fundamental phases outlined in Section 2, a procedure consisting of six steps was applied: initialization and evaluation, classification and grouping, growth process, reproduction process, recombination process, and termination condition check. These procedures were subsequently applied to address the selected ALB problem. Based on the results presented in Section 3, it was observed that the ALB solution using CPA outperformed the ALB solution using the selected heuristics. This was supported by a significant percentage difference in the evaluated smoothness index, with the CPA solution demonstrating a 10.6% improvement over both the LCR and RPW solutions, and a 52.8% improvement over the SCR solution.

In this study, the chosen ALB problem is SALB-1, which is an academic problem. The data collected for this project is also purely academic, sourced from literature. Therefore, it is recommended to implement CPA in real-world assembly line environments using real-world data, addressing real-world problems. This implementation can provide valuable insights into the practical application of CPA and any challenges or limitations that may arise in various industrial contexts.

Moreover, further comparative studies are recommended to evaluate the performance of CPA in the context of ALB, comparing it with other commonly used optimization algorithms (e.g., genetic algorithms, ant colony optimization, and particle swarm optimisation). These studies can offer valuable insights into the strengths and weaknesses of CPA relative to established methods.

Acknowledgement

The authors would also like to thank the Ministry of Higher Education Malaysia for supporting this research under Fundamental Research Grant Scheme Vot No. FRGS/1/2018/TK03/UTHM/02/9 and partially sponsored by Universiti Tun Hussein Onn Malaysia.

References

[1] N. Boysen, M. Fliedner, and A. Scholl, “A classification of assembly line balancing problems,” *Eur J Oper Res*, vol. 183, no. 2, pp. 674–693, Dec. 2007

[2] S. K. M. Hossain, “A NOVEL ALGORITHM FOR SOLVING THE MULTI-OBJECTIVE ASSEMBLY LINE BALANCING PROBLEM,” 2016.

[3] B. Micieta and V. Stollmann, “Assembly Line Balancing,” in *DAAAM International scientific book 2011*, DAAAM International Vienna, Vienna 2011, 2011

[4] D. Kapoor, S. Anand, J. Kulkarni, and A. Abraham, “Intelligent Systems Reference Library 187 Optimization Models in Steganography Using Metaheuristics,” 2020. [Online]. Available: http://www.springer.com/series/8578

[5] K. M. Ong, P. Ong, and C. K. Sia, “A carnivorous plant algorithm for solving global optimization problems,” *Appl Soft Comput*, vol. 98, Jan. 2021,

[6] A. Scholl, “Data of Assembly Line Balancing Problems,” 1993, Accessed: Jan. 09, 2023. [Online]. Available: https://assembly-line-balancing.de/wp-content/uploads/2017/01/Scholl-1993-ALBData.pdf

[7] A. Scholl and C. Becker, “State-of-the-art exact and heuristic solution procedures for simple assembly line balancing,” in *European Journal of Operational Research*, Feb. 2006, pp. 666–693.

[8] S. F. Lakrash, “A Combined Simulation Heuristic Approach to Optimize a Single-product Straight Assembly Line,” 2021.