[1,*]Liyong Zhang

[2]Wentao Xu

[3]Guanbo Wang

[4]Tongjie Li

[5]Juan Wang

[6]Yehu Jiang

[7]Hanlin Zhang

[8]Le Bao

[9]Kyoosik Shin

Research on Assembly Sequence Planning of Hybrid Power Transmission Device Based on Improved Genetic Algorithm

JES

Journal of Electrical Systems

Abstract: - Aiming at the complex optimisation problem involved in the assembly process of DM-i hybrid system, an improved genetic algorithm based on the assembly sequence planning problem is proposed to be investigated. Two matrices, assembly priority and assembly space interference, are used to constrain the assembly relationship of parts, and the feasibility, optimality and flexibility of assembly are considered; the initial population of the genetic algorithm is optimised in terms of algorithmic improvement, and inverse learning is used to generate the initial population and an elite inverse learning mechanism is introduced to avoid the algorithm from falling into a local optimal solution; the search strategy of the algorithm is improved, which consists of four main steps, i.e., binary tournament selection, partial matching crossover, exchange mutation and elite individual retention strategy. The feasibility and superiority of the proposed improved genetic algorithm in solving the assembly sequence planning problem of DM-i hybrid system are verified by example algorithms and on-site assembly verification.

Keywords: Assembly Sequence Planning, Hybrid Transmission Device, Elite Inverse Learning, Genetic Algorithm.

I. INTRODUCTION

Amid the increasing demand in society for sustainable development and environmentally friendly technologies, hybrid systems, as an innovative solution integrating internal combustion engines and electric motors, provide robust support for achieving energy efficiency and reducing emissions. Currently, the escalating stringency of environmental regulations necessitates the manufacturing industry to adopt more efficient and intelligent assembly methods to decrease energy consumption and mitigate adverse environmental impacts. The assembly sequence planning of hybrid powertrains involves intricate optimization challenges, hence the urgent need for effective algorithms and methods to meet the dual challenge of production efficiency and environmental standards amidst evolving market demands. Against this backdrop, intelligent assembly emerges as a pivotal technology in the field of assembly process planning [1], offering robust support for the manufacturing industry's sustainable development in the new era and facilitating the realization of personalized and flexible production.

Hao Bo et al. [2] employed a matrix model to optimize the initial population and enhanced the crossover and mutation patterns of the genetic algorithm for resolution purposes. Feng Zengxi et al. [3] introduced an elite reverse learning strategy into the sparrow search algorithm, where elite individuals undergo reverse learning to generate inverse solutions. Ding Ruicheng et al. [4] guided the search paths of scarab individuals based on the elite individuals in the population, incorporating an elite reverse learning strategy to enhance population diversity. Li Zhijie et al. [5] tackled genetic parameter initialization of the initial population from the perspectives of genetic evolution iterations and individual fitness values. Qin Hongbin et al. [6] implemented updates using adaptive step flight, standard step flight, and crossover, as well as mutation methods. Chen Yanbin et al. [7] established the objective function of assembly sequence planning based on evaluation metrics and corresponding

---

[1] Department of Mechanical Engineering, Anhui Science and Technology University, Chuzhou 233100, China

[2] Department of Mechanical Engineering, Anhui Science and Technology University, Chuzhou 233100, China

[3] Jiangsu Zhonggong High end Equipment Research Institute Co., Ltd., Taizhou 225300, China

[4] Department of Mechanical Engineering, Anhui Science and Technology University, Chuzhou 233100, China

[5] Department of Mechanical Engineering, Anhui Science and Technology University, Chuzhou 233100, China

[6] Anhui Zhongxin Technology Co., Ltd., Chuzhou 233100, China

[7] Department of Mechanical Engineering, Anhui Science and Technology University, Chuzhou 233100, China

[8] Department of Mechatronics Engineering, Hanyang University, Ansan 15588, South Korea

[9] Robotics Department, Hanyang University ERICA, Ansan 15588, South Korea

*Corresponding author: Liyong Zhang

constraints, utilizing genetic algorithms to solve the objective function. Mei Zhen et al. [8] focused on improving algorithm encoding methods, initial population generation, selection strategies, adaptive adjustments of crossover and mutation probabilities, and immigration strategies within a multi-population collaborative evolution setting. Xu Qiang et al. [9] devised an adaptive adjustment mechanism for crossover and mutation probabilities to enhance the genetic algorithm. Jiang Rui et al. [10] ensured population diversity throughout the iteration process by adaptively adjusting mutation probabilities based on population distinctiveness. Huang Xinlin et al. [11] applied a roulette selection method, utilizing a Gaussian probability matrix to influence gene mutations within individuals. Lu [12] used an ant colony algorithm to improve assembly line efficiency by planning assembly sequences and balancing workloads. Zhang et al. [13] used a firefly-inspired algorithm to find the best way to assemble products. Murali [14] combined simulated annealing and artificial immune algorithms to enhance the simulated annealing algorithm with the local search capabilities of the artificial immune algorithm. Li et al. [15] used a Harmony Search algorithm to find the best way to assemble products. Li et al. [16] investigated an assembly sequence planning method based on the fireworks algorithm. Gao et al. [17] used an adaptive gravitational search algorithm to find the best way to assemble products. Mishra [18] developed an assembly sequence optimization method based on the pollen grain algorithm, initially applied to discrete optimization issues like assembly sequence optimization. Su et al. [19] optimized hybrid assembly sequence planning and assembly line balancing using a particle swarm optimization algorithm. GUO et al. [20] introduced an improved cat swarm optimization algorithm to investigate assembly sequence planning problems. Li et al. [21] proposed an enhanced Harmony Search algorithm to tackle assembly sequence planning problems. Somayé Ghandi et al. [22] presented an appropriate optimization algorithm called Break Local Search to achieve high-quality solutions for assembly sequence planning. Li et al. [23] employed an assembly sequence matching planning method to provide optimized assembly sequences. Anil Kumar Gulivindala et al. [24] introduced a novel sub-assembly detection stability concept to minimize the complexity of parallel assembly sequence planning. Liu et al. [25] put forward an enhanced simulated annealing algorithm for optimizing assembly sequence planning.

In addressing the assembly sequence planning problem for the DM-i hybrid system, the choice of genetic algorithms for research and enhancement is motivated by their efficient solving capabilities. Genetic algorithms, as an optimization algorithm, excel in solving high-dimensional nonlinear optimization problems without reliance on gradient information. Assembly sequence planning dilemmas typically involve the combination and arrangement of multiple components, presenting a high level of intricacy. Improved genetic algorithms can evolve populations and genetic operations to search for superior assembly sequences, thereby enhancing problem-solving efficiency. With strong diversity and exploratory characteristics, genetic algorithms, through genetic operations like crossover and mutation, can introduce new gene combinations, increasing individual diversity. This aids in avoiding algorithmic entrapment in local optima, boosts global search capabilities, and facilitates the discovery of superior assembly sequences. In assembly sequence planning problems, considerations typically extend to the assembly constraints among various components, such as assembly order and interference. Enhancements to genetic algorithms involve incorporating these constraints by designing fitness functions and genetic operations to generate assembly sequences that adhere to such restrictions. Algorithmic improvements and optimizations, such as introducing new genetic operations and adjusting parameters, can be customized to fit the unique characteristics of the problem. This adaptability enables improved genetic algorithms to address assembly sequence planning problems of varying scales and complexities effectively, showcasing robust adaptability.

In conclusion, employing improved genetic algorithms for assembly sequence planning research proves effective in solving intricate assembly sequence dilemmas by considering assembly constraints. This approach exhibits efficient solving capabilities, diversity, exploratory characteristics, as well as scalability and flexibility. Focusing on the hybrid power transmission device, this study improves upon traditional genetic algorithms and integrates them with Matlab software to develop a tailored assembly sequence planning program for hybrid power transmission devices. Verification of the proposed algorithm is conducted through simulation of the obtained sequences.

II. Constraints in Assembly Planning

A. Assembly Priority Matrix

Within an assembly, there exist assembly priority relationships between components of different natures and different assembly orientations of the same component. In order to better express these relationships and enhance

the operational efficiency of the algorithm, assembly priority relationships can be structured as a matrix that satisfies certain conditions. Create an n×n matrix A1 to represent assembly priority relationships.

When constructing matrix A, the values of its elements can be determined based on various conditions. These conditions may include functional dependencies between components, physical constraints on assembly directions, and more. By setting different priorities, it is possible to indicate which components should be assembled first and which can be assembled later. The expression for the assembly priority matrix A1 is given by:

$$A_1 = \begin{array}{c} \\ P_1 \\ P_2 \\ \vdots \\ P_n \end{array} \begin{array}{cccc} P_1 & P_2 & \cdots & P_n \\ \begin{bmatrix} I_{11} & I_{12} & ... & I_{1n} \\ I_{21} & I_{22} & ... & I_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ I_{n1} & I_{n2} & ... & I_{nn} \end{bmatrix} \end{array}$$

(1)

Where:
- If $I_{ij}=1$, it signifies that component i has a higher assembly priority than component j.
- If $I_{ij}=0$, it indicates that there is no priority assembly relationship between component i and component j.

B. Assembly Space Interference Matrix

During the assembly process, interference may occur between components due to the assembly sequence and spatial positioning. To determine whether a particular component to be assembled will interfere with components that have not been assembled, an assembly interference matrix can be utilized. The matrix for assembly interference is that records the interference relationships between components, and by analyzing the elements of the matrix, one can ascertain the presence of interference. Create an n×n matrix A_2 to represent assembly interference relationships. The expression for the assembly interference matrix A_2 is given by:

$$A_2 = \begin{array}{c} \\ P_1 \\ P_2 \\ \vdots \\ P_n \end{array} \begin{array}{cccc} P_1 & P_2 & \cdots & P_n \\ \begin{bmatrix} I_{11(o)} & I_{12(o)} & ... & I_{1n(o)} \\ I_{21(o)} & I_{22(o)} & ... & I_{2n(o)} \\ \vdots & \vdots & \vdots & \vdots \\ I_{n1(o)} & I_{n2(o)} & ... & I_{nn(o)} \end{bmatrix} \end{array}$$

(2)

Where, o represents the six orthogonal directions (±x, ±y, ±z). When Iij=1, it signifies interference between component i and component j in the assembly direction o. When Iij=0, it indicates no interference between component i and component j in the assembly direction o.

III. ENHANCEMENTS IN GENETIC ALGORITHMS

A. Optimization of Initial Population

After encoding the research subject for genetic algorithm application, it is essential to generate a set of initial populations as the algorithm's starting point. The size and selection of this initial population significantly impact the final outcomes. If the initial population size is excessively large, it can escalate computational complexity and prolong convergence time. Conversely, opting for a small initial population may lead the objective function to prematurely converge to a local optimal solution. The selection of the initial population can involve random generation or following certain heuristic experiences. A well-chosen initial population can augment search efficiency, facilitating a faster convergence towards a stable state. To ensure algorithmic operational efficiency, as well as diversity and rationality within the initial population, this study opts for a method rooted in reverse learning to generate the initial population. Moreover, to avert algorithmic entrapment in local optimal solutions, an elite reverse learning mechanism is introduced.

Firstly, a population P(N) is randomly generated. Subsequently, a new population, NP(N), is created following the reverse learning approach. These two populations are then combined to form a novel population, S(N). Individuals within S(N) are sorted in ascending order based on their fitness values. The N individuals with the highest fitness values are selected as the initial population. The strategy of reverse learning refers to the definition of each chromosome within the population P(N) using real number encoding, as outlined in the following formula:

$$E = (e_1, e_2, e_3, \cdots, e_i, \cdots, e_n)$$ (3)

Where, $e_i \in [a_i, b_i]$.

the reverse learning point is

$$\bar{E} = (\bar{e}_1, \bar{e}_2, \bar{e}_3, \cdots, \bar{e}_i, \cdots, \bar{e}_n) \tag{4}$$

The generation process for each component of E is expressed as follows:

$$\bar{e}_i = k(a_i + b_i) - e_i \tag{5}$$

Where:

-k is the elite reverse coefficient in the range [0,1];

- e_i denotes a gene in chromosome E;

$\bar{e}$ represents the reverse learning point corresponding to the gene e_i;

- a_i signifies the lower limit of gene e_i;

-b_i represents the upper limit of gene e_i.

B. Enhancements in Search Strategy

In the context of assembly sequence planning for hybrid powertrain systems, improvements are made to traditional genetic algorithms. The genetic operators encompass four primary steps: selection, crossover, mutation, and termination.

By employing the method of binary tournament selection, individuals are chosen, allowing for the comparison of two individuals at a time and selecting the one with a better fitness. This approach offers the flexibility to control selection pressure by adjusting the number of individuals compared each time, thereby regulating the intensity of competition during selection. When the number of individuals selected is small, the selection pressure is lower, advantageous for retaining diversity within the population. Conversely, with a larger selection size, higher selection pressure accelerates the convergence of excellent individuals. This method exhibits strong exploratory and diverse characteristics. As only two individuals are compared at a time, even weaker candidates have opportunities to be selected under certain conditions, preserving population diversity and steering clear of local optima, thus enhancing the algorithm's global search capabilities. Furthermore, this selection method is adaptive: the fitness value of each individual directly influences its probability of being chosen. Individuals with higher fitness have a competitive advantage during selection, increasing the chances of superior individuals being selected and facilitating the propagation and retention of excellent genes. This selection approach is robust against the form of the fitness function and is well-suited for addressing assembly sequence planning problems.

The crossover operation stands as a crucial method for generating new individuals and enriching the population of samples, marking a key distinction from other algorithms. Prior to carrying out the crossover operation, individuals within the population are paired, typically through random pairing methods, followed by the actual crossover operation within each pair. In the realm of assembly sequence planning problems, it is essential to strike a balance: avoiding excessive disruption of certain advantageous traits within individuals while still fostering the creation of a considerable number of new individuals. Partial matching crossover enables the retention of excellent gene segments from parental individuals. By selecting crossover points between two individuals and swapping gene segments between these points, a portion of superior genetic information from the parental individuals can be preserved, facilitating the accelerated propagation and retention of exceptional genes. Through the crossover operation, not only can superior gene segments be preserved, but new gene segments can also be introduced, thereby enhancing individual diversity. This aids in preventing the algorithm from falling into local optima, thereby boosting the algorithm's global search capability. Simultaneously, it can expedite the algorithm's convergence rate. By preserving excellent gene segments and introducing new ones, the spread and combination of superior genes are accelerated, hastening the algorithm's convergence speed and facilitating the discovery of optimal solutions more efficiently. In assembly sequence planning, gene continuity holds significant importance. Partial matching crossover retains gene continuity, ensuring that individuals maintain a rational gene sequence post-crossover. This feature contributes to generating assembly sequences that adhere to assembly requirements, thereby enhancing the optimization performance of the algorithm.

By employing the partially matched crossover method, for the paired two chromosomes, two crossover points are randomly selected to determine the matching segment. According to the mapping relationship of genes on each chromosome in the matching segment, the values of genes inside and outside the matching segment are replaced to generate a new individual. This approach will be utilized for the crossover in this study. The operational procedure is illustrated in Figure 1. If the encoding before the crossover already exists in C1'' or C2'', it will be skipped.

Figure 1: Schematic Diagram of the Partial Matching Crossover Process

In genetic algorithms, the setting of the crossover probability (P_c) is of paramount importance for the evolution speed of the entire population. When P_c is too large, it runs the risk of disrupting valuable genetic information. Conversely, if P_c is too small, it can diminish the algorithm's search efficiency, resulting in a slow population evolution. Moderately increasing P_c in the early stages of evolution can expand the global search scope and accelerate the population update rate. Conversely, in the later stages of evolution, as the population stabilizes, moderately decreasing P_c can help preserve superior gene structures. Moreover, considering that the crossover operation might alter or disrupt gene structures, it is advisable to slightly reduce the crossover probability for individuals with high fitness levels. In contrast, for individuals with lower fitness levels, increasing the crossover probability can facilitate optimization. In conclusion, by adjusting the crossover probability, one can effectively balance global exploration and the preservation of superior genes. The adjustment mechanism is outlined as follows:

$$P_{ci} = \begin{cases} P_{c\max} - \dfrac{(P_{c\max} - P_{c\min})(f' - f_{avg})}{f_{c\max} - f_{avg}} & f' \geq f_{avg} \\ P_{c\max} & f' < f_{avg} \end{cases} \tag{6}$$

$$P_{c\max} = \begin{cases} 0.9 & g \leq G/4 \\ 0.7 & G/4 \leq g \leq 3G/4 \\ 0.5 & 3G/4 < g \leq G \end{cases} \tag{7}$$

Where:

- P_{ci} is the probability that individual i undergoes the crossover operator;

- $P_{c\max}$ is the maximum crossover probability range that can be reached throughout the genetic operation procedure;

- $P_{c\min}$ is the bottom bound of the genetic operation process's crossover probability range;

- f' is the fitness value of the patient undergoing the crossover operation who is the more fit of the two;

- f_{avg} is the population's average level of fitness for each generation;

- $f_{c\max}$ is the population's maximal fitness value at this time;

- G is the maximum number of iterations in the iterative process;

- g is the number of the current iteration.

Mutation refers to the alteration of chromosome genes within an individual during the genetic iteration process, specifically involving the substitution of gene values in the chromosome to generate new individuals. Unlike crossover, which affects two chromosomes simultaneously, mutation acts as a supplementary method for generating new individuals by operating solely on a single chromosome. Mutation plays a pivotal role in determining the genetic algorithm's local search capabilities, distinct from crossover's influence on global search abilities. Through a combination of genetic and mutational operations, the search prowess of genetic algorithms is honed, facilitating the rapid and precise identification of optimal solutions.

In the realm of symbol encoding, it is common practice to employ exchange mutation. Therefore, this study opts for exchange mutation as the method for mutating operations. This choice is made due to its capability of introducing localized alterations within the gene sequence, thereby leading to changes in the assembly order. By exchanging the positions of two genes in the sequence, it becomes possible to modify the order of adjacent elements in the assembly sequence, consequently giving birth to fresh assembly sequences. This process effectively boosts the diversity within the population, facilitates the transfer and preservation of superior genes, and ultimately elevates the fitness level of the population. It is imperative to ensure that the two positions

subjected to exchange mutation are indeed distinct and thoughtfully selected within a reasonable range to prevent the introduction of futile mutations.

Similarly, the setting of the mutation probability (P_m) plays a crucial role in determining the likelihood of individual mutations. Increasing P_m appropriately can enhance the diversity within the population, providing individuals with a chance to escape local optima. However, setting P_m too high may lead the algorithm towards random searching, diminishing efficiency. Conversely, a P_m value that is too low may hinder the generation of new individual structures. Therefore, it is essential to establish an adaptive adjustment formula based on the genetic evolution epochs and the fitness function values of the population individuals, as illustrated below:

$$P_{mi} = \begin{cases} P_{m\max} - \dfrac{(P_{m\max} - P_{m\min})(f - f_{avg})}{f_{m\max} - f_{avg}} & f \geq f_{avg} \\ P_{m\max} & f < f_{avg} \end{cases} \tag{8}$$

$$P_{c\max} = \begin{cases} 0.005 & g \leq G/4 \\ 0.004 & G/4 \leq g \leq 3G/4 \\ 0.003 & 3G/4 < g \leq G \end{cases} \tag{9}$$

P_{mi} - Probability of individual i undergoing the crossover operator;

$P_{m\max}$ - Upper bound of the spectrum of mutation probabilities during genetic operation;

$P_{m\min}$ - lower bound of the spectrum of mutation probabilities during genetic operation;

f - The mutation's fitness value.

In order to preserve exceptional individuals to the greatest extent possible, an elite individual preservation strategy is utilized during the selection process. This strategy involves directly duplicating the individual with the highest fitness function value into the parent population set for crossover mating. An elite individual preservation mechanism is used to replace the individual with the lowest fitness function value in the post-crossover population with the elite individual from before the crossover, once all parent individuals have undergone the crossover operator. This process aims to eliminate lower-quality individuals, thereby ensuring the preservation of elite individuals.

C. Improved Genetic Algorithm Framework

Based on the enhancements outlined for the genetic algorithm, we can derive the structure of the improved genetic algorithm as depicted in Figure 2.
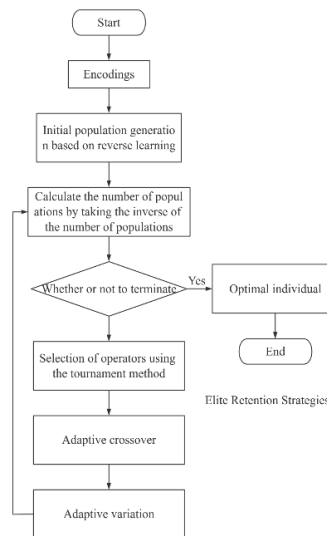

Figure 2: Improved Genetic Algorithm Flowchart

IV. INSTANCE VERIFICATION ANALYSIS

Taking a hybrid power transmission device as the subject of study, a program was developed using the Matlab tool to validate the effectiveness of the algorithm improvement. Assembly connection diagrams and undirected graphs were employed to illustrate the assembly precedence relationships among parts and determine the presence of assembly interferences, as shown in Figure 3 and Figure 4. Part number 36 serves as the

foundational component for the entire assembly process, being the first part determined for assembly and acting as the carrier for the assembly of other parts.
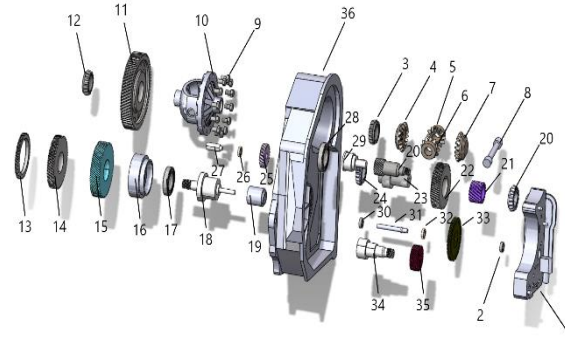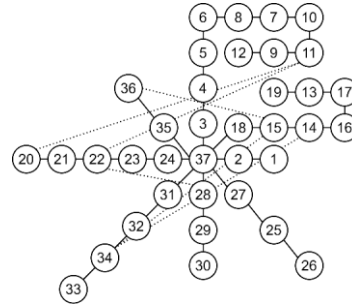

Figure 3: DM-i Hybrid System Structure Diagram

Figure 4: Assembly Relationship Undirected Graph

Subsequently, the obtained assembly relationship information was utilized to generate the assembly precedence matrix and assembly spatial interference matrix as constraints for sequence planning, integrated into the program for solution. Through enhancing the optimization of the initial population and search strategies in the genetic algorithm, the assembly planning problem for the hybrid power transmission device was addressed. A comparative analysis was conducted against the initial genetic algorithm, showcasing the convergence curves of the algorithms as depicted in Figure 5. By recording the optimal assembly sequence derived through the improved genetic algorithm, an assembly timeline Gantt chart was developed, as illustrated in Figure 6.
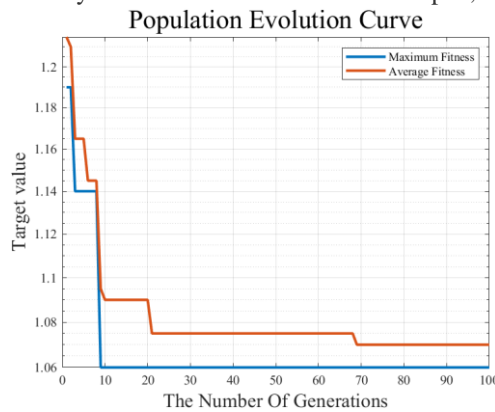

Figure 5: Comparison of Convergence Curves of Improved Genetic Algorithms

After experimental validation, it was observed that with the constraints of the assembly precedence mechanism and assembly spatial interference mechanism in place, the improved genetic algorithm, following enhancements in initial population optimization and search strategies, exhibited faster convergence and superior solution quality compared to the basic genetic algorithm. The optimal assembly sequence derived from the improved genetic algorithm is as follows: $P_{37} \rightarrow P_2 \rightarrow P_1 \rightarrow P_{28} \rightarrow P_{29} \rightarrow P_{30} \rightarrow P_3 \rightarrow P_4 \rightarrow P_5 \rightarrow P_6 \rightarrow P_8 \rightarrow P_7 \rightarrow P_{10} \rightarrow P_{24} \rightarrow P_{23} \rightarrow P_{22} \rightarrow P_{21} \rightarrow P_{11} \rightarrow P_9 \rightarrow P_{12} \rightarrow P_{20} \rightarrow P_{18} \rightarrow P_{15} \rightarrow P_{14} \rightarrow P_{16} \rightarrow P_{17} \rightarrow P_{13} \rightarrow P_{19} \rightarrow P_{35} \rightarrow P_{36} \rightarrow P_{31} \rightarrow P_{32} \rightarrow P_{34} \rightarrow P_{33} \rightarrow P_{27} \rightarrow P_{25} \rightarrow P_{26}$. By employing this optimal assembly sequence, the on-site assembly of the hybrid power transmission device was carried out, as depicted in Figure 7, with an assembly time of T = 1 hour 15 minutes.
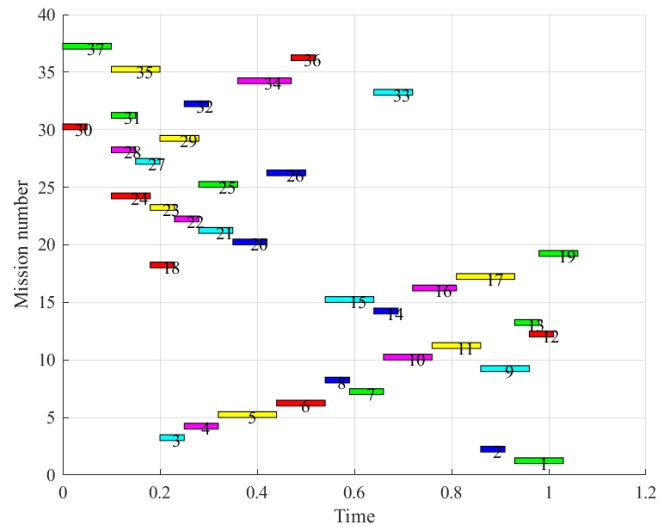
Figure 6: Gantt Chart for Assembly Time

(a) On-site assembly   (b) On-site assembly

Figure 7: Partial On-Site Assembly Drawings

V.   CONCLUSIONS

Research was conducted on the assembly sequence planning of the hybrid power transmission device through the establishment of assembly constraint mechanisms and algorithm optimization. Within the framework of improving the genetic algorithm, the feasibility of these enhancements was ensured, yielding significant optimizations compared to the basic genetic algorithm. This led to accelerated convergence rates and improved solution quality. The results indicate that the improved genetic algorithm is well-suited for assembly sequence planning of hybrid power transmission devices. The optimal assembly sequence derived from the enhanced algorithm resulted in a 20% reduction in assembly time. Furthermore, utilizing Matlab as the programming tool, a software application was designed for algorithm comparison and assembly sequence planning of hybrid power transmission devices.

REFERENCES

[1] Wang H, Rong Y M, Xiang D. Mechanical assembly planning using ant colony optimization. Computer Aided Design, 2014, 47: 59-71.

[2] Hao Bo, Xu Dongping, Wang Mingyang, et al. Intelligent Planning of Aircraft Wing Assembly Sequence Based on Improved Genetic Algorithm. Modular Machine Tool & Automatic Manufacturing Technique, 2021, (06): 135-137+140. DOI:10.13462/j.cnki.mmtamt.2021.06.031.

[3] Feng Zengxi, Li Shiyan, Zhao Jintong, et al. Sparrow Search Algorithm Based on Elite Reverse Learning Strategy. Computer Simulation, 2023, 40 (01): 378-381+393.

[4] Ding Ruicheng, Liu Xue, Yang Menggang, et al. Improved BAS Algorithm Based on Elite Opposition-based Learning. Software Guide, 2023, 22 (05): 57-63.

[5] Li Zhijie, Shi Haoqi, Li Changhua, et al. Image Center Layout Optimization Method Based on Improved Genetic Algorithm. Journal of System Simulation, 2022, 34 (06): 1173-1184. DOI:10.16182/j.issn1004731x.joss.20-1043.

[6] Qin Hongbin, Wang Lingjun, Tang Hongtao, et al. Application Research of Improved Cuckoo Search Algorithm in Assembly Sequence Planning. Machine Tool & Hydraulic, 2024, 52 (02): 12-17.

[7] Chen Yanbin, Li Huanqing. Research on Assembly Sequence Optimization of Aircraft Structural Parts Based on Improved Genetic Algorithm. Machine Building & Automation, 2023, 52 (06): 221-225. DOI:10.19344/j.cnki.issn1671-5276.2023.06.053.

[8] Mei Zhen, Gong Jiacheng, Gao Yichao, et al. Integrated Optimization of Structure-Control System Based on Improved Adaptive Multi-Population Genetic Algorithm. Journal of Central South University (Natural Science Edition), 2024, 55 (02): 799-809.

[9] Xu Qiang, Xu Jianlei, Hu Yanhai, et al. Trajectory Optimization of Robotic Arm Based on Improved Simulated Annealing Genetic Algorithm. Journal of System Simulation, 1-10[2024-03-18]. https://doi.org/10.16182/j.issn1004731x.joss.23-1148.

[10] Jiang Rui, Han Yao, Zhang Dawei, et al. Parallel test task scheduling based on improved adaptive genetic algorithm. Journal of Ordnance Equipment Engineering, 2023, 44 (09): 298-305.

[11] Huang Xinlin, Zhang Longfei, Tang Xiaowei, et al. Path Planning Method for Household Appliance Recycling Vehicle Based on Improved Genetic Algorithm. Journal of Tongji University (Natural Science Edition), 2024, 52 (01): 27-34.

[12] Lu C, Yang Z. Integrated assembly sequence planning and assembly line balancing with ant colony optimization approach. The International Journal of Advanced Manufacturing Technology, 2016, 83(1): 243-256.

[13] Zhang Z, Yuan B, Zhang Z. A new discrete double-population firefly algorithm for assembly sequence planning. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 2016, 230(12): 2229-2238.

[14] Murali G B, Deepak B, Bahubalendruni M V A R, et al. Optimal assembly sequence planning using hybridized immune-simulated annealing technique. Materials Today: Proceedings, 2017, 4(8): 8313-8322.

[15] Li X, Qin K, Zeng B, et al. Assembly sequence planning based on an improved harmony search algorithm. The International Journal of Advanced Manufacturing Technology, 2016, 84(9): 2367-2380.

[16] Li J Y, Lu C. Assembly sequence planning with fireworks algorithm. International Journal of Modeling and Optimization, 2016, 6(3): 195.

[17] Gao B, Zhang S C, Sun Hao, et al. Assembly sequence planning based on adaptive gravitational search algorithm. The International Journal of Advanced Manufacturing Technology, 2021, 115(11): 3689-3700.

[18] Mishra A, Deb S. Assembly sequence optimization using a flower pollination algorithm-based approach. Journal of Intelligent Manufacturing, 2019, 30(2): 461-482.

[19] Su Y, Mao H, Tang X. Algorithms for solving assembly sequence planning problems. Neural Computing and Applications, 2021, 33 (2): 525-534.

[20] Guo J W, Sun Z Z, Tang H, et al. Improved Cat Swarm Optimization Algorithm for Assembly Sequence Planning. Open automation & control systems journal, 2015, 7(1):792-799.

[21] Li X Y, Qin K, Zeng B, et al. Assembly sequence planning based on an improved harmony search algorithm. International Journal of Advanced Manufacturing Technology, 2016, 84(9-12):2367-2380.

[22] Ghandi S, Masehian e. A breakout local search (BLS) method for solving the assembly sequence planning problem. Engineering Applications of Artificial Intelligence, 2015, 39(mar.):245-266.

[23] Li-Li L, Kun C, Jian-Min G, Jun-Kong L, Zhi-Yong G, Hong-Wei D. Research on optimizing-assembly and optimizing-adjustment technologies of aero-engine fan rotor blades. Advanced Engineering Informatics. 2022; 51. doi:10.1016/j.aei.2021.101506

[24] Gulivindala A K, Bahubalendruni M V A R, Varupala S S V P, et al. A heuristic method with a novel stability concept to perform parallel assembly sequence planning by subassembly detection. Assembly Automation, 2020, 40(5): 779-787.

[25] Jun-Kong L, Li-Li L. Optimization of Assembly Sequence Planning of Turbine Low-pressure Rotor Blades Based on the Improved Simulated Annealing Algorithm. Journal of Physics: Conference Series,2021,1939(1):1-7.