Design Studies and Intelligence Engineering
L.C. Jain et al. (Eds.)

doi:10.3233/FAIA231449

Optimization Analysis of Product Design Process Based on Reinforcement Learning Algorithm

Yuping HU[a] and Qian LI[b]
a. Academy ofart and design, Shaoyang Unigersity, Shaoyang 422099, Hunan, China
b. School ofJewelry andArt Design, Wuzhou Unigersity, Wuzhou 543003, Guangxi, China

Abstract: In order to reduce therework iteration in the product development process, the optimization analysis of product design process based on reinforcement learning (multi-objective process optimization genetic algorithm based on design structure matrix (DSM) theory) is proposed. By optimizing the task execution sequence, therework in the product development process can be reduced to compress the progress and reduce the cost. The optimization algorithm is an improved genetic (GA) algorithm, in which time and cost are considered in the fitness function. In the selection, crossover and mutation operators, the strategy of maintaining optimal solution is adopted. The simulation results show that the optimization algorithm can reduce the development time by 30% ~ 40% and the cost by 7% ~ 20% for product development projects with high task coupling. Conclusion: The optimization algorithm can effectively reduce therework iteration in the project development process, thus shortening the product development time and saving the development cost.

Keywords: process optimization; Genetic algorithm; Simulation; Design structure matrix

1 Introduction

In today's era when innovation is the main theme, R&D plays a decisive role for a country and an enterprise, and successful R&D project management is the basis to ensure the success of the project. The 18th National Congress of the Communist Party of China put forward the strategy of "innovation-driven development", and the core of innovation lies in separating the independent innovation ability of enterprises. However, the management level of Chinese enterprises is backward, especially the management ability of large-scale complex R&D projects is weaker, which seriously affects the independent innovation ability of China enterprises. The survey shows that 70% of China's current R&D projects are beyond the estimated time schedule, with an average of 20% to 50% exceeding the planned delivery time. More than 90% of R&D projects are over the budget, and the higher the complexity of the projects, the more they exceed the project plan. Therefore, it is of great theoretical and practical significance to systematically study the efficient management methods and models of R&D projects that adapt to China’s national conditions, and to enhance the independent innovation ability and management ability of enterprises.

This means that the project manager must improve the design and management ability of complex systems, which is closely related to the tools adopted. In recent years, Design Structure Matrix, DSM) based on process management and Quality Function Deployment, QFD) based on meeting customers' needs have been widely used in the process management of complex product design and R&D [2]. QFD, as a quality management technology and method with the idea of source management, is a method of planning and systematic analysis in advance by identifying and analyzing the "customer voice" and transforming it into engineering and management measures, which enables managers to identify the engineering and manufacturing problems in the early stage of the project life cycle and facilitate the development of preventive strategies. DSM provides a better method for designing, analyzing, and organizing complex systems. It can visually analyze complex R&D projects. The "Multi-domain Matrix" (MDM) is the latest development of DSM, which is used to analyze the dependencies between cross- domain elements.

2 Literature review

Understanding the influencing factors of R&D process is the first premise to formulate optimization scheme and carry out implementation activities, and it is also one of the key contents in the theoretical research of R&D process optimization. Different researchers focus on different aspects of the factors that affect the new product development process. Some scholars have expounded the influence of R&D department on new product R&D from the perspective of organizational setting. Sharma, A. and others found through the survey data of 243 R&D departments that an influential R&D department is more likely to achieve high-level R&D performance, and the R&D department can improve its influence in the organization by improving the degree of innovation and customer relationship [4]. Khan, M.I.H. and others empirically tested the effects of variables related to organizational flexibility of R&D departments on habit and behavior innovation and management efficiency in enterprise R&D process by studying 112 companies [5]. There are also some scholars who think that new product development is a collective activity, and the ability of R&D personnel and the communication between R&D personnel have a positive impact on the performance of R&D activities. Yang, W. and others linked the strategic decision of R&D department from 2009 to 2013 with the financial health of the company, and pointed out that in order to improve the financial performance, managers of enterprises (especially small and medium-sized enterprises) should participate more in research activities, and should pay attention to using more structural and functional R&D methods in the allocation of R&D strategic decisions [6]. Arboretti, R. and others analyze that the education, experience, age, and gender of R&D personnel have great influence on new product development from the demographic dimension [7]. There are also a few scholars who extend the research perspective from the inside of the R&D department to the outside and consider the importance of external factors in the product development process. Panzer, M. et al. put forward that the participation of suppliers is the key coordination process of product design and process design, and integrating material suppliers into the process of new product development can improve product quality, reduce the cost of new products, and promote the smooth launch of new products [8]. According to the product life cycle theory, Diao, Y and others divide the process optimization into six stages, namely, finding the optimization project, finding the optimization goal, diagnosing the process problems, redesigning the process, integrating the process and evaluating the optimization results [9].

In this paper, considering both time and cost, the multi-objective process optimization of product development is studied, and an intelligent algorithm for optimizing product development process based on the priority rule of process execution is designed.

3 Research methods

3.1 Parallelproduct development model

3.1.1 Basic assumptions andproblem description

The premise assumptions of the concurrent product development model are as follows:

1) The product development project can be decomposed into multiple tasks, and the completion time and cost of each task obey certain statistical laws; The relationship between tasks can be expressed by Boolean information flow matrix.

2) After the completion of the project, information will be output to other tasks, which will lead to the start or rework of other tasks. Therework probability and rework impact can be estimated in advance, which are expressed by rework probability matrix and rework impact matrix respectively, and do not change with time [10,11].

3) Each task has a maximum number of rework times. When the number of rework times of a task exceeds the specified value, even if other tasks provide it with information, the task will not be reworked (this assumption is to avoid the situation that the project cannot be completed due to endless rework).

4) There are sufficient resources in the project execution process, and multiple projects can be executed in parallel, so there is no resource bottleneck.

In the process of product development, there may be information circulation between tasks. If all the upstream processes for which a task has information input have been completed, the task can be executed [12]. When a task is completed, it will output information to the downstream task, so that the downstream task can start work. At the sametime, it may also provide feedback information to the upstream task, so that the upstream task can correct its work with a certain probability, that is, rework; In addition, when the results of downstream tasks fail to meet the specified requirements, it will also cause rework of upstream processes. The rework of one task may lead to therework of other tasks, thus making the whole project form a complex network structure.

3.1.2 Modeling ofcomplexproduct developmentprocess

Due to the problem of rework in the process of project development, the relationship between the overall project duration and cost and the duration and cost of each task is difficult to be obtained by analytical method, and only approximate solutions can be obtained by simulation. The simulation model is a discrete event simulation model, and the trigger event is that one or more tasks are completed at a certain point; The status includes three aspects: the unfinished quantity of each task, the executable status of each task and the rework times of each task. Given the task execution priority sequence, the product development simulation process is as follows:

Step1: Traverse all tasks. If a task does not depend on other upstream tasks, or if all the upstream tasks on which it depends have been completed, and the task has not yet been completed, it is an executable task, which means that the task can be started at the current time [13,14]. Find all executable tasks, and set the shortest completed task time among the tasks as the current simulation time slice according to the execution time and unfinished quantity of these tasks. dt, and add it to the accumulated time tt. dt. The calculation formula of is the following formula (1):

$$dt = \min_{E_{it} \neq 0} \{V_{it} \cdot T_i\}. \tag{1}$$

Step2: For each executable taski, atdt. The amount of work completed in time isdt / T_i , the execution cost is$C_i \cdot dt / T_i$. Among them: $T_{i.}$ For the taski The execution time, C_i For the taski The execution cost of. Accumulate the cost into the total cost TC, ift Time taski The original remaining workload is set tok_{jt} %, then after time. dt After that, the remaining workload becomes $k_{jt}\% - dt / T_i$. For a certain item in dt Tasks completed in timei If the task outputs information to the task.j , andj At this time has been partially completed, theni The completion of may lead to a partially completed task.j Rework [15]. When the taskj When therework times of are less than the maximum rework times ($nj < M_{rw}$) to generate random numbers.r. if$r \leq R_{ij}$ (R For rework probability matrix), thenj Need to rework. j Add 1 to the rework times, and according to the rework impact matrixP Re - calculationj The unfinished workload is as follows (2):

$$k_{jt}\% + (1 - k_{jt}\%) \times P_{ij}. \tag{2}$$

if$r > R_{ij}$, do notrework. Set the executable status of tasks with a remaining workload of 0 to 0.

Step3: Find and execute all executable tasks again until all tasks are completed.

Because the rework process is random, the project cost and project duration obtained by each simulation are a random value. If the following equation (3) is satisfied after such simulation is performed for many times:

$$\begin{cases} \dfrac{\frac{1}{L}\sum_{l=1}^{L} TT_l - \frac{1}{L+1}\sum_{l=1}^{L+1} TT_l \mid}{\frac{1}{L}\sum_{l=1}^{L} TT_l} < 0.005, \\ \dfrac{\frac{1}{L}\sum_{l=1}^{L} TCl - \frac{1}{L+1}\sum_{l=1}^{L+1} TC_l \mid}{\frac{1}{L}\sum_{l=1}^{L} TC_l} < 0.005, \end{cases} \tag{3}$$

It shows that the simulation results tend to be stable, and the simulation program is stopped. Among them: TT_l andTC_l Respectively the firstl The project duration and project cost obtained from the second simulation, L Is the total number of simulations.

According to the task parameters and task execution sequence of the project, the distribution of completion time and cost of the project can be obtained through multiple simulations, so as to carryout risk analysis and determine the probability that the development project can be completed under the specified time and cost requirements. The formula for calculating the probability that a product development project can be completed as required is the following formula (4):

$$P(T_E, C_E) = 1 - F(T < T_E, C < C_E). \tag{4}$$

3.2 product development process optimization algorithm

According to the correspondence between task execution priority sequence and project time cost, this paper designs an improved genetic algorithm to find the approximate optimal task execution sequence [16]. In the design of fitness function, both time and cost are considered, and in the design of genetic operator, a variety of optimal solution keeping strategies are used, so that the algorithm can converge quickly, but it can breakthrough the local optimal solution. The goal of this algorithm is to reduce unnecessary rework iterations in product development through reasonable project development process design, thus reducing development costs and shortening development time.

The necessity of genetic algorithm struggle for existence often exists among natural creatures, and only those with strong adaptability can survive in the fierce struggle for existence and pass on their excellent genes from generation to generation. The evolution process of natural organisms is consistent, from low to high, simple to complex [17]. Moreover, it is in the genetic process of this generation that the population gradually evolves into individuals who are more adaptable to environmental variability, so that the biological population can continue to develop and improve. With this kind of natural evolution law of survival of the fittest and survival of the fittest, today's genetic algorithm (GA) has been abstractly developed. The optimization algorithm of design structure matrix belongs to sorting combination optimization, so the order of design activities essentially determines the optimal solution of objective function. For sorting combinations, the difference between the sub-optimal solution and the optimal solution is often very small, and the sub-optimal solution retains most of the excellent genes of the optimal solution (that is, the design activity order). For product development activities, the process of finding the optimal product component development sequence is very similar to the process of natural organisms evolving from weak to strong, so genetic algorithm is chosen as the optimization algorithm in this paper.

DSM scheduling strategy is embedded into DSM simulation program, and DSM simulation outputs the value of fitness function corresponding to each chromosome on the premise of satisfying the constraints of optimization problem, thus overcoming the problem that there are many uncertain factors in optimization problem, and it is difficult to provide genetic algorithm with fitness function expressed in analytical form. For each chromosome obtained, DSM simulation is called to obtain the average project duration and cost corresponding to the chromosome, and then the subsequent genetic operation is returned and executed [18]. In each iteration of the algorithm, DSM simulation is used to obtain the fitness function, and the chromosomes are sorted non-dominantly. Through many iterations, the population keeps approaching the Pareto optimal set. The overall framework of the algorithm is shown in Figure 1.

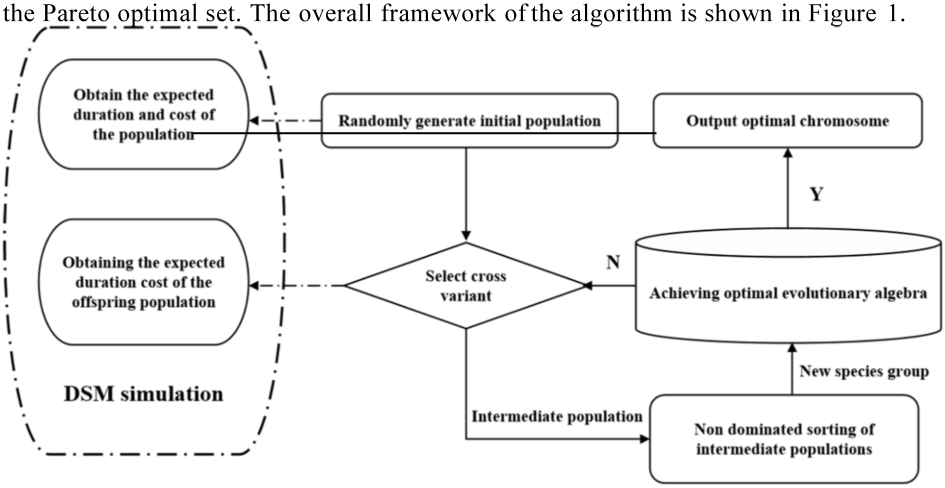

Fig. 1 Overall framework of hybrid algorithm based on simulation and genetic algorithm

3.2. 1 Chromosome coding scheme

This paper adopts task coding method. Individual Id. It's a task sequence Id $= \{i^1, i^2, \ldots, i^n\}$ A that represents the priority of task execution, and each Id Corresponding to a schedule planning scheme. The decoding process is based on the task priority sequenceId. And determine the execution order of each task, the total project completion time and completion cost.

3.2. 2 Fitnessfunction

The shorter the project time and the lower the cost, the greater the individual's adaptability. The importance weight of time and cost can be determined by expert scoring or investigation and interview [19]. Because of the randomness of the simulation process, it is impossible to get the maximum and minimum values of the project time and cost, and it is impossible to standardize the project time and cost. In order to eliminate the influence of dimension, taking the average execution time and cost of the project in the initial situation as the standard, the fitness function can be defined as the following formula (5):

$$\mathrm{Ft(Seq)} = \frac{1}{w_T \frac{\frac{1}{L}\sum_{j=1}^{L} \mathrm{TT_j(Seq)}}{\mathrm{TT_j(Seq0)}} + wc \frac{\frac{1}{L}\sum_{j=1}^{L} \mathrm{TC_j(Seq)}}{\mathrm{TC_j(Seq0)}}} \cdot \tag{5}$$

Among them: Seq Execute a priority sequence for the task; $\mathrm{TT_j}$ (Seq) and$\mathrm{TC_j}$ (Seq) According to the priority sequenceSeq In the firstj The total time and cost of the project obtained in the second simulation; Seq0 Is an initial task priority sequence; LIs the number of simulations; w_T andwc They are the importance weights of time and cost respectively, including $w_T + wc = 1$。

3.2.3 Genetic Operator and Termination Strategy

Assume that the number of individuals in the initial population is M, it is randomly generated. MA 1 ~ n Randomly arranged integer sequences, each sequence is a task execution sequence, corresponding to an individual. In order to realize the optimization of the whole population, individuals with strong vitality are selected from the population to produce a new population. In this paper, the proportional selection operator (roulette selection operator) is adopted, that is, the probability that an individual is selected and passed on to the next generation population is directly proportional to the fitness of the individual.

The crossover algorithm adopts single-point crossover. On the issue of intersection selection, this paper has made some improvements to the traditional intersection method. For two parents i and j, its fitness is respectively Ft (i) andFt(j), the calculation formula of the intersection is the following formula (6):

$$X = \mathrm{Floor}\left(\frac{\mathrm{Ft(i)}}{\mathrm{Ft(i)} + \mathrm{Ft(j)}} * N\right) + 1. \tag{6}$$

Among them: N Is the chromosome length, that is, the number of project tasks; Floor () is an integer function. DaughterId′ Chromosome part 1 (frontX Gene values) were taken from the parent in sequence.i In front ofX Genes, part 2N − X Three genes were taken from the father.j. Slave parentj The first gene of begins to be selected sequentially. If a gene value is duplicated with the gene in the first part, the parent is selected instead of the duplicated gene value.j The next gene value of until the parent is selected.j DediX Genes. The cross- generated offspring has the parent.i The gene ratio of is listed as the following formula (7):

$$\frac{\mathrm{Ft(i)}}{\mathrm{Ft(i)} + \mathrm{Ft(j)}}, \tag{7}$$

Have a father j. The gene proportion of is the following formula (8):

$$\frac{\mathrm{Ft(j)}}{\mathrm{Ft(i)} + \mathrm{Ft(j)}} \tag{8}$$

Through this method, the parent with higher fitness will have more genes to pass on to the next generation, so as to better realize the survival of the fittest. The newly

generated individuals are used to replace the individuals with low fitness in the population.
The mutation algorithm in this paper is: if the chromosome length isN, randomly generate two 1 ~ N Integer betweeni andj , the individuali Bit sumj The gene values at positions are mutually reversed. Randomly select a certain proportion of individuals in the population with probabilityP_m Perform mutation operation to replace individuals with low fitness in the population with newly generated individuals.
When the average fitness change of the upper and lower generations is very small, it can be considered that the algorithm has converged and stopped. The judgment formula is as follows (9):

$$\left|\frac{\sum_{i=1}^{M} Ft(Id_{ki}) - \sum_{i=1}^{M} Ft(Id_{(k+1)i})}{\sum_{i=1}^{M} Ft(Id_{ki})}\right| < 0.005. \tag{9}$$

Among them: k Represents genetic algebra, Id_{ki} Express the first k Daididii An individual, M Represents the population size. Find out the individual with the greatest fitness from the last generation population, decode the gene of the individual and get the optimal solution. Population size P_s , crossover probability P_c , mutation probability P_m , reproductive algebra G_n Etc. can be estimated according to experience, or determined by many experimental comparisons.

4 Result analysis

A simple example is given to illustrate the application of this model in a chip research and development project, which can be roughly divided intoA ~ I These nine development tasks. The completion time of these tasks obeys the normal distribution, and the expected execution time is {30, 40, 35, 39, 52, 55, 62, 29,39} respectively, and the variance is {3.4, 4.2, 3.9, 3.8, 4.6, 4.8, 3.9. The cost per unit time of each task is {2, 3, 2,1, 4, 3, 2, 4, 3} respectively, and the maximum rework times of each task is 5. The relationship between these tasks is shown in Figure 2.

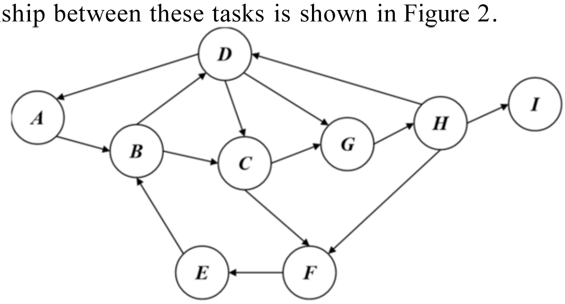

Figure 2 Chip Design Information Flow Diagram

Therework probability matrix and rework impact matrix of this project is shown in Table 1 respectively. If there is norework, the sum of the completion costs of all tasks is 1 019.

Table 1 Project Task Duration and DSM Influence Matrix

task	A	B	C	D	E	F	G	H	I
A	-	0.8(0.9)	-	-	-	-	-	-	-
B	-	-	0.6(0.7)	0.7(0.6)	-	-	-	-	-
C	-	-	-	-	-	0.7(0.6)	0.9(0.8)	-	-
D	0.6(0.8)	-	0.7(0.8)	-	-	-	0.7(0.7)	-	-

E	-	0.8(0.9)	-	-	-	-	-	-	-
F	-	-	-	-	0.8(0.7)	-	-	-	-
G	-	-	-	-	-	-	-	0.8(0.8)	-
H	-	-	-	0.7(0.8)	-	0.9(0.7)	-	-	0.9(0.8)
I	-	-	-	-	-	-	-	-	-

Note: The number before parentheses is the rework probability. R_{ij}, the number in brackets is rework impact rate. P_{ij} 。

The Monte Carlo simulation program is compiled on the platform of Mat lab 7. 0, and the distribution of the completion time and cost of the development project is calculated. The simulation results show that the average value of the simulation results tends to be stable when the number of simulations exceeds 200. In fact, when the simulation process reaches 172 times, the simulation stop condition is reached [20]. When the tasks in this project are executed in the priority order of {A, B, C, D, E, F, G, H, I}, the average completion time of the project is about 588. 0, and the variance is 117. 7. The completion cost is 3 487. 7 and the variance is 710. 4. Set the number of simulations to 300.

Due to the correlation between tasks, the project has been reworked many times during the execution, which greatly increases the construction period and cost. If the project completion time is required to be within 500 and the cost is within 3500, the probability that the project can be completed as required is only 19. 3%.

An improved genetic algorithm optimization program is written on the platform of Mat lab 7. 0 to find the approximate optimal project development process. The parameter is set to population number. M= 50, the number of individuals in each iteration is 20, the crossover probability is 0. 8, and the mutation probability is 0. 1. In order to facilitate observation, the fitness function value is expanded by 10 times, so that it is between 1 and 10. Due to different market conditions, the importance of time and cost to the success of the project will be different. This case only investigates the task execution optimization under the following two special circumstances.

1) There is no requirement for development time, and only the cost is considered, namely $w_C = 1$, $w_T = 0$. At this time, the change of population average fitness is shown in Figure 3. When the genetic algorithm program runs to about 40 generations, the average fitness function of the population tends to be stable. The approximate optimal project execution priority order is {G, C, D, H, B, E, F,A, I}; The average project execution time is 414. 0, and the variance is 100. 7. The average execution cost is 2 808. 0, and the variance is 543. 6. Compared with the original scheme, the time is saved by 29. 6% and the cost is saved by 19. 4%. Under this arrangement of project execution sequence, the probability of project completion time within 500 and completion cost within 3 500 can reach 75. 7%.

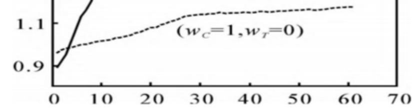

Fig. 3 Variation curve of population average fitness

2) only consider the time, regardless of the development costs, namely $w_C = 0$, $w_T = 1$. At this time, the simulation optimization results are shown in Figure 5. When the genetic algorithm program runs to about 20 generations, the average fitness of the

population tends to be stable. The approximate optimal project execution priority order is {G, C, D, B, E, F, H,A, I}. The average project execution time is 370. 1, and the variance is 70. 2. The average execution cost is 3 249. 8, and the variance is 474. 1. Compared with the original scheme, the time is saved by 37. 1% and the cost is saved by 6. 8%. Under this project execution sequence arrangement, the probability that the project completion time is within 500 and the completion cost is within 3 500 can reach 73. 3%.

By comparing the above simulation results, it can be found that the project execution time is also optimized while optimizing the cost; While optimizing the project, the cost situation has also been improved. Under different weights, the focus of optimization is different. In any case, the probability that the project can be successfully completed under the dual requirements of time and cost is greatly improved. In order to verify the stability of the above optimization results, the optimization program is run again, and the optimization sequences obtained are {g, c, d, h, b, I, e, f, a}, {g, c, d, b, e, f, h, i, a} respectively. Under the above two execution priority sequences, the completion time of software development projects is 418. 7 and 378. 6, and the completion cost is 2 888. 5 and 3 187. 5, respectively, and the data changes are not more than 5%.

The simulation results show that different optimization processes can be obtained according to the preference of managers for schedule and cost. For highly coupled product development projects, this algorithm can effectively reduce the rework iteration in the product development process. In this case, the compression range of development time is between 30% and 40%; The cost reduction range is between 7% and 20%, and the effect is remarkable.

5 Conclusion

Aiming at the problems of task coupling and rework iteration in parallel product development, this paper proposes a multi-objective process optimization algorithm based on genetic algorithm. The algorithm can get different priority sequences of project execution by giving different weights to the project duration and cost. The example analysis shows that the optimization algorithm can effectively reduce the rework iteration in the project development process, thus shortening the product development time and saving the development cost.

Because this paper does not consider the resource constraints in the process of project development, it is not applicable to development projects with strict resource constraints. In addition, this model assumes that therework probability and rework impact are known, which can be estimated by expert experience or historical data. This assumption is usually not valid for small companies lacking historical data and development experience, which is also a limitation of this paper. Future research can further explore the impact of the changes of rework probability matrix and rework impact matrix on product development time and cost.

References

[1] Li, C., Zheng, P., Yin, Y., Wang, B., & Wang, L. (2023). Deep reinforcement learning in smart manufacturing: A review and prospectsCIRP Journal of Manufacturing Science and Technology, 40, 75- 101.

[2] Dabbagh, S. R., Ozcan, O., & Tasoglu, S. (2022). Machine learning-enabled optimization of extrusion-based 3D printing. Methods, 206, 27-40.

[3] Vinitha, N., Vasudevan, J., & Gopinath, K. P. (2023). Bioethanol production optimization through machine learning algorithm approach: biomass characteristics, saccharification, and fermentation conditions for enzymatic hydrolysis. Biomass Conversion and Biorefinery, 13(8), 7287-7299.

[4] Sharma, A., Mukhopadhyay, T., Rangappa, S. M., Siengchin, S., & Kushvaha, V. (2022). Advances in computational intelligence of polymer composite materials: machine learning assisted modeling, analysis and design. Archives of Computational Methods in Engineering, 29(5), 3341-3385.
[5] Khan, M. I. H., Sablani, S. S., Nayak, R., & Gu, Y. (2022). Machine learning-based modeling in food processing applications: State of the art. Comprehensive reviews in food science and food safety, 21(2), 1409- 1438.
[6] Yang, W., Xiang, W., Yang, Y., & Cheng, P. (2022). Optimizing federated learning with deep reinforcement learning for digital twin empowered industrial IoT. IEEE Transactions on Industrial Informatics, 19(2), 1884- 1893.
[7] Arboretti, R., Ceccato, R., Pegoraro, L., & Salmaso, L. (2022). Design of Experiments and machine learning for product innovation: A systematic literature review. Quality and Reliability Engineering International, 38(2), 1131- 1156.
[8] Panzer, M., & Bender, B. (2022). Deep reinforcement learning in production systems: a systematic literature review. International Journal of Production Research, 60(13), 4316-4341.
[9] Diao, Y., Yan, L., & Gao, K. (2022). A strategy assisted machine learning to process multi-objective optimization for improving mechanical properties of carbon steels. Journal of Materials Science & Technology, 109, 86-93.
[10] Haq, Z. U., Ullah, H., Khan, M. N. A., Naqvi, S. R., & Ahsan, M. (2022). Hydrogen production optimization from sewage sludge supercritical gasification process using machine learning methods integrated with genetic algorithm. Chemical Engineering Research and Design, 184, 614-626.
[11] Lee, S., Zhang, Z., & Gu, G. X. (2022). Generative machine learning algorithm for lattice structures with superior mechanical properties. Materials Horizons, 9(3), 952-960.
[12]Mahajan, S., Abualigah, L., Pandit, A. K., & Altalhi, M. (2022). Hybrid Aquila optimizer with arithmetic optimization algorithm for global optimization tasks. Soft Computing, 26(10), 4863-4881.
[13] Sohail, A. (2023). Genetic algorithms in the fields of artificial intelligence and data sciences. Annals ofData Science, 10(4), 1007- 1018.
[14] Deng, W., Zhang, X., Zhou, Y., Liu, Y., Zhou, X., Chen, H., & Zhao, H. (2022). An enhanced fast non-dominated solution sorting genetic algorithm for multi-objective problems. Information Sciences, 585, 441-453.
[15] Muneer, S. M., Alvi, M. B., & Rasool, M. A. (2022). Genetic algorithm based intelligent system for estate value estimation. International Journal of Computational and Innovative Sciences, 1(1), 28-38.
[16] Ji, Y., Liu, S., Zhou, M., Zhao, Z., Guo, X., & Qi, L. (2022). A machine learning and genetic algorithm-based method for predicting width deviation of hot-rolled strip in steel production systems. Information Sciences, 589, 360-375.
[17] Xia, X., Qiu, H., Xu, X., & Zhang, Y. (2022). Multi-objective workflow scheduling based on genetic algorithm in cloud environment. Information Sciences, 606, 38-59.
[18]Abdollahi, J., & Nouri-Moghaddam,B. (2022). Hybrid stacked ensemble combined with genetic algorithms for diabetes prediction. Iran Journal of Computer Science, 5(3), 205-220.
[19] Zhou, J., Huang, S., Zhou, T., Armaghani, D. J., & Qiu, Y. (2022). Employing a genetic algorithm and grey wolf optimizer for optimizing RF models to evaluate soil liquefaction potential. Artificial Intelligence Review, 55(7), 5673-5705.
[20] Shehadeh, A., Alshboul, O., Tatari, O., Alzubaidi, M. A., & Salama, A. H. E. S. (2022). Selection of heavy machinery for earthwork activities: A multi-objective optimization approach using a genetic algorithm. Alexandria Engineering Journal, 61(10), 7555-7569.