Journal of Computers Vol. 32 No. 6, 2021, pp. 98-106
doi:10.53106/199115992021123206008

Applications of an Improved PSO in Integer Linear Programming

Ying Wang*

Chongqing Chemical Industry Vocational College, Chongqing 400020, China
cwc20170302@cqcivc.edu.cn

Received 27 March 2021; Revised 2 July 2021; Accepted 2 August 2021

Abstract. In order to address the problems with production planning for Model Q cars and Model X new energy cars of an automotive group, a mathematical model is proposed, which is intended for a constrained integer linear programming problem. An AI algorithm, i.e. PSO, is introduced to solve this problem, while such methods as penalty function are used to solve the constraint conditions and integer programming problem. Furthermore, in order to overcome the challenge that the standard PSO is very likely to meet a local extremum, an improved PSO is proposed, which, as shown in terms of search computing results, is more accurate and efficient than the standard PSO, thus aptly solving the integer linear programming problem.

Keywords: improved PSO, integer programming, penalty function, stochastic solution

1 Introduction

In the operation and management of production enterprises, there are numerous problems besetting the senior management, e.g. the problems with rent/sale of plants, funding/financing and fund allocation, and venture investment involved in operation plans, as well as the problems with economic life of production equipment, personnel assignment, optimization of logistic/transportation of production materials and finished products, and rational stock cutting involved in production plans. However, there is no quick or easy solution to these problems, and the executives' personal experience always falls short in the ever-changing business environment. As a result, scientific management methods and appropriate management tools become a must in solving these problems. A Harvard series of publications on management [1] define "decision making" as an intellectual activity that devises strategies or methodologies to deal with present or future problems. Managerial decision making is done throughout production and operation or even across our life. Therefore, it is particularly important to boost the efficiency and effectiveness of decision making by implementing scientific theories and methods rationally and effectively.

Many scholars have explored the integer linear programming problem in the field of engineering. For example, Lee et al. [2] have adopted random search, instead of gradient search, for the purpose of engineering problem optimization. Chaabane et al. [3] have proposed a sustainable supply chain design framework based on mixed-integer linear programming, so as to evaluate the tradeoff between the objectives and the environmental ones in the aluminum industry under various constraints. Parisio et al. [4] have translated the microgrid model predictive control problem into an integer linear programming problem to investigate optimization. By proposing the mixed-integer linear programming, Chen et al. [5] maximizes the critical loads, while satisfying the constraints; and their method is applicable to the demands for autonomous communication after disaster events. Amin et al. [6-7] proposed a mixed-integer linear programming model, which is aimed at minimizing the total cost and the integrated model that consists of two stages. Alumur et al. [8] improved the mixed-integer linear programming formula to handily combine most reverse network structures. Pandzic et al. [9] translated the quotation problem into a two-stage stochastic mixed-integer linear programming model, so as to maximize the expected profit of

* Corresponding Author

power plants. Kriett et al. [10] adopted a general mixed-integer linear programming model to minimize the operating costs of residential power grids. Fu et al. [11] investigated an improved mixed integer programming method, finding it can significantly enhance the efficiency.

To sum up, many scholars have studied the problem of integer linear programming, and have achieved a series of achievements, but the research on the problem of integer linear programming by artificial intelligence algorithm is still rare. In order to solve the integer linear programming problem with production planning for Model Q cars and Model X new energy cars of an automotive group, this study establishes a constrained mathematical model, uses the improved PSO to translate the production planning problem into a space search problem, and compares the search computing results of the improved PSO with those of the standard PSO, finding that the former is better, which solves the disadvantage that the traditional particle swarm optimization algorithm is easy to fall into local extremum. This study has high reference value for addressing relevant integer programming problems.

2 Case Description

The traditional Model Q cars produced by the automotive group are among its highly marketable car models. However, the sales of Model Q cars have been tremendously impacted by the emergence of new energy vehicles and the improvement in battery endurance. In 2018, in cooperation with Contemporary Amperex Technology Co., Ltd., this group decided to launch Model X new energy cars. Then in 2020, the new production line officially went live for production. The required total amount of resources and unit consumption as known are listed in Table 1. A survey indicates that the unit price of the Model Q cars of this enterprise was RMB 170,000, 160,000, 150,000, 140,000 and 130,000 yuan in 2016-2020, respectively. On the other hand, the Model X new energy cars are expected to gain a governmental subsidy of RMB 50,000 yuan/unit. It is now necessary to devise a production plan for this automotive group for 2022, with the known total inventory and consumption of a key component listed in Table 1.

Table 1. Total inventory and consumption of a key component

Model Resource	Machine-hour (hour)	Parts (RMB ten hundred)	Labor cost (RMB ten hundred)
Traditional car	2	6	2
New energy car	3	8	1
Total	14000	36000	9000

3 Establishment of the Mathematical Model

An analysis based on the managerial decision-making theory is given below:

(1) Clear objective: Under the constraints of the enterprise, how many units of traditional Model Q and new energy Model X should the group manufacture, in order to maximize its total revenue, while complying with the national policies and fulfilling its enterprise values?

(2) Influencing factors: The direct factors affecting the revenue are unit prices and quantities.

(3) The unit prices and quantities are given below:

Unit price: the unit prices of Models Q and X in 2022 are unknown. However, they are set as PQ = RMB 120,000 and PX = RMB 140,000 (including the government subsidy) on the basis of the long-term strategic forecast of the enterprise.

Quantity: the quantities of Models Q and X in 2022 need to be planned, and they are also known as decision variables. It is assumed the production output of Model Q is Q_1 and that of Model X is X_1 in 2022.

(4) Quantification of the objective -- Objective function: If the total revenue generated from Models Q and X in 2022 is Z, the objective function is $Max(Z) = PQ *Q_1 + PX*X_1$.

(5) The indirect factors affecting the total revenue mainly include labor, parts, production lines and other limited resources. In the same fashion as quantification of the objective, Q_1 represents the unit consumption of the resources by Model Q products, and X_1 the unit consumption by Model X products. The total quantity of every resource is limited, as shown in Formula (1).

$$\begin{cases} 2Q_1 + 3X_1 \le 14000 \\ 6Q_1 + 8X_1 \le 36000 \\ 2Q_1 + X_1 \le 9000 \end{cases} \tag{1}$$

4 Improved PSO Algorithm

4.1 Introduction to PSO

This study adopts the PSO, an AI algorithm used for the purpose of search computing.

As a new evolutionary algorithm developed in recent years, Particle Swarm Optimization (PSO) starts from a stochastic solution to seek the optimal solution through iterative computation. The quality of the solution can be assessed based on the fitness value, and its rules are simple in comparison with the genetic algorithm, because, instead of using the "crossing" and "variation" steps as in the genetic algorithm, it tracks the found optimal values to seek the globally optimal solution. Thanks to its easy realization, high accuracy, fast convergence, and superiority in solving practical problems, this algorithm has drawn extensive attention [12].

PSO originated from an investigation into the feeding behavior of birds. Imagine such a scenario: a flock of birds are searching food in an area, where only one piece of food is available, but none of the birds knows the exact site; however, they know the distances between the food and themselves. Then, what is the optimal approach to finding the food? The simplest and most efficient way is to search the area around the bird closest to the food.

In PSO, the bird in the search space, also known as "particle", represents the solution to each optimization problem. Every particle, with a fitness determined by the optimized function, also features a speed that decides its flying direction and distance, while detecting the search space by following the currently optimal particle. PSO is initialized as a swarm of evenly distributed particles (stochastic solution). Then, the optimal solution is gradually identified through iteration. In each subsequent iterative computation, any particle renews its position and speed by following two extremums: One is the optimal solution identified by the particle itself so far; and the other is the optimal solution found by the particle swarm so far.

In this study, main steps of search computing under PSO are as follows:

Step 1: select the parameters subject to search computing. This study uses two parameters: Q_1 and X_1;

Step 2: based on the experience or relevant data, determine the value ranges of the selected parameters; and generate m sets of data in the value ranges randomly and evenly. In this study, m means the number of particles, which is set as 80); and this number can be increased according to the complexity of the search computing. These 80 particles form a particle swarm, each particle of which represents a set of values;

Step 3: compute the fitness of each particle based on the fitness function;

Step 4: find the historical optimal value of every particle, i.e. the globally optimal value;

Step 5: determine whether the end condition has been satisfied. If so, no further computing is needed, and the global historical optimal value is the output. If not, renews the particle speeds and positions to perform the next iteration; and repeat Steps 3 and 4. The end condition is usually reached when the fitness is less than a specified value or the number of iterations exceeds the upper limit.

4.2 Establishment of the Fitness Function and Constraint Condition

The fitness function is established as follows:

$$f = 12Q_1(i) + 14X_1(i) \tag{2}$$

Where, f is fitness function; and i is the number of particles, which is set as 80 in this study.

To meet the constraint condition of Formula (1), the penalty function is used. So, the fitness function is modified into:

$$f = \begin{cases} 12Q_1(i) + 14X_1(i) & 2Q_1(i) + 3X_1(i) \le 14000 \text{ and} \\ & 6Q_1(i) + 8X_1(i) \le 36000 \text{ and } 2Q_1(i) + X_1(i) \le 9000 \\ 12Q_1(i) + 14X_1(i) - \lambda & 2Q_1(i) + 3X_1(i) \le 14000 \text{ or} \\ & 6Q_1(i) + 8X_1(i) \le 36000 \text{ or } 2Q_1(i) + X_1(i) \le 9000 \end{cases} \quad \textbf{(3)}$$

Where, λ is the penalty function, which is set as 100,000 (RMB 10,000 yuan).

4.3 Realization of Integer Programming

In every round of iteration in PSO, the particle renews its speed and position through the individual extremum and the group extremum:

$$\begin{aligned} V_{id}^{k+1} &= wV_{id}^{k} + c_1 r_1 \left(P_{id}^{k} - X_{id}^{k}\right) + c_2 r_2 \left(P_{gd}^{k} - X_{id}^{k}\right) \\ X_{id}^{k+1} &= X_{id}^{k} + V_{id}^{k+1} \end{aligned} \quad \textbf{(4)}$$

Where, w is the inertia weight, which is set as 0.6 in this study; k is the current number of iterations; V_{id} is the speed of the particle designated as id; X_{id} is the position of the particle designated as id; P_{id} is the individual extremum position of the particle designated as id; P_{gd} is the group extremum position; c1 and c2 are acceleration factors, both set as 1.49445; and r_1 and r_2 are random numbers distributed in the [0, 1] interval.

When solving the integer programming problem by using PSO, as long as X_{id}^{k} and V_{id}^{k} are integers, X_{id}^{k+1} must be an integer, according to Formula (4). The round function in the Matlab software is used to round off X_{id}^{k} and V_{id}^{k} to make them integers, purposed to solve the integer programming problem.

4.4 Improved PSO

In search computing for multiple-peak-point problems, the standard PSO is very likely to encounter a local extremum [13]. Therefore, this study improves Formula (4), i.e., the particle position renewal formula:

$$\begin{aligned} X_{id}^{k+1} &= r_3 && 20\% \textit{ particles update position randomly} \\ X_{id}^{k+1} &= X_{id}^{k} + V_{id}^{k+1} && \textit{Other particles} \end{aligned} \quad \textbf{(5)}$$

Where, r3 is the random number of the search space, which is [0:4000, 0:4000] in this study.

5 Computing Results and Discussion

5.1 Computing Results of the Improved PSO

By using the improved PSO in search computing, this study acquires the computing results as follows: Q1 = 3600, X1 = 1800, and the maximum total revenue is Max(Z) = RMB 684,000,000, as shown in Fig. 1. It can be seen that the optimal solution is found after less than 600 computing steps.

5.2 Discussion

In order to verify the effectiveness of the improved PSO, this study adopts the standard PSO for the computing in the above case, with the search computing results shown in Fig. 2. It can be seen that the standard PSO meets a local extremum in less than 400 computing steps: Q1 = 2904, X1 = 2322, and the maximum total revenue is now Max(Z) = RMB 673,560,000 yuan.

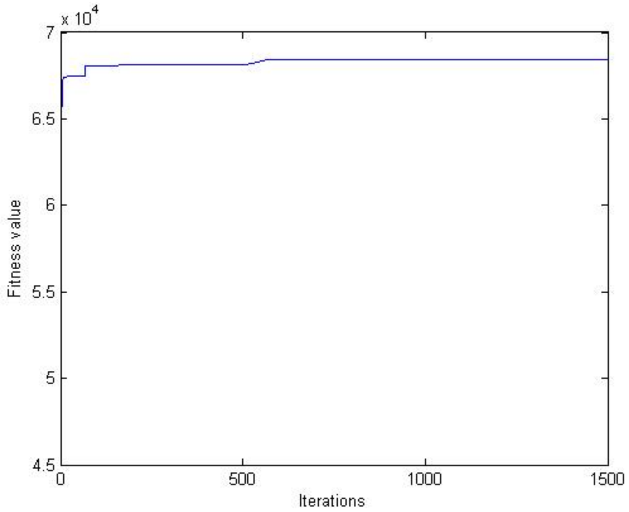


Fig. 1. Variation of the fitness value with the number of iterative steps in the improved PSO

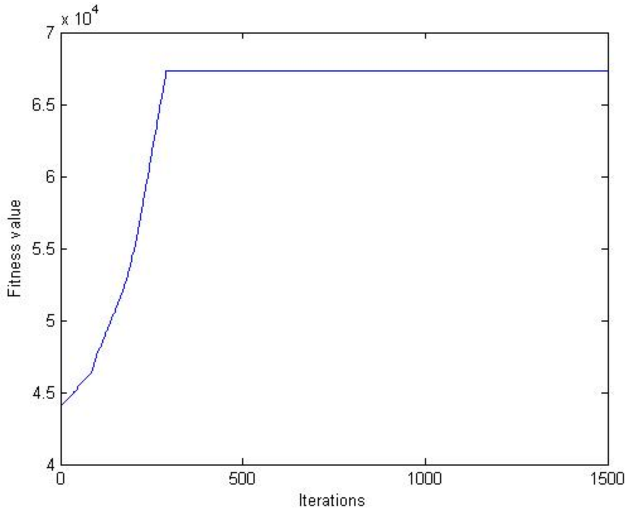
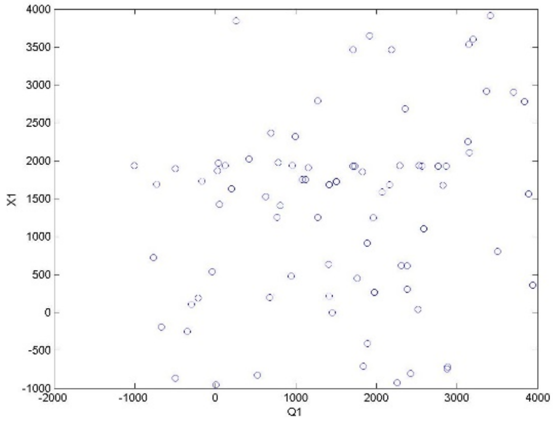
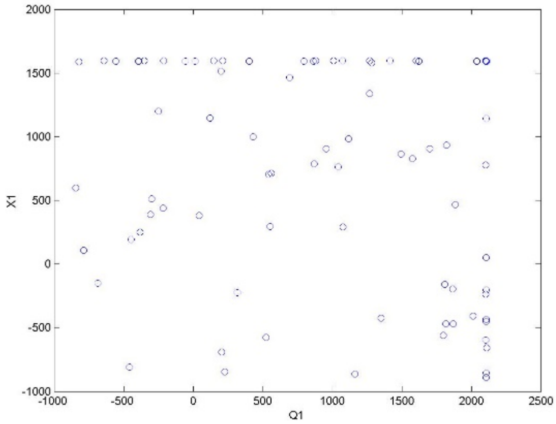


Fig. 2. Variation of the fitness value with the number of iterative steps in the standard PSO

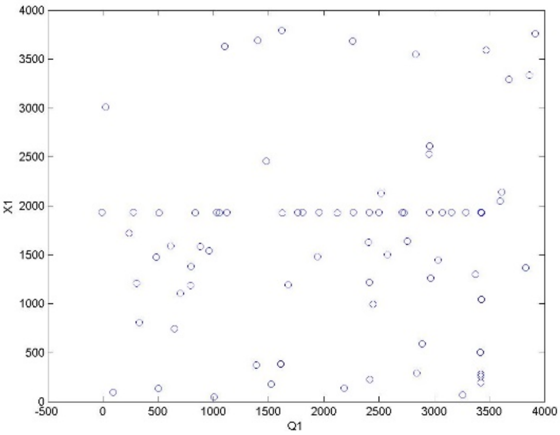
Fig. 3 shows the particle position distribution in every 100 computing steps both in the improved PSO and the standard PSO. It can be seen that the particle positions are in a random distribution at the 100th step both in the improved PSO and the standard PSO. At the 200th step, the standard PSO hits two walls in the vertical and horizontal directions (near $Q_1 = 2400$, $X_1 = 1800$) due to constraint condition; and the improved PSO also hits two walls in the vertical and horizontal directions (near $Q_1 = 3400$, $X_1 = 1900$) in the particle position chart, but the randomly renewed positions of some particles exist outside the walls. At the 300th step, the standard PSO continues the search, and the particle position chart again shows two walls in the vertical and horizontal directions (near $Q_1 = 2800$, $X_1 = 2300$). The wall positions in the improved PSO have no change, while the particles outside the walls are changed to certain extent due to randomly renewed positions of some particles. At the 400th step, the standard PSO once again hits two walls in the vertical and horizontal directions (near $Q_1 = 2900$, $X_1 = 2300$), and many particles are concentrated in the area near $Q_1 = 2900$, $X_1 = 2300$; but the improved PSO hits one horizontal wall in the position chart, with some particles concentrated in the area near $Q_1 = 3400$, $X_1 = 1900$. At the 500th step, the standard PSO finds the local extremum, i.e., $Q_1 = 2904$, $X_1 = 2322$, and lingers near it; in the improved PSO, the particles continue to become concentrated near $Q_1 = 3400$, $X_1 = 1900$, with the horizontal wall gradually disappearing. At the 600th step, the standard PSO continues to linger near the local extremum $Q_1 = 2904$, $X_1 = 2322$; but the improved PSO finds the optimal solution, i.e., $Q_1 = 3600$, $X_1 = 1800$, with most particles concentrated in this this area, and some particles continue the search computing at other positions because of their randomly renewed positions.



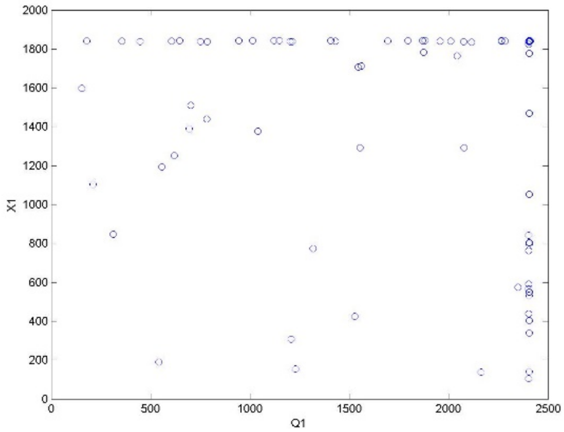
(a) Particle positions at the 100th computing step in the improved PSO



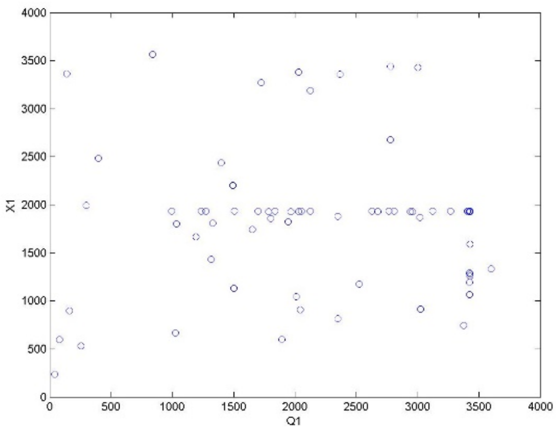
(b) Particle positions at the 100th computing step in the standard PSO



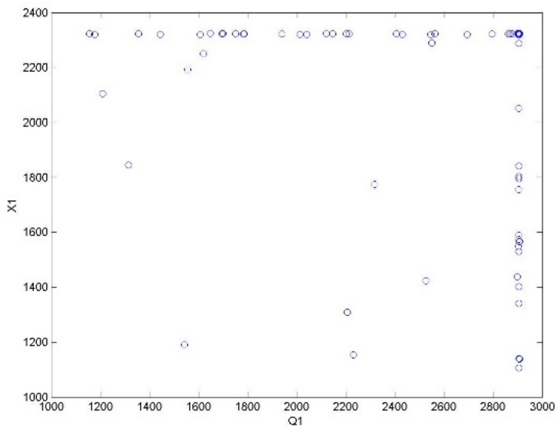
(c) Particle positions at the 200th computing step in the improved PSO



(d) Particle positions at the 200th computing step in the standard PSO

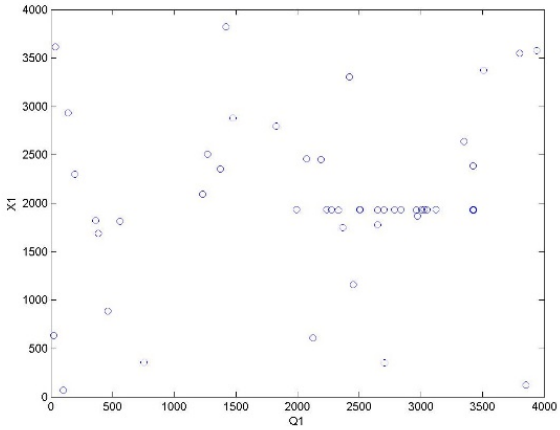


(e) Particle positions at the 300th computing step in the improved PSO

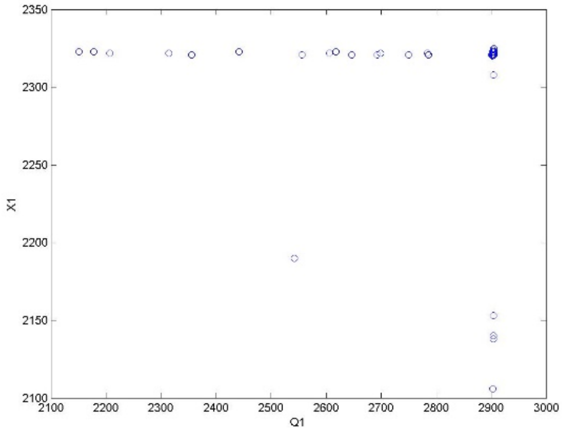


(f) Particle positions at the 300th computing step in the standard PSO

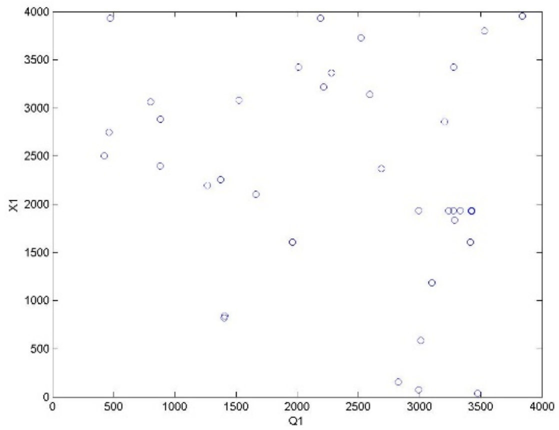
Fig. 3. Position distribution of particles in the improved PSO and the standard PSO



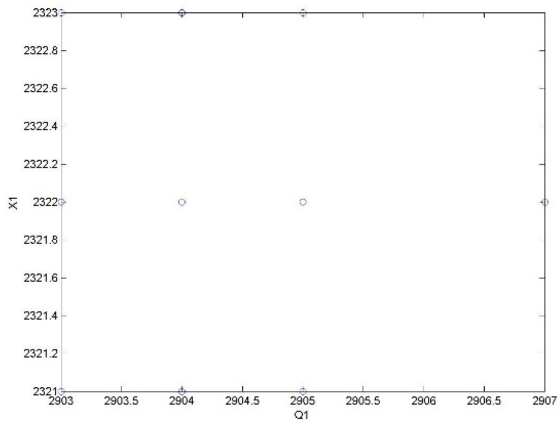
(g) Particle positions at the 400th computing step in the improved PSO



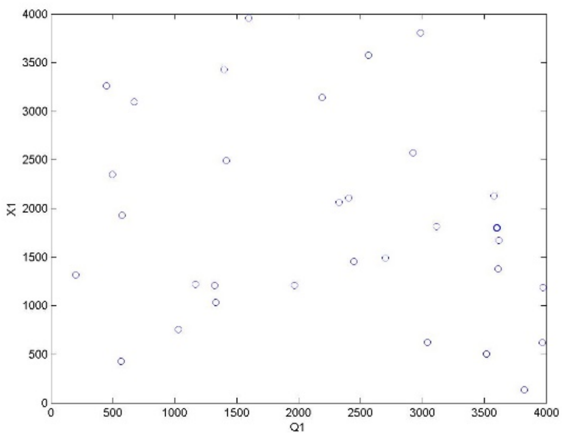
(h) Particle positions at the 400th computing step in the standard PSO



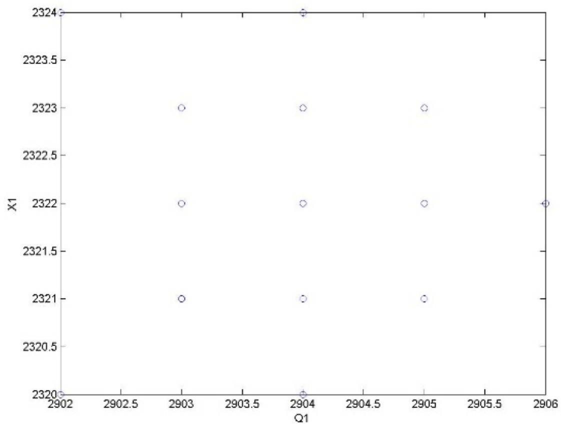
(i) Particle positions at the 500th computing step in the improved PSO



(j) Particle positions at the 500th computing step in the standard PSO



(k) Particle positions at the 600th computing step in the improved PSO



(l) Particle positions at the 600th computing step in the standard PSO

Fig. 3. Position distribution of particles in the improved PSO and the standard PSO (continue)

As can be found in Fig. 3, the improved PSO is superior, since its search results are more accurate and its functionality and cooperation of particles are more rational, while rectifying the standard PSO's shortcoming of being limited by local extremums. In a word, the improved PSO successfully accomplishes the search computing for the integer programming problem in this study.

6 Conclusions

With the improved PSO, this study solves the problem of production planning of an automotive group by translating a mathematical problem of constrained integer linear programming into a particle swarm space search process. The following findings are offered:

(1) A mathematical model is established for the production planning of the automotive group. PSO is used for search computing, and the penalty function and round function are implemented to solve the constraint condition and integer programming problem. Finally, the optimal solution obtained is $Q_1 = 3600$, $X_1 = 1800$; and the maximum total revenue is Max(Z) = RMB 684,000,000 yuan.

(2) To solve the problem that the standard PSO is very likely to encounter a local extremum, an improved PSO is proposed, under which 20% of the particles would renew their positions randomly, while the others renew their positions as per the standard PSO. The computing results in this study indicate that this method can deliver higher search efficiency and more accurate search computing results.

(3) After analyzing the particle positions' distribution during the computing both in the improved PSO and standard PSO, this study proves that the standard PSO is very likely to encounter a local extremum, and its search efficiency is very low in later stages, during which most particles would search near the extremum point. The improved PSO enables more rational functionality and cooperation between particles, because some particles that have been randomly distributed in the search space can continue the search computing even after the extremum points are found in later stages.

It must be pointed out that the improved particle swarm optimization algorithm in this paper transforms the integer linear programming problem into a spatial search problem, and calculates from the random solution. Therefore, there are some limitations in this study, and how to search efficiently is the future research direction.

Acknowledgements

This study was supported by the key project of humanities and social sciences of Chongqing education commission (Grant No. 18SKSJ093), I gratefully acknowledge this support.

References

[1] Harvard Management Series, China International Translation and publishing company, Beijing, China, 1985 (in Chinese).

[2] K.S. Lee, Z.W. Geem, A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice, Computer Methods in Applied Mechanics and Engineering 194(36-38)(2005) 3902-3933.

[3] A. Chaabane, A. Ramudhin, M. Paquet, Design of sustainable supply chains under the emission trading scheme, International Journal of Production Economics 135(1)(2012) 37-49.

[4] A. Parisio, E. Rikos, L. Glielmo, A Model Predictive Control Approach to Microgrid Operation Optimization, IEEE Transactions on Control Systems Technology 22(5)(2014) 1813-1827.

[5] C. Chen, J.H. Wang, F. Qiu, D.B. Zhao, Resilient Distribution System by Microgrids Formation After Natural Disasters, IEEE Transactions on Smart Grid 7(2)(2016) 958-966.

[6] S.H. Amin, G.Q. Zhang, A multi-objective facility location model for closed-loop supply chain network under uncertain demand and return, Applied Mathematical Modelling 37(6)(2013) 4165-4176.

[7] S.H. Amin, G.Q. Zhang, An integrated model for closed-loop supply chain configuration and supplier selection: Multi-objective approach, Expert Systems with Applications 39(8)(2012) 6782-6791.

[8] S.A. Alumur, S. Nickel, F. Saldanha-Da-Gama, V. Verter, Multi-period reverse logistics network design, European Journal of Operational Research 220(1)(2012) 67-78.

[9] H. Pandzic, J.M. Morales, A.J. Conejo, I. Kuzle, Offering model for a virtual power plant based on stochastic programming, Applied Energy 105(2013) 282-292.

[10] P.O. Kriett, M. Salani, Optimal control of a residential microgrid, Energy 42(1)(2012) 321-330.

[11] B. Fu, C.X. Ouyang, C.S. Li, J.W. Wang, E. Gui, An Improved Mixed Integer Linear Programming Approach Based on Symmetry Diminishing for Unit Commitment of Hybrid Power System, Energies 12(5)(2019) 833.

[12] J. Kennedy, R. Eberhart, Particle Swarm Optimization, Proceedings of IEEE International Conference on Neural Networks IV, 1995.

[13] N.K. Jain, U. Nangia, J. Jain, A review of particle swarm optimization, Journal of The Institution of Engineers (India): Series B 99(2018) 407-411.