Metadata of the chapter that will be visualized in SpringerLink

Book Title	Bio-inspired Computing: Theories and Applications
Series Title	
Chapter Title	Modified Mixed-Dimension Chaotic Particle Swarm Optimization for Liner Route Planning with Empty Container Repositioning
Copyright Year	2018
Copyright HolderName	Springer Nature Singapore Pte Ltd.

Author	Family Name	**Yu**
	Particle	
	Given Name	**Mingzhu**
	Prefix	
	Suffix	
	Role	
	Division	Department of Transportation Engineering, College of Civil Engineering
	Organization	Shenzhen University
	Address	Shenzhen, 518060, China
	Email	
Author	Family Name	**Chen**
	Particle	
	Given Name	**Zhichuan**
	Prefix	
	Suffix	
	Role	
	Division	Department of Transportation Engineering, College of Civil Engineering
	Organization	Shenzhen University
	Address	Shenzhen, 518060, China
	Email	
Corresponding Author	Family Name	**Chen**
	Particle	
	Given Name	**Li**
	Prefix	
	Suffix	
	Role	
	Division	College of Management
	Organization	Shenzhen University
	Address	Shenzhen, 518060, China
	Email	Cljx2005@163.com
Author	Family Name	**Qu**
	Particle	
	Given Name	**Rong**
	Prefix	
	Suffix	

	Role	
	Division	School of Computer Science
	Organization	University of Nottingham
	Address	Nottingham, NG8 1BB, UK
	Email	
Corresponding Author	Family Name	**Niu**
	Particle	
	Given Name	**Ben**
	Prefix	
	Suffix	
	Role	
	Division	College of Management
	Organization	Shenzhen University
	Address	Shenzhen, 518060, China
	Email	

Abstract	Empty container repositioning has become one of the important issues in ocean shipping industry. Researchers often solve these problems using linear programming or simulation. For large-scale problems, heuristic algorithms showed to be preferable due to their flexibility and scalability. In this paper we consider large-scale the liner routing planning problem with empty container repositioning (LRPECR) model where allocation strategies and liner routes need to be designed to allocate empty containers from the supply ports to the demand ports. According to the characteristics of the LRPECR model, we combine the path of the ship to the algorithm encoding, set up the fitness function that minimizes the total cost, and use a modified Particle Swarm Optimization (PSO) algorithm to search for optimal shipping routes in a feasible space iteratively. The modified PSO combines chaotic theory and Cat map to overcome the defect of traditional PSO. In addition, we perform chaotic search in different dimensions to enhance the search accuracy of the algorithm that means the increased diversity of search scope. In order to validate our algorithm, standard PSO and GA are chosen as the compared algorithms. Through numerical studies based on real applications, the experimental results demonstrate that the modified PSO is able to find preferable solutions efficiently for the empty container repositioning problem.

Keywords (separated by '-')	Empty container repositioning - Chaotic search - Particle swarm optimization - Cat map - Integer linear programming

Author Proof

Check for updates

Modified Mixed-Dimension Chaotic Particle Swarm Optimization for Liner Route Planning with Empty Container Repositioning

Mingzhu Yu[1], Zhichuan Chen[1], Li Chen[2(✉)], Rong Qu[3], and Ben Niu[2(✉)]

[1] Department of Transportation Engineering, College of Civil Engineering, Shenzhen University, Shenzhen 518060, China

[2] College of Management, Shenzhen University, Shenzhen 518060, China
Cljx2005@163.com

[3] School of Computer Science, University of Nottingham, Nottingham NG8 1BB, UK

Abstract. Empty container repositioning has become one of the important issues in ocean shipping industry. Researchers often solve these problems using linear programming or simulation. For large-scale problems, heuristic algorithms showed to be preferable due to their flexibility and scalability. In this paper we consider large-scale the liner routing planning problem with empty container repositioning (LRPECR) model where allocation strategies and liner routes need to be designed to allocate empty containers from the supply ports to the demand ports. According to the characteristics of the LRPECR model, we combine the path of the ship to the algorithm encoding, set up the fitness function that minimizes the total cost, and use a modified Particle Swarm Optimization (PSO) algorithm to search for optimal shipping routes in a feasible space iteratively. The modified PSO combines chaotic theory and Cat map to overcome the defect of traditional PSO. In addition, we perform chaotic search in different dimensions to enhance the search accuracy of the algorithm that means the increased diversity of search scope. In order to validate our algorithm, standard PSO and GA are chosen as the compared algorithms. Through numerical studies based on real applications, the experimental results demonstrate that the modified PSO is able to find preferable solutions efficiently for the empty container repositioning problem. AQ1

Keywords: Empty container repositioning · Chaotic search · Particle swarm optimization · Cat map · Integer linear programming

1 Introduction

The liner routing problem (LRP) aims to establish a reasonable liner service shipping network between several supply and demand ports. Considering the transportation cost, risk and shipping capacity, shipping routes should be planned between ports. LRP is a combinatorial optimization problem where the ocean carriers plan and deploy the liner routes rationally to satisfy customer demands and to maximize profit.

© Springer Nature Singapore Pte Ltd. 2018
J. Qiao et al. (Eds.): BIC-TA 2018, CCIS 952, pp. 1–12, 2018.
https://doi.org/10.1007/978-981-13-2829-9_27

Author Proof

As an essential issue in contemporary LRP, the efficient and effective repositioning of empty containers can significantly reduce the operation costs of shipping companies. At the same time, it also has positive effect on environment protection and sustainability [1, 2]. Shintani and Imai [3] solved the empty containers repositioning using a Genetic Algorithm (GA) for the first time. Based on this, Sun [4] proposed a model for empty container repositioning and solved it using hybrid genetic algorithm (HGA).

Particle Swarm Optimization (PSO) [5, 6] is one of the most widely-used evolutionary algorithms. However, traditional PSO is prone to suffer from being trapped in local optima, leading to premature convergence [7]. To overcome this defect, many improvements have been proposed, including chaotic search, one of the powerful hybrid algorithms. Chaotic searching algorithm was proposed first by Changkyu et al. [8]. Liu et al. [9] developed an improved PSO combined with chaotic searching algorithm. Tan et al. [10] found that the single-dimension chaotic search can evidently improve the algorithm precision and the efficiency of the chaotic search. On the other hand, Logistic map is frequently used for generating chaos sequence in the majority of chaotic search algorithms. Based on the research in the field of image encryption [11], Wang et al. [12] replaced the Logistic map with Cat map as the chaos sequence generator. The superior properties of Cat map, which are excellent ergodicity and sensitive dependence on initial conditions, were taken by Wang in chaotic search algorithm. Cat map overcomes the disadvantages of Logistic map, including non-uniformity and frequent loop.

Based on above mentioned researches, this paper proposes to solve the liner route planning problem considering empty container repositioning (LRPECR) [4] using a new modified PSO, which assimilated the experiences of chaotic search with Cat map [12] and single-dimensional and multi-dimensional search [10]. We name it as Modified Mixed-dimension Chaotic Particle Swarm Optimization (MDCPSO). The optimization result of MDCPSO are compared with those of standard PSO, GA and HGA.

The rest of the article is organized as follows. Section 2 provides a brief description of MDCPSO. Section 3 presents the LRPECR model and problem descriptions. The results and analysis of the experiments are shown and discussed in Sect. 4. Finally, the concluding remarks are provided in Sect. 5.

2 Modified Mixed-Dimension Chaotic Particle Swarm Optimization

2.1 Particle Swarm Optimization

PSO is a collection of intelligence optimization techniques. The system is initialized with a set of random solutions called “particles”, which move through the search pace to towards the optimal location by iterations. The search of PSO aims to strike a balance between exploration and exploitation. For more detailed information, please refer to [6].

Author Proof

2.2 Modified Mixed-Dimension Chaotic Particle Swarm Optimization

The standard PSO and GA show to usually suffer from being trapped into local optima. Inspired by the research by Tan and Wang [11, 12], the MDCPSO algorithm has been developed to enhance the performance of the standard PSO. The operators of MDCPSO include the chaotic search with cat map, and multi-dimension and single-dimension chaotic search methods.

Two Key Mechanisms of MDCPSO. Before describing how MDCPSO solves the LRPECR model, a brief introduction of two key mechanisms MDCPSO, namely the chaotic search mechanism and multi-dimension and single-dimension chaotic search, is given as follows.

Chaotic Search Mechanism with Cat Map. In order to improve the capability of PSO to escape from the local optimum, the chaotic search is used for constructing the algorithm. Logistic map is used in chaotic search frequently. It can map a number to a set of 0 to 1. Through multiple iterations, the values mapped by Logistic equation will traverse the whole set to achieve chaotic effect. Equation (1) shows the equation of Logistic map. $cx^{(0)}$ is a chaotic variable generated by the global best particles in the PSO system. And n represents the current chaotic iterations.

$$cx^{(n+1)} = 4cx^{(n)}(1 - cx^{(n)}), 0 < cx^{(0)} < 1. \quad (1)$$

In this paper, the chaotic Cat map is used to replace the traditional chaotic Logistic map. Compared with the Logistic map, Cat map as a chaos sequence generator shows to enrich the chaotic search behavior, because it can traverse the whole set of 0 to 1 faster [12]. Equation (2) shows the computational formula of Cat map.

$$\begin{cases} cx^{(n+1)} = (cx^{(n)} + y^{(n)}) mod 1 \\ y^{(n+1)} = (cx^{(n)} + 2y^{(n)}) mod 1 \end{cases}, 0 < cx^{(0)} < 1, 0 < y^{(0)} < 1. \quad (2)$$

Here, *mod* is the modulus operator, and the result will be return to $cx^{(n+1)} \cdot y^{(0)}$. will be created randomly.

Multi-dimension and Single-Dimension Chaotic Search Method. Multi-dimension chaotic search means to map the value of all dimensions by chaotic map function. The existing chaos search method mainly use multi-dimensional chaotic search. However, compared to multi-dimensional chaotic search, single-dimensional search can produce a better search accuracy, because it only changes the value of one single dimension, increasing diversity of search scope. Combining the two different mechanisms, we propose a new method as follows.

Step 1: Initialize parameters of the chaotic system, with a random iteration number $y^{(0)} \left(y^{(0)} \in [0,1]\right)$, iteration counter $n = 1$, the maximum iteration number M of the chaotic system, and a specified input solution x_{spec}.

Author Proof

Step 2: Mapping the specified solution x_{spec} to a set between [0, 1] according Eq. (3). $x_{max,i}$, $x_{min,i}$ represent the maximum and minimum values for each dimension of x_{spec}, respectively. d is the maximum dimension of x_{spec}.

$$cx_i^{(1)} = \frac{x_{spec,i} - x_{\min,i}}{x_{\max,i} - x_{\min,i}}, i = 1, 2, \ldots, d \tag{3}$$

Step 3: Generate a chaotic sequence $cx^{(n+1)}$ by Cat map using Eq. (2).
Step 4: Generate a random number r, r is between [0, 1].
Step 5: If $r < 0.5$, a multi-dimension chaotic local search is performed on $cx^{(n)}$ according to Eq. (4). Here, c represents one of the solution sequences generated by the chaotic search; α is the step length of the chaotic search. In this work, α is set to a random integer number between [−2, 2].

$$c_i^{(n)} = \alpha \cdot cx_i^{(n+1)} + x_{spec,i}, i = 1, 2, \ldots, d \tag{4}$$

Step 6: If $r \geq 0.5$, a single-dimension chaotic local search is performed on $cx^{(n)}$ according to Eq. (5). What is different from **Step 5** is that the single-dimension search does not need to perform chaotic search on each dimension. It only needs to randomly select one of the dimensions to perform the chaotic search.

$$c_i^{(n)} = \alpha \cdot cx_i^{(n+1)} + x_{spec,i}, i = randint(1, d) \tag{5}$$

Step 7: Set iteration counter as $n + 1$. Go to Step 3 if $n \leq M$.
Step 8: Calculate fitness values of all solutions in c and store the best one.
Step 9: Replace x_{spec} by the best solution in c if the fitness value of the best solution in c is better than that of x_{spec}, otherwise x_{spec} will be maintained as the best solution.
Step 10: Stop the chaotic search and output the best solution.

3 MDCPSO for Liner Route Planning

3.1 The LRPECR Model

In this paper the same LRPECR model proposed by Sun in [4] is considered. In the LRPECR model, assignments of repositioning empty containers are required to the known demand ports. The LRPECR problem aims to design liner routes considering the satisfaction of empty containers which need to be transferred from the supply ports to the demand ports. The objective of this model is to plan the liner route with the minimized total cost. The problem is a mixed integer linear programming, and was solved by Sun in [4] using a hybrid GA. The LRPECR model is presented in Eqs. (6–12) and the variable definitions are shown in Table 1.

Mathematically, the objective is to minimize the total cost. The first item of the objective function represents all the costs, including loading cost, unloading cost and

Author Proof

Table 1. Parameters and definitions of LRPECR model

Variables	Definitions
I	The set of all container ports
M	The set of all empty container demand ports
X_{Tj}	The set of all alternative shipping liners to the demand port j, $j \in M$
C_{Lij}	The unit cost of loading an empty container from port i to demand port j, $i \in I, j \in M$
C_{Titj}	The unit cost of repositioning an empty container from port i to demand port j through the shipping liner t, $i \in I, j \in M, t \in X_{Tj}$
C_{Uij}	The unit cost of unloading an empty container from port i to demand port j, $i \in I, j \in M$
C_{Rj}	The unit cost of renting and loading an empty container in port j, $j \in M$
C_{Sj}	The unit cost of repositioning an empty container from the container renter to port j, $j \in M$
D_{Ej}	The demand volume of empty container in demand port j, $j \in M$
S_{Ni}	The volume of available empty container in port i, $i \in I$
Decision variables	Definitions
x_{ij}^E	The volume of empty container repositioned from port i to demand port j, $i \in I, j \in M$
x_{itj}	$x_{itj} = 1$ means that shipping liner t is chosen from X_{Tj}, and port i serves as the starting port, $i \in I, j \in M, t \in X_{Tj}$
x_j^R	The volume of empty container rented from renter to port j, $i \in I, j \in M$

the cost of shipping a container from port i to the known demand port. The second item means the cost of renting containers in port i.

$$\min z = \sum_{j \in M} \sum_{i \in I} \sum_{t \in X_{Tj}} ((C_{Titj} + C_{Lij} + C_{Uij}) \times x_{ij}^E + (C_{Sj} + C_{Rj}) \times x_j^R)$$

Subject to:

$$x_{ij}^E + x_j^R = D_{Ej}, \forall i \in I, \forall j \in M \tag{6}$$

$$\sum_{j \in M} x_{ij}^E \le S_{Ni}, \forall i \in I \tag{7}$$

$$x_{ij}^E, x_j^R \ge 0, \forall i \in I, \forall j \in M \tag{8}$$

$$\sum_{t \in X_{Tj}} x_{itj} \le 1, \forall i \in I, \forall j \in M \tag{9}$$

Author Proof

$$\sum_{t \in X_{Tj}} x_{itj} \times S_{Ni} \geq x_{ij}^{E}, \forall i \in I, \forall j \in M \tag{10}$$

$$M \times x_{ij}^{E} \geq \sum_{t \in X_{Tj}} x_{itj}, \forall i \in I, \forall j \in M \tag{11}$$

$$x_{itj} = \{0, 1\}, \forall i \in I, \forall t \in X_{Tj}, \forall j \in M \tag{12}$$

Equation (6) means that the volume of empty containers repositioned and rented at a certain port should equals to the empty container demand. In Eq. (7) the repositioning volume should be no more than the volume that the port can supply. Equation (8) ensures that the volume of transportation should not be less than 0. Equation (9) defines that there is only one shipping route for one assignment. According to Eqs. (10–11), once the assignment between port i to port j is determined, the transfer volume of empty containers between them must be more than 0, and the first M represents a sufficiently large number. Equation (12) restricts the value of x_{itj} to be 0 or 1.

3.2 MDCPSO for LRPECR Model

Encoding. A new encoding strategy is developed considering the characteristic of the LRPECR model. According to real problem data, a vessel will not berth at more than four ports in one shipping mission, because of the high cost. It is better to adopt the strategy of renting containers at the demand port. Therefore, a sequence of four numbers is used to represent a liner route for one shipping mission. When there are m shipping missions, a particle with $4 \times m$ dimensions is used.

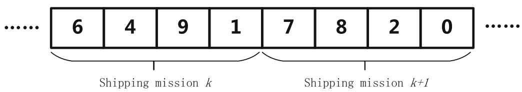

Fig. 1. An example for the encoding scheme for LRPECR

As Fig. 1 shows, in shipping mission k, four positions represent the ports on this route from the supply port to the demand port. The first position represents the demand port, and the second position is for the linked port of the demand port, and so on. The shipping route of mission k is thus port 1 $\rightarrow$ port 9 $\rightarrow$ port 4 $\rightarrow$ port 6. If the number of ports on the liner route is less than four (as shown in shipping mission $k + 1$), the rest of positions will be filled with 0. The shipping route of mission $k + 1$ is port 2 $\rightarrow$ port 8 $\rightarrow$ port 7 in Fig. 1.

Fitness. The location of every particle is different in the search space. The fitness is calculated according to Eq. (13). A greater fitness value means that the particle stays in a better location in the search space, so it will be preserved and to be adapted by the particles with a higher probability in the next iteration. Those particles with poor fitness values will be improved or even eliminated. In this paper, the fitness equation is the same as the objective function.

Author Proof

$$\text{fitness} = \sum_{j \in M} \sum_{i \in I} \sum_{t \in X_{Tj}} (x_{itj} \times (C_{Titj} + C_{Lij} + C_{Uij}) \times x_{ij}^{E} + (C_{Sij} + C_{Rij}) \times x_{ij}^{R}) \tag{13}$$

Computational Steps of the MDCPSO Algorithm for LRPECR. Based on the mechanisms described above, an improved PSO algorithm is designed for solving LRPECR, and is implemented in MATLAB. The standard PSO and GA are chosen as the compared algorithms. In order to make fair comparisons, all algorithms compared use the same encoding, population sizes and iteration numbers. The length of encoding, population size and iteration number L are all set to 200, 50 and 1000. In both PSO and MDCPSO, learning factors $C1$ and $C2$ are set to 1.5. In GA, crossover rate Pc and mutation rate Pm are set to 0.8 and 0.09. The pseudo-code of MDCPSO for LRPECR is shown in Table 2.

Table 2. The pseudo-code of MDCPSO

```
Begin
Initialize parameters;
Initialize a population of particles with random locations and velocities in a domain of feasible
solution, and evaluate fitness value f_i for each particle according to Eq. (13);
Initialize P_g with the best particle within the population;
Initialize P_i with a copy of each particle's location;
For (l = 1 : L ):
  Update velocities and locations for each particle;
  Adjust the location for those particles beyond the boundary of the domain;
  Evaluate the fitness f_i for all particles;
  Update the P_g and P_i ;
  Perform the mixed-dimension chaotic search on P_g (see Section 2.2 Multi-dimension and
  Single-dimension Chaotic Search Method.);
End
Output: the best solution for the LRPECR model
```

4 Experimental Results and Analysis

4.1 Experimental Data

The characteristics of the numerical examples are set the same as those in the LRPECR model in Sun [4]. Computational studies have been conducted on 5 instances. The total number of ports and missions in Instance 1 is 28 and 50, respectively.

Figure 2 shows the distribution of the ports and empty container status for each port in Instance 1. There are 28 ports distributed in this network. The number 0 means that the port needs to be supplemented by transportation form supply ports or renting. In Sun's case study, it was found that the cost of adopting a full leasing strategy was lower

Author Proof

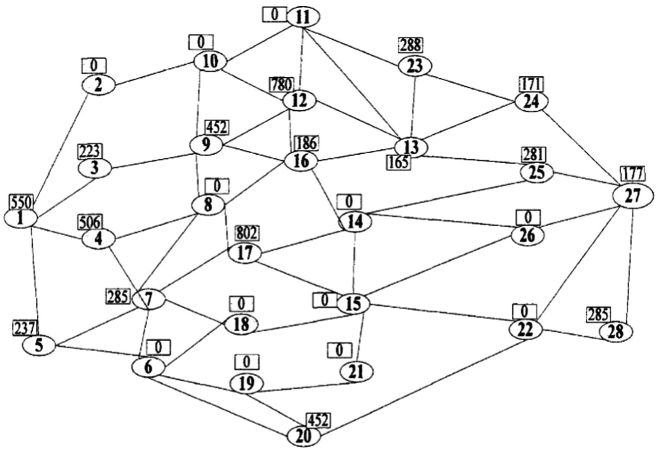

Fig. 2. Distribution and empty container status in Instance 1, Sun [4]

than the cost obtained by their optimization. The original data has thus been adjusted by increasing the cost of renting empty containers. There are 50 missions of assignment, and the target port and other details of parameters are shown in Appendix. For all instances 10 runs are conducted to obtain the average objective values and computation times.

4.2 Results and Analysis

Figure 3 and Table 3 show the results of Instance 1. MDCPSO shows to outperform GA and standard PSO. In the first 50 iterations, results from MDCPSO have exceeded the other two. In Fig. 3 we can observe that the convergence speed of MDCPSO is

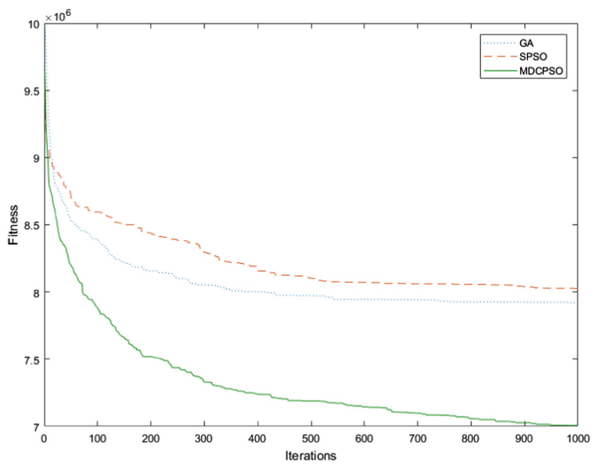

Fig. 3. Convergence of GA, PSO and MDCPSO for Instance 1

Author Proof

much faster than GA and PSO. That means MDCPSO can find a better solution in a short period of time. After 300 generations, both GA and PSO show to have converged, however MDCPSO still maintains an improving trend towards better solutions.

Table 3. The average results for Instance 1

	GA	SPSO	MDCPSO
Average fitness	7.9214e+06	8.0269e+06	7.0075e+06

Table 4. The optimized empty container allocation strategies based on MDCPSO of mission 1 to 16

Mission	Route	x_{ij}^E	x_j^R	Mission	Route	x_{ij}^E	x_j^R
1	1-1	260	0	9	5-5	97	13
2	2-2	0	500	10	6-6	0	100
3	2-2	0	340	11	6-6	0	110
4	3-3	130	0	12	6-6	0	310
5	1-3	290	100	13	6-6	0	150
6	3-3	93	37	14	16-8-7	150	0
7	4-8-9-3	450	0	15	7-7	285	35
8	5-5	140	0	16	9-9	300	0

Table 5. The optimized empty container allocation strategies based on MDCPSO of mission 17 to 50

Mission	Route	x_{ij}^E	x_j^R	Mission	Route	x_{ij}^E	x_j^R
17	9-9	152	238	34	20-20	170	0
18	9-9	0	260	35	20-20	60	0
19	12-12	230	0	36	17-15-21	380	0
20	12-12	180	0	37	17-14-15-21	110	0
21	13-13	90	0	38	17-15-21	172	8
22	12-13-24-13	230	0	39	20-22	170	0
23	4-8-16-14	56	44	40	28-22	285	215
24	13-25-14	36	254	41	23-23	288	190
25	12-16-14	140	50	42	23-23	0	190
26	13-25-14	75	195	43	23-23	0	170
27	15-15	0	200	44	24-24	90	0
28	15-15	0	120	45	24-24	81	129
29	15-15	0	80	46	25-25	281	149
30	15-15	0	330	47	25-25	0	150
31	17-17	140	0	48	26-26	0	230
32	17-17	0	100	49	27-27	177	193
33	18-18	0	110	50	20-22-27	52	158

Author Proof

MDCPSO has been run 10 times in MATLAB, leading to the best solution of $6,845,666, while the best result from HGA by Sun is $11,387,000 [4]. Note that the renting cost of container has been raised based on Sun's model. MDCPSO shows to perform better than HGA in solving the RLPECR model. The detailed liner route plan and strategies of the empty container allocation are shown in Tables 4 and 5.

In order to confirm the effectiveness of MDCPSO in different scale of RLPECR, another four instances with different numbers of ports and assignments have been tested. They respectively are Instance 2 (35 assignments in 25 ports), Instance 3 (65 assignments in 31 ports), Instance 4 (80 assignments in 34 ports) and Instance 5 (95 assignments in 37 ports).

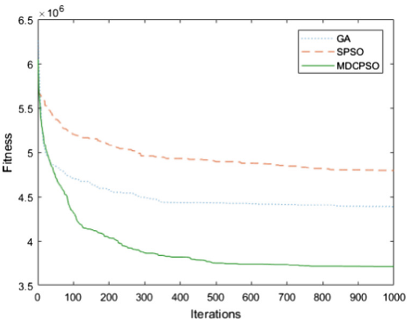

Fig. 4. Comparisons of algorithms for Instance 2

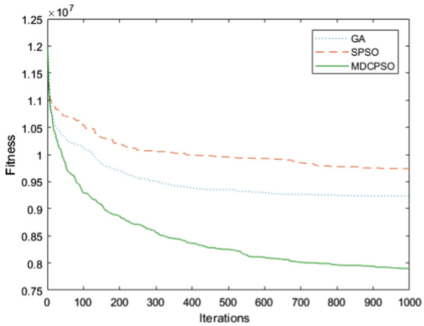

Fig. 5. Comparisons of algorithms for Instance 3

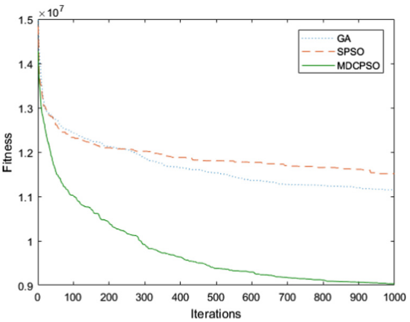

Fig. 6. Comparisons of algorithms for Instance 4

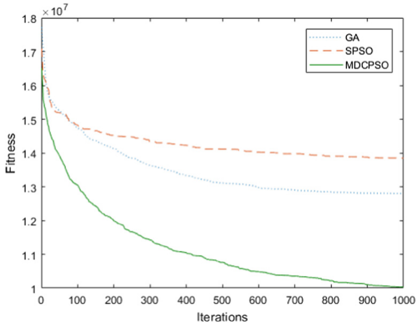

Fig. 7. Comparisons of algorithms for Instance 5

Based on above experiment results in Figs. 4, 5, 6 and 7 and Table 6, we can conclude that MDCPSO performs significantly better than PSO and GA in all instances. In addition, the convergence of standard PSO has shown to be always inferior to GA. When the mixed-dimension chaotic search with Cat map is integrated with PSO, the improvement is significant. It proves the effectiveness of those mechanisms on PSO.

Author Proof

Table 6. The average results from 10 runs of the three algorithms

Scale	GA	SPSO	MDCPSO
Instance 2	4.3839e+06	4.7942e+06	**3.7108e+06**
Instance 3	9.2304e+06	9.7392e+06	**7.8939e+06**
Instance 4	1.1147e+07	1.1514e+07	**9.0342e+06**
Instance 5	1.2797e+07	1.3845e+07	**1.0034e+07**

5 Conclusions

This paper proposed an improved new particle swarm optimization (PSO) algorithm, namely MDCPSO, for solving the liner routing planning problem with empty container repositioning (LRPECR) model based on chaotic PSO. MDCPSO employs the powerful search capability of the chaotic algorithm with Cat map and the superior searching precision of the mixed-dimension search. In order to evaluate the effectiveness of MDCPSO, two widely used algorithms, GA and PSO, are compared. The experimental results show that the performance of MDCPSO is outstanding in solving LRPECR. In the future, the key mechanisms of MDCPSO will be combined or integrated with other heuristic algorithms for addressing the LRPECR model with extended constraints.

Acknowledgment. This work is partially supported by The National Natural Science Foundation of China (Grants Nos. 71571120, 71271140, 61472257), Natural Science Foundation of Guangdong Province (2016A030310074).

References

1. Dong Ping, S., Jonathan, C.: Empty container repositioning in liner shipping. Marit. Policy Manag. **36**(4), 291–307 (2009)
2. Yoonjea, J., Subrata, S., Debajypti, C., Ilkyeong, M.: Direct shipping service routes with an empty container management strategy. Transp. Res. Part E Logist. Transp. Rev. **118**, 123–142 (2018)
3. Shintani, K., Imai, A.: The container shipping network design problem with empty container repositioning. Transp. Res. Part E Logist. Transp. Rev. **43**(1), 39–59 (2007)
4. Sun, J.: Optimizing empty container allocation based on hybrid genetic algorithm. Dalian Maritime University (2009). in Chinese
5. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: IEEE International Symposium on MICRO Machine and Human Science, pp. 39–43 (2002)
6. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: 1995 Proceedings of IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)
7. Van den Bergh, F.: An analysis of particle swarm optimization. Ph.D. dissertation, Department of Computer Science, University of Pretorua, Pretoria (2006) AQ2
8. Changkyu, C., Ju-Jang, L.: Chaotic local search algorithm. Artif. Life Robot. **2**(1), 41–47 (1998)
9. Liu, B., Wang, L., Jin, Y.H., Tang, F., Huang, D.X.: Improved particle swarm optimization combined with chaos. Chaos Solitons Fractals **25**(5), 1261–1271 (2005)

Author Proof

10. Tan, Y., Tan, G.Z., Deng, S.G.: Hybrid particle swarm optimization with differential evolution and chaotic local search to solve reliability-redundancy allocation problems. J. Cent. South Univ. (Engl. Ed.) **20**(6), 1572–1581 (2013)
11. Chen, G., Mao, Y., Chui, C.K.: A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos Solitons Fractals **21**(3), 749–761 (2004)
12. Wang, F., Dai, Y.S., Wang, S.S.: Modified chaos-genetic algorithm. Comput. Eng. Appl. **46** (6), 29–32 (2010)

Author Proof

Author Query Form

Book ID : **474701_1_En**

Chapter No : **27**

Springer
the language of science

Please ensure you fill out your response to the queries raised below and return this form along with your corrections.

Dear Author,
During the process of typesetting your chapter, the following queries have arisen. Please check your typeset proof carefully against the queries listed below and mark the necessary changes either directly on the proof/online grid or in the 'Author's response' area provided below

Query Refs.	Details Required	Author's Response
AQ1	Kindly provide email address for the corresponding author.	
AQ2	Please check and confirm if the author names and initials are correctly identified in Ref. [7].	