METHODOLOGIES AND APPLICATION

A hybrid particle swarm optimization algorithm for RFID network planning

Yating Cao[1] · Jing Liu[1] · Zhouwu Xu[1]

Abstract

The radio frequency identification (RFID) technology is widely used for object identification and tracking applications, which brings the most challenging RFID network planning (RNP) problem. However, existing RNP methods have some defects, such as the number of readers is uncertain and objectives conflict each other. In this paper, we propose a hybrid particle swarm optimization algorithm with K-means clustering and virtual forces for RNP, which is named as HPSO-RNP. HPSO-RNP can search the number of readers automatically and initialize the coordinates of readers through the K-means algorithms. Virtual force is integrated into the random movement to adjust the location of readers during the search process of PSO. Moreover, we consider four objective functions in a hierarchical manner. To compare HPSO-RNP with the existing method, extensive experiments are conducted on eight RNP benchmark datasets and the results validate that the performance of the proposed method is superior for planning RFID networks in terms of the number of readers, interference, power and load balance.

Keywords Particle swarm optimization · Radio frequency identification · RFID network planning · K-means algorithm · Virtual force

1 Introduction

The Radio Frequency Identification (RFID) technology is a kind of wireless communication technology that can be used to identify targets automatically. The RFID technology has been widely used in many fields with the rapid development of wireless sensing technology and automatic identification technology in recent years, such as supply chain management, inventory management, and innovative applications (Wan 1999; Krohn et al. 2005).

A typical RFID system usually consists of three components: RFID readers, tags, and a computer system linked to readers. The communication range of an RFID system is limited. Multiple readers must be deployed in the network. It can also cause conflicts between readers, making the tag information unreadable. Therefore, we need to determine the total number of readers and locations where they should be placed to cover tags efficiently. More and more researchers regard this problem as a network planning problem, namely, RFID network planning (RNP) problem. It is an NP-hard problem with many targets (coverage, interference between readers, power, load balance) and many constraints (the coordinates of readers and the radius of power) (Gong et al. 2012; Lu and Yu 2014; Dong et al. 2007).

At present, considerable achievements have been made for RNP problems (Huang and Chang 2011; Chen et al. 2010a, b). Most of the existing methods are based on evolutionary algorithms (EAs) and swarm intelligence (SI) optimization techniques (Chen and Zhu 2008), such as genetic algorithm (GA) (Guan et al. 2006; Yang et al. 2009), particle swarm optimization (PSO) (Bhattacharya and Roy 2010; Gong et al. 2012, 2014; Antonis et al. 2019), evolutionary strategy (ES), differential evolution (DE) (Seok et al. 2010), and improved on these algorithms, such as decomposition-based multiple objective evolutionary algorithms (Zhang and Li 2007; Özdemir et al. 2013; Tuba et al. 2017; Xu et al. 2018; Zhang et al. 2016; Zhao et al. 2017) and cooperative artificial bee colony

✉ Jing Liu
   neouma@163.com
   https://faculty.xidian.edu.cn/LJ22/zh_CN/index.htm

[1] School of Artificial Intelligence, Xidian University, Xi'an 710071, China

algorithm (Ma et al. 2014). In (Liu et al. 2016), an efficient and fast kinematics-based algorithm is developed for optimizing RFID networks. In the literature, only a few works consider adjusting the number of readers automatically (Zhang and Liu 2017). And these algorithms usually put excessive readers into the working area initially and then remove readers one by one until the number of readers is minimum. Most of these algorithms need a predefined number of readers at the beginning of the algorithm (Antonis et al. 2019). Presets can affect the quality of network planning seriously. If the number of readers is set to too small, the tags cannot be covered completely in RFID systems. On the contrary, redundant readers not only waste resources but also cause interference. Moreover, almost all the previous algorithms for RNP apply a weighted-sum method to optimize the combined objectives. However, it is noticed that the priority of four objectives for RNP is clear in applications (Antonis et al. 2019; Zhang and Liu 2017).

In this paper, we propose a hybrid particle swarm optimization algorithm with K-means clustering (Harrington 2013) and virtual forces, named as HPSO-RNP (Chen et al. 2011), to solve RNP. The number of readers is determined automatically and coordinates of readers are initialized through the K-means algorithm. The new reader is kept putting into the working space until tags are covered within an assumed radius. During the evolutionary process of our proposed PSO algorithm, we use a hierarchical approach to solve multiple objectives of RNP (MORNP) according to the priority. Moreover, we introduce the virtual force when particles are moving. There is a repulsive force between readers that cover the same tags. And unoverwritten tags are attracted to the readers (Chen et al. 2010a, b).

As far as RFID network planning is concerned, there is still space for research for static tag coverage problems. Compared with the existing algorithm, our method has advantages in many aspects. The main contributions of this paper are summarized as follows.

1. The K-means algorithm is used to search the number of readers automatically and initialize the location of readers.
2. To increase the search capability of the proposed algorithm, we introduce virtual forces to the PSO algorithm. This method improves tag coverage and reduces the interference rate between readers.
3. Achieve the optimal planning solutions by hierarchically optimizing four conflicting objectives in MORNP.

In the experiments, eight RNP instances are used to test the performance of HPSO-RNP, and the CA-RNP and MOEAD-RNP (Zhang and Liu 2017; Liu et al. 2016; Zhao et al. 2017; Shinde et al. 2019) are taken as the baseline. The results show that HPSO-RNP obtains promising results in solving the RNP problem. It has better performance in

terms of the number of readers, interference between readers, power, and load balance.

The rest of the paper is organized as follows. An RFID network planning model is formulated in Sect. 2. Afterward, a novel PSO algorithm is proposed for solving the RNP in Sect. 3. The experimental results are presented in Sect. 4. Finally, conclusions are given in Sect. 5.

2 RFID network planning problem model

In this section, for RFID network planning, a mathematical optimization model is proposed to reflect many requirements in practice, such as coverage, interference between readers, power, and load balance. These can be regarded as evaluation indexes to evaluate the RFID system.

2.1 RFID system

A complete RFID system consists of tags, readers, antenna, middleware, interface, and host system software. We formulate a simplified model to account for the RFID network planning problem. Readers and tags transmit energy through electromagnetic waves and then read the tag information to complete the data transmission. The working principle of the RFID system is shown in Fig. 1.

2.1.1 Tag

Each tag is identified by a unique electronic identification code. They are divided into three types: passive tags, semi-passive tags, and active tags. The passive tags have no internal power supply. Semi-active tags are similar to the passive, but with a small power supply. The active tags have their own power supply. At present, passive labels are mainly used in practical applications because of the advantages of them compared with others. The core role of the tags is to respond to the request of readers and send out information contained in the Electronic Product Code (EPC) identifier.
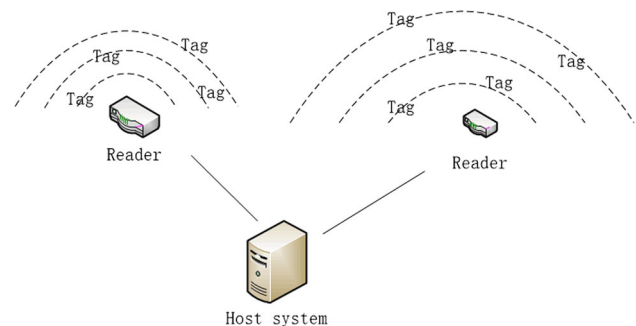

Fig.1 A typical RFID system and its components

2.1.2 Reader

The reader is the device that reads (and sometimes writes) tag information. Its antenna is mainly used to transmit radio frequency signals between tags and readers. Based on the Friis transmission equation, we can calculate the received power by a reader from a passive tag as follows:

$$P_R = P_T G_R^2 G_T^2 \left(\frac{\lambda}{4\pi d}\right)^4 T^2 \tag{1}$$

where P_T is the transmitted power of the reader's antenna, G_T and G_R are the gains of a reader's and a tag's antenna, respectively, d is the distance between the reader and the tag, λ is the wavelength, and T is the backscatter transmission. It can be seen from Eq. (1) that P_R is related to the distance d between readers and tags. Based on this observation, we simplify the model as follows. A tag is covered by a reader if it is within the communication range of the reader; otherwise, the communication between readers and tags is not established.

2.2 RFID system evaluation index

2.2.1 Coverage

For RFID network planning, coverage is the most important and necessary performance. Tags detected by the readers are called cover-points. Different coverage models have different definitions of coverage points. We assume that the coverage model is a Boolean model. If a tag is overwritten by any reader, the tag is the coverage point.

Thus, the overall coverage is defined as follows:

$$\text{COV} = \sum_{t=0}^{t=N_t} p(t, r)/N_t \times 100\% \tag{2}$$

The Boolean model is built as follows:

$$p(t,r) = \begin{cases} 1 & d(t,r) < = \text{Radius} \\ 0 & d(t,r) > \text{Radius} \end{cases} \tag{3}$$

In this case, RS is represented as the set of readers and TS is denoted as the set of tags. Here N_t is the number of tags, $d(t,r)$ is the Euclidean distance between a tag t and a reader r. Radius is used to represent the power radius of readers. If there is a reader $r \in RS$ and $d(t, r) < = R$, the tag t is covered. The algorithm is expected to deploy readers to increase coverage using fewer readers.

In short, the metric has the highest priority. In the RFID network, in order to establish communication between readers and tags, it is necessary for tags to be within the range of readers.

2.2.2 Interference

For large-scale RFID systems, there will be overlap between the working-scope of readers, which will generate interference between readers and even affect the work of the RFID system. The interference between readers is divided into two categories.

One is the interference between readers. If the transmit signal of a reader is very strong, it will affect the communication between the other readers and tags. For instance, as is shown in Fig. 2, if the reader R_1 emits a strong signal and it is detected by the reader R_2 which is establishing communication with the tag T, the communication will be destroyed. In this case, we assume that distance d is the minimum distance of interference between the two readers. $d(R_i, R_j)$ is the distance between readers R_i and R_j. Hence, when $d(R_i, R_j) > d$, there will be no interference between readers. In this case, we can ignore the interference.

Another interference is caused by the overlapping work scope of readers. If a tag is in the range of several readers simultaneously, communication between readers and tag is disrupted. As shown in Fig. 3, when tag T is within the power radius of both reader R_1 and reader R_2, R_1 and R_2, respectively, establish communication with the tag T at the same time, and the signals they generate interfere with each other.

Here, the coverage model is a Boolean model. The interference of RFID networks is defined as follows:

$$\text{INT} = \sum_{t=0}^{t=N_t} \text{rat}(t)/N_t \tag{4}$$

$$\text{rat}(t) = \begin{cases} (\text{num}(t) - 1)/N_t & \text{num}(t) > = 1 \\ 0 & \text{num}(t) < 1 \end{cases} \tag{5}$$

where $\text{num}(t)$ is the total number of readers that cover tag t and N_t is the number of tags. $\text{rat}(t)$ represents the interference rate of the tag. When $\text{num}(t) < 1$ and $\text{rat}(t)$ is zero, no reader can detect tag t and there is no interference.

At the same time, as establishing communication in the RFID network, the quality of communication is also crucial. To ensure the transmission of information, ideally,
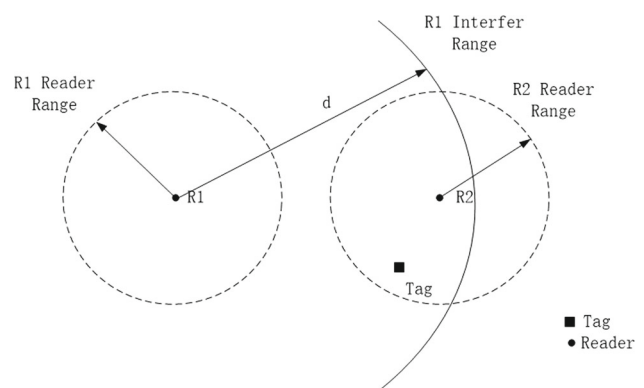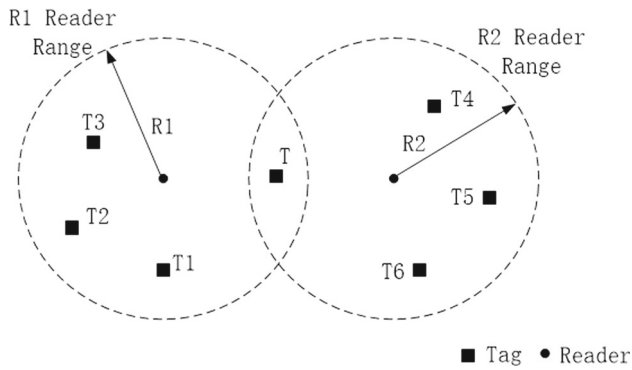

Fig. 2 The interference between readers

Fig. 3 Interference between readers when working areas overlap

each tag is covered by only one reader. When tags are fully covered, we give priority to interference.

2.2.3 Power

In the RFID model, we can measure the power of the system by counting the total radius of readers. In order to save energy and reduce cost, it should be reduced as much as possible. To ensure the communication quality between readers and tags, the power is the objective with the lower priority in RFID network planning. The total power of all readers in the RFID network is defined as follows:

$$POW = \sum_{r \in RS} Radius_r \tag{6}$$

where $Radius_r$ is the power radius of reader r.

Without loss of generality, the number of readers is proportional to the total power. Therefore, this performance is measured by the number of readers.

2.2.4 Load balance

In a large-scale RFID system, a homogeneous distribution of readers is also necessary and important (Dong et al. 2008), which means more stable and efficient (Dong et al. 2008). In a real scenario, the power that the reader provides to tags is not infinite. If there are too many tags in the coverage of a reader, it is overworked and the communication quality is poor. On the contrary, if the number of tags covered by the reader is below the average, it causes the waste of resources and increases the cost of network planning. Therefore, building such an RFID system with a balanced system is something to look forward to. For an RFID system, load balancing is the objective with the lowest priority, which is defined as follows:

$$LB = \prod_{r=1}^{m} 1/N_r \tag{7}$$

where N_r is the number of tags covered by reader r. This objective aims to minimize the variance of load balances

by adjusting the locations and power radius of readers (Dong et al. 2007).

3 The proposed hybrid PSO algorithm for RNP

3.1 Original PSO algorithm

Particle swarm optimization algorithm, which was first proposed by Kennedy and Eberhart in 1995 (Eberhart and Kennedy 1995), employs a swarm of particles to simulate individual birds. It initializes randomly and iterates constantly to search for the optimal solution. Each particle i has its own memory of the best historical individual pBest and the best individual gBest across the whole population, which are updated in the evolutionary process.

The position and velocity of particle i are represented as:

$$X_i = (X_i^1, X_i^2, X_i^3, \ldots, X_i^D) \tag{8}$$

$$V_i = (V_i^1, V_i^2, V_i^3, \ldots, V_i^D) \tag{9}$$

And a previous best position vector pBest and the best individual gBest are represented as:

$$pBest_i = (pBest_i^1, pBest_i^2, \ldots, pBest_i^D) \tag{10}$$

$$gBest = (gBest^1, gBest^2, \ldots, gBest^D) \tag{11}$$

The velocity and location of particles are updated according to Eqs. (12) and (13).

$$V_i^d = \omega * V_i^d + c_1 * rand_1^d * (pBest_i^d - X_i^d) + c_2 * rand_2^d \\ * (gBest^d - X_i^d) \tag{12}$$

$$X_i^d = X_i^d + V_i^d \tag{13}$$

where $d = 1,2,\ldots D$ represents each dimension; $i = 1,2,\ldots M$ is the index of each particle; w is called inertia weight; c_1 and c_2 are coefficients; $rand_1^d$ and $rand_2^d$ are random numbers distributed in [0, 1].

The process of original PSO process is described as follows:

1. The position and velocity of the particles are initialized randomly.
2. For each particle i, evaluate the fitness according to the objective functions.
3. Update pBest and gBest according to the fitness of particles.
4. Update particles' velocities and positions according to Eqs. (12) and (13).
5. Loop to Step (2) until a stopping criterion.

3.2 HPSO-RNP algorithm

In this paper, we propose a hybrid PSO algorithm to solve the RNP problem. It can be seen that the *K*-means algorithm and virtual force are embedded in the algorithm during the general procedure of PSO. Detailed implementations of the algorithm are described in this section.

3.2.1 Particles represent

For RFID network planning, a particle represents a solution to the problem, which includes the number, location and power radius of readers, and is defined as Eq. (14).

$$F_i = \{x_{i1}, x_{i2}, \ldots, x_{ik}, y_{i1}, y_{i2}, \ldots, y_{ik}, rd_{i1}, rd_{i2}, \ldots rd_{ik}\} \tag{14}$$

where $i = 1, 2, \ldots, pop$. pop is the population size. Moreover, (x_{id}, y_{id}) is the coordinate of the dth reader. rd_{id} is the power radius of the dth reader. k represents the number of cluster centers in the RFID system. n represents the number of tags in the network.

3.2.2 Initialization of individual particles

In HPSO-RNP, the locations of tags are randomly distributed in the working space. We use the K-means algorithm for initialization. Each reader can be regarded as a central point of the cluster and tags are considered as classification points. For a start, the coordinates of readers are generated randomly within the scope. In order to ensure the quality of communication, it is necessary to calculate the Euclidean distance between tags and readers. For each tag, it is divided into the nearest cluster. According to the distributed situation of tags, the coordinates of the cluster center point are recalculated according to Eqs. (15) and (16). When the center of cluster changes, we repeat the above process for reclassification until it no longer changes. The process of initialization is described in Algorithm 1.

$$x_j = \frac{1}{Nt_j} \sum_{i=1}^{Nt_j} x_{ij} \tag{15}$$

$$y_j = \frac{1}{Nt_j} \sum_{i=1}^{Nt_j} y_{ij} \tag{16}$$

where (x_j, y_j) represents the coordinates of jth cluster center, Nt_j represents the number of tags in the jth cluster, and (x_{ij}, y_{ij}) represents the coordinates of the ith point in the jth cluster. We represent the center of cluster as r_j, $j \in 1 \ldots k$. A set of tags divided into the cluster r_j is represented as TS_j.

Algorithm1: Initialization individual (K-means)

Input:

 k : number of readers in the RFID network;

 t_i : position of tags, where $t_i \in TS$;

 r_j : random initial position of k readers, where $r_j \in RS$;

 r_{ori_j} : location information of k readers, where $r_{ori\_j} \in RS_{ori}$;

Output:

 RS^k : deployment location of k readers.

$mark \leftarrow 1;$

while *(mark = 1)* **do**

 for each tag $t_i \in TS$ **do**

  for each reader $r_j \in RS^k$ **do**

   Calculate distance $d_j(t_i, r_j);$

  end for;

  Obtain the closest cluster min_j, which is $min(d_j(t_i, r_j), r_j \in RS);$

  Divide the tag t_i into cluster $RS_{min\_j};$

 end for;

 Update the location of readers:

 for each cluster TS_j **do**

  $r_{o\_r\_ij} \leftarrow r_j;$

  Calculate the mean of all points in cluster TS_j and take it as center;

  $r_j \leftarrow (x_j, y_j);$

 end for

 for each cluster TS_j;

  if $(r_{ori\_j} == r_j)$ **then**

   mark←0;

   break;

  end if;

 end for;

end while.

3.2.3 Selection mechanism for the number of readers

In this paper, we define the minimum number of readers required in the RFID network as the initial number. To ensure full coverage, we can obtain radius rd_j using Eqs. (17) and (18), which calculates the farthest distance $d(t_{ij}, r_j)$ between the points of cluster t_{ij} and the center point r_j. If the radius is within the range of requirements, the number of readers is reasonable and the next step can be carried out. Otherwise, it is necessary to change the number of readers in the network. We update the number of readers as shown in Eq. (19), and perform the K-means operator again to search the cluster points. We repeat the above steps until the radius of readers lies within the required range. At this point, the number of readers in the network can be considered as reasonable. The selection mechanism for the number of readers is described in Algorithm 2.

$$radius_j = \max(d(t_{ij}, r_j)) \quad t_{ij} \in TS_j \tag{17}$$

$$d(t_{ij}, r_j) = \sqrt{(x_{t_{ij}} - x_{r_j})^2 + (y_{t_{ij}} - y_{r_j})^2} \tag{18}$$

$$k' = k + 1 \tag{19}$$

3.2.4 Updating individual particles

In PSO, a particle moves to a new solution according to its own position, the best historical individual *pBest*, and the best individual *gBest* across the whole population. We recalculate the location of readers by using Eqs. (12) and (13). After each iteration, we calculate the fitness function for each particle. As described in Sect. 2, the fitness function of RNP includes maximizing tag coverage, minimizing interference between readers, minimizing total power, and minimizing load balancing. Individuals *pBest* and *gBest* are updated in a hierarchical manner according to the four objective functions. The update process of particles is described in Algorithm 3.

Algorithm2: Selection mechanism for the number of readers

Input:

 k : number of readers in the network;

 r_j : location information determined by the KM operator;

 flag: flag which determines whether to adjust the number of readers;

Output:

 k': number of readers in the network when the radius satisfies hypothesis;

 r_j': location of readers when the number of readers is k';

flag ← 1;

for each reader $r_j \in RS^k$ **do**

 for each tag $t_{ij} \in TS_j$ **do**

 Calculate distance $d_i(t_{ij}, r_j)$ $t_{ij} \in TS_j$

 end for;

 Obtain the maximum distance as the radius of the reader, $d_{\max\_t}(t_{ij}, r_j)$;

 $rd_j \leftarrow d_{\max}(t_{ij}, r_j)$

 If rd_j is beyond the range:

 k++;

 break;

 else;

 flag=0;

 end if;

end for.

Algorithm3: Updating individual particles

Input:

    pop: initial population;

Output:

    pBest: best historical individual;

    gBest: best individual across the whole population;

Updating individual: recalculate the locations of readers using Eqs. (12) and (13);

Evaluation: evaluate the four objective function values according to Eqs. (2)-(7);

if (*COV* larger) *then*

    pBest ← the current individual;

else if (*COV* smaller) *then*

    break;

end if;

if (*Inter* smaller) *then*

    pBest ← the current individual;

else if (*Inter* larger) *then*

    break;

end if;

if (*Power* smaller) *then*

    pBest ← the current individual;

else if (*Power* larger) *then*

    break;

end if;

if (*Load-balancing* smaller) *then*

    pBest ← the current individual;

else if (*Load-balancing* larger) *then*

    break;

end if;

gBest ← the best *pBest*;

The process of PSO-RNP is described as follows:

1. The position, velocity, and the number of particles are initialized using the k-means algorithm.
2. For each particle i, evaluate the fitness function according to Sect. 2.
3. Update pBest and gBest in a hierarchical manner.
4. Update particles' velocities and positions according to Eqs. (12) and (13).
5. Loop to Step (2) until a stopping criterion.

3.2.5 Adjustment of readers location by virtual force

During the process of PSO optimization, there are many tags that are overwritten by multiple readers at the same time. In other words, if a tag is assigned to the nearest cluster A but is within the coverage of another cluster B, there is interference that affects the communication between tags and readers. Then the virtual force is performed. It is assumed that there are negative forces between readers that interfere with each other. These forces are conducted to adjust the location of readers for avoiding interference. However, some tags are not overwritten on account of readers moving. Therefore, they have the positive force on the reader closest to them, which is used to improve the coverage of tags in the network. Two processes, named exclusion and attraction operator, are designed, respectively.

In the exclusion operator, if the tag is covered by multiple readers, we use the reader closest to the tag as a conference point and move the rest readers to avoid interference. As shown in Fig. 4, reader A is closer to reader B and tag C is right on the overlapping region of them. Therefore, the communication between the tag and readers is disturbed by each other. To improve the quality of communication, we should adjust the position of readers. Obviously, the tag is closer to reader A than to reader B. The conference point is reader A, and reader B is
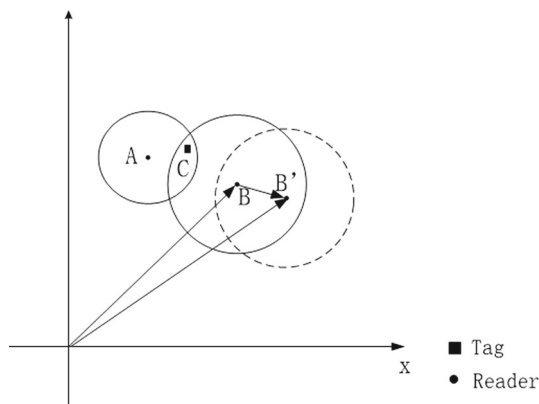
Fig. 4 The schematic diagram of exclusion operator

repelled by negative forces and moves to the location of B'
according to Eqs. (20) and (21), where a is a random
number between 0 and 1, representing the degree of
rejection between the readers. (x_A, y_A), (x_B, y_B) represent
the coordinates of readers A and B, respectively. (x_B', y_B')
represents the coordinates when reader B has been moved
by the exclusion operator. The above reduces the inter-
ference between readers.

$$x_{B'} = x_B + \alpha \times (x_B - x_A) \tag{20}$$

$$y_{B'} = y_B + \alpha \times (y_B - y_A) \tag{21}$$

If there are tags in the network that is not overwritten by
readers, the reader closest to the tag moves closer to it
under a positive force. The attraction operator is shown in
Fig. 5. Obviously, tag C is not covered in the network. And
reader A is the closest reader to the tag. So reader A is
attracted by the tag and moves to A' according to Eqs. (22)
and (23), where β is a random number between 0 and 1,
representing the attraction of a tag to the closest reader.
The same as the exclusion operator, $\left(x_B', y_B'\right)$ is the coor-
dinates of the position after moving when it is attracted.
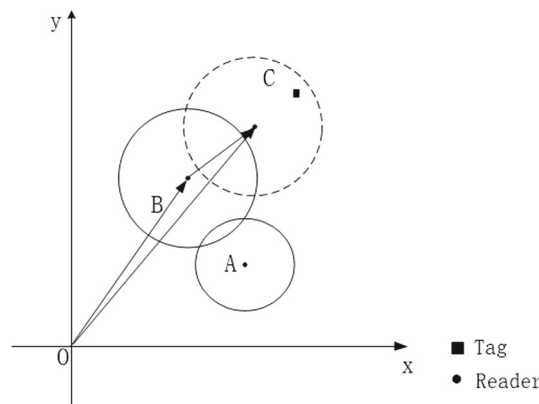
Fig. 5 The schematic diagram of attraction operator

$$x_{B'} = x_B + \beta \times (x_A - x_B) \tag{22}$$

$$y_{B'} = y_B + \beta \times (y_A - y_B) \tag{23}$$

3.2.6 Implementation of HPSO for RNP

The flowchart of the HPSO-RNP algorithm is shown in
Fig. 6. Firstly, the initial number of readers is set to 3 in the
network. Under selection mechanism for the number of
readers, it is constantly adjusted until the radius of readers
is within the range. Then the individuals are initialized
using the K-means algorithm. Moreover, we introduce the
virtual force during the optimization process of PSO to
adjust the location of readers. The evaluation measures of
particles mainly include coverage, interference, total
power, and load balancing. The individual is updated with
a hierarchical mechanism based on the priority of the
objective function. In large-scale label coverage problems,
Euclidean distance measurement is somewhat inaccurate
on account of the different radius of readers.

The implementation of HPSO-RNP is described in
Algorithm 4.

Algorithm4: HPSO-RNP

Input:

N_t : number of tags in the network;

gen : number of times for operating HPSO-RNP operator;

Output:

k' : number of readers in the network;

r_j'' : locations of readers in the network, $r_j'' \in RS$;

Initialization: initialize the locations of readers r_j with K-means;

Selection Mechanism: calculate the number of readers in the network;

$(k', r_j') \leftarrow$ **Selection Mechanism**(k, r_j);

$t \leftarrow 0$;

while $(t < gen)$ *do*

    Evaluation: evaluate the performance of RFID system in a hierarchical manner according to Eqs. (2)-(7);

    Updates: update the best individual using PSO;

    $pBest \leftarrow$ **Updates**(pop);

    $gBest \leftarrow$ the best $pBest$;

    Adjustments: adjust by virtual force during the process of PSO optimization;

    $(r_j'') \leftarrow$ **Adjustments** $(gBest)$;

    $t \leftarrow t+1$;

end while.
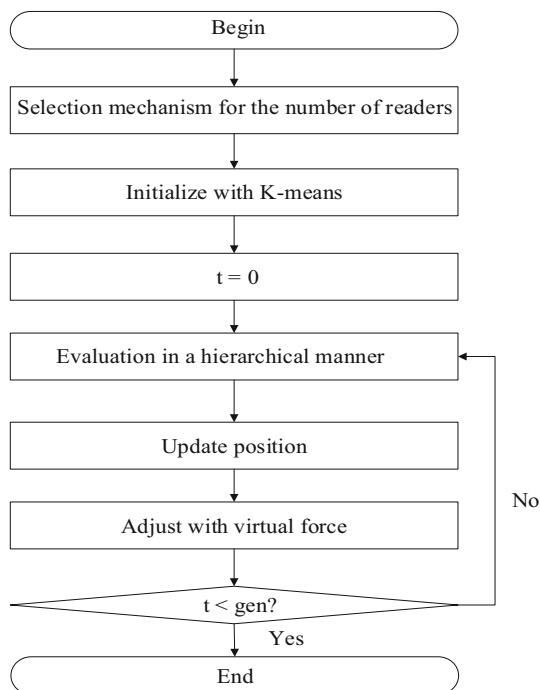
Fig.6 Flowchart of the proposed HPSO-RNP

4 Experimental results and analysis

4.1 Configuration of the scenario

In the experiments, we employ eight RNP instances to evaluate the performance of the HPSO-RNP algorithm fully. C_30, C_50, and C_100 are samples with clustered distributed tags. R_30, R_50, and R_100 are samples with uniformly distributed tags. C_30 and R_30 contains 30 tags, C_50 and R_50 contains 50 tags, and C_100 and R_100 contains 100 tags. The following tags are generated randomly in a $50 \, \mathrm{m} \times 50 \, \mathrm{m}$ working space. C_500 and R_500 are generated based on C_100 and R_100. These two instances are tested on a $150 \, \mathrm{m} \times 150 \, \mathrm{m}$ working space.

For the HPSO-RNP, the size of a population is set to 10 and *gen* is set to 1000. The initial number of readers k in the RFID network is set to 3. When the number of tags distributed randomly increases, the initial value k also increases to reduce the computational complexity, which $k \leqq k_{\min}$ and $k_{\min}$ represents the minimum value for deploying readers in the network. We assume the radius of readers is randomly initialized from 8 to 15m.

4.2 Experimental results of HPSO-RNP

In order to evaluate the effectiveness of our proposed approach, we conduct a comparative experiment with two existing algorithms including the CA-RNP (Zhang and Liu

2017) and MOEAD-RNP (Liu et al. 2016; Zhao et al. 2017; Shinde et al. 2019). Moreover, we make a conceptual comparison of the proposed RNP method with the decomposition-based multi-objective firefly algorithm.

Zhao et al. proposed a method based on decomposition multi-objective firefly algorithm, which can be applied to RFID network planning. The most striking difference between our proposed method and HPSO-RNP is that the former updates fitness function according to the Pareto of multiple objectives and the radius of readers is fixed. Moreover, it needs a predefined number of readers at the beginning of the algorithm. The number and radius of readers in the network can be determined by our algorithm. Meanwhile, we consider four objective functions in a hierarchical manner. Therefore, our algorithm is more efficient.

The solution of HPSO-RNP on these eight instances is shown in Figs. 7, 8, 9 and 10.

Figure 7a is the solution when the number of tags evenly distributed in the network is 30. Obviously, tags are covered completely and the interference between readers is 0. It indicates that communication between readers and tags can be established well. As shown in Fig. 7a_1, the number of readers that should be placed into the network is 5 and the load balance is 0.950. In the similar way in Fig. 7a_2, the number of readers is 4, and the load balancing is 0.5611.

Figure 7b is the solution when the number of tags in the network is 50. All tags in the network are completely covered. And there is no interference between readers. As the deployment of readers in Fig. 7b_1 shows, there are 6 readers in the network to maintain communication. The load balancing is 0.7968 and the total power is 68.7989. In Fig. 7b_2, the number of readers in the network is 5, the load balancing is 0.5543 and the total power is 57.4834. We can see, the more readers in the network, the greater the total power.

As the number of tags in the network increasing, the problem of network planning becomes more and more complicated. Figure 7c represents the results that the number of tags in the network is 100. As shown in Fig. 7c_1, the number of readers deployed in the network is 7, the interference between readers is 0, and the power is 88.837. The load balancing is 0.6025. In Fig. 7c_2, the number of readers deployed in the network is 6. The interference is 0, the total power is 78.5317, and load balancing is 0.3774.

Figure 7d represents the results when the number of tags in the network is 500. All tags are still covered and the interference between readers is low. The result is similar to that of the network with 100 tags. It shows that the algorithm HPSO-RNP can still get better results in more complex instances.
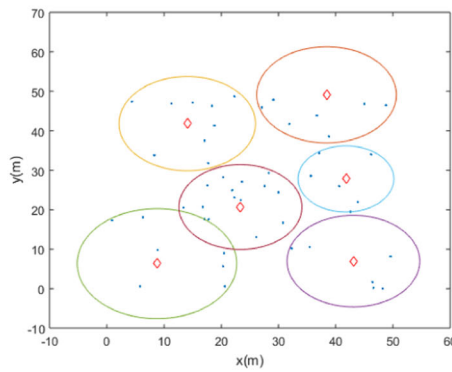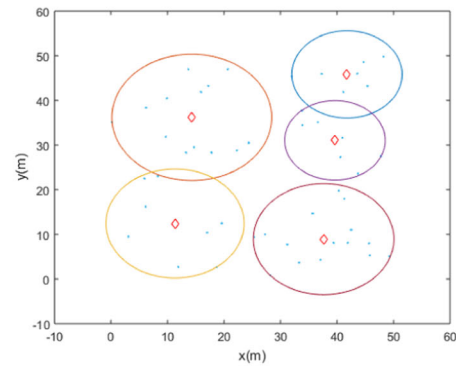
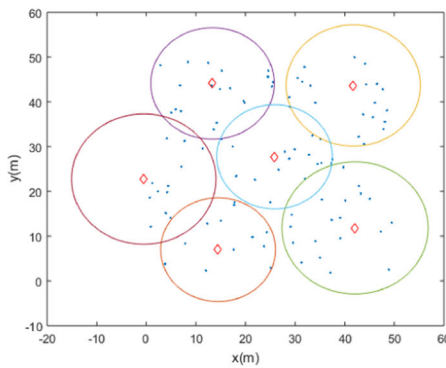**Fig.7** Results of HPSO-RNP on eight RNP instances
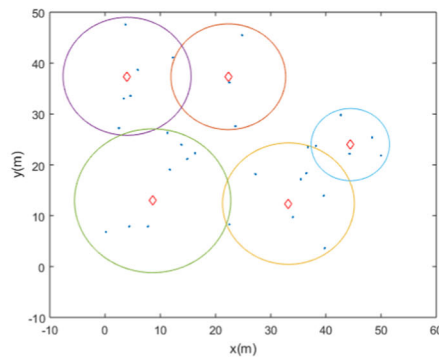
(a_1) R_30
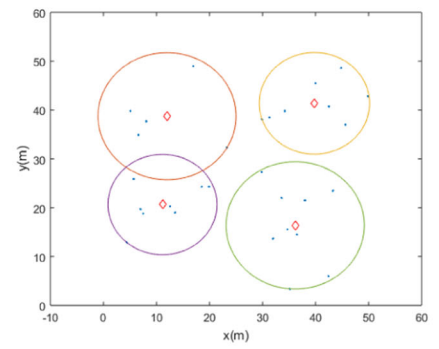
(a_2) R_30

(b_1) R_50

(b_2) R_50

(c_1) R_100

(c_2) R_100
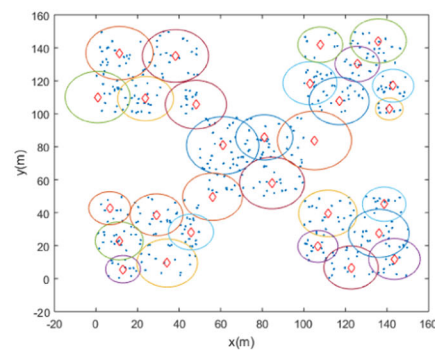
(d_1) R_500

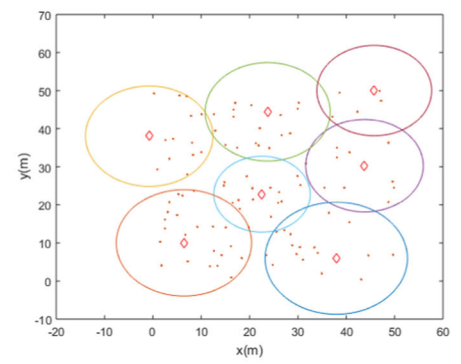**(d_2)** R_500

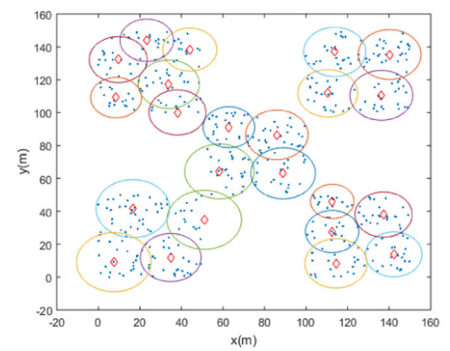Fig. 8 Results of HPSO-RNP
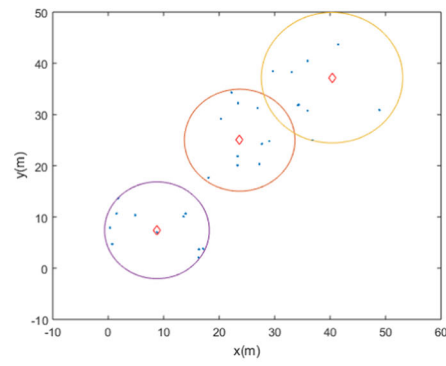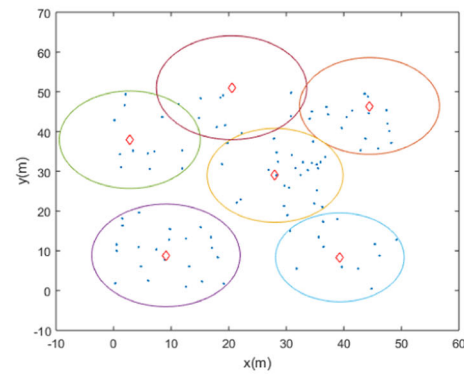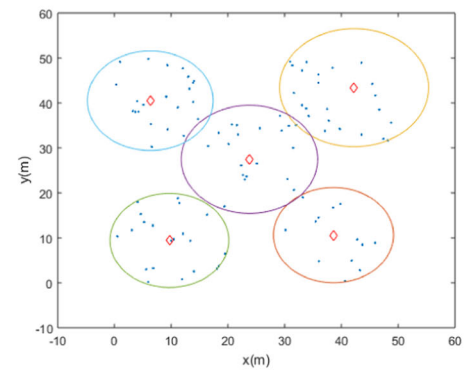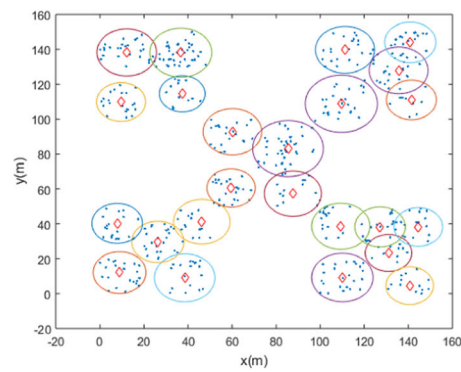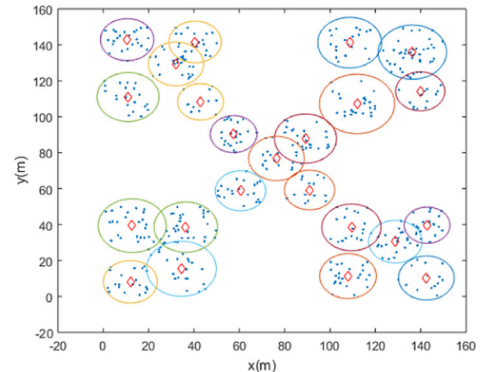on clustered instances

(a_1) C_30
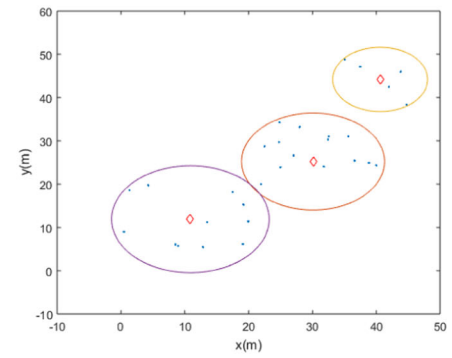
(a_2) C_30
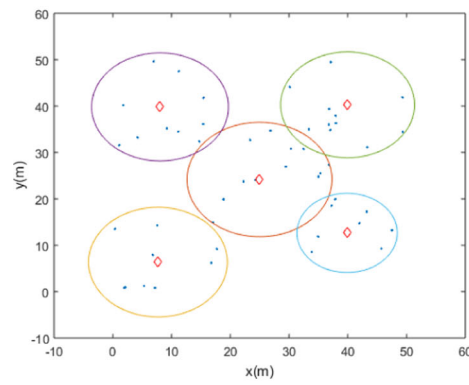
(b_1) C_50
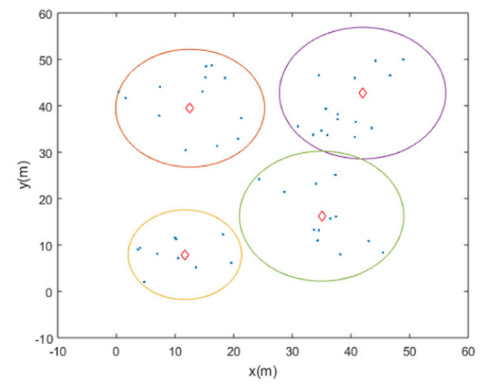
(b_2) C_50

(c_1) C_100

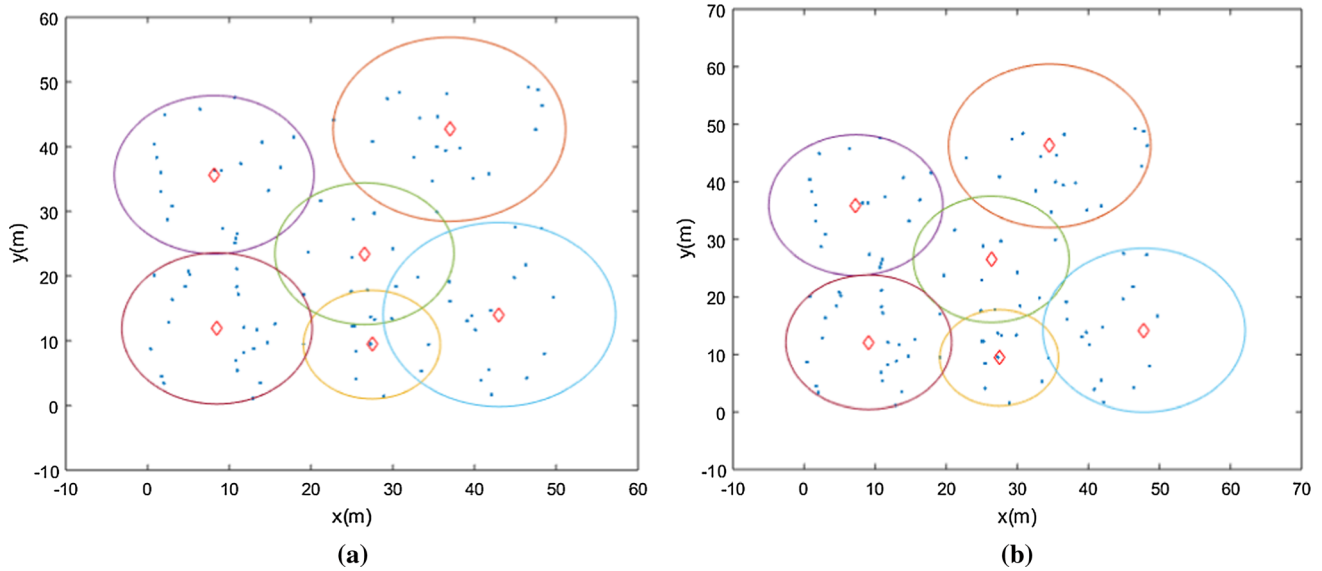(c_2) C_100

(d_1) C_500

(d_2) C_500

Fig.9 a The solution before adding virtual force, **b** the solution adding exclusive and attraction operator
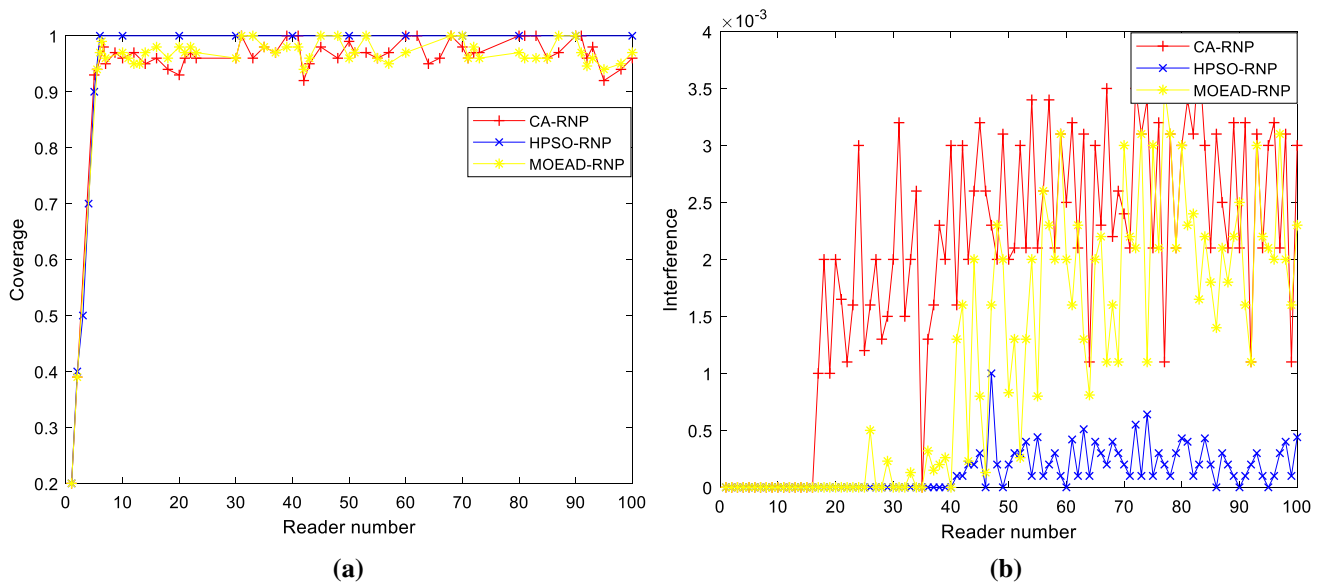

Fig.10 Comparison results of system indexes between different algorithms **a** coverage and **b** interference

Similarly, Fig. 8 is the solution when the tags in the network are clustered. The results show that our algorithm has greater advantages over the previous. The coverage, anti-interference degree, total power, and load balancing have been greatly improved.

As the number of tags increases, the algorithm PSO for RNP still has some disadvantages. For example, the convergence speed of PSO is too slow, which is easy to fall into local optima and the communication quality cannot be guaranteed. Therefore, we introduce virtual force during the optimization process of PSO.

In order to illustrate the effectiveness of virtual force, we conduct a comparative experiment. Figure 9a is the solution before adding virtual force. Figure 9b is the solution of our algorithm. The coverage remains 1, and interference is reduced from 0.0011 to 0.0003. The result shows that the virtual force is effective for PSO optimization, which can avoid interference without affecting coverage.

According to the above experimental results, we can get the tables as follows. The performance of CA-RNP, MOEAD-RNP, MOFA-RNP and HPSO-RNP is compared in Table 1. It is easy to see that our algorithm has better performance in four metrics when the number of tags in the network is different.

Table 1 The comparison in terms of system indexes among different algorithms and instances

Algorithm	Mean				Best			
	Count	COV (%)	Inter	Balance	Count	COV (%)	Inter	Balance
Results for benchmark C30								
CA-RNP	3	95.99	0.008	0.3044	3	100	0	0.3068
MOEAD-RNP	3	99.82	0.002	0.3126	3	100	0	0.3197
HPSO-RNP	3	100	0	0.3353	3	100	0	0.3083
Results for benchmark C50								
CA-RNP	5	94	0.0036	0.5386	5	96	0.0008	0.5305
MOEAD-RNP	5	98	0.0017	0.5614	5	98	0	0.5467
HPSO-RNP	4.75	100	0	0.4186	4	100	0	0.3269
Results for benchmark C100								
CA-RNP	5	95	0.0017	0.2790	5	98	0	0.2327
MOEAD-RNP	5.8	98	0.0018	0.3176	5	99	0.0011	0.2580
MOFA-RNP	7	95.1	0.0056	0.3082	5	96	0.0024	0.2473
HPSO-RNP	5.6	100	0	0.3345	5	100	0	0.2590
Results for benchmark R30								
CA-RNP	5.5	93.33	0.00165	1.1549	5	93.33	0.0011	0.9909
MOEAD-RNP	5.5	94.99	0.00055	1.1240	5	96.66	0	0.7010
HPSO-RNP	5	100	0	0.7166	4	100	0	0.5611
Results for benchmark R50								
CA-RNP	6.66	98	0.0032	1.1132	6	98	0.0020	0.9075
MOEAD-RNP	6.33	98	0.0017	1.0400	6	98	0.0008	0.6960
HPSO-RNP	5.66	100	0	0.6513	5	100	0	0.5543
Results for benchmark R100								
CA-RNP	8.66	99	0.0027	0.8544	8	99	0.0022	0.8079
MOEAD-RNP	7	97.33	0.0014	0.4054	6	99	0.0011	0.2710
MOFA-RNP	9	95.1	0.0057	0.4024	8	97.33	0.0011	0.3425
HPSO-RNP	7	100	0	0.4833	6	100	0	0.3774

Figure 10a, b illustrates two main system indexes (coverage and interference) change when the number of readers in the working area changes by comparing these three algorithms. As we can see from Fig. 10, when there are no enough readers in the network to cover the tags, the coverage obtained by CA-RNP and MOEAD-RNP is the same. However, the coverage value of our algorithm can reach 1 in a shorter time with the increase in the number of readers. It is difficult to achieve full coverage when the number of tags is large for other algorithms.

We can see that HPSO-RNP has a good performance in these eight instances from Figs. 7, 8, 9 and 10. In the case with the same distribution of tags, tags are completely covered and the interference between readers is low. The total power and load balancing have been improved greatly. Moreover, it is more efficient than previous studies with more tags, such as the CA-RNP and MOEAD-RNP.

5 Conclusions

In this paper, a hybrid PSO algorithm is proposed to the problem of RFID network planning, which does not need to artificially estimate the number of readers deployed in the network. Instead, we initialize the number of readers in the network through K-means algorithm and optimize it automatically. Moreover, virtual force is introduced during the optimization process of PSO. And we take four objectives into account simultaneously and handle them in a hierarchical manner.

In the experiment, the performance of our algorithm is compared with that of multi-objective evolutionary algorithm based on decomposition (MOEAD-RNP) and curling algorithm based on kinematics (CA-RNP). The experimental results show that our proposed method has improvement in terms of number of readers, interference, total power, and load balance. In our future work, we plan to apply the hybrid PSO algorithm to the RFID network planning of probability coverage model. Furthermore, we

can develop dynamic algorithm to solve dynamic RNP problems with readers in motion.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

Antonis G, Stavroula S, Aggelos B, John S (2019) Introduction of dynamic virtual force vector in particle swarm optimization for automated deployment of RFID networks. In: 13th European conference on antennas and propagation (EuCAP 2019)

Bhattacharya I, Roy U (2010) Optimal placement of readers in an RFID network using particle swarm optimization. Int J Comput Netw Commun (IJCNC) 2(6):225–234

Chen H, Zhu Y (2008) RFID network planning using evolutionary algorithms and swarm intelligence. In: Proceedings of fourth international conference on wireless communications, networking and mobile computing. Dalian, China, pp 1–4.

Chen Y-H, Horng S-J, Run R-S, Lai J-L, Chen R-J, Chen W-C, Pan Y, Takao T (2010a) A novel anti-collision algorithm in RFID systems for identifying passive tags. IEEE Trans Industr Inf 6(1):105–121

Chen H, Zhu Y, Hu K (2010b) Multi-colony bacteria foraging optimization with cell-to-cell communication for RFID network planning. Appl Soft Comput 10(2):539–547

Chen H, Zhu Y, Hu K, Ku T (2011) RFID network planning using a multi-swarm optimizer. J Netw Comput Appl 34(3):888–901

Dong Q, Shukla A, Shrivastava V, Agrawal D, Banerjee S, Kar K (2007) Load balancing in large-scale RFID systems. In: Infocom IEEE international conference on computer communications, pp 2281–2285.

Dong Q, Shukla A, Shrivastava V, Agrawal D, Banerjee S, Kar K (2008) Load balancing in large-scale RFID systems. Comput Netw 52(9):1782–1796

Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. In: MHS'95. Proceedings of the sixth international symposium on micro machine and human science, Nagoya, Japan, pp 39–43.

Gong Y, Shen M, Zhang J, Kaynak O, Chen W, Zhan Z (2012) Optimizing RFID network planning by using a particle swarm optimization algorithm with redundant reader elimination. IEEE Trans Industr Inf 8(4):900–912

Gong M, Cai Q, Chen X, Ma L (2014) Complex network clustering by multi-objective discrete particle swarm optimization based on decomposition. IEEE Trans Evol Comput 18(1):82–97

Guan Q, Liu Y, Yang Y, Yu W (2006) Genetic approach for network planning in the RFID systems. In: Sixth international conference on intelligent systems design and applications. Jinan, China, pp 567–572

Harrington P (2013) Machine learning in action. Posts&Telecom Press, Beijing

Huang H, Chang Y (2011) Optimal layout and deployment for RFID systems. Adv Eng Inform 25(1):4–10

Krohn A, Zimmer T, Beigl M, Decker C (2005) Collaborative sensing in a retail store using synchronous distributed jam signaling. In: Proceedings of the 3rd international conference on pervasive computing. Munich, pp 237–254.

Liu J, Wu C, Cao J, Wang X, Teo K (2016) A binary differential search algorithm for the multidimensional knapsack problem. Appl Math Model 10(23–24):9788–9805

Lu S, Yu S (2014) A fuzzy k-coverage approach for RFID network planning using plant growth simulation algorithm. J Netw Comput Appl 39(1):280–291

Ma L, Hu K, Zhu Y, Chen H (2014) Cooperative artificial bee colony algorithm for multi-objective RFID network planning. J Netw Comput Appl 42:143–162

Özdemir S, Attea B, Khalil Önder A (2013) Multi-objective evolutionary algorithm based on decomposition for energy efficient coverage in wireless sensor networks. Wirel Pers Commun 71(1):195–215

Seok J-H, Lee J-Y, Oh C, Lee J-J, Lee HJ (2010) RFID sensor deployment using differential evolution for indoor mobile robot localization. In: The 2010 IEEE/RSJ international conference on intelligent robots and systems. Taipei, Taiwan, pp 3719–3724

Shinde S, Devika K, Thangavelu S, Jeyakumar G (2019) Multi-objective evolutionary algorithm based approach for solving Rfid reader placement problem using weight-vector approach with opposition-based learning method. Int J Recent Technol Eng (IJRTE) 7(5):2277–3878

Tuba V, Alihodzic A, Tuba M (2017) Multi-objective RFID network planning with probabilistic coverage model by guided fireworks algorithm. In: 2017 10th international symposium on advanced topics in electrical engineering (ATEE), IEEE. Bucharest, Romania, pp 882–887

Wan D (1999) Magic medicine cabinet: a situated portal for consumer healthcare. In: Proceedings of the international symposium on handheld and ubiquitous computing. Germany, pp 352–355.

Xu Y, Ding O, Qu R, Li K (2018) Hybrid multi-objective evolutionary algorithms based on decomposition for wireless sensor network coverage optimization. Appl Soft Comput 68:268–282

Yang Y, Wu Y, Xia M, Qin Z (2009) A RFID network planning method based on genetic algorithm. In: 2009 international conference on networks security, wireless communications and trusted computing. Wuhan, Hubei, China, pp 534–537

Zhang Q, Li H (2007) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans Evol Comput 11(6):712–731

Zhang T, Liu J (2017) An efficient and fast kinematics-based algorithm for RFID network planning. Comput Netw 121:13–24

Zhang J, Tang Q, Li P, Deng D, Chen Y (2016) A modified MOEA/D approach to the solution of multi-objective optimal power flow problem. Appl Soft Comput 47:494–514

Zhao C, Wu C, Chai J, Wang X, Yang X, Lee J, Kim M (2017) Decomposition-based multi-objective firefly algorithm for RFID network planning with uncertainty. Appl Soft Comput 55:549–564