ORIGINAL ARTICLE

Biomimicry of symbiotic multi-species coevolution for discrete and continuous optimization in RFID networks

Na Lin a,b, Hanning Chen a,*, Shikai Jing c, Fang Liu b, Xiaodan Liang b

a School of Computer Science and Software, Tianjin Polytechnic University, Tianjin 300387, China
b Beijing Shenzhou Aerospace Software Technology Co. Ltd., Beijing 110000, China
c School of Mechanical Engineering, Beijing Institute of Technology, Beijing 110081, China

Received 11 October 2016; revised 25 December 2016; accepted 8 January 2017
Available online 25 January 2017

KEYWORDS
Symbiosis; Particle swarm optimization; Multi-swarm coevolution; Global optimization; RFID

Abstract In recent years, symbiosis as a rich source of potential engineering applications and computational model has attracted more and more attentions in the adaptive complex systems and evolution computing domains. Inspired by different symbiotic coevolution forms in nature, this paper proposed a series of multi-swarm particle swarm optimizers called PS2Os, which extend the single population particle swarm optimization (PSO) algorithm to interacting multi-swarms model by constructing hierarchical interaction topologies and enhanced dynamical update equations. According to different symbiotic interrelationships, four versions of PS2O are initiated to mimic mutualism, commensalism, predation, and competition mechanism, respectively. In the experiments, with five benchmark problems, the proposed algorithms are proved to have considerable potential for solving complex optimization problems. The coevolutionary dynamics of symbiotic species in each PS2O version are also studied respectively to demonstrate the heterogeneity of different symbiotic interrelationships that effect on the algorithm's performance. Then PS2O is used for solving the radio frequency identification (RFID) network planning (RNP) problem with a mixture of discrete and continuous variables. Simulation results show that the proposed algorithm outperforms the reference algorithms for planning RFID networks, in terms of optimization accuracy and computation robustness.

1. Introduction
Natural selection tends to enable living organisms to adapt to their environments by the processes of interaction (Bonabeau et al., 1999). As in nature, individuals interact constantly. Within a species or a population, individuals can gain useful information to improve their adaptation by interacting with
Biomimicry of symbiotic multi-species coevolution

In this paper, we implement an entire symbiotic system which consists of both heterogeneous and homogeneous coevolution aspects in formulating our symbiotic simulation models (Chen and Zhu, 2008). We introduced several symbiotic species each possessing a number of individuals into this coevolution model to represent the “biological community”. The coevolution process in our model is hierarchical and contains three levels (i.e. individual level, species level and community level). Each individual of the biological community evolves based on the knowledge of integration of itself (individual-level evolution, associate with individual’s own cognition), its species member (species-level coevolution, associate cooperative interaction within species) and its symbiotic partners from other species (community-level coevolution, associate heterogeneous coevolution between individual from different species). Clearly we model more details of the social behaviors in nature ecosystems and tie this model closer to natural evolution. Since the community is made up of a swarm of agents who are species while each species is made up of a swarm of species member (individual), our swarms-within-swarms model is instantiated as a symbiotic coevolutionary optimization algorithm, namely particle swarms swarm optimizer (PS2O).

Inspired by different symbiotic coevolution forms in nature, this work proposed four versions of PS2O to mimic mutualism, commensalism, parasitism, and competition mechanism, respectively. In order to evaluate the performance of the PS2O algorithms, extensive studies based on five widely used benchmark functions have been carried out. The simulation results are encouraging: all the PS2O algorithms have markedly superior search performance in terms of accuracy, robustness and convergence speed on all benchmark functions. We also simulated the coevolution process of a number of distinct species in our biological community model, and the simulation result is consistent with the natural phenomenon (i.e., the biological diversity is sustained after many coevolutionary generations, and the adaptive abilities of the interacting individuals/species are greatly increased.).

Radio Frequency Identification (RFID) technology as a new inventory tracking technology has achieved significant development in many practical industrial scenarios, where much great application potential has been realized and many are being explored. In many real-world RFID applications, such as production, logistics and supply chain management, more and more readers are deployed in order to provide complete coverage of all the tags in the given area. Hence, in this paper, we develop an optimization model for distributed decision making of reader management in large-scale RFID network based on the evolutionary and swarm-based methods. Here the RNP problem is described and formulated as a two-level model, which is in order to minimize the total installation cost of RFID network in the top-level, while maximizing tag coverage and network reliability, and minimizing reader interference in the lower-level. Then, in order to solve this complex mixed discrete-continuous problem effectively and efficiently, we provide some initial insights into this potential by evaluating PS2O on real-world RNP cases, which focus on distributed decision making of a 30-reader RFID network and a 50-reader RFID network. The simulation results, which are compared to other state-of-the-art methods, show the superiority of the proposed algorithm.

2. Symbiotic coevolution

Symbiosis, initially defined by Anton de Bary in 1879, is simply the living together of organisms from different species (Frank, 1995). Here, we denote symbiosis as relationships that are constant and intimate between dissimilar species. Symbiosis is almost ubiquitous in nature. There are practically no plants or animals free of symbionts (organisms in symbiotic relationship) living on or in them. Research shows different types of symbiotic interactions. Some involve internal interactions, like bacteria in human intestines. Some of these interactions have seemed to lead to the evolution of organisms (e.g., the eukaryote cells, form which all plants and animals are descended have symbiotic origin). Others appear to be purely behavioral, as in a human-honeyguide mutualism that is discussed above. Currently, enlightened evolutionary theory recognizes symbiosis as an integral process, and a fundamental source of innovation in evolution (Chen et al., 2010a,b).

It is impossible to describe a set classifications for all types of symbiotic coevolution. Since an arbitrary number of individuals can be involved it would be ludicrous to try. However, the typical approach is to look at how pairs of individuals can form symbiotic relationships. It is possible to describe the various forms of symbiosis that exist in nature in the following listing.

Commensalism: commensalism occurs between two individuals Host and Commensal if and only if Host antagonizes Commensal and Commensal does not affect the fitness of Host (i.e., Commensal benefits from Host and Host is unaffected).

Competition: competition occurs between two individuals Competitor A and B if and only if both Competitors antagonize each other (i.e., both Competitors suffer because of each other).

Mutualism: mutualism occurs between two Symbionts A and B if and only if Symbiont A antagonizes Symbiont B and Symbiont B antagonizes Symbiont A (i.e., both Symbions benefit from and/or need each other).

Predation: predation occurs between two individuals Predator and Prey if and only if Predator antagonizes Prey and Prey antagonizes Predator (i.e., Predator benefits from Prey while Prey suffers due to Predator).

3. Model

In this paper, we mimic four types of symbiotic coevolution, namely mutualism, commensalism, predation, and competition, to design our multi-swarm algorithms.
3.1. Canonical PSO (PSO)

The canonical PSO is a population-based technique, similar in some aspects to evolutionary algorithms (Sumathi et al., 2008), except that potential solutions (particles) move, rather than evolve, through the search space. The rules of particle dynamics that govern this movement are inspired by models of swarming and flocking. In PSO population, each particle has a position and a velocity, and experiences linear spring-like attractions toward two attractors:

- Its previous best position.
- Best position of its neighbors.

In mathematical terms, the ith particle is represented as $x_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$ in the D-dimensional space, where $x_{id} \in [l_d, u_d], d \in [1, D]$, $l_d$, $u_d$ are the lower and upper bounds for the dth dimension, respectively. The rate of velocity for particle $i$ is represented as $v_i = (v_{i1}, v_{i2}, \ldots, v_{iD})$. The velocity changes in probabilities that a bit will be in one state or the other, except that potential solutions (particles) move, rather than evolve, through the search space.

For particle $i$, its previous best position is $p_{id}$. The best position of any particles in its neighborhood, and $\chi$ is called constriction factor (Mendes et al., 2004), give by:

$$\chi = \frac{2}{\sqrt{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}}}$$

where $\varphi = c_1 + c_2$, $\varphi > 4$.

Kennedy and Eberhart proposed a binary PSO (Kennedy and Eberhart, 2001) in which a particle moves in a state space restricted to zero and one on each dimension, in terms of the changes in probabilities that a bit will be in one state or the other. The velocity formula (1) remains unchanged except that $x_{id}$, $p_{id}$ and $p_{gd}$ are integers in [0, 1] and $v_{id}$ must be constrained to the interval [0.0, 1.0]. This can be accomplished by introducing a sigmoid function $S(v)$, and the new particle position is calculated using the following rule:

If $\text{rand} < S(v_{id})$, then $x_{id} = 1$; else $x_{id} = 0$; (3)

where $\text{rand}$ is a random number selected from a uniform distribution in [0.0, 1.0] and the function $S(v)$ is a sigmoid limiting transformation as follows:

$$S(v) = \frac{1}{1 + e^{-v}}$$

3.2. Predation $PS^2O$ ($PS^2O$-P)

Inspired by predation phenomenon, in our $PS^2O$-P algorithm, a population consists of one predator swarm (symbiont S1) and one prey swarm (symbiont S2). In the commensal swarm, each particle adjusts its trajectory according to its own experience, the experience of its swarm members, and the experience of its symbiotic partners in the prey swarm. Hence, the evolution equation of the predation swarm is the same as the commensal swarm in $PS^2O$-C algorithm (Eqs. (6) and (7)). While in the prey swarm, the prey particles move away from those positions that are found by predator swarm and explore new sections of the search space, then at each generation $t$, the prey swarm moves away from those positions that are found by predator swarm and explores the search space.

In our experimental studies, a set of 5 benchmark functions was employed to evaluate the $PS^2O$ algorithms in comparison with others. The following benchmark functions can be grouped as continuous unimodal function Rosenbrock (f1), continuous multimodal functions Rastrigrin (f2) and Griewank (f3), and discrete functions Goldberg’s order-3 (f4) and Bipolar order-6 (f5) (Chen et al., 2009). The fitness landscapes of these five functions are illustrated in Figs. 1–5, respectively. The formulas of these functions are presented below:

4. Benchmark test

In our experimental studies, a set of 5 benchmark functions was employed to evaluate the $PS^2O$ algorithms in comparison with others. The following benchmark functions can be grouped as continuous unimodal function Rosenbrock (f1), continuous multimodal functions Rastrigrin (f2) and Griewank (f3), and discrete functions Goldberg’s order-3 (f4) and Bipolar order-6 (f5) (Chen et al., 2009). The fitness landscapes of these five functions are illustrated in Figs. 1–5, respectively. The formulas of these functions are presented below:

4.1. Rosenbrock function

$$f_1(x) = \sum_{i=1}^{n} 100 \times (x_{i+1} - x_i^2)^2 + (1 - x_i)^2$$

(10)
4.2. Rastrigrin’s function

\[ f_2(x) = \sum_{i=1}^{n}(x_i^2 - 10\cos(2\pi x_i)) + 10 \]  

(11)

4.3. Griewank function

\[ f_3(x) = \frac{1}{4000}\sum_{i=1}^{n}x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \]  

(12)

4.4. Golderg’s order-3

The fitness \( f \) of a bit-string is the sum of the result of separately applying the following function to consecutive groups of three components each:

\[
f_4(x) = \begin{cases} 
0.9 & \text{if } |y| = 0 \\
0.6 & \text{if } |y| = 1 \\
0.3 & \text{if } |y| = 2 \\
1.0 & \text{if } |y| = 3 
\end{cases}
\]  

(13)

If the string size (i.e. the dimension of the problem) is \( D \), the maximum value is \( D/3 \) for the string 111...111. In practice, we will then use as fitness the value \( D/3-f \) so that the problem is now to find the minimum 0.

4.5. Bipolar order-6

The fitness \( f \) is the sum of the result of applying the following function to consecutive groups of six components each:

\[
f_5(x) = \begin{cases} 
1.0 & \text{if } |y| = 0 \text{ or } 6 \\
0.0 & \text{if } |y| = 1 \text{ or } 5 \\
0.4 & \text{if } |y| = 2 \text{ or } 4 \\
0.8 & \text{if } |y| = 3 
\end{cases}
\]  

(14)

The maximum value is \( D/6 \). In practice, we will use as fitness the value \( D/6-f \) so that the problem is now to find the minimum 0.

Experiments were conducted with four variations of PS2O according to the four symbiotic coevolutionary relationships.

To evaluate the performance of the proposed PS\(^2\)O, the canonical PSO was also used for comparisons.

The population size of all algorithms used in our experiments was set at 60. For canonical PSO, the learning rates
5. Application

The details of proposed PS2O to solve the RNP problem are described as follows.

5.1. Solution representation of RNP problem

5.1.1. Definition of continuous individual

Fig. 11 shows an example of a working area containing 100 RFID tags and 1 RFID reader, where the following three continuous decision variables are chosen in this work:

\[ X: \text{the x axis coordinate value of the RFID reader.} \]
\[ Y: \text{the y axis coordinate value of the RFID reader.} \]
\[ P: \text{the read range (i.e. radiate power level) of the RFID reader.} \]

That is, for base-level searching, each individual in PS2O population can be encoded into solution’s representation shown in Table 2. We employ a representation that each solution is characterized by a \( D = 3 N_t^r (N_t^r \text{ is the total number of available readers that deployed in time step t}) \) dimensional real number vector. In the representation, \( 2 N_t^r \) dimensions denote the coordinates of the readers in the 2-dimensional working area, and the other \( 1 N_t^r \) dimensions denote the interrogation range of each reader (which is determined by the radiate power).

Fig. 11 shows an example of a working area containing 100 RFID tags and 1 RFID reader, where the following three continuous decision variables are chosen in this work:

\[ X: \text{the x axis coordinate value of the RFID reader.} \]
\[ Y: \text{the y axis coordinate value of the RFID reader.} \]
\[ P: \text{the read range (i.e. radiate power level) of the RFID reader.} \]

That is, for base-level searching, each individual in PS2O population can be encoded into solution’s representation shown in Table 2. We employ a representation that each solution is characterized by a \( D = 3 N_t^r (N_t^r \text{ is the total number of available readers that deployed in time step t}) \) dimensional real number vector. In the representation, \( 2 N_t^r \) dimensions denote the coordinates of the readers in the 2-dimensional working area, and the other \( 1 N_t^r \) dimensions denote the interrogation range of each reader (which is determined by the radiate power).

5.1.2. Definition of binary individual

For the top-level searching, the possible solution is represented as an individual with binary encoding scheme according to different time steps. The ith individual of the tth time step is defined as follows:

\[
X_{ij}^T = (x_{i1}^T, x_{i2}^T, \ldots, x_{iN_{\text{max}}}^T), \quad x_{ij}^T \in [0, 1], \quad j = 1, 2, \ldots, N_{\text{max}}
\]

Here \( N_{\text{max}} \) denotes the maximum number of RFID readers according to the total investment for the RFID network. Each element \( r_i \in [0, 1], i = 1, 2, \ldots, N_{\text{max}}, \) in the solution vector is corresponding to the presence or absence of the ith reader. That is, a bit “0” in a solution vector indicated the absence of the corresponding reader, while a bit “1” means the reader’s presence.

Here each element \( X_{ij}^T \) in the dimension is corresponding to the presence or absence of the jth reader that can only be “0” or “1”. A bit “0” in an individual indicated the absence of the corresponding reader. Otherwise a bit “1” in an individual indicated the presence of the corresponding reader. A reader number change example of an individual that represents a potential network planning solution (maximum 6 RFID readers according to the total investment for the RFID network) along the time axis is shown in Fig. 12.

For time step 1, a binary-number individual [0 1 0 0 1 0] is a possible planning solution of a RFID network containing 2 readers. The second and fifth bits mean that the no. 2 and 5 readers can be installed and work together in the
\[ \min_{X_k^i} F_R(X_k^i) = \sum_{j=1}^{n} w_j f_j(X_k^i | X_k^j) = w_1 C(X_k^i | X_k^j) + w_2 I(X_k^i | X_k^j) + w_3 E(X_k^i | X_k^j) + w_4 L(X_k^i | X_k^j) \]  

Fig. 6 Simulation result on Rosenbrock.

\[ \min_{X_k^i} F_R(X_k^i) = \sum_{j=1}^{n} w_j f_j(X_k^i | X_k^j) = w_1 f_1(1 - \exp \left( \sum_{u=1}^{n} c_u x_u T \right)) + F_R(x_1^i) + \sum_{u=1}^{n} c_u x_u T - I_{max} \]

first evolutionary time step; for time step 2, this individual has evolved to be a four readers network; finally, the maximum 6 readers have been installed in order to obtain optimal tag coverage rate.

**Table 1** Performance of all algorithms on benchmark functions f1–f5.

| Func. | PS²O-C | PS²O-P | PS²O-M | PS²O-T | PSO |
|-------|--------|--------|--------|--------|-----|
| f1    | Best   | 1.0803e-009 | 2.2127  | 3.9646e-010 | 1.9175 | 0.0101 |
|       | Worst  | 3.3329  | 3.2295  | 2.9866  | 3.0551 | 4.1661 |
|       | Mean   | 0.3999  | 0.8763  | 0.1793  | 0.5315 | 0.5966 |
|       | Std    | 1.0016  | 0.9846  | 0.1838  | 1.3783 | 1.3883 |
|       | Rank   | 2       | 5       | 3       | 4     |     |
| f2    | Best   | 0       | 0.0428  | 0.1120  | 0.5472 | 0.9950 |
|       | Worst  | 0       | 1.0821  | 2.9546  | 2.9281 | 8.9546 |
|       | Mean   | 0       | 0.9950  | 1.9899  | 1.1429 | 4.3778 |
|       | Std    | 0       | 0.8256  | 1.2950  | 1.8891 | 1.9658 |
|       | Rank   | 1       | 2       | 4       | 3     | 5     |
| f3    | Best   | 0       | 0       | 0       | 0.0011 | 0.0295 |
|       | Worst  | 0       | 0       | 0       | 0.1137 | 0.1697 |
|       | Mean   | 0       | 0       | 0       | 0.0532 | 0.0773 |
|       | Std    | 0       | 0       | 0       | 0.0273 | 0.0367 |
|       | Rank   | 1       | 1       | 1       | 4     | 5     |
| f4    | Best   | 0       | 0       | 0       | 0     | 2.4000 |
|       | Worst  | 0       | 0       | 0       | 0     | 3.6000 |
|       | Mean   | 0       | 0       | 0       | 0     | 2.1833 |
|       | Std    | 0       | 0       | 0       | 0     | 0.2614 |
|       | Rank   | 1       | 1       | 1       | 1     | 5     |
| f5    | Best   | 0       | 0       | 0       | 0     | 1.8000 |
|       | Worst  | 0       | 0       | 0       | 0     | 2.8000 |
|       | Mean   | 0       | 0       | 0       | 0     | 3.1800 |
|       | Std    | 0       | 0       | 0       | 0     | 0.3295 |
|       | Rank   | 1       | 1       | 1       | 1     | 5     |
Fig. 7  Simulation result on Rastrigrin.

Fig. 8  Simulation result on griedwank.

Fig. 9  Simulation result on Golderg’s order-3.
5.2. Model transformation

Then, the base-level objective formulated in Chen et al. (2010a, b) is Eq. (16) to the following optimization problem:

\[ g \text{ is the user defined positive number, } w_6 \text{ is the punishment coefficient, } X_{B^*} \text{ is the last best base-level decision, and the notation } (\cdot)^+ \text{ is defined as follows Eq. (18):} \]

\[
(\cdot)^+ = \begin{cases} \sigma & \text{if } \sigma > 0 \\ 0 & \text{else} \end{cases}
\]

5.3. Implementation of the PS2O algorithm for the RNP problem

To apply the PS2O algorithms to solve the RNP problem, the following steps should be taken and repeated.

Step 1. The first step in top-level is to randomly initialize the binary PS2O-based upper-population. Each individual \(X^T_i\) in the top-level is an instruction and is communicated to the base-level to drive a base-level search process (Step 2–4).

Step 2. For each top-level instruction \(X^T_i\), the base-level randomly initialize a corresponding continuous PS2O-based lower-population. At each iteration in base-level, for each individual \(X_{B^*}^B\), evaluate its fitness using the base-level optimization function, i.e., Eq. (16).

Step 3. Compare the evaluated fitness values for all individuals in lower-population. Then update the each base-level individual by its updating rules according to the PS2O algorithm.

*Fig. 10*  Simulation result on Bipolar order-6.

*Fig. 11*  A 30 m x 30 m working area with one RFID reader and 100 tags. The radiate power level (dBm) marked on the right diagram sidebar and the yellow cycle represents the read range of the RFID reader.

Table 2  Representation of an individual solution \(X_i\) for the base-level population.

| Reader1 variables | Reader2 variables | ... | ReaderNr variables |
|-------------------|-------------------|-----|--------------------|
| \(x^B_{i1}\) | \(x^B_{i2}\) | \(y^B_{i1}\) | \(y^B_{i2}\) | \(p^B_{i1}\) | \(p^B_{i2}\) | ... | \(x^B_{iN_r}\) | \(y^B_{iN_r}\) | \(p^B_{iN_r}\) |

(Nr is the max number of RFID readers that deployed in the working area).

*Fig. 12*  The individual evolution process in top-level.

\[
(\sigma)^+ = \begin{cases} \sigma & \text{if } \sigma > 0 \\ 0 & \text{else} \end{cases}
\]
Step 4. The base-level search process is repeated until the maximum number of base-level iteration is met. Then send the last best base-level decision variable $X^b_i$ to the top-level for the fitness computation of the top-level individual $X^t_i$.

Step 5. With the base-level reaction $X^b_i$, each top-level individual $X^t_i$ is evaluated by the following top-level fitness function, i.e., Eq. (17).

**Table 3** The example for Cd100, and Rd500.

| Reader specification | Topology specification |
|----------------------|------------------------|
| Cd100                |                        |
| Reader number        | 10                     |
| Radiated power       | 0.1–2 W                |
| Interrogation range  | 3–4 m                  |
| Interference range   | 3.5–4.5 m              |
|                      |                        |
|                      |                        |
|                      |                        |
| Rd500                |                        |
| Reader number        | 50                     |
| Radiated power       | 0.1–2 W                |
| Interrogation range  | 3–4 m                  |
| Interference range   | 3.5–4.5 m              |
|                      |                        |
|                      |                        |

**Fig. 13** The RFID network planning process based on PS^2O.

**Fig. 14** The test working area tags deployment: (a) cluster deploying with 100 tags (Cd100), (b) randomly deploying with 500 tags (Rd500).
Step 6. Compare the evaluated fitness values for all individuals in top-population. Then update each top-level individual by its updating rules according to the PS\textsuperscript{2}O algorithm. The top-level computation is repeated until the maximum number of top-level iteration is met.

The flowchart of this RFID network planning process is illustrated in the diagram given in Fig. 13.

5.4. Simulation configurations

The readers used here are mobile and the tags are passive. According to the references (Wang and Tseng, 2008; Dobkin, 2004), the related RFID readers’ parameters can be set as in Table 3. Here the interrogation range according to the reader radiated power is computed as in (Dobkin, 2004).

The proposed algorithm is evaluated against two different RNP instances, namely Cd100, and Rd500. The instance of Cd100 is tested on a 30 m $\times$ 30 m working space with 100 clustered distributed tags. Another instance, namely Rd500, contains 500 randomly distributed tags in a 150 m $\times$ 150 m working space (shown in Fig. 14). Experiments were conducted on the RNP cases to compare the mutualism version of PS2O with successful EA, and SI algorithms, including original artificial bee colony algorithm (ABC) (Karaboga and Basturk, 2007), canonical PSO with constriction factor (PSO), and genetic algorithm with elitism (EGA) (Sumathi et al., 2008).

All the control parameters for the compared EA and SI algorithms are set to be default of their original literatures: the limit parameter of ABC is set to be $SN \times D$, where $D$ is the dimension of the problem and $SN$ is the number of employed bees; for canonical PSO, the learning rates $c1$ and $c2$ are both set as 2.05 and the constriction factor $\chi = 0.729$; for EGA, crossover rate of 0.8, mutation rate of 0.01, and the global elite with a rate of 0.06 are adopted (Sumathi et al., 2008); and all the algorithms’ population size is 50. For PS\textsuperscript{2}O, the number of swarms and swarm size can be set by 10 and 5, and the constriction factor is used with 0.729, and then the learning rates $c1 = c2 = c3 = 1.3667$.

5.5. Simulation results

Figs. 15 and 16 demonstrate the reader locations and radiated power contours with redundant reader elimination strategy on the two cases –Cd100 and Rd500, in which the best results obtained by the PS\textsuperscript{2}O-based schedule and other algorithms
are compared in a visible way. A contour in the figures represents the points with the same radiated power (which equals to the value assigned to that contour), which means that the power peaks in the working area are the points where the readers are placed in the figures. Then, the signal strength decreases with respect to the distance to the readers.

From these figures, it is clearly visible that PS2O deploys the minimal number of readers on the optimal positions to

![Fig. 16 Reader location and received power distribution on Rd500 by involved algorithms with redundant reader elimination](image)

| Table 4 | Comparisons of the results of involved algorithms for solving instance Cd100. |
|---------|-----------------------------------------------------------------|
| Algorithm | Mean | Best |
|          | Coverage (%) | Reader | Interface | Power | Coverage (%) | Reader | Interface | Power |
| PS2O     | 99.8  | 9      | 0.2121    | 41.6543 | 100    | 8      | 0.0321    | 41.6956 |
| EGA      | 96.22 | 10     | 0.2311    | 40.9332 | 98.84  | 10     | 0.1342    | 40.9875 |
| ABC      | 98.33 | 9      | 0.1612    | 40.3223 | 100    | 9      | 0.0932    | 40.4534 |
| PSO      | 98.43 | 9      | 0.2876    | 40.6422 | 100    | 9      | 0.0932    | 40.4534 |

| Table 5 | Comparisons of the results of involved algorithms for solving instance Rd500. |
|---------|-----------------------------------------------------------------|
| Algorithm | Mean | Best |
|          | Coverage (%) | Reader | Interface | Power | Coverage (%) | Reader | Interface | Power |
| PS2O     | 99.80 | 45     | 0.2874    | 41.7324 | 100    | 40     | 0.0452    | 41.6645 |
| EGA      | 97.65 | 48     | 0.2425    | 40.8433 | 98.62  | 47     | 0.2165    | 40.7375 |
| ABC      | 98.65 | 46     | 0.1734    | 40.4523 | 100    | 42     | 0.1232    | 40.3452 |
| PSO      | 99.65 | 47     | 0.2904    | 40.7424 | 99.89  | 43     | 0.1045    | 40.6735 |
ensure almost full coverage. Other algorithms, such as EGA, PSO, and ABC, have redundant readers and cannot obtain full coverage for Cd100 and Rd500, just as shown in Tables 4 and 5, which indicates that the proposed PS2O outperforms these compared algorithms EGA, PSO and ABC.

These experimental results show that the PS2O-based schedule with redundant reader elimination is superior to that using a fixed number of readers in the optimization. Because the required number of readers is hard to estimate, and we should note that deploying too many readers in the RFID network is not cost-efficient, and on the other hand too few readers cannot guarantee the coverage. In contrast, the two-level RFID reader network manage model based on distributed decision making allows intelligent algorithm adaptively delete redundant readers during the evolution, and finally obtain a high coverage using a proper number of readers. This can be explained why PS2O-based schedule with redundant reader elimination performs most powerfully in the experiment.

6. Conclusions

The models of evolution that are being used in evolutionary and swarm intelligence algorithms are becoming increasingly complex. Hence, this work proposed a symbiotic multi-swarm coevolution optimization model named PS2O. The model has been instantiated as four novel multi-species optimizers, which extends the dynamics of the canonical PSO algorithm by adding a significant ingredient that takes into account four different symbiotic coevolution mechanisms respectively.

A set of 5 benchmark functions including unimodal, multimodal, and discrete cases have been used to test the proposed PS2O algorithms in comparison with the canonical PSO. The simulation results show that, for all benchmark functions, the PS2O variants reach remarkably better fitness values than PSO. We also simulated the coevolution process of symbiotic species’ diversity in our model. The results capture some important aspects of the dynamics of biological coevolution that some evolutionary biologists believe takes place in nature. PS2O is then employed to solve the real-world RNP problem on two different-scale instances, namely Cd100 and Rd500. The simulation studies show that the PS2O remarkably outperforms other algorithms.

Acknowledgement

This research is partially supported by National Natural Science Foundation of China and Grants 51378350, 5157518, 61602343, 61305082, and 51607122.

References

Bonabeau, E., Dorigo, M., Theraulaz, G., 1999. Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, New York.
Chen, H.N., Zhu, Y.L., 2008. Optimization based on symbiotic multi-species coevolution. Appl. Math. Comput. 205, 47–60.
Chen, H.N., Zhu, Y.L., Hu, K.Y., 2009. Cooperative bacterial foraging optimization, Discrete Dynamic in Nature and Society.
Chen, H.N., Zhu, Y.L., Hu, K.Y., 2010a. Virtual enterprise risk management using artificial intelligence. Math. Prob. Eng., 1–20
Chen, H.N., Zhu, Y.L., Hu, K.Y., 2010b. Discrete and continuous optimization based on multi-swarm coevolution. Nat. Comput. 9, 659–682.
Dobkin, D.M., 2004. The RF in RFID: passive UHF RFID in practice. Elsevier, 68–77.
Frank, A., 1997. Models of symbiosis. Am. Nat. 150, 80–99.
Frank, S., 1995. The origin of synergistic symbiosis. J. Theor. Biol. 176, 403–410.
Karaboga, D., Basturk, B., 2007. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm. J. Global Optim. 39 (3), 459–471.
Kennedy, J., Eberhart, R.C., 2001. Swarm Intelligence. Morgan Kaufmann Publishers, San Francisco.
Mendes, R., Kennedy, J., Neves, J., 2004. The fully informed particle swarm: simpler, maybe better. IEEE Trans. Evol. Comput. 8, 204–210.
Niu, B., Zhu, Y.L., He, X.X., Zeng, X.P., Wu, H., 2007. MCPSO: a multi-swarm cooperative particle swarm optimizer. Appl. Math. Comput. 185, 1050–1062.
Samath, S., Hamsapriya, T., Surekha, P., 2008. Evolutionary Intelligence: An Introduction to Theory and Applications with Matlab. Springer.
Wang, Y.C., Tseng, Y.C., 2008. Distributed deployment schemes for mobile wireless sensor networks to ensure multilevel coverage. IEEE Trans. Parallel Distrib. Syst. 19 (9), 1280–1294.