

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# Use of Particle Multi-Swarm Optimization for Handling Tracking Problems

*Hiroshi Sho*

## Abstract

As prior work, several multiple particle swarm optimizers with sensors, that is, MPSOS, MPSOIWS, MCPSOS, and HPSOS, were proposed for handling tracking problems. Due to more efficient handling of these problems, in this chapter we innovate the strategy of information sharing (IS) to these existing methods and propose four new search methods that are multiple particle swarm optimizers with sensors and information sharing (MPSOSIS), multiple particle swarm optimizers with inertia weight with sensors and information sharing (MPSOIWSIS), multiple canonical particle swarm optimizers with sensors and information sharing (MCPSOSIS), and hybrid particle swarm optimizers with sensors and information sharing (HPSOSIS). Based on the added strategy of information sharing, the search ability and performance of these methods are improved, and it is possible to track a moving target promptly. Therefore, the search framework of particle multi-swarm optimization (PMSO) is established. For investigating search ability and characteristics of the proposed methods, several computer experiments are carried out to handle the tracking problems of constant speed I type, variable speed II type, and variable speed III type, which are a set of benchmark tracking problems. Owing to analyze experimental results, we reveal the outstanding search performance and tracking ability of the proposed search methods.

**Keywords:** swarm intelligence, particle multi-swarm optimization, information sharing, sensor, tracking performance

## 1. Introduction

Generally, the task of tracking a moving target is an important subject as a real-world problem, for example, traffic management, mobile robot, safety guidance, image object recognition, industrial controls, etc. which are frequently taken up in various application fields [1–5]. In order to deal with dynamic optimization problems, many search methods are applied, and the approach of particle swarm optimization (PSO) in the area of swarm intelligence is one of them [6–11].

The technique of PSO is very easy to implement and extend. Based on its basic search mechanism, main advantages have three built-in features: (i) information exchange, (ii) intrinsic memory, and (iii) directional search, compared to other existing heuristic and evolutionary techniques such as genetic algorithms (GA), evolutionary programming (EP), evolution strategy (ES), and so on [12–15]. This is

a reason why the technique of PSO is attracting attention and used in different fields such as science, engineering, technology, design, automation, communication, etc.

As is well-known, the search tasks handled by the technique of PSO are a mass of static optimization problems. The cause is simple for that the best information, that is, the best solution of swarm search and the best solution of each particle itself, is only recorded and renewed. Due to environmental change, the retained best information is not modified to normally search. Thus, its mechanism cannot adapt environment change or a moving target for dealing with tracking problems. Because of overcoming the disadvantage of the technique of PSO, and extending the range of its applications for dealing with dynamic optimization problems (including tracking problem), it is necessary to improve its search functions by adding some strategies into the mechanism of PSO [16, 17].

As prior work on handling the tracking problems by PSO, under a certain dynamic environment, we have proposed not only three single particle swarm optimizer with sensors, which are PSOS, PSOIWS, and CPSOS [18], but also four multiple particle multi-swarm optimizers with sensors which are MPSOS, MPSOIWS, MCPSOS, and HPSOS<sup>1</sup> [19]. And for confirming the search effectiveness of these proposed methods, several computer experiments were carried out to handle the tracking problems of constant speed I type, variable speed II type, and variable speed III type that belong to a set of benchmark problems.

In general, the search ability and performance of multiple particle swarms are better than single particle swarm for handling same tracking problem. The comparative experiments on the finding were verified in literature [20]. According to the obtained experimental results of the four multiple particle swarm optimizers with sensors, MPSOIWS and HPSOS are better in search ability. MCPSOS is better in convergence. MPSOS is better in the robustness with respect to variation in sensor setting parameters. And many know-hows on the useful knowledge such as their experimental findings are obtained [19]. As the search characters of particle multi-swarm optimization (PMSO<sup>2</sup>), however, the search information (i.e., best solution) obtained from each particle swarm is not shared to explore. For dealing with this issue, we proposed a special strategy called information sharing and introduced it to effectively solve static optimization problems [21].

In order to acquire further the search ability and performance of PMSO in dealing with dynamic optimization problems, we innovate the strategy of information sharing into the previous four multiple particle multi-swarm optimizers with sensors and firstly propose the four new search methods, that is, multiple particle swarm optimizers with sensors and information sharing (MPSOSIS), multiple particle swarm optimizers with inertia weight with sensors and information sharing (MPSOIWSIS), multiple canonical particle swarm optimizers with sensors and information sharing (MCPSOSIS), and hybrid particle swarm optimizers with sensors and information sharing (HPSOSIS<sup>3</sup>).

This is a novel approach for the technology development and evolution of PMSO itself. The crucial idea is to add the special confidence term into the updating rule of the particle's velocity by the best solution found out by particle multi-swarm search to enhance the intelligent level of whole particle multi-swarm and build a

---

<sup>1</sup>HPSOS has the search characteristics of PSOS, PSOIWS, and CPSOS, which is a mixed search method.

<sup>2</sup>PMSO, generally, is just a variant of PSO based on the use of multiple particle swarms (including sub-swarms) instead of a single particle swarm during a search process.

<sup>3</sup>HPSOSIS has the search characteristics of PSOSIS, PSOIWSIS, and CPSOSIS, which is a proposed method in PMSO.

new framework of PMSO [22]. Based on the improvement of the confidence terms, it is expected to acquire the maximization of potential search ability and performance of the four basic search methods of PMSO under the context of any adjunctive computation resource.

Due to the revelation of the outstanding search ability and performance of the proposed MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS, we take more detailed data from the computer experiments. Based on these obtained data, furthermore, we clarify the characteristics and search ability of the proposed methods by analysis and comparison. This is the major goal of this research.

The rest of this chapter is organized as follows: Section 2 briefly introduces three basic search methods of PSO and these methods with sensors. Section 3 describes the proposed four search methods of PMSO in detail. Section 4 implements several computer experiments and analyzes the obtained results for investigating the search ability and performance of these new search methods. Finally, the concluding remarks and future research appear in Section 5.

## 2. Basic search methods of PSO

In spite of the fact that there are a lot of search methods derived from the technique of PSO, they have evolved and developed from three basic search methods of PSO [23]. These search methods, that is, the particle swarm optimizer (the PSO) [24, 25], particle swarm optimizer with inertia weight (PSOIW) [26, 27], and canonical particle swarm optimizer (CPSO) [28, 29], are common ground for technology development of PSO and PMSO.

For the sake of convenience to the following specific description, let the search space be  $N$ -dimensional,  $\Omega \in R^N$ , the number of particles in a swarm be  $Z$ , the position of the  $i$ th particle be  $\vec{x}^i = (x_1^i, x_2^i, \dots, x_N^i)^T$ , and its velocity be  $\vec{v}^i = (v_1^i, v_2^i, \dots, v_N^i)^T$ , respectively.

### 2.1 Method of the PSO

The original particle swarm optimizer is firstly created by Kennedy and Eberhart in 1995. The method of a population-based stochastic optimization search is referred to as the PSO.

In beginning of the PSO search, the position and velocity of the  $i$ th particle are generated at random; then they are updated by the following formulation:

$$\vec{x}_{k+1}^i = \vec{x}_k^i + \vec{v}_{k+1}^i \quad (1)$$

$$\vec{v}_{k+1}^i = w_0 \vec{v}_k^i + w_1 \vec{r}_1 \otimes (\vec{p}_k^i - \vec{x}_k^i) + w_2 \vec{r}_2 \otimes (\vec{q}_k - \vec{x}_k^i) \quad (2)$$

where  $w_0$  is an inertia weight,  $w_1$  is a coefficient for individual confidence, and  $w_2$  is a coefficient for swarm confidence.  $\vec{r}_1$  and  $\vec{r}_2 \in^N$  are two random vectors in which each element is uniformly distributed over the range  $[0, 1]$ , and the symbol  $\otimes$  is an element-wise operator for vector multiplication.  $\vec{p}_k^i (= \arg \max_{j=1, \dots, k} \{g(\vec{x}_j^i)\})$ , where  $g(\cdot)$  is the criterion value of the  $i$ th particle at time-step  $k$  is the local best position of the  $i$ th particle until now, and  $\vec{q}_k (= \arg \max_{i=1, 2, \dots} \{g(\vec{p}_k^i)\})$  is the global best position among the whole swarm.

In the PSO,  $w_0 = 1.0$  and  $w_1 = w_2 = 2.0$  are used. Since  $w_0 = 1.0$ , so the convergence of the PSO is not good in whole search process [30]. It has the characteristics of global search.

## 2.2 Method of PSOIW

For improving the convergence and search ability of the PSO, Shi and Eberhart modified the updating rule of the particle's velocity shown in Eq. (2) by constant reduction of the inertia weight over time-step as follows:

$$\vec{v}_{k+1}^i = w(k)\vec{v}_k^i + w_1\vec{r}_1 \otimes (\vec{p}_k^i - \vec{x}_k^i) + w_2\vec{r}_2 \otimes (\vec{q}_k^i - \vec{x}_k^i) \quad (3)$$

where  $w(k)$  is a variable inertia weight which is linearly reduced from a starting value,  $w_s$ , to a terminal value,  $w_e$ , with the increment of time-step  $k$  given by

$$w(k) = w_s + \frac{w_e - w_s}{K}k \quad (4)$$

where  $K$  is the maximum number of time-step for PSOIW searching. In the original PSOIW, the boundary values are adopted to  $w_s = 0.9$  and  $w_e = 0.4$ , respectively, and  $w_1 = w_2 = 2.0$  are still used as the PSO.

Since the linear change of inertia weight from 0.9 to 0.4 in a search process, PSOIW has the characteristics of asymptotical/local search, and its convergence is so good in whole search process.

## 2.3 Method of CPSO

For the same purpose as the above described, Clerc and Kennedy modified the updating rule for the particle's velocity in Eq. (2) by a constant inertia weight over time-step as follows:

$$\vec{v}_{k+1}^i = \Phi \left( \vec{v}_k^i + w_1\vec{r}_1 \otimes (\vec{p}_k^i - \vec{x}_k^i) + w_2\vec{r}_2 \otimes (\vec{q}_k^i - \vec{x}_k^i) \right) \quad (5)$$

where  $\Phi$  is an inertia weight corresponding to  $w_0$ . In the original CPSO,  $\Phi = w_0 = 0.729$  and  $w_1 = w_2 = 2.05$  are used.

It is clear that since the value of inertia weight,  $\Phi$ , of CPSO is smaller than 1.0, the convergence of its search is guaranteed by compared with the PSO search [30, 31]. It has the characteristics of local search.

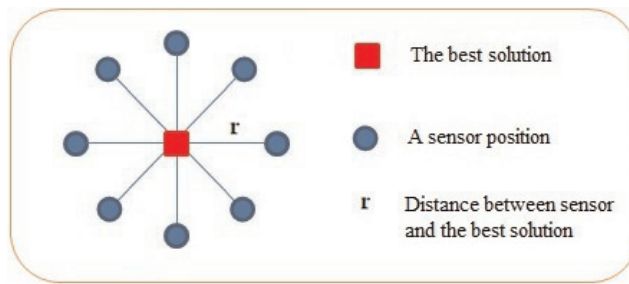
## 2.4 Basic search methods with sensors

We introduce the correspond to these foregoing search methods which are particle swarm optimizers with sensors to handle dynamic optimization problems. With adding sensors into the search methods of every particle swarm optimizer described in Sections 2.1–2.3, it is possible to sense environmental change and a moving target for improving the search ability and performance.

As an example, **Figure 1** shows the positional relationship between the best solution and sensors.

In a search process, the best solution of entire particle swarm is always set as the origin of the sensor setting. Based on the sensing information (i.e., the measuring position and its fitness value) of each sensor, we can observe the change of the





**Figure 1.**  
 Configuration of sensors.

surrounding environment and the moving target. In particular, updating the best solution by Eq. (6) is an important search information:

$$\vec{q}_k = \begin{cases} \vec{y}_k^b, & \text{if } g_t(\vec{y}_k^b) = \max_{j=1,2,\dots} \{g_t(\vec{y}_k^j)\} > g_t(\vec{q}_k); \vec{q}_k, & \text{otherwise} \end{cases} \quad (6)$$

where  $\vec{y}_k^j$  is the  $j$ th sensor's position (i.e., solution) at time  $k$ ,  $\vec{y}_k^b$  is the best solution for sensor detection, and  $g_t(\cdot)$  is the criterion for evaluation at time  $t$ .

On the other hand, regarding whether there are environmental change and a moving target or not, it is implemented by using the following judgment criterion:

$$\Delta_k = g_t(\vec{q}_{k-1}) - g_{t-1}(\vec{q}_{k-1}) < 0 \quad (7)$$

where  $\Delta_k$  is the difference in the fitness values between different functions at the best solution  $\vec{q}_{k-1}$ .

If the judgment result of Eq. (7) is satisfied in a search process, the moving target has occurred. The particle swarm is initialized at the time and then continuous to begin particle swarm search. However, such initialization is not considered on the continuity of environmental change; it is implemented around the coordinate origin of the search range. As a new problem in the situation, if the distance before and after movement becomes smaller, the time loss to search is greater for finding out the new best solution.

By changing the coordinate origin of the initialization to the position of the best solution, the above difficulty can be dissolved. Therefore, the best solution of whole particle swarm is intermittently updated by sensing information.

And adding the judgment operation of Eqs. (6) and (7) into each method described in Sections 2.1–2.3, the constructions of the search methods, that is, particle swarm optimizer with sensors (PSOS), particle swarm optimizer with inertia weight with sensors (PSOIWS), and canonical particle swarm optimizer with sensors (CPSOS), can be conflated and completed to deal with the given tracking problems.

### 3. Basic search methods of PMSO

Formally, there are a lot of the methods about PMSO [32]. For understanding the formation and methodology of these proposed methods, let us assume that the multi-swarm consists of multiple single swarms. The corresponding three kinds of particle swarm optimizers described in Sections 2.1–2.3 can be generated by construction and parallel computation [33]. Therefore, these constructed particle multi-swarm optimizers, i.e. multiple particle swarm optimizers (MPSO), are

multiple particle swarm optimizers with inertia weight (MPSOIW), multiple canonical particle swarm optimizers (MCPSO), and hybrid particle swarm optimizers (HPSO), respectively.

Based on the development of the search methods in Section 2.4, similarly, multiple particle swarm optimizers with sensors (MPSOS), multiple particle swarm optimizers with inertia weight with sensors (MPSOIWS), multiple canonical particle swarm optimizers with sensors (MCPSOS), and hybrid particle swarm optimizers with sensors (HPSOS) were acquired by programming [19].

However, all of their updating rules have two confidence terms in the Eqs. (2), (3), and (5) to be only used for calculating the particle's velocity. Because of the use of the mechanism to search, they are called as the elementary basic methods with sensors of PMSO which have the same updating rule of the particle's velocity [20, 34].

For improving the search ability and performance of the previous described elementary multiple particle swarm optimizers, furthermore, we add the special confidence term into the updating rule of the particle's velocity by the best solution found out by the multi-swarm search, respectively. According to this extended procedure, the four basic search methods of PMSO, that is, MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS, can be constructed [18]. Consequently, these basic search methods of PMSO augmented with the strategy of multi-swarm information sharing are proposed [22].

It is clear that the added confidence term perfectly is in accordance with the fundamental construction principle of PSO. And the effectiveness of the methods has been verified by our experimental results [21].

### 3.1 Method of MPSOSIS

On basis of the above description of PMSO, as the mechanism of the proposed MPSOSIS, the updating rule of each particle's velocity is defined as follows:

$$\vec{v}_{k+1}^i = w_0 \vec{v}_k^i + w_1 \vec{r}_1 \otimes (\vec{p}_k^i - \vec{x}_k^i) + w_2 \vec{r}_2 \otimes (\vec{q}_k - \vec{x}_k^i) + w_3 \vec{r}_3 \otimes (\vec{s}_k - \vec{x}_k^i) \quad (8)$$

where  $\vec{s}_k (= \arg \max_{j=1, \dots, S} \{g(\vec{q}_k)_j\})$ .  $S$  is the number of the used swarms and is the best solution chosen from the best solution of each swarm,  $w_3$  is a new confidence coefficient for the multi-swarm, and  $\vec{r}_3$  is a random vector in which each element is uniformly distributed over the range  $[0, 1]$ .

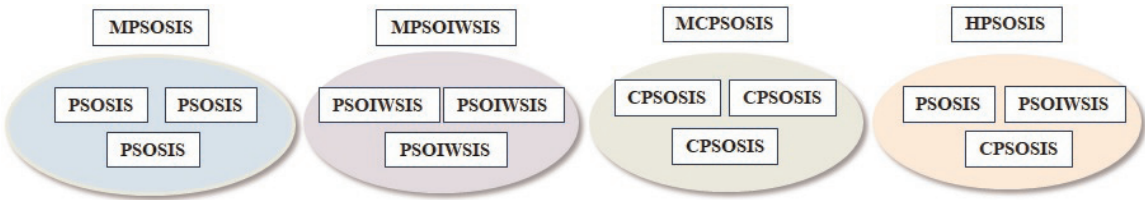
Since  $w_0 = 1.0$  is used in each particle swarm search, the convergence of MPSOSIS is not better than the PSO.

### 3.2 Method of MPSOIWSIS

In same way as the mechanism of MPSOSIS, the updating rule of each particle's velocity of the proposed MPSOIWSIS is defined as follows:

$$\vec{v}_{k+1}^i = w(k) \vec{v}_k^i + w_1 \vec{r}_1 \otimes (\vec{p}_k^i - \vec{x}_k^i) + w_2 \vec{r}_2 \otimes (\vec{q}_k - \vec{x}_k^i) + w_3 \vec{r}_3 \otimes (\vec{s}_k - \vec{x}_k^i) \quad (9)$$

Since Eqs. (3) and (6) are alike in formulation, the description of the symbols in Eq. (9) is omitted. Similarly, the convergence of MPSOIWSIS is as same as that of PSOIW.



**Figure 2.**  
*The constitutional concept of the proposed four basic search methods with sensors of PMSO.*

### 3.3 Method of MCPISOIS

Similar to the mechanism of MPSOSIS, the updating rule of each particle's velocity of the proposed MCPISOIS is defined as follows:

$$\vec{v}_{k+1}^i = \Phi \left( \vec{v}_k^i + w_1 \vec{r}_1 \otimes (\vec{p}_k^i - \vec{x}_k^i) + w_2 \vec{r}_2 \otimes (\vec{q}_k^i - \vec{x}_k^i) + w_3 \vec{r}_3 \otimes (\vec{s}_k^i - \vec{x}_k^i) \right) \quad (10)$$

Likewise, the description of the symbols in Eq. (10) is omitted. Since  $\Phi = w_0 = 0.729$ , the convergence of MCPISOIS is as same as that of CPSO.

### 3.4 Method of HPSOSIS

Based on the three search methods described in Sections 3.1–3.3, there are the three updating rules of each particle's velocity in the proposed HPSOSIS. The mechanism of HPSOSIS is determined by Eqs. (8)–(10).

Due to the mixed effect and performance in whole search process, global search and asymptotical/local search are implemented simultaneously for dealing with a given optimization problem. It is obvious that HPSOSIS has all search characteristics of the three basic methods, that is, PSO, PSOIW, and CPSO. Similarly, the convergence of HPSOSIS is as same as that of HPSOS.

Based on the development of these methods in Section 2.4, here, we propose the four basic methods with sensors of PMSO and describe the search methods with sensors, that is, MPSOSIS, MPISOIWSIS, MCPISOIS, and HPSOSIS, respectively, by constructing the particle swarm optimizers with sensors described in Sections 3.1–3.3.

For indicating the image relation of the above described methods with sensors, **Figure 2** simply shows the constitutional concept of the proposed four basic search methods with sensors of PMSO. It is clear that HPSOSIS is a mixed method which is composed of PSOSIS, PSOIWSIS, and CPSOSIS. Thus, HPSOSIS has different characteristics of the above methods as a special basic search method with sensors of PMSO [18].

Regarding the convergence of the above proposed methods, it can be said that the MPSOSIS has the characteristics of global search, MPISOIWSIS has the characteristics of asymptotical/local search, and MCPISOIS has the characteristics of local search. With different search features, HPSOSIS has the characteristics of the above three search methods. In a search process, it is expected to improve the potential search ability and performance of PMSO without additional calculation resource.

## 4. Computer experiments and result analysis

Due to the track of a moving target, the setting parameters of each proposed method described in Section 2.1 are used in every search case. The main parameters are shown in **Table 1** for the following computer experiments.



Parameter	Value
Number of the used swarms, $S$	3
Number of particles in a swarm, $Z$	10
Total number of particle search, $K$	800
Radius of moving target, $R$	2.0
Number of sensors, $m$	5, 8, 11, 14
Sensing distance, $r$	0.0, 0.1, ..., 1.0

**Table 1.**  
Major parameters for handling the given tracking problems in computer experiments.

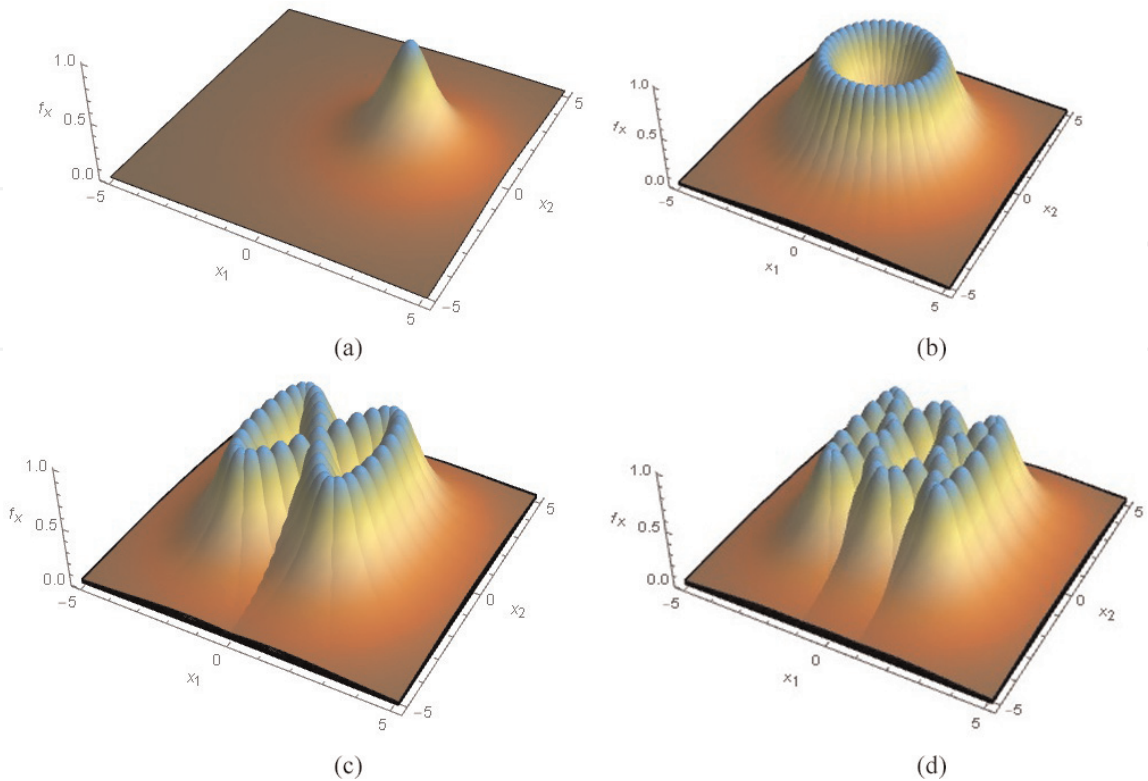
The computing environment and software tool are given as follows:

- DELL: OPTIPLEX 3020, Intel(R) core (TM) i5-4590
- CPU: 3.30GHz; RAM: 8.0GB
- Mathematica: ver. 11.3

The tracking problems of constant speed I type, variable speed II type, and variable speed III type are used in the following computer experiments. A target object and its moving trajectories are shown in **Figure 3**. The search range of all cases is limited to  $\Omega \in (-5.12, 5.12)^2$ .

The criterion of the moving target is expressed as follows:

$$g_t(\vec{x}_k) = \frac{1}{1 + (x_k^1 - x_a(t))^2 + (x_k^2 - x_b(t))^2} \Big|_{t \in T} \tag{11}$$



**Figure 3.**  
Trajectories of the moving target. (a) Target object, (b) moving trajectory of constant speed I type, (c) moving trajectory of variable speed II type, and (d) moving trajectory of variable speed III type.

where  $(x_a(t), x_b(t))$  is the center coordination (position) of the moving target at  $t$  time.  $T$  is a set of time series.

Specifically, for the moving trajectory of constant speed I type,  $(x_a(t), x_b(t))$  is given as follows:

$$\begin{pmatrix} x_a(t) \\ x_b(t) \end{pmatrix} = R \times \begin{pmatrix} \cos\left(\frac{2\pi}{K} \times t\right) \\ \sin\left(\frac{2\pi}{K} \times t\right) \end{pmatrix}, \quad t \in T = \{0, 20, 40, \dots, K\} \quad (12)$$

where  $K$  is the total number of searching on the whole circle of the particle swarm search. The target object goes 40 steps with a regular interval for the tracking problem of constant speed I type, the radius  $R$  of its trajectory is 2.0, and the fitness value of center (vertex) position of the moving target is 1.0.

The moving trajectories of variable speed II type and variable speed III type and their passing points,  $(x_a(t), x_b(t))$ , are determined by adjusting the coefficient of the time  $t$  up to two and three times for calculating  $x_b(t)$ , respectively.

The difficulty index ( $DI$ ) for handling these tracking problems shown in **Figure 3(b)–(d)** is defined as follows:

$$DI = \frac{D_{max}}{D_{min}} \quad (13)$$

where  $D_{max}$  and  $D_{min} (\neq 0)$  are the distance between maximum and minimum of the moving target object used, respectively.

By concreting calculation, the  $DI$ s of the tracking problems of constant speed I type, variable speed II type, and variable speed III type are 1.0, 3.06, and 5.39, respectively.

#### 4.1 Characteristics of tracking target

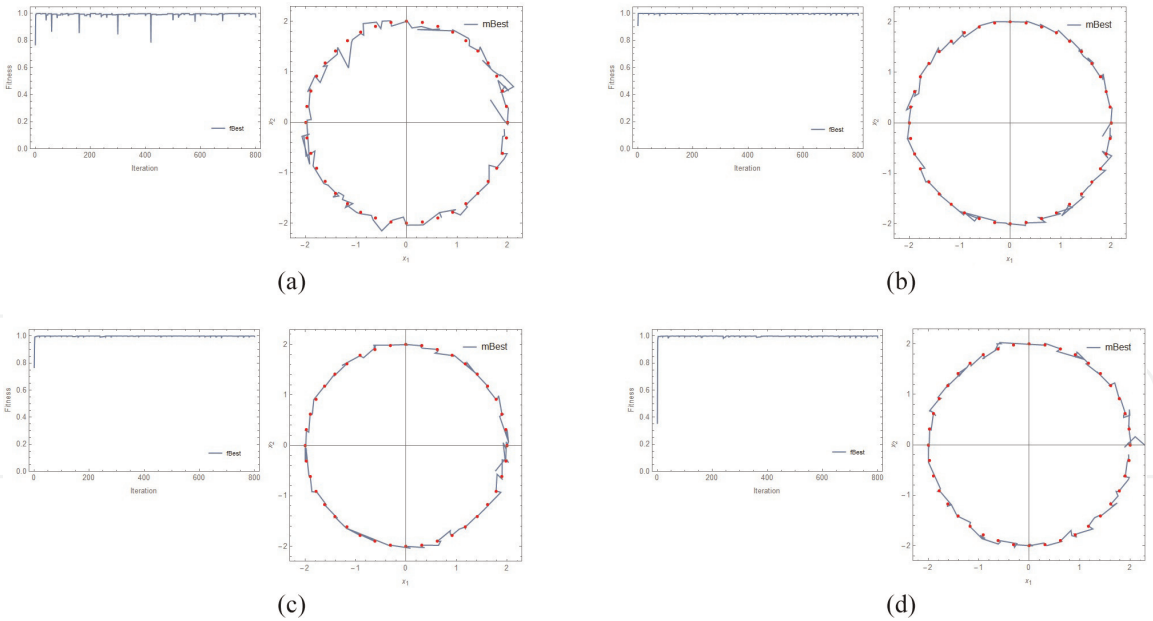
In this section, we implement the proposed methods, that is, MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS, respectively, for handling the three tracking problems shown in **Figure 3** and investigating their search ability and performance in detail.

First, MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS were performed<sup>4</sup> to handle the tracking problem of constant speed I type which has a low-level of difficulty, respectively. As an example, the obtained change patterns of the fitness value of the best solution and the moving trajectory are shown in **Figure 4**.

We can see that the obtained variation of the best solution in whole search process from the left parts of **Figure 4** and the search trajectories are beautifully drawn from the right parts of **Figure 4(a)–(d)**, except for **Figure 4(a)**. And comparing to the left parts of **Figure 4(a)–(d)**, a big difference of the search state is clear with the origin of searching range as the center of initialization and the best solution as the center of initialization. The moving trajectories of the latter are relatively flat.

Moreover, when the target object moves, the fitness value of the best solution of the particle multi-swarm suddenly drops, then it rapidly rises with the subsequent search, and it is found that the peak of the target object is attained again. On the other hand, depending on the variation in the fitness value in the time space of

<sup>4</sup>The search time is about 1.3 s for handling the tracking problem of constant speed I type.

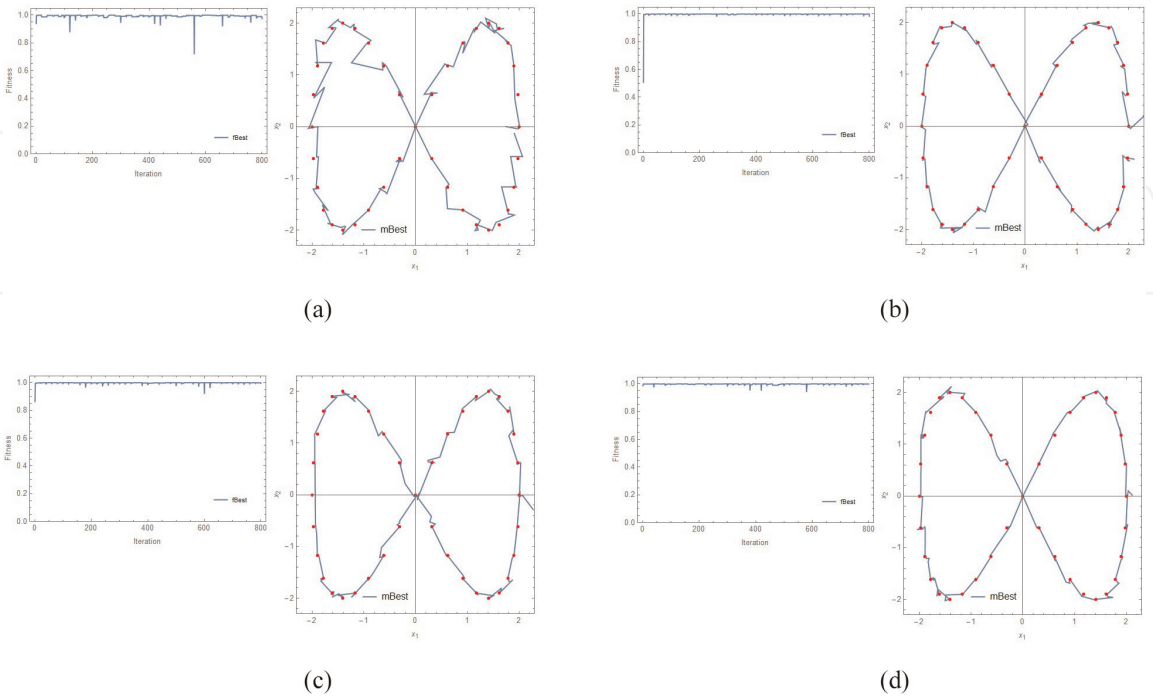


**Figure 4.** The moving trajectory of the best solution for handling the tracking problem of constant speed I type. Left part, time space; right part, search space. (a) MPSOSIS case, (b) MPSOIWSIS case, (c) MCPSOSIS case, and (d) HPSOSIS case.

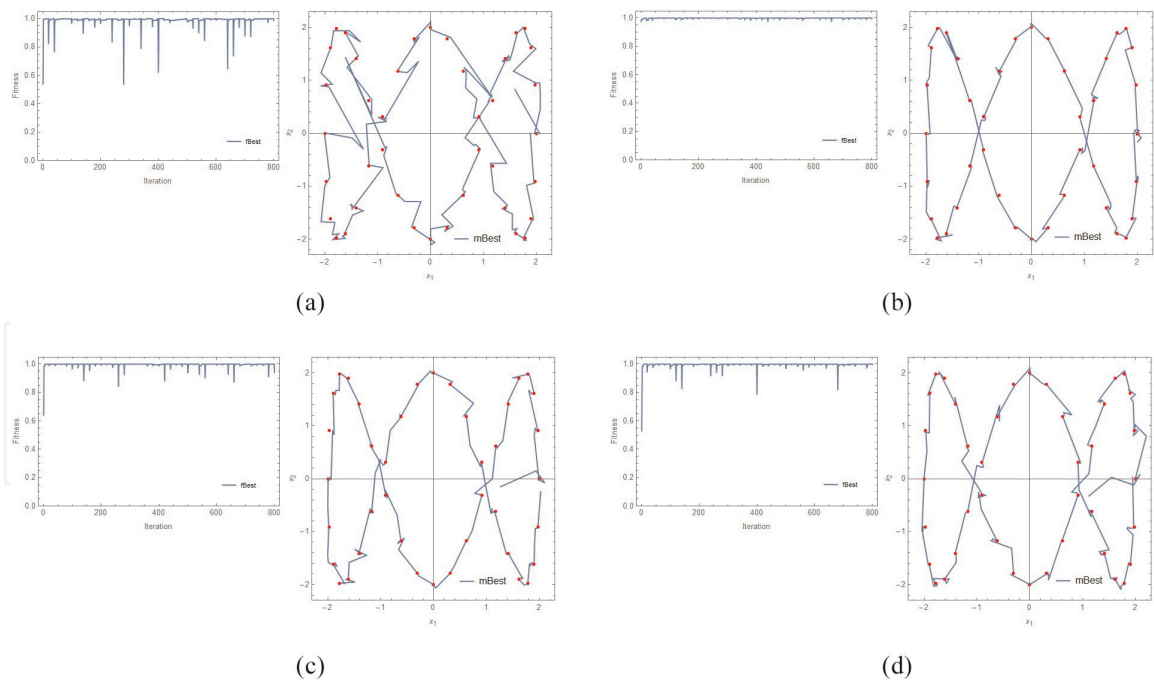
**Figure 4**, the obtained results show that MPSOIWSIS, MCPSOSIS, and HPSOSIS have good search ability and tracking performance depending on the variation patterns of the fitness values on the search space.

Next, for handling the tracking problem of variable speed II type which has a middle level of difficulty, MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS were performed, respectively. The obtained experimental results are shown in **Figure 5**.

We can see that the variation of the obtained best solution in whole search process from the left parts of **Figure 5(a)–(d)**, and the moving trajectories of



**Figure 5.** The moving trajectory of the best solution for handling the tracking problem of variable speed II type. Left part, time space; right part, search space. (a) MPSOSIS case, (b) MPSOIWSIS case, (c) MCPSOSIS case, and (d) HPSOSIS case.



**Figure 6.**  
 The moving trajectory of the best solution for handling the tracking problem of variable speed III type. Left part, time space; right part, search space. (a) MPSOSIS case, (b) MPSOIWSIS case, (c) MCPSOSIS case, and (d) HPSOSIS case.

variable speed II type are drawn almost smoothly from the right parts of **Figure 5** except for **Figure 5(a)**. Then, compared to the variation in the fitness value in the time space of **Figure 5**, it is found that the falling range of the fitness value of the best solution is slightly bigger due to the increase in difficulty of the given search problem.

Subsequently, for handling the tracking problem of variable speed III type which has a high level of difficulty, MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS were performed, respectively. **Figure 6** shows the obtained experimental results.

Similarly, we can see that the variation of search patterns in the time space of **Figure 6(a)–(d)** for handling the given tracking problem. Except for the search result of **Figure 6(a)**, the search trajectories of **Figure 6(b)–(d)** are roughly drawn. Then, compared with the variation in the fitness value in the time space of **Figure 6**, it is found that the falling variation of the fitness value of the best solution is bigger due to the increase in the difficulty of the given tracking problem.

The moving trajectories of MPSOIWSIS, MCPSOSIS, and HPSOSIS are roughly drawn. Corresponding to this situation, it is clear that the smoothness of the moving trajectory gradually deteriorated as the difficulty level of the tracking problem increased. In addition, we can see that MPSOIWSIS, MCPSOSIS, and HPSOSIS are more susceptible to target variation compared with MPSOSIS.

## 4.2 Effect of the number and sensing distance of sensors

For objectively and quantitatively evaluating the tracking ability and performance of the proposed methods, we use an indicator such as cumulative fitness (*CF*) for estimating the moving trajectory of the best solution. The *CF* is defined as the following equation:

$$CF = \frac{1}{K} \sum_{t \in T, k=1}^K g_t(\vec{s}_k) \quad (14)$$



Consequently, by changing the number  $m$  of the used sensors and changing the sensing distance  $r$ , we implemented MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS to investigate their search ability and performance, respectively.

Hereinafter, we change the number  $m$  of the used sensors and the sensing distance  $r$  and implement the proposed methods for handling the given tracking problems.

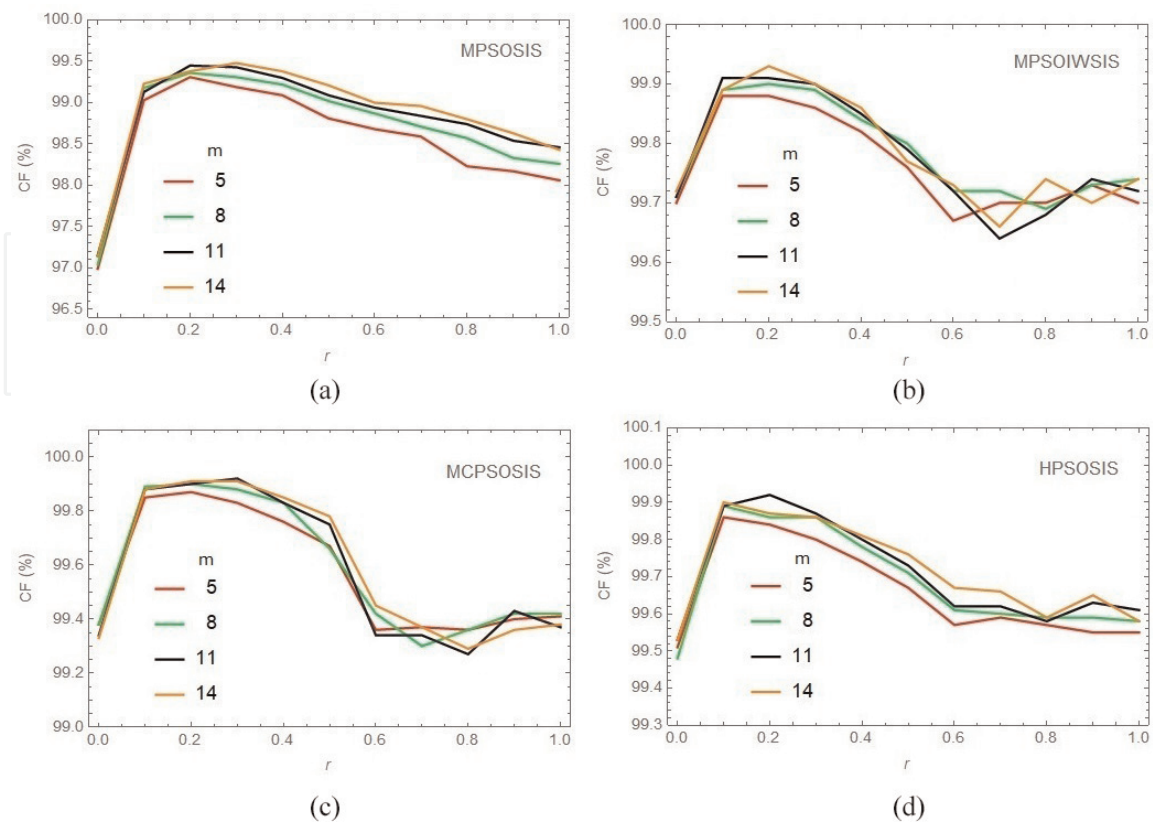
First, computer experiments were carried out to handle the tracking problem of constant tracking I type. In this case, the obtained search results (average value of running ten times) are shown in **Figure 7**.

Comparing the search results of MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS shown in **Figure 7**, it is found that the difference in tracking performance regarding the existence of sensors is very large with regard to the search ability. That is, when  $r = 0$ , they become the search results of the existing methods, that is, MPSOIS, MPSOIWIS, MCPSOIS, and HPSOIS, and the significance of the proposed methods is suggested as compared with these search methods.

On the other hand, when the sensing distance  $r$  exceeds 0.5 or more, it can be confirmed that the tracking performance of MPSOIWSIS, MCPSOSIS, and HPSOSIS becomes low and unstable. The tracking performance is relatively high within a certain range of the sensing distance  $r$  of the sensor. And when the number  $m$  of sensors exceeds 8, there is not much difference in the search ability of these methods themselves.

Second, computer experiments were carried out to handle the tracking problems of variable speed II type and variable speed III type. The obtained search results are shown in **Figures 8** and **9**, respectively.

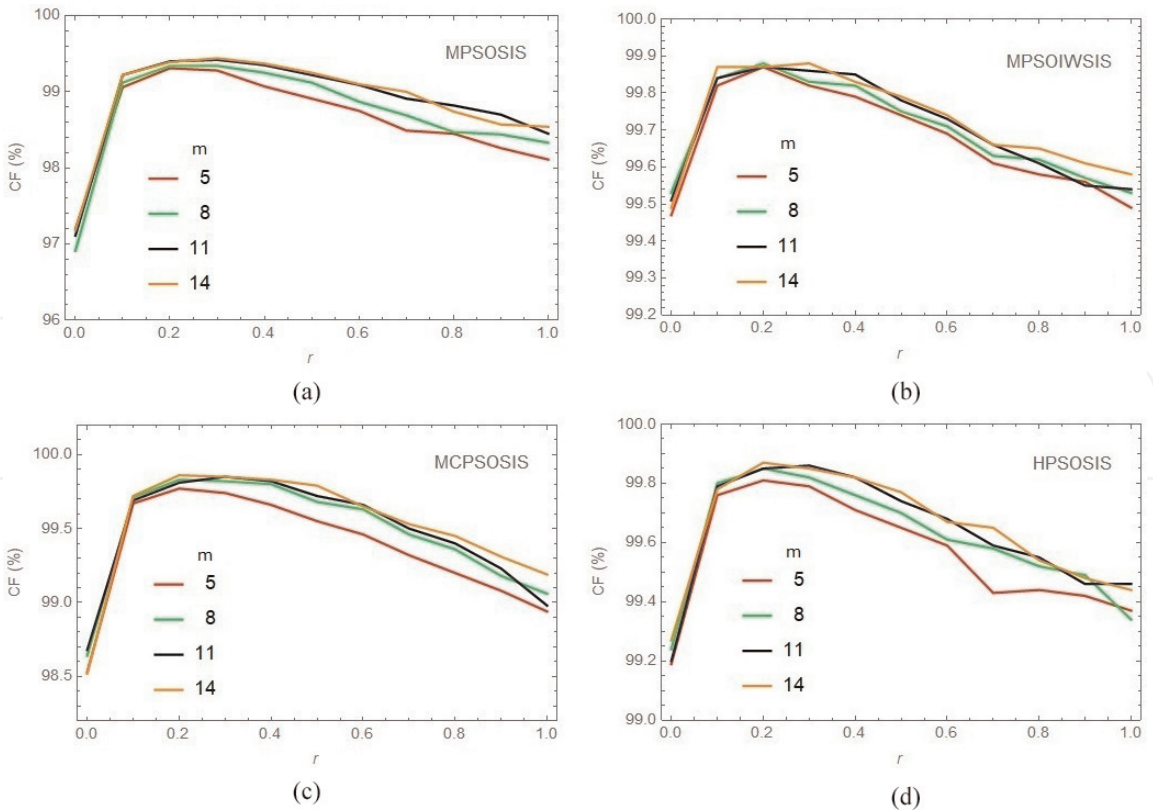
By comparing the search results shown in **Figures 7–9**, it is clear that each proposed search method has high tracking ability in each case. As the main search characteristics, we can see that as the sensing distance  $r$  of the sensor increases and



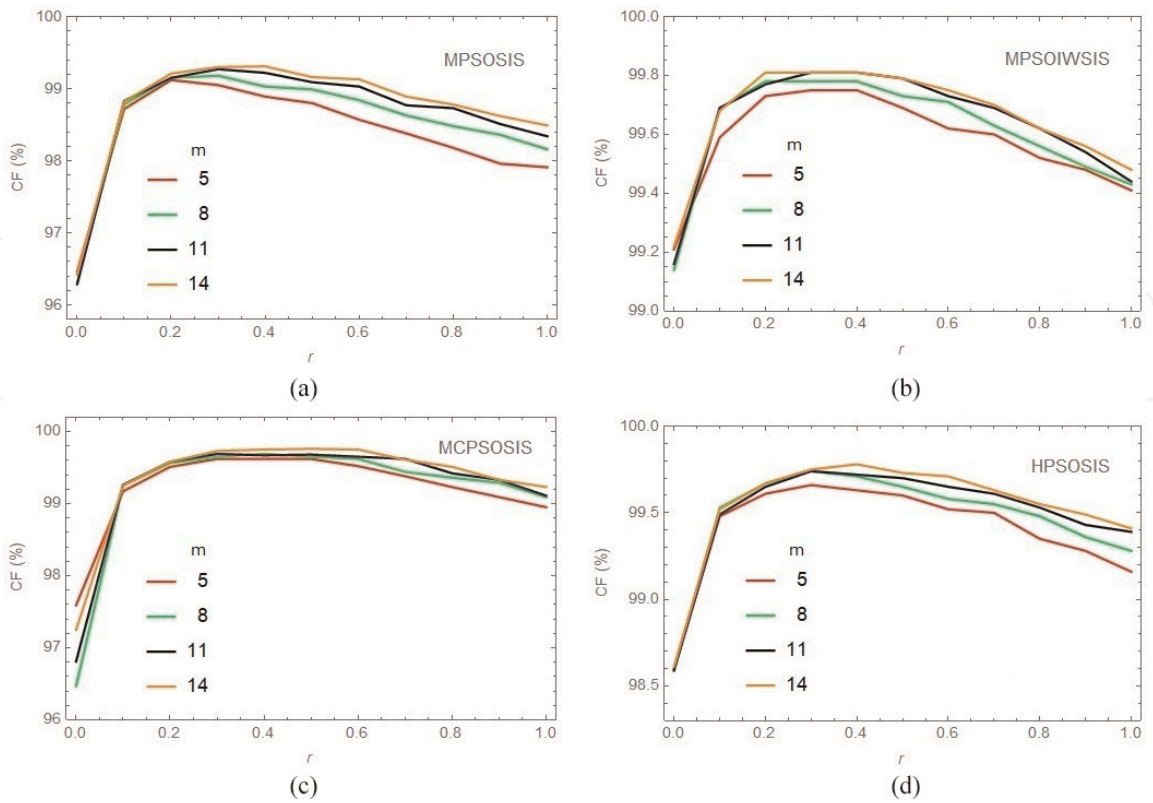
**Figure 7.**

Effect of handling the tracking problem of constant speed I type with adjustment of the number  $m$  and sensing distance  $r$  of sensors. (a) MPSOSIS case, (b) MPSOIWSIS case, (c) MCPSOSIS case, and (d) HPSOSIS case.

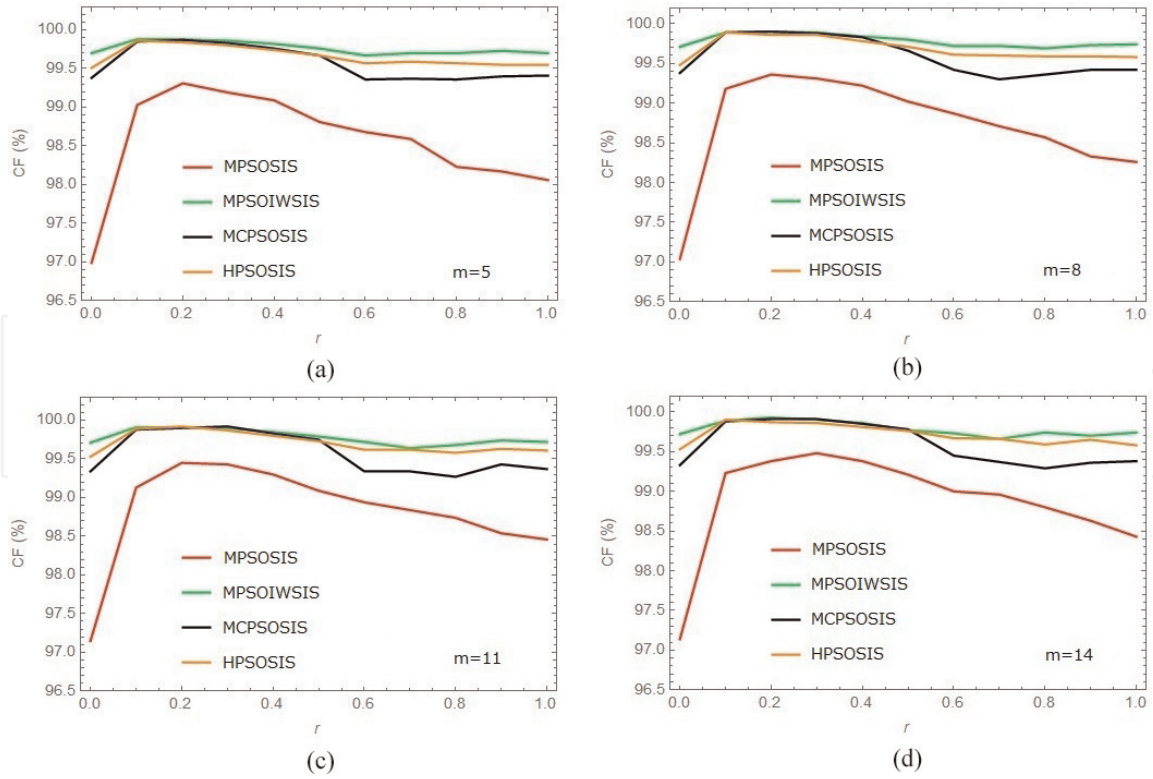




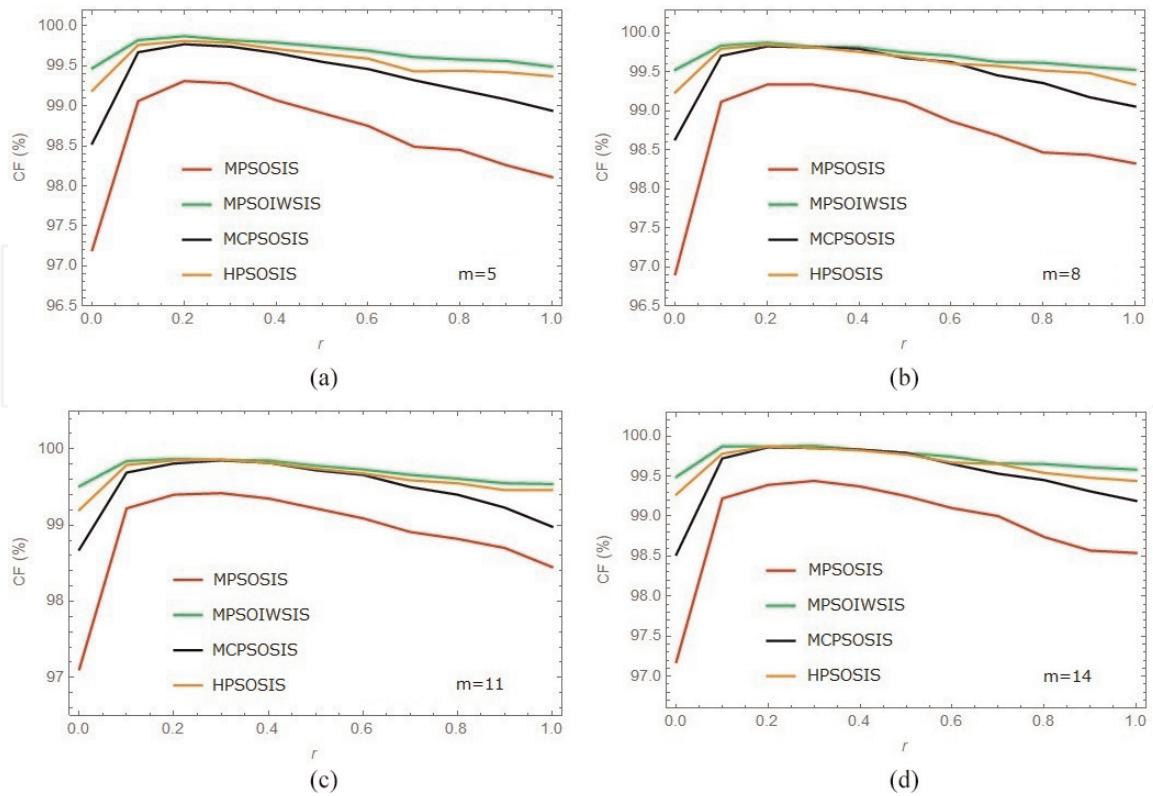
**Figure 8.**  
Effect of handling the tracking problem of variable speed II type with adjustment of the number  $m$  and sensing distance  $r$  of sensors. (a) MPSOSIS case, (b) MPSOIWSIS case, (c) MCPSOSIS case, and (d) HPSOSIS case.



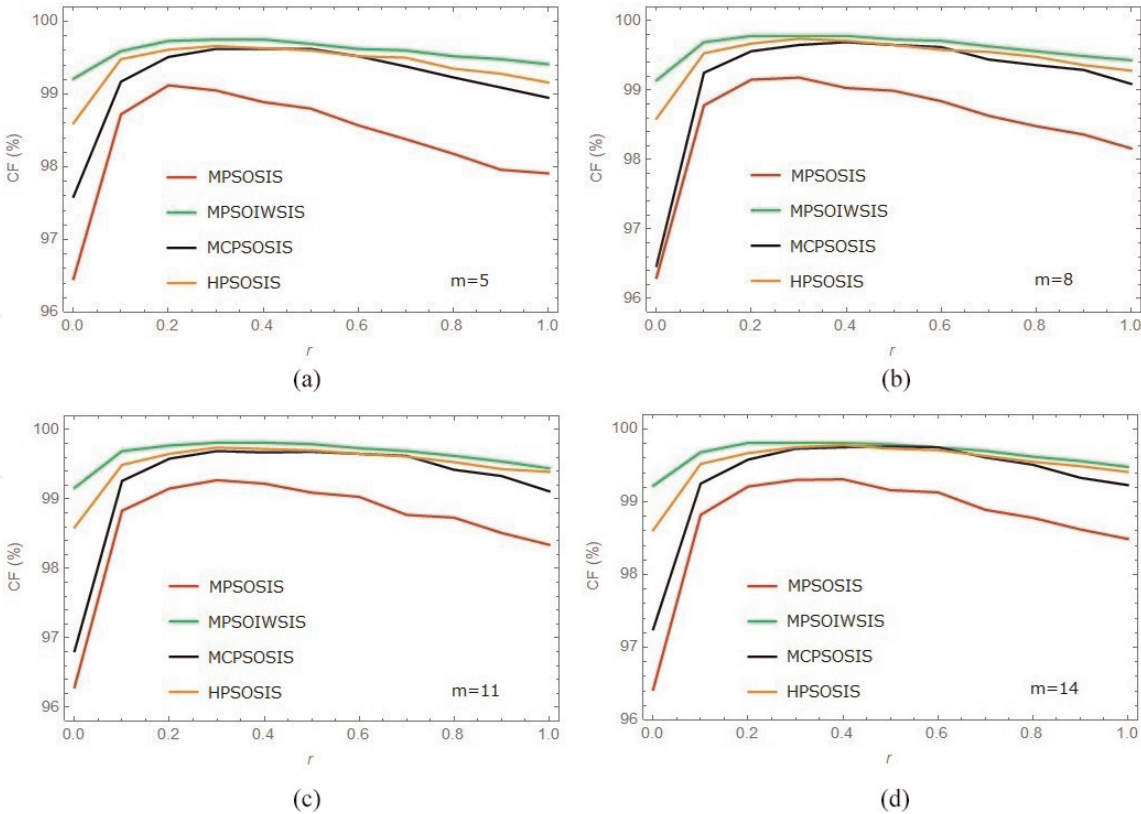
**Figure 9.**  
Effect of handling the tracking problem of variable speed III type with adjustment of the number  $m$  and sensing distance  $r$  of sensors. (a) MPSOSIS case, (b) MPSOIWSIS case, (c) MCPSOSIS case, and (d) HPSOSIS case.



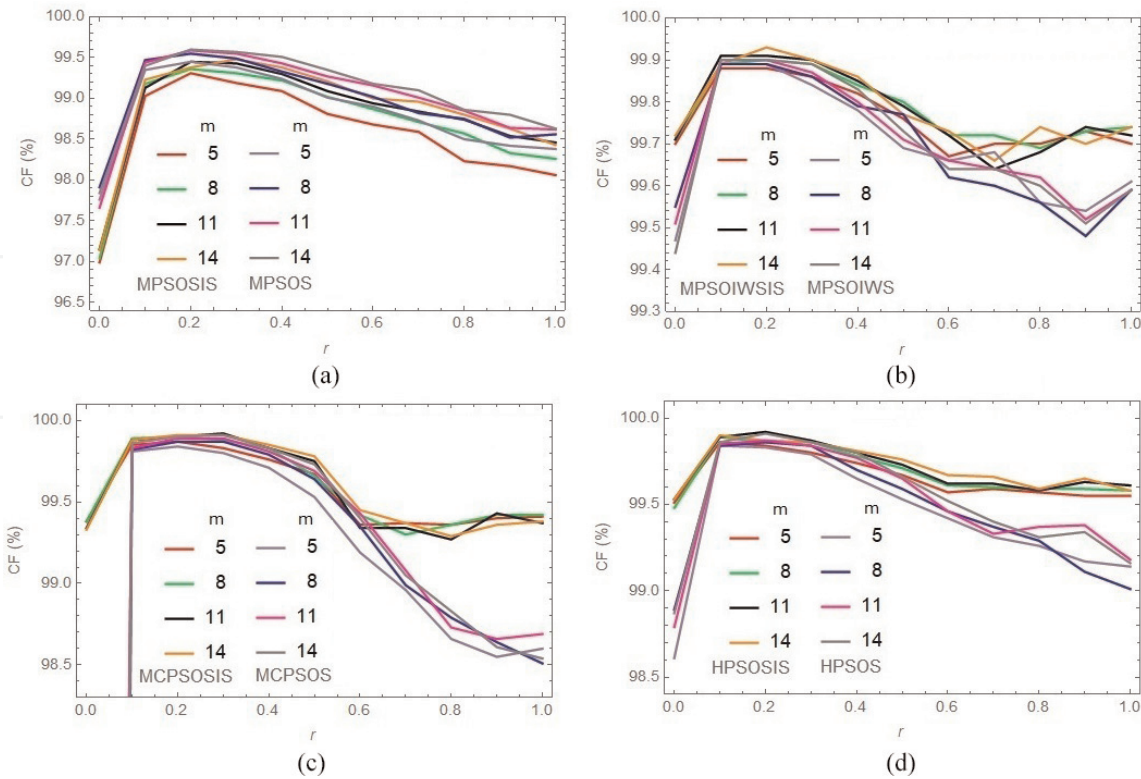
**Figure 10.** Search ability of each proposed method for handling the tracking problem of constant speed I type. (a)  $m = 5$  case, (b)  $m = 8$  case, (c)  $m = 11$  case, and (d)  $m = 14$  case.



**Figure 11.** Search ability of each proposed method for handling the tracking problem of variable speed type. (a)  $m = 5$  case, (b)  $m = 8$  case, (c)  $m = 11$  case, and (d)  $m = 14$  case.

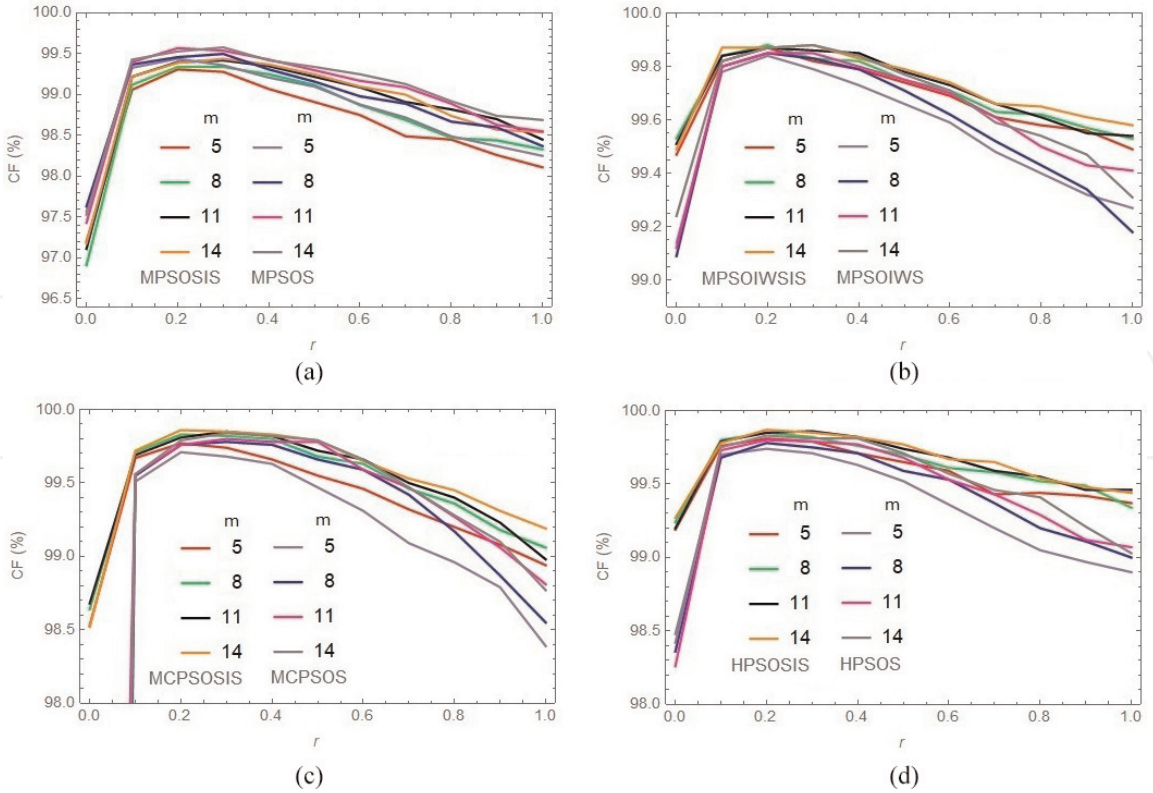


**Figure 12.**  
Search ability of each proposed method for handling the tracking problem of variable speed III type. (a)  $m = 5$  case, (b)  $m = 8$  case, (c)  $m = 11$  case, and (d)  $m = 14$  case.

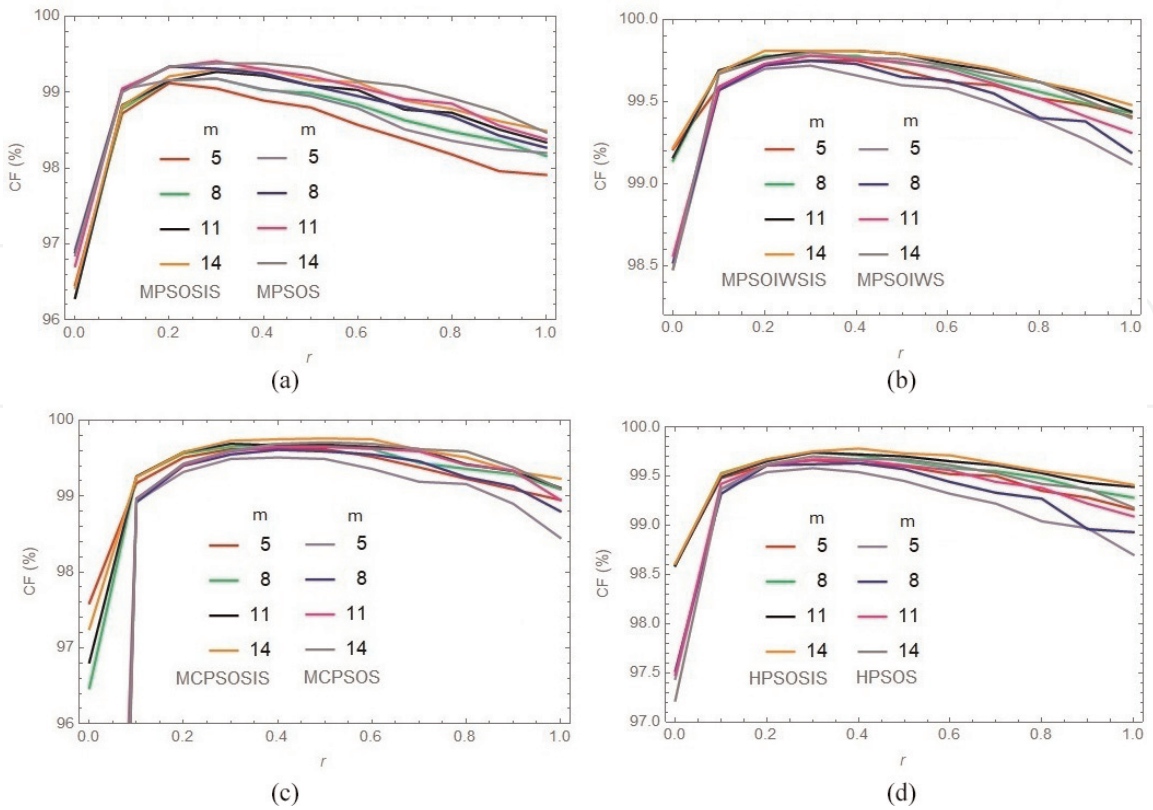


**Figure 13.**  
The best and average solutions for handling the tracking problem of constant speed I type. (a) MPSOSIS and MPSOS case, (b) MPSOIWSIS and MPSOIWS case, (c) MCPSOSIS and MCPSOS case, and (d) HPSOSIS and HPSOS case.





**Figure 14.**  
The best and average solutions for handling the tracking problem of variable speed II type. (a) MPSOSIS and MPSOS case, (b) MPSOIWSIS and MPSOIWS case, (c) MCPSOSIS and MCPSOS case, and (d) HPSOSIS and HPSOS case.



**Figure 15.**  
The best and average solutions for handling the tracking problem of variable speed III type. (a) MPSOSIS and MPSOS case, (b) MPSOIWSIS and MPSOIWS case, (c) MCPSOSIS and MCPSOS case, and (d) HPSOSIS and HPSOS case.

the fitness value of the best solution gradually increases and gradually decays after passing through a certain peak value of the *CF*.

### 4.3 Performance comparison of the proposed methods

In this section, we compare the search performance of the four proposed methods, that is, MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS, by handling the same tracking problem. **Figure 10** shows the search results obtained by handling the tracking problem of constant speed I type.

We can see clearly that the search performance of MPSOSIS is the lowest regardless of the number of sensors used. For the remaining three proposed methods, that is, MPSOIWSIS, MCPSOSIS, and HPSOSIS, it is obvious that the search performance of MCPSOSIS is good within a certain range of the sensing distance  $r$ . Overall, it is clear that the search performance of MPSOIWSIS and HPSOSIS is relatively better in search ability. As the sensing distance  $r$  increases, all of their cumulative fitness values gradually decrease.

Similarly, **Figures 11** and **12** show the search results obtained by handling the tracking problems of variable speed II type and variable speed III type, respectively. Observing the obtained search results of both, it is almost the same as the finding obtained from the data analysis of the search result in **Figure 10**. In particular, it is found that the search performance of each proposed method is very lower when sensors are not used. In this case, the proposed methods (i.e., MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS) correspond to the existing methods (i.e., MPSOIS, MPSOIWIS, MCPISOIS, and HPSOIS). Thus, the important role of the used sensors is clearly shown.

### 4.4 Performance comparison without the strategy of information sharing

In order to investigate the effectiveness of the proposed methods under the situation of multiple particle swarm search, computer experiments on the existing search methods, that is, MPSOS, MPSOIWS, MCPSOS, and HPSOS, were implemented.

For intuitive comparison of both, the obtained results are shown in **Figures 13** and **15**, respectively. When there is no sensor, it is understood that the difference between them is the largest. It is also found that the attenuation of the cumulative fitness of the latter becomes relatively fast as the sensing distance  $r$  increases.

Except for the results in **Figures 13(a)**, **14(a)**, and **15(a)**, we discovered that the search results of the proposed methods, that is, MPSOIWSIS, MCPSOSIS, and HPSOSIS, are better than the existing methods, that is, MPSOIWS, MCPSOS, and HPSOS, except for the MPSOSIS case. Therefore, the effectiveness of the information sharing strategy is confirmed even in the case of multiple particle swarm search. The obtained results in **Figures 14** and **15** show that the attenuation of the existing methods becomes faster as  $r$  increases. However, with respect to handling the three kinds of tracking problems, further investigation and confirmation are required as to why the former's search results are generally lower in the maximum value of the cumulative fitness by the latter.

## 5. Conclusions and future research

In this chapter, we proposed the four new search methods, that is, MPSOSIS, MPSOIWSIS, MCPSOSIS, and HPSOSIS, to deal with dynamic optimization problems. For investigating and comparing their tracking ability and performance, we modified the number of sensors and adjusted the sensing distance to implement



computer experiments. As the given tracking problems, we used a set of benchmark problems of constant speed I type, variable speed II type, and variable speed III type.

Computer experiments were carried out to handle each given tracking problem. Based on various experimental results obtained, the prominent search ability and performance of each proposed search method is confirmed.

Specifically, regarding search performance of the proposed methods, it is found that the obtained search results of MPSOIWSIS, MCPSOSIS, and HPSOSIS are better than the existing methods, that is, MPSOIWS, MCPSOS, HPSOS, MPSOIWIS, MCPISOIS, and HPSOIS. Also, in addition to enhancing the processing capacity for dealing with the given tracking problems, the efficiency of the search itself is also improved. However, in order to obtain good tracking ability and performance, it is necessary to select an appropriate value for the sensing distance of the sensor.

As future research subjects, based on the sensing information obtained from the sensors, we will advance the development of PMSO [22], that is, introducing the strategy of sharing information during the search and raising the intellectual level in particle multi-swarm search. Alternatively, the proposal methods utilizing the excellent tracking ability of MPSOIWSIS and HPSOSIS are applied extensively to dynamic search problems such as identification of control systems and recurrent network learning.


## Author details

Hiroshi Sho

Department of Human Intelligence Systems, Kyushu Institute of Technology, Japan

\*Address all correspondence to: zhang@brain.kyutech.ac.jp

## IntechOpen

© 2019 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

## References

- [1] Gaing ZL. A particle swarm optimization approach for optimum design of PID controller in AVR system. *IEEE Transactions on Energy Conversion*. 2004;**19**(2):384-391
- [2] Gallego N, Mocholi A, Menendez M, Barrales R. Traffic monitoring: Improving road safety using a laser scanner sensor. In: *Proceedings of Electronics, Robotic and Automotive Mechanics Conference (CERMA'09)*; Cuernavaca Morelos, Mexico. 2009. pp. 281-286
- [3] Rai K, Seksena SBL, Thakur AN. A comparative performance analysis for loss minimization of induction motor drive based on soft computing techniques. *International Journal of Applied Engineering Research*. 2018; **13**(1):210-225
- [4] Tehsin S, Rehman S, Saeed MOB, Riaz F, Hassan A, Abbas M, et al. Self-organizing hierarchical particle swarm optimization of correlation filters for object recognition. *IEEE Access*. 2017;**5**: 24495-24502. DOI: 10.1109/ACCESS.2017.2762354
- [5] Zhang Y, Wang S, Ji G. A comprehensive survey on particle swarm optimization algorithm and its applications. Hindawi Publish Corporation, *Mathematical Problems in Engineering*. 2015;**2015**:38. Article ID 931256. DOI: 10.1155/2015/931256
- [6] Alam A, Dobbie G, Koh YS, Riddle P, Rehman SU. Research on particle swarm optimization based clustering: A systematic review of literature and techniques. *Swarm and Evolutionary Computation*, Elsevier. 2014;**17**(2014): 1-13. DOI: 10.1016/j.swevo.2014.02.001 [Accessed: February 8, 2019]
- [7] Cui X, Potok TE. Distributed adaptive particle swarm optimizer in dynamic environment. In: *IEEE International Parallel and Distributed Processing Symposium*; Long Beach, CA. 2007. pp. 1-7
- [8] Rezazadeh I, Meybodi MR, Naebi A. Adaptive particle swarm optimization algorithm for dynamic environment. In: *2011 Third International Conference on Computational Intelligence, Modelling & Simulation*. 2011. pp. 120-129
- [9] Spall JC. Stochastic optimization. In: Gentle J, et al. editors. *Handbook of Computational Statistics*. Heidelberg, Germany: Springer; 2004. pp. 169-197
- [10] Xia X, Gui L, Zhan Z-H. A multi-swarm particle swarm optimization algorithm based on dynamical topology and purposeful detecting. *Applied Soft Computing*. 2018;**67**:126-140. Available from: <https://www.sciencedirect.com/science/article/pii/S1568494618301017> [Accessed: February 6, 2019]
- [11] Yu X, Estevez C. Adaptive multiswarm comprehensive learning particle swarm optimization. *Information*. 2018;**9**(173):15. DOI: 10.3390/info9070173 [Accessed: February 6, 2019]
- [12] Fogel LJ, Owen AJ, Walsh MJ. On the evolution of artificial intelligence. In: *Proceedings of the Fifth National Symposium on Human Factors in Electronics*; San Diego, CA, USA; 1964. pp. 63-76
- [13] Goldberg DE. *Genetic Algorithm in Search Optimization and Machine Learning*. Reading: Addison-Wesley; 1989
- [14] Holland H. *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI, USA: University of Michigan Press; 1975
- [15] Reyes-Sierra M, Coello CAC. Multi-objective particle swarm optimizers:

A survey of the state-of-the-art. *International Journal of Computational Intelligence Research*. 2006;2(3): 287-308

[16] Blackwell TM. Swarms in dynamic environments. In: *Proceedings of the 2003 Genetic and Evolutionary Computation Conference, LNCS*; 2003. pp. 1-12

[17] Hu X, Eberhart RC. Adaptive particle swarm optimization: Detection and response to dynamic systems. In: *Proceedings of the 2002 IEEE Congress on Evolutionary Computations, Vol. 2*; Honolulu, HI, USA; 2002. pp. 1666-1670

[18] Sho H. The search feature of particle swarm optimizer with sensors in dynamic environment. *IEICE Technical Report*. 2017;117(63):77-82. (In Japanese)

[19] Sho H. Use of multiple particle swarm optimizers with sensors on solving tracking problems. *IEICE Technical Report*. 2018;118(7):9-14. (In Japanese)

[20] Zhang H. An analysis of multiple particle swarm optimizers with inertia weight for multi-objective optimization. *IAENG International Journal of Computer Science*. 2012;39(2):10

[21] Sho H. Particle multi-swarm optimization: A proposal of multiple particle swarm optimizers with information sharing. In: *Proceedings of 2017 10th International Workshop on Computational Intelligence and Applications*; Hiroshima, Japan; 2017. pp. 109-114. DOI: 10.1109/IWCIA.2017.8203570

[22] Sho H. Expansion of particle multi-swarm optimization. *Artificial Intelligence Research*. 2018;7(2):74-85. DOI: 10.5430/air.v7n2p74

[23] del Valle Y, Venayagamoorthy GK, Mohagheghi S, Hernandez JC, Harley

RG. Particle swarm optimization: Basic concepts, variants and applications in power systems. *IEEE Transactions on Evolutionary Computation*. 2008;12(2): 171-195. DOI: 10.1109/TEVC.2007.896686

[24] Eberhart RC, Kennedy J. A new optimizer using particle swarm theory. In: *Proceedings of the sixth International Symposium on Micro Machine and Human Science*; Nagoya, Japan; 1995. pp. 39-43. DOI: 10.1109/MHS.1995.494215

[25] Kennedy J, Eberhart RC. Particle swarm optimization. In: *Proceedings 1995 IEEE International Conference on Neural Networks*; Perth, Australia; 1995. pp. 1942-1948

[26] Eberhart RC, Shi Y. Comparing inertia weights and constriction factors in particle swarm optimization. In: *Proceedings of the 2000 IEEE Congress on Evolutionary Computation. Vol. 1*; San Diego, CA; 2000. pp. 84-88. DOI: 10.1109/CEC.2000.870279

[27] Shi Y, Eberhart RC. A modified particle swarm optimizer. In: *Proceedings of the IEEE International Conference on Evolutionary Computation*; Anchorage, Alaska, USA; 1998. pp. 69-73. DOI: 10.1109/ICEC.1998.699146

[28] Clerc M, Kennedy J. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*. 2000;6(1): 58-73. DOI: 10.1109/4235.985692

[29] Clerc M. *Particle Swarm Optimization*. UK: ISTE Ltd; 2006

[30] Trelea IC. The particle swarm optimization algorithm: Convergence analysis and parameter selection. *Journal of Information Processing Letter*. 2003;85:317-325. DOI: 10.1016/

S0020-0190(02)00447-7 [Accessed:  
December 11, 2018]

[31] Innocente MS, Sienz. Particle swarm optimization with inertia weight and constriction factor. In: Proceedings of International Conference on Swarm Intelligence; Cergy, France; 2011. pp. 1-11

[32] El-Abd M, Kamel MS. A taxonomy of cooperative particle swarm optimizers. International Journal of Computational Intelligence Research. 2008;4(2):137-144

[33] Sedighizadeh D, Mashian E. Particle swarm optimization networks, taxonomy and application. International Journal of Computer Theory and Engineering. 2009;1(5):486-502. DOI: 10.7763/IJCTE.2009.V1.80

[34] Zhang H. A new expansion of cooperative particle swarm optimization. In: Proceedings of the 17th International Conference on Neural Information Processing (ICONIP2010), Part I, LNCS 6443, Neural Information Processing—Theory and Algorithms; Sydney, Australia; 2010. pp. 593-600