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

**6,900**
Open access books available

**185,000**
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

CLARIVATE ANALYTICS

**BOOK CITATION INDEX**

INDEXED

**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

# Synchronous and Asynchronous Communication Modes for Swarm Robotic Search*

Songdong Xue[1], Jin Li[1], Jianchao Zeng[1], Xiaojuan He[2] and Guoyou Zhang[2]
*[1]Complex System and Computational Intelligence Laboratory,
Taiyuan University of Science and Technology, 66 Waliu Road, Taiyuan 030024, Shanxi
[2]College of Electrical and Information Engineering,
Lanzhou University of Technology, 85 Langongping, Lanzhou 730050, Gansu
China*

## 1. Introduction

Swarm robots are special multi-robots and usually considered being controlled with swarm intelligence-based method to complete some assigned complex tasks (Dorigo and Sahin, 2004). Similar to the biological counterparts in nature, swarm intelligence among such artificial system is emerged from local interactions between individual robots or individual robot and its environment (Beni, 2005; Şahin, 2005). It is obvious that interactions play a crucial role in emergence of swarm intelligence in swarm robotics (Schmickl and Crailsheim, 2008). In other words, communication mode taken in control process of swarm robotic search is important. How to control swarm robots with certain communication mode? We can borrow ideas from swarm intelligence-based optimization algorithms in general, and the particle swarm optimization (PSO) algorithm in particular, since the case of swarm robotic search can be mapped to the case of functions optimization with PSO. Later, this method is named as the extended particle swarm optimization (EPSO) method (Pugh and Martinoli, 2007). The particle swarm optimization algorithm is a global, stochastic search one, being derivative-free and population-based style (Schutte *et al.*, 2004). As one of tools of systemic modeling and cooperative control, it can be used to model swam robotic systems and control robots cooperatively. Bio-inspiringly, this algorithm works in parallel in nature. Learning from this, we can control swarm robotic search with special communication modes in similar way. As for the parallel algorithms, they can be classified by granularity (Xu and Zeng, 2005). Wang *et al.* (2007) present a parallel version of PSO based on parallel model with controller. Its communication cycle affects speedup of the algorithm. Huang and Fan (2006) propose parallel version of PSO by island population modeling. It partitions the group into several sub-groups and places them on different processors to evolve, communicating timely in the evolution procedures. Zhao *et al.* (2005) introduce an idea of migration into PSO, present a parallel version based on multi-groups evolving simultaneously. All sub-groups are collected to get the optima by comparison after several iterations. Then the particle having best

fitness is transferred into all sub-groups as a migration. These mentioned algorithms are all attributed to coarse-grained parallelism. To the contrary, the fine-grained parallel algorithms are characteristic with majority advantages, i.e., maintaining diversity of groups, restraining pre-mature and holding the highest degree of parallelism. Therefore, Schutte *et al.* (2004) develop an approach to implement parallel PSO on multi-processors environment. Especially, to overcome the communication bottle-neck due to massively increasing in size of fine-grained PSO (Li *et al.*, 2006), Chang *et al.* (2005) design three types of communication strategies according to the degree of correlation between parameters. The above parallel versions of PSO are all synchronous paradigms. However, Koh *et al.* (2006) point out that asynchronous algorithm can increase efficiency in heterogeneous environment. Typically, the asynchronous PSO version proposed by Luo and Zhong (2005) makes each particle acted as an independent individual and search performed asynchronously. Aiming at the differences of heterogeneous computing environments and the cost of fitness evaluate, Venter and Sobieszczanki-Sobieksi (2006) introduce asynchronous pattern into PSO for speedup enhancement.

As mentioned above, swarm robots controlling inevitably involves parallel operation too (Henrich and Honiger, 1997). No doubt the extended PSO approach has to take parallelism into account in control algorithms designing. The individual robots distributing in search space makes cooperation control algorithms parallel in nature. Besides, differences in sampling frequency of sensors carried by robots and communication delays make it more realistic to control swarm robots in an asynchronous fashion. How to move nature-inspired algorithms to parallel, asynchronous and decentralized environment (Ridge *et al.*, 2005)? There has so far been fairly little research in this area. For this end, within the field of swarm robotic systems, one area that has received more attention is target search, where a group of robots work together to localize one or more targets. As a first step, a single target is considered here. Searching can be done massively in parallel, significantly decreasing the time-consuming to locate the targets and improving robustness against failure of single agents by redundancy as well as individual simplicity. Then, the problem of parallel asynchronous control swarm robots, i.e., swarm robotic search with different specific communication modes is proposed. As for this chapter, the remainder is organized as follows. Section 2 maps swarm robotic search to the particle swarm optimization algorithm. Then it models the swarm robots with EPSO method and describes the control following swarm intelligence principles. For taking the control mechanism effect, several definitions and assumptions are given for problem simplification without misunderstanding. In Section 3, the problem of communication modes in swarm robotic search is introduced. In order to describe the corresponding strategies developed and taken in swarm robotic search taken place in obstacle-free environment steadily, we set about this section from analyzing the properties of different versions of PSO because of the mapping relationship between swarm robotic search and PSO. Then the synchronous and asynchronous communication modes are discussed. Also, the corresponding control algorithm descriptions with specific communication modes are given here. Based on this, the authors explain the simulation settings, propose evaluate metrics such as searching efficiency and energy consuming, show the results from simulations, and discuss the implications through statistical evidence presentation in Section 4. Finally, we conclude this chapter in Section 5 by summarizing our current work and accompanying the future work.

## 2. Modeling and controlling

By extending PSO to model swarm robotic systems, Pugh and Martinoli (2007) firstly and Zeng and Xue (2010) subsequently investigate the problem of target search in an ideal

environment. In PSO-type swarm robotic search, modeling and control mechanism are involved inevitably.

### 2.1 Mapping from swarm robotic search to PSO

Swarm intelligence is inspired from phenomena of individuals following simple rules only but emerging intelligence through local interactions in biological communities. Generally speaking, PSO is viewed as optimization tool, in which particles are guided by the best positions having optimal fitness to get the solution of given functions. Here, each particle has perfect knowledge about environment and its neighbor particles. While swarm robotic search works depending on individual robots' experience and social experience, for robots move according to their own behavioral decision making. The former comes from signals measurement by robot itself, and the latter from local communications within robot's communication neighborhood. Consider the similarities and differences between the two cases, we can map swarm robotic search to PSO, see Figure 1.
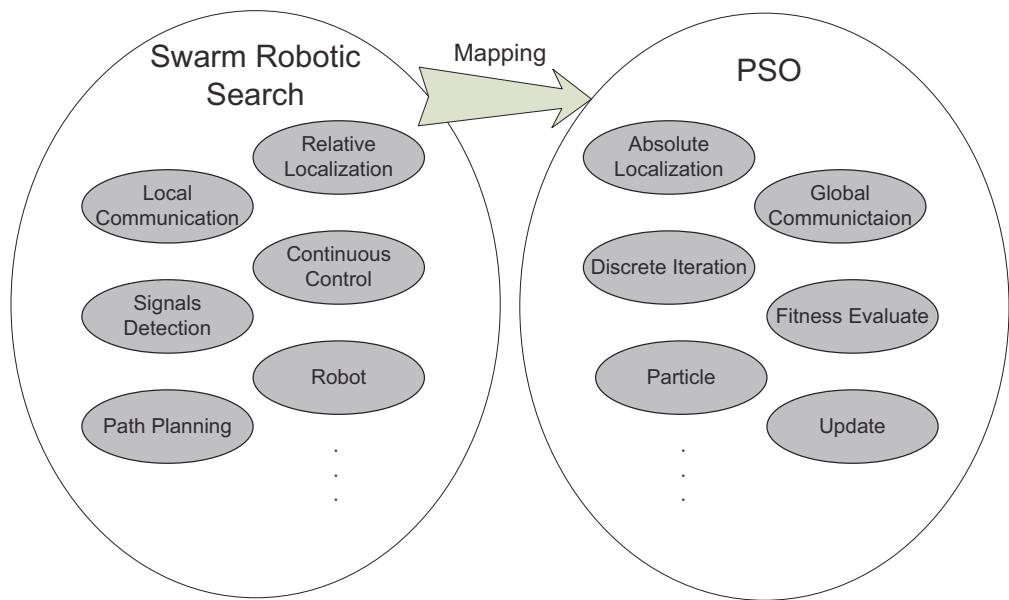


Fig. 1. Mapping swarm robotic search to the particle swarm optimization algorithm. Note that several one-to-one relationships of robot-to-particle, signals detection-to-fitness evaluate and local communication-to-global communication are the most important.

The PSO algorithm makes information about environment or potential solution shared among particles anywhere, as long as the particles are in the search space. Similarly, robots always have strict limitations on their maximum communication range due to the limited power consuming. In this case, we define robot's neighborhood structure as all others within some fixed geometrical distance from it. Since robots are constantly in motion, this means such structure is dynamic and time varying (Xue and Zeng, 2008a).

### 2.2 EPSO-based modeling

Based on the above mapping relationship between swarm robotic search and PSO, EPSO method can be taken to model the swarm robotic system (Pugh and Martinoli, 2007; Xue and Zeng, 2008b). In order to understand such method well, let us examine the particle swarm optimization algorithm at first. Particle swarm optimization is based on the sociological behaviors associated with bird flocking and other animals' moving (Zeng *et al.*, 2004). Each particle is capacitated to fly over the space with changeable velocity. And a series of positions

in which particles are situated are viewed as potential solutions of problem. Then, the best position of particle itself and swarm respectively having the best fitness can be decided. Farther, the behavior of particle can be adjusted according to its inertia, individual experience (cognition) and social experience (learning). The velocity and position update equations of standard PSO at time $kt + 1$ are executed as follows:

$$\mathbf{v}_{k+1}^i = \omega_k \mathbf{v}_k^i + c_1 r_1 (\mathbf{p}_k^i - \mathbf{x}_k^i) + c_2 r_2 (\mathbf{p}_k^g - \mathbf{x}_k^i) \tag{1}$$

$$\mathbf{x}_{k+1}^i = \mathbf{x}_k^i + \mathbf{v}_{k+1}^i \tag{2}$$

where $\mathbf{x}_k^i$ is the position vector of particle $i$ at time $kt$, and $\mathbf{v}_k^i$ the corresponding vector of velocity, subscript $k$ the abbreviation of time increment $kt$. Note that the two vectors have the same dimensional variables. While $\mathbf{p}_k^i$ and $\mathbf{p}_k^g$ are the best-found positions of particle $i$ itself and the swarm before time $k$ respectively. The coefficient matrix $\omega_k$ is diagonal matrix whose diagonal elements are inertia coefficients with value range [0, 1] to slow down over time to prevent explosions of the swarm and ensure ultimate convergence. Similarly, $r_1$, $r_2$ are diagonal matrices whose diagonal elements are sampling of uniformly-distributed random variable in [0, 1]. And $c_1$, $c_2$ are diagonal matrices whose diagonal elements are cognition and social acceleration constants, respectively.

Then we can extend the standard version of PSO for swarm robots and farther model the swarm robotic system with the EPSO method, as is shown below:

$$\mathbf{v}_{k+1}^i = w_k \mathbf{v}_k^i + c_1 r_1 (\mathbf{p}_k^i - \mathbf{x}_k^i) + c_2 r_2 (\mathbf{p}_k^g - \mathbf{x}_k^i) \tag{3}$$

$$\mathbf{v}_{k+\Delta k}^i = \mathbf{v}_k^i + (\mathbf{v}_{k+1}^i - \mathbf{v}_k^i)\Delta k \tag{4}$$

$$\mathbf{x}_{k+\Delta k}^i = \mathbf{x}_k^i + \Delta k \mathbf{v}_{k+\Delta k}^i \tag{5}$$

where $\mathbf{x}_{k+1}^i$ is the expected velocity vector of robot $i$ at time $k + 1$. $\Delta k$ is a factor to decrease the step taken when robots move about in the search space. By the way, we add the $\Delta k$ factor in order to make individual robots moved "smoothly" , and therefore a more refined search may be carried out. In addition, the parameter $\Delta k$ is somehow different from the others, as it is not related to the physical nature of the problem. However, we can also understand it in this fashion: robot in real world has inertia due to its mass (Xue and Zeng, 2008b).

### 2.3 PSO-type controlling

As is shown in the above subsection, swarm robots can be modeled mathematically taking the form of PSO-type iteration equations. Therefore, the cooperative control over swarm robots can be carried out following swarm intelligence principles. Farther comparison of the two cases can be made. First, both work on the base of fitness evaluate, or signals detection. While the relative independency of individual robots demands fitness evaluate being complemented in their on-board processors rather than in processing center. And the limitations on hardware and power supply make it impossible that robots interact successfully beyond the maximum communication range of robots. Apparently, the swarm that each robot dwells in differs from others, since every robot selects itself and all other robots within some distance in the search space as its evolving swarm.

### 2.3.1 Time-varying character swarm

Each individual robot of the swarm system selects the close near neighbors as its temporary swarm members only because those neighbors within its maximum communication range are capable of interacting with it through communication. Accordingly, a concept of time-varying character swarm (TVCS) for computational evolution is presented naturally, see Figure 2. Take the position of robot $i$ at time $t$ as the center, the maximum communication range $R$ of robot $i$ as radius for a circle neighborhood constructing. The set of those robots covered by this neighborhood is named as TVCS of robot $i$ at time instance. The size of TVCS depends on the maximum communication range of robot $i$ and the relative position relationship of robots at time $t$. This implies the property of time-varying in character swarm of swarm robotic system, since a robot may be close to few or many other robots at different time instance.
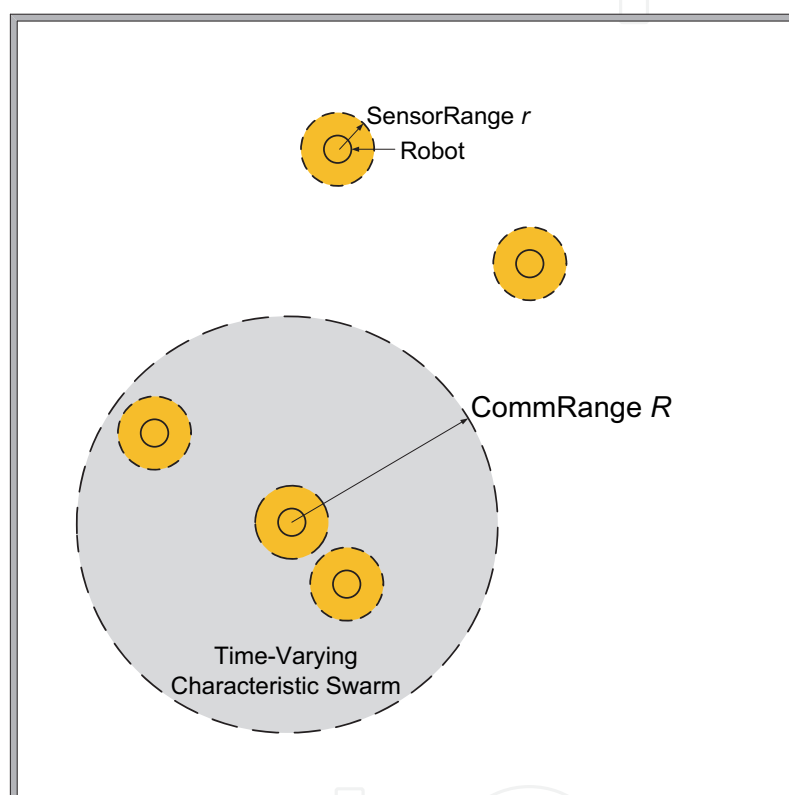


Fig. 2. Individual robot's time-varying character swarm constructing for swarm robots control following swarm intelligence principles according to the involved robot's maximum communication radius, which means this robot can interact with others within its TVCS only.

### 2.3.2 Signals detection

In PSO-type algorithms, motion control of individual robot depends on both its cognitive position and the best-found social position. While the two best-found experiential positions come from position evaluate. Each individual robot of the swarm is assumed to be equipped with one sensor to detect the intensity of signal emitted from potential target. This is of theoretical significance only. In fact, there are multiple types of signals in searching environment. To accomplish the search task, there is need to appropriately fuse the real-time heterogeneous signals with fusion algorithms to determine the decision sensor under different sensory conditions (Xue and Zeng, 2008c), following the working principle of swarm intelligence-based method. However, we still simplify the detection process as making

measurement with a single sensor, for exploring the key effects caused by communication modes here. We simply this evaluate mechanism here by assuming each robot has a sensor to detect the intensity of the target signal within its maximum detection radius. This intensity $\mathrm{I}(d_i)$ is determined with model below:

$$\mathrm{I}(d_i) = \begin{cases} 0, & d_i > r \\ \frac{P}{d_i^2} + \eta(), & \text{otherwise} \end{cases} \tag{6}$$

where $P$ is the target signal power, $d_i$ the distance from robot $R_i$ to target, $r$ the radius of sensor detection and $\eta()$ a sampling of additive Gaussian noise.

### 2.3.3 Position evaluate and cognitive decision

As to individual robot $R_i$, its cognitive position at time $t$ is determined following the rule:

$$p_i^*(t) = \begin{cases} x_i(t), & \text{if } \mathrm{I}(x_i(t) \geq \mathrm{I}(p_i^*(t-1)) \\ p_i^*(t-1), & \text{otherwise} \end{cases} \tag{7}$$

where $p_i^*(t)$ is the cognitive position of robot $R_i$ at time $t$, $x_i(t)$ the current position, and $\mathrm{I}()$ the simplified evaluate function of measurement readings of target signals.

### 2.3.4 Best-found position in TVCS

Based on the definitions of TVCS and signals evaluate, the best-found position in swarm robots can be decided with the criterion:

$$p_{(i)}^*(t) = p_k^*(t), \arg_k \max\{\mathrm{I}(p_k^*(t)), k \in R_i\text{'s TVCS}(t)\} \tag{8}$$

where $p_{(i)}^*(t)$ be the best-found position within the TVCS of robot $R_i$ at time $t$.

## 3. Communication modes

Now swarm robots can be controlled with EPSO-based method. But a problem should be considered. In PSO-type control, target signals have to be detected in parallel for position evaluate and such swarm system should be controlled in an asynchronous manner. Therefore, we present asynchronous communication mode in case of target search. Specifically, each robot independently detects signals emitted from target in a fine-grained parallel way and compares intensity of signals with the best in its TVCS. Then velocities and positions of individual robots are updated immediately. But the shared information within TVCS is updated asynchronously. As comparison, a synchronous mode is also given in this section.
We set about this problem at the beginning of analysis on the characteristics of PSO. The standard PSO algorithm has a key idea about velocity and position of particle (Eberhart and Shi, 2001; Kennedy and Eberhart, 1995; Zeng *et al.*, 2004), which is used to optimize nonlinear functions at the beginning of development and is extended to more applications gradually. The algorithm tries to find potential solutions of problem by imitating behaviors of social creature, e.g., birds flying over space. Taking fitness of given function as evaluate metrics, this algorithm adjusts the velocities and positions of particles representing solutions of problem to obtain optimum eventually. PSO is based on possessing many desirable properties that we would like to transfer to our PSO-type swarm robotic systems. One of them is that PSO operates in parallel and asynchronously (Ridge *et al.*, 2005), which is consistent with the biological significance of swarm algorithm. Thus we proceed with the analysis on characteristics of different versions of PSO.

### 3.1 Synchronous v.s. asynchronous

To explore characteristic of different versions of PSO, we can start in accordance with two issues, i.e., fitness evaluate in a serial or parallel way, synchronous or asynchronous communication mode of sensing and reacting to environment as well as velocity- and position-evolution. Therefore, we divide the different versions of PSO into four patterns.

### 3.1.1 Serial evaluate and synchronous update

Particle swarm optimization is traditionally considered to be implemented in serial and synchronous on single-processor computing environment. The execution procedure can be described with the following pseudo code (Koh *et al.*, 2006), see Algorithm 1. It can be seen that fitness evaluate of all particles is carried out one by one in optimization process through cost function computation. And the best positions both of particle itself (cognitive) and in its TVCS are determined by fitness comparison in the same way. Then the update of all velocities as well as positions occurs simultaneously at each iteration.

---

**Algorithm 1** PSO with characteristics of serial evaluate and synchronous update.

---

1: initialize algorithm constants
2: initialize all particle velocities, positions
3:       **For** $k = 1$, number of iterations
4:             **For** $i = 1$, number of particles
5:                   evaluate cost function
6:             **End**
7:       check convergence
8:       update $\mathbf{p}_k^i$ , $\mathbf{p}_k^g$ , $\mathbf{v}_{k+1}^i$ , $\mathbf{x}_{k+1}^i$
9:       **End**
10: output results

---

### 3.1.2 Serial evaluate and asynchronous update

Immediately updates on velocity, position of certain particle as well as its history cognition and the best of swarm are carried out as soon as completing evaluate on the cost function of this particle. The procedure can be elaborated with the following pseudo code (Koh *et al.*, 2006), see Algorithm 2. It is clear that the evaluate and update process on different particles are not completed at the same time.

---

**Algorithm 2** PSO with characteristics of serial evaluate and asynchronous update.

---

1: initialize algorithm constants
2: initialize all particle velocities, positions
3:       **For** $k = 1$, number of iterations;
4:             **For** $i = 1$, number of particles
5:                   evaluate cost function
6:                   check convergence
7:                   update $\mathbf{p}_k^i$, $\mathbf{p}_k^g$, $\mathbf{v}_{k+1}^i$, $\mathbf{x}_{k+1}^i$
8:             **End**
9:       **End**
10: output results

---

### 3.1.3 Parallel evaluate and synchronous update

The most obvious PSO parallel implementation is to simplify fitness evaluate for particles at iteration in parallel, without changing the overall logic of the algorithm itself (Venter and Sobieszczanki-Sobieksi, 2006). And the property of synchronous refers to all particles being sent to parallel computing environment and moving from the current iteration to the next only if the fitness of all particles has been gotten (Schutte *et al.*, 2004). To demonstrate the internal relationship of logic better, we illustrate with flow chart rather than pseudo code, as showed in Figure 3 (Koh *et al.*, 2006). In this case, the existence of load imbalance in computing environment may significantly affect parallel performance. These factors are shown below:
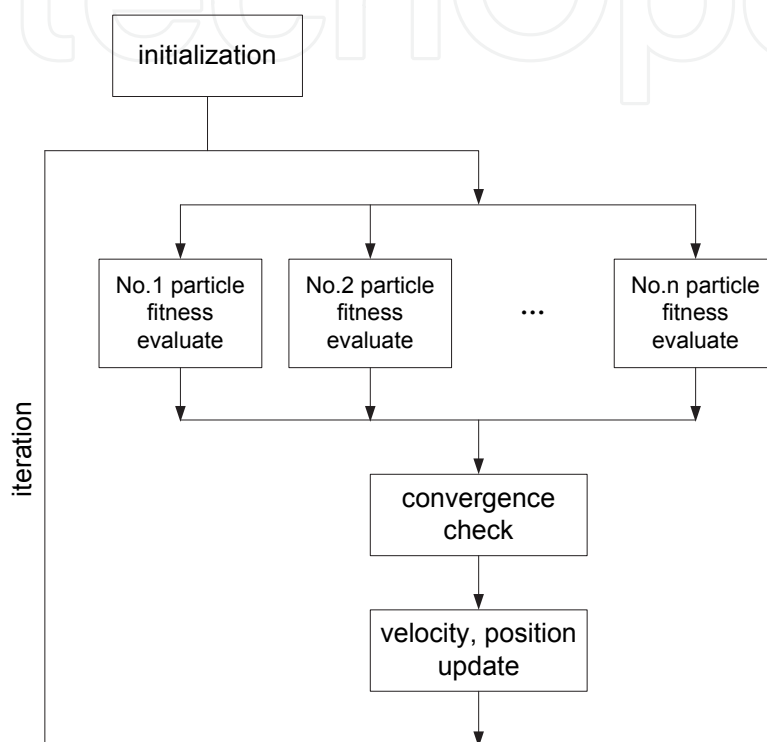


Fig. 3. Visual sketch for a version of PSO having characteristics of parallel evaluate on fitness and synchronous update for velocity and position.

- a heterogeneous distributed computing environment where processors with varying computational speed are combined into a parallel computing environment;

- time spent in fitness evaluate, i.e., using a numerical simulation to evaluate each particle, where the required simulation time depends on the particle being analyzed;

- the number of particles cannot be equally distributed among the processors in the computing environment, i.e., having a swarm size that is not an integer multiple of the number of processors (Koh *et al.*, 2006).

### 3.1.4 Parallel evaluate and asynchronous update

Parallel implementations being asynchronous in PSO can make the algorithmic computation efficiency enhanced (Venter and Sobieszczanki-Sobieksi, 2006). The asynchronous approach does not need a synchronous point to determine new velocities and positions, as showed in Figure 4 (Koh *et al.*, 2006). The optimization can proceed to the next iteration without waiting for the completion of all functions evaluate from the current iteration.
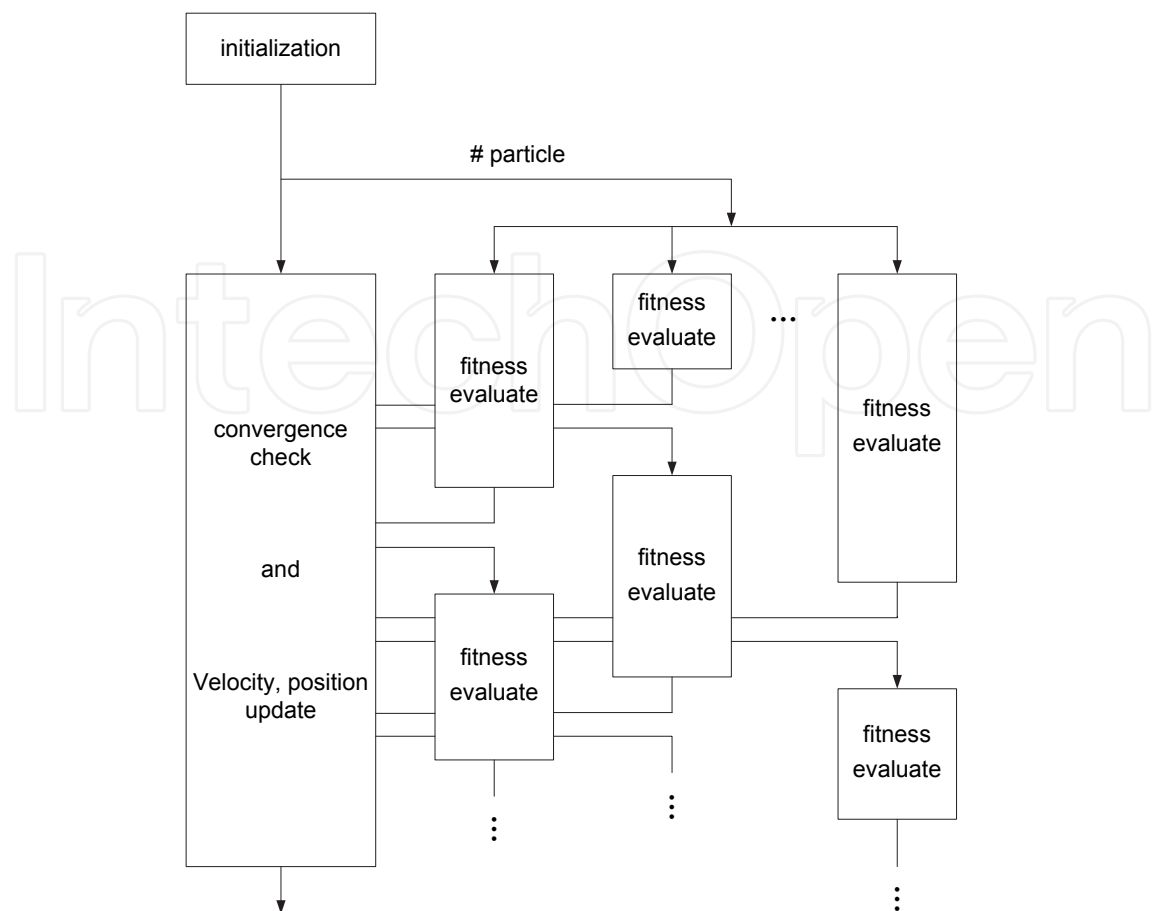
Fig. 4. Visual sketch for a version of PSO having characteristics of parallel evaluate on fitness and asynchronous update for velocity and position.

As stated above, different versions of PSO have different algorithmic properties when implementation. But the most desirable properties that we would like to transfer to our swarm robotic system may be controlled in parallel and asynchronously (Ridge *et al.*, 2005). Indeed, as one of nature-inspired algorithms, parallel and asynchronous version of PSO makes algorithm more efficient in execution.

### 3.2 Communication modes taken in swarm robotic search

Now, let us examine the case of swarm robotic search in a closed obstacle-free environment. According to the analysis above, controlling over robots should be done in a fine-grained way, as individual robots detect target signals independently at the same time to determine the best-found position by signals intensity comparison with their respective TVCS neighbors (Xue and Zeng, 2008b). Due to heterogeneous hardware caused by parameter distribution of sensors, difference of detection and evaluate time required among different positions because of signals diffusing, part of swarm robots completing signals detection early have to wait for synchronous update. The reason is that the update depends on the slowest robot (Schutte *et al.*, 2004). By this means, velocities and positions of all robots are updated at the same time after evaluate fully completing.

Here, asynchronous communication mode refers to that each robot compares at once with the optimal value of the swarm after iterating in the iteration process, if their detective signals are discovered stronger, updates immediately the optimal value of the swarm, thereby, other

robots can share the experience timely, without having to wait until given some synchronous moment, then realizes the non-synchronization of the robots in the search process.

### 3.2.1 Communication trigger

Clearly, the key to asynchronous implementation of control algorithm is to partition the individual from the group update behavior to take the different property into account. These update behaviors include updating the individual robot and the shared information within the swarm histories. Similarly, as for the asynchronous particle swarm optimization, the update action starts after fitness evaluate, while the update on swarm starts in the last at each iteration (Venter and Sobieszczanki-Sobieksi, 2006). For the target search with swarm robots, detection of target signals depends only on their respective on-board processors rather than processing center. In other words, such center does not exist in swarm robotic system. The processors work independently and in parallel between one another. The individual robot updates its velocity, position and history as soon as it completes target signals measurement and makes decision on the best-found by comparison with the best of its TVCS (Zhao $et al.$, 2005). But the update on the shared information should start in accordance with some special asynchronous control strategy. In fact, this is the decision on communication triggers. Differing from the ideal particles in PSO, robot possesses mass in real world that causes it to have inertia when moves about in the search environment. Therefore, as for same an evolution position of certain particle, it is unlimited to reach at any speed in PSO case, while robot may arrive at the same position in several sampling times due to constraint of kinematics and dynamics because the evolution position is only expected (Pugh and Martinoli, 2007). These factors should be taken consideration when we design the asynchronous interaction strategies. Based on this, some update strategies have been developed. One is communication cycle-based control principle. Here, communication cycle is named as evolution iterations. Similar to the coarse-grained parallel particle swarm optimization, we can make robot $R_i$ communicate every $n$ iterations to decide the best-found position within robot $R_i$'s TVCS (Huang and Fan, 2006; Wang $et al.$, 2007; Zhao $et al.$, 2005). To improve systematic efficiency, a communication cycle can be assigned to several fixed times of sampling periods (Xue $et al.$, 2009). Besides, different robots can be allowed to have different sampling frequencies. On the other hand, the best-found fitness value and position of TVCS should be remembered in memory before the next iteration starts. Another update strategy is evolution position-based control principle (Xue and Zeng, 2009). According to this control principle, update of the shared information does not been carried out in the current iteration before the previous evolution position has not been reached. That is to say, robots communicate when they arrive at the decisive expected or desired evolution positions regardless of the iteration history and the next iteration required. No communication between two consecutive ideal evolution positions makes robot moving continuously, saving power and decreasing communication time-consuming.

### 3.2.2 Synchronous case in swarm robotic search

The difference between synchronous and asynchronous control lines in their update types and opportunities (Zhao $et al.$, 2005). As for the synchronous mode, update time points depend on the last particle completing fitness evaluate at each step. Thus the communication triggers do not need to consider in synchronous mode. As comparison, refers to the updates of all the robots being synchronous at same iterative procedure, they are aiming at the optimal value in the current iteration step, after all the updates are completed, all the robots proceed towards the goal synchronously, and accomplish the search task with common integral cognitive level.

### 3.3 Algorithm description

The corresponding algorithms with different communication modes can be listed in the following. Of them, the synchronous communication version is taken according to the characteristics of processing moments on signals detection, search completion judging as well as velocity and position updating, see Algorithm 3. Different from the synchronous communication mode, the moment that robot updates shared information of TVCS is more flexible in asynchronous communication mode. The details of corresponding algorithm can be found in Algorithm 4.

---

**Algorithm 3** Controlling robot $R_i$ involved in swarm robotic search with synchronous communication mode. For convenience, the default variables are for $R_i$ unless those are marked explicitly for TVCS.

---

1: **initialize**
2:        $j \leftarrow 1$
3:        velocity $V(t=0)$, position $X(t=0)$
4:        target signals measurement $I(t=0)$
5:        $I_{best} \leftarrow I(t=0)$, $X_{best} \leftarrow X(t=0)$
6:        $I_{best} \leftarrow I(t=0)$, $X_{best} \leftarrow X(t=0)$ for TVCS
7: **while** target is not found out
8:        **for** $i=1; i <= popsize; i++$
9:             calculate expected and real velocities
10:       calculate position
11:       **end**
12:       target signals measurement
13:       update $I_{best}$ and $X_{best}$
14:       update $I_{best}$ and $X_{best}$ of TVCS
15:       calculate velocity
16:       move ahead one step
17:       $j \leftarrow j+1$
18: **end**

---

## 4. Simulations

The two control algorithms are performed and repeated for 10 runs respectively, for both are characteristic of random search in nature and the comparison can be made with statistics gotten from enough simulations.

### 4.1 Parameter settings

The parameters about working environment, individual robot and swarm robots affect the results directly. Thus the important parameters and their common configurations used in simulations are given in Table 1. The farther information of the symbols can be found in the third column of this table.

### 4.2 Performance metrics

In order to comparatively evaluate the running performance of control algorithms with different communication modes, some quantitative metrics need to be designed in advance. Here, two performance figures are considered and presented. Both are based on the definition of search success. We can define such term as the best position of swarm closes to target

**Algorithm 4** Controlling robot $R_i$ involved in swarm robotic search with asynchronous communication mode. Note that symbols share the same meanings as in Algorithm 3.

```
 1: initialize
 2:        set counter j ← 1
 3:        velocity V(t = 0), position X(t = 0)
 4:        target signals measurement I(t = 0)
 5:        I_best ← I(t = 0), X_best ← X(t = 0)
 6:        I_best ← I(t = 0), X_best ← X(t = 0) for TVCS
 7:        calculate number of neighbors neighbor_number in TVCS
 8: while target is not found out
 9:        for i = 1; i <= neighbor_number; i + +
10:               target signals measurement
11:               if best-found is gotten
12:                      update I_best and X_best
13:               end
14:               calculate expected V_expec and real velocity V_real
15:               calculate X
16:        end
17:        target signals measurement
18:        update I_best and X_best
19:        update I_best and X_best for TVCS
20:        calculate velocity
21:        move ahead one step
22:        j ← j + 1
23: end
```

| Symbol | Value | Meaning |
|---|---|---|
| $Space$ | $500 \times 500$ | size of searching environment |
| $StartArea$ | $160 \times 160$ | start area for robots at the beginning of simulations |
| $popsize$ | $3, 5, 8, 10$ | number of individual robots in swarm robotic system |
| $R_{detec}$ | $250, 125$ | maximum detection radius of sensors |
| $R_{comm}$ | $250$ | maximum communication radius of robot |
| $V_{max}$ | $5$ | maximum moving velocity of robot |
| $P$ | $1600$ | signals power emitted from target |
| $T$ | $70$ | constant of inertia element |
| $\Delta t$ | $0.8$ | contracted factor for a moving step of robots |

Table 1. Parameter settings taken in simulations. Note that those parameters expressed in certain dimensions are assigned a corresponding proper one respectively.

enough so as that at least one robot can identify the target with some sensory ability (Kowadlo *et al.*, 2006). It means that if one or more robots approach and reach the area where target locates, the run is considered successful.

### 4.2.1 Search efficiency

Search efficiency is defined as reciprocal of mean steps required for one successful search. In fact, it concerns search speed by counting time steps for completion of a successful search, which indicates the elapsed time in a single simulation run indirectly. Because the sampling

cycle in simulations has been determined in advance, a simple relation between time step and spent time can be established. The value equals to the reciprocal of average steps taken by all individual robots in a successful search. Clearly, the more the average time steps, the lower the search efficiency, and vice versa.

### 4.2.2 Energy consuming

The metric is distance principle-based one. It is expressed in form of the sum of passed distance of all the individual robots when the swarm robotic search task is completed. Since the energy consumption of robot is fixed per distance unit, the average energy consumption of individual robots can measure aspect of algorithm performance in economical efficiency. Compared with energy consuming, search efficiency seems more important in swarm robotic search control evaluate because we concern about higher algorithmic speed.

### 4.3 Results

Simulations with different setting configurations are conducted and repeated for 10 runs respectively, for reducing the effect caused by the inherent randomness from the swarm intelligence-based control algorithms. Then the results are shown in some figures and tables. Of them, the running screenshots of swarm robotic system with different sizes when simulated programs terminate at first, see Figure 5 for details. We can find that the robots start searching from the lower left corner of the working space at the beginning of each simulation, being limited to a area of $160 \times 160$. While the target position is initialized at the same time, being limited to the upper right corner of the searching environment. As for the statistics from simulations, they are shown in Table 2 – 5, and Figure 6 – 9.

| comm. mode | 3-rob swarm | 5-rob swarm | 8-rob swarm | 10-rob swarm |
|---|---|---|---|---|
| synchronous mode | 98.2 ±6.11 | 94.3 ±13.23 | 93.9 ±3.81 | 93.2 ±7.16 |
| asynchronous mode | 95.4 ±5.48 | 91.6 ±5.83 | 93.2 ±7.35 | 92.3 ±6.10 |

Table 2. Average control time steps spent in completion of target search under conditions of $R_{detec} = 250, R_{comm} = 250$.

| comm. mode | 3-rob swarm | 5-rob swarm | 8-rob swarm | 10-rob swarm |
|---|---|---|---|---|
| synchronous mode | 1165.3 ±74.52 | 1863.1 ±259.72 | 2965.3 ±119.51 | 3665.9 ±254.90 |
| asynchronous mode | 979.9 ±86.76 | 1681.2 ±144.11 | 2603.9 ±220.70 | 3315.4 ±186.62 |

Table 3. Average total distance spent by all robots for target search success under conditions of $R_{detec} = 250, R_{comm} = 250$.

### 4.4 Discussions

We can comparatively analyze and discuss the indications surrounding simulation results, trying to reveal effects of different communication modes on swarm robotic search.

- As for the same task and same parameter setting configurations, time steps decrease as swarm size increases regardless of which communication mode taken in control algorithm. It indicates that search efficiency enhances as swarm size expands.

(a) 3-rob swarm



(b) 5-rob swarm


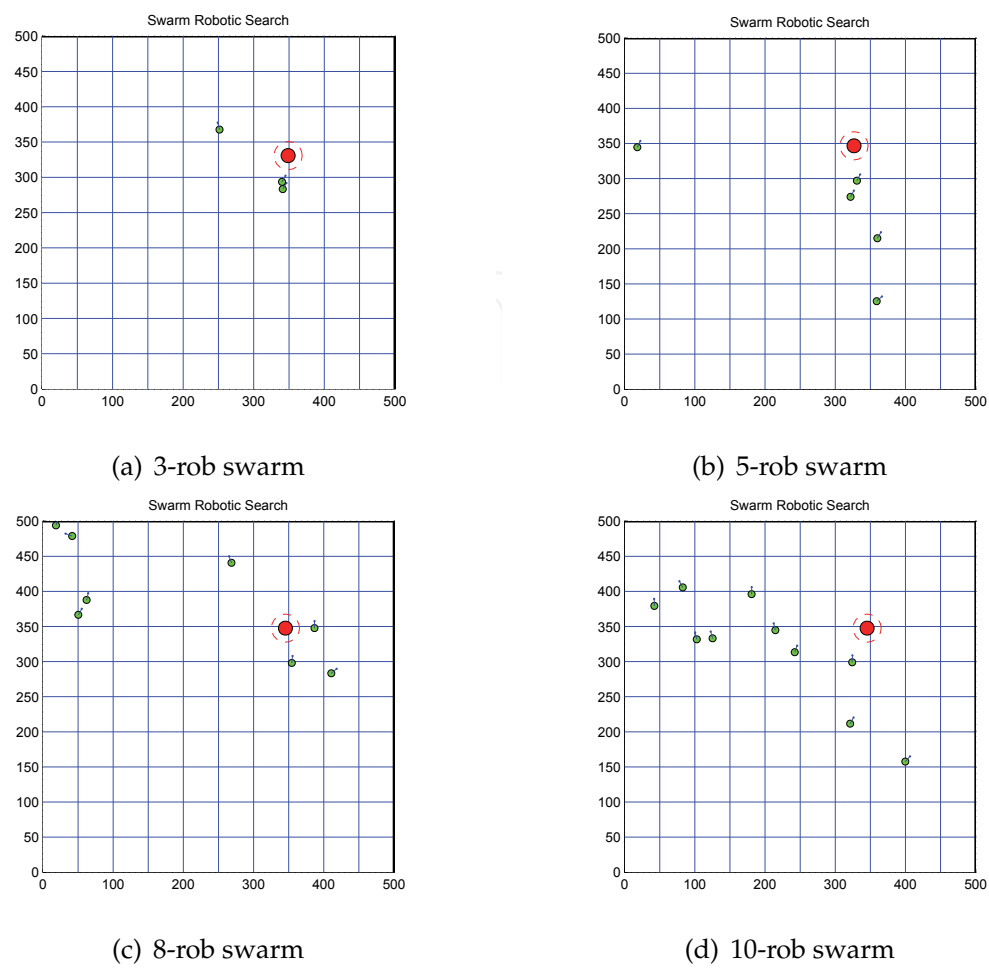
(c) 8-rob swarm



(d) 10-rob swarm

Fig. 5. Screenshots of swarm robotic search in cases of different size swarms. Note that at least one robot closes to the target enough when search succeeds.

| comm. mode | 3-rob swarm | 5-rob swarm | 8-rob swarm | 10-rob swarm |
|---|---|---|---|---|
| synchronous mode | 91.3 $\pm 3.36$ | 97.1 $\pm 10.15$ | 97.9 $\pm 4.91$ | 89.8 $\pm 6.55$ |
| asynchronous mode | 97.8 $\pm 7.40$ | 93.1 $\pm 7.43$ | 90.5 $\pm 3.96$ | 89.1 $\pm 4.88$ |

Table 4. Average control time steps spent in completion of target search under conditions of $R_{detec} = 125, R_{comm} = 250$.

| comm. mode | 3-rob swarm | 5-rob swarm | 8-rob swarm | 10-rob swarm |
|---|---|---|---|---|
| synchronous mode | 1081.0 $\pm 40.95$ | 1917.8 $\pm 207.36$ | 3117.9 $\pm 154.86$ | 3546.5 $\pm 269.06$ |
| asynchronous mode | 1017.6 $\pm 82.28$ | 1786.8 $\pm 146.27$ | 2628.1 $\pm 244.16$ | 3139.2 $\pm 274.70$ |

Table 5. Average total distance spent by all robots for target search success under conditions of $R_{detec} = 125, R_{comm} = 250$.
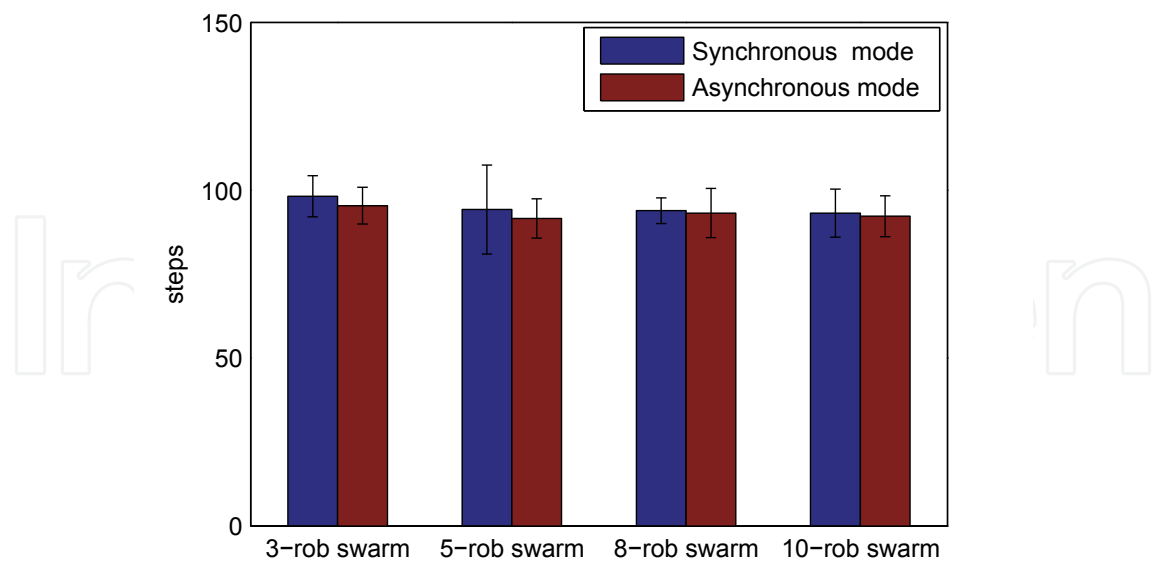
Fig. 6. Average time steps required to complete target search for 10 runs under conditions of $R_{detec} = 250, R_{comm} = 250$. Note that search efficiency is inversely proportional to time steps.
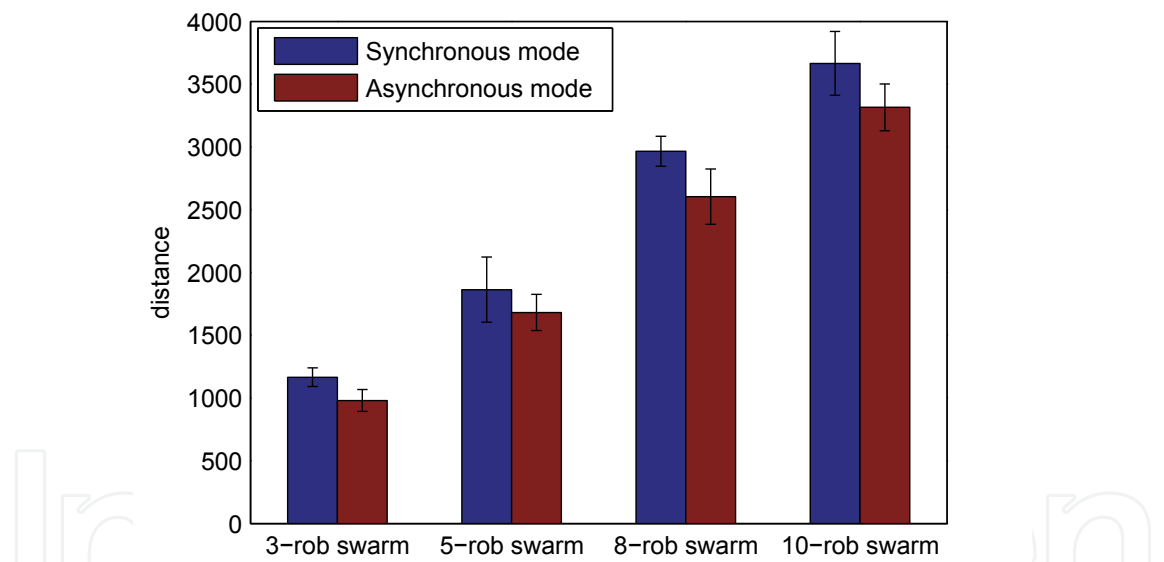


Fig. 7. Average total distance passed by all robots for 10 runs search success under conditions of $R_{detec} = 250, R_{comm} = 250$.

• As for the same parameter setting configurations, average total distance required for a success search varies in the same direction as swarm size increases regardless of which communication mode taken in control algorithm. It indicates that energy consuming largens as swarm size scales expansion. We can think that swarm robots carry out task of target search at compromise of time consuming and energy consuming.

• Swarm robotic search with asynchronous communication mode runs more efficiently than with synchronous communication mode, which seems to indicate that information and experience of certain dominant individual can be shared timely among its society for others
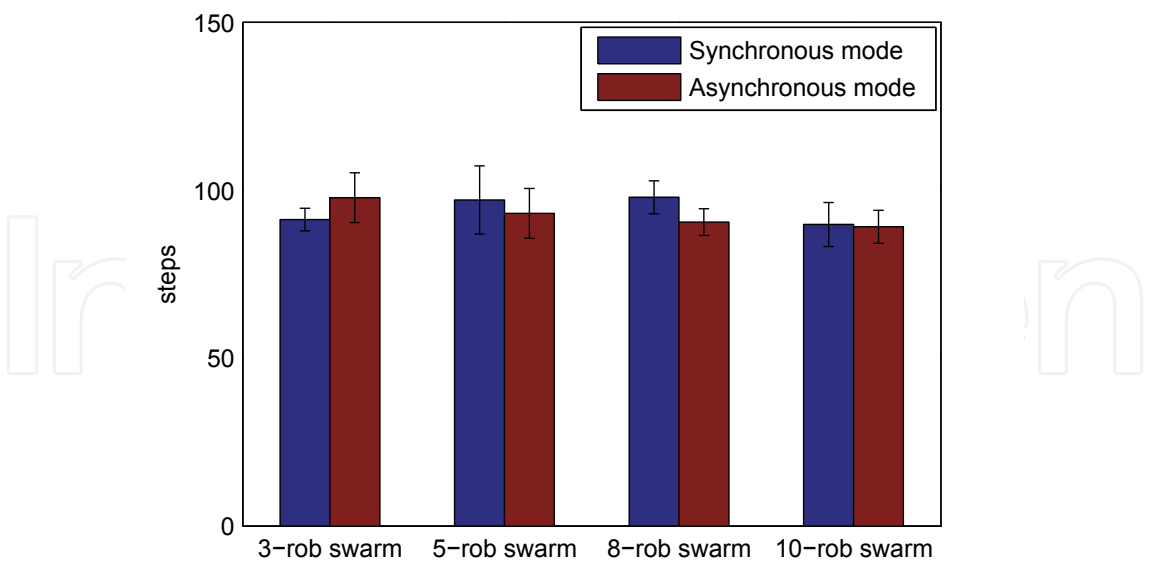
Fig. 8. Average time steps required to complete target search for 10 runs under conditions of $R_{detec} = 125, R_{comm} = 250$.
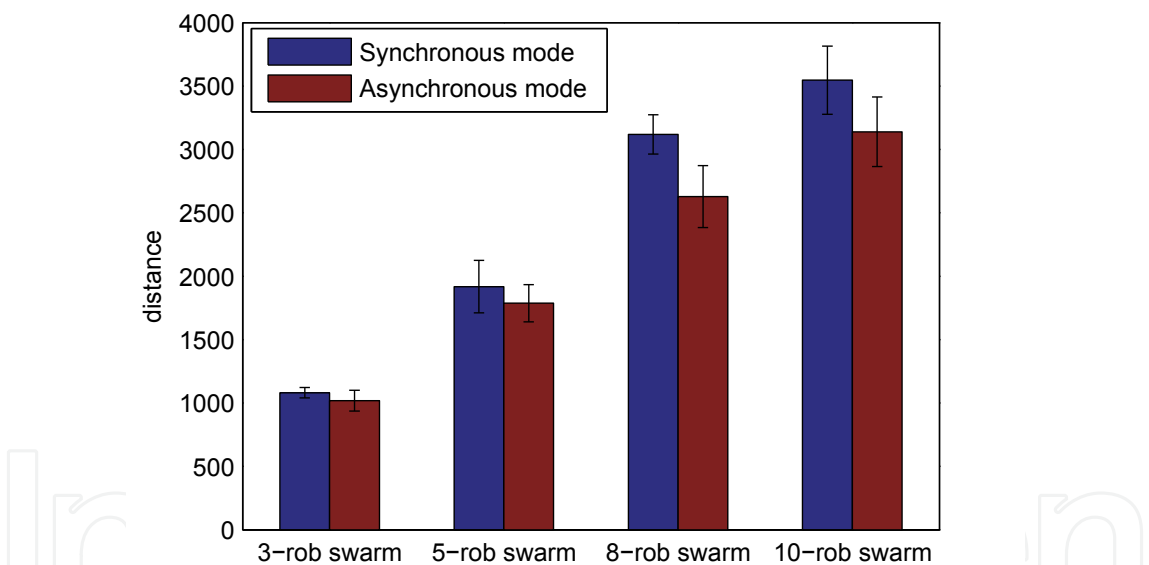


Fig. 9. Average total distance passed by all robots for 10 runs search success under conditions of $R_{detec} = 125, R_{comm} = 250$.

behavior decision making. The result seems that robots can adjust its own motion velocity and position and finally fasten the search process by learning from the optimal neighbors at different time steps.

• As to different parameter setting configurations, such as detection radius is set $R_{detec} = 250$ and $R_{detec} = 125$ respectively but the communication radius remains the same, search efficiencies and energy consuming do not vary obviously. The reason maybe that at the beginning of simulations, robots are far from the potential target so as not to be capable of

detecting target signals either for case of $R_{detec} = 250$ or for $R_{detec} = 125$. Therefore, robots only move randomly without help of the social experience sharing.

## 5. Conclusions

By extending the particle swarm optimization algorithm, we model swarm robotic system and control it in PSO-type way for carrying out target search task. Because of the relationship between swarm robotic search and PSO, some ideal characteristics of PSO can be transferred to case of swarm robotic search. Inspired from asynchronous versions of PSO, we develop control strategy with asynchronous communication mode for search efficiency enhancement. To reveal the effect, we compare the algorithm with synchronous communication mode. From the statistics of simulation, a conclusion can be drawn that asynchronous communication mode is more efficient than synchronous mode under conditions of the same parameter settings in search efficiency. In our future work, we will explore the effect how to vary as controlled object and task change.

## 6. References

Beni, G. (2005) From swarm intelligence to swarm robotics, *Lecture Notes in Computer Science*, vol. 3342, pp. 1–9.

Şahin, E. (2005) Swarm robotics: From sources of inspiration to domains of application, *Lecture Notes in Computer Science*, vol. 3342, pp. 10–20.

Dorigo, M. and Sahin, E. (2004) Swarm robotics–special issue editorial, *Autonomous Robots*, vol. 17, nos. 2–3, pp. 111–113.

Schmickl, T. and Crailsheim, K. (2008) Trophallaxis within a robotic swarm: bio-inspired communication among robots in a swarm, *Autonomous Robots*, vol. 25, no. 1, pp. 171–188.

Chang, J.F., Chu, S.C., Roddick, J. F. and et al. (2005) A Parallel Particle Swarm Optimization Algorithm with Communication Strategies, *Journal of Information Science and Engineering*, no. 21, 809–818.

Eberhart, R.C. and Shi, Y. (2001) Particle swarm optimization: developments, applications and resources, in *Proceedings of the 2001 IEEE Congress on Evolutionary Computation*, Piscataway, NJ, USA, vol. 1, pp. 81–86.

Henrich, D. and Honiger, T. (1997) Parallel Processing Approaches in Robotics, in *Proceedings of IEEE International Symposium on Industrial Electronics*, Guimaraes, Portugal, July 7–11.

Huang, F. and Fan, X.P. (2006) Parallel Particle Swarm Optimization Algorithm with Island Population Model, *Journal of Control and Decision*, vol. 21, no. 2, pp. 175–179, 188. (in Chinese)

Kennedy, J. and Eberhart, R. (1995) Particle swarm optimization, in *Proceedings of IEEE International Conference on Neural Networks*, vol. 4.

Koh, B., George, A.D., Haftka, R.T. and et al. (2006) Parallel Asynchronous Particle Swarm Optimization, *International Journal of Numerical Methods in Engineering*, vol. 67, pp. 578–595.

Kolda, T.G. and Torczon, V.J. (2003) Understanding Asynchronous Parallel Pattern Search, *High Performance Algorithm and Software for Nonlinear Optimization*, pp. 316–335.

Kowadlo, G., Rawlinson, D., Russell, R.A. and et al. (2006) Bi-modal search using complementary sensing (olfaction/vision) for odour source localization, in *Proceedings of the IEEE International Conference on Robotics and Automation*, Orlando.

Li, J.M., Wan, D.L., Chi, Z.X. and et al. (2006) A Parallel Particle Swarm Optimization Algorithm Based on Fine-Grained Model with GPU-Accelerating, *Journal of Harbin Institute of Technology*, vol. 38, no. 12, pp. 2162–2166. (in Chinese)

Luo, J.H. and Zhong, Z.N. (2005) Research on the Parallel Simulation of Asynchronous Pattern of Particle Swarm Optimization, *Journal of Computer Simulation*, vol. 22, no. 6, pp. 68–70, 78. (in Chinese)

Pugh, J. and Martinoli, A. (2007) Inspiring and Modeling Multi-Robot Search with Particle Swarm Optimization, in *Proceedings of the 4th IEEE Swarm Intelligence Symposium*, Honolulu, Hawaii, USA, April, 1–5.

Zeng, J.C. and Xue, S.D. (2010) Modeling and Simulation Approaches to Swarm Robotic Systems, *Journal of System Simulation*, vol. 22, no. 6, pp. 1327–1330. (in Chinese)

Ridge, E., Kudenko, D., Kazakov, D. and et al. (2005) Moving Nature-Inspired Algorithms to Parallel, Asynchronous and Decentralised Environment, *Frontiers in Artificial Intelligence and Applications*, Vol. 135, pp. 35–49.

Schutte, J.F., Reinbol, J.A., Fregly, B.J. and et al. (2004) Parallel Global Optimization with the Particle Swarm Optimization, *International Journal of Numerical Methods in Engineering*, vol. 61, no. 13, pp. 2296–2315.

Venter, G. and Sobieszczanki-Sobieksi, J. (2006) A Parallel Particle Swarm Optimization Algorithm Accelerated by Asynchronous Evaluations, *Journal of Aerospace Computing, Information, and Communication*, no. 3, pp. 123–137.

Wang, Y.Y., Zeng, J.C. and Tan, Y. (2007) Parallel Particle Swarm Optimization Based on Parallel Model with Controller, *Journal of System Simulation*, vol. 19, no.10, pp. 2171–2176. (in Chinese)

Xu, Y.Z. and Zeng, W.H. (2005) The Development of Parallel Evolutionary Algorithms, *Pattern Recognition and Artificial Intelligence*, vol. 21, no. 2, pp. 183–192. (in Chinese)

Xue, S.D. and Zeng, J.C. (2008a) Using Relative Localization Observations for Swarm Robots Search, in *International Conference on Modelling, Identification and Control*, Shanghai, China, June 29–July 2.

Xue, S.D. and Zeng, J.C. (2008b) Control over Swarm Robots Search with Swarm Intelligence Principles, *Journal of System Simulation*, vol. 20, no. 13, pp. 3449–3454.

Xue, S.D. and Zeng, J.C. (2008c) Swarm Robots Search under Conditions of Heterogeneous Sensory Signals Fusion, in *Proceedings of the 2008 International Conference on Genetic and Evolutionary Methods*, Las Vegas, Nevada, USA, July 14–17.

Zeng, J.C., Jie, J. and Cui, Z.H. (2004) *Particle Swarm Optimization*, Beijing: Science Press. (in Chinese)

Zhao, Y., Yue, J.G., Li, G.Y. and et al. (2005) A Parallel Particle Swarm Optimization Algorithm Based on Multigroup for Solving Complex Functions Optimization, *Journal of Computer Engineering and Applications*, vol. 41, no. 16, pp. 58–60, 64. (in Chinese)

Xue, S.D., Zhang, J.H. and Zeng, J.C. (2009) Parallel asynchronous control strategy for target search with swarm robots, *International Journal of Bio-Inspired Computation*, vol. 1, no. 3, pp. 151–163.

Xue, S.D. and Zeng, J.C. (2009) Controlling swarm robots for target search in parallel and asynchronously, *International Journal of Modelling, Identification and Control*, vol. 8, no. 4, pp. 353–360.

**Mobile Robots - Control Architectures, Bio-Interfacing, Navigation, Multi Robot Motion Planning and Operator Training**
Edited by Dr. Janusz Będkowski

ISBN 978-953-307-842-7

Hard cover, 390 pages

**Publisher** InTech

**Published online** 02, December, 2011

**Published in print edition** December, 2011

The objective of this book is to cover advances of mobile robotics and related technologies applied for multi robot systems' design and development. Design of control system is a complex issue, requiring the application of information technologies to link the robots into a single network. Human robot interface becomes a demanding task, especially when we try to use sophisticated methods for brain signal processing. Generated electrophysiological signals can be used to command different devices, such as cars, wheelchair or even video games. A number of developments in navigation and path planning, including parallel programming, can be observed. Cooperative path planning, formation control of multi robotic agents, communication and distance measurement between agents are shown. Training of the mobile robot operators is very difficult task also because of several factors related to different task execution. The presented improvement is related to environment model generation based on autonomous mobile robot observations.

**How to reference**
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Songdong Xue, Jin Li, Jianchao Zeng, Xiaojuan He and Guoyou Zhang (2011). Synchronous and Asynchronous Communication Modes for Swarm Robotic Search, Mobile Robots - Control Architectures, Bio-Interfacing, Navigation, Multi Robot Motion Planning and Operator Training, Dr. Janusz Będkowski (Ed.), ISBN: 978-953-307-842-7, InTech, Available from: http://www.intechopen.com/books/mobile-robots-control-architectures-bio-interfacing-navigation-multi-robot-motion-planning-and-operator-training/synchronous-and-asynchronous-communication-modes-for-swarm-robotic-search

# INTECH

open science | open minds