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

3,800 Open access books available
116,000 International authors and editors
120M Downloads

154 Countries delivered to
TOP 1% Our authors are among the 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

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com
Multi-Ring Particle Swarm Optimization

Carmelo J. A. Bastos-Filho, Danilo F. Carvalho, Marcel P. Caraciolo, Péricles B. C. Miranda and Elliackin M. N. Figueiredo
Department of Computing and Systems, University of Pernambuco 
Brazil

1. Introduction

Many real world problems are quite expensive to be solved by using exhaustive computation. Because of this, many metaheuristics have been proposed recently to deal with optimization and search problems. Although many approaches have been proposed based on local search, the current research in Evolutionary Computation (EC) is mainly focused on algorithms that are based on populations and inspired in nature (Schwefel, 1981; Bäck, 1996; Fogel, 1996; Beyer, 2001; Beyer, 2002).

In this context, Swarm intelligence techniques, mainly Particle Swarm Optimization (PSO), have been widely applied to solve many different types of optimization problems. PSO was proposed by Kennedy and Eberhart (Kennedy & Eberhart, 1995) and the main idea is to explore the intelligence originated by the social knowledge disseminated by the agents during the search process. Due to its great performance in difficult optimization tasks and its simplicity, PSO became a popular, easy-to-use and extendable technique (Abido, 2001; Eberhart & Shi, 2001; Parsopoulos & Vrahatis, 2001; Agrafiotis & Ceder, 2002; Coello Coello & Lechuga, 2002; Parsopoulos et al., 2002; Shi & Krohling, 2002; Praveen et al., 2007; Santana-Quintero ET AL., 2008).

By applying PSO to problems where the feasible solutions are too much difficult to find, more sophisticated approaches are required, mainly for hyper dimensional spaces. Many variations on the basic PSO form have been explored, most of them proposing different velocity update equations (Shi & Eberhart, 1998; Clerc & Kennedy, 2002; Kennedy, 2005; Jiao, et al., 2008). Other approaches attempt to change the structure of the swarm (Kennedy & Mendes, 2002; Suganthan, 1999) and its communication schema (Engelbrecht, 2005; Sutton et al., 2006).

To disseminate information within the swarm is the key of any swarm intelligence based algorithm. PSO, like others swarm algorithms, makes use of its own information exchange methods to distribute the best positions found during the algorithm execution (Engelbrecht, 2005). The way used by the swarm to distribute this information lies in the social structure formed by the particles. Variations in this structure can improve the algorithm performance. By using local communication, each particle of the swarm has its own neighborhood. Many papers (Engelbrecht, 2005; Carvalho & Bastos-Filho, 2008; Carvalho & Bastos-Filho, 2009) have shown that it improves the exploration capacity of the entire swarm but decreases the convergence speed of the algorithm. It means that the algorithm obtains better solutions, but it needs more iterations to reach the convergence.
Other important and already known PSO algorithm variation is the Cooperative Particle Optimizer (CPSO) (van den Bergh & Engelbrecht, 2001; van den Bergh & Engelbrecht, 2004) which is based on the cooperative GA proposed by Potter and De Jong (Potter & De Jong, 1994). This approach proposes a performance improvement of the PSO algorithm by splitting the solution vector into different components that will be optimized by different sub-swarms cooperatively. CPSO inserts another parameter called split factor which can be interpreted as the number of sub-swarms that compose the entire swarm.

By using the ring structure as an inspiration, a novel PSO topology called Multi-ring PSO (MRPSO) based on coupling different rings was recently proposed (Bastos-Filho et al. 2008). This novel approach makes use of a diversity mechanism provided by ring rotations in order to improve the exploration of the search space. The structure of the proposed topology is formed by ring layers connected by axis that cross the layers through the particles. Each single particle, which belongs to a specific layer, is automatically connected to other particles in neighbour ring layers. These connections establish the communication between neighbour particles which is a peculiar neighbourhood scenario based on local communication. The ring layers increase the capacity to avoid stagnation states making use of a process and an attribute called rotation skill and trigger criterion, respectively, also described in the chapter. Furthermore, we included a novel concept proposed by Cai et al. (Cai et al. 2008) that assigns a grade to the particles based on their fitness. This information is used to switch the neighbourhood inside the rings.

The rest of this paper is organized as follows. Section 2 shows a brief review about PSO. It also describes some well known topologies. In Section 3 the proposed Multi-ring topology and its features are described. Section 4 is devoted to the Dispersed based Multi-ring approach. The simulation setup used by the experiments and the chosen benchmark functions are presented in Section 6. In section 7 the analysis of Multi-ring and Multi-ring Dispersed are shown. The results of a comparison with some well-known topologies are presented too.

2. Background

2.1 Particle Swarm Optimization - PSO

The basic form of the PSO algorithm is composed by a group particles, where each particle represents a possible solution to the problem. Each particle maps the attributes of a solution, i. e., the position of one particle represents exactly the values of the attributes of a solution. Besides, particles move inside the search space reaching new positions. After some iterations, particles tend to a global optimum based on two basic components: the cognitive component and the social component. The first one considers the best position already found by the particle itself (pbest). The second one use the best position attained by all the particles of the swarm (gbest) in order to guide the swarm to a common spot in the search space. Based on these two information components, each particle updates its own velocity. The velocity is a vector that directs the particle to the next position.

As the original PSO is based on the global knowledge of the swarm, it has a low capacity for local search. Thus, other approaches were proposed in order to avoid this disadvantage. Shi and Eberhart (Shi & Eberhart, 1998) proposed a velocity update equation that uses an inertia factor \( w \) (see equation 1). This factor balances the exploration-exploitation trade-off by controlling the contribution of the previous velocity in the evaluation of the current velocity.
of each particle. For each iteration of the PSO algorithm the velocity \( \mathbf{v}_i(t) \) contribution is decreased. Thus, the exploitation level of the swarm is increased.

\[
\mathbf{v}_i(t + 1) = w \mathbf{v}_i(t) + c_1 r_1 (\mathbf{p}_i(t) - \mathbf{x}_i(t)) + c_2 r_2 (\mathbf{g}_i(t) - \mathbf{x}_i(t)),
\]

where \( w \) is the inertia factor, \( \mathbf{x}_i(t) \) is the current position of particle \( i \) in time step \( t \), \( c_1 \) and \( c_2 \) are positive acceleration constants, and \( r_1 \) and \( r_2 \) are two random numbers generated by a uniform distribution in the interval \([0,1]\). Recently, Jiao et al. (Jiao et al., 2008) proposed a dynamic inertia weight for the particle swarm optimization algorithm. They showed that it can improve the quality of the solutions obtained by the algorithm.

Another popular approach for the PSO velocity update equation involves a constriction factor which implies a velocity value reduction at each time step (Clerc & Kennedy, 2002). Under some peculiar conditions, it can guarantee both convergence and a high degree of exploration. Both inertia and constriction approaches balance exploration and exploitation abilities and improve the convergence speed and the quality of the solutions found by the algorithm (Bratton & Kennedy, 2007).

### 2.2 Topologies

The communication between particles inside a swarm is one of the most important mechanisms of PSO. This mechanism distributes information already achieved by the

![Fig. 1. Commonly used particle topologies: (a) Star topology used in gbest, (b) Ring topology used in lbest, (c) Von Neumann topology and (d) Four clusters topology.](http://www.intechopen.com)
swarm through all particles, updating the swarm status. Global and local communication structures are the two main structures used for particles’ communication. Considering that particles share information acquired through a structured swarm, this model of communication can be seen as a topology. Some of the well-known topologies are depicted in Figure 1.

Each topology shown in Figure 1 has its own way to spread information through the swarm (Kennedy, 1999; Kennedy & Mendes, 2002).

Star and ring topologies are shown in Figure 1(a) and Figure 1(b), respectively. Star topology is composed by a fully-connected structure (gbest) that allows each particle of the swarm to share information globally. In a global neighbourhood, information is instantaneously disseminated to all particles, quickly attracting the entire swarm to the same swarm region. To avoid local minima, ring topology (Figure 1(b)) make use of local neighbourhoods (lbest), i.e., particles just communicate with its immediate neighbours. Therefore, the swarm needs a great number of iterations to converge, but higher quality solutions can be found.

In Figure 1(c), a grid of particles forming a peculiar social structure is shown. This structure is very useful for many optimization problems (Kennedy & Mendes, 2002; Peer et al., 2003) and is known as Von Neumann topology.

Based on clusters of particles, the four clusters topology is shown in Figure 1(d). Each cluster communicates with the others by connections previously defined between the particles (Mendes et al., 2003; Engelbrecht, 2005). The number of connections originated from a cluster is equal to the number of neighbour clusters.

3. Multi-ring topology

3.1 Multi-ring concepts

Multi-Ring is a novel PSO topology that uses the ring structure as an inspiration. By using local communication, each particle of the swarm has its own neighbourhood. Many papers have shown that it improves the exploration capacity of the entire swarm but decreases the convergence time of the algorithm (Engelbrecht, 2005; Carvalho & Bastos-Filho, 2008; Carvalho & Bastos-Filho, 2009). A topology based on coupling different rings with dynamic commutation of the neighbourhood is proposed, which aims to obtain better solutions as the benefit of the local communication. At the same time, we expect to reduce the number of iterations to reach a convergence as the benefit of the dynamic neighbourhood mechanism. The proposed topology is formed by particles arranged in multiple rings layers, where each particle of the ring can communicate with some particles of neighbour rings. In order to

Fig. 2. Multi-Ring Topology.
couple different ring layers, the number of particles in each ring must be the same. Therefore, each single particle, which belongs to a specific layer, is automatically connected to other particles from neighbour ring layers, as shown in Figure 2. The structure in Figure 2 illustrates an example of the multi-ring topology with 3 rings (A, B and C) of 3 particles.

3.2 Communication between different rings

The particles from different ring layers communicate with each other by using a peculiar neighbourhood scenario. The Figure 3 can be used to illustrate this scenario. Considering a ring layer $r_{l_0}$, each particle $r_{l_0(i)}$ is directly connected with the particles $r_{(k-1)(i)}$ and $r_{(k+1)(i)}$ of the next and previous ring layers. During the algorithm execution, one particle $r_{l_0(i)}$ will exchange information with a neighbourhood composed by \{ $r_{l_0(i-1)}$, $r_{l_0(i+1)}$, $r_{(k-1)(i)}$, $r_{(k+1)(i)}$ \}.

Fig. 3. Ring layers structure.

One special observation must be considered for the first and last ring layers of the topology. The communication between the particles located at those two rings is partially limited, since both do not have a complete ring neighbourhood, it means that they only have a unique ring layer directly connected to them. Therefore, the information exchange for one particle belonging to a ring $r_{l_1}$ is limited by the set \{ $r_{l_1(i-1)}$, $r_{l_1(i+1)}$, $r_{l_{(k-1)(0)}}$ \}. The same restriction must be considered for the ring $r_{l_n}$, which the neighbourhood is composed by the set \{ $r_{l_{(k-1)(0)}}$, $r_{l_{(k+1)(0)}}$, $r_{l_{(n-1)(0)}}$ \}. One particle which belongs to a ring layer $r_{l_0}$ exchange information with other layers $r_{l(k)}$ and $r_{l(k+1)}$ in order to make use of the social information achieved by the other layers. If one particle just exchanges information with the same neighbourhood through the entire execution of the algorithm, the neighbourhood would be static. In this case, the topology is just a simple local approach. To avoid this static possibility, one important process was added to the proposed topology, called rotation skill.

3.3 Ring layer rotation skill

One of the biggest problems of PSO is the stagnation. In nonlinear multimodal problems, which have many local minima regions, the swarm can be stagnated in one those regions of the search space during the algorithm execution. In those regions, the particles can spend
many time steps without varying the value of the best position found by their neighbourhoods. It can cause a stagnation state in the entire swarm. Hence, the proposed strategy to avoid stagnation is to add the ability of rotation to each ring layer. By adding the rotation skill, the position of the particles in the ring can be altered. Hence, the neighbourhood is changed. It can move the swarm from one region to another, increasing the exploration of the space search by considering different local minima regions.

In the Multi-Ring approach, the particles of different rings exchange information between themselves using local communication. The particle that belongs to a ring \( r_{(k)} \) exchanges information with other rings \( r_{(k-1)} \) and \( r_{(k+1)} \), aiming to transmit its previous experience. Each ring layer of the Multi-Ring has the ability to self-rotate. It means that if the ring layer \( r_{(k)} \) does not improve its own best position, defined as \( \text{ringbest} \), it will be rotated. In the rotation process, each particle in the ring layer has its index changed to \( i = (i+d) \mod (n_l) \), where \( d \) is the rotation distance and \( n_l \) is the number of particles in a layer. Thus, the new neighbourhood will be composed by \{ \( r_{(k-1)}(i-1) \), \( r_{(k-1)}(i+1) \), \( r_{(k+1)}(i+d) \), \( r_{(k+1)}(i+d) \) \}. One can note that the main difference is the communication replacement between the neighbour ring layers.

Figure 4 shows an example of the rotation process. Initially, the particle E communicates with \{D,F,B,H\}. After the rotation, the neighbourhood of the particle E is changed to \{D,F,A,G\}. This process improves the social information exchange. Besides, it also improves the convergence capacity of the swarm. One can note that all the particles of the same ring layer change its neighbourhood.

![Fig. 4. Rotation skill example.](image-url)

The main mechanism of the rotation process is the trigger criterion that will be used to invoke the rotation skill. It is based on the idle status of the best solution found in a specific ring layer, i.e. the number of iterations that the best solution found by a specific ring does not change. To clarify the process described above, consider a ring layer \( r_{(k)} \). If the best solution found remains with the same value during \( t_r \) iterations, so this ring layer will be rotated after \( t_r \) iteration.

Therefore, the use of the exploration capacity provided by the local communication combined with rotation mechanisms helps to avoid the stagnation of the solution found by the swarm during the execution of the PSO algorithm.
3.4 Multi-ring PSO algorithm

Inspired by the entity described in the previous section, the PSO algorithm with this novel topology has some new specific steps that must be considered in the execution process of the algorithm. Figure 5 illustrates the Multi-Ring topology containing an example with two ring layers \((r_{l(1)}\) and \(r_{l(2)}\)) of three particles.

![Multi-Ring topology of 2 ring layers with 3 particles.](image)

In the basic PSO algorithm, one of the steps is the definition of the initial \(p_{best}\) and \(g_{best}\), which represent the best position found by the particle and swarm, respectively. The manner how the value of \(p_{best}\) is updated or chosen is not influenced by the use of Multi-Ring topology, since it is related to the individual experience of the particle. Therefore, all the particles of the swarm define the \(p_{best}\) in the same way.

For each iteration, each ring layer performs a search and marks the particle that reached the best position of the entire ring layer. Different from \(g_{best}\), in the Multi-Ring approach, the definition of this new position called \(ring_{best}\), considers only the particles within the respective ring layer. Therefore, each ring layer will have its own \(ring_{best}\). After that, the \(ring_{bests}\) of each ring layer are compared with each other, and the best one will be selected as the best solution. The pseudo-code of the algorithm is presented in Figure 6.

```
Algorithm 1: Multi-Ring based particle swarm updating position, velocity and cognitive and social information during the search process.

1. initializeSwarm()
2. while step <= numberOfSteps do
3.   foreach ring of the swarm do
4.     rotate();
5.     foreach particle of the ring do
6.       updateVelocityAndPosition();
7.       calculateFitness();
8.       updateInformation();
9.     end
10. // marks the best particle of each ring
11.   defineRingBest();
12. end
13. evaluateBestRing();
14. return bestPositionFound
```

![Multi-Ring particle swarm optimization algorithm.](image)

After the \(p_{best}\) evaluation, the algorithm performs the selection of the best particle of the respective ring layer. This process is repeated for all the ring layers that compose the Multi-
Ring topology (see Line 10 in Figure 6). Line 12 of Figure 6 shows that the selection of the best \textit{ringbest} for the topology occurs at the end of each algorithm iteration. The \texttt{updateVelocityAndPosition()} and \texttt{updateInformation()} processes are the core of any PSO algorithm. These two processes are described in the pseudo-codes of Algorithms 2 and 3 that are shown in Figures 7 and 8, respectively.

\begin{algorithm}
\caption{Position of one particle being updated based on PSO velocity update equation.}
\begin{algorithmic}[1]
\While{foreach dimension do}
\State newVelocity $\leftarrow$ currentVelocity $+ c_1 \ast r_1 \ast (\text{bestIndividualPosition} - \text{currentPosition}) + c_2 \ast r_2 \ast (\text{neighborhoodBestPosition} - \text{newPosition})$
\State newPosition $\leftarrow$ currentPosition + velocity
\EndWhile
\end{algorithmic}
\end{algorithm}

Fig. 7. Particle swarm position and velocity updating algorithm.

\begin{algorithm}
\caption{Particle information update method.}
\begin{algorithmic}[1]
\If{currentFitness $<$ bestIndividualFitness}
\State bestIndividualFitness $\leftarrow$ currentFitness
\EndIf
\If{currentFitness $<$ neighborhoodBestFitness}
\State neighborhoodBestFitness $\leftarrow$ currentFitness
\EndIf
\end{algorithmic}
\end{algorithm}

Fig. 8. Particle information updating algorithm.

The other process involved in Algorithm 1 execution is \texttt{rotate()} (see Line 4), which belongs only to the multi-ring topology approach proposed here. One can note that this step is done before the \textit{fitness} evaluation. Figure 9 can be used to represent this process. Consider that the rotation trigger criterion used is the number of successive iterations that the best solution found (\textit{ringbest}) by the ring layer has not improved and \textit{d} is the number of positions that the ring will be rotated. Using \textit{t} to represent the trigger mechanism and \textit{X\textsubscript{i}} to represent the particle \textit{i} of the ring layer \textit{X}, the ring layer \textit{X} will be rotated if the value of its best position has not improved after \textit{t} iterations. Otherwise, if the respective number of iterations is below \textit{t}, the algorithm does not rotate the ring layer \textit{X} and the number of iterations is incremented. This process is done for all ring layers of the Multi-ring topology. When the trigger criterion is satisfied, the ring layer is rotated by \textit{d} particles and the number of iterations that the \textit{ringbest} did not improved is reset. It implies in a new neighbourhood for each particle of the rotated ring and, as a result, it improves the social information exchange. One can note that there is a trigger criteria \textit{t\textsubscript{r(k)}} for each ring layer \textit{r\textsubscript{l(k)}}. The figure 10 illustrates the rotation algorithm process described above.

4. Dispersed particle swarm optimization

4.1 Dispersed PSO concepts
In particle swarm optimization (PSO) literature, the published social coefficients settings aims to increase the search around the swarm memory. However, few approaches
take into account the useful information inside the particle’s memories. Thus, to improve the convergence speed, a new approach considering the social coefficient was proposed (Cai et
al, 2008) by introducing an explicit selection pressure, in which each particle decides its search direction toward the personal memory or swarm memory. The fitness value of the current position of each particle represents its adaptive capability, and it can be selected to distinguish between search modes. In this approach, the better the fitness of the current position is, more adaptation this particle can get to enhance its survival ratio.

Since the literature consider the extreme situations such as global or local approaches, they neglect the differences between the particles in the search process. These settings lose some information that can be useful to find the global optima or escape from local optima. To minimize this problem, Cai et al. (Cai et al., 2008) proposed a new index to indicate performance differences, named Grade(t). The definition of the index is provided as follows:

\[ \text{Grade}_i(t) = \frac{f_{\text{worst}}(t) - f_{\text{best}}(t)}{f_{\text{worst}}(t) - f_{\text{best}}(t)}, \]

where \( f_{\text{worst}}(t) \) and \( f_{\text{best}}(t) \) are the worst and the best fitness values of the swarm at time step \( t \), respectively. In most cases, if the fitness value of particle \( i \) is better than the fitness of particle \( u \), the probability of \( i \)'s neighborhood falls into a global optima is higher than the particle \( u \). Thus, the particle \( i \) should give more attentions to exploit its neighbourhood. On the contrary, it may tend to explore other regions with a larger probability. Therefore, the best solution should perform a complete search around its historical best position and the worst solution should perform a global search around the best solution found so far.

Then, the dispersed social coefficient of particle \( i \) at time \( t \) is set as follows:

\[ c_{2,i}(t) = c_{\text{low}} + (c_{\text{up}} - c_{\text{low}}) \text{Grade}_i(t), \]

where \( c_{\text{up}} \) and \( c_{\text{down}} \) are two predefined numbers, and \( c_{2,i}(t) \) represents the social coefficient of particle \( j \) at time \( t \). One can note that better particles will present higher values for \( c_{2} \).

### 4.2 Multi-ring dispersed particle swarm optimization

Based on the new manner to determine the acceleration coefficients, the dispersed PSO approach was included into the Multi-Ring Topology. By applying the \( \text{Grade}(t) \) to determine the communication rules inside each ring, we expect that it will improve the performance of the search made by the Multi-Ring PSO algorithm.

The grade criterion, which is calculated by evaluating a fitness comparison, will make the best solutions perform searches around its vicinities and the worst solutions look for the history of a larger part of the swarm, in order to increase its experience.

Figure 11 illustrates the grade strategy in action. If the \( \text{Grade}(t) \) is above a threshold (good solutions), we reduce the neighbourhood of the particle. In this case, the neighbourhood of the particle \( i \) is defined as \( \{ r_{i,0}, r_{i,1}, r_{i,2}, r_{i,3}, r_{i,4}, r_{i,5}, r_{i,6}, r_{i,7}, r_{i,8}, r_{i,9} \} \). Otherwise, the particle should expand its communication range to achieve more information, which it would get.
information from all particles in the same ring in order to increase its experience. One can note that rings which the ringbest are below the threshold will perform searches using a more connected structure in order to achieve better regions in the search space. In spite of this, one should note that the communication between the ring layer remains local.

Fig. 11. Grade criteria at Multi-Ring PSO.

5. Simulation setup

In Table 1, seven commonly used benchmark functions are presented. Their names, definitions, search space, initialization range, and optimum compose the columns of Table 1. All the functions described are used as minimization problems classified as unimodal (f1) or multimodal (f2, f3, f4, f5, f6, and f7) containing many local minima and a high level of complexity. For the simulations, each function was tested with 30 dimensions. Each simulation was performed using PSO with inertia factor. The reduction of $w$ is linear and starts at 0.9 decreasing to 0.4 along the iterations. The number of rings was varied based on a swarm with 30 particles. Thus, the number of ring layers is defined as $n_l = 30 / n$. The rotation distance is equal to a half of the ring layer size ($d = n_l / 2$) and the trigger criterion was $t_r = 15$.

We used $c_1$ and $c_2$ equal to 2.05. In the MRDPSO, we used $c_{low} = 1.5$ and $c_{up} = 2.5$.

The number of iterations used in a single simulation was 10,000. After 30 executions, the mean and standard deviation were stored. The initial positions of the entire swarm are defined randomly inside the initialization range of the benchmark function used. Comparisons with other topologies were also provided for convergence performance validation in both inertia and dispersed approaches.
Evolutionary Computation

534


Rosenbrock \( f_1 \) 
\[ f_1 = \sum_{i=1}^{n} \left[ 100 (x_{i+1} - x_i)^2 + (1 - x_i)^2 \right] \] 
\((-30, 30)^D\) \((15, 30)^D\) \(1.0^D\)

Ackley \( f_2 \) 
\[ f_2 = -20 \exp -0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2} - \exp \left( \frac{1}{n} \sum_{i=1}^{n} \cos (2 \pi x_i) \right) + 20 + e \] 
\((-32, 32)^D\) \((16, 32)^D\) \(0.0^D\)

Griewank \( f_3 \) 
\[ f_3 = \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} \cos \left( \frac{x_i}{\sqrt{i}} \right) \] 
\((-600, 600)^D\) \((300, 600)^D\) \(0.0^D\)

Penalized P8 \( f_4 \) 
\[ f_4 = \frac{\pi}{n} \left[ 10 \sin^2 (\pi y_1) + \sum_{i=2}^{n} (y_i - 1)^2 \right] \] 
\(-50, 50)^D\) \((25, 50)^D\) \(-1.0^D\)

Penalized P16 \( f_5 \) 
\[ f_5 = 0.1 \sin^2 (3 \pi y_1) + \sum_{i=2}^{n} (y_i - 1)^2 + \sin^2 (3 \pi x_i) + (x_1 - 1)^2 \sum_{i=2}^{n} x_i \mu (x, 5, 100, 4) \] 
\(-50, 50)^D\) \((25, 50)^D\) \(1.0^D\)

Rastrigin \( f_6 \) 
\[ f_6 = 10 n + \sum_{i=1}^{n} x_i^2 - 10 \cos (2 \pi x_i) \] 
\((-5.12, 5.12)^D\) \((2.56, 5.12)^D\) \(0.0^D\)

Schwefel 2.6 \( f_7 \) 
\[ f_7 = -\sum_{i=1}^{n} x_i \sin \left( \sqrt{x_i} \right) \] 
\((-500, 500)^D\) \((-500, -250)^D\) \(420.96^D\)

Table 1. Benchmark functions used in the experiments.

### 6. Results

This section shows the results of the experiments involving the benchmark functions described previously. In the first section (Section 6.1) of the simulation results, we vary the number of ring layers. The second one (Section 6.2) presents a comparison between each Multi-ring approach and the topologies already mentioned in Section 2 of this paper.

#### 6.1 Ring layer variation

Table 2 shows the results of MRPSO varying the number of ring layers in simulations involving each benchmark function. For function \( f_6 \), the structure composed by 3 ring layers behaved better. The 5-ring MRPSO achieved a better performance for functions \( f_6, f_2 \) and \( f_5 \), and the 10-ring behaves better for \( f_5, f_3, \) and \( f_7 \).

One can note that the best choice for function \( f_6 \) was 3-rings layers followed by the 5-rings configuration. Thus, in general, the 5-rings configuration achieved the most satisfactory fitness values. This is due to the fact that maintaining a balance between the degree of
Multi-Ring Particle Swarm Optimization

Table 2. Number of ring layers variation of MRPSO.

<table>
<thead>
<tr>
<th>Function</th>
<th>MRPSO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3 ring layers</td>
</tr>
<tr>
<td></td>
<td>Mean</td>
</tr>
<tr>
<td>(f_1)</td>
<td>17.48</td>
</tr>
<tr>
<td>(f_2)</td>
<td>3.19E-9</td>
</tr>
<tr>
<td>(f_3)</td>
<td>0.0097</td>
</tr>
<tr>
<td>(f_4)</td>
<td>2.56E-32</td>
</tr>
<tr>
<td>(f_5)</td>
<td>2.72</td>
</tr>
<tr>
<td>(f_6)</td>
<td>16.3</td>
</tr>
<tr>
<td>(f_7)</td>
<td>3305.57</td>
</tr>
</tbody>
</table>

For functions \(f_2\), \(f_3\) and \(f_5\) the 3-rings configuration achieved the best fitness values. The configuration with 10 ring layers achieved better performance in functions \(f_1\) and \(f_4\). Again, we can consider that a configuration with 5-rings is enough to reach satisfactory fitness values in complex and different search spaces.

Table 3. Number of ring layers variation of MRDPSO.

<table>
<thead>
<tr>
<th>Function</th>
<th>MRDPSO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3 ring layers</td>
</tr>
<tr>
<td></td>
<td>Mean</td>
</tr>
<tr>
<td>(f_1)</td>
<td>12.99</td>
</tr>
<tr>
<td>(f_2)</td>
<td>6.83E-15</td>
</tr>
<tr>
<td>(f_3)</td>
<td>0.70E-4</td>
</tr>
<tr>
<td>(f_4)</td>
<td>4.08E-32</td>
</tr>
<tr>
<td>(f_5)</td>
<td>0.28</td>
</tr>
<tr>
<td>(f_6)</td>
<td>14.725</td>
</tr>
<tr>
<td>(f_7)</td>
<td>4063.20</td>
</tr>
</tbody>
</table>

For the dispersed approach for the proposed topology (MRDPSO) (see Table 3) the 5-rings configuration achieved the best overall performance.

The dispersed coefficient improves the convergence speed, thus is clearly expected that MRDPSO will converge faster than MRPSO. Therefore, the 5-rings configuration can be considered the best configuration choice in both MRPSO and MRDPSO.

Varying the number of rings, both MRPSO and MRDPSO can fit to different kinds of problems. Hence, the number of rings can be varied according to the problem to be solved.
6.2 Topologies comparison

The results between the comparison of MR PSO and MRDPSO and other topologies are shown in Table 4 and Table 5, respectively. For all the functions, the fitness values presented by the Multi-ring approaches are the ones that achieved the best performance or similar performance to the best performance.

<table>
<thead>
<tr>
<th>f</th>
<th>MRPSO</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>SD</td>
<td>Mean</td>
<td>SD</td>
<td>Mean</td>
<td>SD</td>
<td>Mean</td>
<td>SD</td>
<td>Mean</td>
<td>SD</td>
</tr>
<tr>
<td>f₁</td>
<td>15.05</td>
<td>7.87</td>
<td>17.712</td>
<td>6.3686</td>
<td>18.215</td>
<td>12.995</td>
<td>39.54</td>
<td>29.36</td>
<td>88.15</td>
<td>44.51</td>
</tr>
<tr>
<td>f₂</td>
<td>2.28E-9</td>
<td>3.53E-9</td>
<td>1.26E-14</td>
<td>3.04E-15</td>
<td>9.61E-8</td>
<td>8.70E-8</td>
<td>1.81E-10</td>
<td>1.86E-10</td>
<td>0.16</td>
<td>0.51</td>
</tr>
<tr>
<td>f₃</td>
<td>0.00509</td>
<td>0.0063</td>
<td>0.01</td>
<td>0.02</td>
<td>0.0031</td>
<td>6.4E-3</td>
<td>0.008</td>
<td>0.01</td>
<td>0.04</td>
<td>0.03</td>
</tr>
<tr>
<td>f₄</td>
<td>1.62E-32</td>
<td>2.23E-32</td>
<td>5.02E-32</td>
<td>5.52E-32</td>
<td>6.22E-33</td>
<td>7.82E-33</td>
<td>9.4E-33</td>
<td>1.27E-32</td>
<td>9.42E-32</td>
<td>5.05E-32</td>
</tr>
<tr>
<td>f₅</td>
<td>0.56</td>
<td>1.03</td>
<td>4.46</td>
<td>8.31</td>
<td>1.33</td>
<td>1.47</td>
<td>1.00</td>
<td>1.64</td>
<td>52.67</td>
<td>39.25</td>
</tr>
<tr>
<td>f₆</td>
<td>16.3</td>
<td>5.19</td>
<td>15.35</td>
<td>5.93</td>
<td>21.94</td>
<td>5.33</td>
<td>19.58</td>
<td>4.89</td>
<td>19.58</td>
<td>7.37</td>
</tr>
<tr>
<td>f₇</td>
<td>2686.44</td>
<td>456.16</td>
<td>2164.17</td>
<td>368.74</td>
<td>2964.79</td>
<td>389.31</td>
<td>2898.37</td>
<td>347.01</td>
<td>3633.34</td>
<td>519.53</td>
</tr>
</tbody>
</table>

Table 4. Comparison between MRPSO and other topologies.

<table>
<thead>
<tr>
<th>f</th>
<th>MRDPSO</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>SD</td>
<td>Mean</td>
<td>SD</td>
<td>Mean</td>
<td>SD</td>
<td>Mean</td>
<td>SD</td>
<td>Mean</td>
<td>SD</td>
</tr>
<tr>
<td>f₁</td>
<td>8.44</td>
<td>5.97</td>
<td>32.8099</td>
<td>33.7074</td>
<td>15.3180</td>
<td>18.0088</td>
<td>10.8321</td>
<td>13.8913</td>
<td>83.8264</td>
<td>54.0686</td>
</tr>
<tr>
<td>f₂</td>
<td>6.83E-15</td>
<td>1.44E-15</td>
<td>1.13E-14</td>
<td>3.60E-15</td>
<td>9.73E-13</td>
<td>5.43E-13</td>
<td>7.54E-13</td>
<td>9.01E-16</td>
<td>0.2947</td>
<td>0.6127</td>
</tr>
<tr>
<td>f₃</td>
<td>0.70E-4</td>
<td>0.01</td>
<td>0.0134</td>
<td>0.0137</td>
<td>4.09E-11</td>
<td>2.24E-10</td>
<td>0.0103</td>
<td>0.014</td>
<td>0.0529</td>
<td>0.0498</td>
</tr>
<tr>
<td>f₅</td>
<td>0.28</td>
<td>0.42</td>
<td>0.7080</td>
<td>1.5170</td>
<td>0.1375</td>
<td>0.22</td>
<td>0.16</td>
<td>0.401</td>
<td>63.50</td>
<td>40.47</td>
</tr>
<tr>
<td>f₇</td>
<td>3969.54</td>
<td>563.79</td>
<td>2320.08</td>
<td>355.31</td>
<td>3610.09</td>
<td>461.8253</td>
<td>3889.64</td>
<td>503.86</td>
<td>3750.91</td>
<td>334.53</td>
</tr>
</tbody>
</table>

Table 5. Comparison between MRDPSO and other topologies.

For function $f₁$ and $f₂$, MRPSO reaches the best fitness values ($F = 15.05$ and $F = 0.56$). For all the other functions used, it reaches values close to the best performance achieved by some well known topologies. A good example is the performance for functions $f₆$ and $f₇$, in which
the only topology that achieved fitness values better than the ones achieved by MRPSO was the global topology. Hence, considering that MRPSO is a local communication topology, the results presented by Table 4 are strongly satisfactory. Table 5 shows the results of the MRDPSO and topologies enabled with the dispersion grade strategy. For function $f_1$, $f_2$, and $f_6$ the MRDPSO achieves the best fitness values ($F=8.44$, $F=6.83E-15$ and $F=13.929$). For the functions $f_3$ and $f_4$, it reaches a good value for the mean fitness with low standard deviation, close to the best performance achieved by well known topologies. For the function $f_7$ MRDPSO achieves a value for the mean fitness in the same order of magnitude of Dispersed PSO with local, Von Neumann and Four Clusters topologies. Therefore, considering all results shown at the Table 5, the MRDPSO achieves good solutions for both unimodal and multimodal problems.

7. Conclusions

This paper has proposed a novel topology for Particle Swarm Optimization based on a multi layer structure based on local communication (ibest). The Multi-ring PSO (MRPSO) and also its combination with the Dispersed PSO approach (MRDPSO) are thoroughly described. The first one introduces a new feature called rotation skill that minimizes the stagnation probability of the PSO algorithm. In the second one, some changes in the communication rules inside the rings were made by using the dispersion grade attribute of Dispersed PSO approach.

The Multi-ring approaches achieved better performance or close to well-known topologies used for comparison. This fact exalts how satisfactory the results involving MRPSO and MRDPSO were. By varying the number of rings, the size of each ring and how the rotation skill work, the proposed approach could be used to reach good quality solutions for both unimodal and multimodal problems. Besides, it helps to avoid the swarm of being trapped at local minima regions, making it moves to new regions of space search in order to help the PSO algorithm to converge to the best solution.

Due to its peculiar features, the proposed PSO topology is an interesting approach able to be applied in different kind of optimization problems.

8. References


This book presents several recent advances on Evolutionary Computation, specially evolution-based optimization methods and hybrid algorithms for several applications, from optimization and learning to pattern recognition and bioinformatics. This book also presents new algorithms based on several analogies and metaphors, where one of them is based on philosophy, specifically on the philosophy of praxis and dialectics. In this book it is also presented interesting applications on bioinformatics, specially the use of particle swarms to discover gene expression patterns in DNA microarrays. Therefore, this book features representative work on the field of evolutionary computation and applied sciences. The intended audience is graduate, undergraduate, researchers, and anyone who wishes to become familiar with the latest research work on this field.

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