A COMPARATIVE STUDY OF NEIGHBORHOOD TOPOLOGIES FOR PARTICLE SWARM OPTIMIZERS

Angelina Jane Reyes Medina[1], Gregorio Toscano Pulido[2] and José Gabriel Ramírez Torres[3]
CINVESTAV-Tamaulipas. Km. 6 carretera Cd. Victoria-Monterrey, Cd. Victoria, Tamaulipas, 87261, Mexico

Keywords: Particle swarm optimization, Neighborhood topologies, Parameter setting.

Abstract: Particle swarm optimization (PSO) is a meta-heuristic that has been found to be very successful in a wide variety of optimization tasks. The behavior of any meta-heuristic for a given problem is directed by both: the variation operators, and the values selected for the parameters of the algorithm. Therefore, it is only natural to expect that not only the parameters, but also the neighborhood topology play a key role in the behavior of PSO. In this paper, we want to analyze whether the type of communication employed to interconnect the swarm accelerates or affects the algorithm convergence. In order to perform a wide study, we selected six different neighborhoods topologies: ring, fully connected, mesh, toroid, tree and star; and two clustering algorithms: k-means and hierarchical. Such approaches were incorporated into three PSO versions: the basic PSO, the Bare-bones PSO (BBPSO) and an extension of BBPSO called BBPSO(EXP). Our results indicate that the convergence rate of a PSO-based approach has an strongly dependence of the topology used. However, we also found that the topology most widely used is not necessarily the best topology for every PSO-based algorithm.

1 INTRODUCTION

The behavior of any meta-heuristic for a given problem is directed by both: the variation operators, and the values selected for the parameters of the algorithm (parameter setting). The parameter setting issue plays a key role on the performance of any meta-heuristic. Tuning well these parameters is a hard problem, since they can usually take several values, and therefore, the number of possible combinations is usually high.

Kennedy & Eberhart (Kennedy and Eberhart, 2001) proposed an approach called "particle swarm optimization" (PSO) which was inspired on the choreography of a bird flock. The approach can be seen as a distributed behavioral algorithm that performs (in its more general version) multidimensional search. In the simulation, the behavior of each individual is affected by either the best local (i.e., within a certain neighborhood) or the best global individual.

Several PSO proposals have been developed in order to improve the performance of the original algorithm (Kennedy, 2003; Kennedy and Eberhart, 2001; Omran et al., 2008). Such approaches have shown that the convergence behavior of PSO is strongly dependent on the values of the inertia weight, the cognitive coefficient and the social coefficient (Clerc and Kennedy, 2002; van den Bergh, 2002). Other proposals have sought to eliminate the dependence of such parameters in order to avoid the parameter setting problem. Investigations within the particle swarm paradigm have found that the particles' interconnection topology interact directly with the function being optimized (Kennedy, 1999). These studies have shown theoretically that the neighborhood topology affects (significantly) the performance of a particle swarm and that the effect depends on the function. Thus, some types of interconnection topologies can work well for some functions, while the same topologies can present problems with other test functions (Kennedy, 1999). Despite the key role that the topology plays in PSO, it has been barely studied.

The remainder of the paper is organized as follows: Section 2 provides an overview of PSO and its variants used in this paper. Neighborhood topologies and clustering algorithms are presented in Section 3. Section 4 presents and discusses the results of the performed experiments. Finally, Section 5 shows the concluding remarks and future work.

Jane Reyes-Medina A., Toscano Pulido G. and Gabriel Ramírez-Torres J. (2009).
A COMPARATIVE STUDY OF NEIGHBORHOOD TOPOLOGIES FOR PARTICLE SWARM OPTIMIZERS.
In *Proceedings of the International Joint Conference on Computational Intelligence*, pages 152-159
DOI: 10.5220/0002324801520159

2 PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) is an stochastic, population-based optimization algorithm proposed by Kennedy and Eberhart in 1995 (Kennedy and Eberhart, 1995), that simulates the social behavior of bird flocks or school fish. In PSO, a swarm of particles fly through hyper-dimensional search space being attracted by both, their personal best position and the best position found so far within a neighborhood. Each particle represent a solution to the optimization problem. The position of each particle is updated using equation (1) which is composed by the best position visited by itself (i.e. its own experience or y in equation (2)) and the position of the best particle in its neighborhood determined by the communication topology used (Kennedy and Mendes, 2002; Kennedy, 1999).

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1), \tag{1}$$

$$\begin{aligned} v_{ij}(t+1) &= wv_{ij}(t) + c_1 r_{1j}(t)(y_{ij}(t) - x_{ij}(t)) \\ &\quad + c_2 r_{2j}(t)(\widehat{y}_{ij}(t) - x_{ij}(t)), \end{aligned} \tag{2}$$

for $i = 1, \cdots, s$ and $j = 1, \cdots, n$

where w is the inertia weight (Shi and Eberhart, 1998), s is the total number of particles in the swarm, n is the dimension of the problem (i.e. the number of parameters of the function being optimized), c_1 and c_2 are the acceleration coefficients, r_{1j}, $r_{2j} \sim U(0,1)$, $\mathbf{x}_i(t)$ is the position of particle i at time step t, $\mathbf{v}_i(t)$ is the velocity of particle i at time step t, $\mathbf{y}_i(t)$ is the personal best position of particle i at time step t, and $\widehat{\mathbf{y}}_\mathbf{i}$ is the neighborhood best position of particle i at time step t.

Empirical and theoretical studies have shown that the convergence behavior of PSO is strongly dependent on the values of the inertia weight and the acceleration coefficients (van den Bergh and Engelbrecht, 2006; Clerc and Kennedy, 2002). Wrong choices of such parameters may produce divergent or cyclic particle trajectories. Several recommendations for values of such parameters have been suggested in the specialized literature (Storn and Price, 1997), although these values are not universally applicable to every kind of problem.

A large number of PSO variations have been developed, mainly to improve the accuracy of solutions, diversity, convergence or to eliminate the parameters dependency (Kennedy and Eberhart, 2001). van den Bergh and Engelbrecht (van den Bergh and Engelbrecht, 2006) and Clerc y Kennedy (Clerc and Kennedy, 2002) proved that each particle converges to a weighted average of its personal best and neighborhood best position, that is,

$$\lim_{t \to +\infty} x_{ij}(t) = \frac{c_1 y_{ij} + c_2 \widehat{y}_{ij}}{c_1 + c_2} \tag{3}$$

This theoretically derived behavior provides support for the Bare-bones PSO (BBPSO). BBPSO was proposed by Kennedy in 2003 (Kennedy, 2003). The BBPSO replaces the equation (1) and (2) with equation (4),

$$x_{ij}(t+1) = N\Big(\frac{y_{ij}(t) + \widehat{y}_j(t)}{2}, |y_{ij}(t) - \widehat{y}_j(t)|\Big) \tag{4}$$

Particle positions are therefore randomly selected from N which is a Gaussian distribution with: mean, equal to the average weighted of its personal best and the global best positions (i.e. the swarm attractor) and; deviation $y_{ij}(t) - \widehat{y}_j(t)$ which approximate zero as t increases.

Kennedy also proposed an alternative version of the BBPSO (EXP). He replaced the equations (1) and (2) with equation (5),

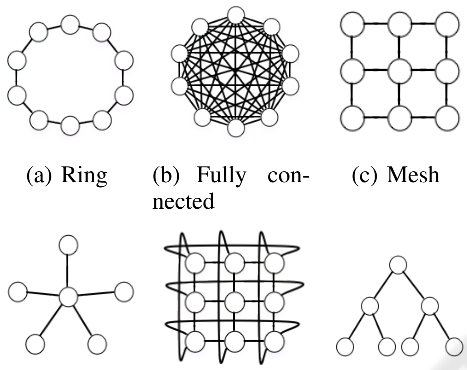
$$x_{ij}(t+1) = \begin{cases} N\big(\frac{y_{ij}(t)+\widehat{y}_j(t)}{2}, |y_{ij}(t) - \widehat{y}_j(t)|\big) & \text{if } U(0,1) \\ & > 0.5 \\ y_{ij}(t) & \text{otherwise} \end{cases} \tag{5}$$

Based on the above equation, there is a 50% chance that the j the dimension of the particle dimension changes to the corresponding personal best position. This version of PSO biases towards exploiting personal best positions.

3 DESCRIPTION OF OUR EXPERIMENT

In PSO, each particle inside of the swarm belongs to an specific communication neighborhood. Therefore, it was natural that several studies were performed in order to determine whether the neighborhood topology could affect the convergence (Kennedy, 1999; Kennedy and Mendes, 2002; Jian et al., 2004). These studies relied on theoretical proposals and implementations of neighborhood topologies commonly used by PSO. In such studies, some neighborhood topologies have performed better than others (Kennedy, 1999). However, only a few topologies and problems were tested at a time. Therefore, our hypothesis to perform this study was that the topology used in a particle swarm might affect the rate and degree to which the swarm is attracted towards a particular region. Thus, the present study focused on several swarm topologies, where connections were undirected, unweighted, and they no vary over the course of a trial. The neighborhoods topologies were constructed based on the index of each particle, then each particle has a unique identifier in the entire population. We also decided to study how clustering algorithms can improve the performance of the PSO. In clustering algorithms, the euclidean distance were used as a measure and the connections were updated dynamically on each iteration of the trail.

Since this paper analyzes the ring, fully connected, mesh, toroidal, tree, star topologies shown in Figure 1 and two clustering algorithms: k-means and hierarchical in order to determined whether the type of communication employed to interconnect the swarm accelerates or affects the algorithm convergence. We described them below:



(a) Ring (b) Fully connected (c) Mesh

(d) Star (e) Toroidal (f) Tree

Figure 1: Neighborhood topologies used in this study.

3.1 Ring Topology

The ring topology is also known as the *lbest* version in PSO (see Figure 1(a)). In this topology each particle is affected by the best performance of its k immediate neighbors in the topological population. In one common *lbest* case, $k = 2$, the individual is affected by only its immediately adjacent neighbors.

In the ring topology, the neighbors are closely connected and thus, they react when one particle has a raise in its fitness, this reaction dilutes proportionally with respect to the distance. Thus, it is possible that one segment of the population might converge on a local optimum, while another segment of the population might converge to a different point or remain searching. However, the optima will eventually pull the swarm.

3.2 Fully Connected Topology

Fully connected topology is also known as the full topology (see Figure 1(b)). All nodes in this topology are directly connected among each other. In PSO this topology is also known as the PSO's *gbest* version, in which all particles in the entire swarm direct their flight toward the best particle found in the whole population (i.e. every particle is attracted to the best solution found by any member of the swarm). That is,

$$\hat{\mathbf{y}}_i(t) \in \{\mathbf{y}_0(t), \mathbf{y}_1(t), \ldots, \mathbf{y}_s(t)\} = min\{f(\mathbf{y}_0(t)), f(\mathbf{y}_1(t)), \ldots, f(\mathbf{y}_s(t))\}, \quad (6)$$

Kennedy et al. suggested (Kennedy and Eberhart, 2001; Kennedy and Mendes, 2002) that *gbest* populations tend to converge more rapidly to a optima than *lbest* populations, but also, they are more susceptible to converge to a local optima. However, this topology is the most used by far.

3.3 Star Topology

In *star* topology, the information passes through only one individual (see Figure 1(d)). One central node influences and it is influenced by all other members of the population.

In this article, the central particle of the star topology is selected randomly. In each time step t, all particles of the entire swarm directs their flight toward one particle (the central particle), and the central particle directs its flight toward the best particle of the neighborhood. The star topology, effectively isolates individuals from each other, since information has to be communicated through the central node. This central node compares the performance of every individual in the population and adjusts its own trajectory toward the best of them. Thus the central individual serves as a kind of buffer or filter, slowing the speed of transmission of good solutions through the population. The buffering effect of the central particle should prevent premature convergence on local optima; this is a way to preserve diversity of potential problem solutions, though, it was expected that it might destroy population's collaboration ability.

3.4 Mesh Topology

In this type of topology (see Figure 1(c)), one node is connected to several nodes, commonly each node is connected to four neighbors (in this case, we connect each node to the ones which are in the north, south, east and west of the particle's location).

In the mesh topology, the particles in the corners are connected with its two adjacent neighbors. The particles on the mesh's boundaries will have tree adjacent neighbors and the particles on the mesh's center will have four adjacent neighbors. Thus, there exists overlapping neighbors in each particle, allowing redundancy in the search process. The particles will be assigned to each node of the mesh from left to right and top-down. The mesh remains the same form throughout the algorithm's execution.

3.5 Tree Topology

It is also known as a hierarchical topology (see Figure 1(f)). This topology has a central root node (the top level of the hierarchy) which is connected to one or more individuals that are one level lower in the hierarchy (i.e., the second level), while each of the second level individuals that are connected to the top level central root individual will also have one or more individuals which are one level lower in the hierarchy (i.e., the third level) connected to it (the hierarchy of the tree is symmetrical).

The tree topology is constructed as a binary tree (using the particle's index as nodes), the root node is selected randomly among swarm and the remaining particles are distributed in the tree branches. The nodes (particles) in the tree must be, as possible, balanced in the tree branches. The root node searches for the best fitness obtained by their children (i.e. the second level) to redirected its flight. The second level nodes search for the best fitness found by both, children and father.

3.6 Toroidal Topology

Topologically, a torus is a closed surface defined as the product of two circles (see Figure 1(e)). This topological torus is often called as Clifford torus.

The toroidal topology is similar to the mesh topology, except that all particles in the swarm have four adjacent neighbors. As is shown in Figure 1(e), the toroidal topology connects every corner particle with its symmetrical neighbor. The same occurs with the toroid boundaries. The assignment from particles to nodes will be similar to the mesh topology assignment.

3.7 Clustering Algorithms

Clustering is defined by the average number of neighbors that any two connected nodes have in common (Kennedy, 1999). In PSO, the particles naturally cluster in more than one region of the search space usually indicate the presence of local optima. It seems reasonable to investigate whether information about the distribution of particles in the search space could be exploited to improve particle trajectories (Kennedy and Eberhart, 2001).

We have implemented two types of clustering algorithms: k-means and hierarchical clustering.

- K-means clustering algorithm:
  Given a data set through a certain number of clusters (assume k clusters) fixed *a priori*. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes a different result. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. In PSO, we assume to have 4 clusters. We used the euclidean distance, in order to associate a particle to its nearest centroid.
- Hierarchical clustering algorithm:
  Hierarchical clustering considers the distance between one cluster and another cluster to be equal to the shortest distance from any member of one cluster to any member of the other cluster. If the data consists of similarities, then hierarchical clustering considers the similarity between one cluster and another cluster to be equal to the greatest similarity from any member of one cluster to any member of the other cluster.

  In order to implement the hierarchical clustering algorithm into PSO, we defined 4 clusters to be searched for. The shortest distance between particles were calculated using the euclidean distance.

3.8 Test Functions

Nine test functions were selected from the specialized literature. Such test functions are described below:

A. *Sphere* function, defined as
$f(\mathbf{x}) = \sum_{i=1}^{N_d} x_i^2$,
where $\mathbf{x}^* = 0$ and $f(\mathbf{x}^*) = 0$ for $-100 \leq x_i \leq 100$

B. *Schwefel's* problem , defined as
$f(\mathbf{x}) = \sum_{i=1}^{N_d} |x_i| + \prod_{i=1}^{N_d} |x_i|$,
where $\mathbf{x}^* = 0$ and $f(\mathbf{x}^*) = 0$ for $-10 \leq x_i \leq 10$

C. *Step* function, defined as:
$f(\mathbf{x}) = \sum_{i=1}^{N_d} (\lfloor x_i + 0.5 \rfloor)^2$,
where $\mathbf{x}^* = 0$ and $f(\mathbf{x}^*) = 0$ for $-100 \leq x_i \leq 100$

D. *Rosenbrock* function, defined as
$f(\mathbf{x}) = \sum_{i=1}^{N_d-1} (100(x_i - x_{i-1}^2)^2 + (x_{i-1} - 1)^2)$,
where $\mathbf{x}^* = (1, 1, \ldots, 1)$ and $f(\mathbf{x}^*) = 0$ for $-30 \leq x_i \leq 30$

E. *Rotated hyper-ellipsoid* function, defined as
$f(\mathbf{x}) = \sum_{i=1}^{N_d} (\sum_{j=1}^{i} x_j)^2$,
where $\mathbf{x}^* = 0$ and $f(\mathbf{x}^*) = 0$ for $-100 \leq x_i \leq 100$

F. *Generalized Schwefel* Problem 2.26, defined as:
$f(\mathbf{x}) = -\sum_{i=1}^{N_d} (x_i \, sin(\sqrt{|x_i|}))$,
where $\mathbf{x}^* = (420.9687, \ldots, 420.9687)$ and $f(\mathbf{x}^*) = -4426.407721$ for $-500 \leq x_i \leq 500$

G. *Rastrigin* function, defined as:
$f(\mathbf{x}) = -\sum_{i=1}^{N_d}(x_i^2 - 10cos(2\pi x_i) + 10)$,
where $\mathbf{x}^* = 0$ and $f(\mathbf{x}^*) = 0$ for $-5.12 \leq x_i \leq 5.12$

H. *Ackley's* function, defined as
$f(\mathbf{x}) = -20exp\Big(-0.2\sqrt{\frac{1}{30}\sum_{i=1}^{N_d}x_i^2}\Big) - exp\Big(\frac{1}{30}\sum_{i=1}^{N_d}cos(2\pi x_i)\Big) + 20 + e$,
where $\mathbf{x}^* = 0$ and $f(\mathbf{x}^*) = 0$ for $-32 \leq x_i \leq 32$

I. *Griewank* function, defined as:
$f(\mathbf{x}) = \frac{1}{4000}\sum_{i=1}^{N_d}x_i^2 - \prod_{i=1}^{N_d}cos\Big(\frac{x_i}{\sqrt{i}}\Big) + 1$,
where $\mathbf{x}^* = 0$ and $f(\mathbf{x}^*) = 0$ for $-600 \leq x_i \leq 600$

4 DISCUSSION OF RESULTS

This section compares the performance of the basic PSO, BBPSO and BBPSO (EXP) algorithms discussed in Section 2. We implemented: ring, full, star, mesh, toroidal and tree neighborhood topologies and; k-means and hierarchical clustering algorithms on each PSO algorithm. It is important to note that neighborhood topologies were determined using particle indexes and were not based on any spatial information. For both clustering algorithms, the euclidean distance (spatial information) has been used to determined the distance among particles.

For the basic PSO algorithm, we used $w = 0.72$ and $c_1 = c_2 = 1.49$. These values have been shown to provide good results (Clerc and Kennedy, 2002; van den Bergh, 2002; van den Bergh and Engelbrecht, 2006).

For all the algorithms used in this section, the swarm size was $s = 50$. 200 iterations were perform by each algorithm (24 algorithms, since there were implemented 6 topologies + 2 clustering techniques in 3 PSO variants). Every resulting approach were executed 30 independent runs. These values were used as defaults for all experiments which use static control parameters. Also, the distribution of the particles were 10×5 when the mesh and toroidal topologies were used. For hierarchical and k-means clustering algorithms, 4 groups were asked for.

Since there are 3 algorithms, 8 different neighborhood topologies, and 9 test functions, it was difficult to show numerical results. In order to present such results in a friendly-comparison way, we selected to show the numerical results as box-plot graphics. In Figures 2 and 3 shows such results.

From Figures 2 and 3, it is easily to see that the BBPSO (EXP) version is highly dependent of the interconnection topology, since it only presented a good behavior when the fully connected, star and hierarchical approaches were used. However, when the remain topologies were used, it presented the worst results among the three algorithms. Therefore, we can say that PSO and BBPSO algorithms showed a more robust behavior.

When PSO was used, the best topologies which showed better results were toroidal, ring and mesh (see Figures 2(a), 2(b), 2(c), 2(e), 3(b), 3(c), 3(d), which corresponds to the sphere, schwefel, step, rotated hyper-ellipsoid, ratrigin, ackley and griewank test functions, respectively). For the generalized schwefel test function (see Figure 3(a)) the k-means clustering algorithm was the approach which perform the best. For the Rosenbrock test function all topologies and clustering algorithms performed well. In summary, when toroidal, mesh and ring topologies were used, PSO presented an good performance. When BBPSO was used, the topologies which performed better were full, star and hierarchical clustering algorithms in all test functions (see Figures 2 and 3).

In our opinion, BBPSO presented the best performance, since it outperformed the other two PSO approaches in six out of nine test functions (see results shown in Figures 2(a), 2(b), 2(c), 3(b), 3(c) and 3(d) which corresponds to sphere, schwefel, step, ratrigin, ackley and griewank test functions). BBPSO obtained similar results with respect to the results obtained by PSO in Rosenbrock test function (see results shown in Figure 2(d)). The original PSO algorithm outperform the others two PSO approaches in two out of nine test functions (see results shown in Figures 2(e) and 3(a) which corresponds to the rotated hyper-ellipsoid and the generalized schwefel test functions).

From our results, we can conclude that, the topology plays a key role in PSO. The original PSO approach should be used with the toroidal, mesh and ring topologies, whilst BBPSO should be used with the fully connected, star or hierarchical clustering methods.

5 CONCLUSIONS AND FUTURE WORK

Our main conclusions are the following:

- We found that the use of mesh, toroidal and ring topologies promote better convergence rates in the PSO algorithm.
- The use of the fully connected, star and hierarchical clustering approaches promote better convergence rates in the BBPSO algorithm.
- The topology most widely used (fully connected topology) did not perform well in PSO algorithm

whilst presented a good performance in BBPSO.

- Ring topology (which it is the another topology widely used) presented a good convergence rate.
- The good selection of a topology can increase the performance of a PSO-based algorithm.

Some possible paths to extend this work are the following:

- Experiment with other PSO's models and differential evolution algorithms.
- To include the parameter's values w, c_1 and c_2 in a similar study, in order to identify the relation among parameters (including the topology).

ACKNOWLEDGEMENTS

The first author acknowledges support from CONACyT through a scholarship to pursue graduate studies at the Information Technology Laboratory at CINVESTAV-IPN. The second author gratefully acknowledges support from CONACyT through project 90548. Also, This research was partially funded by project number 51623 from "Fondo Mixto Conacyt-Gobierno del Estado de Tamaulipas"

REFERENCES

Clerc, M. and Kennedy, J. (2002). The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6(1):58–73.

Jian, W., Xue, Y., and Qian, J. (2004). Improved particle swarm optimization algoritnms study based on the neighborhoods topologies. In *The 30th Annual Conference of the IEEE Industrial Electronics Society,2004. IECON 2004*, volume 3, pages 2192–2196, Busan, Korea.

Kennedy, J. (1999). Small worlds and mega-mind: Effects of neighborhood topology on particle swarm performance. In *Proceedings of the IEEE Congress on Evolutionary Computation, 1999. CEC 99*, volume 3, pages 1931–1938, Washington, DC, USA.

Kennedy, J. (2003). Bare bones particle swarms. In *Proceedings of the IEEE Swarm Intelligence Symposium, 2003. SIS '03*, pages 80–87, Piscataway, NJ. IEEE Press.

Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of the IEEE International Joint Conference on Neural Networks*, pages 1942–1948, Piscataway, NJ. IEEE Press.

Kennedy, J. and Eberhart, R. C., editors (2001). *Swarm Intelligence*. Morgan Kaufmann, San Francisco, California.

Kennedy, J. and Mendes, R. (2002). Population structure and particle performance. In *Proceedings of the IEEE Congress on Evolutionary Computation, 2002. CEC '02*, pages 1671–1676, Washington, DC, USA. IEEE Computer Society.

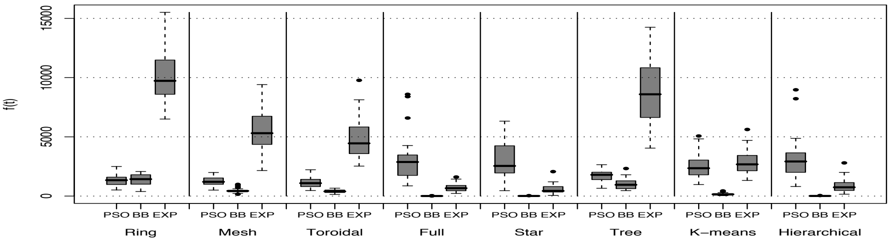
Omran, M., Engelbrecht, A., and Salman, A. (2008). Bare bones differential evolution. *European Journal of Operational Research*, 196(1):128–139.

Shi and Eberhart, R. (1998). A modified particle swarm optimizer. In *Proceedings of the IEEE Congress on Evolutionary Computation, 1998*, pages 69–73, Anchorage, AK, USA.

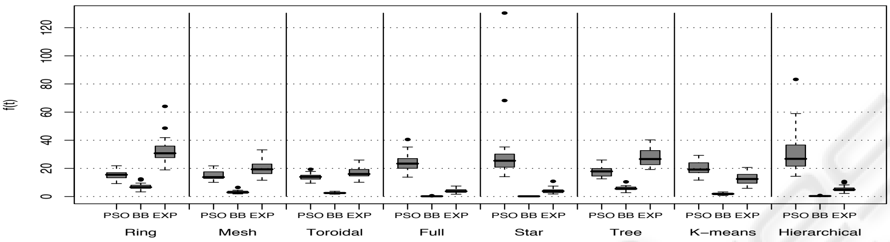
Storn, R. and Price, K. (1997). Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):359–431.

van den Bergh, F. (2002). *An Analysis of Particle Swarm Optimizers*. PhD thesis, Department of Computer Science, University of Pretoria, Pretoria, South Africa.

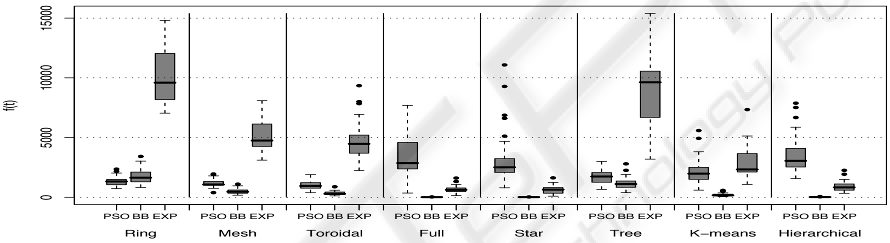
van den Bergh, F. and Engelbrecht, A. (2006). A study of particle swarm optimization particle trajectories. *Information sciences*, 176 (8):937–971.



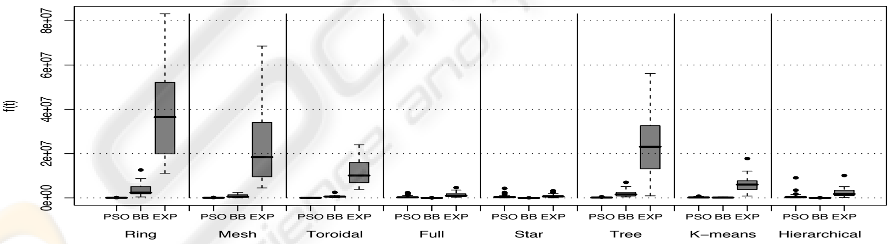
(a) Sphere test function



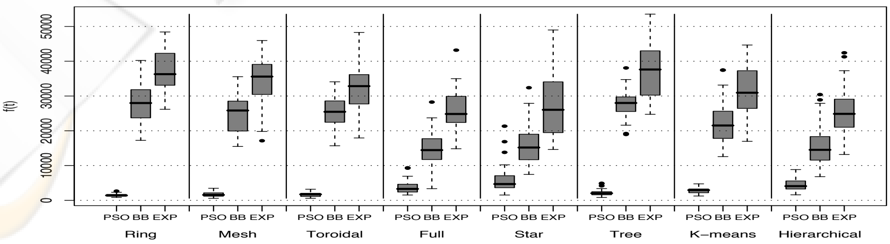
(b) Schwefel test function



(c) Step test function

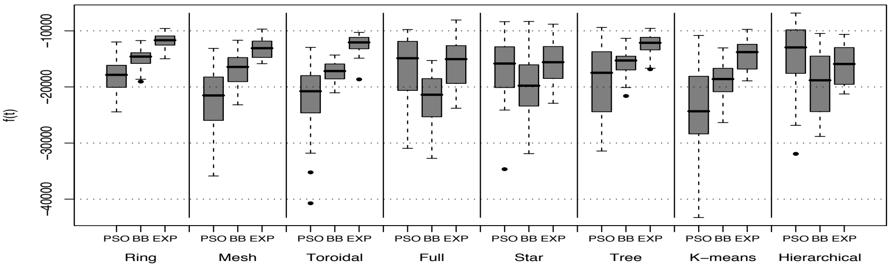


(d) Rosenbrock

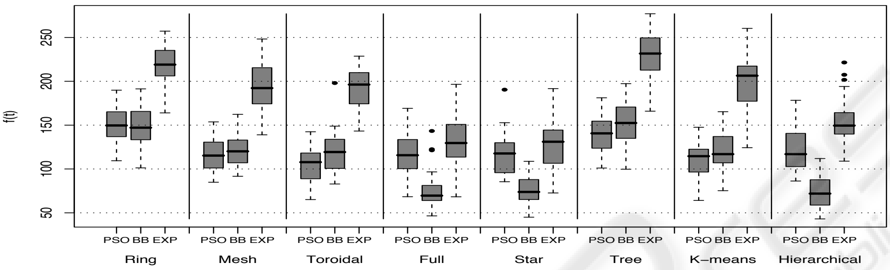


(e) Rotated hyper-ellipsoid function

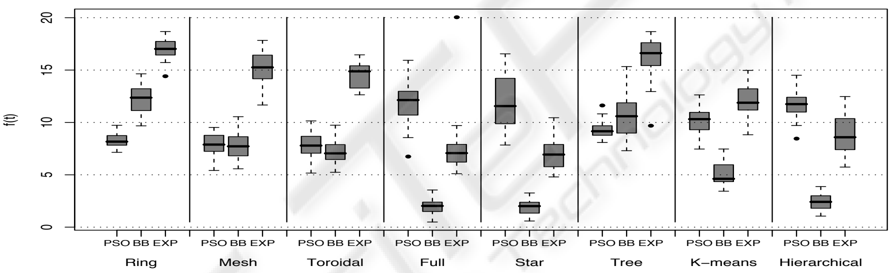
Figure 2: Box-plots produced from the results of 30 independent runs for the ring, mesh, toroidal fully connected, star, and tree topologies; and k-means and hierarchical clustering algorithms.



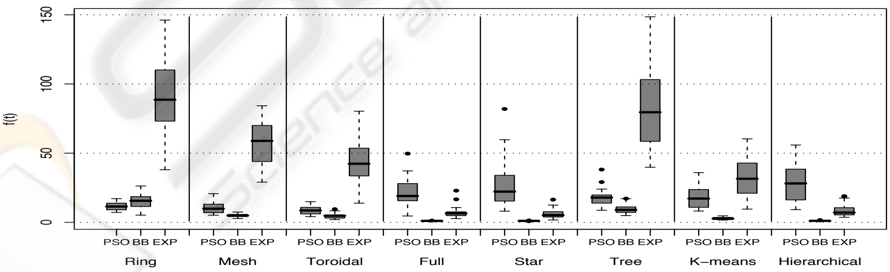
(a) Generalized Schwefel test function



(b) Ratrigin test function



(c) Ackley test function



(d) Griewank test function

Figure 3: Box-plots produced from the results of 30 independent runs for the ring, mesh, toroidal fully connected, star, and tree topologies; and k-means and hierarchical clustering algorithms.