

UPCommons

Portal del coneixement obert de la UPC

<http://upcommons.upc.edu/e-prints>

Sakamoto, S. [et al.] (2016) Node placement in Wireless Mesh Networks: a comparison study of WMN-SA and WMN-PSO simulation systems. *2016 19th International Conference on Network-Based Information Systems, NBIS 2016, Technical University of Ostrava, Ostrava, Czech Republic, 7-9 September 2016*. IEEE. Pp. 1-8. Doi: <http://dx.doi.org/10.1109/NBiS.2016.31>.

© 2016 IEEE. Es permet l'ús personal d'aquest material. S'ha de demanar permís a l'IEEE per a qualsevol altre ús, incloent la reimpressió/reedició amb fins publicitaris o promocionals, la creació de noves obres col·lectives per a la revenda o redistribució en servidors o llistes o la reutilització de parts d'aquest treball amb drets d'autor en altres treballs.

Sakamoto, S. [et al.] (2016) Node placement in Wireless Mesh Networks: a comparison study of WMN-SA and WMN-PSO simulation systems. *2016 19th International Conference on Network-Based Information Systems, NBIS 2016, Technical University of Ostrava, Ostrava, Czech Republic, 7-9 September 2016*. IEEE. Pp. 1-8. Doi: [http://dx.doi.org/ 10.1109/NBIS.2016.31](http://dx.doi.org/10.1109/NBIS.2016.31).

(c) 2016 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other users, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted components of this work in other works.

Node Placement in Wireless Mesh Networks: A Comparison Study of WMN-SA and WMN-PSO Simulation Systems

Shinji Sakamoto*, Tetsuya Oda[†], Makoto Ikeda[†], Leonard Barolli[†], Fatos Xhafa[‡] and Isaac Woungang[§]

**Graduate School of Engineering,
Fukuoka Institute of Technology (FIT)
3-30-1 Wajiro-Higashi, Higashi-Ku, Fukuoka 811-0295, Japan
E-mail: shinji.t.sakamoto@gmail.com*

*[†]Department of Information and Communication Engineering,
Fukuoka Institute of Technology (FIT)
3-30-1 Wajiro-Higashi, Higashi-Ku, Fukuoka 811-0295, Japan
E-mail: oda.tetsuya.fit@gmail.com, makoto.ikd@acm.org, barolli@fit.ac.jp*

*[‡]Department of Languages and Informatics Systems,
Technical University of Catalonia
C/Jordi Girona 1-3, 08034 Barcelona, Spain
E-mail: fatos@lsi.upc.edu*

*[§]Department of Computer Science,
Ryerson University
350, Victoria street, Toronto Ontario, M5B 2K3, Canada
E-mail: iwoungan@scs.ryerson.ca*

Abstract—With the fast development of wireless technologies, Wireless Mesh Networks (WMNs) are becoming an important networking infrastructure due to their low cost and increased high speed wireless Internet connectivity. In our previous work, we implemented a simulation system based on Simulated Annealing (SA) for solving node placement problem in wireless mesh networks, called WMN-SA. Also, we implemented a Particle Swarm Optimization (PSO) based simulation system, called WMN-PSO. In this paper, we compare two systems considering calculation time. From the simulation results, when the area size is 32×32 and 64×64 , WMN-SA is better than WMN-PSO. When the area size is 128×128 , WMN-SA performs better than WMN-PSO. However, WMN-SA needs more calculation time than WMN-PSO.

Keywords—Wireless Mesh Networks, Simulated Annealing, Particle Swarm Optimization, Node Placement, Calculation Time.

I. INTRODUCTION

The wireless networks and devices are becoming increasingly popular and they provide users access to information and communication anytime and anywhere [1]–[11]. Wireless Mesh Networks (WMNs) are gaining a lot of attention because of their low cost nature that makes them attractive for providing wireless Internet connectivity. A WMN is dynamically self-organized and self-configured, with the nodes in the network automatically establishing and maintaining mesh connectivity among them-selves (creating,

in effect, an ad hoc network). This feature brings many advantages to WMNs such as low up-front cost, easy network maintenance, robustness and reliable service coverage [12]. Moreover, such infrastructure can be used to deploy community networks, metropolitan area networks, municipal and corporative networks, and to support applications for urban areas, medical, transport and surveillance systems.

Mesh node placement in WMN can be seen as a family of problems, which are shown (through graph theoretic approaches or placement problems, e.g. [13], [14]) to be computationally hard to solve for most of the formulations [15]. In fact, the node placement problem considered here is even more challenging due to two additional characteristics: (a) locations of mesh router nodes are not pre-determined (any available position in the considered area can be used for deploying the mesh routers) and (b) routers are assumed to have their own radio coverage area. Here, we consider the version of the mesh router nodes placement problem in which we are given a grid area where to deploy a number of mesh router nodes and a number of mesh client nodes of fixed positions (of an arbitrary distribution) in the grid area. The objective is to find a location assignment for the mesh routers to the cells of the grid area that maximizes the network connectivity and client coverage.

Node placement problems are known to be computationally hard to solve [16]–[18]. In some previous works,

intelligent algorithms have been recently investigated [19]–[27].

In this work, we consider as metrics for optimization the Size of Giant Component (SGC) and the Number of Covered Mesh Clients (NCMC). We compare the following two simulation systems for solving node placement problem in WMN considering calculation time:

- Simulated Annealing (SA) based system;
- Particle Swarm Optimization (PSO) based system.

The rest of the paper is organized as follows. The mesh router nodes placement problem is defined in Section II. We present our proposed and implemented simulation systems in Section III. The simulation results are given in Section IV. Finally, we give conclusions and future work in Section V.

II. NODE PLACEMENT PROBLEM IN WMNS

For this problem, we have a grid area arranged in cells we want to find where to distribute a number of mesh router nodes and a number of mesh client nodes of fixed positions (of an arbitrary distribution) in the considered area. The objective is to find a location assignment for the mesh routers to the area that maximizes the network connectivity and client coverage. Network connectivity is measured by SGC of the resulting WMN graph, while the user coverage is simply the number of mesh client nodes that fall within the radio coverage of at least one mesh router node and is measured by NCMC.

An instance of the problem consists as follows.

- N mesh router nodes, each having its own radio coverage, defining thus a vector of routers.
- An area $W \times H$ where to distribute N mesh routers. Positions of mesh routers are not pre-determined and are to be computed.
- M client mesh nodes located in arbitrary points of the considered area, defining a matrix of clients.

It should be noted that network connectivity and user coverage are among most important metrics in WMNs and directly affect the network performance.

In this work, we have considered a bi-objective optimization in which we first maximize the network connectivity of the WMN (through the maximization of the SGC) and then, the maximization of the NCMC.

In fact, we can formalize an instance of the problem by constructing an adjacency matrix of the WMN graph, whose nodes are router nodes and client nodes and whose edges are links between nodes in the mesh network. Each mesh node in the graph is a triple $\nu = \langle x, y, r \rangle$ representing the 2D location point and r is the radius of the transmission range. There is an arc between two nodes u and ν , if ν is within the transmission circular area of u .

Algorithm 1 : Pseudo-code of SA.

```

 $t := 0$ 
Initialize  $T$ 
 $s0 := \text{Initial\_Solution}()$ 
 $v0 := \text{Evaluate}(s0)$ 
while (stopping condition not met) do
  while  $t \bmod \text{MarkovChainLen} = 0$  do
     $t := t+1$ 
     $s1 := \text{Generate}(s0, T)$  //Move
     $v1 := \text{Evaluate}(s1)$ 
    if  $\text{Accept}(v0, v1, T)$  then
       $s0 := s1$ 
       $v0 := v1$ 
    end if
  end while
   $T := \text{Update}(T)$ 
end while
return  $s0$ 

```

III. PROPOSED SIMULATION SYSTEMS

A. Simulated Annealing

1) *Description of Simulated Annealing:* SA algorithm [28] is a generalization of the metropolis heuristic. Indeed, SA consists of a sequence of executions of metropolis with a progressive decrement of the temperature starting from a rather high temperature, where almost any move is accepted, to a low temperature, where the search resembles Hill Climbing. In fact, it can be seen as a hill-climber with an internal mechanism to escape local optima (see pseudo-code in Algorithm 1). In SA, the solution s' is accepted as the new current solution if $\delta \leq 0$ holds, where $\delta = f(s') - f(s)$. To allow escaping from a local optimum, the movements that increase the energy function are accepted with a decreasing probability $\exp(-\delta/T)$ if $\delta > 0$, where T is a parameter called the “temperature”. The decreasing values of T are controlled by a *cooling schedule*, which specifies the temperature values at each stage of the algorithm, what represents an important decision for its application (a typical option is to use a proportional method, like $T_k = \alpha \cdot T_{k-1}$). SA usually gives better results in practice, but uses to be very slow. The most striking difficulty in applying SA is to choose and tune its parameters such as initial and final temperature, decrements of the temperature (cooling schedule), equilibrium and detection.

Evaluation of fitness function: An important aspect is the determination of an appropriate objective function and its encoding. In our case, the fitness function follows a hierarchical approach in which the main objective is to maximize the size of giant component in WMN.

Neighbor selection and movement types: The neighborhood $N(s)$ of a solution s consists of all solutions that are accessible by a local move from s . We have considered

three different types of movements. The first, called *Random*, consists in choosing a router at random in the grid area and placing it in a new position at random. The second move, called *Radius*, chooses the router of the largest radio and places it at the center of the most densely populated area of client mesh nodes. Finally, the third move, called *Swap*, consists in swapping two routers: the one of the smallest radio situated in the most densely populated area of client mesh nodes with that of largest radio situated in the least densely populated area of client mesh nodes. The aim is that largest radio routers should serve to more clients by placing them in more dense areas.

We also considered the possibility to combine the above movements in sequences of movements. The idea is to see if the combination of these movements offers some improvement over the best of them alone. We called this type of movement *Combination*:

$$\begin{aligned} &< \text{Random}_1, \dots, \text{Random}_k; \\ &\quad \text{Radius}_1, \dots, \text{Radius}_k; \\ &\quad \text{Swap}_1, \dots, \text{Swap}_k >, \end{aligned}$$

where k is a user specified parameter.

2) *Acceptability Criteria*: The acceptability criteria for newly generated solution is based on the definition of a threshold value (accepting threshold) as follows. We consider a succession t_k such that $t_k > t_{k+1}$, $t_k > 0$ and t_k tends to 0 as k tends to infinity. Then, for any two solutions s_i and s_j , if $\text{fitness}(s_j) - \text{fitness}(s_i) < t_k$, then accept solution s_j .

For the SA, t_k values are taken as accepting threshold but the criterion for acceptance is probabilistic:

- If $\text{fitness}(s_j) - \text{fitness}(s_i) \leq 0$ then s_j is accepted.
- If $\text{fitness}(s_j) - \text{fitness}(s_i) > 0$ then s_j is accepted with probability $\exp[(\text{fitness}(s_j) - \text{fitness}(s_i))/t_k]$ (at iteration k the algorithm generates a random number $R \in (0, 1)$ and s_j is accepted if $R < \exp[(\text{fitness}(s_j) - \text{fitness}(s_i))/t_k]$).

In this case, each neighbour of a solution has a positive probability of replacing the current solution. The t_k values are chosen in way that solutions with large increase in the cost of the solutions are less likely to be accepted (but there is still a positive probability of accepting them).

B. PSO

In PSO a number of simple entities (the particles) are placed in the search space of some problem or function and each evaluates the objective function at its current location. The objective function is often minimized and the exploration of the search space is not through evolution [29]. However, following a widespread practice of borrowing from the evolutionary computation field, in this work, we consider the bi-objective function and fitness function interchangeably. Each particle then determines its movement through

Algorithm 2 Pseudo code of PSO.

```

/* Generate the initial solutions and parameters */
Computation maxtime:=  $T_{max}$ ,  $t = 0$ ;
Number of particle-patterns:=  $m$ ,  $2 \leq m \in \mathbf{R}^1$ ;
Particle-patterns initial solution:=  $\mathbf{P}_i^0$ ;
Global initial solution:=  $\mathbf{G}^0$ ;
Particle-patterns initial position:=  $\mathbf{x}_{ij}^0$ ;
Particles initial velocity:=  $\mathbf{v}_{ij}^0$ ;
PSO parameter:=  $\omega$ ,  $0 < \omega \in \mathbf{R}^1$ ;
PSO parameter:=  $C_1$ ,  $0 < C_1 \in \mathbf{R}^1$ ;
PSO parameter:=  $C_2$ ,  $0 < C_2 \in \mathbf{R}^1$ ;
/* Start PSO */
Evaluate( $\mathbf{G}^0, \mathbf{P}^0$ );
/* "Evaluate" does calculate present fitness value of each
Particle-patterns. */
while  $t < T_{max}$  do
    /* Update velocities and positions */
     $\mathbf{v}_{ij}^{t+1} = \omega \cdot \mathbf{v}_{ij}^t$ 
         $+ C_1 \cdot \text{rand}() \cdot (\text{best}(\mathbf{P}_{ij}^t) - \mathbf{x}_{ij}^t)$ 
         $+ C_2 \cdot \text{rand}() \cdot (\text{best}(\mathbf{G}^t) - \mathbf{x}_{ij}^t)$ ;
     $\mathbf{x}_{ij}^{t+1} = \mathbf{x}_{ij}^t + \mathbf{v}_{ij}^{t+1}$ ;
    Update_Solutions( $\mathbf{G}^t, \mathbf{P}^t$ );
    /* "Update_Solutions" compares and updates the
    Particle-pattern's best solutions and the global best
    solutions if their fitness value is better than previous.
    */
    Evaluate( $\mathbf{G}^{(t+1)}, \mathbf{P}^{(t+1)}$ );
     $t = t + 1$ ;
end while
Update_Solutions( $\mathbf{G}^t, \mathbf{P}^t$ );
return Best found pattern of particles as solution;

```

the search space by combining some aspect of the history of its own current and best (best-fitness) locations with those of one or more members of the swarm, with some random perturbations. The next iteration takes place after all particles have been moved. Eventually the swarm as a whole, like a flock of birds collectively foraging for food, is likely to move close to an optimum of the fitness function.

Each individual in the particle swarm is composed of three \mathcal{D} -dimensional vectors, where \mathcal{D} is the dimensionality of the search space. These are the current position \vec{x}_i , the previous best position \vec{p}_i and the velocity \vec{v}_i .

The particle swarm is more than just a collection of particles. A particle by itself has almost no power to solve any problem; progress occurs only when the particles interact. Problem solving is a population-wide phenomenon, emerging from the individual behaviors of the particles through their interactions. In any case, populations are organized according to some sort of communication structure or topology, often thought of as a social network. The topology typically consists of bidirectional edges connecting pairs of

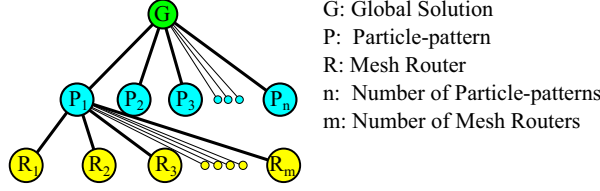


Figure 1. Relationship among global solution, particle-patterns and mesh routers.

particles, so that if j is in i 's neighborhood, i is also in j 's. Each particle communicates with some other particles and is affected by the best point found by any member of its topological neighborhood. This is just the vector \vec{p}_i for that best neighbor, which we will denote with \vec{p}_g . The potential kinds of population "social networks" are hugely varied, but in practice certain types have been used more frequently.

In the PSO process, the velocity of each particle is iteratively adjusted so that the particle stochastically oscillates around \vec{p}_i and \vec{p}_g locations.

We propose and implement a new simulator that uses PSO algorithm to solve the node placement problem in WMNs. We call this simulator WMN-PSO. Our system can generate instances of the problem using different iterations of clients and mesh routers.

We present here the particularization of the PSO algorithm (see Algorithm 2) for the mesh router node placement problem in WMNs.

Initialization: Our proposed system starts by generating an initial solution randomly, by *ad hoc* methods [30]. We decide the velocity of particles by a random process considering the area size. For instance, when the area size is $W \times H$, the velocity is decided randomly from $-\sqrt{W^2 + H^2}$ to $\sqrt{W^2 + H^2}$.

Particle-pattern: A particle is a mesh router. A fitness value of a particle-pattern is computed by combination of mesh routers and mesh clients positions. In other words, each particle-pattern is a solution as shown in Fig. 1. Therefore, the number of particle-patterns is a number of solutions.

Fitness function: One of most important thing in PSO algorithm is to decide the determination of an appropriate objective function and its encoding. In our case, each particle-pattern has an own fitness value and compares other particle-pattern's fitness value in order to share information of global solution. The fitness function follows a hierarchical approach in which the main objective is to maximize the SGC in WMN. Thus, the fitness function of this scenario is defined as

$$\text{Fitness} = 0.7 \times \text{SGC}(\mathbf{x}_{ij}, \mathbf{y}_{ij}) + 0.3 \times \text{NCMC}(\mathbf{x}_{ij}, \mathbf{y}_{ij}).$$

Routers replacement method: A mesh router has x , y positions and velocity. Mesh routers are moved based on velocities. There are many moving methods in PSO field, such as:

Table I
COMMON SIMULATION PARAMETERS.

Parameters	Values
Clients distribution	Normal distribution
Number of mesh routers	16
Number of mesh clients	48
Total iterations	6400
Iteration per phase	32
Area size	From 32×32 to 128×128
Radius of a mesh router	From 2.0 to 15.0

Table II
RELATIONSHIP OF AREA SIZE WITH RADIUS OF A MESH ROUTER PARAMETERS.

Area size	Radius of a mesh router
32×32	From 2.0 to 6.0
64×64	From 3.0 to 9.0
128×128	From 5.0 to 15.0

Constriction Method (CM)

CM is a method which PSO parameters are set to a week stable region ($\omega = 0.729$, $C_1 = C_2 = 1.4955$) based on analysis of PSO by M. Clerc et. al. [31], [32].

Random Inertia Weight Method (RIWM)

In RIWM, the ω parameter is changing randomly from 0.5 to 1.0. The C_1 and C_2 are kept 2.0. The ω can be estimated by the week stable region. The average of ω is 0.75 [32].

Linearly Decreasing Inertia Weight Method (LDIWM)

In LDIWM, C_1 and C_2 are set to 2.0, constantly. On the other hand, the ω parameter is changed linearly from unstable region ($\omega = 0.9$) to stable region ($\omega = 0.4$) with increasing of iterations of computations [32], [33].

Linearly Decreasing Vmax Method (LDVM)

In LDVM, PSO parameters are set to unstable region ($\omega = 0.9$, $C_1 = C_2 = 2.0$). A value of V_{max} which is maximum velocity of particles is considered. With increasing of iteration of computations, the V_{max} is kept decreasing linearly [34].

Rational Decrement of Vmax Method (RDVM)

In RDVM, PSO parameters are set to unstable region ($\omega = 0.9$, $C_1 = C_2 = 2.0$). The V_{max} is kept decreasing with the increasing of iterations as

$$V_{max}(x) = \sqrt{W^2 + H^2} \times \frac{T - x}{x}.$$

Where, W and H are the width and the height of the considered area, respectively. Also, T and x are the total number of iterations and a current number of iteration, respectively [35].

IV. SIMULATION RESULTS

In this section, we show simulation results using WMN-SA and WMN-PSO systems. In this work, we consider

Table III
WMN-SA PARAMETERS.

Parameters	Values
SA temperature	1
Replacement method	Combination

Table IV
WMN-PSO PARAMETERS.

Parameters	Values
Number of particle-patterns	32
Replacement method	LDVM

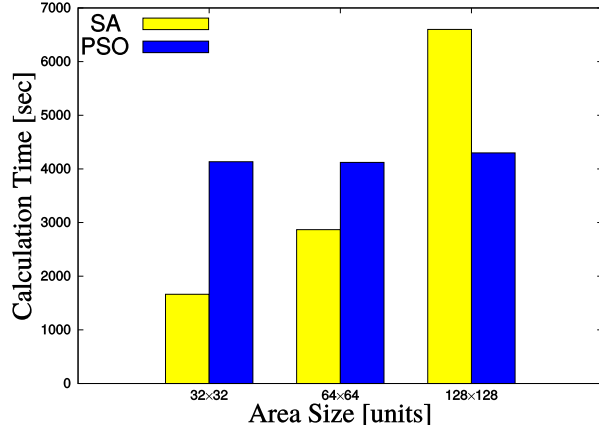


Figure 2. Comparison of WMN-SA and WMN-PSO calculation time for different area size.

the distribution of mesh clients as normal distribution. For comparison of the calculation time, we consider the area size from 32×32 to 128×128 . The number of mesh routers is considered 16 and the number of mesh clients 48. The total number of iterations is considered 6400 and the iterations per phase is considered 32. In SA, we set SA temperature 1. In PSO, we consider the number of particle-patterns 32.

The simulation parameters and their values for both WMN-SA and WMN-PSO are shown in Table I. We show the relationship of area size and radius of a mesh router in Table II. The radius of a mesh router is decided randomly. The WMN-SA parameters and WMN-PSO parameters are shown in Table III and Table IV, respectively. We conducted simulations 30 times, in order to avoid the effect randomness and create a general view of results.

We show the simulation results from Fig. 2 to Fig. 5. In Fig. 2, we show the calculation time of WMN-SA and WMN-PSO systems. We can see that the WMN-PSO needs more calculation time than WMN-SA when the area size is small. However, calculation time of WMN-SA is exponentially increased with increasing of area size. On the other hand, WMN-PSO calculation time is almost constant. In Fig. 3, Fig 4 and Fig. 5, we evaluate the simulation results by using 2 metrics (SGC and NCMC). In Fig. 3, we consider the area size 32×32 . The WMN-SA converges very fast

and its performance is very good. It should be noted that also WMN-PSO has archived maximal values of SGC and NCMC.

In Fig. 4, we consider the area size 64×64 . Comparing the performance with Fig. 3, the WMN-SA converges slower than the area size 32×32 , but still has a good performance. However, the WMN-PSO has almost the same performance as in Fig. 3.

In Fig. 5, we can see that with increasing of the area size to 128×128 , the performance of WMN-SA is decreased, but the system has still good behavior. On the other hand, the performance of WMN-PSO is decreased much more. However, the WMN-PSO calculation time is better than WMN-SA.

V. CONCLUSIONS

In this work, we implemented two simulation systems based on SA and PSO (called WMN-SA and WMN-PSO) in order to solve the mesh router placement problem in WMNs. We compared the performance of WMN-SA and WMN-PSO systems by simulations.

From the simulation results, we conclude as the following.

- When the area size is 32×32 and 64×64 , WMN-SA has better performance than WMN-PSO.
- When the area size is 128×128 , WMN-SA performs better than WMN-PSO. However, WMN-SA needs more calculation time than WMN-PSO.

In our future work, we would like to evaluate the performance of the proposed system for different parameters and patterns. Moreover, we would like to compare its performance with other algorithms.

ACKNOWLEDGEMENT

This work is supported by a Grant-in-Aid for Scientific Research from Japanese Society for the Promotion of Science (JSPS KAKENHI Grant Number 15J12086). The authors would like to thank JSPS for the financial support.

REFERENCES

- [1] M. Ikeda, "Analysis of Mobile Ad-hoc Network Routing Protocols using Shadowing Propagation Model," *International Journal of Space-Based and Situated Computing*, vol. 2, no. 3, pp. 139–148, 2012.
- [2] O. Boyinbode, H. Le, and M. Takizawa, "A Survey on Clustering Algorithms for Wireless Sensor Networks," *International Journal of Space-Based and Situated Computing*, vol. 1, no. 2, pp. 130–136, 2011.
- [3] E. Kulla, G. Mino, S. Sakamoto, M. Ikeda, S. Caballé, and L. Barolli, "FBMIS: A Fuzzy-Based Multi-interface System for Cellular and Ad Hoc Networks," *IEEE International Conference on Advanced Information Networking and Applications (AINA-2014)*, pp. 180–185, 2014.

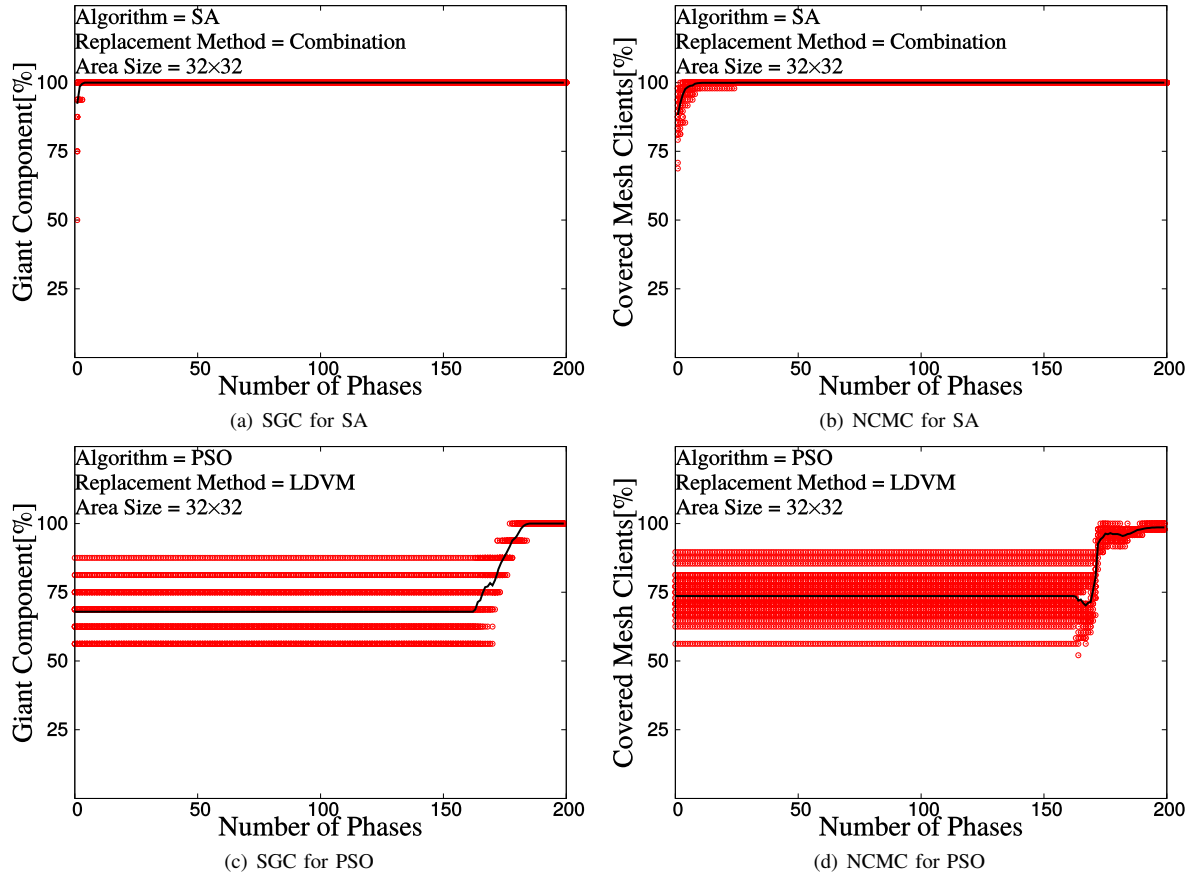


Figure 3. Simulation results for different algorithms when the area size is 32×32 .

- [4] T. Inaba, S. Sakamoto, E. Kulla, S. Caballe, M. Ikeda, and L. Barolli, "An Integrated System for Wireless Cellular and Ad-Hoc Networks Using Fuzzy Logic," *International Conference on Intelligent Networking and Collaborative Systems (INCoS-2014)*, pp. 157–162, 2014.
- [5] M. Hiyama, E. Kulla, M. Ikeda, and L. Barolli, "Evaluation of MANET Protocols for Different Indoor Environments: Results from a Real MANET Testbed," *International Journal of Space-Based and Situated Computing*, vol. 2, no. 2, pp. 71–82, 2012.
- [6] M. Hiyama, S. Sakamoto, E. Kulla, M. Ikeda, and L. Barolli, "Experimental Results of a MANET Testbed for Different Settings of HELLO Packets of OLSR Protocol," *Journal of Mobile Multimedia*, vol. 9, no. 1-2, pp. 27–38, 2013.
- [7] M. Ikeda, T. Honda, and L. Barolli, "Performance of Optimized Link State Routing Protocol for Video Streaming Application in Vehicular Ad-hoc networks Cloud Computing," *Concurrency and Computation: Practice and Experience*, vol. 27, no. 8, pp. 2054–2063, 2015.
- [8] K. Goto, Y. Sasaki, T. Hara, and S. Nishio, "Data Gathering using Mobile Agents for Reducing Traffic in Dense Mobile Wireless Sensor Networks," *Mobile Information Systems*, vol. 9, no. 4, pp. 295–314, 2013.
- [9] F. Xhafa, J. Sun, A. Barolli, A. Biberaj, and L. Barolli, "Genetic Algorithms for Satellite Scheduling Problems," *Mobile Information Systems*, vol. 8, no. 4, pp. 351–377, 2012.
- [10] A. Aikebaier, T. Enokido, and M. Takizawa, "TMPR-scheme for Reliably Broadcast Messages Among Peer Processes," *International Journal of Grid and Utility Computing*, vol. 2, no. 3, pp. 175–182, 2011.
- [11] M. Ikeda, "End-to-End Single and Multiple Flows Fairness in Mobile Ad-hoc Networks," *Journal of Mobile Multimedia*, vol. 8, no. 3, pp. 204–224, 2012.
- [12] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless Mesh Networks: A Survey," *Computer Networks*, vol. 47, no. 4, pp. 445–487, 2005.
- [13] S. N. Muthaiah and C. P. Rosenberg, "Single Gateway Placement in Wireless Mesh Networks," *Proc. of 8th International IEEE Symposium on Computer Networks*, pp. 4754–4759, 2008.
- [14] A. A. Franklin and C. S. R. Murthy, "Node Placement Algorithm for Deployment of Two-tier Wireless Mesh Networks," *Proc. of Global Telecommunications Conference*, pp. 4823–4827, 2007.

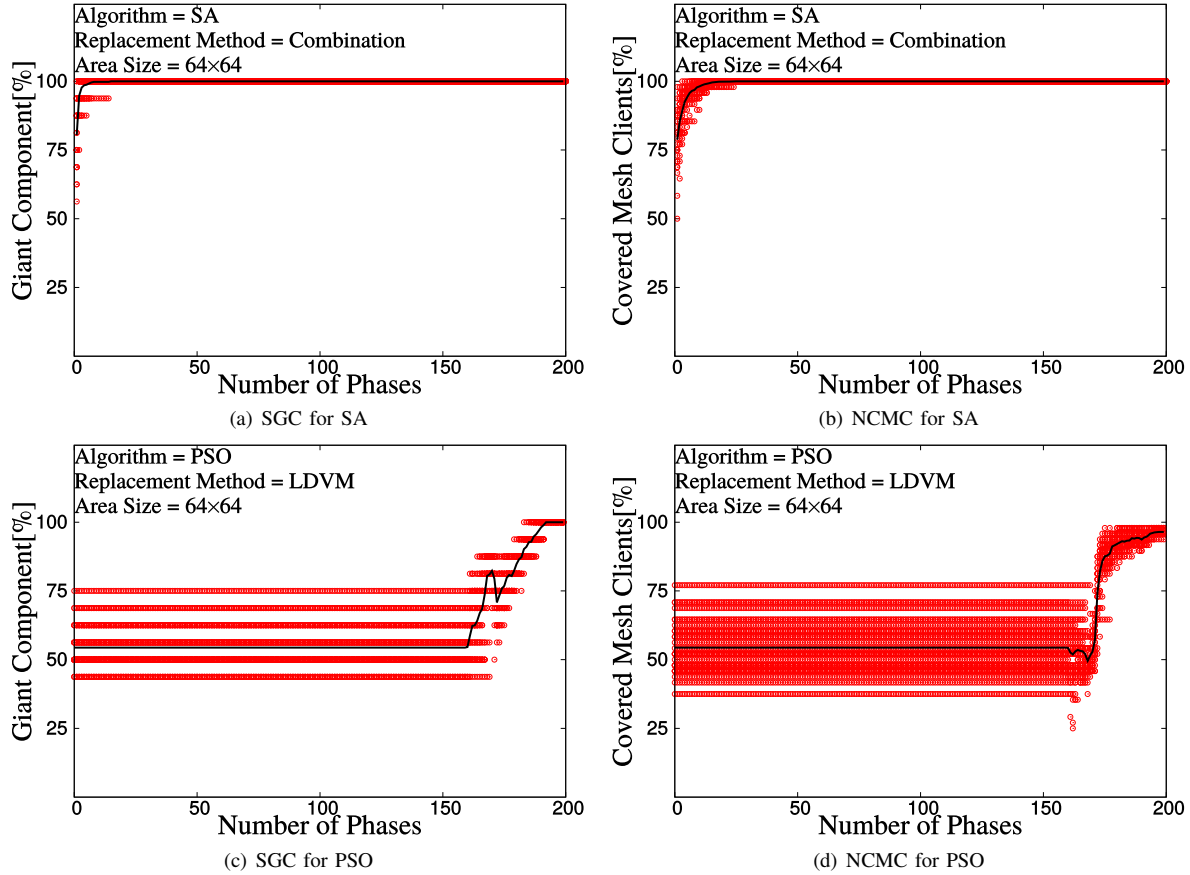


Figure 4. Simulation results for different algorithms when the area size is 64×64 .

- [15] T. Vanhatupa, M. Hannikainen, and T. Hamalainen, "Genetic Algorithm to Optimize Node Placement and Configuration for WLAN Planning," *Proc. of 4th IEEE International Symposium on Wireless Communication Systems*, pp. 612–616, October 2007.
- [16] A. Lim, B. Rodrigues, F. Wang, and Z. Xu, "k-Center Problems with Minimum Coverage," *Computing and Combinatorics*, pp. 349–359, 2004.
- [17] T. Maolin *et al.*, "Gateways Placement in Backbone Wireless Mesh Networks," *International Journal of Communications, Network and System Sciences*, vol. 2, no. 1, p. 44, 2009.
- [18] J. Wang, B. Xie, K. Cai, and D. P. Agrawal, "Efficient Mesh Router Placement in Wireless Mesh Networks," *Proc. of IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS-2007)*, pp. 1–9, 2007.
- [19] S. Sakamoto, E. Kulla, T. Oda, M. Ikeda, L. Barolli, and F. Xhafa, "A Comparison Study of Simulated Annealing and Genetic Algorithm for Node Placement Problem in Wireless Mesh Networks," *Journal of Mobile Multimedia*, vol. 9, no. 1-2, pp. 101–110, 2013.
- [20] T. Hoshi, Y. Kumata, and A. Koyama, "A Proposal and Evaluation of Access Point Allocation Algorithm for Wireless Mesh Networks," *International Conference on Network-Based Information Systems (NBIS-2013)*, pp. 389–394, 2013.
- [21] T. Oda, A. Barolli, F. Xhafa, L. Barolli, M. Ikeda, and M. Takizawa, "Performance Evaluation of WMN-GA for Different Mutation and Crossover Rates Considering Number of Covered Users Parameter," *Mobile Information Systems*, vol. 8, no. 1, pp. 1–16, 2012.
- [22] S. Sakamoto, E. Kulla, T. Oda, M. Ikeda, L. Barolli, and F. Xhafa, "Performance Evaluation Considering Iterations per Phase and SA Temperature in WMN-SA System," *Mobile Information Systems*, vol. 10, no. 3, pp. 321–330, 2014.
- [23] M. R. Girgis, T. M. Mahmoud, B. A. Abdullatif, and A. M. Rabie, "Solving the Wireless Mesh Network Design Problem using Genetic Algorithm and Simulated Annealing Optimization Methods," *International Journal of Computer Applications*, vol. 96, no. 11, pp. 1–10, 2014.
- [24] S. Sakamoto, E. Kulla, T. Oda, M. Ikeda, L. Barolli, and F. Xhafa, "A Comparison Study of Hill Climbing, Simulated Annealing and Genetic Algorithm for Node Placement Problem in WMNs," *Journal of High Speed Networks*, vol. 20, no. 1, pp. 55–66, 2014.
- [25] T. Oda, A. Barolli, E. Spaho, F. Xhafa, L. Barolli, and M. Takizawa, "Evaluation of WMN-GA for Different Mu-

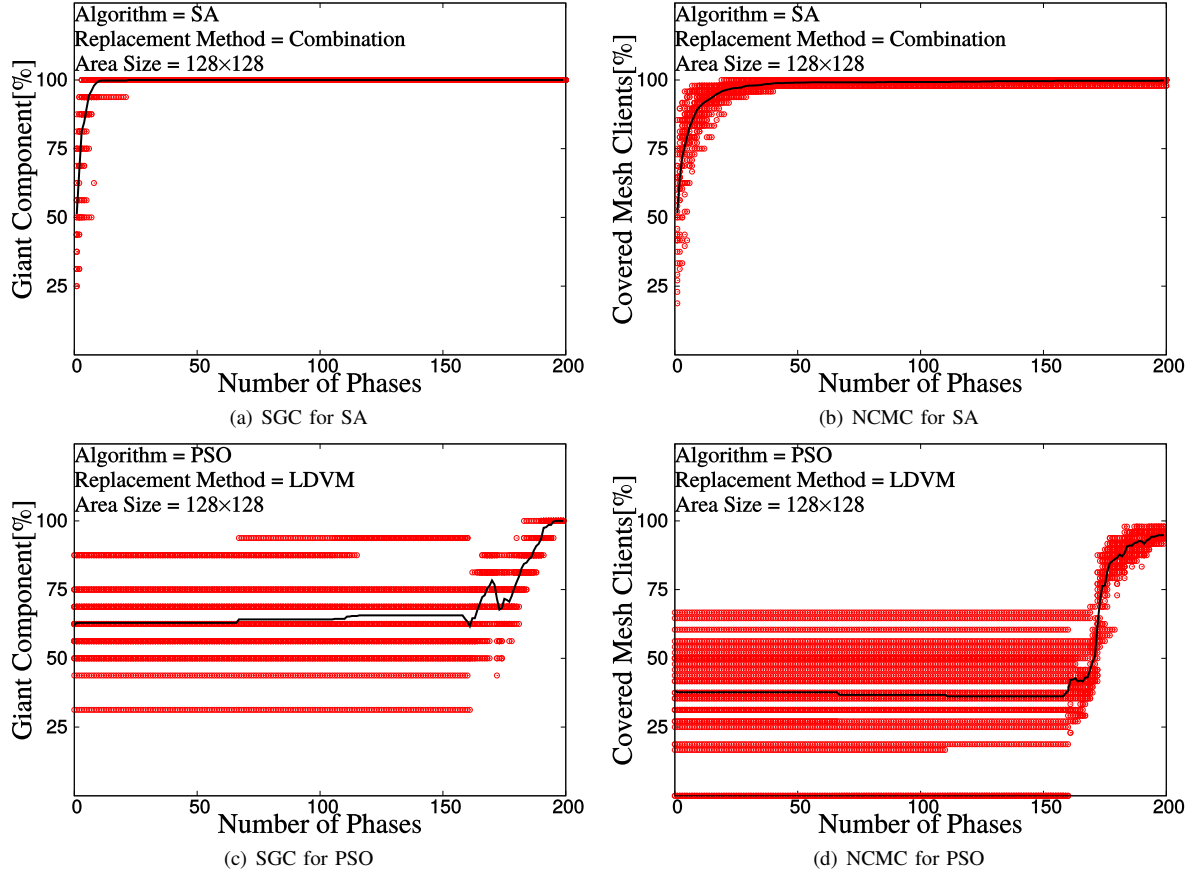


Figure 5. Simulation results for different algorithms when the area size is 128×128 .

- tation Operators," *International Journal of Space-Based and Situated Computing*, vol. 2, no. 3, pp. 149–157, 2012.
- [26] E. Amaldi, A. Capone, M. Cesana, I. Filippini, and F. Malucelli, "Optimization Models and Methods for Planning Wireless Mesh Networks," *Computer Networks*, vol. 52, no. 11, pp. 2159–2171, 2008.
- [27] S. Sakamoto, A. Lala, T. Oda, V. Kolici, L. Barolli, and F. Xhafa, "Application of WMN-SA Simulation System for Node Placement in Wireless Mesh Networks: A Case Study for a Realistic Scenario," *International Journal of Mobile Computing and Multimedia Communications (IJMCMC)*, vol. 6, no. 2, pp. 13–21, 2014.
- [28] C.-R. Hwang, "Simulated annealing: theory and applications," *Acta Applicandae Mathematicae*, vol. 12, no. 1, pp. 108–111, 1988.
- [29] R. Poli, J. Kennedy, and T. Blackwell, "Particle Swarm Optimization," *Swarm intelligence*, vol. 1, no. 1, pp. 33–57, 2007.
- [30] F. Xhafa, C. Sanchez, and L. Barolli, "Ad hoc and Neighborhood Search Methods for Placement of Mesh Routers in Wireless Mesh Networks," *Proc. of 29th IEEE International Conference on Distributed Computing Systems Workshops (ICDCS-2009)*, pp. 400–405, 2009.
- [31] M. Clerc and J. Kennedy, "The Particle Swarm-Explosion, Stability, and Convergence in a Multidimensional Complex Space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.
- [32] Y. Shi, "Particle Swarm Optimization," *IEEE Connections*, vol. 2, no. 1, pp. 8–13, 2004.
- [33] Y. Shi and R. C. Eberhart, "Parameter Selection in Particle Swarm Optimization," *Evolutionary programming VII*, pp. 591–600, 1998.
- [34] J. F. Schutte and A. A. Groenwold, "A Study of Global Optimization using Particle Swarms," *Journal of Global Optimization*, vol. 31, no. 1, pp. 93–108, 2005.
- [35] S. Sakamoto, T. Oda, M. Ikeda, L. Barolli, and F. Xhafa, "Implementation of a New Replacement Method in WMN-PSO Simulation System and Its Performance Evaluation," *The 30th IEEE International Conference on Advanced Information Networking and Applications (AINA-2016)*, pp. 206–211, 2016.