International Journal of
Intelligent Systems and Applications in Engineering
ISSN:2147-6799 www.atscience.org/IJISAE
Advanced Technology and Science
Original Research Paper

Particle Swarm Optimization Based Approach for Location Area Planning in Cellular Networks

Mays S. Algebary*[1], Issmail M. Ellabib[1], Ahmed B. Elwhishi[2]

Received 27th September 2014, Accepted 22th December 2014

Abstract: Location area planning problem plays an important role in cellular networks because of the trade-off caused by paging and registration signalling (i.e., location update). Compromising between the location update and the paging costs is essential in order to improve the performance of the network. The trade-off between these two factors can be optimized in such a way that the total cost of paging and location update can be minimized along with the link cost. Due to the complexity of this problem, meta-heuristic techniques are often used for analysing and solving practical sized instances. In this paper, we propose an approach to solve the LA planning problem based on the Particle Swarm Optimization (PSO) algorithm. The performance of the approach is investigated and evaluated with respect to the solution quality on a range of problem instances. Moreover, experimental work demonstrated the performance comparison in terms of different degree of mobility, paging load, call traffic load, and TRX load. The performance of the proposed approach outperform other existing meta-heuristic based approaches for the most problem instances.

Keywords: *Particle Swarm Optimization, Simulated Annealing Optimization, Ant Colony Optimization, Location Management in Cellular Networks, Swarm Intelligence.*

1. Introduction

The main purpose of the network planning is to manage the resources of the mobile cellular networks in order to reduce the cost of operation and meet the quality of service. [3]. There are two basic operations involved with location management; location update and paging. The goal is to partition the network into a given number of location areas such that the total cost of paging and location update is maintained at its minimum. In particular, upon the arrival of a mobile-terminated call, the system tries to find the mobile terminal by searching for it among a set of Base Transceiver Stations (BTSs) over the current region of the mobile. This search is called paging, and the set of Cells of BTSs in which a mobile is paged is called the Location Area (LA). At each LA boundary crossing, mobile terminals register (i.e., update) their new location through signalling in order to update the location management databases. Finding appropriate size or the number of LAs is essential for reducing the costs of paging and location updates signalling [3].

Finding the optimal number of location areas and the corresponding configuration of the partitioned network has motivated many researchers as it is a difficult combinatorial optimization problem which is classified as an NP- hard problem [3], [6]. The previously reported studies divided the whole problem into two sub-problems, the cell-to-switch assignment problem, and the LA planning problem [2]. Many approaches have been introduced to solve the cell-to-switch assignment problem in the literature, such as Genetic Algorithm [5], Evolutionary Algorithm [4], Ant Colony System [8], and Particle Swarm Optimization [11]. However, these approaches are based on the static location management scheme, where the service area is divided into fixed LAs sizes, and the users in a given region are assigned to the same LA regardless of their characteristics. For the economic feasibility of any communication system, the good design method should optimize the network cost, while considering some factors such as traffic, bandwidth, and capacity. The weakness of the above approaches is that the paging cost is not considered. Moreover, it is assumed that each switch manages only one LA that is equal the size of the cells belonging to that MSC, which further degrades the quality of the solution. On the other hand, most of recent personal mobile network systems, including the GSM system, are employing the zone based scheme due to high mobility and increase of subscribers [6]. In this scheme, the service area is divided into groups of cells forming LAs. The mobile terminals update their locations only when they leave their current LAs and enter new LAs. The optimization of this scheme has not been widely studied, unlike that of static based scheme. Only few studies have addressed the LA planning problem for the zone based scheme such as Simulated Annealing Algorithm [6] and Ant Colony System [1]. In this paper the Particle Swarm Optimization (PSO) algorithm is adapted to provide a solution to the LA planning problem for the zone based scheme, since most of the existing personal mobile networks use the zone based scheme in practice and include all its realistic objectives and constraints.

The rest of the paper is organized as follows: Section.2 presents the proposed Particle Swarm Optimization (PSO) approach for the LA planning problem. Section.3 provides a set of computational experiments for the analysis and performance comparison. Finally, some concluding remarks are provided in Section.4.

2. The Proposed Particle Swarm Optimization (PSO) Approach

LA planning problem is defined as follows: the service area is divided into a number of Location Areas (LAs) and each LA

[1] *Tripoli University, Computer Engineering Department, Tripoli, Libya*
[2] *Faculty of Electronics Technology, Tripoli, Libya*
* *Mays S. Algebary: Email: mais.muhsen@yahoo.com*
Note: This paper has been presented at the International Conference on Advanced Technology&Sciences (ICAT'14) held in Antalya (Turkey), August 12-15, 2014.

consists of a number of Base Transceiver Station (BTSs). The problem arises from the trade-off between the location update and the paging costs, along with the consideration of the link cost which represents the distance between BTSs, Base Station Controllers (BSCs), and Mobile Service Switching Centres (MSCs). The challenge is to assign BTSs to BSCs, BSCs to MSCs, and BTSs to LAs considering the assignment costs such that the total cost is minimized and a set of realistic constraints must be satisfied. For more details on the mathematical problem formulation, the reader may refer to [1] and [6].

2.1. PSO algorithm

Particle Swarm Optimization (PSO) is a population based search algorithm inspired by bird flocking and fish schooling originally designed and introduced by (Kennedy et al; 1997) [9]. In contrast to evolutionary computation paradigms such as Genetic Algorithm, a swarm is similar to a population, while a particle is similar to an individual. Typically, the particles fly through a multidimensional search space in which the position of each particle is adjusted according to its own experience and the experience of its neighbours. In binary (discrete) version, each particle is composed of D elements which indicate a potential solution [9]. The appropriateness of the solution is evaluated by a fitness function. Each particle is considered as a position in a D-dimensional space and each element of a particle position can take the binary value of 0 or 1 in which 1 means "included" and 0 means "not included". Each element can change from 0 to 1 and vice versa. Also, each particle has a D-dimensional velocity vector, the elements of which are in range [Vmin, Vmax]. Velocities are defined in terms of probabilities that a bit will be in one state or the other. At the beginning of the algorithm, a number of particles and their velocity vectors are generated randomly. Then, the velocity (V) and position vector (X) are updated iteratively using (Equation.1) and (Equation.2), respectively,

$$V_i^{t+1}(j) = W\, V_i^t(j) + C_1 r_1 \left(pbest_i^t(j) - X_i^t(j) \right) + C_2 r_2 \left(nbest_i^t(j) - X_i^t(j) \right) \tag{1}$$

$$X_i^{t+1}(j) = \begin{cases} 1 & if \ \frac{1}{1+exp(-V_i^{t+1}(j))} > r_{ij} \\ 0 & otherwise \end{cases} \tag{2}$$

$X_i^t(j)$ is the position of j^{th} element of i^{th} particle in t^{th} iteration. $V_i^t(j)$ is the velocity of j^{th} element of the i^{th} particle in t^{th} iteration. The *pbest* is the current best position of the particle, and *nbest* is the best position found so far. C_1 and C_2 are positive acceleration constants which control the influence of *pbest* and *nbest* on the search process. r_1 and r_2 are random values in the range [0, 1] that are sampled from a uniform distribution. W is called inertia weight, which is introduced to control the exploration and exploitation abilities of the swarm. r_{ij} of (Equation.2) is random number on the range [0, 1]. The advantage of the PSO algorithm is the combination of local search methods (through self-experience) with global search methods (through neighbouring experience). However, the set of parameters in (Equation.1) need to be tuned sensibly in order to balance exploration and exploitation for the search of high quality solution. For more details on the PSO algorithm, the reader may refer to [7] and [9].

2.2. Adaption of PSO algorithm

In order to adapt PSO algorithm for solving the LA planning problem, we represent the problem in the form of three following connections: the BTS-BSC connection, BSC-MSC connection, and BTS-LA connection such that each particle can be encoded as three $m \times n$ matrices, which called the position matrices of the particle. m represents the number of BTSs in the matrix of BTS-BSC connection and BTS-LA connection, and the number of BSCs in the matrix of BSC-MSC connection. n represents the number of BSCs in the matrix of BTS-BSC connection, the number of MSCs in the matrix of BSC-MSC connection, and the number of LAs in BTS-LA connection. Each position matrix in each particle has the following properties:

- All the elements of the position matrices have either the value of 0 or 1.
- In each row of these matrices only one element is 1 and others are 0.

In position matrix 1, each column represents a BSC and each row represents allocated BTS in a particular BSC. In position matrix 2 each column represents MSC and each row represents BSC allocated in a particular MSC. In position matrix 3 each column represents LA and each row represents BTS allocated in a particular LA. For instance, (Table.1) illustrates the position matrix 1 (BTS-BSC connection) for one particle; the position matrix 1 illustrates that BTS1 is assigned to BSC1, BTS2 is assigned to BSC2, BTS3 is assigned to BSC2, etc. The same interpretation is applied for the other matrices

Table 1. Position matrix 1 for a single particle

	BSC1	BSC2	BSC3	BSC4
BTS1	1	0	0	0
BTS2	0	1	0	0
BTS3	0	1	0	0
BTS4	0	0	1	0
BTS5	0	0	0	1

Velocity of each particle is considered as a m×n matrix whose elements are in range [V*min*,V*max*]. In other words, if V_k is the velocity matrix of k^{th} particle, then:

$$V_k^t(i,j) \in [V_{min}, V_{max}] \quad (\forall i,j), i \in \{1..m\}, j \in \{1..n\} \tag{3}$$

pbest and *nbest* are $m \times n$ position matrices and their elements are 0 or 1. $pbest_k$ represents the best position that k^{th} particle has visited since the first iteration and $nbest_k$ represents the best position that k^{th} particle and its neighbors have visited from the beginning of the algorithm. For updating *nbest* in each iteration, *pbests* are used so that if the fitness value of *pbest* is greater than *nbest*, then *nbest* is replaced with *pbest*. (Equation.4) is applied for updating the velocity matrix and (Equation.5) is applied to update the position matrix of each particle.

$$V_k^{(t+1)}(i,j) = W . V_k^t(i,j) + c_1 r_1 \left(pbest_k^t(i,j) - X_k^t(i,j) \right) + c_2 r_2 \left(nbest_k^t(i,j) - X_k^t(i,j) \right) \tag{4}$$

$$X_k^{(t+1)}(i,j) = \begin{cases} 1 & if \ \left(V_k^{(t+1)}(i,j) = \max\left\{ V_k^{(t+1)}(i,j) \right\} \right) \\ 0 & otherwise \end{cases} \tag{5}$$

Where, $V_k^t(i,j)$ denotes the element in i^{th} row and j^{th} column of the k^{th} velocity matrix in t^{th} iteration of the algorithm, and $X_k^t(i,j)$ denotes the element in i^{th} row and j^{th} column of the

k^{th} position matrix in t^{th} iteration. (Equation.5) illustrates that in each column of position matrix a value 1 is assigned to the element whose corresponding element in velocity matrix has the max value in its corresponding column. If there is more than one element in the column of velocity matrix with max value, then one of these elements is selected randomly and 1 assigned to its corresponding element in the position matrix. (Figure.1) illustrates the basic steps of the adapted structure of the PSO algorithm for solving the LA planning problem.

```
- Initialize particles with random positions.
 - Initialize each particle with random velocity.
 - Initialize pbest and nbest.
While iterations<= Max_of _iterations
For each particle
     Calculate the fitness value from the obj. function.
     If current fitness value >fitness value of the pbest
                    Then update pbest
End for
-Update nbest from pbest.
-Apply assignment criteria (determine the assignment type).
- For each particle
          - Update the particle velocity using Equation.4
          - Update the particle position using Equation.5
- End For
End while
Return the solution from the nbest.
```

Figure 1. The adapted structure of PSO algorithm.

2.3. Neighbourhood Structure (Assignment Types)

In this paper, three types of assignments are introduced for the connection of BTSs, BSCs, and MSCs. The feasible solution can be generated by any of the three types of assignment and each assignment affects the network structure in different ways.

- BTS to BSC Assignment: In this type, the BTS-BSC connection or position matrix of the particle is considered. (Equation.4) will be used to update the velocity matrix related to BTS-BSC connection, and the BTS-BSC position matrix is consequently updated by (Equation.5). Thereafter the new load on each BSC is updated and all the constraints affected by this BTS-BSC new connection are checked. If these constraints are violated then this particle is marked at the end of the iteration as invalid (infeasible solution) otherwise it is marked as valid. Finally, for each MSC starting with one LA, BTSs are assigned to an LA. If LA capacity (i.e., paging capacity of BTSs) reaches to its limit, then a new LA is created and remaining BTSs started to be assigned to that LA. Here, the aim is to create the minimum number of LAs for each MSC. The reason of marking the particles at the end of each iteration as valid or invalid is that; the infeasible solutions that is found with invalid particles which means they did violate the constraints, may eventually lead to better positions/solutions, or may not. Thus, it is might be better to use some of these invalid particle in each iteration. In order to control the number of invalid particles in each iteration so they will not be the majority and eventually affect the quality of the solution with iterations, a mechanism of accepting a predefined ratio of invalid particle to the valid once in each iteration has been used. Thus, number of invalid particles can be controlled in each iteration.
- BSC to MSC Assignment: In this assignment type, velocity and position matrices are updated on the BSC-MSC connection without affecting the BTS-BSC connection. First, the BSC-MSC velocity matrix is updated using (Equation.4) and then the BSC-MSC position matrix is updated using (Equation.5). Due to the new connections caused by (Equation.5), the feasibility of the capacity and proximity constraints of that new BSC-MSC connection must be checked. Particles that violate the constraints will be marked as invalid at the end of the iteration for the same purpose illustrated in BTS-BSC assignment type. Finally, for each MSC starting with one LA, BTSs are assigned to an LA. If LA capacity (paging capacity of BTSs) reaches to its limit, then a new LA is created and remaining BTSs started to be assigned to that LA.
- BTS to LA Assignment: In this assignment type, the particle changes the BTS-LA assignment without affecting the BTS-BSC connection i.e., the particles search for all the LAs residing within the same BSC. The assignment is done by updating the velocity matrix of BTS-LA connection using (Equatio.4), then the BTS-LA position matrix is updated using (Equation.5). Subsequently, the capacity constraint of BTS-LA connection is checked, in case this constraint is violated then this particle will be marked as invalid, the position/solution will not be accepted, and the particle will continue with its old connection/position (solution).

3. Computational Experiments and Analysis

Several experiments are conducted for parameter setting and performance evaluation of the PSO based approach. An extensive performance study is carried out to evaluate the effectiveness of the proposed approach for the LA planning problem. The proposed PSO is compared to the Simulated Annealing (SA) and the Ant Colony System (ACS) based approaches. For more details on SA and ACS based approaches for solving LA problem, the reader may refer to [6] and [1], respectively. The proposed approach and the other two meta-heuristic based approaches are implemented using visual C++ and executed under Windows8 operating system, with Intel® core ™ i5-3210M 2.50 GHz CPU and 12 GB RAM. The computation time of the approaches varies with the problem size. The data sets vary according to different patterns (high, moderated, and low) of mobility, paging rate, call traffic load, and TRX load.

3.1. Parameter Setting

The velocity update equation (Equation.4) depends on a number of parameters that need to be determined in order to provide high quality solution. In this paper, the PSO approach is applied to a typical moderately size sample network that consists of 398 BTSs, 2 BSCs, and 1 MSC for the parameter setting experiments. The preliminary experiments are performed starting with an initial setting of the parameters based on values previously reported in the literature.

The PSO tends to have more global search ability at the beginning of the run while having more local search ability near the end of the run in order to refine a candidate solution.

The inertia weight (W) has some valuable information about previously explored directions. It governs how much of the previous velocity should be retained from the previous time step [10]. The best found starting value of W is 1.2 and gradually declined towards 0 during the search.

The acceleration parameters C_1 and C_2 denote the direction of the particle towards optimal positions. They represent "cognitive" and "social" component, respectively. They affect how much the particle's local best and the global best influence its movement. From the experimental results it found that giving equal chances to exploration and exploitation didn't improve the cost as good as when giving different values for C_1 and C_2. It has found that choosing a larger cognitive parameter, C_1 than a social parameter, C_2, ($i.e.$, C_1=1.5 , C_2=0.5) gives the best result.

To ensure convergence of the heuristic, every velocity vector is bounded component-wise by minimum and maximum values [Vmin, Vmax]. These parameters are proved to be crucial, because the maximum velocity Vmax serves as a constraint to control the global exploration and exploitation ability of the particle swarm. The best result is achieved when Vmax and Vmin are equal to 50 and 0.05, respectively.

The type of assignment is chosen according to some probabilities assigned for each type of assignment. The best quality solutions were obtained, when the probability assignment of the BTS-BSC is 0.2, BSC-MSC assignment is 0.5, and BTS-LA assignment is 0.3.

The quality of solution is affected by the number of particles participating in the search process. More particles indicate a more cooperative interaction. To determine the appropriate number of particles, the results obtained for different numbers of particles, and it has found that the best number of particle is 30.

The parameters values have been tested and verified with different problem instances in terms of network cost. (Figure.2) depicts the cost of solution found at different stages of a typical run of PSO based approach for a sample network.

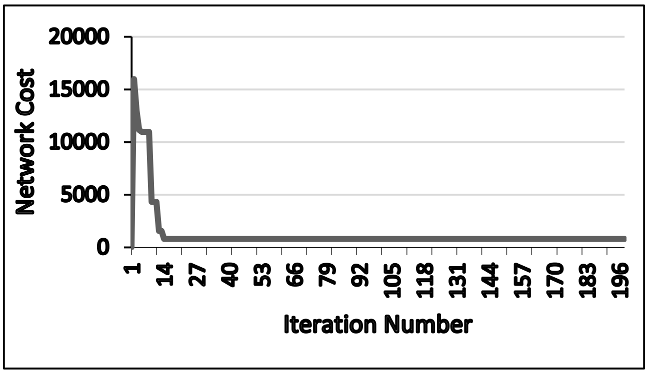

Figure 2. A typical run of the PSO based approach on a sample network.

3.2. Performance Evaluation and Comparison

To demonstrate the potential of applying the PSO algorithm for solving the LA planning, the PSO approach is compared with other existing meta-heuristic approaches. The PSO is compared with the SA and the ACS approaches on different network data sets. The network consists of 203 BTSs, 6 BSCs, and 3 MSCs. The parameters values of the SA and the ACS are set as reported in [6] and [1], respectively. To achieve the feasibility of the data generated for the network, the generation of the network loads is done according to the average values measured on the GSM network [6]. The three approaches are evaluated on the basis of the average of 30 independent runs for each instance.

Results denoted with (*) represent the results of SA and ACS as reported in [6] and [1], respectively. As shown in (Table.2), the PSO obtained the best solution for all datasets.

In what follow, the performance of the three approaches is investigated and evaluated on the four data sets with low and high values of the network factors such as mobility, paging rate, call traffic load, and TRX load. Each network factor value is scaled up (denoted by H*i*) and scaled down (denoted by L*i*) for data set *i*.

Table 2. Performance comparison between different approaches

Data Set	SA	ACS	PSO
1*	9880	9685	7687
2	7453	6866	6256
3	8951	8148	6058
4	6529	5997	5332

3.3. Effect of Mobility

To achieve a network with high mobility, the value of the handover rate of the network is scaled up by 70%. For low mobility, the same values of the handover rate is scaled down (divided by two). The other factor values remain unchanged. (Figure.3) demonstrates the performance for the three approaches (SA, ACS, and PSO) on four data sets for high mobility (H1, H2, H3, and H4) and for low mobility (L1, L2, L3, and L4).

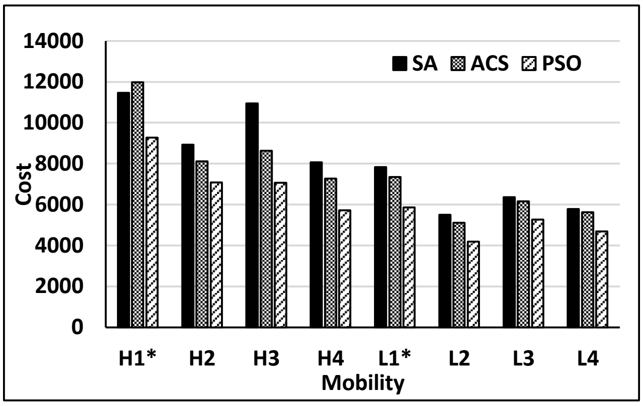

Figure 3. Performance comparison between different approaches with different mobility.

In (Figure.3), the obtained costs have increased as a result of the high crossing rate between the neighboring cells. When the handover rate increases, the position update equation (Equation.5) is affected by the handover constraints and the obtained costs are slightly higher, but still the PSO has the best solution compared to SA, and ACS. For network with a low mobility, the obtained cost is low compared to that of the high mobility case.

3.4. Effect of Paging Load

The network with high and low paging load are generated to examine the performance of the PSO algorithm. The paging load results from the number of mobile terminated calls, generated in a unit time. A high paging load means the number of mobile terminated calls is high, and a low paging load means the number of mobile terminated calls is low. A high paging load is obtained by increasing the values of the paging load for each BTS by 70%. The low paging load is obtained by scaling down the values of the paging load by 50%. The other values remain unchanged. (Figure.4) demonstrates the performance for the three approaches (SA, ACS, and PSO) on the four data sets for high paging rate (H1, H2, H3, and H4) and for low paging rate (L1, L2, L3, and L4).For the high paging load and low paging load, (Figure.4) shows the PSO presents better results compared to SA and ACS.

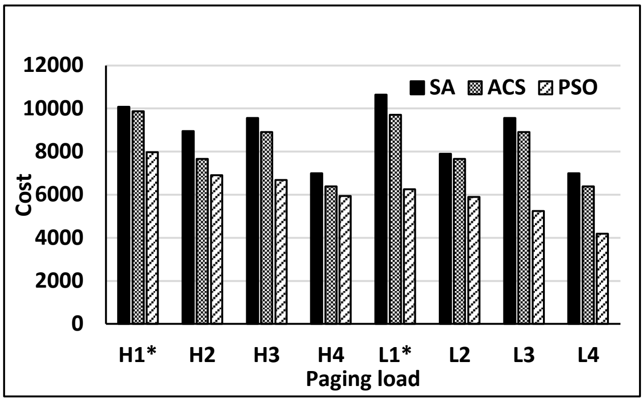

Figure 4. Performance comparison between different approaches with different paging load.

3.5. Effect of Call Traffic Load

A high traffic load is generated by scaling up the values of the call traffic load by 70%, and the low call traffic load is generated by dividing the call traffic load by two. The other factor values remain unchanged. (Figure.5) demonstrates the performance for the three approaches (SA, ACS, and PSO) on the four data sets for high call traffic load (H1, H2, H3, and H4) and for low call traffic load (L1, L2, L3, and L4).

(Figure.5) illustrates the performance of the PSO over the ACS and the SA. If we compare the results in (Figure.5) with the results in (Table.2), it noticed that these results are almost the same for high and low call traffic load for all data sets. This can be interpreted as the call traffic is a constraint which can be handled as long as the capacity of BSC and MSC is enough to handle those loads, it doesn't affect the total cost in direct way. In the meanwhile it would make the search space more tightened for the particles, which would eventually affect the quality of the solution and the constraint might be violated due to the restricted search space.

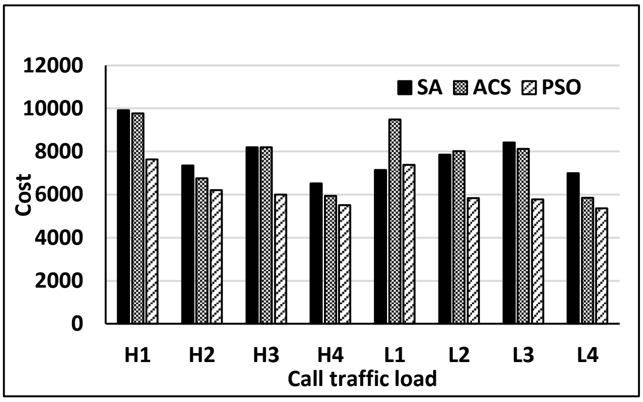

Figure 5. Performance comparison between different approaches with different call traffic load.

3.6. Effect of TRX

High and low TRX are generated by scaling up the moderate values by 70%, and down by 50%, respectively. The other factor values remain unchanged.

(Figure.6) demonstrates the performance for the three approaches (SA, ACS, and PSO) on the four data sets for high TRX (H1, H2, H3, and H4) and for low TRX (L1, L2, L3, and L4). (Figure.6) illustrates the performance of the PSO over the ACS and the SA. It noticed that solutions of high TRX is slightly better than low TRX. The reason is that the capacity constraints (in the case of the low TRX) are relaxed. As a result, the search space becomes larger and finding better solutions becomes harder.

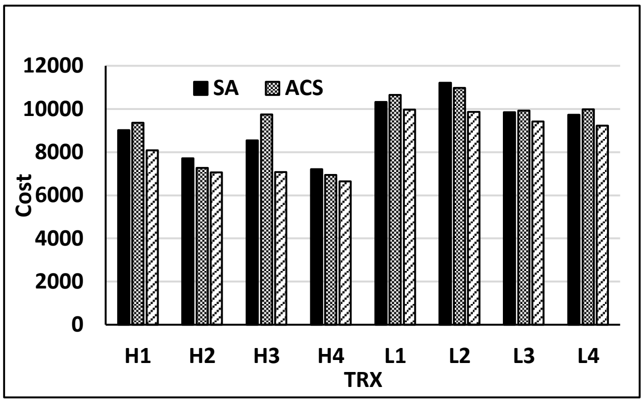

Figure 6. Performance comparison between different approaches with different TRX.

4. Conclusions

The most important benefit of optimized LA planning is preventing unnecessary radio resource usage that can instead be allocated for the communication of the subscribers. The network resources can be utilized more efficiently, and the network construction costs can be reduced by applying meta-heuristic to cellular network planning problems which leads to the proper use of limited radio resources. In this paper, the particle swarm optimization algorithm is adapted to solve the LA planning problem. The main goal is to develop an approach that can solve the LA planning problem more efficiently with particular emphasis on the tuning of the PSO parameters and the resource assignment criteria.

The potential improvement has been accomplished through the design and the analysis of PSO approach. The experimental results have shown that the PSO approach outperforms other meta-heuristic approaches on different data sets with different network loads of practical cellular network. However, further improvement to enhance the proposed approach is still needed. Of particular importance is investigation on the parameter sensitivity.

Acknowledgment

This research is partially supported by the Libyan National Agency for Scientific Research (NASR). This support is gratefully acknowledged.

References

[1] A. Elwhishi, I. Ellabib, and I. El-Feghi,"Ant Colony Optimization for Location Area Planning Problem in Cellular Networks," In the 9th International Arab Conference on Information Technology, Hammamet-Tunisia, December 16-18, 2008.

[2] A. Merchant and B. Sengupta, "Assignment of cells to switches in PCS networks," IEEE/ACM Transactions on Networking 3 (5), 521–526 October, 1995.s

[3] A. Mukherjee, S. Bandyopadhyay, and D. Saha, "Location Management and Routing in Mobile Wireless Networks, Mobile communication Series, Artech House INC., ISBN:1-58053-355-8, 2003.

[4] A.Quintero and S. Pierre, "Evolutionary Approach to optimize the assignment of Cells to Switches in Personal Communication Networks," Wireless Communications Vol.26 (9) pp. 927-938, June, 2003.

[5] C. Hedible and S. Pierre,"A Genetic Algorithm for Assigning Cells to Switches in Personal Communication Networks," IEEE Canadian Review 44, pp. 21–24, 2003.

[6] I. Demirkol, C. soy, M. Caglayun, and H. Delic, "Location

Area Planning in Cellular Networks Using Simulated Annealing," Proc. INFOCOM, pp. 566-573, March 2001.
[7] R. Eberhart and J. Kennedy, "A New Optimizer Using Particle Swarm Theory," in Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, pp. 39-43,1995.
[8] R. Fournier and S. Pierre,"Assigning Cell to Switches in Mobile Networks using an Ant Colony Optimization heuristics," Computer Communications28, pp. 65-73, 2005.
[9] J. Kennedy, R. Eberhart, "A Discrete Binary Version of the Particle Swarm Algorithm," IEEE international conference on Systems, Man, and Cybernetics, pp. 4104 – 4108, 1997.
[10] Y. Chun-man, G. Bao-long, W. Xian-xiang, "Empirical Study of the Inertia Weight Particle Swarm Optimization with Constraint Factor," International Journal of Soft Computing And Software Engineering (JSCSE) ISSN: 2251-7545 Vol.2, No2, 2012.
[11] K. Udgata1, U. Anuradha, G. Pavan Kumar," Assignment of Cells to Switches in a Cellular Mobile Environment using Swarm Intelligence," IEEE, International Conference on Information Technology (ICIT), pp. 189-194, 2008.