ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2015

Hybrid Genetic Algorithm-Simulated Annealing Approach for Reporting Cell Planning Problem of Location Management

Meghna H. Vekariya, Prof. Prashant B. Swadas

M.E. Student, Dept. of Computer Engineering, B.V.M. Engineering College, Vallabh Vidyanagar, Gujarat, India

Associate Professor, Dept. of Computer Engineering, B.V.M. Engineering College, Vallabh Vidyanagar, Gujarat, India

ABSTRACT: Location management, which is about keeping track of current location of mobile terminals with the objective of minimizing cost involved, is an important issue. In reporting cell location management approach, some cells are designated as reporting cells. In order to minimize the location management cost, the reporting cells must be determined carefully. Determining a set of reporting cells is a combinatorial optimization problem. In this paper, Genetic algorithm (GA) in combination with Simulated Annealing (SA) is used to solve the reporting cell planning problem. Effectiveness of the proposed method is demonstrated by comparing it with the results of other methods in literature for different test networks.

KEYWORDS: Location Management, Cellular Network, Reporting Cell Planning, Genetic Algorithm (GA), Simulated Annealing (SA).

I. INTRODUCTION

In wireless environment, location management, which is related to keeping track of each mobile terminal's current location, is an important issue. Location management consists of two basic operations: location update and paging. Location update is required to inform the network about new location of a terminal when a terminal moves from one location to another. Paging is required to find exact location of a terminal to provide services to that terminal, like forwarding an incoming call to a terminal. These operations use limited wireless resources like bandwidth and power. To efficiently utilize the resources most of the wireless networks use cellular architecture. In cellular architecture whole coverage area is divided into number of subareas; each subarea is called a cell.

There are two extreme approaches: always-update and never-update. In always-update approach, as a terminal moves from one cell to another, it performs update operation so update cost is very high but as at any time network has exact information of each terminal's location, it requires no paging effort, so paging cost is negligible. On the other hand, in never-update approach terminal never performs update operation. Whenever paging is to be performed, network has to page all the cells in the network. Thus, paging cost is higher while update cost is negligible. There are many other location management schemes, which can be categorized as static or dynamic. In static schemes where and when to perform update is predetermined while in dynamic schemes where and when to perform update operation are determined as per the terminals' call and mobility patterns dynamically. In another way the schemes can be categorized as global and local (per-user based). In global schemes, all terminals are required to perform update operation at same set of cells. In local schemes, each user is free to determine when and where to perform update operation based on its own call and mobility pattern. Time-based, distance-based and movement-based schemes are dynamic. Location area and reporting cells are static approaches [7][12][20]. In this work, Hybrid Genetic Algorithm-Simulated Annealing (Hybrid GA-SA) is used to solve the reporting cell planning problem.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2015

II. RELATED WORK

In [1] author uses fact that user activities are normally fixed at given time and this information is stored in mobility data table which is used for location update and paging. In [2] bloom filter based approach is used for paging which uses two vectors; bloom filter identity vector assigned to each terminal and cell vector of each cell. In [3] location area planning is done using simulated annealing. In [4] simulated annealing is used with compact index which is used to choose best configuration of location area in initializing solution as well as modifying solution. In [5] authors have used simulated annealing to solve reporting cell planning problem. In [14] three artificial life techniques, tabu search, ant colony optimization and genetic algorithm are used for reporting cell planning problem. In [15] authors have used differential evolution to solve the reporting cell planning. In [16] improved differential evolution is used where new approach to initialize population is used and local search is used to improve search performance. In [17] particle swarm optimization is applied to reporting cell planning. In [18] geometric particle swarm optimization and Hopfield neural network with ball dropping is used for reporting cell planning.

III. LOCATION MANAGEMENT COST IN REPORTING CELL APPROACH

In reporting cell approach, some cells are designated as reporting cells. A terminal needs to perform update operation when it enters in a new reporting cell. Every cell periodically broadcasts whether it is a reporting cell or not so that a terminal can determine if it has entered in a reporting cell or not and accordingly whether to perform update operation or not. When call arrives for a terminal, network will page the cells which are within the vicinity of the reporting cell last reported by that terminal. Vicinity of a reporting cell includes the non-reporting cells which are directly reachable from that reporting cell without crossing any other reporting cell. Each reporting cell is within the vicinity of itself. So Vicinity value of a reporting cell is total number of cells within its vicinity including the cell itself. Vicinity value of a non-reporting cell is, maximum of the vicinity values of reporting cells to which that non-reporting cell belongs [7][19]. A service are with four reporting cells is shown in Fig. 1.

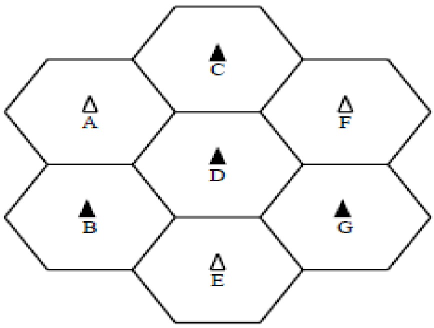

Fig. 1: A Service Area with Four Reporting Cells [7]

Here B, C, D and G are reporting cells. Suppose a terminal moves from B to A and then A to C it performs update operation. Vicinity of cell C includes cells A, F and C. Vicinity values of reporting cells B, C, D and G are 3, 3, 4 and 3 respectively. Vicinity values of all non-reporting cells A, E and F is 4 as all are within the vicinity of D which has vicinity value 4.

Here every operation, location update and paging has a cost associated with it and there is always a trade-off between the two costs. To calculate cost, two factors are considered. One is mobility weight (w_{mi}) of cell i and another is call arrival weight (w_{ci}) of cell i. Here wmi, is defined as frequency of movement into cell i and wci is defined as frequency of call arrival into cell i. Total location management cost is sum of both the costs. Update cost is related to reporting cells and mobility weight while paging cost is related to call arrival weight and vicinity. Formulas for update cost N_{LU}, paging cost N_P and total location management cost are given below.

$$N_{LU} = \Sigma\ w_{mi} \qquad \text{eq. (1)}$$

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2015

$$i \in S$$

$$N_P = \sum_{j=0}^{N-1} w_{cj} \cdot v(j) \qquad \text{eq. (2)}$$

Here N is total number of cells in given network and S is set of reporting cells. Using the equations 1 and 2 location management cost can be calculated as given in equation 3. C is the cost ratio of location update and paging. Studies show that cost of location update operation is much higher than paging cost so C = 10 is used in most of the studies. v(j) is vicinity value of cell j.

$$\text{Total cost} = C \cdot N_{LU} + N_P \qquad \text{eq. (3)}$$

In this work, cost per call arrival which is obtained by dividing total cost by total number of calls arrived is used.

$$\text{Cost per call arrival} = \text{Total cost} / \sum w_{cj} \qquad \text{eq. (4)}$$

IV. HYBRID GENETIC ALGORITHM-SIMULATED ANNEALING FOR REPORTING CELL PLANNING PROBLEM

Genetic Algorithm (GA) is a biologically inspired optimization and search technique developed by Holland. It mimics the evolution of single cell organisms described by Charles Darwin. It is mainly useful in the conditions where the solution space to be searched is huge and sequential search is computationally expensive and time consuming. GA is a part of the group of evolutionary algorithms. GA works with a population of individuals. Individual is also called chromosome which represents a solution to the given problem. It uses three main operations of natural evolution: selection, crossover and mutation on individuals of population to generate next generation [21][22][23][26].

Simulated Annealing is a probabilistic, random search method developed by S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi [9][5][6]. It is used to solve a large number of combinatorial optimization problems. The method is analogous to the way how physical annealing takes place in nature. In annealing if only downward moves are involved then the system may get stuck into local minima. Simulated Annealing resolves the difficulty by permitting upward moves with the probability specified by:

$$P=e^{-\Delta E/T} \qquad \text{eq. (5)}$$

Where ΔE is change in energy, T is temperature.

In this approach Simulated Annealing is used within the Genetic algorithm to determine a set of reporting cells. Simulated annealing probability is used in mutation operation to determine whether to accept or reject the solution and guide the search process to get optimal solution efficiently.

Pseudo code for hybrid Genetic Algorithm-Simulated Annealing for Reporting Cell Planning Problem is given below:

1. Select population size N, crossover probability Pc, mutation probability Pm, and elitism value E. Select maximum number of generations, G_{max}.
   Set counter generation number, G=0 and number of solutions in given generation, S=0.
   Initialize population randomly.
2. Calculate cost of each solution in the pool of current generation, CGP.
3. Select two solutions from the CGP using tournament selection.
4. Perform four-point crossover operation on the selected solution with probability Pc and if the solutions generated are not same as any of the solutions in the pool of next generation then add then add them to the pool of next generation, NGP. According reset S.
5. Repeat the steps 3-4 until the number of solutions, S in the pool of next generation reaches N-E.
6. Perform mutation operation on each solution in the pool of next generation with duplication check and accept or reject new solution based on the SA probability.
7. Add first E best solutions from pool of current generation into the pool of next generation.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2015

8. Replace current population by the pool of next generation, i.e. CGP=NGP, reset S=0 and set G=G+1.
9. If G> G_{max} then terminate else go to Step-3.

To apply hybrid GA-SA method to reporting cell planning problem it is required to define certain parameters carefully. Different parameters used for hybrid GA-SA approach are described below:

1. Individual or Solution representation
2. Fitness/Objective function
3. Initializing population
4. Selection mechanism
5. Crossover method
6. Mutation method with simulated annealing
7. Avoiding duplicates

1. Individual or solution representation

For the given problem each solution represents a reporting cell configuration. For an N cell network, solution length is N bit. Each bit represents a cell. Bit value '1' means the cell is reporting cell and '0' means non reporting cell. An example is shown below:

Cell Value	0	1	1	.	.	.	.	.	.	0
Cell Number	1	2	3	.	.	.	.	.	.	N

Fig. 2: Solution Representation for Genetic Algorithm

2. Fitness/Objective function

Here as a fitness function, cost function given by Equation 4 is used. As the problem is of minimization, a solution with higher cost value is less fit and vice versa.

3. Initializing population

Initial population is generated randomly and the population size i.e. the number of individuals in a population used is two times the number of cells in the given network.

4. Selection mechanism

Selection process involves choosing solutions from the current population, which will undergo crossover and mutation operations and generate off-springs. One basic method is roulette wheel selection which is based on survival of fittest. But disadvantage of the method is fittest individuals have more chance of being selected so, population will converge fast but it may lead to local optimal solution. To avoid this, tournament selection with elitism is used. It is like holding a tournament among T individuals, where T is the tournament size.

First of all t individuals are selected randomly and the winner of the tournament is the individual with highest fitness (lowest cost) of all the t competitors. Larger value of t means high selection pressure, i.e. more chance of selecting fit individuals and vice versa. So, tournament size affects the performance achieved. Normally size of 2 is used. Also here elitism is used where first E fittest solutions from current population are copied to next generation so that best solution obtained so far can be retained, where E can be selected based on population size.

5. Crossover method

Crossover combines two parent solutions, with some crossover probability, to generate new off-springs. It is analogous to the biological crossover which combines the features of parent solutions and form new solution. For optimization problem it helps in exploring the solution space in initial generations and helps in exploiting search space for latter generations. By combining better features of parents it helps generating better solution. There are different types of crossover methods.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2015

In this work four-point crossover has been used, in which four points are randomly selected and for each point a random number is generated. If the number is less than or equal to crossover probability then information between the point and next point is swapped between two parent solutions else the information are copied to off- springs as it is.

Suppose crossover probability is 0.7 and random numbers generated are 0.6, 0.8, 0.85 and 0.5. Then crossover will be performed at first and last point.

Parent 1	0	1	1	0	1	0	0	1	1	1
Parent 2	1	0	0	1	0	0	1	0	0	0

Random crossover points

After crossover

Offspring 1	0	1	0	1	1	0	0	1	0	0
Offspring 2	1	0	1	0	0	0	1	0	1	1

Fig. 3: Four-Point Crossover

6. Mutation method with Simulated Annealing

In GA mutation is used to introduce diversity. It helps algorithm escaping from local minima and avoiding premature convergence. Here a random number is selected for each bit position. Ifthe number for a bit position satisfies the mutation probability then the value at that position is modified. Normally low mutation rate is used as high rate leads to random search. Suppose mutation probability is 0.2 and random numbers generated for each bit position are such that, at position 4 and 7 values are less than 0.2. Then values at 4 and 7 will be mutated as shown below.

Parent 1	0	1	1	0	1	0	0	1	1	1
				↓			↓			
Offspring1	0	1	1	1	1	0	1	1	1	1

Fig. 4: Mutation Operation

After performing mutation, solution is accepted or rejected based on the probability used in Simulated Annealing. During initial generations probability of accepting worse solutions is more. It gradually decreases with the generations so that at later stages few worse solutions will be accepted, guiding the search process.

If the cost of the mutated solution is less than that of the original solution then mutated solution is accepted. Otherwise, a number R is randomly generated between 0 and 1. If $R < e^{-\Delta E/T}$ then accept the mutated solution else reject it, where ΔE is change in cost and T is temperature or control parameter.

7. Avoiding Duplicates

After crossover is performed, before accepting the solutions it is to be checked that the solutions are not present in the pool of next generation, if they are, they are discarded. This helps to avoid premature convergence and population being filled with same solutions. Similarly, after mutation new solution is checked to avoid duplicates.

V. Experimental Results

The proposed method is implemented using Java and run on PC with a 2GHz processor and 2GB RAM. To evaluate the performance of algorithms, data set from [14] and [18] are used, which includes Test-Network (TN) of size 4×4,

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2015

6×6 and 8×8. Test networks TN-1, TN-2 and TN-3 are from [14] and TN-4, TN-5, TN-6 and TN-7 are networks 1, 3, 5, and 6 respectively from [18]. Here cost per call arrival given in eq.(4) is used for comparison.

For Hybrid GA-SA Crossover Probability Pc = 0.7 and Mutation Probability Pm = 0.05 are used. Population size is two times number of cells in the network. Elitism rate is 10% of the population size. Initial temperature used for SA probability in mutation is 50 and temperature is reduced after every 20 generations. The algorithm is run for 300 generations. Algorithm is executed 10 times for each Test-Network.

Minimum cost values obtained by Hybrid SA-GA are compared with always-update and never-update approach as shown in Fig. 5. The result shows that Hybrid GA-SA performs better.

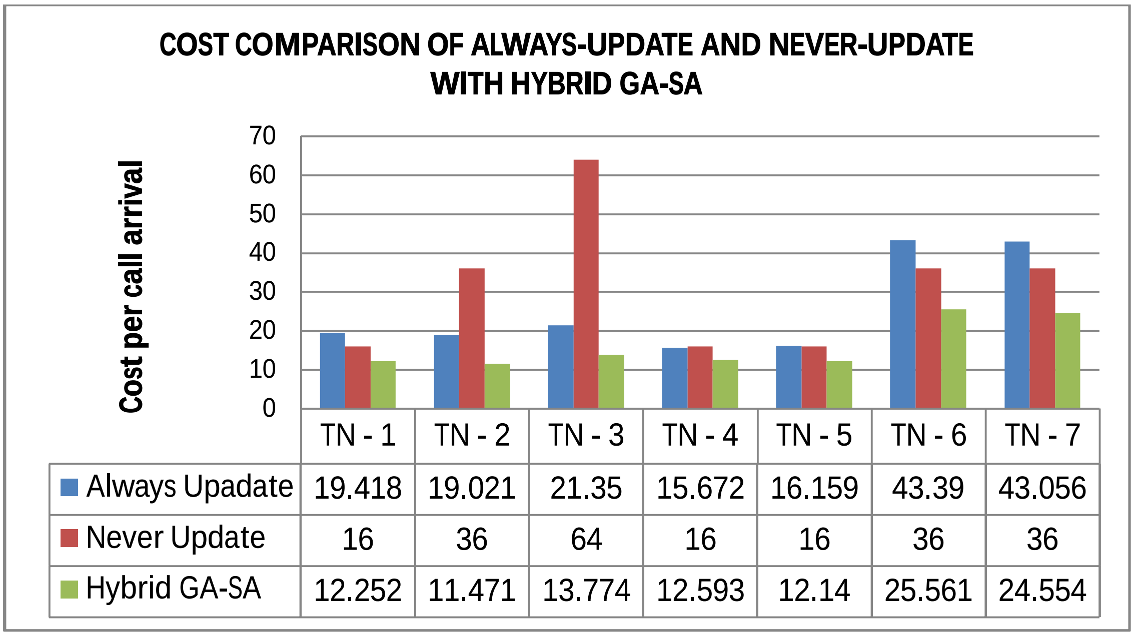

Fig. 5: Cost Comparison of Always-Update, Never-Update with Hybrid Genetic Algorithm-Simulated Annealing

Minimum cost values obtained by Hybrid GA-SA are compared with tabu search (TS), ant colony algorithm (AC) [14], geometric particle swarm optimization (GPSO) and Hopfield neural network with ball dropping (HNN-BD) [18] as shown in Table 1. The result shows that Hybrid GA-SA performs better or similar compared to others.

	HYBRID GA-SA	TS[14]	AC[14]	HNN+BD [18]	GPSO [18]
TN – 1	12.252	12.252	12.252	-	-
TN – 2	11.471	11.471	11.471	-	-
TN – 3	13.774	13.782	13.782	-	-
TN – 4	12.593	-	-	12.593	12.593
TN – 5	12.140	-	-	12.140	12.140
TN – 6	25.561	-	-	25.561	25.561
TN – 7	24.554	-	-	24.576	24.576

Table 1: Comparison of Minimum Cost Obtained by Hybrid Genetic Algorithm-Simulated Annealing with Other Algorithms

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2015

Next, average cost values on 10 runs obtained by Hybrid GA-SA are compared with other algorithms as shown in Table 2. Also, accuracy i.e. minimum found percentage (how many times minimum value is found) is compared, shown in Fig. 6. The results of Table 2 and Fig. 6 show that Hybrid GA-SA is more reliable.

	HYBRID GA-SA	TS[14]	AC[14]	HNN+BD [18]	GPSO [18]
TN – 1	12.252	12.252	12.252	-	-
TN – 2	11.471	11.471	11.471	-	-
TN – 3	13.774	13.791	13.860	-	-
TN – 4	12.593	-	-	12.605	12.593
TN – 5	12.140	-	-	12.231	12.140
TN – 6	25.561	-	-	25.575	25.561
TN – 7	24.586	-	-	24.792	24.655

Table 2: Comparison of Average Cost Obtained by Hybrid Genetic Algorithm-Simulated Annealing with Other Algorithms

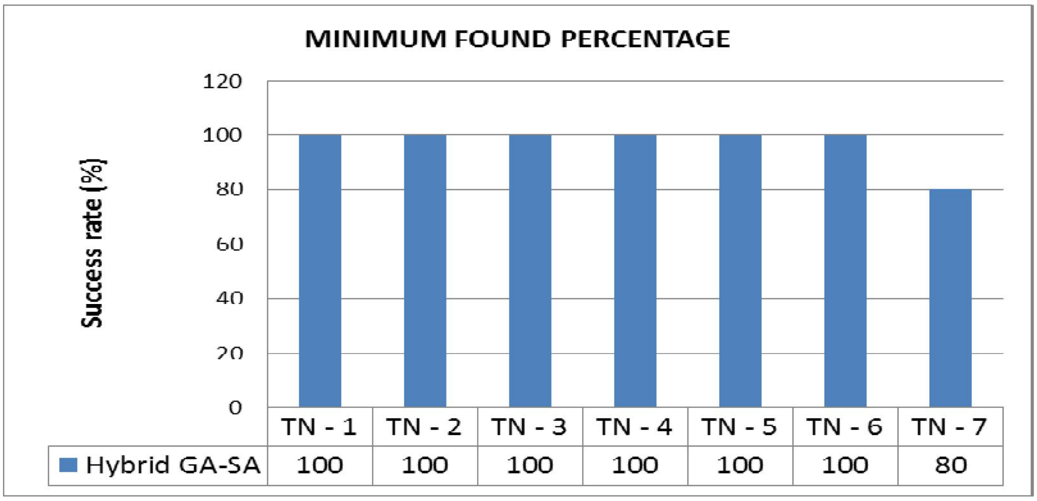

Fig. 6: Minimum Found Percentage for Hybrid Genetic Algorithm-Simulated Annealing

VI. CONCLUSION

This paper presents Hybrid Genetic Algorithm-Simulated Annealing to find optimal configuration of reporting cells such that location management cost is minimized. Simulated Annealing is used within mutation operation of Genetic Algorithm to guide the search process. Compared to other algorithms, minimum cost values obtained by the proposed approach are better or similar. Compared to always-update and never-update, given approach performs better. In terms of average cost, given approach performs similar or better than other algorithms. Also average cost and minimum found percentage shows robustness and accuracy of the proposed method.

REFERENCES

1. Alok Sahelay, Ramratan Ahirwal and Y. K. Jain, "Timer-Based Location Management to Improve Hit Ratio in Cellular Network Using Cache Memory", International Journal of Information and Education Technology, Vol. 2, No. 5, October 2012
2. Pragyan Acharya, Sudhansu Sekhar Singh, School of Electronics Engineering, KIIT University, Bhubaneswar, India, "Bloom Filter- Cost Effective Paging Mechanism", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 2 (4) , 1509-1512, 2011.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2015

3. Javid Taheri, Albert Y. Zomaya, "A Simulated Annealing Approach for Mobile Location Management", Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium IPDPS'05 IEEE, 2005
4. Ki-Dong Kim, Sung Soo Kim, Eui-Seok Byeon, Hwan Kim V. Mani, Jae-Ki Moon and Si-Hwan Jang, "Location Area Planning Using Simulated Annealing With A New Solution Representation", International Journal of Innovative Computing, Information and Control ICIC International 2012 ISSN 1349-4198 Volume 8, Number 3(A), pp. 1635-1644, March 2012.
5. Falguni Mehta, Prashant Swadas, "A Simulated Annealing Approach to Reporting Cell Planning Problem of Mobile Location Management", International Journal of Recent Trends in Engineering, Vol. 2, No. 2, November 2009.
6. C.Ashok Baburaj etc. al., "A Review On Various Network Results For Reporting Cell Planning In Genetic Algorithm Technique", International Journal of Engineering Science and Technology Vol. 2(9), pp. 4088-4094, 2010.
7. Jingyuan Zhang, "Location Management in Cellular Networks", Wiley Series on Parallel and Distributed Computing, 2002
8. C.Ashok BabuRaj, S. Thabasu Kannan, "Effective and Efficient Mining of Data in Mobile Computing", IAENG International Journal of Computer Science, 32:4, IJCS_32_4_5, 2006.
9. S. Kirkpatrick, C. D. Gelatt, Jr., M. P. Vecchi (1983), "Optimization by Simulated Annealing", Science, New Series, Vol. 220, No. 4598, pp. 671-680, 1983.
10. Rob. A. Rutenbar, "Simulated Annealing Algorithms: An Overview", IEEE Circuits and Devices Magazine, 1989.
11. P. Demestichas, N. Georgantas, E. Tzifa, V. Demesticha, M. Striki, M. Kilanioti, M. Theologou, National Technical University Of Athens, Computer Science Division, Athens, Greece, "Computationally Efficient Algorithms For Location Area Planning In Future", Cellular Systems, Elsevier, Computer Communications pp. 1263-1280, 2000.
12. Bhavneet Sidhu, And Hardeep Singh, "Location Management in Cellular Networks", International Journal of Electrical, Robotics, Electronics and Communications Engineering Vol. 1 No. 1, 2007
13. Dan wang, Congcong Xiong and Wei Huang, "Group Search Optimizer for the Mobile Location Management Problem", Hindawi Publishing Corporation, The Scientific World Journal Volume, 2014.
14. Riky Subrata and Albert Y. Zomaya, "Artificial Life Techniques for Reporting Cell Planning in Mobile Computing", Parallel and Distributed Processing Symposium, IEEE, 2002.
15. Sonia Almeida-LuzMiguel, A.Vega-Rodriguez, "Applying Differential Evolution to the Reporting Cells Problem", Proceedings of the International Multi-conference on Computer Science and Information Technology, pp. 65 – 71, IEEE Proceedings of the International Multi-Conference on Computer Science and Information Technology, pp. 65 – 71, 2008.
16. Wen-Hong Wang, Feng-Rui Wang, Quan-Ke Pan, Feng-Chao Zuo, "Improved Differential Evolution Algorithm for Location Management in Mobile Computing", Intelligent Systems and Applications, IEEE, 2009.
17. S. Kim, G. Kim, J. Byein, J. Thaeri, "Particle Swarm Optimization for Location Mobility Management", International Journal of Innovative Computing, Information and Control ICIC, Vol. 8, No. 2, December 2012.
18. Enrique Alba, José García-Nieto, Javid Taheri and Albert Zomaya, "New Research in Nature Inspired Algorithms for Mobility Management in GSM Networks", Evoworkshops 2008, LNCS 4974, pp. 1-10, Springer- 2008
19. A. Bar-Noy and I. Kessler, "Tracking Mobile Users in Wireless Communications Networks", IEEE Transactions on Information Theory, Vol. 39, Nov. 1993.
20. Amotz Bar-Noy, Ilan Kessler and Moshe Sidi "Mobile Users: To Update or Not to Update?" Wireless Networks 1, 175-185, 1995.
21. Anshul Sharma, Anuj Mehta, "Review Paper of Various Selection Methods in Genetic Algorithm", ijarccse, Volume 3, Issue 7, July 2013
22. Khalid Jebari, Mohammed Madiafi, "Selection Methods for Genetic Algorithms, IJES, 3(4), 333-344, December 2013.
23. David Beasley, David R. Bull, R. R. Martin, "An Overview of Genetic Algorithm: Part1, 2", University Computing, 1993.
24. Christian Bluma, Jakob Puchinger, Gunther R. Raidl, Andrea Roli, "Hybrid Metaheuristics in Combinatorial Optimization: A Survey", Elsevier, Applied Soft Computing, 2011.
25. L. Jourdan, M. Basseur, E.G. Talbi, "Hybridizing Exact Methods and Metaheuristics: A Taxonomy", European Journal of Operational Research, Elsevier, 2009.
26. M. Mitchell, "An Introduction to Genetic Algorithms", MIT Press, 1998.