A Combined Genetic-Neural Algorithm for Mobility Management

Javid Taheri and Albert Y. Zomaya
School of Information Technologies
The University of Sydney, Sydney, NSW 2006, Australia
{javidt, zomaya}@it.usyd.edu.au

Abstract

This work presents a new approach to solve the location management problem by using the location areas approach. A combination of a genetic algorithm and the Hopfield neural network is used to find the optimal configuration of location areas in a mobile network. Toward this end, the location areas configuration of the network is modeled so that the general condition of all the chromosomes of each population improves rapidly by the help of a Hopfield neural network. The Hopfield neural network is incorporated into the genetic algorithm optimization process, to expedite its convergence, since the generic genetic algorithm is not fast enough. Simulation results are very promising and they lead to network configurations that are unexpected but very efficient.

1. Introduction

Numerous companies and service providers are pursuing a fully integrated service solution for wireless mobile networks. Current voice, fax, and paging services will be combined with data transfer, video conferences, and other mobile multimedia services to make the next generation of wireless mobile networks, such as IMT-2000 and UMTS networks, more attractive. Basically, these networks have been designed to support a true combination of both real-time and non-real-time services and then form a global personal communication network [1, 2]. In order to support such a wide range of data transfer and user applications, *mobility management* has to be considered when designing infrastructure for wireless mobile networks.

Mobility management requests are often initiated either by a mobile terminal movement (crossing a cell boundary) or by deterioration in the quality of the signal received on a currently allocated channel. Due to the anticipated increase in the usage of the wireless services in the future, the next generation of mobile networks should be able to support a huge number of users and their bandwidth requirements.

Figure 1 shows two instances of a GSM network [1-4]. In this network, cells are grouped together into regions. Each region contains the whole allotted frequency spectrum, while each cell of the group just uses a part of the allocated frequency. The same frequency can be used in other regions by carefully considering the minimum distance between cells to avoid cross talking [1-4]. On the other hand, as the demand for wireless services increase, the size of the cells becomes smaller and the reusability of the allocated frequencies becomes more serious. As a result, network management, and consequently, location management turns into a serious problem, and, efficient techniques will be needed to ensure delivery of all incoming calls even in the tiniest of cells.

Basically, location management consists of two main parts: location update and location inquiry. In location update, every mobile terminal updates its location in the network and notifies the network its current location, while, location inquiry is performed by the network itself. In this process, the network tries to locate the user based on his last known location. Location update is usually performed when the user changes its place in the network, whereas location inquiry is usually performed when the network tries to direct an incoming call to the customer.

However, location update strategies can be categorized into two main groups: dynamic and static. In dynamic schemes, different network topologies are considered for different users [5-8]. These topologies are highly related to the movement pattern and calling behavior of each user. On the other hand, in static schemes, the network has a unique

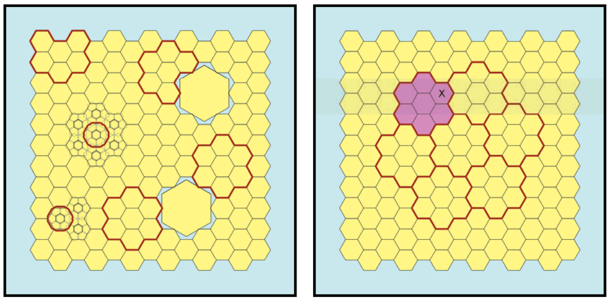


Figure 1: Two examples of a GSM network and its Location Areas configuration

1-4244-0054-6/06/$20.00 ©2006 IEEE

behavior for all users, like current GSM networks. It is obvious that dynamic schemes are much more complex than static ones and require more computation capabilities in the network. Thus, implementing static schemes is more popular [9-19]. Nevertheless, a static location update strategy must use minimum network resources to manage user tracking and should not require massive computations as well. Always update strategies, never update strategies, time based strategies, movement based strategies, distance based strategies, paging cells schemes, and location areas schemes can be mentioned as the most common techniques used for static location updating [9-19].

In this paper, a combination of a Genetic Algorithm (GA) and Hopfield Neural Network (HNN) is used with the location areas scheme to assign optimal location areas in a mobile network. In sections 2 and 3, general overview of the genetic algorithm and the Hopfield neural network is presented respectively. Section 4 provides more details on the location area scheme. In section 5, a description of calculating the location management cost is provided. Section 6 shows how the GA and HNN can be used to solve the problem under consideration while section 7 represents the main approach to solve this problem in this work. In sections 8 and 9, results of simulation are given followed by conclusion in section 10.

2. Genetic Algorithm Optimizer

A genetic algorithm is one of the most popular search algorithms, which is based on mechanisms of the natural selections. The overall structure of a generic GA is shown in Figure 2. The main aspects that should be taken into account in deploying this optimization algorithm to find the optimal solution of a sample problem are: parameterization of the problem, definition of the fitness function, selection of the best chromosomes, recombination of parent chromosomes, definition of the crossover, mutation and elitism operators, generation of the next population, and, termination of the algorithm.

In summary, this algorithm is normally initiated with a set of possible solutions of the problem, known as *Initial Population*. The initial population consists of chromosomes, and, each chromosome consists of several genes (Figure 2). During the optimization process, the chromosomes are evaluated by the genetic optimizer and best of them are selected to generate the next population. Crossover, Mutation, and Elitism are the operators used to generate next population from the initial one [17, 20-22].

3. Hopfield Neural Network Optimizer

The HNN can be used to solve a wide range of optimization problems [18][23-26]. A HNN consisting of n neurons and will have an associated Lyapanov energy (objective) function as follow:

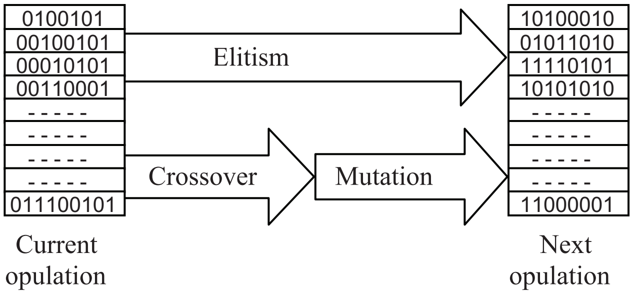


Figure 2: Overall procedure of a GA

$$E = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,j}^{obj} g_i(u_i) g_j(u_j) - \sum_{i=1}^{n} I_i^{obj} g_i(u_i)$$

The cost function can be defined as $E(x_1, x_2, ..., x_n)$ in which the x_i 's are variables that take integer values, and have to be minimized with respect to some predefined constraints. Moreover, suppose that the above constraints can be regarded as non-negative cost functions, $C_i(x_1, x_2, ..., x_n)$, where $C_k(x_1, x_2, ..., x_n)$ take the values of zero when $x_1, x_2, ..., x_n$ are at their desired values and satisfy their constraints. Now, with the help of Lagrangian coefficients and augmenting all the constraints and cost functions together, with the function F, the problem is converted to an unconstrained one, F_c. Now, the aim is to minimize this new function as:

$$F_c(x_1, x_2, ..., x_n) = F(x_1, x_2, ..., x_n) + \alpha.\sum_{k=1}^{n} C_k(x_1, x_2, ..., x_n) \quad (1)$$

By considering the cost function and the constraints as energy functions in the following forms:

$$F = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,j}^{obj} x_i x_j - \sum_{i=1}^{n} I_i^{obj} x_i \quad (2)$$

$$C_k = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,j}^{k} x_i x_j - \sum_{i=1}^{n} I_i^{k} x_i \quad (3)$$

It can be easily seen that the equivalent unconstrained optimization problem F_c, can be given as:

$$F_c(x_1,...,x_n) = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}[w_{i,j}^{obj} + \alpha.(w_{i,j}^{1} + w_{i,j}^{2} + ... + w_{i,j}^{n})] x_i x_j - \sum_{i=1}^{n}[I_i^{obj} + \alpha.(I_i^1 + I_i^2 + ... + I_i^n)] x_i \quad (4)$$

Note that equation (4) has been written in the form of an energy function. Thus, if an optimization problem can be formulated as (4) (like the location management problem), it can be solved by using this artificial neural network.

4. Location Areas Scheme

One of the most famous and effective ways to reduce the total cost of a mobile managements system is the location areas scheme that is currently used in the current GSM network. In this scheme, adjacent cells in the network are bounded to make some kind of larger cells. In this case, whenever a mobile terminal enters a new LA, it updates its location, and, as long as it travels within the same LA, it never updates its location again. On the other hand, whenever there is an incoming call, and since the network doesn't know the exact location of the user in the current LA, it pages the user in all the cells of the last updated LA. For example, assume a user enters the LA marked in gray in Figure 1 from cell marked as 'X'. If the user's motion is confined to cells of this marked LA, then, it will never update its location again. Therefore, in the case of an incoming call, the user will be paged in all cells of this LA.

5. Location Update Cost

Many algorithms have been proposed to solve the location management problem [5-19]. However, there is a need to develop a framework that can be used to compare these techniques. Toward this end, a location management cost needs to be defined to evaluate each approach. The location management cost usually consists of two main parts: updating cost and paging cost. Updating cost is the portion of the total cost due to location updates performed by mobile terminals in the network while paging cost is caused by the network during a location inquiry when the network tries to locate a user. The total cost of location management involves other parameters, like, the cost of database management to register users' locations, the cost of the wired network (backbone) that connects the base stations to each other, the cost of switching between base stations in the case of handoff and call diverting, and several other components. However, these costs are assumed to be the same for all location management strategies in general. As a result, the combination of location update and paging costs are considered to be sufficient to compare the different approaches. Therefore, the total cost of a location management scheme is given as [15-19]:

$$Cost = \beta \times N_{LU} + N_P$$

Where N_{LU} is the total number of location updates, N_P represent the total number of paging transactions, and β is a constant representing the cost ratio of a location update to a paging transaction in the network. The number of location updates is usually caused by the movement of the user in the network, while, the number of the paging transactions is highly related to the number of incoming calls. Evidence shows that the cost of each location update is much higher than the cost of a paging transaction, because of a complex procedure that has to be executed every time a location update is performed [15-19]. On the other hand, most of the calls of a mobile user are the incoming calls. Therefore, if the user moves in the network without making any call, the network will undergo a huge number of useless transactions. Thus, in most cases the cost of a location update is considered to be 10 times more than that of a paging transaction, i.e. β=*10* [15-19].

Based on the LA scheme, a location update transaction takes place when a user changes its current LA. In this case, all entrances to LAs of the network should be added to compute this cost. On the other hand, the paging cost must be considered when a user has an incoming call. Thus, all cells must be considered in calculating this cost [15-19].

6. Different Approaches to Solve the Location Management Problem

Although there are few approaches that have been proposed to solve such problem, the only two approaches that are directly related to the work in this paper are described here.

6.1. Genetic Algorithm

This approach which formerly has been done in one of our works [17] is based on using a modified GA to find the optimal location areas for a mobile network. In this approach, fixed length chromosomes with the size of the number of cells in the network are used. In this approach, each gene of each chromosome represents the LA number which the cell belongs to. For example, assume the current LA scheme of the network as Figure 3. The chromosome

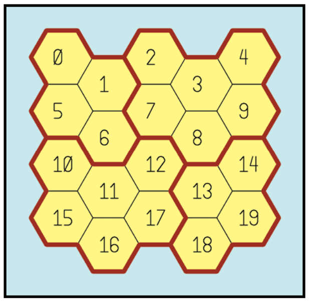


Figure 3: LA configuration

Genes of a sample LA Chromosome																				
CN	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
LA	0	0	2	2	2	0	0	2	2	2	1	1	1	3	3	1	1	1	3	3

Figure 4: Corresponding LA chromosome with its genes for Figure 3

that represents this configuration is shown in Figure 4. Note that, each gene contains a decimal number to show the LA the cell belongs to. For example the cells 2, 3, 4, 7, 8 and 9 belong to the same LA (i.e. LA=2) as shown in Figures 3 and 4.

6.2. Hopfield Neural Network

This approach is another method we formerly used to solve this problem [18]. In this work, based on equation (4) the state vector of the HNN, X (x_i s), is considered to have two different parts for location updates and call arrives as follows:

$$X = \begin{bmatrix} x_0 & x_1 & \cdots & x_{N-1} & x_N & x_{N+1} & \cdots & x_{2N-1} \end{bmatrix}^T$$

where x_0 to x_{N-1} is the location update part and x_N to x_{2N-1} is the call arriving part, and, *'N'* is the total number of cells in the network.

However, the presented HNN model is designed to take a LA configuration of a network, and then, tries to modify its LAs in order to reduce the total cost of the network management step by step.

Note that, other considerations like: generating an initial answer, defining a framework to modify the state vector and reducing the number of variations for the next move must be taken into account to get a reasonable answer while using this technique [18]

7. Combination of Genetic Algorithm and Hopfield Neural Network in solving the Mobile Location Management Problem

This method which is going to be presented in this paper is the combination of the two methods we formerly used to solve this problem. The main reason of combining these two techniques is to exploit the advantages that both techniques offer while omitting the drawbacks that emerge from using them individually. Basically, GA is a method which has been proved to find the optimal answer of an optimization problem in unlimited number of iterations. On the other hand, in our case, each cycle of running such approach takes a lot of time because of the complex calculations that have to be performed to correct chromosomes after generating each generation [17]. As a result, this method can not be run for so many iterations to ensure reaching the optimal answer.

HNN on the other hand, is a technique which doesn't take so much time to run for a cycle. However, getting the optimal answer (or even a relatively reasonable sub-optimal answer) in this technique is highly related to the initial condition the algorithm is started from. In fact, if the algorithm is started from an appropriate initial answer, reaching the optimal answer is so probable. In contrast, starting from an inappropriate condition could end up a solution far from optimality.

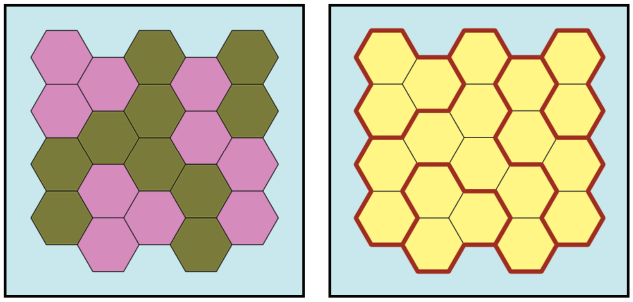
Based on the above observations [17,18], it can be easily seen that, GA is really good at finding answers close to the optimal one although catching it may need endless time for it. In contrast, HNN can be really effective if it is launched from an initial condition close to the optimal solution.

Therefore, if these two characteristics can be properly combined together, an effective algorithm can be made to find the optimal answer (inherited from GA) in relatively less number of iterations (inherited from HNN) than the original GA. The approach we proposed in this paper is a framework toward this end.

However, before describing the entire solution methodology, it is worthwhile mentioning a very important issue regarding the network generation for the stated problem. Basically, in almost all the available literature to solve this problem, the cell attributes of the network are generated randomly. In general, two independent attributes for each cell are considered, *the number of call arrivals* and *the number of location updates*, and both are randomly selected. In other words, the number of call arrivals and location updates of adjacent cells are not correlated, and furthermore, these numbers are completely independent for each cell. However, these numbers are highly correlated in a real network. Therefore, in generating these numbers for each cell in this paper, a set of sophisticated routines are used to generate the users' profiles, and consequently, the network attributes of each cell [27]. As a result, the generated network configurations better match real world traffic.

7.1. Generating the initial population

Generating the first population of a GA plays an important role in finding the solution. This population must be rich enough to generate all possible solutions in the following generations although going for highly populated generations may effectively sluggish the convergence speed of the algorithm. Therefore, it is usually generated in



(a): Two assigned LAs (b): Splitting of the LAs

Figure 5: LA configuration used to generate the initial population

a way that represents the answer space homogenously.

To generate each chromosome in this work, a simple technique is used to make GSM like LAs [17]. It is simply assumed that there are only two LAs in the network. Thus, each cell of the network is considered to be in one of these two LAs with the probability of 0.5. It is obvious that these two LAs are highly scattered. Then, these scattered LAs are split into proper ones (regarding the fact that all LA must be solid) to build the initial population. Figure 5a shows an instance of the LA scheme when only two LAs are considered in the network, while Figure 5b shows the result after splitting the scattered LAs [17].

7.2. Defining the crossover operator

The crossover operation considered here is similar to the ones used in a generic GA. The only difference is in the number of breakpoints used for the crossover operation. In this approach, two chromosomes are broken into a random number that falls between 1 and the *maximum number of crossover points* which is predefined before launching the algorithm. Then, the genes are swapped between these breakpoints with the crossover probability [17].

7.3. Defining the mutation operator

The mutation operator used here is different from the one used in a generic GA. In this work, three types of mutations are defined and one of them is selected randomly for the mutation operation on the generated chromosomes by the crossover operation. These mutation operations are designed to perform gene mutation and LA mutation. In the *Gene Mutation* a new LA number is assigned to each gene (cell number) of a chromosome with the probability of mutation. In *Split LA Mutation*, a LA in a chromosome is split into random smaller LAs with the probability of *Split LA*. And finally in the *Merge Las Mutation*, two random LAs of a chromosome are merged (form a unique LA) with the probability of *Merge LA*. All these parameters are predefined before launching the algorithm.

7.4. Defining the elitism operator

The Elitism operation is quite similar to the generic GA. In this case, a predefined number of the best chromosomes are directly copied into the next generation [17].

7.5. Choosing the selection method

For this approach, a roulette wheel is used to select chromosomes. In this case, the slices (portions) of the wheel surface are assigned related to chromosomes fitness values [17].

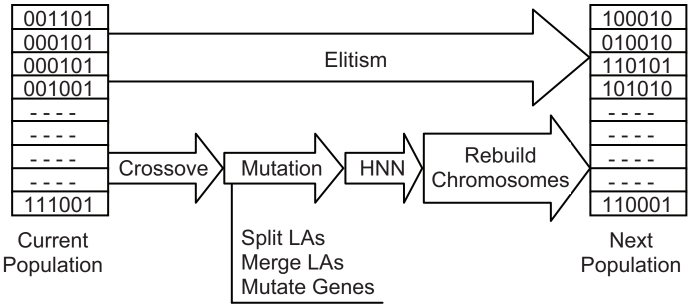


Figure 6: Overall loop of the modified GA-HNN

7.6. Defining the fitness Function

In this case, the fitness value of each chromosome is defined as follows:

$$Fitness\,Value = \frac{10000}{Energy\ Value + 1}$$

while the energy value is the total cost of the network with the LA configuration corresponding to the current chromosome.

7.7. Choosing the termination condition

Due to the difficulty of the problem, the maximum number of iterations is chosen for this approach.

7.8. Hopfield Neural Network in chromosome modification

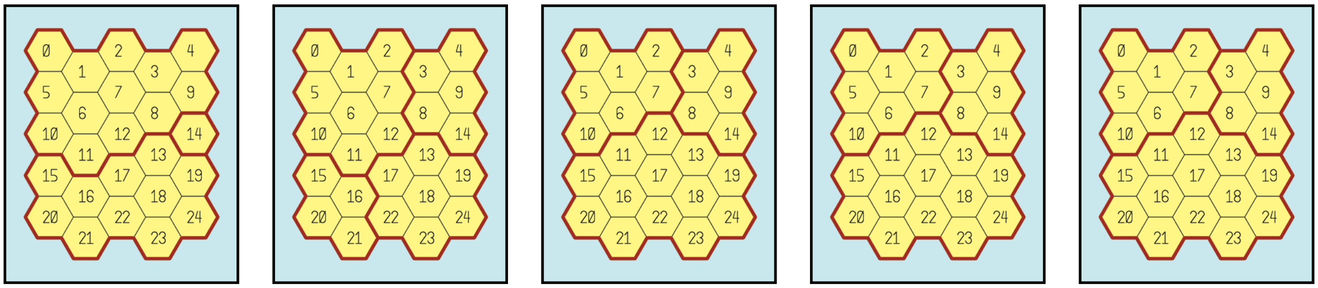
The HNN obtains the chromosomes from the Mutation operator and tries to improve them by a couple of iterations.

7.9. Defining the overall procedure

The overall procedure of this technique is given in Figure 6. As it can be seen, the mutation operator is split into three parts and another process is added before copying each chromosome into the next generation. This process, namely *Rebuild Chromosome*, gets a chromosome and checks its genes in case of scattered LAs. If there are scattered LAs in the generated chromosome, this routine splits them into feasible LAs and makes them practical.

8. Simulation Results

In this section, several test cases are presented to gauge the efficiency of the proposed approach. Three different networks are borrowed from our previous approaches for a better examination and comparison with this new approach with varying number of cells. Thus, to show the performance of the proposed algorithm, the results of



(a): GA (b): HNN (c): GA-HNN1 (d): GA-HNN2 (e): GA-HNN3

Figure 7: Different solutions for different approaches for 5x5 Network

solving these test networks by our former approaches, GA [17] and HNN [18], are presented here as well.

8.1. 5×5 Network

This network is shown in Figure 7 with its solutions found by GA, HNN, GA-HNN1, GA-HNN2 and GA-HNN3 respectively. In general GA-HNNx represents the approach of using the HNN to modify the chromosomes for x iterations. Figure 8 in this case represents the total cost of the network management for these approaches.

8.2. 5×7 Network

The solutions for this network are shown in Figure 9 while Figure 10 represents their quality.

8.3. 7×7 Network

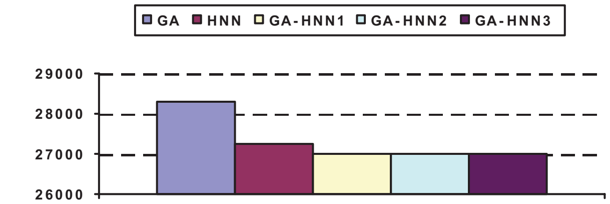
The solutions for this network are shown in Figure 11 while Figure 12 represents their quality.

9. Discussion and Analysis

A number of observations were made and will be explained below. Some of these observations have been seen in our previous approaches too [17,18].

9.1. Shape of the location areas

Despite the fact that the current GSM networks deploys circular LA, the current work showed that more efficient LAs could have a variety of configurations, e.g. rectangular or triangular as seen in Figures 7, 9, and 11.



Method	GA	HNN	GA-HNN1	GA-HNN2	GA-HNN3
Cost	28299	27249	26990	26990	26990

Figure 8: Network cost for solutions in Figure 7

9.2. Number of cells in a location area

Another interesting results obtained is the number of cells in each location area. In the GSM networks the cells are grouped into 7-by-7 to make a location area. This work shows that different LAs should have different number of cells. The number of cells in a LA is highly correlated to the number of users passing through the cells and the number of calls they receive at each cell.

9.3. Number of neighbors for each location area

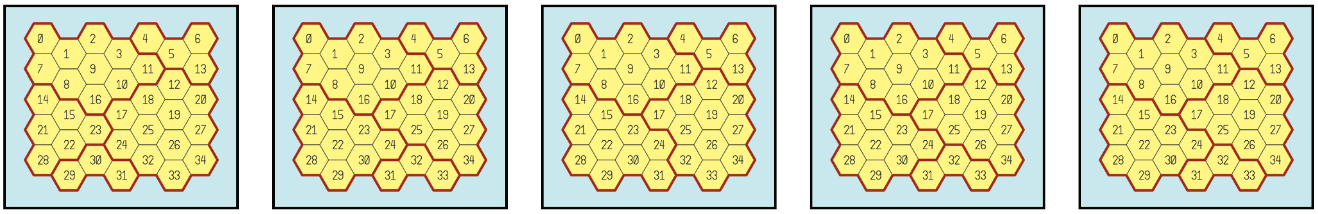
In almost all the cases, each location area is adjacent to two to four other location areas although this number is six for the currently implemented GSM networks (Figure 1). The main reason for this phenomenon is the reduction in the number of location updates and consequently the reduction in the cost of network management. The smaller the number of neighbors of a location area, the less updates are made by mobile users.

9.4. Boundary cells

Boundary cells of a LA tend to be less busy and have lower traffic than other cells in the same LA. For example, cells 19, 20, 26, 27, and 34 in Figure 9 are busy cells in comparison to cells 12, 18, 17, 25, 32, and 33 that have the less location updates and call arrivals.

9.5. Compare GA to other approaches

Based on the total cost of the network management in these test networks (Figures 7, 9 and 11), one can easily see that, although the GA is supposed to find the optimal answer, it wasn't able to do that always. The main reason of such inefficiency is in the number of iterations need to be performed by a GA, infinite iterations in theory. Since the GA approach hasn't got the chance to be run for infinite cycles and the algorithm is terminated after a predefined number of iterations, its solution is usually sub-optimal.



(a): GA (b): HNN (c): GA-HNN1 (d): GA-HNN2 (e): GA-HNN3

Figure 9: Different solutions for different approaches for 5x7 Network

9.6. Compare HNN to other approaches

Using this technique to solve these test networks, usually leads us to a sub-optimal answer which is better than that of the GA. However, because of the high speed of convergence in this technique, its final solutions are not usually optimal as well.

9.7. Compare GA-HNNx to other approaches

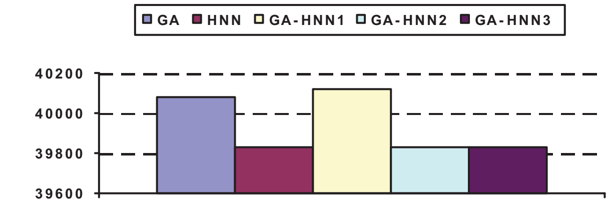
Based on the algorithms performances, this technique ends up better solutions always. Although GA-HNN3 managed to find the best answer in all tested networks, other approaches with less number of HNN iterations (GA-HNN1 and GA-HNN2) were effective too.

9.8. General overview of all answers

Although the final cost of the network management in all the mentioned approaches are really close to each other (less than 2 percent) their final answers are quite dissimilar, Figures 7, 9 and 11.

10. Conclusion

In this paper, a new approach based on the combination of the Genetic Algorithm and a Hopfield Neural Network was presented to find the optimal location areas configuration of a mobile network. In this new approach, the total cost of mobility management is related to the fitness value of the genetic optimizer. Implanting a HNN in the GA optimization process makes the performance of such algorithm quite different. The results show that the current GSM networks are far from optimality. The main findings of this work that could influence the design of the future GSM networks are relevant to the shape of location areas, the number of cells in each location area, the number of neighbors for each location area, and the location areas boundary cell properties.



Method	GA	HNN	GA-HNN1	GA-HNN2	GA-HNN3
Cost	40085	39832	40117	39832	39832

Figure 10: Network cost for solutions in Figure 9

References

[1] Y.-B. Lin, I. Chlamatac, "Wireless and Mobile Network Architecture", John Wiley and Sons, Inc. 2001.

[2] D.P. Agrawal, Q-A.Zeng, "Introduction to Wireless and Mobile Systems", Thomson Brooks/Cole Inc., 2003.

[3] Spirito, M.A., Mattioli, A.G., "On the hyperbolic positioning of GSM mobile stations ", Signals, Systems, and Electronics, 1998. ISSSE 98. 1998 URSI International Symposium on , 29 Sep-2 Oct 1998, Page(s): 173 -177.

[4] Wylie-Green, M.P., Wang, P., "GSM mobile positioning simulator", Emerging Technologies Symposium: Broadband, Wireless Internet Access, 2000 IEEE , 2000 , Page(s): 5 pp

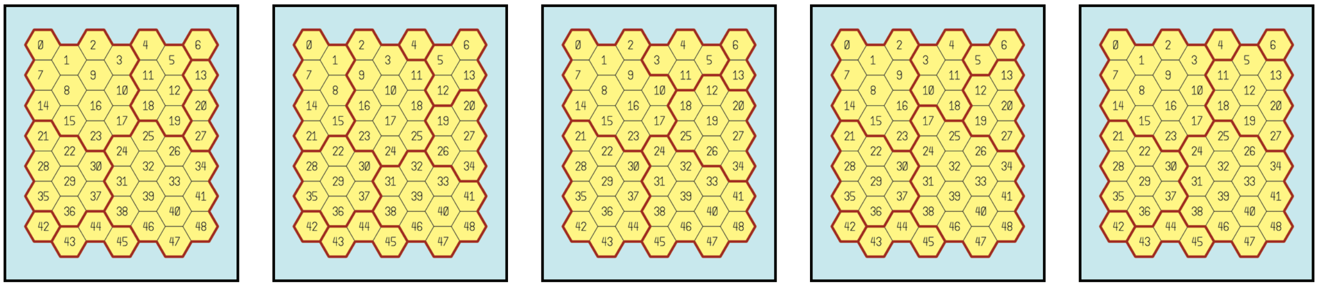
[5] Liang, B., Haas, Z.J., "Predictive distance-based mobility management for PCS networks", INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, Volume: 3 , 21-25 Mar 1999, Page(s): 1377 -1384 vol.3.

[6] Hsiao-Kuang Wu, Ming-Hui Jin, Jorng-Tzong Horng, Chen-Yi Ke, "Personal paging area design based on mobile's moving behaviors", INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, Volume: 1 , 2001, Page(s): 21 -30 vol.1.

[7] Yi-Bing Lin, Pei-Chun Lee, Chlamtac, I., "Dynamic periodic location area update in mobile networks", Vehicular Technology, IEEE Transactions on , Volume: 51 Issue: 6 , Nov 2002, Page(s): 1494 -1501.

[8] Akyildiz, I.F., Wang, W., "A dynamic location management scheme for next-generation multitier PCS systems", Wireless Communications, IEEE Transactions on, Volume: 1 Issue: 1 , Jan 2002, Page(s): 178 -189.

[9] Wong, V.W.-S., Leung, V.C.M., "An adaptive distance-based location update algorithm for PCS networks", Communications, 2001. ICC 2001. IEEE International Conference on , Volume: 7 , 2001, Page(s): 2001 -2005 vol.7.



(a): GA (b): HNN (c): GA-HNN1 (d): GA-HNN2 (e): GA-HNN3

Figure 11: Different solutions for different approaches for 7x7 Network

[10] Kuo-Hsing Chiang, Shenoy, N., Asenstorfer, J., “Intelligent handover and location updating control for a third generation mobile network”, Global Telecommunications Conference, 1998. GLOBECOM 98. The Bridge to Global Integration. IEEE , Volume: 4 , 1998, Page(s): 1963 -1969 vol.4.

[11] Kim, T.K., Leung, C., “Generalized paging schemes for cellular communication systems”, Communications, Computers and Signal Processing, 1999 IEEE Pacific Rim Conference on , 1999, Page(s): 217 -220.

[12] Subrata, R., Zomaya, A.Y., “Location management in mobile computing”, Computer Systems and Applications, ACS/IEEE International Conference on. 2001, Page(s): 287 -289.

[13] Demirkol, I., Ersoy, C., Caglayan, M.U., Delic, H., “Location area planning in cellular networks using simulated annealing”, INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, Volume: 1 , 2001, Page(s): 13 -20 vol.1.

[14] Catovic, A., Tekinay, S., “Geolocation updating schemes for location aware services in wireless networks”, Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. IEEE, Volume: 1, 2001, Page(s): 378 -382 vol.1.

[15] Ricky Subrata, Albert Y. Zomaya, “A Comparison of Three Artificial Life Technique for Reporting Cell Planning in Mobile Computing”, IEEE Transactions on Parallel and Distributed Systems, Vol. 14, No.2, February 2003, Page(s): 142 -153.

[16] Ricky Subrata, Albert Y. Zomaya,Evolving “Cellular Automata for Location Management in Mobile Computing Networks”, IEEE Transactions on Parallel and Distributed Systems, Vol. 14, No.1, January 2003, Page(s): 13 -26.

[17] Javid Taheri, Albert Y. Zomaya. "A Genetic Algorithm for Finding Optimal Location Area Configurations for Mobility Management," lcn, pp. 568-577, The IEEE Conference on Local Computer Networks 30th Anniversary (LCN'05) l, 2005.

[18] Javid Taheri, Albert Y. Zomaya. "The Use of a Hopfield Neural Network in Solving the Mobility Management Problem," *icps*, pp. 141-150, 2004 IEEE/ACS International Conference on Pervasive Services (ICPS'04), 2004.

[19] Javid Taheri, Albert Y. Zomaya. "A Simulated Annealing Approach for Mobile Location Management," ipdps, p. 194, 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) - Workshop 6, 2005.

[20] Melanie Mitchell, “An Introduction to Genetic Algorithms (Complex Adaptive Systems)”, MIT Press; Reprint edition, February 6, 1998.

[21] David E. Goldberg, “Genetic Algorithms in Search, Optimization, and Machine Learning”, Addison-Wesley Pub Co; January 1, 1989.

[22] Wolfgang Banzhaf, Peter Nordin, Robert E. Keller, Frank D. Francone, “Genetic Programming : An Introduction on the Automatic Evolution of Computer Programs and Its Applications”, Morgan Kaufmann; 1st edition December 1997.

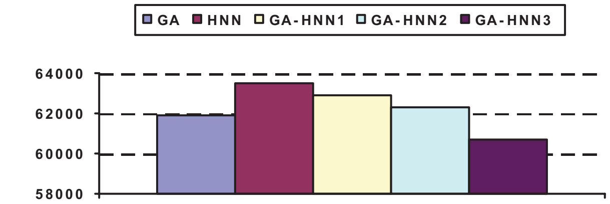
[23] J. Hopfield, D.Tank, “Neural Computation of Decosion in Optimization Problem”, Biological Cybernetics, Vol 52, 1985.

[24] S.Y.Kung, “Digital Neural Networks”, Prentice Hall, N.J., 1993.

[25] P. Arabie, L.J. Hubert, G.De Soete, “Clustring and Classification”, World Scientific Publishing Co. Pte Ltd, 1996.

[26] Kebi B. Irani, Suk I. Yoo, “A Methodology for Solving Problems: Problem Modeling and Heuristic Generation”, IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 10, No. 5, September 1988.

[27] Taheri, J. and Zomaya, A.Y., 2005, “Realistic Simulations for Studying Mobility Management Problems”, International Journal of Wireless and Mobile Computing Vol. 1, No. 8.



Method	GA	HNN	GA-HNN1	GA-HNN2	GA-HNN3
Cost	61938	63516	62916	62253	60696

Figure 12: Network cost for solutions in Figure 11