Solving a Realistic Location Area Problem Using SUMATRA Networks with the Scatter Search Algorithm

Sónia M. Almeida-Luz
Dept. of Informatics Engineering
Sch. of Tech. and Mgmt., Polytechnic Institute of Leiria
2400 Leiria, Portugal
sluz@estg.ipleiria.pt

Miguel A. Vega-Rodríguez, Juan A. Gómez-Pulido, Juan M. Sánchez-Pérez
Dept. Technologies of Comp. and Communications
Escuela Politécnica, University of Extremadura
s/n. 10071 Cáceres. Spain
{mavega, jangomez, sanperez}@unex.es

Abstract— **This paper presents a new approach based on the Scatter Search (SS) algorithm applied to the Location Management problem using the Location Area (LA) scheme. The LA scheme is used to achieve the best configuration of the network partitioning, into groups of cells (location areas), that minimizes the costs involved. In this work we execute five distinct experiments with the aim of setting the best values for the Scatter Search parameters, using test networks generated with realistic data [1]. We also want to compare the results obtained by this new approach with those achieved through classical strategies, other algorithms from our previous work and also by other authors. The simulation results show that this SS based approach is very encouraging.**

Keywords-location management; location area problem; mobile networks; scatter search; evolutionary algorithms; optimization

I. INTRODUCTION

The development of network infrastructures has growing in the last decade, principally those directed for personal communication networks (PCN) [2], because they must support the increase of user services and the communications that enable all the users to make or receive calls at any time of the day and for, or from, any location. In order to these networks support the mobility of users and be able to find them, also when they change their location, it is necessary to consider mobility management and more precisely location management (LM) when the network infrastructures are defined.

The location management is partitioned in two main operations: location update that corresponds to the notification of current location, performed by mobile terminals when they change their location in the mobile network, and location inquiry that represents the operation of determining the location of the mobile user terminal, performed by the network when it tries to direct an incoming call to the user.

There exist several strategies of location management that are mainly divided into static and dynamic schemes [3]. Static schemes are more common in the actual mobile networks. Furthermore, as static techniques, the most common ones are always-update, never-update, and location area (this one presented in section 2) schemes [3], among others.

Always-update and never-update are the two simple location management strategies. In the always-update strategy, each mobile terminal performs a location update every time it enters on a new cell, but no search operation would be required for incoming calls, because it is considered that all cells have different location areas. For the never-update strategy no location update is performed but, when there is an incoming call, a search operation is executed with the objective of finding the corresponding user; because all cells are considered as belonging to the same location area.

In this paper the Scatter Search (SS) algorithm is applied to solve the location management problem using the Location Area (LA) scheme. Section II presents an overview of the Location Area problem and the involved costs. In section III, the SS algorithm is explained. Section IV provides the implementation details of the SS based approach. In section V are exposed and analyzed the experimental results of the five experiments and after that, the results are compared with other strategies and algorithms. Finally, section VI includes conclusions and future work.

II. LOCATION AREA PROBLEM

In the Location Area (LA) scheme, the network is partitioned into groups of cells and each group corresponds to a LA (also designated as region). Because of that it is possible to say that the LA problem can be defined as: the problem of finding an optimal configuration of location areas with the objective of minimizing the location management cost. The location management cost normally is divided in two main parts: location update cost and location paging cost [4, 5].

The location update (LU) cost corresponds to the cost involved with the location updates performed by mobile terminals in the network, when they change their location to another location area.

The location paging (P) cost is caused by the network when it tries to locate a user's mobile terminal, during the location inquiry, and normally the number of paging

978-0-7695-3872-3/09 $26.00 © 2009 IEEE
DOI 10.1109/ISDA.2009.51

transactions is directly related to the number of incoming calls (for more details see [6, 7]).

The location management cost involves other parameters and components, but those are considered to be equal for all strategies [5]. Therefore, these other parameters do not influence the comparison of different strategies, and we will not consider them for the total cost. In conclusion, the combination of location update cost and location paging cost is sufficient to compare different strategy results.

The formula to calculate the total cost of location management [5, 8] is:

$$\text{Cost} = \beta \text{ x } N_{LU} + N_P. \tag{1}$$

The total cost of location updates is given by N_{LU}, the total cost of paging transactions is given by N_P, and finally β is a ratio constant used in a location update relatively to a paging transaction in the network. The cost of each location update is considered to be much higher than the cost of each paging transaction, due to the complex process that must be executed for each location update performed. Because of this, the cost of a location update is normally considered to be 10 times greater than the cost of paging, that is, β=10 [4].

III. Scatter Search Algorithm

Scatter search (SS) is an evolutionary algorithm introduced by Glover in 1977 [9]. SS is characterized by five main components [10, 11]: *Diversification Generation method*, *Improvement method*, *Reference Set Update method*, *Subset Generation method* and *Solution Combination method*. The pseudo-code of the SS algorithm is presented in Fig. 1 (see [10, 11] for more details).

```
SS Algorithm
Start with Population P=Ø. Use Diversification Generation method
to create a solution x and improve it with the Improvement method.
If x ∉ P add x to P.
Repeat this step until get PSize different solutions.
Use the Reference Set Update method to create RefSet={x^1,…,x^b}
with b/2 best solutions and b/2 most diverse solutions, of P.
Evaluate the RefSet solutions and order the solutions, using their
fitness function.
Make NewSolution=TRUE.
while (NewSolution) do
   Make NewSolution=FALSE
   Use the Subset Generation method and create all different subsets
   while(Exist subsets not examined) do
       Select a subset and label it as examined
       Apply the Solution Combination method to the solutions
             of the subset
     Considering x as the improved solution:
     if (f(x) < f(x^b)) and (x ∉ RefSet) then
             Set x^b = x and order solutions of RefSet
             Make NewSolution = TRUE
     end if
   end while
end while
```

Figure 1. Pseudo-code for Scatter Search algorithm.

IV. Implementation Details

Before the exposition of our experiments and results we start detailing the source, definition and preparation of the test network and then we explain the major decisions about the total cost calculus. Subsequently we expose the most significant decisions and adjustments of our algorithm implementation, relatively to the specific problem of the LAs.

A. SUMATRA: BALI-2 Network

Unlike our previous works [6, 7] and like in our previous work [12], in this paper we use realistic data. These data were obtained from SUMATRA [1]. SUMATRA traces are based on real user network behavior and are well validated against real world data.

The SUMATRA traces are compound by four distinct traces, each one representing a different situation in a mobile network. From these four traces, we use the BALI-2, because it includes the 24 hours call and movement trace for the San Francisco Bay Area cellular network [1]. This test network is compound by 90 cells and 66,550 mobile users.

B. Fitness Function using a Two-Step Paging

Like in our previous work [12] where we apply the Differential Evolution algorithm, in this new approach we will apply one process of paging in two steps. This process was initially proposed by Subrata and Zomaya in [8] and applied to dynamic LAs, with the objective of maintain a minimum time delay to locate each user and in order to preserve the required level of quality of service (QoS).

In our problem, we will apply this process of two-step paging to the static strategy of LAs. In the first step it is done the paging to the last known localization of the user (it is considered that the initial localization of each user is known). If the user is not found in the first step, it is applied the second step that consists in making a network wide search. But, considering that we are applying the LA strategy, this search is just conducted over the other cells (except the one already paged in the first step) that compound the respective LA. With this process we try to obtain a compromise between the rapid location of the user and the required level of Quality of Service (QoS).

In our approach the fitness function corresponds to the calculus of the total cost of location management, which is defined according to the equation (1). This means that for each individual generated (composed of a number of LAs), we will calculate its fitness value, which corresponds to the sum of the total cost of each of those LAs, calculated based on the update cost and on the two-step paging cost.

C. Parameters Definition

Taking into account the details of SS algorithm implementation (Fig. 1), in the *diversification generation* method the initial population is created considering only two LAs, where one of them is set to each cell with a probability of 50% .We also decided to apply a local search in the boundary cells of each LA as the *improvement* method. We are using a static LA scheme and in this implementation it is allowed the use of distinct numbers of cells for each location area. We also set in the *subset generation* method the definition of subsets of size 2. Relatively to the *combination* method we developed a crossover that could be applied to a maximum of four crossover points according to a predetermined probability. In conclusion, our SS uses four core parameters: initial population size *PSize*; reference set size *RSSize*; probability of combination (crossover) *Cr*; and the number of iterations of local search *nLS*. Furthermore, the *RSSize* is divided into two parameters, the size of the quality solutions *nQrs* and the size of the diversity solutions *nDrs*.

To start the experiments we set the following values: *PSize*=100; *RSSize*=10; *nQrs*=5; *nDrs*=5; *Cr*=0.2; *nLS*=1 based on what several authors suggest [10, 11].

D. Individuals Validation

When an individual is generated we must consider that an invalid configuration network may be created. This is because with the application of the algorithm it is possible that we have scattered LAs. This means that we may have cells, which are not connected, attributed to the same LA in distinct places of the network (see Fig. 2), but in reality that is not possible and we must correct or discard the individual.

To solve this problem we created a set of methods to validate and make feasible each potential solution. The first method is to split these scattered LAs into small ones. Then, the second method is defined to merge LAs, with the purpose of not having only one cell belonging to a LA, when all their neighbor cells belong to different LAs. Finally, after this, we have a third method to renumber the LAs because during all the process some LA numbers may have been deleted.

This process must be repeated for all the individuals that are generated, to assure that the final solution will be a valid one.

V. Experiments and Analysis

In this section we pretend to expose the experiments realized with this approach and analyze the results obtained. So, in order to analyze and compare results, the values of always-update and never-update strategies were calculated for the test network, using the two-step paging process. We also want to compare our results with those accomplished by other authors and with other approach developed by us.

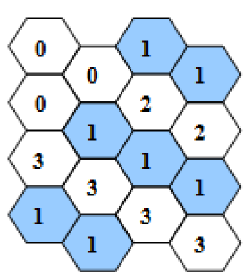

Figure 2. Scattered location area (LA 1).

With the objective of study in more detail the best configuration of SS, we have executed five distinct experiments. For each experiment, and for every combination of parameters, 30 independent runs have been performed in order to assure its statistical relevance. For each test executed we calculate the location management cost for the complete network trace, as well as for each one of the 24 hours that the trace includes.

A. Experiment 1 – Defining the PSize

The objective of the first experiment is to define the most adequate size of the population (parameter *PSize*). To accomplish that we started by testing the *PSize* with the value 10 and follow increasing it until a *PSize* of 200 individuals, passing by the values 25, 50, 75, 100, 125, 150 and 175.

Evaluating the results, that are presented in Table I, we can observe that there is not a linear evolution of results but the best fitness value is obtained with the value 100 and the average of fitness normally has a positive evolution until the value of 175. Considering this, and evaluating the partial results, corresponding to every hour, that present the same behavior of results, we decided to proceed for the second experiment with *PSize*=175.

B. Experiment 2 – Defining the RSSize

The definition of the number of individuals that will compound the *RefSet* is the objective of the second experiment. We set the *PSize* to 175, which was determined in the first experiment, maintain the other parameters with the original values and test *RSSize* with all these values: 2, 4, 6, 8, 10, 12, 14, 16, 18 and 20.

After obtaining the results using all the different values we could observe (see Table II) that along the *RSSize* increasing, the evolution of the fitness average was positive until the *RSSize*=16, where the best fitness average is obtained. Analyzing also the hourly partial results we observed that is with this size of *RefSet* that the algorithm performs better, so we decided to follow with *RSSize*=16 to the next experiment.

TABLE I. Experiment 1: Defining the *PSize*

BALI-2 - 90 Cells Network: Fitness Evaluation			
PSize	***Best***	***Average***	***St. Dev.***
10	2,770,067	2,818,716.90	27,390.90
25	2,767,978	2,803,911.27	24,357.52
50	2,770,869	2,809,024.83	27,375.63
75	2,764,829	2,801,723.63	25,186.14
100	2,759,702	2,796,282.33	20,532.83
125	2,767,072	2,799,137.70	22,368.35
150	2,766,971	2,796,110.03	20,071.66
175	2,760,254	2,795,013.63	23,582.40
200	2,762,865	2,795,753.40	23,653.35

TABLE II. EXPERIMENT 2: DEFINING THE *RSSIZE*

BALI-2 - 90 Cells Network: Fitness Evaluation			
RSSize	***Best***	***Average***	***St. Dev.***
2	2,782,198	2,830,068.43	38,891.06
4	2,764,336	2,808,567.07	21,205.40
6	2,762,892	2,795,416.67	20,759.17
8	2,758,960	2,795,839.00	22,041.31
10	2,752,114	2,787,986.60	19,602.47
12	2,773,825	2,796,021.23	12,576.84
14	2,760,288	2,792,865.00	17,799.80
16	2,756,319	2,787,297.53	14,768.22
18	2,760,702	2,791,011.27	14,930.30
20	2,753,510	2,788,964.70	17,642.09

C. Experiment 3 – Defining nQrs and nDrs

The third experiment has the objective of defining the best division of the *RefSet* between quality and diversity solutions, this means defining the best values to the *nQrs* and the *nDrs*. We started by setting *PSize*=175 and *RSSize*=16 from the earlier experiments, and maintain the initial values for other parameters, then we tested the division of the *RSSize* within all the possible combinations (considering that their sum must be equal to 16 and knowing that the *nDrs* must be equal or lower to the *nQrs*).

Once obtained all the results, presented in Table III, we observed that, it is with a lower number of diversity solutions that the best results are achieved, more precisely is the combination *nQrs*/*nDrs*=15/1 that performs better and presents the best average of fitness. The same conclusions were obtained when we observed the partial results that correspond to every hour.

D. Experiment 4 – Defining the Cr

In the fourth experiment we intent to determine the probability of applying the combination method (crossover) that permits to obtain the best results. For this experiment we initialize the value of *Cr* with a probability of 0.1, and then the algorithm was also evaluated with the values 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8 and 0.9. The rest of parameters maintain the values obtained in our previous experiments.

TABLE III. EXPERIMENT 3: DEFINING *NQRS* AND *NDRS*

BALI-2 - 90 Cells Network: Fitness Evaluation			
nQrs/nDrs	***Best***	***Average***	***St. Dev.***
8/8	2,761,964	2,790,928.67	14,869.34
9/7	2,757,650	2,789,194.10	17,533.81
10/6	2,752,113	2,788,186.63	19,526.60
11/5	2,750,811	2,791,526.33	17,955.39
12/4	2,758,285	2,786,020.23	12,583.77
13/3	2,762,273	2,789,746.50	18,781.92
14/2	2,758,913	2,789,163.97	16,631.93
15/1	2,756,878	2,785,816.50	17,232.25

TABLE IV. EXPERIMENT 4: DEFINING THE CR

BALI-2 - 90 Cells Network: Fitness Evaluation			
Cr	***Best***	***Average***	***St. Dev.***
0.01	2,763,306	2,786,296.57	15,203.90
0.03	2,756,006	2,788,594.63	16,850.68
0.05	2,757,326	2,787,588.63	16,078.42
0.07	2,762,509	2,791,238.77	18,136.82
0.09	2,758,355	2,786,450.40	15,672.59
0.1	2,758,038	2,785,114.07	16,880.10
0.2	2,753,510	2,787,702.53	17,755.62
0.3	2,757,764	2,791,493.97	21,324.76
0.4	2,759,561	2,789,114.90	17,719.35
0.5	2,757,753	2,790,212.87	17,534.44
0.6	2,755,951	2,786,621.40	17,093.97
0.7	2,758,960	2,791,630.77	18,397.30
0.8	2,762,715	2,789,253.87	19,496.06
0.9	2,760,781	2,790,843.20	14,624.44

Considering the results obtained we observed (see Table IV) that *Cr*=0.1 was the one that accomplish the best results. But, because it was the lower value we decided to test *Cr* also with the values 0.01, 0.03, 0.05, 0.07 and 0.09. Finally, as we can observe in Table IV, it is *Cr*=0.1 the most adequate decision to pass for the last experiment.

E. Experiment 5 – Defining the nLS

Finally in this last experiment we pretend to elect the most adequate number of local search iterations *nLS* in each solution improvement. So, we fixed the best values for each parameter considering the results of the previous experiments and testing the following values for *nLS*: 1, 2, 3, 4, 5, 6, 7, 8 and 9.

Analyzing the main results, presented in Table V, and also the partial results relatively for each hour, we could conclude that is until *nLS*=7 that the average of fitness presents a positive evolution.

After these five experiments we have achieved the best configuration for the SS parameters applied to the LA problem, using SUMATRA networks: *PSize*=175, *RSSize*=16, *nQrs*=15, *nDrs*=1, *Cr*=0.1, and *nLS*=7.

TABLE V. EXPERIMENT 5: DEFINING THE NLS

BALI-2 - 90 Cells Network: Fitness Evaluation			
nLS	***Best***	***Average***	***St. Dev.***
1	2,757,537	2,786,416.73	17,606.37
2	2,758,728	2,780,192.40	12,963.89
3	2,755,244	2,778,035.83	17,274.04
4	2,758,285	2,782,365.50	12,394.67
5	2,755,248	2,775,712.13	8,638.13
6	2,758,960	2,777,731.33	11,880.25
7	2,756,836	2,773,205.40	9,228.69
8	2,757,639	2,776,201.30	11,010.79
9	2,754,052	2,774,331.77	9,655.85

F. Comparison of Results

After obtaining the best configuration of SS using SUMATRA network, we decided to calculate the total cost of the network, using the classical strategies always-update and never-update, with the objective of comparing results and analyzing our approach. Observing the results, shown in Fig. 3, we could conclude that our approach always performs better than the classical ones.

If we compare the results obtained with this SS based approach with the ones that we achieved in a previous work, where we used a Differential Evolution (DE) based approach [12] (in this work we optimized the DE parameters to the LA problem), we also conclude that SS performs better than DE (see Fig. 3), because with SS our best solution has a cost of 2,756,836 and with DE the best solution had a cost of 2,799,289.

Finally, if we compare our results with those obtained by other authors, as Subrata and Zomaya [8], the conclusion is very promising because we are using static LAs and they use dynamic LAs strategies like Distance Based Location Area (DBLA). Our best fitness result is 2,756,836 cost units and their result, using DBLA, is 2,695,282 cost units [8].

G. Analysis of the Hourly Total Costs

Considering that BALI-2 [1] includes the 24 hours trace, we also calculated the partial results, of each hour, for the classical strategies always-update and never-update.

In Fig. 4 we illustrate the LM cost obtained with the classical strategies, the DE based approach [12] and the SS based approach, and we could conclude that also in the partial results the SS outperforms the others.

If we compare these LM costs for each hour with the ones obtained by Subrata and Zomaya [8], we may say that they are very similar, which gives us the notion that our approach is very competitive and viable, considering that we are using static LAs and they are using dynamic LAs.

VI. Conclusions and Future Work

In this paper we discuss the use of the scatter search (SS) algorithm to solve the location area problem. We had determined the best configuration of SS, applied to realistic networks, and, after a big number of experiments, the best value for its parameters are *PSize*=175, *RSSize*=16, *nQrs*=15, *nDrs*=1, *Cr*=0.1 and *nLS*=7.

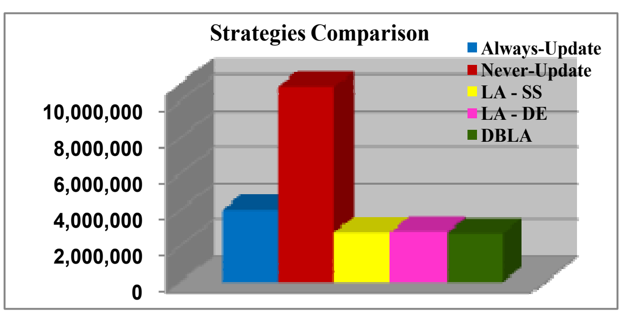

Figure 3. Comparison of strategies/algorithms results.

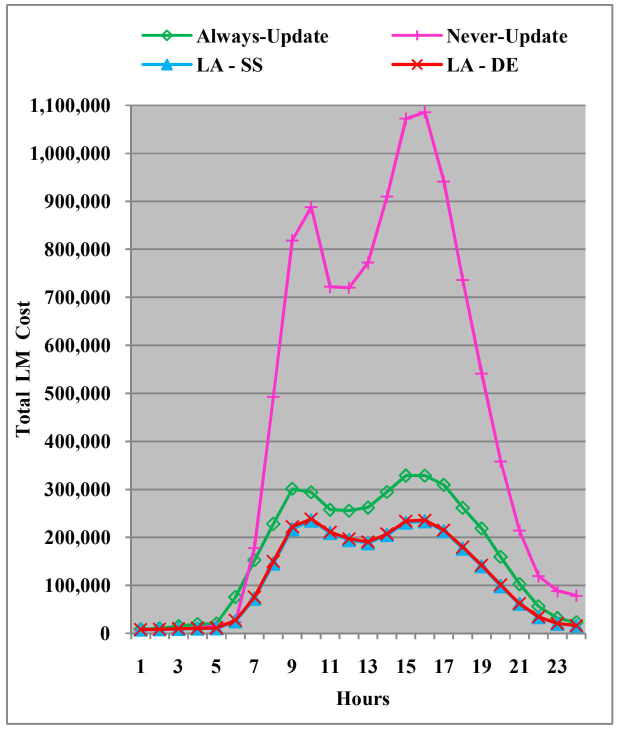

Figure 4. Comparison of LM costs for each hour.

The results obtained by the SS algorithm outperforms the results obtained with the classical strategies always-update and never-update, as well as the results obtained by us in a previous work where we applied a DE algorithm to the LAs problem.

If we compare our implementation results with the ones of other authors, which are using dynamic LAs, it is possible to conclude that they are considered very competitive because they are similar, when applied to the same test networks, and we are using static LAs.

Future work includes the plan of applying other evolutionary algorithms to the LA problem and making the comparison of their results with the ones accomplished by the SS algorithm and also with the DE algorithm. Other interesting future work is the obtaining of a bigger number of realistic networks.

Acknowledgment

This work was partially funded by the Spanish Ministry of Science and Innovation and FEDER under the contract TIN2008-06491-C04-04 (the M* project). Thanks also to the CIIC and the Polytechnic Institute of Leiria, for the economic support offered to Sónia M. Almeida-Luz to make this research.

References

[1] Stanford University Mobile Activity TRAces (SUMATRA): http://infolab.stanford.edu/sumatra/, (2009)

[2] K. Pahlavan, A.H. Levesque, Wireless Information Networks, John Wiley & Sons, Inc, 1995.

[3] V.W.S. Wong, V.C.M. Leung, “Location Management for Next-Generation Personal Communications Networks,” IEEE Network, Oct. 2000, vol. 14, no. 5, pp. 18-24.

[4] P.R.L. Gondim, “Genetic Algorithms and the Location Area Partitioning Problem in Cellular Networks,” Proc. IEEE 46th Vehicular Technology Conf. Mobile Technology for the Human Race, May 1996, vol. 3, pp. 1835-1838.

[5] J. Taheri, A. Y. Zomaya, “A Genetic Algorithm for Finding Optimal Location Area Configurations for Mobility Management,” Proc. 30th Anniversary of the IEEE Conference on Local Computer Networks (LCN), Nov. 2005, pp. 568-577.

[6] S.M. Almeida-Luz, M.A. Vega-Rodríguez, J.A. Gómez-Pulido, J.M. Sánchez-Pérez, “Defining the Best Parameters in a Differential Evolution Algorithm for Location Area Problem in Mobile Networks,” Proc. EPIA 2007 – 13th Portuguese Conference on New Trends in Artificial Intelligence; (Eds). APPIA, Dec. 2007, ISBN: 978-98-995-6180-9; pp. 219-230.

[7] S.M. Almeida-Luz, M.A. Vega-Rodríguez, J.A. Gómez-Pulido, J.M. Sánchez-Pérez, “A Differential Evolution Algorithm for Location Area Problem in Mobile Networks,” Proc. SoftCOM 2007 - 15th International Conference on Software, Telecommunications and Computer Networks, Sept. 2007, ISBN: 953-6114-95-X, pp. 1-5, doi:. 10.1109/SOFTCOM.2007.4446118.

[8] R. Subrata and A.Y. Zomaya, "Dynamic Location Area Scheme for Location Management," Telecommunication Systems, 2003, Vol. 22, No. 1-4, pp. 169-187.

[9] F. Glover, “Heuristics for integer programming using surrogate constraints,” Decision Sciences, 1977, Vol. 8, pp. 156-166, doi:10.1111/j.1540-5915.1977.tb01074.x.

[10] R. Martí, M. Laguna, and F. Glover, “Principles of Scatter Search,” European Journal of Operational Research, 2006, Vol. 169, pp. 359-372, doi:10.1016/j.ejor.2004.08.004.

[11] M. Laguna, K.P. Hossell, R. Martí, Scatter Search: Methodology and Implementation in C. Kluwer Academic Publishers, Norwell, 2002.

[12] S. Almeida-Luz, M. Vega-Rodríguez, J. Gómez-Pulido, J. Sánchez-Pérez, “Applying Differential Evolution to a Realistic Location Area Problem Using SUMATRA,” Proc. of The Second International Conference on Advanced Engineering Computing and Applications in Sciences (ADVCOMP 2008), IEEE Computer Society, Oct. 2008, pp. 170-175, doi: 10.1109/ADVCOMP.2008.19.