A Novel Genetic Algorithm with Asexual Reproduction for the Maximum Lifetime Coverage Problem in Wireless Sensor Networks

Antonina Tretyakova
Polish-Japanese Institute of Information Technology
Koszykowa 86, 02-008 Warsaw, Poland
Email: antoninat@onet.pl

Franciszek Seredynski
Cardinal Stefan Wyszynski University in Warsaw
Dep. of Mathematics and Natural Science,
Woycickiego 1/3, 01-938 Warsaw, Poland
Email: fseredynski@gmail.com

***Abstract*—In this paper, we propose a novel evolutionary algorithm called Genetic Algorithm with Asexual Reproduction (GAwAR) to solve Maximum Lifetime Coverage (MLC) Problem in Wireless Sensor Networks (WSN). We use for GAwAR a binary coding of the problem, develop asexual operator of crossover and operator of mutation in which knowledge about MLC problem is incorporated, and apply deterministic selection. We compare the proposed algorithm with a standard Genetic Algorithm with elitist strategy. We show that the proposed GAwAR significantly outperforms the standard Genetic Algorithm.**

Keywords- Wireless Sensor Network; Maximum Lifetime Coverage Problem; Genetic Algorithms.

I. INTRODUCTION

WSN is a widely developing area of technology. They are getting involved in many spheres of human vital activities, such as military, healthcare, biomedicine, environment observation, etc. According to an application area different tasks are set before WSN, among which are monitoring environment, gathering data, transmission data to a sink, etc. Due to their tiny construction WSN nodes have limitations of energy power, computing power, sensing range, transmission distance and bandwidth. There are well-known optimization problems in the literature [1] [7] [9] related to WSN, among which are the sensor deployment problem, the coverage problem, the routing problem and the MLC problem, which will be the subject of this paper. One of the most important issues related to these problems is minimizing energy consumption in order to prolong the lifetime of WSN.

All these aforementioned problems are computationally difficult problems and belong to the class of NP-hard problems. Different solutions have been proposed to solve different variants of MLC problem. Among them are approximation algorithms [18] proposed for some variants of MLC problem, linear programming [11] [15] and different heuristics [17] [19] [20] [21]. Recently, a number of approaches based on application of evolutionary algorithms have been also proposed [2] [6] [23] to deliver approximate solutions.

Our recent study [10] comparing performance of genetic algorithms (GA), memetic algorithms and a local optimization scheme have shown that general solvers based on standard evolutionary schemes do not provide satisfactory solutions. Prevailing number of currently applied GAs to solve different problems are based on sexual reproduction, where two randomly selected individuals (solutions of a problem) create two offspring, which are a subject of selection and mutation. Some recent studies [22] have shown that succesfull evolutionary search may be conducted also by using asexual reproduction. Therefore, in this paper, we develop a novel evolutionary algorithm with asexual reproduction, where basic evolutionary operators such as selection and crossover are performed on single individuals. Additionally, we incorporate a specific knowlege about the problem into operators of crossover and mutation. We compare our approach with an approach based on a standard GA with sexual reproduction and elitist strategy incorporated into GA.

The remainder of the paper is organized as follows. The next section outlines MLC problem. In Section III, we introduce some preliminary notions. Section IV introduces our proposed greedy heuristic to solve MLC problem. In Section V, we describe two genetic algorithm - based solutions. Section VI contains results of experimental study of these algorithms. The conclusion is presented in the last section.

II. MAXIMUM LIFETIME COVERAGE PROBLEM STATEMENT

Let us consider a sensor network $S = \{s_1, \ldots, s_N\}$ consisting of N sensor nodes randomly distributed over a given *target field* F, a two-dimensional rectangular area of $W \times H\, m^2$. The target field F is uniformly divided on points of interest (POIs) with a step g (see Figure 1).

Sensors are responsible for detection of an intruder (a target point) and sending an alarm message to the sink node. A sensor is defined as a point of coordinates (x_s, y_s). All sensor nodes have the same sensing range R_s, communication range R_c and battery capacity b. The coverage model of a sensor node is assumed a disk model [9].

It is assumed that each sensor can work in two modes: *active mode* and *sleeping mode*. In active mode a sensor observes an area within its sensing range and can transmit or receive a signal.

 ISBN: 978-1-61208-310-0

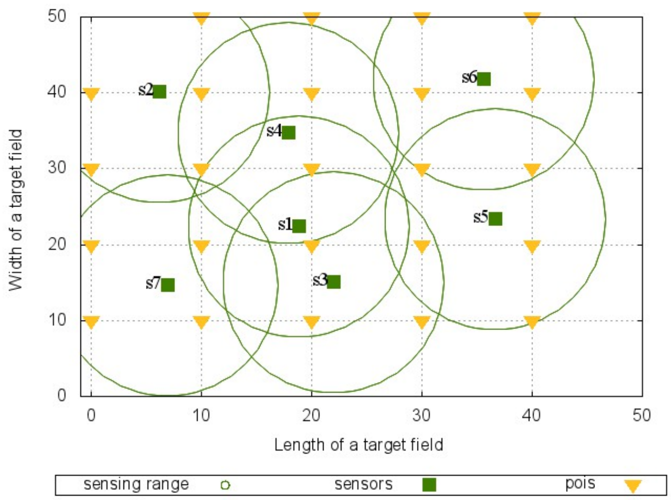


Figure 1. An example of sensor network deployed over the target field divided on POIs.

Below we give a number of definitions concerning the problem statement.

Definition1. A sensor $s(x_s, y_s)$ covers a POI *p(x, y)* denoted as POI_{obs} iff the Euclidean distance *d(s,p)* between them is less than the sensing range R_s.

Definition2. Coverage of a target field F at $i-th$ time period t_i denoted as $cov(i)$ is defined as a ratio of observed $POIs_{obs}$ by a network of active sensors S to all $POIs$, i.e.

$$cov(i) = \frac{|POIs|_{obs}}{|POIs|} \quad (1)$$

In this paper, a sensor is assumed to consume energy for monitoring area and it depends on its sensing range R_s. Considering a homogeneous sensor network, where all sensors have the same sensing range, the energy consumption per time interval is constant.

An objective in a point coverage problem is to cover a set of discrete points (targets), while an area coverage problem aims to cover the whole target field. In grid approaches, where the sensing field is uniformly divided on discrete points, called points of interest POIs [2] or targets, the area coverage problem can be considered as a point coverage problem. In this paper, we consider Maximum Lifetime Coverage Problem as a scheduling problem applied to WSN solving the point coverage problem regarding to prolongation the lifetime of WSN.

MLC has as objective to prolong lifetime of WSN by minimizing a number of redundant sensors during each time interval in order to minimize energy consumption. Lifetime of WSN is defined as a maximal number of consecutive time intervals, during which the coverage requirement is met, i.e.

$$Lifetime(q) = max\{m|(\forall i) i < m \quad cov(i) \geq q - \delta\} \quad (2)$$

Coverage requirement is given by a coverage degree k and a coverage ratio q, which means that at least q-th part with small declination δ of all targets is covered by at least k sensors. A point coverage problem with k coverage degree is denoted in literature as k-coverage problem. Further, we assume k to be equal to 1.

Maximum Lifetime Coverage Problem can be stated as follows:

- Given a set of numbers $POIs = \{1, 2, ..., P\}$, each element represents an ordinal number of a POI (a target), given a family of N subsets $S = \{S_1, S_2, ..., S_N\}$, where each element $S_i \subseteq POIs$, $i = 1, 2, ..., N$, is related to covered POIs by i-th sensor, and given an integer number b.
- Find a maximal number m of subsets $\{S'_1, S'_2, ..., S'_m\}$, where $S'_j \subseteq S$, such that the number of covered elements $|\cup_{S_i \in S'_j} S_i|$ meets the coverage ratio (Eq. 3) and each element S_i of the family S is included in b subsets $\{S'_{j_1}, S'_{j_2}, ..., S'_{j_b}\}$ (Eq. 4), i.e.

$$(\forall j)_{j=1,...,m} | \frac{|\cup_{S_i \subseteq S'_j} S_i|}{|POIs|} \geq q - \delta \quad (3)$$

$$(\forall i)(\exists j_1, ..., j_b)|S_i \in S'_{j_1}, ..., S_i \in S'_{j_m}, \quad (4)$$
$$\text{where } i = 1, ..., N \text{ and } (\forall k)_{k=1,...,b}|1 \leq j_k \leq m$$

An objective of searching a maximal number m of subsets satisfying (3) is equivalent to lifetime maximization and corresponds to scheduling activities of sensor nodes. The last equation (4) corresponds to the battery capacity restriction.

In section IV, we present a heuristic to solve the MLC problem. Also we have taken into account the following assumptions:

- i-th subset S'_i represents the network of active sensors during the i-th time interval,
- duration of all time intervals are the same,
- a number b is predefined and corresponds to the battery capacity of a sensor node, so that initial battery capacity is sufficient for every sensor's activity during the b time intervals (rounds).

III. PRELIMINARY NOTIONS

In this section, we describe a schedule solution representation and present classification of time intervals of WSN work according to coverage quality. For this purpose we introduce notions of $Redundant$, $Excellent$ and $Unsatisfactory$ $Subsequences$ (RS, ES and US). In Subsection 3.2, a volume of search space is evaluated.

A. Solution representation

A solution is encoded as an $T_{max} \times N$ table (further we call it as a *schedule* or a *schedule solution*), where T_{max} is predefined in time intervals meeting the following condition:

$$b < Lifetime(q) < T_{max} << N \times b, \quad (5)$$

where T_{max} should to be set greater than $Lifetime(q)$ (2). The upper bound $N \times b$ arises as a maximal number of different subsets, each of which consists of one element (such elements N) and, according to battery capacity requirement, can appear in the schedule b times.

In the schedule the parameter T_{max} is related to a number of columns. N is a number of sensors in the sensor network and concerning the number of rows in the schedule solution's table. A column of the schedule contains a network of active sensors during the certain time interval. Each row of the table is related to one of the sensors and represents its schedule of activity over all period of time T_{max}.

Binary coding is used, so that "1"s value corresponds to active state of a sensor, "0" corresponds to a sleeping state. Cells of the table are filled by "1"s and "0"s values in such a way that battery capacity restriction is met, i.e. each row of the table contains b ones and $T_{max} - b$ zeros. An exemplary schedule solution correspondingly to the network configuration from Figure 1 is depicted in Figure 2 (left).

A schedule solution is associated with a sequence of T_{max} numbers called a *coverage string*, i.e

$$coverage\ string = \{cov(1), ..., cov(T_{max})\}, \qquad (6)$$

where for every $i = 1, ..., T_{max}$ $cov(i)$ is counted according to (1) and $|POIs_{obs}| = |\cup_{j=1} POIs_{obs}(s_j)|$. These numbers represent a coverage of the target field during a corresponding time interval. For the individual from Figure 2 (left) correspondingly to WSN from Figure 1, the coverage string is the following {0.48, 0.36, 0.56, 0.24, 0.64, 0.0, 0.6}.

B. Solution space

Solution space contains all possible schedules and its volume depends on such parameters as a number of sensors N, battery capacity b and timeline division T_{max}. For a sensor a number of different schedules is equal to a number of different combinations of b ones and $T - b$ zeros. From combinatorics one can find that it is equal to $\frac{T_{max}!}{b!(T_{max}-b)!}$.

The volume of solution space for N sensors is presented by the following equation:

$$\frac{T_{max}!}{b!(T_{max}-b)!}^{N} \qquad (7)$$

For example, in the experiments we have WSN consisting of 100 sensors ($N = 100$), battery capacity b is equal to 15 and maximal number of intervals T_{max} is equal to 150, the search space in this case contains 148976491201904240^{100} elements.

C. Redundant, Excellent and Unsatisfactory Subsequences (RS, ES and US)

A searching process conducted by our proposed heuristics is based on the following classification of columns in a schedule. All columns of the schedule solution's table are divided on three groups called three subsequences:

1) *Redundant Subsequence (RS)*,
2) *Excellent Subsequence (ES)*,
3) *Unsatisfactory Subsequence (US)*.

Each subsequence groups time intervals such that a network of active sensors covers the target area with certain coverage ratio.

RS subsequence is introduced in order to reveal time intervals during which we potentially have redundant sensors and we wish to shift elements from *RS* into *ES*. *RS* is defined as a sequence of time intervals $\{i\}$, during which the coverage is greater than the coverage ratio q on at least δ, i.e.

$$cov(i) > q + 2\delta, \qquad (8)$$

where δ is a small value representing a predefined declination from coverage ratio q.

ES subsequence consists of time intervals $\{i\}$ in the schedule during which the coverage of the target field is within δ range from given coverage ratio q:

$$|cov(i) - \delta| \leq q + \delta \qquad (9)$$

We use *ES* as a mark of high quality of the schedule solution regarding to lifetime. In order to prolong the WSN's lifetime a number of elements in ES should be increased and their values should be less than elements included in *US*.

US subsequence is defined as time intervals $\{i\}$ in the schedule during which the coverage of the target field is less than the coverage ratio q on at least δ, i.e.

$$cov(i) < q \qquad (10)$$

Let us denote a number of elements in RS, ES and US as N_R, N_E and N_U respectevely.

IV. Genetic algorithm - based solution to MLC problem in WSN

Evolutionary approaches are based on improving the initial population of individuals (schedules) through repetitive application of selection, crossover and mutation operators. In this section we describe a solution based on Genetic algorithm (GA) approach, see Algorithm 1 in Figure 2. Individuals in a population are encoded as tables such as described in Section III.A. As fitness function Lifetime(q) (Equation 2) metric is used.

At the beginning a population of schedules is initialized in such a way that battery constraint is met, i.e. each row of the table is filled by b ones at random, the rest cells are filled by zeros. We use tournament selection scheme, where in each tornament between m randomly chosen individuals the winner goes to the next phase of reproduction. Further, selected individuals create offspring by applying the crossover operator with probability p_c. Their offspring are mutated with probability p_m and pass to the next generation. Lastly, the elitist strategy is applied, i.e. the best individual

 ISBN: 978-1-61208-310-0

Algorithm 1 GA

present an instance of WSN
present a grid target field
initialize a population P of N_p individuals (schedules)
compute fitness function
for i = 1 to G **do**
  tournament selection
  crossover
  mutation
  compute fitness function
  apply elitist strategy
end for
choose the best individual according to fitness function

Figure 2. Genetic Algorithm.

Algorithm 2 GAwAR

present WSN
present grid target field
initialize a population of N_p individuals (schedules)
compute fitness function
for i = 1 to G **do**
  for each individual **do**
    create by crossover an offspring from a single parent
    mutate offspring
    compute the fitness function of offspring
    select to a new population the better individual from the parent and its offspring
  end for
end for
The best schedule from the last generation is a solution of the algorithm

Figure 3. Genetic Algorithm with Asexual Reproduction.

from the previous population goes to the next generation without changing, replacing the worst offspring.

These steps are repeated through G generations.

1) Representation: Individuals (or chromosomes) of a population are represented by solutions and encoded as described ones in Section III.A.

2) Fitness function: Individuals are evaluated according to fitness function introduced in Section II as Lifetime(q) metric (Equation 2).

3) Genetic operators: Genetic operators work in the following way.

Firstly, in a generation a tournament selection is applied. In each tournament m competitors participate. We use an elitist selection, where the best E individuals are copied to the population without changes.

In our algorithm crossover is an analogy of simple single-point crossover operator, which proceed in two steps. First, individuals of the newly reproduced population are mated at random. Second, each pair of schedules undergoes crossing over as follows: an integer position k along the rows of the tables is selected uniforly at random between 1 and a number of rows N less one [1, N-1]. Two new individuals are created by swapping all values in the rows between $k+1$ and N.

Binary mutation is used with probability p_m.

4) Correction of individuals: Under crossover operator application the battery capacity condition in offspring schedules is kept. But mutation can change a number of activity time interval of several sensors in a chromosome. Therefore, each individual is corrected in such a way that, in case of battery is overused the genes corresponding to the last active time intervals are zeroised. Otherwise, randomly chosen zero genes in the row related to the disturber sensor change their values to value one.

V. A Novel Genetic Algorithm with Asexual Reproduction to MLC problem in WSN

In this section, we propose an Genetic Algorithm with Asexual Reproduction (GAwAR) to solve MLC problem in WSN. Firstly, for a given WSN and grid target field a population of N_p individuals are generated. Solutions are encoded such as described in Section III.A. In evolutionary strategies in each generation an individual of the population produces an offspring by using a crossover operation, further it is mutated and deterministic selection applied to chose individuals for the next generation (see, Algorithm 2 in Figure 3).

A. Representation

Individuals of a population are represented by solutions and encoded as described in Section III.A.

B. Fitness function

Individuals are evaluated according to fitness function introduced in Section II as Lifetime(q) metric (Equation 2).

C. Genetic operators

Genetic operators work in the following way. In GAwAR crossover is a modification of two operators: inver-over and position-based crossover [5] [8] adopted to solve MLC problem. The crossover is executed on each single individual from the population and consists of the following two steps:

- searching pairs of values in the same row in two consecutive US columns, such that the first value of the pair (the value from the first US column) is equal to 0, and the second value of the pair is equal to 1;
- swap these values.

In another words, in the result the first selected column contains ”1” in all cells, if at least one of the corresponding

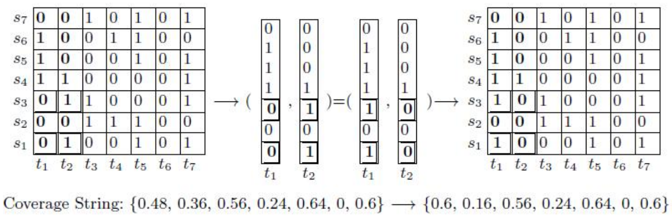


Figure 4. An example of a crossover in GAwAR.

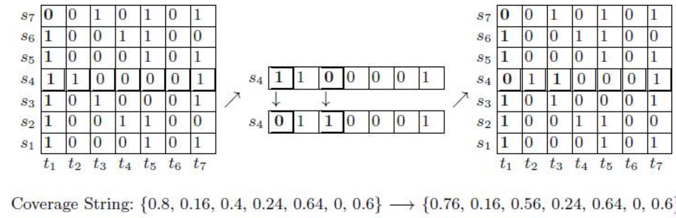


Figure 5. An example of a mutation in GAwAR.

cells has contained ”1”. The second column contains the rest of two values which was not used in the first one.

The aim of crossover is to shift several columns from US toward ES or RS.

An example of an offspring obtained by crossover is outlined in Figure 4. The schedule is constructed for a sensor network depicted in Figure 1 for seven consequtive time intervals. For coverage ratio q equal to 0.6 and coverage declination δ equal to 0.05, US consists of the four elements $\{1, 2, 4, 6\}$. As N_U=4, then the procedure repeats k or no more than 6 times. For instance, the initial schedule involves two columns corresponding to t_1 and t_2 time intervals. In the offspring solution two pairs related to sensors s_1 and s_3 in the two chosen columns change their values. Coverage strings of the parent and its offspring solutions are presented in the figure (left) and (right) respectevely.

The purpose of the crossover operator is to shift several columns from US toward ES or RS. But, during the evolution process, it may happen that, after applying crossover once, the chromosome does not improve its quality. Thus, we apply k-crossover operator, which is equivalent to use crossover k times to columns (1, 2), (1, 3), ..., (1, k+1) respectevely. k-crossover is executed on k pairs of columns from US consequently. In our example, after performing crossover for columns (1, 2) the operator will be continued for columns (1, 4) and (1, 6).

It is worth to notice that in the proposed algorithm crossover operator produces a single offspring from a single parent. This offspring will replace the parent if it is better or the parent will pass to the next generation. Therefore, at the beginning of evolution k is equal to 1. In the next generation k increases by 1 for the individual, if its fitness function value decreases.

Mutation is applied to each individual from the population. Mutation is based on reciprocal exchange of two gene's values in the individual. The first gene is taken from RS column with probability p_i:

$$p_i = \begin{cases} 0, & \text{if the } i-th \text{ gene in the column is equal to 0,} \\ \frac{1}{n_1}, & \text{if the } i-th \text{ gene in the column is equal to 1,} \end{cases}$$

where n_1 is a number ”1” genes in the column. Therefore, the first selected gene is equal to 1. The second mutated gene is taken as the first ”0” gene from US and from the same row as previous gene was. The selected genes swap their values. If there is no a gene with the value ”0” in US, mutation does not make any change in the chromosome.

Therefore, the first selected cell is equal to 1. The second selected cell is taken as the first ”0” cell from US and from the same row as previous cell was. The selected cells swap their values. If there is no a cell with the value ”0” in US, the parent solution coincides with its offspring.

Mutation is applied N_R times, once for each RS column.

An example of mutation is shown in Figure 5. A schedule constructed for a sensor network depicted in Figure 1 for seven consecutive time intervals. For coverage ratio q equal to 0.6 and coverage declination δ equal to 0.05, coverage string as follows {0.8, 0.16, 0.4, 0.24, 0.64, 0, 0.6}, where RS contains one element t_1. Therefore, mutation is executed once via changing the first column in the schedule. From the t_1 column with equal probabilities p_i = 0.16 for all cells of ”1”s values the one is taken, for an example the 4-th cell was selected. The chosen ”1” cell changes its value into ”0”, while the first ”0” cell of the corresponding row from US (in the picture such cell is from t_3 column) changes its value into opposite one. Coverage of the schedule for the first time interval (from the column t_1) decreases, while the coverage for the t_3 increases. Coverage string of the offspring solution becomes the following: {0.76, 0.16, 0.56, 0.24, 0.64, 0, 0.6}.

Lastly, deterministic selection is applied, where the best individual from each pair: a parent and its offspring passes to the next generation. These steps are repeated until stop condition is met. The best schedule from the last generation is a result of the algorithm.

VI. EXPERIMENTAL RESULTS

In this section, we present some results of experimental study of the proposed algorithms.

The experimental study was conducted in two steps. Firstly, several experiments were made in order to estimate the best values for parameters of the algorithms. The next step of experiments was to compare these two algorithms with the best parameters sets.

For evaluating the solutions, we rely on our network simulator, written in Java. The experiments were run on standard PC computer with two cores 1.66GHz CPU and 1GB RAM.

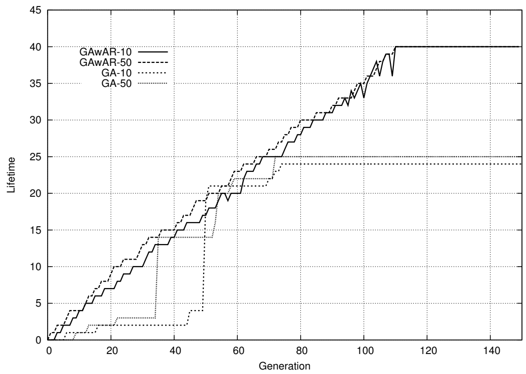


Figure 6. An examplary run for the best solution of algorithms: GAwAR-10, GAwAR-50, GA-10 and GA-50.

Sensors are randomly deployed over the target field F of dimensions $(L \times L)m^2$, where in all experiments we assume $L = 100$. POIs are uniformly distributed over the target field F in every g m, where $g = 20$.

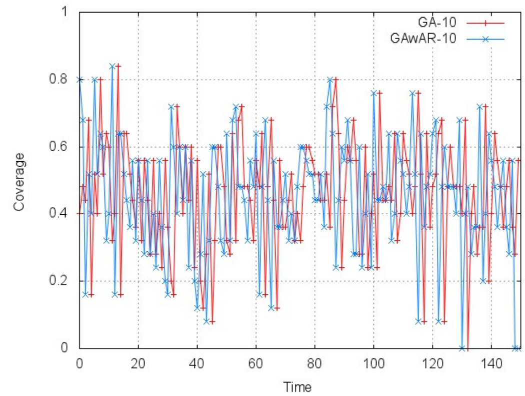
WSN consists of 100 sensors, sensing range R_s is equal to $20m$ and each node has a battery capacity $b = 10$. The size of an schedule solution is equal to $N \times T_{max}$, when $N = 100$ and $T_{max} = 150$. We consider that required coverage ratio is 90%, where q is equal to 0.91 and δ=0.01.

We assume two sizes of a population 10 and 50 individuals for each of considered algorithms, let us called them as GA-10, GA-50, GAwAR-10 and GAwAR-50 respectevely. The rest parameters: elite size is equal to 1, a number of competitors in a tournament m is equal to 2, crossover and mutation probabilities p_c and p_m are equal to 0.06 and 0.01. Evolution process of the algorithms is considered over 150 generations ($G = 150$).

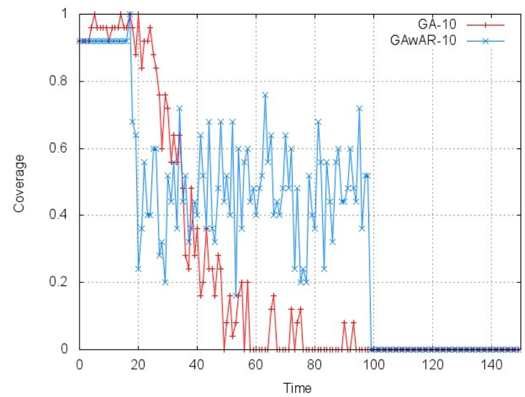
We run 5 times each of these algorithms: for five different randomly created WSN configurations. Exemplary runs with the best result for the first WSN instance of GA-10, GA-50, GAwAR-10 and GAwAR-50 are presented in Figure 6. Figure 7 presents the $CoverageString$ of the best solution obtained by GAwAR-10 and GA-10 for the first WSN instance in first (a), 50 (b), and the last (c) generation. The summarized results are presented in Table I. In the table from the left to right there are maximum, average with standard deviation values of Lifetime(0.9) obtained by each of the algorithms. The values in the two last column outlines a number of times a maximum solution by the algorithm was achieved and its average run time. Bold figures represent the best result for the WSN instance.

VII. CONCLUSION

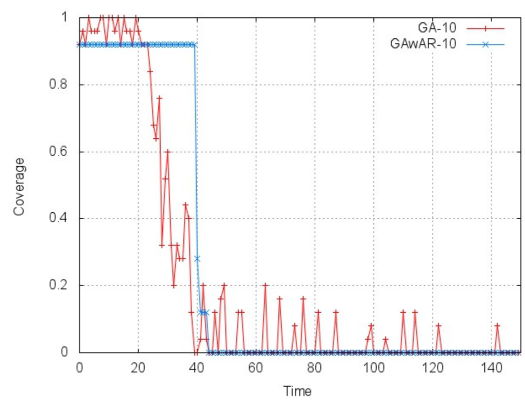
In this paper, we have proposed a novel evolutionary algorithm with asexual reproduction to solve MLC problem. Due to this approach it was possible to incorporate a specific knowledge about the problem into genetic operators of crossover and mutation. The performance of the proposed



(a)



(b)



(c)

Figure 7. CoverageString of the best solution obtained by GAwAR-10 and GA-10 in first (a), 50 (b) and the last (c) generation.

algorithm has been compared with one delivered by a standard GA. The preliminary results of experimental study of both algoritms have shown that the proposed GAwAR algorithm due to special design of its genetic operators significantly outperforms the standard GA-based approach, regarding to lifetime values and runnable time as well. The purpose of current and future studies is a more detailed study

 ISBN: 978-1-61208-310-0

Table I
LIFETIME(0.9): MAXIMAL, AVERAGE WITH STANDARD DEVIATION VALUES OF LIFETIME(0.9) AND NUMBER OF TIMES OF MAXIMUM ACHIEVED FOR GAWAR-10, GAWAR-50, GA-10 AND GA-50.

WSN	Algorithm	Max	Avg $\pm$ σ	Times of Max	T_{exec} [s]
Instance 1	GAwAR-10	**40**	38.8 $\pm$ 0.96	1	**21**
	GAwAR-50	**40**	**39.6 $\pm$ 0.24**	**3**	**111**
	GA-10	24	21.8 $\pm$ 1.46	1	39
	GA-50	25	23.4 $\pm$ 2.24	2	186
Instance 2	GAwAR-10	**33**	32.2 $\pm$ 0.56	2	**22**
	GAwAR-50	**33**	**32.6 $\pm$ 0.24**	**3**	**117**
	GA-10	19	18 $\pm$ 1.2	2	40
	GA-50	20	19 $\pm$ 0.4	1	231
Instance 3	GAwAR-10	**45**	42.8 $\pm$ 1.76	1	**22**
	GAwAR-50	**45**	**44 $\pm$ 0.8**	**2**	**127**
	GA-10	24	22.6 $\pm$ 1.04	1	36
	GA-50	26	23.8 $\pm$ 1.36	1	185
Instance 4	GAwAR-10	38	37.4 $\pm$ **0.24**	**2**	**23**
	GAwAR-50	**39**	**38 $\pm$ 0.4**	1	**122**
	GA-10	22	21.8 $\pm$ 0.16	4	39
	GA-50	23	22.2 $\pm$ 0.56	2	157
Instance 5	GAwAR-10	**55**	**53.2 $\pm$ 0.96**	1	**22**
	GAwAR-50	52	50.8 $\pm$ 0.96	2	**101**
	GA-10	31	29 $\pm$ 2	1	38
	GA-50	33	31.4 $\pm$ 1.04	1	162

of the proposed algorithm and using it for different variants of MLC problem.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their constructive comments and suggestions which improved the quality of the presentation.

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, ”Wireless Sensor Networks: A Survey,” Computer Networks, vol. 38, No. 4, Mar. 2002, pp. 393-422.

[2] C. P. Chen, and et al, ”CoCMA: Energy-efficient coverage control in cluster-based wireless sensor networks using a memetic algorithm,” Sensors, June 2009, pp. 4918-4940.

[3] M. R. Garey and D. S. Johnson, Computers and intractability: a guide to the theory of NP-completeness, W.H. Freeman and Co, 1979.

[4] N. D. Georganas and D. Tian, ”A coverage-preserving node scheduling scheme for large wireless sensor networks,” WSNA ’02 Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications, 2002, pp. 32-41.

[5] D. E. Goldberg, Genetic algorithms in search, optimization, and machine learning, Addison-Wesley Pub,1989.

[6] S. Samarah, A. Boukerche and X. Fei, ”A bio-inspired coverage-aware scheduling scheme for wireless sensor networks,” Proc. IPDPS Workshops, 2010, pp. 1-8.

[7] B. Wang, Coverage Control in Sensor Network, Springer-Verlag, London, 2010.

[8] G. Tao and Z. Michalewicz, Inver-over Operator for the TSP, Proceedings of the 5th Parallel Problem Solving from Nature LNCS, 1998, pp. 803-812.

[9] B. Wang, ”Coverage Problems in Sensor Networks: A Survey,” ACM Comp. Serv., 43, 4, October 2011, pp. 32-85.

[10] A.Tretyakova and F. Seredynski, Application of Evolutionary Algorithms to Maximum Lifetime Coverage Problem in Wireless Sensor Networks, 27 IEEE International Parallel & Distributed Processing Symposium, NIDISC Workshop, Boston, USA, May 2013.

[11] M. Cardei, M. T. Thai, Y. Li, and W. Andwu, Energy-Efficient Target Coverage In Wireless Sensor Networks. In Proceedings of the IEEE Conference on Computer Communications (InfoCom), 2005, pp. 1976-1984.

[12] C. E. Jones, K. M. Sivalingam, P. Agrawal, and J. C. Chen, A survey of energy efficient network protocol(I), Wireless Networks, 7 (2001), pp. 343-358.

[13] J. Kahn, R. H. Katz, and K. Pister, Next Century Challenges: Mobile Networking for Smurt Dust, ACM MOBICOM Conference, Aug. 1999, pp. 271-278.

[14] X. Y. Li, P. J. Wan, and O. Frieder, Coverage in Wireless Ad-hoc Sensor Networks, IEEE Transactions on Computers, Vol 52 (2002), pp. 753-763.

[15] M. Cardei and D. Z. Du, Improving Wireless Sensor Network Lifetime through Power Aware Organization, ACM Wireless Networks, Vol 11, May 2005, No 3, pp. 333-340.

[16] S. Slijepcevic and M. Potkonjak, Power Efficient Organization of Wireless Sensor Networks, IEEE International Conference on Communications, Vol 2, June 2001, pp. 472-476.

[17] S. Mini, K. Siba, K. Udgata, and L. S. Sabat, M-Connected Coverage Problem inWireless Sensor Networks, ISRN Sensor Networks, Vol 2012.

[18] C. Liu and G. Cao, Distributed Critical Location Coverage in Wireless Sensor Networks with Lifetime Constraint, INFOCOM, 2012 Proceedings IEEE, pp. 1314-1322.

[19] L. Ding, W. Wu, J. Willson, L. Wu, Z. Lu, and W. Lee, Constant-Approximation for Target Coverage Problem in Wireless Sensor Networks INFOCOM, 2012 Proceedings IEEE, pp. 1584-1592.

[20] Y. Liu and W. Liang, Prolonging Network Lifetime for Target Coverage in Sensor Networks, Wireless Algorithms, Systems, and Applications LNCS, Vol 5258, 2008, pp. 212-223.

[21] T. Zhao and Q. Zhao, Lifetime Maximization Based on Coverage and Connectivity in Wireless Sensor Networks Journal of Signal Processing Systems, Vol 57, 2009, pp. 385-400.

[22] A. Simões and E. Costa, Using Genetic Algorithms with Sexual or Asexual Transposition: a Comparative Study, Evolutionary Computation, 2000, pp. 1196-1203.

[23] B. Sahoo, V. Ravu, and P. Patel, Observation on using Genetic Algorithm for Extending the Lifetime of Wireless Sensor Networks. IJCA Special Issue on 2nd National Conference-Computing, Communication and Sensor Network (CCSN), 2011, pp. 9-13.