Matlab Simulations of Ad Hoc Sensors Network Algorithms

Amir J. Majid; Ph.D.
College of Engineering
Ajman University of Science & technology
UAE
abac.majid.a@ajman.ac.ae

***Abstract*—** Algorithms, implemented for the problem of ad hoc wireless sensors network, are simulated on Matlab platform, with a step-by- step evaluations of a case study. The main goal is to maximize the lifetimes of sensors by sharing sensors subsets, which cover a number of targeted zones, according to their minimum coverage failure probabilities. Different sensor subsets are activated according to their coverage failure probabilities, as well as a minimum specified value of coverage failure probability.

Keyword s- *sensor network, ad hoc, algorithm, failure probability, lifetime, coverage*

*****

I. INTRODUCTION

Sensor networks are normally deployed in areas of interest such as home appliances, healthcare applications, environment monitoring, etc. in order to collect information about events of these areas. Although using a large number of wireless low power ad hoc networks would be adequate, they are short lived, unreliable and limited radio range, memory and processing capacities[1]. An important function of these wireless sensor networks is to sense signals in remote and inaccessible environments, in which preserving their energy and prolonging network lifetime, is critical, and in which, their area coverage is to be maintained.

Area coverage can be resolved either by deploying sensors to cover sensing zones completely, or make sure that all zones are covered by a certain number of sensors, such as one-coverage or k-coverage [2][3], or select active sensors in a densely deployed network to cover all zones [4][5][[6][7][8]. The last case is known as an Activity Scheduling Problem (ASP) [9], which is divided into four classes: area, barrier, patrol or target coverage, in which this paper is focused on [10].

In order to maximize network lifetime and preserving zones coverage, many algorithms propose to organize sensors in a number of subsets, such that each set completely covers all zones, thus enabling time schedules for each subset to be activated at a time, thus removing redundant sensors which may waste energy and consequently reduce network lifetime [11]. To solve this problem, many algorithms are applied such as generic, linear programming, greedy algorithms [12][13][14][15][16]. One important technique is to improve reliability in cases when sensors may become unavailable due to physical damage, lack of power or malfunctioning.

In this paper, algorithms and their simulations of wireless sensor networks are implemented to include network lifetime reliability and lower failure probability of the sensor subsets which cover and monitor all zone targets. This problem has been addressed in the literature before; namely the α-Reliable Maximum Sensor Coverage (α-RMSC) problem. A number of algorithms, are introduced for a general S-T (sensor-target) coverage situation; each with a special task in a step-by-step simulation manner.

II. CASE STUDY

A general sensor-target (S-T) case is implemented in which three targeted zones are to be covered y four sensors, as depicted in fig.1, in which they are distributed randomly over a two dimensional planner view.

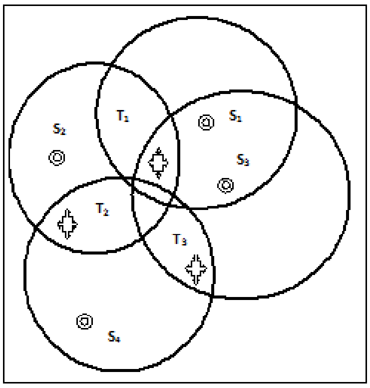

Figure 1; Planner view of four sensors and three target zones.

It can be seen that sensor S_1 covers target T_1 only, whereas S_2 covers T_1 and T_2, sensor S_3 covers T_1 and T_4, and sensor S_4 covers T_2 and T_3. It is assumed that two dimensional coverage is used with the sensors allocated apart from the targeted zones' centers. Thus each sensor covers each target with a certain failure probability value (*sfp*), ranging from 0 to 1. A value of sensor failure probability of 1 indicates no coverage.
Since each target is covered by one or more sensors, 100% coverage can be achieved in which alternative sensors alone or in groups, or subsets, can be switched on and off in such a way so that the lifetime of all sensors may be increased. The following figure depicts the values of *sfp* of each sensor to each target

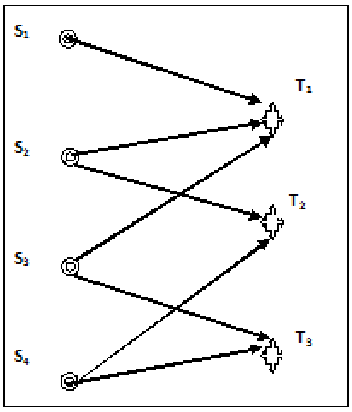

Figure 2; Sensor failure probabilities.

It can be seen that in order to insure 100% coverage of the targeted zones, there exists 9 possible sensor subsets or groups:{1,4}, {1,2,3}, {1,2,4}, {2,3}, {2,4}, {1,2,3,4}, {2,3,4}, {1,3,4} and {3,4}. The values of sensor-target failure probability are listed as shown in the following table:

Table I, Sensor failure probabilities

Sensor	Target	SFP
1	1	0.7
2	1	0.3
2	2	0.5
3	1	0.2
3	3	0.9
4	2	0.7
4	3	0.4

III PROBLEM FORMULATION:

An S-T coverage problem is for S sensors covering T targets according to failure probabilities of a number of different subset groups of sensors, in which the target failure probability *tfp* of *j* targets by $r \in [1, k]$ sensors subsets are:

$$Cfp_r = 1 - \prod (1 - tfp_j) \qquad (1)$$

$$tfp_j = \prod sfp_{ij} \qquad (2)$$

where sfp_{ij} is the failure probability of sensor i to target j, and cfp_r is coverage failure probability of a subset or group of sensors covering all targeted zones, which is assumed to be less than α; a predefined maximum failure probability *tfp* is target failure probability of one targeted zone by all sensors.

It's required to find these k sensors subsets activation in order to maximize the network lifetime as

$$\text{Max} \sum tw_k \qquad (3)$$

Where tw_k is lifetime of each sensor subset, with the assumption that lifetime of each sensor is normalized to a value of *1*.

IV. Input Modules

The algorithms of solving the α-RMSC problem depends on the way of defining sensors-targets coverage and inputting sensors failure probabilities as follows:

1. Inputting (n-m) data for n sensors to m targets as shown. Entered value 1 indicates no coverage for the specific S-T pair.

```
function In(n,m)
for i=1:n
for j=1:m
sfp(i,j)= input(['Input Sensor Failure
Probability from Sensor ' num2str(i) ' to
Target 'num2str(j) '= ']);
end; end;
disp(' '); disp('sfp=  '); disp(sfp);
end
```

Calling this function script will lead to a 4 x 3 matrix of sensors failure probabilities as shown below:

```
>> In(4,3)
Input Sensor Failure Probability from Sensor 1 to
Target 1= 0.7
.. (12 values must be inputted)
sfp=
    0.7000    1.0000    1.0000
    0.3000    0.5000    1.0000
    0.2000    1.0000    0.9000
    1.0000    0.7000    0.4000
```

2. The following algorithm is used by inputting vector *d* of decimal numbers for each of the *n* sensors covering *m* targets

```
function sfp=Input_Decimal_to_Binary(d,n,m)
for i=1:n
while d(i) ~= 0
for j=1:m
if rem(d(i),2)==0
bb( m-j+1)=0;
else bb( m-j+1)=1;
end
  d(i)=floor(d(i)/2);
end; end;
sfp(i,:)=bb;
end
disp(sfp);
for i=1:n
for j=1:m
if sfp(i,j)~=1;
sfp(i,j)=input('Input Sensor Failure
Probability = ')
else sfp(i,j)=1;
end; end; end;
disp(sfp);
end
```

 ISSN: 2321-8169
288 - 293

i.e. entering binary sensors data, in which each sensor is assigned to a decimal number corresponding to binary digits for each target, then the failure probabilities for only active sensors to targets are entered For example; a 7 assigned to sensor 2 of a S-T system of 3-4, corresponds to sensor 2 covering targets 1-2-3. The following output screen demonstrates this:

```
>> Input_Decimal_to_Binary(d,4,3)
Input Sensor 1 decimal_number = 4
Input Sensor 2 decimal_number = 6
Input Sensor 3 decimal_number = 5
Input Sensor 4 decimal_number = 3

Sensor-Target matrix is:

  1   0   0
  1   1   0
  1   0   1
  0   1   1

Input Sensor Failure Probability = 1 to 1 = 0.7
Input Sensor Failure Probability = 2 to 1 = 0.3
Input Sensor Failure Probability = 2 to 2 = 0.5
Input Sensor Failure Probability = 3 to 1 = 0.2
Input Sensor Failure Probability = 3 to 3 = 0.9
Input Sensor Failure Probability = 4 to 2 = 0.7
Input Sensor Failure Probability = 4 to 3 = 0.4

 0.7000   1.0000   1.0000
 0.3000   0.5000   1.0000
 0.2000   1.0000   0.9000
 1.0000   0.7000   0.4000
```

V. TARGET FAILURE PROBABILITY

This algorithm is to calculate failure probability of all sensors (i=1 to n) to target *j* (j=1 to m), according to the formula:

$$tfp_j=\prod sfp_{ij} \quad (4)$$

where sfp_{ij} are sensor failure probabilities for a number of sensors to any target. The following algorithm does this, followed by execution run for a different values of subsets; i.e. {1}, {1,3} and {1,2,3,4}:

```
function tfp=Target_Failure_Probability(sfp,in,m)
for i=1:m
tfp(i)=1;
for j=in
tfp(i)=tfp(i)*sfp(j,i);
end; end;
disp(' '); disp('Target Failure Probability for
sensors subset :');
disp (in); disp('is ='); disp(' ');
disp(tfp);
end
```

```
>> Target_Failure_Probability(sfp,in,3)
Target Failure Probability for sensors subset :
     1

   is =

    0.7000    1.0000    1.0000

>> Target_Failure_Probability(sfp,in,3)

Target Failure Probability for sensors subset :
    1     3

   is =

    0.1400    1.0000    0.9000


>> Target_Failure_Probability(sfp,in,3)

Target Failure Probability for sensors subset :
   1     2     3     4

   is =

    0.420   0.3500    0.3600
```

It can been seen that as the number of sensors in one subset increased, faillure probaiity is reduced. Note that a failure probabillity of 1 in one of the targets, idicate no coverage to that target zone. So for the case of {1}, targets 2 and 3 are not covered, and for subset {1,3}, target 2 is not covered.

VI. SENSOR COVER FAILURE PROBABILITY

This algorithm is to calculate the coverage of the *k* sensors subsets to the *m* targets, according to $scfp_r=1-\prod(1-tfp_j)$, where $r \in [1, k]$; in which target failure probability *tfp* is entered as a vector for the *m* individual targets.

```
function [scfp]=Sensor_Cover_Failure_Probability(tfp,in,m)
disp(' '); disp('Sensor Cover Failure Probability for sensors :
');
disp(in); disp(' is = ');
for i=1:in
scfp(i)=1;
for j=1:m
   scfp(i)=scfp(i)*(1-tfp(j));
end
scfp(i)=1-scfp(i);
end
disp([scfp]);
end
```

The execution of this algorithm function script is as follows:

```
>> Sensor_Cover_Failure_Probability(tfp,in,3)

Sensor Cover Failure Probability for sensors :
   1    2    3    4
 is =
   0.6015
```

There are 9 possible sensor subsets among the maximum of 2^4 possible subsets, which can cover all targeted zones. These subsets are: {1,4}, {1,2,3}, {1,2,3,4}, {1,2,4}, {1,3,4}, {2,3}, {2,4}, {2,3,4}, {3,4}. The Sensor subset coverage of these 9 subsets, are simulated and the result is listed in the following table:

Table II; Nine sensor subsets, shown in the first row against coverage failure probabilities shown in the second row

{1,4}	{1,2,3}	{1,2,3,4}	{1,2,4}	{1,3,4}	{2,3}	{2,4}	{2,3,4}	{3,4}
0.9460	0.9521	0.6015	0.6919	0.8349	0.9530	0.7270	0.6090	0.8464

It can be shown that this failure probability is minimum for subset {1,2,3,4} in which all sensors are active, whereas it's maximum when only 2 or 3 sensors active as a group in a subset. Note that there exists no subset with only one sensor to cover all targets, as formulated in this case example.

VII. Alpha-Reliable Maximum Sensor Coverage:

In this algorithm, firstly all sensors coverage failure probability are checked to be less than a certain assigned value of α, as listed in this algorithm:

```
function coverage=Less_Min_Coverage(scfp,k,alpha)
disp(['Sensor subsets less then alpha of  '
num2str(alpha) ' are = ']);
count=0;
for i=1:k
if scfp(i) < alpha
   count=count+1;
coverage(count)=scfp(i);
end; end;
disp (coverage);
end
```

Running this function script will list the following output:

>> Less_Min_Coverage(scfp,k,alpha)
Sensor subsets less then alpha of 0.75 are =
0.6015 0.6919 0.7270 0.6090

These coverage failure probabilities correspond to the following subsets: {1,2,3,4},{1,2,4},{2,4}and {2,3,4}.

Different values of α can be chosen to select appropriate active sensor subsets; the higher α value the more subset choices. As shown above, 4 subsets are selected for α<0.75. It can deduced, that in order to maximize life time of sensors, it would be appropriate to activate may sensor subsets to operate at different times, thus elongating their lifetime. But this would be on the expense of coverage failure probabilities.

Then the weight factor indices *w's* are assigned to each sensor as well as to each target according to the importance of contributing sensors and targets to be covered. These weight indices are dependent on several factors, such as priority of targeted zones or sensors reliability, and therefore they will be included in the coverage lifetime of the contributed subsets. For evenly distribution of sensors and targets priorities, a value of unity is assigned to all *w's*
Then, the maximum network lifetime is calculated according to $Max \sum tw_k$ as shown in the following algorithm:

```
function coverage_time=Coverage_Time(coverage,w)
disp(['Coverage Times for Subsets ' num2str(ss) ' are=
'])
tt=0;
ii=length(coverage);
for i=1:ii
tt=tt+(1-coverage(i));
end
for i=1:length(coverage)
coverage_time(i)=(1-coverage(i))*ii * w(i)/tt;
end
disp (coverage_time);
end
```

Running this algorithm function script would list the following coverage times. It is assumed, that each sensors lifetime is normalized to reference value of 1:

>> Coverage_Time(coverage,w)
Coverage Times for Subsets are=
1.1630 0.8992 0.7967 1.1411

It can be seen that only subset {1,2,3,4}, {1,2,4} and{2,4} can be used for maximum lifetime of [1.1630 +0.8992 +0.7967]= 2.8589 times the sensor lifetime which is assumed to be 1; i.e. 285%.

The following tables depict subsets' coverage times together with the respective sensor subsets for each value of alpha from approximately 0.6 to 0.95

Table III, α<0.65

alpha=0.65	Subsets SS	1,2,3,4
	Coverage Failure Probability CFP	0.6015
	Coverage Time CT	1

As shown, there is only one subset for this α value, in which all sensors 1,2,3 and 4 operate in the same time, making network lifetime to be a maximum of 1.

Table IV, α<0.75

alpha=0.75	SS	1,2,3,4	1,2,4	2,4	2,3,4
	CFP	0.6015	0.6919	0.7270	0.6090
	CT	1.1630	0.8992	0.7967	1.1411

 ISSN: 2321-8169
288 - 293

It can be seen that as α is increased to 0.75, four subsets are assigned to operate at different times, thus elongating network lifetime to 2.8589 times the sensor lifetime.

Table V, α<0.85

alpha=0.85						
SS	1,2,3,4	1,2,4	1,3,4	2,4	2,3,4	3,4
CFP	0.6015	0.6919	0.8349	0.7270	0.6090	0.8464
CT	1.4154	1.0943	0.5864	0.9696	1.3887	0.5456

As shown here, 6 different subsets re selected for α<0.85, elongating network lifetime further to 3.45, whereas Table VI depicts maximum network lifetime of 3.85 when a value of α of less than 0.95 is chosen. Seven different sensors subsets are selected, on the expensive of high failure probability values.

Table VI, α<0.95 (for the following two tables)

alpha=0.95			
Subsets (SS)	1,4	1,2,3,4	1,2,4
Coverage Failure Probability	0.9460	0.6015	0.6919
Coverage Time	0.2168	1.6001	1.2371

alpha=0.95				
SS	1,3,4	2,4	2,3,4	3,4
CFP	0.8349	0.7270	0.6090	0.8464
CT	0.6629	1.0962	1.5700	0.6168

The above results of different values of α from 0.65 to 0.95, are plotted against network lifetimes as shown in the following figure:

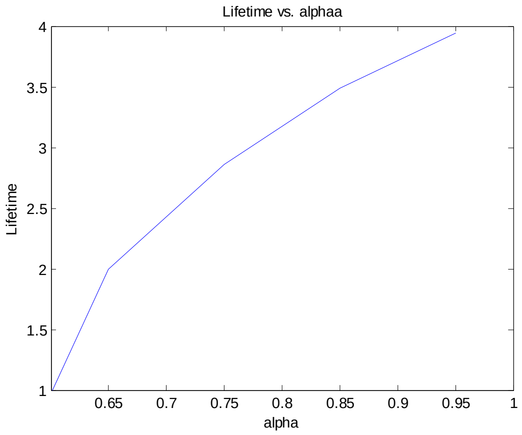

Figure 3, Network lifetime against failure probability

It can be seen that as demanded failure probability is increased, network lifetime is increased, but saturated to a maximum value of 4 since there are 4 sensor lifetimes which can be operated individually. Further, it is shown that network lifetime is dropped to a value of 1 when α reaches the value of least failure probability of all sensors. This would reduce options of manipulating with failure probabilities and the network lifetime options. As stated, if one sensor is shared with more than one subset, then the total activation time of that sensor cannot exceed its normalized lifetime.

To find all sensors coverage contributing subsets, a more general algorithm can be used in which there exists a maximum of 2^n subsets for n sensors. Subsets which contain only one sensor, cannot increase network lifetime more than reference 1. Such implementation of switchable sensors subsets can be achieved using microcontrollers or smart sensors.

VIII. Conclusion

Algorithms of ad hoc wireless sensors network, have been implemented and simulated on Matlab platform. The sensors lifetimes have been maximize according to the coverage failure probabilities of all sensors subsets which cover all target zones. The algorithms can be applied to any number of sensors and target zones in any random fashion.

A case study of 4 sensors targeting 3 zones, has been used in a step-by-step simulations, for different values of alpha (minimum sensors coverage failure probability) from 0.65 to 0.95, in which the normalized network lifetime is increased from 1 to 3.85

As expected, it has been shown that network lifetime can be increased with increasing alpha as well as reducing individual sensors failure probabilities.

REFERENCES

[1] I.F. Akyildiz, et.al., "A Survey on sensor networks", Communication Magazine, IEEE, 40(8), pp. 102-114, 2002

[2] Y.C. Wang, et. Al., "Efficient placement and dispatch of sensors in a wireless sensor network", IEEE Transactions on Mobile Computing, 7(2), pp. 262-274, 2008.

[3] C.F. Huang, et.al., "The coverage problem in a wireless sensor network, Mobile Networks and Applications", 10(4), pp. 519-528, 2005.

[4] T. Yan, et.al., "Differentiated Surveillance for sensor networks, Proceedings of the 1st International Conference on Embedded Networked Sensor Systems", pp. 51-62, ACM, 2003.

[5] M. Cardei, et.al., "Energy-efficient target coverage in wireless sensor networks", IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies Proceedings, Volume 3, pp. 1976-1984, 2005.

[6] S. Kumar, et.al., "Barrier coverage with wireless sensors", Proceedings of the 11th Annual International Conference on Mobile Computing and Networking, pp. 284-298, ACM, 2005

[7] C. Gui, et.al., "Virtual patrol: a new power conservation design for surveillance using sensor networks", IEEE Proceedings of the 4th International Symposium on Information Processing in Sensor Networks, pp. 33, 2005.

[8] K.P. Shih, et.al., "On connected target coverage for wireless heterogeneous sensor networks with multiple sensing units", Sensors 9(7), pp. 5173-5200, 2009.

[9] S.Y. Wang, et.al., "Preserving target area coverage in wireless sensor networks by using computational geometry", IEEE Wireless Communications and Networking Conference (WCNC), 2010 IEEE, pp. 1-6, 2010.

[10] K.P. Shih, et.al., "A distributed active sensor selection scheme for wireless networks", IEEE Computers and Communications, Proceedings of the 11th IEEE Symposium, pp. 923-928, 2006.

[11] J. Chen, et.al., "Modeling and extending lifetime of wireless sensor networks using genetic algorithm", Proceedings of the First ACM/SIGEVO Summit on Genetic and Evolutionary Computations, pp. 47-54, ACM, 2009.

[12] H. Zhang, et.al., "A distributed optimum algorithm target coverage in wireless sensor networks", IEEE Asia-Pacific Conference on Information Proceedings, Volume 2, pp. 144-147, 2009.

[13] H. Zhang, "Energy-balance heuristic distributed algorithm for target coverage in wireless sensor networks with adjustable sensing ranges", IEEE Asia-Pacific Conference on Information Proceedings, Volume 2, pp. 452-455, 2009.

[14] A. Dhawan, et.al., "A distributed algorithm framework for coverage problems in wireless sensor networks", International Journal of Parallel, Emergent and Distributed Systems, 24(4), pp. 331-348, 2009.

[15] J. Wang, et.al., "Priority-based target coverage in directional sensor networks using genetic algorithm", Computers & Mathematics with Applications, 57(11), pp. 1915-1922, 2009.

[16] Z. Hongwu, et.al., "A heuristic greedy optimum algorithm for target coverage in wireless sensor networks", IEEE Pacific-Asia Conference on Circuits, Communication and Systems, pp. 39-42, 2009.