Energy Efficient Routing in Wireless Sensor Networks

Nurefşan Sertbaş Bogazici University, Department of Computer Engineering
34342, Bebek, Istanbul, Turkey

I. Abstract

Energy conservation in sensors is one of the most significant design issues in wireless sensor networks. We propose an algorithm which aims to maximize the lifetime of the network by minimizing the energy consumption of the nodes. We measure the network lifetime in terms of equal periods of time named rounds. We follow a methodology that based on modification of the Dijkstra algorithm. We use different evaluation metrics and we evaluate our method using these metrics. Based on the simulation results we show that our algorithm performs better than early version of the Dijkstra algorithm.

II. Introduction

In this project, we aim to design and implement a simple wireless sensor network routing by considering it as an optimization problem with multiple constraints.

A. Architecture of wireless sensor networks

First of all is important to emphasize that sensors are small devices that has the ability of short distance communication. They are constrained in bandwidth and battery [1].

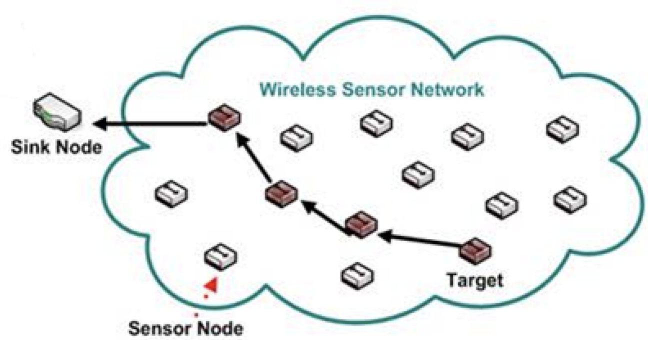

Fig. 1. Simple sensor network architecture [2]

Basically, each sensor collects data from its sensing range and transmit this data to the gateway node which is called as sink [3]. Then, sink transmit this data to some remote control station. Therefore, it is essential to use these resources effectively during the transmission with the help of efficient routing strategy. Additionally, the topology of the sensor networks changes very frequently for many cases so it is very important to track the new topology dynamically and update the routing paths accordingly [3].

B. Routing in Wireless Sensor Networks

As we mentioned before, WSNs suffer from several limitations such as energy, bandwidth, central processing unit, and storage [4]. It means that routing in an efficient manner plays important role in WSNs.

In literature, several routing protocols has been proposed taking different criteria into account. Determining routing path according to the maximum available power, minimum number of hopes or minimum energy can be the best for special cases [3]. In Sankar and Liu [5] a proposed method aims to maximize the network lifetime with flow conservation and energy limitation constraints. Similarly, a model to maximize the nw lifetime under the battery and capacity constraints has been worked in [6]. A multi-objective routing protocol has been proposed in [7] to maximize the lifetime, minimization of energy consumption, minimization of delay and secure routing. Also, mobility based, location based, Qos based routing approaches are very commonly used in different cases [4].

III. Modelling and Problem Formulation

A. Problem Statement

Wireless sensor network optimization problems are generally multi-objective which means that there are more than one desirable criteria compete with each other. And, the designer should choose the optimum scenario. For our problem there exist several parameters; some of them are given and some of them can be selected by the designer (system parameters).

1) Given Workload Parameters: The total number of nodes to be located and their location information

- Number of nodes
- Number of Sinks
- Physical Topology
- Coverage
- Sensor characteristics
  -Initial energy
  -Sensing power
  -Standby power
  -Tx/Rx power
  -Sensing range

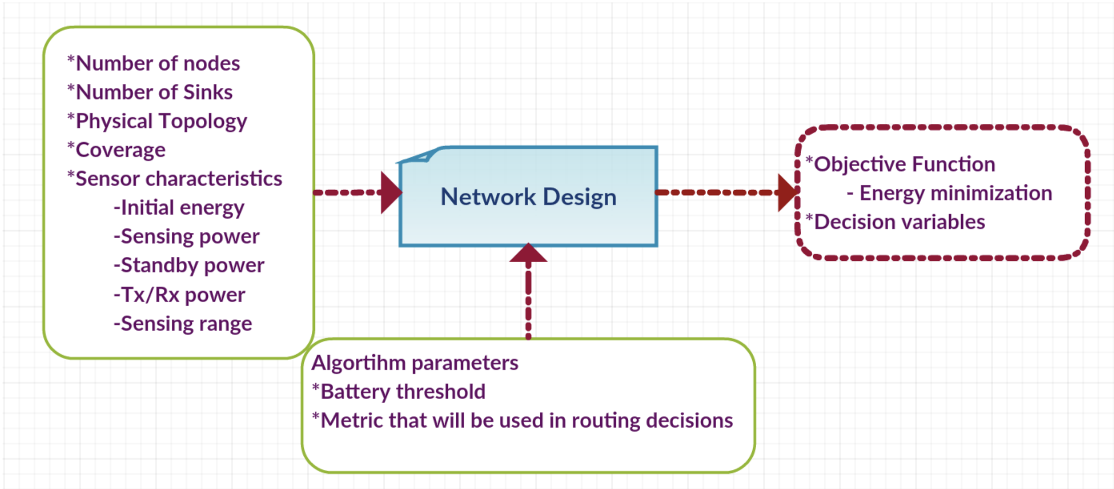

Fig. 2. Problem input and outputs

2) System parameters: There exist algorithm parameters that designer is able to set optimum values.

- Battery threshold that will be used in the routing algorithm to be considered node as unavailable
- Metric that will be used in routing algorithm

B. Objective Function and Constraints

Our main objective is maximizing the network lifetime by minimizing energy consumption. For this aim we should formulate the problem first then, give the optimization equations. Assume that N is the set of all sensor nodes in the network and S is the set of all sinks in the network. Initially, power consumption of node i will be stated as [8]

$$P_{ij} = E_{standby} + E_{sensingpower} + E_{Tx} * f_{ij} + E_{Rx} * f_{ji} \quad (1)$$

where f_{ij} is the flow term and P_{ij} is the total power consumption of the node i when transmitting flow to node j. This will be used in calculating residual energy on that node.

Also, we need to define a term $EdgeW_{ij}$ (edge weight will be used) which is the edge weight between node i and j. Note that we will define edge weight in terms of the consumed power on that edge (by using loss model related with distance) and the function of the residual energy. In that definition, the edge weight proposed in [9] is used by ignoring some terms for simplicity. Thus, the optimizations problem is stated as as follows.

1) Objective Function: Our objective is minimizing energy consumption which can be formulated as

$$\sum_{i,j \in N} EdgeW_{ij} * f_{ij} \quad (2)$$

2) Constraints: Under the following constraints
From the flow conservation principle we can define the data produced at node i as G_i = outgoing data incoming data

$$\sum_{i,j \in N} f_{ij} - \sum_{i,j \in N} f_{ji} \quad (3)$$

where $\forall i \in N$.

*Considering the link/edge capacities should not be violated

$$0 < f_{ij} < C_{ij} \quad (4)$$

where C_{ij} is the link/edge capacity.

*Also, flow values which are defined as packet rate should be integer values

$$f_{ij} \in Z^{+} \quad (5)$$

*The sink should not transmit data to sensor nodes

$$\sum_{i \in S, j \in N} f_{ij} = 0 \quad (6)$$

*Consumed power should be equal or less than the battery of the node. Assume that initial battery of the node i is represented as E_{init} and consumed energy is represented as E_i then

$$E_{init} \geq E_i = \sum_{i \in N} E_{Tx} * f_{ij} + \sum_{i \in N} E_{Rx} * f_{ji} \quad (7)$$

3) Decision Variables: We have two decision variables as following
f_{ij} : Amount of flow from node i to node j
E_i : Energy of node i

Constraint (5) ensures that the decision variable f_{ij} takes a non-negative value. Constraint (7) ensures that the decision variable E_i takes value which is greater or equal than the initial energy of the node.

C. System Model

We have made the following assumptions about the network, then we implement our algorithm for the network.

- We consider only proactive sensor networks where each node generates equal amount of data per time. unit[10]. We assume that each packet has same length which is 200 bits.
- Energy loss for the transmitter or receiver circuitry is constant per bit communicated[11].
- Energy spent in transmitting a bit over a distance d is proportional to d^2.
- We assume that channel access is collision free for all the nodes.
- We use equal periods of time called rounds. At the beginning of each round path from node to sink is computed
- The sensing power of the sensors is ignored because it is too smaller than the Tx/Rx power.
- We do not consider the sleep states of nodes. It is outside the scope.
- Delay and packet loss will not be considered in the evaluation phase because of the same reason.
- We assumed that number of sink is given as 1 and it is placed at the center of the field.

IV. Methodology

A. Description of the solving technique

In order to minimize energy consumption, we implement a Dijkstra shortest path algorithm with different metrics instead of distance. We tried to modify shortest path approach because it is already used for minimizing energy consumption in several works [12]. The new metric can be formulated as,

Metric for the edge between i and j

$$\frac{F_{residual}}{d^n a} \tag{8}$$

where

$F_{residual}$is the sum of residual energies of node i and node j,
d is the distance between node i and node j,
n is the radio transmission exponent,
a is the number of neighbor nodes.

We have used new metric given in Eq. 8 in Dijkstra algorithm. Each term has a contribution to the solution as following

- Nodes with higher energy should be preferred so, $F_{residual}$ is at numerator.
- The nearest node should be selected as much as possible. Therefore, d is at denominator.
- If the number of neighbors are too many, it is likely that adjacent nodes will forward their data through that node. For instance, assume that there is a route like 1-3-5 to transfer node 1s data to sink (node 5). At node 3 the consumed energy is increased by $E_{Tx} + E_{Rx}$ for forwarding node 1s data.

Moreover, we have used a battery threshold in decision phase. In other words, if the battery of the node reaches the threshold, it does not accept any other nodes traffic it just sends its own data. For the rejected node, the process is skipped for one round and new route is found in the next round. From the energy consumption perspective, some nodes consumed most of their energy faster than the others. It may make the network disconnected [13]. Also, nodes which are close to sink likely to be dead before the other nodes because they consumed their energy for forwarding other nodes' traffics additional to their own traffic. There should be a mechanism to overcome this problem and increase the lifetime of the network. Therefore, we use threshold considering that problem.

Note that, the edge weights are dynamically changing in each round and we select the min one in each round.

B. Modified Dijkstra Algorithm for Energy Optimized Routing

Pseudo code for the implementation [14]

1) Initialize all nodes with infinite metric
2) Start from source node and calculate edge metric based on the formulation given in Eq. 8 for all neighbor nodes if exists. Otherwise, there is no any node in its sensing range, go step 1 and execute algorithm for the next node.
3) If the calculated one is smaller than the current one, update the edge weight with the new one (Include min weighted edge in path)
4) Continue until the destination node(sink)
5) -If there is a path from source to destination, check the battery of the nodes on that path. If any node had been reached its threshold already, go Step 1 and re-execute the algorithm for same node.

   -If there is no path from source to destination or there exists a path and all nodes on that path have battery under threshold assigned that path to the node. Send data over that route. Repeat the procedure for remaining nodes.

The details of the algorithm and how it operates are given in Fig. 3 as a flow chart. It is assumed that there exist N sensors in the network.

V. Simulation

We have implemented the algorithm given in Section IV. For that aim, we have used MATLAB as a environment and also visualization tool.

We have considered a scenario that each sensor tries to send data with same periods like t, 2t,3t, . Therefore, time interval between in each execution of the simulation method is assumed as t. The detailed explanation can be found in code manual.

As an example, the number of packets both received and transmitted is given in below bar plot. From the figure, we can say that node-2, node-17 and node-18 are likely to close the sink because they are used to forward other nodes' packets to sink.

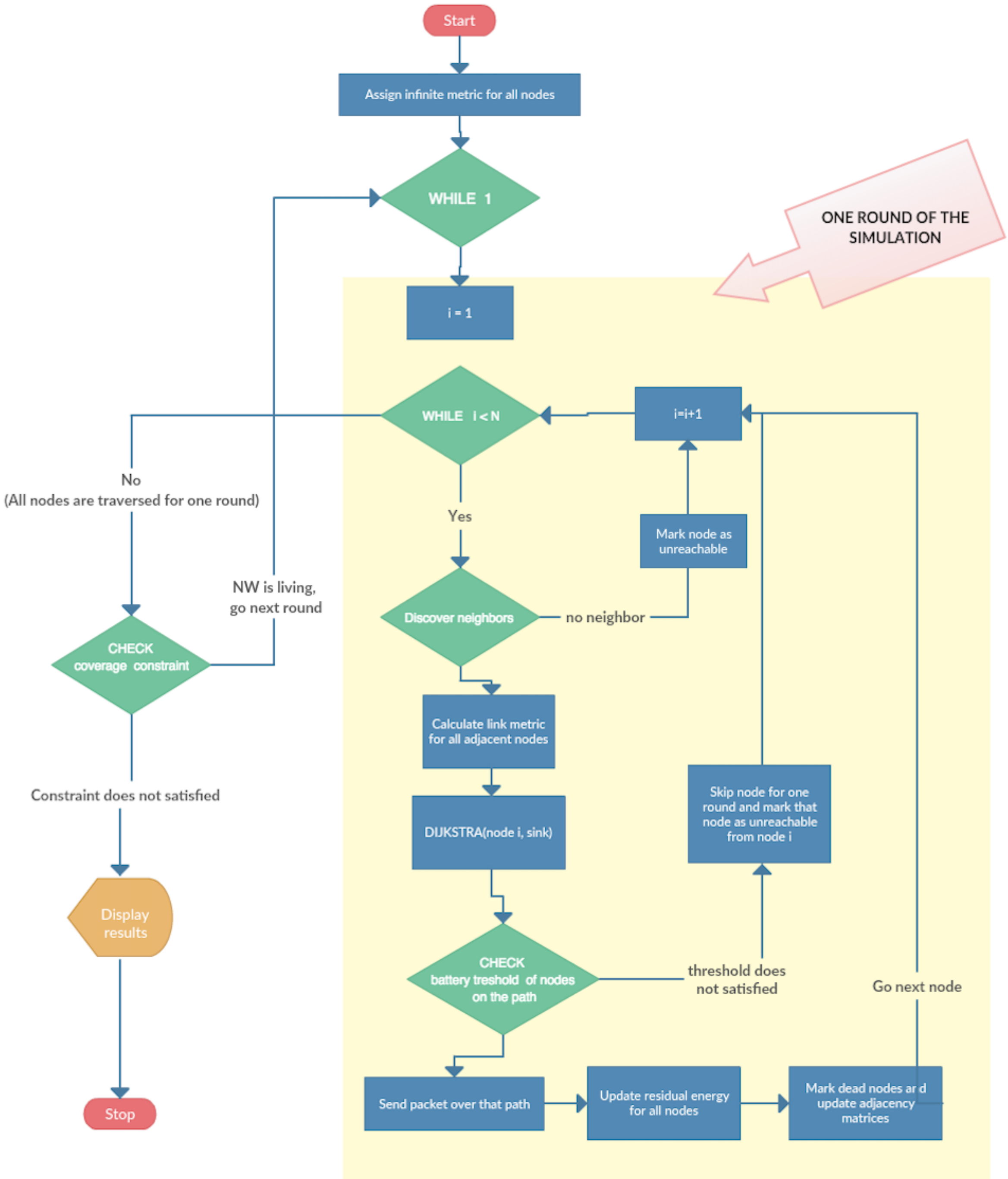

Fig. 3. Flowchart for the proposed algorithm

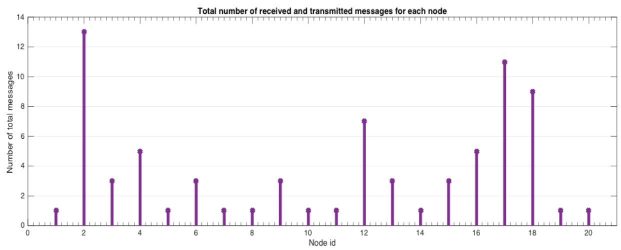

Fig. 4. Total number of packets per each node

Also, we have checked that the implemented algorithms are performed properly and satisfies the predefined constraints. For example, the flow matrix is given below for 20 nodes and 1 sink (node-21). First row represents the node ids; second row represents total number of received packets. Similarly, total number of transmitted packets is given in third row.

It can be said that, constraint 6 is satisfied. As clearly seen from the 4th row and 21st column, the sink(node-21) does not transmit data to sensor nodes.

4x21 double

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
2	0	6	1	2	0	1	0	0	1	0	0	3	1	0	1	2	5	4	0	0	20
3	1	7	2	3	1	2	1	1	2	1	1	4	2	1	2	3	6	5	1	1	0
4	1	13	3	5	1	3	1	1	3	1	1	7	3	1	3	5	11	9	1	1	20

Fig. 5. Flow matrix

A. Simulation parameters

The parameters used in simulation are given in Table I.

TABLE I. SIMULATION PARAMETERS

Einit	0.5 joule
Etx	100 njoule/bit
Erx	50 njoule/bit
Packet size	200 bits

We have used fixed size packets each one is 200 bits. Therefore, 2.10^{-5} joule energy is consumed for transmitting and 10^{-5} joule energy is consumed for receiving a packet [11].

Note that we have used

- MATLAB R2015b as a simulation environment
- Mac OSX 2,9 GHz Intel Core i5, 16 GB 1867 MHz DDR3 as a processing hardware

B. Case Study: How the simulation work over minimal example

In this part, we run the simulation for a small example and the system parameters are given in the below table.

We have created 5 nodes and 6th node as a sink node. The randomly deployed topology is given below.

TABLE II. USED PARAMETERS

Number of nodes	5
Battery threshold	60
Coverage threshold	80
Sensing range	20
Field	60*60
Random Topology	

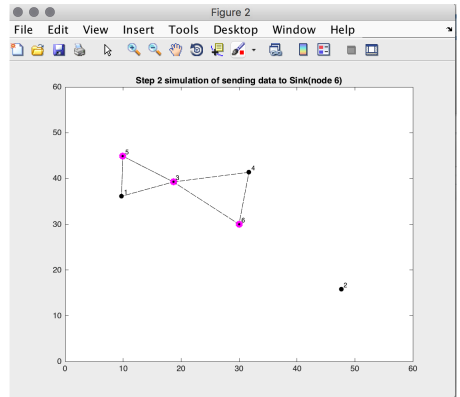

Fig. 6. Randomly created topology

Command Window

```
Simulation has been started with 5 nodes
Creating topology..
Network visualizing...
** Round 1 **
     1     3     6

     2     6

     3     6

     4     6

     5     3     6

Node 2 is out of the range (no communication)

    0.4982    0.5000    0.4871    0.4974    0.4978
```

Fig. 7. Starting of the simulation

Then, all nodes discover their neighbors and distances are measured.

6x6 double

	1	2	3	4	5	6
1	0	0	9.4726	0	8.7726	0
2	0	0	0	0	0	0
3	9.4726	0	0	13.2084	10.3991	14.6208
4	0	0	13.2084	0	0	11.4812
5	8.7726	0	10.3991	0	0	0
6	0	0	14.6208	11.4812	0	0

Fig. 8. Distance between nodes

Using the formula in Eq.8, edge weights are calculated and the algorithm is executed with this metrics. If battery constraint is satisfied the flow is sent through the founded path.

6x6 double

	1	2	3	4	5	6
1	0	0	717.8362	0	307.8372	0
2	0	0	0	0	0	0
3	717.8362	0	0	1.3957e...	865.1285	1.7101e...
4	0	0	1.3957e...	0	0	527.2743
5	307.8372	0	865.1285	0	0	0
6	0	0	1.7101e...	527.2743	0	0

Fig. 9. New edge metrics between nodes(initial case)

Note that, battery threshold is 60 and E_{init} is 0.5 therefore, if battery of the any node on the path decreases below to the 0.3 it starts to reject incoming traffic, it only transmits its own traffic. After the round 16, node 3 reach its battery threshold and changes operating mode to only-transmission.

```
** Round 16 **
     1     3     4     6

     3     4     6

     4     6

     5     3     4     6

    0.4699    0.5000    0.3096    0.3921    0.4640
** Round 17 **
     1     3     4     6

     3     4     6

     4     6

     5     3     4     6

    0.4680    0.5000    0.2986    0.3811    0.4617
Node 3 reach its treshold, status changed to only Tx mode
```

Fig. 10. Node 3 reaches to battery threshold

Also, we used 3.10^{-5} as a level to assumed to be dead. We thought that if sensors have sensitive data there may required some additional operations to preserve secrecy (eg. revocation problem). There as you can see from following output node 3 reaches that value and counted as dead.

Note that node 2 is out of the sensing area. In other words it has no communication with any other nodes therefore coverage is 80/100 as initially. The simulation lasts until first node die because coverage decreases to 60/100 as can be seen from following output.

```
Node 4 reach its treshold, status changed to only Tx mode
Node 3 dead


****End of the simulation****
Num of living nodes 3  out of 5, after 44 rounds
Num of nodes out of the coverage is 1
Simulation has been done with 4 active nodes
Total consumed energy in network: 1.1225
Consumed energy per sensor: 0.2806
**** **** **** **** **** ****

Total number of received messages by sink is 176
Elapsed time is 195.090788 seconds.
>>
```

Fig. 11. Simulation ends due to the coverage constraint

VI. Experiments and Results

In the last part of the paper, we aimed to show the effects of workload and system parameters on the performance. Therefore, we will use several metrics in the evaluation.

A. *Phases of Evaluation*

In the first phase, we aimed to show the effect of different factors by giving two leveled values. For instance, we will run our algorithm in a small area and large area for this phase. Then, we will examine the effect of the field size incrementally in the second phase. Assuming that there exists k factors which means that 2^k different case is exist. Again assuming that r is the number of repetitions for each case, we will have $2^k.r$ experiments in this part. We used 5 different factors and repeated each case 5 times. Then in the second phase, we repeat the experiments for the intermediate values to model the behavior of the algorithm.

Note that, we use Dijkstra shortest path algorithm as a competitor although its objective is shortest path rather than energy minimization. However, we followed such a way because

- Distance has a significant role in energy consumption proportional with d^2. Therefore, it is important to use shortest paths as much as possible
- There exist several works by using same approach with us.
- We modified Dijkstra algorithm and try to make it energy efficient.

But it is still may sound unreasonable. Therefore, we have also used statistical performance results of the algorithm as a alternative evaluation mechanism aiming to show quality of the results.

1) Effect of different number of nodes: We kept constant other parameters as given in below table

TABLE III. Used parameters

Battery threshold	60
Coverage threshold	80
Sensing range	20
Field	50*50
Random Topology	

TABLE IV. N=10 NODE, MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	102	3.5852	988.1666
2	72	2.1635	655.9035
3	66	4.4455	710.3966
4	76	3.6446	808.6829
Mean	79	3.4597	790.7874

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	48	2.7936	505.1928
2	61	1.9990	558.1920
3	64	4.1053	672.0270
4	62	3.0749	654.2507
Mean	58.75	2.9932	597.4156

TABLE V. N=30 NODE, MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	43	9.4763	1864.8463
2	46	9.2132	2125.5723
3	58	11.1824	2853.4950
4	66	10.4670	3173.9205
Mean	53.25	10.0847	2504.5

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	38	6.8942	1645.97059
2	39	7.0836	1602.925674
3	55	9.7450	2565.123519
4	67	10.7009	3282.702688
Mean	49.75	8.6059	2274.2

TABLE VI. N=50 NODE, MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	65	18.9230	7263.75013
2	57	15.4296	8258.040457
3	58	16.4500	7930.465633
Mean	60	16.9342	7817.4

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	62	17.5341	6722.4758
2	52	13.9871	6883.0913
3	51	13.8037	9282.2044
Mean	55	15.1083	7629.3

The algorithms both proposed and Dijkstra takes approximately 9 hours therefore taking only one result takes 18 hours. Because of that we are not able to execute that case multiple times. The results are given in below.

TABLE VII. N=100 NODE, MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	70	35.0313	39203.3087
2	58	31.18	31650.0148
3	63	33.21	37310.2287
Mean	63.6667	33.1404	36055

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	59	31.1327	30200.4111
2	50	25.6712	22938.1432
3	52	26.4312	24150.8727
Mean	53.6667	27.7450	25763

2) Effect of field size: We kept constant other parameters as given in below table

TABLE VIII. USED PARAMETERS

Battery threshold	60
Coverage threshold	80
Sensing range	20
Node number	20
Random Topology	

TABLE IX. 20*20 - MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	208	8.3433	6358.6395
2	186	7.7917	6276.748404
3	205	7.9191	8043.466177
Mean	199.66	8.0180	6893.0

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	197	7.7202	5958.1383
2	185	7.6105	6540.7118
3	201	7.6485	7290.8948
Mean	194.33	7.6597	6596.6

We repeated the experiments for 100*100 and 150*150, the results are same with the 200*200 due to the sensing range. Due to the 20 m fixed sensing range, increasing the field size does not changes the topology the number of nodes which is able to connect sink is too small. Therefore, we decided to keep size small by considering the sensing range.

TABLE X. 30*30 - MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	114	7.2465	3320.5635
2	133	8.6984	3902.6279
3	108	7.5360	3309.9136
Mean	118.33	7.8270	3511

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	102	6.7029	3020.6688
2	116	6.2960	3269.1791
3	101	7.0608	3029.5687
Mean	106.33	6.6866	3106.5

TABLE XI. 40*40 - MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	82	7.1258	2205.3421
2	92	7.2940	2684.4113
3	59	6.0877	1600.9100
Mean	77.6667	6.8358	2163.6

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	74	6.5895	1968.5861
2	74	6.3300	2218.414783
3	70	7.8084	2060.87845
Mean	72.6667	6.9093	2082.6

TABLE XII. 200*200 - MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	1	0	21.7895
2	1	0	20.9995
3	1	0	22.0343
Mean	1	0	21.6077

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	1	0	21.9486
2	1	0	21.7291
3	1	0	22.2203
Mean	1	0	21.9660

Due to the 20 m sensing range of the sensors, network become disconnected when field size is increased. Also, note that our algorithm works better than the other in both small and large field cases.

3) Effect of the coverage: We kept constant other parameters as given in below table

TABLE XIII. USED PARAMETERS

Battery threshold	60
Field	50*50
Sensing range	20
Node number	20
Random Topology	

TABLE XIV. COVERAGE:30-MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	94	9.3223	2198.0152
2	296	9.5517	4373.618698
3	121	8.1371	2700.973421
Mean	170.3333	9.0037	3090.9
(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	106	9.4513	2041.1887
2	277	9.0497	4194.184679
3	118	7.9497	2944.758630
Mean	167	8.8169	3060.0

TABLE XV. COVERAGE:60-MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	78	8.6002	1866.3548
2	62	7.2340	1570.1790
3	185	7.6943	3890.3346
Mean	108.33	7.8428	2442.3
(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	81	8.3476	1810.1711
2	64	6.4751	1498.5560
3	84	8.0901	2328.8219
Mean	76.33	7.6376	1879.2

TABLE XVI. COVERAGE:90-MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	36	4.5643	926.4164
2	72	6.6638	2475.4664
3	74	4.7235	2018.127885
Mean	60.66	5.3172	1806.7
(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	34	3.7988	841.2999
2	39	5.3156	1130.609305
3	63	6.0398	1607.081202
Mean	45.33	5.0514	1193.0

As coverage constraint is increased, the network closes to die early in both algorithms as expected. However, our algorithm works better than the other one in all cases.

4) Effect of the battery threshold: We kept constant other parameters as given in below table

TABLE XVII. USED PARAMETERS

Coverage threshold	80
Field	50*50
Sensing range	20
Node number	20
Random Topology	

TABLE XVIII. BATTERY:50-MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	52	5.9176	1240.8119
2	46	5.5614	1125.9844
3	55	7.0387	1663.594141
Mean	51	6.1726	1343.5
(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	39	5.0721	950.610326
2	33	4.2621	849.084441
3	56	6.2656	1704.38615
Mean	42.66	5.1999	1168.0

TABLE XIX. BATTERY:70-MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	71	7.5898	1752.3481
2	86	6.9645	2577.9937
3	46	5.4135	1174.5709
Mean	67.66	6.6559	1835.0
(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	51	5.6921	1211.6325
2	73	5.7377	2387.5625
3	33	3.9593	878.52598
Mean	52.33	5.1297	1492.6

TABLE XX. BATTERY:90-MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	34	5.4917	808.3373
2	52	6.6477	1391.152694
3	63	6.0398	1799.406752
Mean	49.66	6.0597	1333.0
(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	34	4.5553	767.0559
2	50	5.7016	1110.588368
3	63	6.0398	1607.081202
Mean	49	5.4322	1161.6

Note that we used battery threshold in our algorithm not for the Dijkstra. We have run the experiments on the same randomly created topology.

As clearly seen from the Fig. 13, the best value for the threshold is approximately 70. We will use that value in the second part pf the experiments.

5) Effect of the sensing range: We kept constant other parameters as given in below table

TABLE XXI. USED PARAMETERS

Battery threshold	60
Field	50*50
Coverage threshold	80
Node number	20
Random Topology	

TABLE XXII. RANGE:10-MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	1	0	23.9747
2	1	0.0068	23.3054
3	1	0.0002	23.1837
Mean	1	0.0023	0.4879

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	1	0	22.7136
2	1	0.0011	23.5966
3	1	0.0002	23.1774
Mean	1	0.001	23.1625

TABLE XXIII. RANGE:20-MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	58	6.7764	1497.1908
2	87	7.7635	2302.2417
3	80	6.4421	2031.4458
Mean	75	6.9940	1943.6194

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	56	5.9889	1406.5018
2	71	6.3538	1843.9738
3	70	5.9185	1730.7377
Mean	65.66	6.0871	1660.4044

TABLE XXIV. RANGE:30-MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	54	7.4978	1557.7051
2	84	8.0901	2649.6635
3	46	6.7790	1644.6207
Mean	61.33	7.4556	1950.7

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	48	6.6133	1340.2596
2	75	7.073	2179.2051
3	40	5.8545	1340.130968
Mean	54.33	6.5136	1619.9

TABLE XXV. RANGE:40-MY HEURISTIC(A)-COMPETITOR(B)

(A)	Num of rounds	Consumed energy[j]	Run time[sec]
1	44	6.7294	1421.3041
2	40	6.3158	1243.8216
3	51	6.5196	2224.0387
Mean	45	6.5216	1629.7

(B)	Num of rounds	Consumed energy[j]	Run time[sec]
1	45	6.7674	1357.1602
2	36	5.6181	1054.9935
3	42	5.6208	2164.6626
Mean	41	6.0021	1525.6

If sensing range of the sensor is increased, routing decisions will be effected because the number of one-hop neighbors for a sensor node increases [15].

For very small sensing ranges such as 10 m, it is likely that the network will disconnected because we have 50*50 field. Also, it is reasonable that lifetime of the network becomes smaller while sensing range is increasing.

Visualization of the results are given at the end of the paper (See Figure 12-16).

B. Statistical Results of Evaluation

In this part, we decided to keep following parameters as constant and show how our solution behaves while the number of nodes increasing. In other words, we used the best values which are found from the experimental results in the first part and we try to test is our solution is scalable or not.

TABLE XXVI. USED PARAMETERS

Battery threshold	70
Field	50*50
Coverage threshold	80
Sensing range	20
Random Topology	

We have measure additional metrics to show the quality of the algorithm and see the behavior of the algorithm. The metrics are given

- The number of messages received by sink
- Consumed energy per round
- Number of rounds first node to die
- Network lifetime which is number of rounds
- Total Consumed energy
- Run time of the algorithm

TABLE XXVII. N=20 NODE RESULTS

	Num of rounds	Consumed energy[j]	Run time[sec]	Num of received messages	First node die at round
1	58	7.3379	1544.454355	1129	45
2	59	6.5235	1498.070399	1166	54
3	88	6.6855	2001.606815	1407	38
4	63	6.5148	1446.134109	1072	37
5	45	5.8835	1100.674066	858	30
6	52	7.1164	1414.874028	1009	45
7	54	5.8001	1345.450634	903	15
8	72	8.4370	1875.9455	1417	57
9	55	7.0560	1377.184373	1089	24
10	46	5.5659	1059.2204	786	28
Mean	59.2	6.6921	1466.4	1083.6	37.3000
Std	164.6222	0.7280	87386	44711	177.7889

TABLE XXVIII. N=40 NODE RESULTS

	Num of rounds	Consumed energy[j]	Run time[sec]	Num of received messages	First node die at round
1	59	12.2128	4432.2283	2331	50
2	55	12.3697	4501.2810	1985	36
3	64	12.7039	5341.1296	2428	38
4	54	13.5817	4199.9467	2057	32
5	55	12.3697	4291.7471	1985	36
6	64	13.1681	5083.8737	2304	28
7	85	14.7498	7062.0624	3175	46
8	72	14.4748	6160.5437	2779	52
9	54	13.5817	4483.2750	2057	32
10	55	14.1225	4456.4077	2020	25
Mean	61.7000	13.3335	5001.2	2312.1	37.5000

TABLE XXIX. N=60 NODE RESULTS

	Num of rounds	Consumed energy[j]	Run time[sec]	Num of received messages	First node die at round
1	58	21.5709	13374.134473	3396	46
2	68	20.3497	17309.49963	4009	58
3	69	22.8187	17645.181414	4086	56
4	74	19.9178	20275.272604	4344	55
5	43	17.8110	10632.260225	2535	37
6	70	21.9605	18532.212697	4101	48
7	63	21.4542	15456.470318	3629	36
8	70	19.1725	21661.354799	4314	43
9	48	20.0075	10881.450122	2788	30
10	5	21.1255	10713.653448	3451	50
Mean	62.1000	20.6188	15648	3665.3	45.9000
Std	10.2572	1.4687	4083.5	626.5777	9.3743

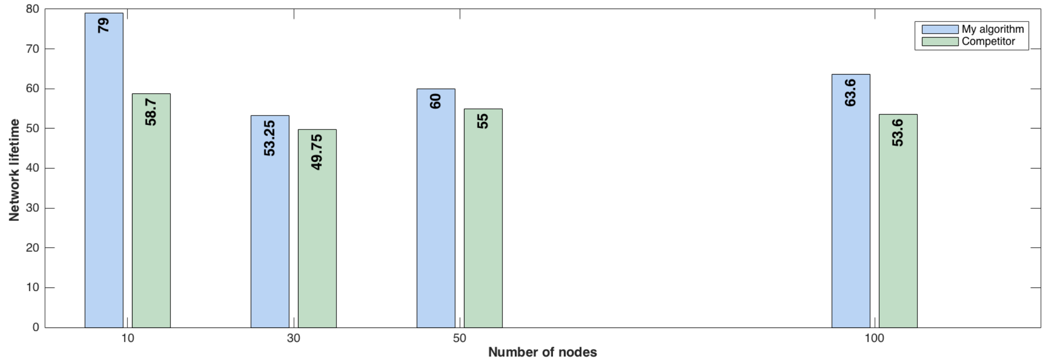

Fig. 12. Bar plot for different number of nodes case

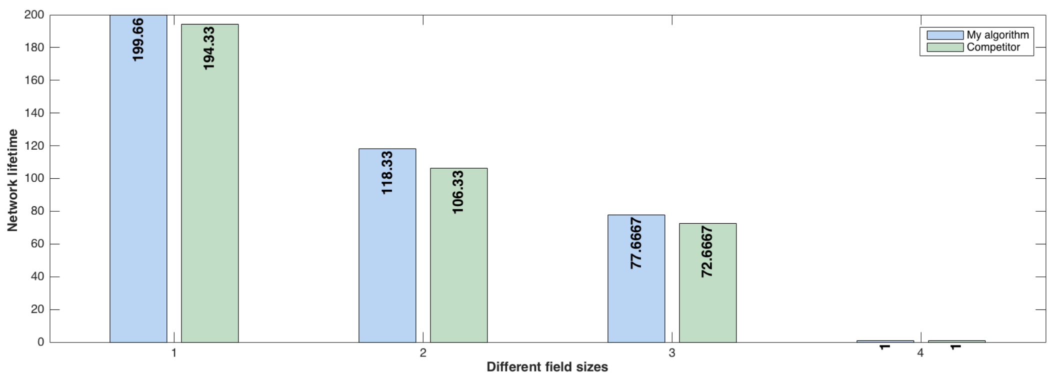

Fig. 13. Bar plot for different field sizes 20*20,30*30,40*40,200*200 respectively

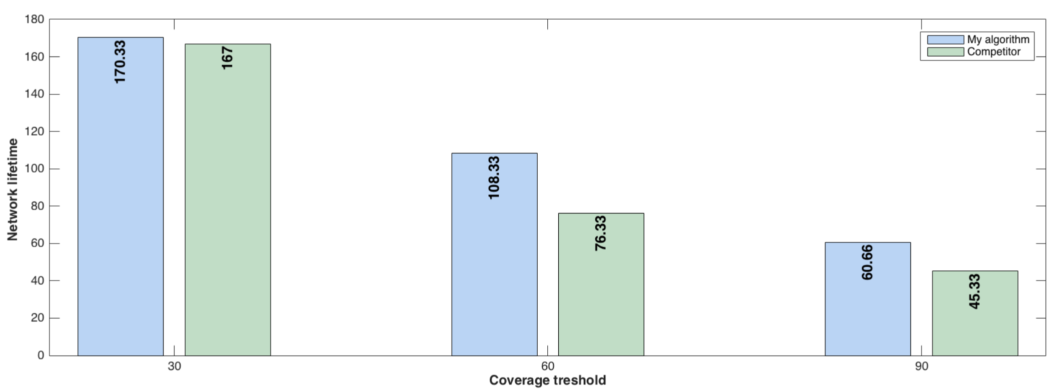

Fig. 14. Bar plot for different coverage conditions

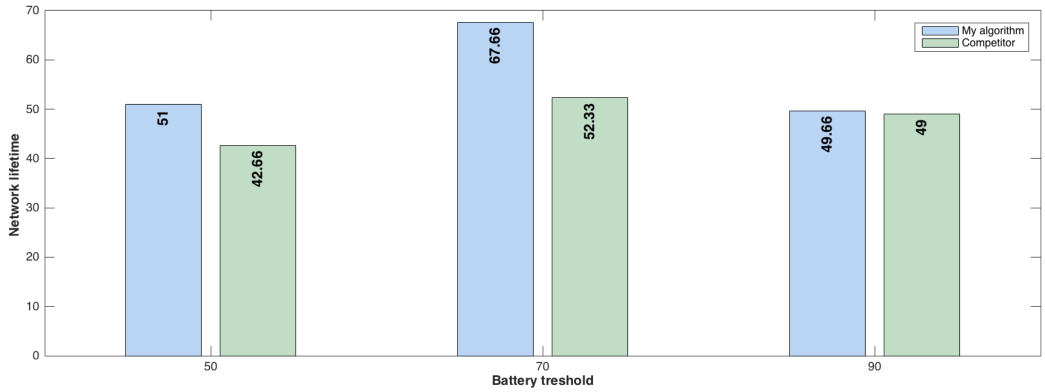

Fig. 15. Bar plot for different battery thresholds

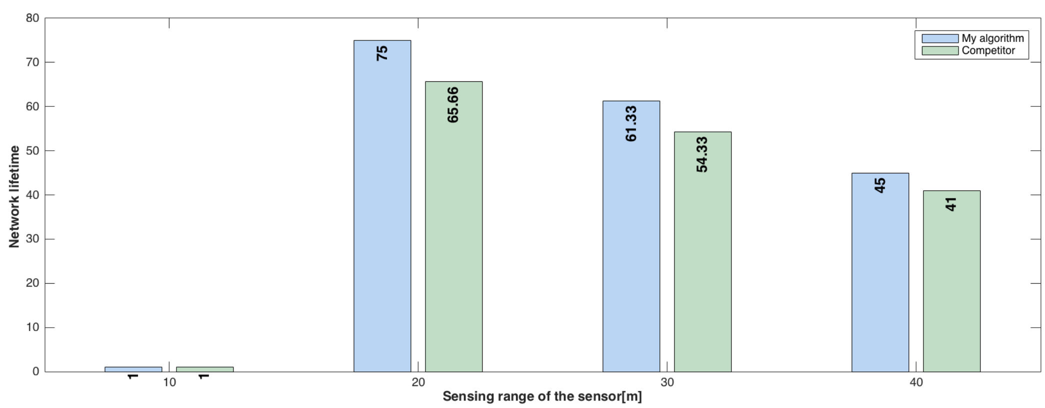

Fig. 16. Bar plot for different sensing ranges

VII. Conclusion and Future Works

To sum up, we have contributed the Dijkstra algorithm in two way. First, we define a new path metric which considers the energy minimization. Secondly, we use battery threshold mechanism to save the nodes close to sink. Our proposed solution is feasible due to the satisfied object function constraints. Experiments show that we reach our aim, the proposed algorithm works efficiently. As a future work, the algorithm may be adopted multiple sinks scenario. In that case, the lifetime will increase due to the decreasing distance between sensor and sink.

Note that you can access the code, code manual and final presentation using https://github.com/sertbasn1/WSNProject.git.

References

[1] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad hoc networks*, vol. 3, no. 3, pp. 325–349, 2005.

[2] E. Real and E. Romero, "Pfc: Design and implementation of a dynamic wireless sensor network." [Online]. Available: http://elb105.com/pfc-design-and-implementation-of-a-dynamic-wireless-sensor-network/

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.

[4] D. Goyal and M. R. Tripathy, "Routing protocols in wireless sensor networks: a survey," in *2012 Second International Conference on Advanced Computing & Communication Technologies*. IEEE, 2012, pp. 474–480.

[5] A. Sankar and Z. Liu, "Maximum lifetime routing in wireless ad-hoc networks," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 2. IEEE, 2004, pp. 1089–1097.

[6] M. Patel, S. Venkateson, and R. Chandrasekaran, "Energy-efficient capacity-constrained routing in wireless sensor networks," *International Journal of Pervasive Computing and Communications*, vol. 2, no. 2, pp. 69–80, 2007.

[7] X. Yao and X. Zheng, "A secure routing scheme based on multi-objective optimization in wireless sensor networks," in *Computational Intelligence and Security, 2008. CIS'08. International Conference on*, vol. 1. IEEE, 2008, pp. 436–441.

[8] C. Hua and T.-S. P. Yum, "Optimal routing and data aggregation for maximizing lifetime of wireless sensor networks," *IEEE/ACM Transactions on networking*, vol. 16, no. 4, pp. 892–903, 2008.

[9] K. Akkaya and M. Younis, "An energy-aware qos routing protocol for wireless sensor networks," in *Distributed Computing Systems Workshops, 2003. Proceedings. 23rd International Conference on*. IEEE, 2003, pp. 710–715.

[10] A. Manjeshwar and D. P. Agrawal, "Teen: Arouting protocol for enhanced efficiency in wireless sensor networks."

[11] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on*. IEEE, 2000, pp. 10–pp.

[12] "A wireless sensor network scheme for power optimization," *Ritu et al, Int.J.Computer Technology and Applications*, vol. 5, no. 4, p. 14341437. [Online]. Available: http://www.ijcta.com/documents/volumes/vol5issue4/ijcta2014050413.pdf

[13] A. Gagarin, S. Hussain, and L. T. Yang, "Distributed search for balanced energy consumption spanning trees in wireless sensor networks," in *Advanced Information Networking and Applications Workshops, 2009. WAINA'09. International Conference on*. IEEE, 2009, pp. 1037–1042.

[14] "dijkstras algorithm continued." [Online]. Available: https://courses.cs.washington.edu/courses/cse326/06au/lectures/lect22.pdf

[15] M. A. Youssef, M. F. Younis, and K. A. Arisha, "A constrained shortest-path energy-aware routing algorithm for wireless sensor networks," in *Wireless Communications and Networking Conference, 2002. WCNC2002. 2002 IEEE*, vol. 2. IEEE, 2002, pp. 794–799.