INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 2)
Available online at: www.ijariit.com

Maximize lifetime of Wireless Sensor Network by optimal sink deployment and sensor to sink routing

Manpreet Singh
manpreet1639@gmail.com
Adesh College of Engineering and Technology, Faridkot, Punjab

ABSTRACT

Wireless sensor networks typically contain hundreds of sensors. The sensors collect data and relay it to sinks through single hop or multiple hop paths. Sink deployment significantly influences the performance of a network. Since the energy capacity of each sensor is limited, optimizing sink deployment and sensor-to-sink routing is crucial. In this paper, this problem is modeled as a mixed integer optimization problem. Then, a novel layer-based diffusion particle swarm optimization method is proposed to solve this large-scale optimization problem. In particular, two sensor-to-sink binding algorithms are combined as inner layer optimization to evaluate the fitness values of the solutions. Particle swarm optimization constitutes currently one of the most important routing algorithms. Its popularity has stimulated the emergence of various variants of swarm-inspired techniques, based in part on the concept of pairwise communication of numerous swarm members solving the optimization problem in hand. PSO is a population-based optimization method. PSO consists of a swarm of particles which move towards an optimal solution of the problem. Particle uses their position and velocities (P Best & G Best) to find an optimal route. This paper show better result of Maximizing lifetime of wireless sensor networking by using PSO routing Fully Informed Particle Swarm Optimization (FIPSO) and Diffusion PSO Performance of those algorithms is also evaluated over a set of benchmark instances Compared to existing methods that the sinks are selected from candidate positions, our method can achieve better performance since they can be placed freely within a geometrical plane. Several numerical examples are used to validate and demonstrate the performance of our method. The reported numerical results show that our method is superior to those existing. Furthermore, our method has good scalability which can be used to deploy a large-scale sensor network.

Keywords— *Wireless Sensor Network, Sink placement, Particle Swarm Optimization (FIPSO and Diffusion PSO), Lifetime, Energy*

1. INTRODUCTION

Wireless Sensor Network (WSN) has been widely used in many real-world applications over the last decades. It comes under Wireless Adhoc Networks. With the advancement of both electronic and wireless communication technologies, the scale of a WSN has become larger and larger. In general, a WSN contains a large number of low cost and energy constrained sensor devices in a given area for monitoring the environmental conditions such as temperature, humidity, pressure and seismic condition. WSNs play a very important role in the internet of Things (IoT).

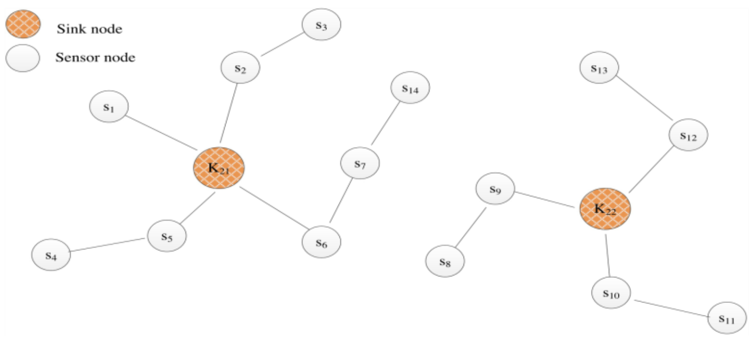

Fig. 1: Wireless Sensor Network

In a WSN, sensors collect data or act as traffic relays for transferring data from sensors to sinks through one hop or multiple hop routing. The energy capacity of each sensor is limited. Once one of the sensors in the network is out of power, the WSN cannot work as designed. Thus, it is important to design an energy efficient WSN so that lifetime of a network is maximized

To design an energy efficient WSN, an effective method is to deploy multiple sinks instead of a single sink. An example of the WSN with the multiple sinks is shown in Fig. 1. During the design of a network with multiple sinks, we need to determine the number of sinks, and more important than that, we need to determine the optimum location of sinks. This problem is referred to as sink deployment. Sink deployment is often formulated as an optimization problem. Traditional sink deployment assumes that only a single sink is deployed in a network. As a result, the sensors within a single hop from a sink will be exhausted and the performance of this network will be decreased significantly which is known as the energy hole phenomenon. This is because most of the traffic from other sensors will be routed through that sink. This problem becomes worse if the scale of a network becomes larger. To overcome this drawback, multiple sinks are deployed in order to prolong lifetime and reduce load balance and delay of a network
In a WSN with multiple sinks, there are two ways to prolong the lifetime of this network: optimizing the locations of sinks and selecting the best routes from sensors to sinks. Traditionally, the best locations of sinks are determined through clustering methods where the routes from sensors to sinks are assumed to be given. Moreover, the best locations of sinks are assumed to be selected from candidate positions. But the routes from sensors to sinks are not considered. To supplement this drawback, a genetic algorithm is proposed in to bind sensors to sinks. However, there is still a lack of a method to optimize locations of sinks and routes from sensors to sinks simultaneously. Intuitively, we can combine the binding strategy in and the clustering method to optimize locations of sinks and routes from sensors to sinks alternatively. However, the alternating method is with high complexity and might not lead to a truly optimal solution. How to optimize locations of sinks and routes from sensors to sinks simultaneously is clearly important but challenging.

In this paper, we will study this problem and develop an efficient method to solve it. More specifically, through analysing the original problem and customizing the algorithm, we decompose the original problem into two subproblems. Then, a layer- based algorithm includes the inner optimization algorithm to bind sensor to sink and the outer optimization algorithm to optimize sink deployment simultaneously.

1.1 Problem Formulation

Consider a WSN as an undirected graph G (V, E), where V as a set of nodes and E as the set of edges in the network. The set of sensors is denoted as $V_s = \{1, 2\ldots N\}$ and set of sinks is denoted as $V_{si} = \{N+1, N+2\ldots N+M\}$, where N is the total number of sensors and M is the total number of sinks. The position of the sensors and the sinks are denoted as $\alpha_1, \alpha_2, \alpha_3 ... \alpha_N \ldots \alpha_{N+M}$. It is assumed that sensors have already been deployed and their positions are known and positions of sinks are required to be determined. The position of the sensors and sinks are represented as $\alpha_s = [\alpha_1, \alpha_2\ \alpha_3 \ldots \alpha_N]$ and $\alpha_{si} = [\alpha_{N+1}, \alpha_{N+2}, \alpha_{N+3} \ldots\ \alpha_{N+M}]$. After determining the optimal position of the sinks, the cost of the network is calculated. This cost includes hardware and energy cost of sensors and sink. Now, the problem is formulated as:

$$\text{P1: Optimal position } (\alpha_{si}) \tag{1}$$

$$\text{P2: Cost}_{min} \Rightarrow (\text{Lifetime of the network})_{max} \tag{2}$$

2. METHODOLOGY

2.1 Topology

All sensors are assumed to be fixed and their positions are known. The sensors are distributed in a square with the size equal to 600m x 600m. The number of sensors already deployed is 200. A number of sinks that has to be deployed are 5. The minimum distance between nodes is 20m. The maximum transmission range of each node is 60m. All sensors have the same data generation rate and initial energy 1 J. The sensors transmit data to the sink node in a single hop or multihop paths.

2.2 Sink Deployment

To find the optimal location of the sink, PSO (particle swarm optimization) is used. PSO is a population-based optimization method. PSO consists of a swarm of particles which move towards an optimal solution of the problem. Particle uses their position and velocities to represent the movements. The position of the particles is updated iteratively. The particle with their and other member's position finds the optimal location in the search space.

PSO algorithm:

PSO consists of a predefined number of particles say Np, called Swarm. A particle P_i, $1 \leq i \leq Np$ has position $X_{i,d}$ and velocity $V_{i,d}$, $1 \leq d \leq D$ in the d^{th} dimension of the search space. The dimension D is the same for all particles. The search space is the set of all possible solutions. A fitness function for each particle is used to evaluate the optimum solution. In the initialization phase, each particle is assigned with a random position and velocity to move in the search space. During each iteration, each particle finds it's own best i.e. personal best called $PBest_i$ and the global best called Gbest. To reach the global best solution, it uses its personal and global best to update the velocity $V_{i,d}$ and position $X_{i,d}$ using the following equations:

$$V_{i,d}(t+1) = w^*V_{i,d}(t) + c_1{}^*r_1(P_{i,d}(t)-X_{i,d}(t)) + c_2{}^*r_2(G(t)-X_{i,d}))$$

$$X_{i,d}\ (t+1) = X_{i,d}(t) + V_{i,d}(t+1)$$

where $0 < \omega < 1$ is the inertia weight, c1, c2, $0 \leq c1, c2 \leq 2$ are the acceleration coefficients and, r_1,r_2, $0 < r_1,r_2 < 1$ are the randomly generated values. This updation of velocity and position is updated in all iterations till maximum iterations reach. After getting a newly updated position, the particle evaluates the fitness function and updates Pbesti as well as Gbest.

Calculation of Fitness function:

(a) Euclidian Distance: Sensor consumes some energy to send data to the sinks. To maximize the lifetime of the network, we need to reduce the distance between sensors and sinks.

$$F_1 = \sum_{i=1}^{N} \min_{k=N+1}^{N+M} dis(S_i,S_k)$$

(b) Hop Count: During the communication between sensor and sink, the sensor uses multihop routes to send the data. To decrease the delay we need to minimize the number of hop count.

$$F_2 = \sum_{i=1}^{N} \min_{k=N+1}^{N+M} \text{Hop count}(Si, Sk)$$

$$\text{Minimize fitness} = \alpha * F_1 + \beta * F_2$$

2.3 Analysis of energy consumption and cost

2.3.1 Energy Model: Each sensor in a network not only sense data but also relays data. Sensors sense the data and send it to a sink via a single hop or multiple hop paths. Suppose that each sensor node ($i \varepsilon V_s$) has energy consumption for generating data, transmitting data, receiving data and for other network operations e.g routing, idle listening. The energy consumed by the i^{th} sensors per unit time is:

$$E_i = (e_{tx}+e_{rx}+e_{se})L+e_{other} \tag{3}$$

Where e_{tx} is the energy consumption of transmitting unit data for the sensor, e_{rx} is the energy consumption of receiving unit data for the sensor, e_{se} is the energy consumption sensing unit data for sensor and e_{other} is the energy consumption of other network operations for a sensor, L is the data size generated per unit time by the sensor. Next, the energy consumption of all the sensors per unit time is:

$$E_s = \sum_{i=1}^{N} E_i$$

$$E_s = \sum_{i=1}^{N} (e_{tx}+e_{rx}+e_{se})L+e_{other} \tag{4}$$

All the data are transmitted to sink. The energy consumption of sink includes receiving, transmitting, aggregating data and others. Data size received per unit time by the sink is:

$$r = L \sum_{i=1}^{N} i = L (N^2+N)/2 \tag{5}$$

Sink aggregates all the data. The data size of per unit time sent by the sink is

$$s = mr + c$$

m, is the compression ratio and c is constant.

$$s = mL(N^2+N)/2 + c$$

Multiple sinks are deployed in the network. So, The energy consumption of k^{th} where ($k \varepsilon V_{si}$) sink per unit time is:

$$E_k = e^*_{tx}s + e^*_{rx}r + e^*_{agg}r + e^*_{other} \tag{6}$$

$$E_k = e^*_{tx}(mL(N^2+N)/2) + e^*_{rx}(L(N^2+N)/2) + e^*_{agg}(L(N^2+N)/2) + e^*_{other}$$

So, the energy consumption of all the sinks is:

$$E_{si} = \sum_{k=N+1}^{N+M} E_k$$

$$E_{si} = \sum_{k=N+1}^{N+M} (e^*_{tx}(mL(N2+N)/2) + e^*_{rx}(L(N2+N)/2) + e^*_{agg}(L(N2+N)/2) + e^*_{other}) \tag{7}$$

The energy consumption of network per unit time is:

$$E = E_s + E_{si}$$

$$E = [\sum_{i=1}^{N} (e_{tx}+e_{rx}+e_{se})L+e_{other}] + [\sum_{k=N+1}^{N+M} (e^*_{tx}(mL(N2+N)/2) + e^*_{rx}(L(N2+N)/2) + e^*_{agg}(L(N2+N)/2) + e^*_{other})] \tag{8}$$

2.3.2 Cost Model: The cost of the network includes hardware cost of sensors and sinks and cost of energy. The number of sensors is:

$$n = \sum_{i=1}^{N} i = (N^2+N)/2$$

The number of sinks:

$$n^* = \sum_{k=N+1}^{N+M} k = ½ M(M+2N+1)$$

Hardware cost (C_1) = an +a*n*
a = cost of one sensor
a* = cost of one sink

The cost of all energy
C_2 = bET
b = cost per unit energy
E = energy consumption of the network
T = lifetime of all the nodes

$$\text{The total cost of the network } (C) = C_1 + C_2$$
$$= an + a^*n^* + bET$$
$$= a(N^2+N)/2 + a^*((1/2M(M+2N+1)) + bET$$

Therefore cost of the network is dependent on the energy consumption of the sensors and sinks.

Algorithm:

1. Find initial sinks using Fuzzy C-means algorithm (using inbuilt MATLAB function '*fcm*')
2. Sink placement optimization using PSO:
   (a) Initialize the parameters: Number of particles(nps), maximum iteration(IterMax), boundary limits of particles(bmin, bmax), velocity limits(vmin,vmax)
   (b) Generate Initial population and cost using the following formula:

for m=1 to population size do,

pop(m)=Sink+[bmin+(bmax-bmin)*rand(Ns,2)]

vel(m)= vmin+(vmax-vmin)*rand(Ns,2)

$$\text{Cost (m)}= \sum_{i=1}^{N} \sum_{k=N+1}^{N+Ns} \text{Min Dist(Si, Sk)}$$

end for

Find minimum cost and the index of the swarm, [Min,Idx]=min(Cost);

for i=1 to IterMax

- Update all the particles using formula:

for m=1 to population size do

vel(m)=w*vel(m)+c1*rand*(pbest(m)-pop(m))+c2*rand*(gbest-pop(m)); pop(m)=pop(m)+vel(m);

$$\text{NewCost (m)}= \sum_{i=1}^{N} \sum_{k=N+1}^{N+Ns} \text{Min Dist(Si, Sk)}$$

end for

- Again find the minimum of the new cost, [NewMin,NewIdx]=min(NewCost);

If NewMin<Min

gbest=pop(Newidx);

Min=NewMin;

end if

for m=1 to population size do

if NewCost(m)<Cost(m)

pbest(m)=pop(m);

end if

end for

end for

- Optimized Sink=gbest.

Routing PSO:

Input:

The coordinates of senors and sinks;

The maximal number of iteration *maxgen*.

Output:

Routes from the sensor to sink, network lifetime T (G).

a) Create a topology structure according to the coordinations of sensors and sinks;
b) for i=1 to population size do
c) Generate initial particles p_i
d) Evaluate each particle p_i
e) pbesti ← p_i
f) if fitness(gbest) < fitness(p_i) then
g) gbest ← p_i
h) end if
i) end for
j) t ← 1;
k) while t<maxgen do
l) for i=1 to population size do
m) Updating new particle p_i according to formula (4.6)
n) Evaluate the fitness of particle p_i
o) if fitness(p_i)>fitness($pbest_i$) then
p) pbesti ← p_i
q) end if
r) if fitness(p_i)>fitness(gbest) then
s) gbest ← p_i
t) end if
u) end for
v) end while
w) return *gbest*;

Route=*gbest*;

2.4 Energy Calculation:

$$E= [\sum_{i=1}^{N}.(e_{tx}+e_{rx}+e_{se})L+e_{other}] + [\sum_{k=N+1}^{N+M}.LR\ [e^*_{tx}\ (m+c)+e^*_{rx}+ e^*_{agg}] + e^*_{other}]$$

2.5 Cost calculation

$$C= an + a^*n^* + bET$$

Here we little modify PSO using FIPSO for the better result by adding its Equation as Constriction factor . Fully Informed Particle Swarm Optimization constitutes one of the heuristic algorithms derived from the basic PSO paradigm. The idea of using information from a group of particle's Km neighbors, rather than just the best one – as in traditional canonical PSO – was first proposed by Suganthan in 1999 [13]. It was also included in the complete Fully Informed PSO procedure suggested by Mendes, Kennedy and Neves. In each iteration of the algorithm particle's position xm(k) is updated by moving it iteratively along vector vm(k) with the coordinates n = 1,..., N adjusted as follows:

$$v_{mn}(k+1) = X\left[v_{mn}(k) + \frac{1}{K_m}\right]\sum_{j=1}^{K_m} U(0,\varphi)\,(x_{N_m(j)m}(k)^* - x_{mn}(k))$$

3. EXPERIMENTAL ANALYSIS

In table 1, the value of the fitness function is given with 20 sensors and 3 sinks using the Diffusion PSO+BS and Diffusion PSO+PSO algorithm. Open the sub-folder named 20 nodes and 3 sinks in the main folder named New Codes. Now run the main_PSO.m code to obtain the fitness value. The value of the fitness obtained from the above code is improved.

The new parameters or modifications that we do in the code is-

(a) Number of population=30.
(b) Maximum number of iterations=50.
(c) Maximum and minimum value of inertia=0.9 and 0.2.
(d) The code has been run for the 20 times and the following values are noted-
(e) Maximum value of fitness.
(f) Minimum value of fitness.
(g) Average value of fitness.
(h) Standard deviation.
(i) Time for calculating the fitness.

The table 2.m code shows the results according to the modified parameters, which are better from the table2 in the base paper. The time is also shown below in the command window.

In table 2, the value of the fitness function is given with 200 sensors and 7 sinks using the K-means, GA, PSO, Diffusion PSO+BS and Diffusion PSO+PSO algorithm. Open the sub-folder named 200 nodes and 7 sinks in the main folder named New Codes. Now run the main_PSO.m code to obtain the fitness value. The value of the fitness obtained from the above code is improved.

The new parameters or modifications that we do in the code is:

(a) Number of population=30.
(b) Maximum number of iterations=50.
(c) Maximum and minimum value of inertia=0.9 and 0.2;
(d) The code has been run for the 20 times and the following values are noted-
(e) Maximum value of fitness.
(f) Minimum value of fitness.
(g) Average value of fitness.
(h) Standard deviation.
(i) Time for calculating the fitness.

The table 3.m code shows the results according to the modified parameters, which are better from the table3 in the base paper. The time is also shown below in the command window.

In table 3, the value of the fitness function is given with 300 sensors and 5 sinks using the K-means, GA, PSO, Diffusion PSO+BS and Diffusion PSO+PSO algorithm. Open the sub-folder named 300 nodes and 5 sinks in the main folder named New Codes. Now run the main_PSO.m code to obtain the fitness value. The value of the fitness obtained from the above code is improved.

The new parameters or modifications that we do in the code is:

(a) Number of population=30.
(b) Maximum number of iterations=50.
(c) Maximum and minimum value of inertia=0.9 and 0.2;
(d) The code has been run for the 20 times and the following values are noted-
(e) Maximum value of fitness.
(f) Minimum value of fitness.
(g) Average value of fitness.
(h) Standard deviation.
(i) Time for calculating the fitness.

The table4.m code shows the results according to the modified parameters, which are better from the table4 in the base paper. The time is also shown below in the command window.

Table 1: Experimental analysis

Algorithm	Min	Avg	Max	St_Dev	Time
'Diffusion PSO+BS'	535.67	557.96	591.79	16.56	35.1
'Diffusion PSO+FIPSO'	616.11	623.06	632.14	10.14	150.15

Table 2: Experimental analysis

Algorithm	Min	Avg	Max	St_Dev	Time
'K-means'	61.66	95.57	112.5	16.2	12.5
'GA'	123.6	170.16	214.54	20.79	59.8
'PSO'	85.76	129.09	181	20.56	63.1
'Diffusion PSO+BS'	105.67	145.96	191.79	14.87	127.3
'Diffusion PSO+FIPSO'	142.25	193.72	229.12	15.11	369.72

Table 3: Experimental analysis

Algorithm	Min	Avg	Max	St_Dev	Time
'K-means'	40.61	60.58	76.9	9.65	15.5
'GA'	79.19	98.81	114.59	11.5	172.9
'PSO'	60.19	80.38	97.69	12.21	136.5
'Diffusion PSO+BS'	89.67	2.23	121.14	9.26	221.5
'Diffusion PSO+FIPSO'	106.11	131.32	147.71	10.15	619.21

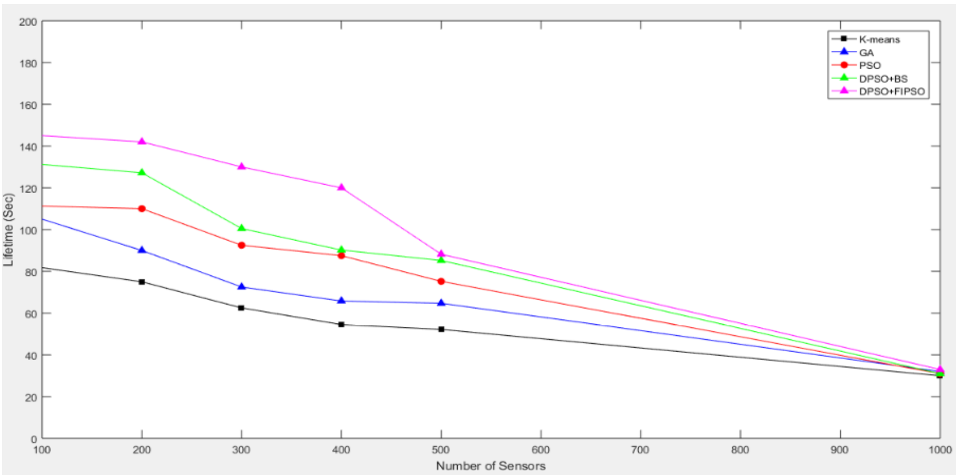

Fig. 2: Graph of number of sensors and their lifetime

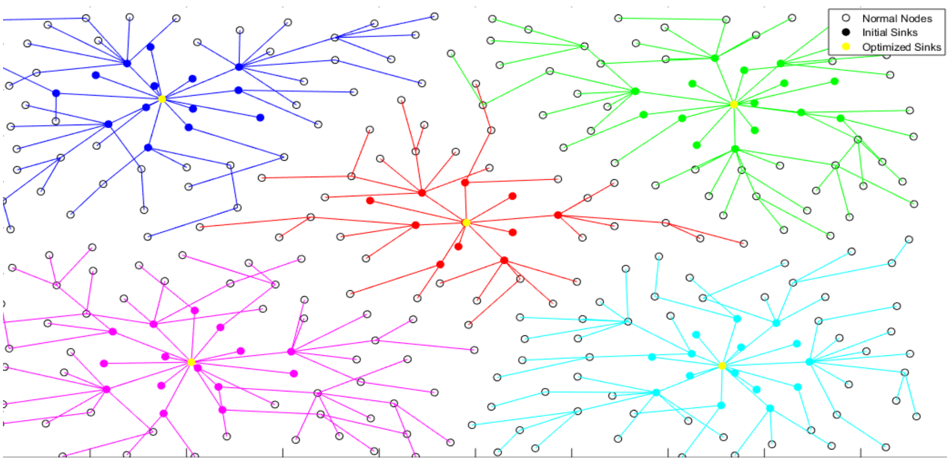

Fig. 3: Optimum Sink placement by PSO Routing (FIPSO+ Diffusion PSO)

4. CONCLUSION

In this paper, we studied the optimal deployment of sinks and sensor to sink routing to prolong life time of a network. Different from existing works, the sinks are considered to be deployed freely in a plane and sensors are bound to sinks simultaneously. Traditionally, this problem is solved based on clustering-based algorithms. Here we formulated it as a mixed integer optimization problem through integrating sink placement and sensor to sink routing. For a large scale network, the problem contains a large number of variables which is out of capability of any existing optimization method to solve it. To tackle this challenge, we proposed a layer-based optimization approach for sink deployment and sensor-to-sink routing. Several experiments were conducted to evaluate the performance of our proposed approaches. From experimental results obtained, we can conclude that our proposed approaches outperform the existing algorithms. The life time of the network is significantly improved through joint optimization of sink placement and sensor-to-sink binding. This work can be easily extended to deploy gateways or relays in a heterogeneous WSN. Moreover, our approaches are in polynomial complexity in terms of the scale of the network. Thus, they can be applied to large scale network planning problems.

5. REFERENCES

[1] J. Chai , C. Wu , C. Zhao , H.-L. Chi , X. Wang , B.W.-K. Ling , K.L. Teo , Reference tag supported RFID tracking using robust support vector regression and Kalman filter, Adv. Eng. Inform. 32 (2017) 1–10 .
[2] Chuanxin Zhao, kwang Hyo Jung, Maximizing lifetime of wireless sensor network via joint optimizing sink placement
[3] Fully Informed Swarm Optimization Algorithms: Basic Concepts, Variants and Experimental Evaluation
[4] Szymon Łukasik1,21 Department of Automatic Control and IT Cracow University of Technology ul. Warszawska 24, 31-155 Krakow, Poland Email: {szymonl,pkowal}@pk.edu.plPiotr
[5] D. Ganesan , R. Cristescu , B. Beferull-Lozano , Power-efficient sensor placement and transmission structure for data gathering under distortion con- straints, ACM Trans. Sens. Netw. 2 (2006) 155–181 .
[6] E. Oyman , C. Ersoy , Multiple sink network design problem in large scale wireless sensor networks, in Proceedings of the 2004 IEEE International Conference on Communications (ICC 2004), IEEE Computer Society Press, 2004, pp. 3663–3667 .
[7] H. Safa , W. EI-Hajj , H. Zoubian , A robust topology control solution for the sink placement problem in wsns, J. Netw. Comput. Appl. 39 (2014a) 70–82 .
[8] H. Safa , M. Moussa , H. Artail , An energy efficient genetic algorithm based approach for sensor-to-sink binding in multi-sink wireless sensor networks, Wirel. Netw. 20 (2014b) 177–196 .
[9] C. Wu , K.L. Teo , V. Rehbock , H.H. Dam , Global optimum design of uniform fir filter bank with magnitude constraints, IEEE Trans. Signal Process. 56 (2008) 5478–5486 .
[10] F. Guerriero , A. Violi , E. Natalizio , V. Loscri , C. Costanzo , Modelling and solving optimal placement problems in wireless sensor networks, Ad Hoc Netw. 35 (2011) 230–241 .
[11] S. Sengupta , S. Das , M. Nasir , B. Panigrahi , Multi-objective node deployment in wsns: in search of an optimal trade-off among coverage lifetime, energy consumption and connectivity, Eng. Appl. Artif. Intell. 26 (2013) 405–416.
[12] L. Rossi , B. Krishnamachari , C.-C. Kuo , Optimal sink deployment for distributed sensing of spatially nonstationary phenomena, in Proceedings of Global Telecommunications Conference (GLOBECOM '07), 2007, pp. 1124–1128.
[13] E.M. Arkin , A. Efrat , J. Mitchell , V. Polishchukc , S. Ramasubramaniand , S. Sankararamanb, J. Taherib , Data transmission and base-station placement for optimizing the lifetime of wireless sensor networks, Ad Hoc Netw. 12 (2014) 201–218.
[14] C. Wu , C. Li , Q. Long , A dc pogramming approach for sensor network localization with uncertainties in anchor positions, J. Ind. Manag. Optim. 10 (2014) 817–826 .
[15] Z. Vincze , R. Vida , A. Vidacs , Deploying multiple sinks in multi-hop wireless sensor networks, in Proceedings of IEEE International Conference on Pervasive Services, IEEE Computer Society Press, 2007, pp. 55–63.
[16] J. Huang , Y. Meng , X. Gong , Y. Liu , Q. Duan , A novel deployment scheme for green internet of things, IEEE Internet Things J. 1 (2014) 196–205.

