Sensors & Transducers, Vol. 163 , Issue 1, January 2014, pp. 247-254

Sensors & Transducers

A Delay-Sensitive Connected Target Coverage Algorithm in Wireless Sensor Networks

[1, 2] Junbin Liang, [2] Ming Liu, [2*] Xiaoyan Kui
[1] School of Computer and Electronic Information, Guangxi University,
Guangxi, Nanning, 530004, China
[2] School of Information Science and Engineering, Central South University,
Hunan, Changsha, 410083, China
E-mail: xykui@csu.edu.cn

Received: 8 January 2014 /Accepted: 29 January 2014 /Published: 31 January 2014

Abstract: The issue of guaranteeing the network QoS (target coverage, network connectivity, etc.) to maximize the lifetime in wireless sensor networks (WSNs) has been widely studied in recent years. In some delay-sensitive sensor networks (fires, gas leaks, explosions, etc.), sensor nodes must transmit their data to sink within a limited period to monitor the critical physical environment. In order to study connected target coverage in such delay-sensitive sensor networks, we are the first one to propose the Delay-Constraint Connected Target Coverage (DCCTC) problem and study the following works specifically: 1) we model DCCTC problem as a Height Limited Maximum Cover Tree (HLMCT) problem, and then give an upper bound on the network lifetime for HLMCT problem; 2) we develop a fast heuristic algorithm, named HLCWGC; 3) we study the performance of HLCWGC algorithm by comparing it with other existing algorithms improved to solve HLMCT problem. Simulation results show that HLCWGC algorithm can achieve a better performance than other improved algorithms in the delay-sensitive sensor networks. *Copyright © 2014 IFSA Publishing, S. L.*

Key words: Connected target coverage, Lifetime maximization, Delay constraint.

1. Introduction

Wireless sensor networks (WSNs) are composed of a large number of low-energy and low-cost nodes that communicate with each other through single-hop or multi-hop wireless links [1, 2]. WSNs can be applied to many areas, such as battlefield surveillance, biological detection, home appliance, smart space. A basic and important function of WSNs is to monitor areas or targets for a long period, and the data can be effectively transmitted to the sink, which is called connected target coverage problem. Sensors are often deployed in remote or inaccessible environments where replenishing the energy of sensors is usually impossible.

A critical issue in some delay-sensitive applications (fires, gas leaks, explosions, etc.) is conserving the energy of sensors and transmits their data to sink within a limited period, which guarantees the network can last for a long periods [15-18]. Besides target coverage and network connectivity, we should take data delay into consideration in delay-sensitive applications. Many researches have focused on the connected target coverage problem, but they always ignored the data delay in sensor network.

Coverage problem in WSNs has been intensively investigated. Based on different requirements of network connectivity, existing related works on coverage problem can be divided into three categories: 1) coverage problem which does not meet the requirements of network connectivity.

2) 1-connected target coverage problem. 3) k-connected target coverage problem. We concisely discuss them in the following.

1) Coverage problem which does not meet the requirements of network connectivity only considers covering all the target nodes [3-6], while the sensor data can be sent to sink through other methods (such as hierarchical network). Sljepcevic [3] proposed a new algorithm which guarantees all the network area can be covered. M. Cardei et al. modeled the discrete target coverage problem into a NP complete problem, which is also a disjoint set covering problem [4]. M.Caidei et al. also extend the schema that they proposed in [4] and raised an intersection set covering problem [5], for example, one node can be covered by many coverage sets at the same time. M. Caidei [6] further expanded their work, they assumed that each node has multiple sensing ranges to cover the target nodes.

2) 1-connected target coverage problem requires that all target nodes can be covered and guarantees at least one path existing from the sensor nodes to the sink. Lu [7] considered an energy model of nodes with multiple sensing ranges. At first, it constructed a virtual skeleton, ensuring that all nodes in the virtual skeleton or a node can be directly connected with the nodes in the virtual skeleton, and then considered the target coverage. However, the energy model in [7] only considered sensing energy, and it was not necessary to build such global connected skeleton. Q. Zhao [8] proposed an energy model considering the energy consumption in data receiving and data transmission, but it did not consider how to send data to the sink, the schema only realized based on a single routing mechanism. Q. Zhao et al. [9] modeled connected target coverage (CTC) problem as a maximum cover tree (MCT) problem, proved that MCT problem is a NP-complete problem, and then gave an upper bound for MCT problem. Zhao then presented an approximate algorithm App_MCT and a greedy algorithm CWGC to solve the MCT problem. The simulation results verified the validity of App_MCT algorithm and CWGC algorithm.

3) k-connected target coverage problem requires that all target nodes can be covered and guarantees more than one paths existing from the sensor nodes to the sink. Li [10] proved that k-connected target coverage problem is NP hard, and then proposed two heuristic algorithms to select the minimum nodes to work, which satisfied the k connectivity and target coverage. [11] presents a target coverage schema with QoS requirements, which can achieve a good performance in wireless sensor networks.

However, these above connected target coverage algorithms were designed for the requirements of network coverage or network connectivity, however, they were not suitable for delay-sensitive networks, because the network delay caused by those algorithms were too larger to be accepted by the practical application.

The contributions of this paper are:

1) Aiming at the application in delay-sensitive network, we first propose the delay-constraint connected target coverage (DCCTC) problem and model DCCTC problem as a height limited maximum cover tree (HLMCT) problem; we also prove that HLMCT is a NP-complete problem, and then give an upper bound on the network lifetime for HLMCT problem; and then, we develop a fast heuristic algorithm, named HLCWGC to solve HLMCT problem.

2) We analyze the performance of our algorithms through simulations. By comparing HLCWGC algorithm with other existing algorithms, HLCWGC algorithm achieves a better performance than other improved algorithms in the delay-sensitive sensor networks, which verifies the efficiency of our algorithm.

The rest of paper is organized as follows. Section 2 describes delay-constraint connected target coverage (DCCTC) problem and gives some related definitions. We also model DCCTC problem as a height limited maximum cover tree (HLMCT) problem and then prove that HLMCT is a NP-complete problem. In section 3, we describe our proposed HLCWGC algorithm in detailed, and analyze the cost for our algorithm. In section 4, we conduct simulations to verify the performance of HLCWGC. Section 5 concludes this paper and proposes the future work.

2 Problem Statements

2.1. Network Model

Assume that s1, s2,…, sN are N sensors in a WSN and R is the sink. All sensors are randomly deployed to continuously cover M targets p1, p2,…, pM. S and P are different sets of nodes, where S={ s1, s2,…, sN } (|S|=N) and P={ p1, p2,…, pM } (|P|=M). We use a directed graph G=(V,E) to represent the relationship among S (the set of sensor nodes), P (the set of target nodes) and R (the sink node), where V= $S \cup T \cup R$, E stands for transmission edges and coverage edges [12]. If |si—sj|<Rc, we say that (si, sj) $\in E$; if |si—pj|<Rs, we say (si,pj) $\in E$.

Within the operation time τ, we use $S_s(\tau)$ to represent the set of nodes participating in the activity of monitoring objectives in the operation time τ, we use $S_r(\tau)$ to represent the set of nodes participating in the activity of transmission data in the operation time τ, and we use $S_a(\tau)$ to represent the set of active nodes in the operation time τ. Obviously, $S_a(\tau) = S_s(\tau) \cup S_r(\tau), S_a(\tau) \subseteq S$.

Assume that 1) each node will not change its Active/Sleep status within the operation time τ; 2) the sampling frequency for each node is the same; 3) the data cannot be aggregated in the network, when a node receives N*B(τ) (N represents the number of neighbors) data from all of its neighbors, the data will be transmitted by the node directly; 4) delay between single-hop nodes is the same, we also ignore sleep waiting time and packages conflict in the MAC layer.

2.2. Definitions

Before diving into solving the delay-constraint connected target coverage problem, we give some fundamental definitions and notations used throughout this paper. In wireless sensor network, the data dissemination delays mostly influenced by the maximum number of hops from the node to the sink, that is to say, the larger the hops from the sink, the more delays exist between the node and the sink. So, we use the height of the routing tree (using the sink node as the root) to express the maximum delay of the network.

Definition 1: Delay-Constraint Connected Target Coverage (DCCTC) problem.

Given M targets with known locations and a delay-constraint of Δ WSN with N sensors randomly deployed in the area. To meet the requirements in the following conditions, we schedule the sensor nodes' sleep and active activities such that network lifetime can be maximized.

1) Each target can be continuously observed by at lease one node in the network.

2) The data of target node can be transmitted through the active nodes' single hop or multi-hop paths to the sink within the time of Δ.

Definition 2: Maximum Cover Tree (MCT) [9] problem.

Given a directed graph G= (V,E), the maximum height of the tree is H, and the initial energy of each node is E0(s), where V= $S \cup T \cup R$. E consists of the communication and monitoring edges. There are a serial trees $T(\tau_1)$, $T(\tau_2)$,..., $T(\tau_x)$, which height is constraint, and the operational time of these trees are τ_1 , τ_2 ,.... τ_x , the network can achieve the maximum lifetime denoted as L. We can express this problem by the following mathematical representation:

Maximize L = $\sum_{i=1}^{x}\tau_i$

Subject to: $\sum_{i=1}^{x}E(s,T(\tau_i)) \le E_0(s), \forall s \in S$,

where we use $T(\tau) = (S_s(\tau) \cup S_r(\tau), E(\tau))$ to represent the height-constraint trees constructed in operational time τ . The height-constraint tree has the following characteristics:

Sink is the root of the tree.

Each leave of the tree is a sampling node.

Each target can connect with at least one sampling node.

Each sampling node can have at most H-1 ancestors.

Each node of the tree $T(\tau)$ uses the same energy consumption model which is used in [9]:

$$E(s,T(\tau)) = \begin{cases} e_s B(\tau) + e_{trans} B(\tau), s \in S_s(\tau) and s \notin S_r(\tau) \\ (e_{trans} + e_r) B(\tau) D(s,T(\tau)), s \notin S_s(\tau) and s \in S_r(\tau) \\ e_s B(\tau) + e_{trans} B(\tau) + \\ (e_{trans} + e_r) B(\tau) D(s,T(\tau)), s \in S_s(\tau) \cap S_r(\tau) \\ 0, s \notin S_s(\tau) and s \notin S_r(\tau) \end{cases},$$

where $T(\tau)$ represents the trees constructed in operational time τ , $B(\tau)$ is the number of packets collected in operational time τ , e_s and e_r represent the energy dissipated by sensing and transmission of one bit data, respectively. e_{trans} represents the sending energy of a node, $e_{ij}^t = e_t + b \cdot d_{ij}^{\alpha}$ (where si is the sender, sj is the receiver, d_{ij} is the distance between si and sj, α is the attenuation factor of path, et and b are constant), so, $e_{trans} = e_{ij}^t = e_t + b \cdot d_{ij}^{\alpha}$, $D(s,T(\tau))$ represents the number of offspring nodes of s in tree $T(\tau)$.

Definition 3: Height Limited Maximum Cover Tree (HLMCT) problem.

Given a directed graph G= (V,E), the maximum height of the tree is H, and the initial energy of each node is E0(s), where V= V= $S \cup T \cup R$. E consists of the communication and monitoring edges. There are serial trees $T(\tau_1)$, $T(\tau_2)$,..., $T(\tau_x)$ with a limited height, and the operational time of these trees are τ_1 , τ_2 .... τ_x , the network can achieve the maximum lifetime denoted as L. We can express this problem by the following mathematical representation:

Maximize L = $\sum_{i=1}^{x}\tau_i$

Subject to: $\sum_{i=1}^{x}E(s,T(\tau_i)) \le E_0(s), \forall s \in S$

$Height(T(\tau_i)) \le H, 1 \le i \le x$

Theorem 1: HLMCT is a NP-complete problem.

Proof: MCT problem [9] is only one special case of HLMCT with $H = \infty$, and MSC problem [5] is one another special case of HLMCT with $H = 1$. According to the constraint strategy [13], because MCT problem and MSC problem are both NP-complete, so HLMCT is also NP-complete.

2.3. Upper Bound for HLMCT Problem

We will further relax HLMCT problem into a LP problem in this section, and then prove that LP problem is an upper bound for HLMCT problem.

We will find the height limited trees $T(\tau_1)$, $T(\tau_2)$,..., $T(\tau_x)$ in graph G, if the minimum hops from the node to the sink is larger than H, we call it LH node; if the minimum hops from the node to the sink is equal to H, we call it EH node; and if the minimum hops from the node to the sink is smaller than H, we call it SH node.

Each sensor in WSN has P sensing ranges. In order to solve the problem, we usually use a round-based set cover schema. We organize the sensors in sets, such that only one set is responsible for monitoring the targets and other sensors are in sleep mode. Next, we formally define the Multiple Sensing Ranges-Set Cover problem (MSR-SC).

Lemma 1: There are no LH nodes in the solution set of HLMCT.

Proof: If si is one of the LH nodes, since the minimum hops is larger than the height limitation of H, the data sensed by si are invalid when transmit to sink, thus, si also cannot forward the data sending from other nodes.

Lemma 2: If any solution of HLMCT contains EH nodes, then, EH nodes must be the leaves nodes.

Proof: If assuming EH node is not a leaf node, then the EH node will forward the data collected from other nodes, the delay of transmitting data to sink must exceed H, this is not possible to exist, so, EH node must be a leave node.

Rule 1: According to the following steps, we can adjust graph G to G'.

Find out the minimum hop between the node and the sink in graph G;

According to Lemma 1, Delete all the LH nodes and their communication and monitoring edges directly in graph G, which can get graph G1;

According to Lemma 2, Delete all the communication edges of (SH, EH) in graph G1, which can get graph G'.

In graph G', we will model the HLMCT problem into a linear programming problem as follows:

L respects the network lifetime, s_i spends the duration time of t_{im} to monitor the target p_m , F_{ij} is the total number of data from s_i flows to s_i , F_{iR} is the total number of data from s_i flows to R. IN_i represents the endpoint node set that has communication edges pointing to s_i , and OUT_i represents the endpoint node set that has communication edges leaving from s_i .

Maximize L

Subject to target constraint:

$$\sum_{i=1}^{N} t_{im} - L = 0 \;;\forall p_m \in P , \tag{1}$$

Flow constraint:

$$\sum_{j\in OUT_i} F_{ij} + F_{iR} - \sum_{j\in IN_i} F_{ji} - f_s \sum_{p_m\in P} t_{im} = 0 \;;\forall s_i \in S , \tag{2}$$

Energy constraint

$$\sum_{j\in OUT_i} F_{ij}\cdot e_{ij}^t + F_{iR}\cdot e_{iR}^t + \sum_{j\in IN_i} F_{ji}\cdot e_r + \sum_{p_m\in P} t_{im}\cdot f_s e_s \le E_0(s_i) \;; \quad \forall s_i \in S , \tag{3}$$

Theorem 2: The optimal solution of the above linear programming program is exactly the optimal solution's upper bound of HLMCT.

Proof: If we can prove that one optimal solution of HLMCT is also one feasible solution of the above LP problem, then, we can get the conclusion that the optimal solution of LP problem must be the optimal solution's upper bound of HLMCT. Assuming that there are many solutions of HLMCT problem, such as { $T(\tau_1)$,…, $T(\tau_x)$ } and{ τ_1 ,…, τ_x }, etc. Each target can be covered at least once in the duration of the operating time τ_a , we can get that:

$$\sum_{i=1}^{N} t_{ima} = \tau_a , \forall p_m \in P , \tag{4}$$

where t_{ima} represents the duration of s_i monitoring the target p_m within time τ_a . Equation (4) can be also success in other operating time, where a $\in$ [1,x], so, we can get that:

$$\sum_{a=1}^{x}\sum_{i=1}^{N} t_{ima} = \sum_{a=1}^{x} \tau_a \;,\; p_m \in P$$

That is to say, $\sum_{i=1}^{N} t_{im} - L = 0$, thus, Equation (1) is right, and Equation (2) and Equation (3) are also right, so, we can draw the conclusion that the optimal solution of the above linear programming program is exactly the optimal solution's upper bound of HLMCT.

Although we relax HLMCT problem into a linear programming program, we does not consider the delay constraint in this linear programming program. So, the upper bound L_{LP} is very relaxed in some certain extent.

3. The Design and Analysis of HLCWGC

3.1. HLCWGC Algorithm

The input of HLCWGC are S, P, R and τ , the output of HLCWGC are a series of height limited cover trees (T_1 ,…, T_x). For the sake of brevity in describing our algorithm in the following, we introduce some symbols that will be used in our algorithm here. Let S_l be the alive nodes set, S_s is another set of alive node, and nodes in S_s can cover the target. P_s is the target set that can be covered by node s. w_s represents the path weight of a node in HLMWCT. W(s) is the contribution of node s, where $s \in S_s$. R(s, T) represents the path of node s to sink R in tree T, where R(s, T) = < s, s_1 ,…, R>. $\overline{\mathrm{R}}(s,T)$ represents the set of node's paths in tree, except the source node s and destination node R. The algorithm shown in Fig. 1 gives detailed description of HLCWGC.

1. $S_l = S; S_s = \varnothing; x = 1;$
2. for each $s \in S_l$
3. $E_r(s) = E_0(s);$
4. if $P_s \neq \varnothing$, $S_s = S_s \cup \{s\}$;end if
5. end for
6. while $\cup_{s \in S} P_s = P$ **and** $S_l \neq \varnothing$,
7. phase1:
8. for each link(s_i, s_j) $w_{ij} = e_{ij}^t \times E_0(s_i) / E_r(s_i)$; end for
9. Build a tree with **HLSPT Algorithm**
10. phase 2:
11. $S_s' = \varnothing$; $P' = \varnothing$; $T_x = \varnothing$; $\tau_x = \tau$
12. while $P' \neq P$,
13. Find a sensor s^* with max profit W(s^*)
14. $S_s' = S_s' \cup \{s^*\}$; $P' = P' \cup P_{s^*}$
15. for each $s \in \overline{R}(s^*, T_x)$
16. $w_s = w_s + (e_{trans} + e_r)B(\tau) \times w_s / E_r(s)$
17. end for
18. end while
19. phase3:
20. for each $s \in S_s'$, $T_x = T_x \cup R(s, T_x)$; end for
21. for each $s \in T_x$, $\tau_x = \min(\tau_x, \dfrac{E_r(s)}{E_s(\tau_x, T_x)}\tau_x)$
22. for each $s \in T_x$, $E_r(s) = E_r(s) - E(s, T_x(\tau))$; end for
23. Remove some nodes and edges according to **Rule 1**; x =x+1;
24. end while

Fig. 1. HLCWGC algorithm.

We can see from Fig. 1 that 1-5 lines are the initial process of HLCWGC, each cover tree produced by HLCWGC will work for a fixed time of τ . If the nodes have no energy or the energy is not enough to sustain the system operating time of τ_i , we call them low energy node; If the nodes have enough energy, but the minimum hop is larger than H, we call them solitary far nodes. We will delete all the low energy nodes and solitary nodes from the network before each operation of building a new coverage tree, then there only effective nodes exist in the graph of the network. In our algorithm, every round will be finished in the following time:

In every round, the algorithm will be finished in the following time:

$$\tau_k = \min(\tau, \min_{s \in T_k}(\frac{E_r(s)}{E_s(\tau, T_k)}\tau))$$

In our algorithm, the construction of every cover tree contains three stages, in the first stage (line 7-9 in Fig. 1), an energy effective height-limited connect tree is constructed, the tree can connect each survival node; In the second stage (line 10-18 in Fig. 1), the algorithm greedily select nodes to cover all the target; The third stage (line 19-22 in Fig. 1) is used to update the energy of nodes.

In the first stage, algorithm gives each directed edge a weight value, such as the weight value of directed edge <si, sj> can be calculated by:

$$w_{ij} = \frac{e_{ij}^t \times E_0(s_i)}{E_r(s_i)}$$

Then a height-limited tree is constructed, since building a minimum weight height-limited tree is a NP-complete problem, so, we develop a heuristic algorithm, named HLSPT to construct the minimum weight height-limited tree, HLSPT can returns the path weight of nodes in the tree, the detailed description of HLSPT is shown in Fig. 2.

Construct a minimum communication weight tree

In the second stage, we use a greedy strategy to select nodes to cover all the targets. At first, the algorithm gives the node s a weight denoted by Profit(s):

$$\Pr ofit(s) = \frac{|P_s - P_s \cap P'|}{w_s},$$

where $|P_s - P_s \cap P'|$ represents the set of nodes that cannot be covered within the sensing range of node s, w_s is the path weight of node s, and w_s is exactly the sum of the weights of all edges in $R(s,T)$. The algorithm continually selects the node with the maximum weight (line 13 in Fig. 2) to cover the target until all targets have been covered. When a node s has been selected as a sampling node, then the node in $\overline{R}(s,T)$ will update their path weights (line 16 in Fig. 2).

In the third stage, the selected sampling nodes and $\overline{R}(s,T)$ will be added to the tree T, and their energy will be updated according to their energy model.

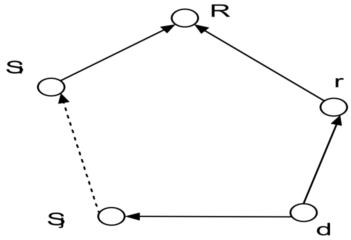
3.2. Construction Algorithm Description of HLSPT

The input of the algorithm is the graph G processed by rule 1, the output of the algorithm is the tree T with a minimum weight. The tree T uses all the effective sensor nodes in graph G as its children, and uses sink as its root.

Line 1-4 are the initial processes of the algorithm, for each node u, denote its original level in tree T as Level(u), denote its father node as dad(u), denote its path weight as W(u), and the minimum hops as hops(u), the status of the node in tree T is Status(u), and the father of the node in minimum hops algorithm is denoted as dad1(u); Line 5-16 is similar to a Dijastra algorithm, in this algorithm, construct a height-limited minimum weight tree, but this tree is not a spanning tree of graph G. Since some nodes have shorter hops to the sink, they would not be chosen in our algorithm.

For example, in the following figure, node d can connect R through the two hops path <d,r,R>, but its path weight is larger than that of <d,Sj…Si,R>, so, node d would not be added into path <d,r,R>. May be <d,Sj…Si,R> would exceed the height limitation of the tree, so, node d cannot be added to path <Sj…Si,R> to construct the path <d,Sj…Si,R>.

Line 17-27 uses a data structure queue Q to carry on the layer traversal of graph G, using dad1(u) and hops(u) to denote the father of node u and the minimum hops of node u to the sink, respectively. Line 28-40 use a heuristic method to find a minimum weight path satisfying the height limitation to reach the sink for the "off tree" nodes after running the algorithm which similar to Dijastra algorithm, and change the path weight information for the nodes.




```
1.  for each node s do : Level(s) =0;dad(s)=0;dad1(s)=0;hops(s)=0;W(s)=0;Status(s)="offtree";visited=0
2.  for each edge <s,R> ∈ G do
3.    Level(s)=1; dad(s)=s; W(s)= Wij (s,u); Status(s)="fringe";
4.  end
5.  while there is at least one "fringe" status node in G do
6.    select a fringe node v with the smallest weight among all "fringe" nodes
7.        Status(v) ="intree";
8.      for each edge(u,v) ∈ G do
9.        if Status(u) = "fringe" and Level(v)+1 ≤ H and W(v) > W(u)+ Wij (u,v) then
10.         Level(u)=Level(v)+1;Status(u)="fringe"; dad(u) = v; W(v)= W(u)+ Wij (u,v);
11.       end if
12.       if Status(u)="offtree" and Level(v)+1 ≤ H then
13.         Level(u)=Level(v)+1; Status(u)="fringe"; dad(u) = v; W(v)= W(u)+ Wij (u,v);
14.       end if
15.     end for
16. end while
17. Queue Q; Stack Sk; EnQueue(Q, R); hops(R)=0;dad1(R)=-1; visisted(R)=1;
18. While notEmpty(Q)
19.   v= DeQueue (Q);
20.   for each edge(u,v) ∈ G do
21.     if visited(u)=0 then
22.       EnQueue(Q,u); hops(u)=hops(v)+1; dad1(u)=v; visited(u)=1;
23.     else if W(v) + Wij (u,v) < W(dad1(v)) + Wij (u,dad1(v)) and hops(u)>hops(v) then
24.       dad1(u)=v;
25.     end if
26.   end for
27. end while
28. for each node u such that Status(u)="offtree" do
29.   dad(u)=dad1(u); v=dad1(u); PushStack(Sk,u);
30.   while dad1(v) ≠ -1 do
31.     if Level(v) +hops(u)-hops(v) ≤ H then
32.       break;
33.     else
34.       dad(v)=dad1(v); PushStack(Sk,v); v=dad1(v) ;
35.     end if;
36.   end while
37.   while notEmptyStack(Sk)
38.     x = popStack(Sk); y = dad(x); W(x) = W(y)+ Wij (x,y);
39.   end while
40. end for
```

Fig. 2. HLSPT algorithm.

3.3. Cost of HLCWGC

We use the solution L_{LP} of linear programming problem to represent the upper lifetime of network. It is easy to know, the time for each round is either τ or the duration of one node come to die, so, we can define the number of the cover trees within $N + L_{LP} / \tau$. The maximum cost of each round is MN, so it is easy to obtain that the time complexity of HLCWGC is O ($(N + L_{LP})MN / \tau$).

4. Performance Evaluation

In this section, we will compare HLCWGC with other three algorithms: CWGC, HLMSC-EWARE and HLMSC-SPT. The delay constraint problem is not considered by CWGC algorithm, HLCWGC adopts the same greedy strategy to select the source nodes, but they use different methods to generate the skeleton of the tree to transmit the data. HLMSC-EWARE is a MSC algorithm that considers the delay constraint problem, and it adopts the greedy strategy in [5] to select the source nodes, but HLMSC-

EWARE use the height-limited weight tree generated by HLSPT algorithm to transmit the data. While, in HLMSC_SPT algorithm, the data collected by source nodes will be transmitted through the shortest path to the sink, and it also adopts a greedy strategy in [5] to select the source nodes. In the simulations, we will evaluate the metrics of maximum delay of network and network lifetime among above four algorithms.

We simulate a network with sensors and targets randomly deployed in a 100 m×100 m area, the sink is located in the center of the area, and its coordinate is (50,50). We assume that each node has the same initial energy of 20 Joule (J), the other parameters are set as: $e_t = 50\,nj/bit$, b=100 pJ/m4, $\alpha = 4$, $e_r = 150\,nJ/bit$, $e_s = 150\,nJ/bit$, the data sampling frequency is 10 kbps[14], all the nodes have same sensing radius of Rs and some transmission radius of Rc, where $R_s = 20\,m$ and $R_c = 40\,m$.

4.1. The relationship between HLCWGC and L_{LP}

We assume there are 20 targets in the area, and we vary the number of sensors from 50 to 110 with an increment of 10, the upper of the network delay H are set as 20 and 10, respectively. The maximum network lifetime of HLCWGC under different L_{LP} are shown in Table 1.

Table 1. The comparison between HLCWGC and L_{LP}.

Sensor	HLCWGC		L_{LP}	
	Lifetime (H=20)	Lifetime (H=10)	Lifetime (H=20)	Lifetime (H=10)
50	294.08	292.65	2892.4	2892.4
60	847.15	614.48	3808.1	3808.1
70	1070.9	968.81	4055.4	4055.4
80	1934.7	1042.1	4435.7	4435.7
90	1890.4	1387.9	4671.5	4671.5
100	2822.5	1550.4	5028.6	5028.6
110	2867.9	1968.7	5249.9	5249.9

In Table 1 simulation results show that when the number of sensors becomes larger, the maximum lifetime achieved by HLCWGC become similar. Since the network delay is not considered in the linear programming process of L_{LP}, so there are larger differences between L_{LP} and the network lifetime achieved by HLCWGC. On the other hand, we set the simulation scene with the same initial nodes of SH nodes, so, the lifetime of the network is the same when H=20 and H=10.

4.2. The Relationship Between HLCWGC and the Size of the Network

We will set two scenes, scene 1 (network of small size): there are 80 sensors and 10 targets in this scene; scene 2 (network of big size): there are 200 sensors and 25 targets in this scene. We set network delay as 20 and 10, respectively, and study the network lifetime performance of different algorithms. The simulation results are shown in Table 2.

In Table 2, HLCWGC achieves a larger network lifetime than HLMSC-EWARE and HLMSC-SPT in the same upper limitation of network delay no mater how does the network size vary. Obviously, the network lifetime of HLCWGC is shorter than that of CWGC, but the maximum network delay of CWGC is apparently beyond the upper bound of the network delay constraint.

Table 2. The performance comparisons under different network sizes.

	Scene 1		Scene 2	
	Lifetime	Delay	Lifetime	Delay
HLCWGC	1944.9	20	5849.7	20
HLMSC-EWARE	650	20	3250	20
HLMSC-SPT	450	20	1050	20
HLCWGC	1287.12	10	2917.5	10
HLMSC-EWARE	600	10	2150	10
HLMSC-SPT	450	10	1050	10
CWGC	2631.85	29.6	8037.6	43.5

4.2. The Relationship Between HLCWGC and the Delay Constraint of the Network

We assume there are 60 sensors and 20 targets in the area, and we vary the delay constraint of network from 4 to 32 with an increment of 4. We study the performance of network lifetime among different algorithms, the simulation results are shown in Fig. 5.

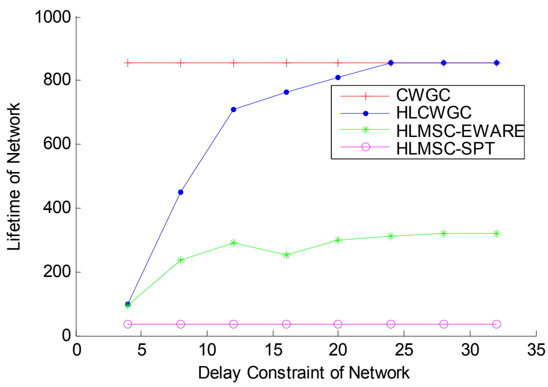


Fig. 5 The relationship between lifetime and delay constraint of network.

In Fig. 5, HLCWGC achieves a larger network lifetime than HLMSC-EWARE and HLMSC-SPT in different upper limitation of network delay constraint. In our simulation, when we set the delay constraint of network to 24, the network lifetime of HLCWGC is similar to that of CWGC.

5. Conclusions

This paper proposes the delay-constraint connected target coverage (DCCTC) problem in delay-sensitive sensor networks. Firstly, we model DCCTC problem as a height limited maximum cover tree (HLMCT) problem, and prove that HLMCT is a NP-complete problem, and then we give an upper bound on the network lifetime for HLMCT problem. We develop a fast heuristic algorithm, named HLCWGC to solve the HLMCT problem. Simulation results show that HLCWGC can achieve a better performance than other existing algorithms. In the future, we will consider the network with higher fault tolerance and higher target coverage ability, and the network can achieve the maximum network lifetime under the delay constraint request.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (Grant No. 61103245), the Natural Science Foundation of Guangxi (Grant No. 2012GXNSFBA053163).

References

[1]. Ren Fengyuan, Huang Haining, and Lin Chuang, Wireless sensor networks, *Journal of Software*, Vol. 14, No. 7, 2003, pp. 1282-1291.

[2]. Cui Li, Ju Hailin, Miao Yong, et al., Overview of wireless sensor networks, *Journal of Computer Research and Development*, Vol. 42, Issue 1, 2005, pp. 163-174.

[3]. S. Slijepcevic, M. Potkonjak, Power efficient organization of wireless sensor networks, in *Proceedings of the IEEE International Conference on Communications*, Helsinki, Finland, June 2001, Vol. 2, pp. 472-476.

[4]. M. Cardei and D.-Z. Du, Improving wireless sensor network lifetime through power aware organization, *Wireless Networks*, Vol. 11, No. 3, May 2005, pp. 333–340.

[5]. M. Cardei, M. Thai, Y. Li, and W. Wu. Energy-efficient target coverage in wireless sensor networks, in *Proceedings of the 24th conference of the IEEE Communications Society (INFOCOM'05)*, March 2005, pp. 1976-1984.

[6]. M. Cardei, J.Wu, M. Lu, and M. O. Pervaiz, Maximum network lifetime in wireless sensor networks with adjustable sensing ranges, in *Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2005, Vol. 3, pp. 438-445.

[7]. M. Lu, J. Wu, M. Cardei, M. Li, Energy efficient connected coverage of discrete targets in wireless sensor networks, in *Proceedings of the International Conference, Computer Networks and Mobile Computing (ICCNMC'05)*, Zhang Jiajie, P. R. China, 2005, pp. 43-52.

[8]. Q. Zhao and M. Gurusamy, Lifetime maximization using observation time scheduling in multi-hop sensor networks, in *Proceedings of the IEEE/Create Network International Workshop on Broadband Advanced Sensor Networks (BroadNets)*, 2005.

[9]. Q. Zhao and M. Gurusamy, Lifetime maximization for connected target coverage in wireless sensor networks, *IEEE/ACM Transactions on Networking*, Vol. 16, No. 6, December 2008, pp. 1378-1391.

[10]. Deying Li, Jiannong Cao, Ming Liu, and Yuan Zheng, K-connected target coverage problem in wireless sensor networks, *Lecture Notes in Computer Science*, Vol. 4616, 2007, pp. 20-31.

[11]. Yu Gu, Hengchang Liu and Baohua Zhao, Target coverage with QoS requirements in wireless sensor networks, in *Proceedings of the IEEE International Conference on Intelligent Pervasive Computing*, October 2007, pp. 35-38.

[12]. A. Boukerche, R. Pazzi, and R. Araujo, A fast and reliable protocol for wireless sensor networks in critical conditions monitoring applications, in *Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems MSWiM'04*, Venice, Italy, October 2004, pp. 157-164.

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

[14]. J.-H. Chang and L. Tassiulas, Maximum lifetime routing in wireless sensor networks, *IEEE/ACM Transactions on Networking*, Vol. 12, No. 4, 2004, pp. 609–619.

[15]. P. J. Chuang, Y. C. Yu, and C. S. Lin, Reliable and congestion-controlling transport in wireless sensor networks, *Journal of the Chinese Institute of Engineers*, Vol. 36, No. 1, 2013, pp. 2-16.

[16]. A. F. Liu, Z. M. Zheng, C. Zhang, et al., Secure and energy-efficient disjoint multi-path routing for WSNs, *IEEE Transactions on Vehicular Technology*, Vol. 61, Issue 7, 2012, pp. 3255-3265.

[17]. A. F. Liu, Z. H. Liu, et al., An elaborate chronological and spatial analysis of energy hole for wireless sensor networks, *Computer Standards and Interfaces*, Vol. 35, Issue 1, 2013, pp. 132-149.

[18]. K. Ota, M. Dong, Z. Cheng, J. Wang, X. Li, and X. Shen, ORACLE: mobility control in wireless sensor and actor networks, *Elsevier Computer Communications*, Vol. 35, Issue 9, 2012, pp. 1029-1037.