

Localized Low-Power Topology Control Algorithms in IEEE 802.15.4-based Sensor Networks

Jian Ma¹, Min Gao¹, Qian Zhang², Lionel M. Ni¹ and Wenwu Zhu²

¹Department of Computer Science
Hong Kong University of Science and Technology
Kowloon, Hong Kong, China
{majian, mgao, ni}@cs.ust.hk

²Wireless and Networking Group
Microsoft Research Asia
Hai Dian District, Beijing, China
{qianz, wwzhu}@microsoft.com

Abstract

Sensor networks have emerged as a promising technology with various applications, and power consumption is one of the key issues. Since each full function device can act as a coordinator or a device in IEEE 802.15.4 standard, 802.15.4-based sensor networks have various possible network topologies. In this paper, we try to construct network topologies with small number of coordinators while still maintaining network connectivity. By reducing the number of coordinators, the average duty cycle is reduced and the battery life is prolonged. Three topology control algorithms are proposed in this paper. Self-pruning is the simplest one with $O(1)$ running time. Ordinal pruning significantly improves self-pruning in terms of power saving with $O(n)$ running time. Layered pruning is a tradeoff between the first two pruning algorithms with $O(\sqrt{n})$ running time and a little higher power consumption than ordinal pruning. Furthermore, all three algorithms are independent of the physical radio propagation characteristics.

1. Introduction

Recent advances in wireless communications and micro-electromechanical systems (MEMS) have enabled the development of low-cost sensor networks. Sensor networks can bring a wide range of promising applications such as habitats monitoring, earthquake

report, location tracking, surveillance and healthcare [1] [2] [3].

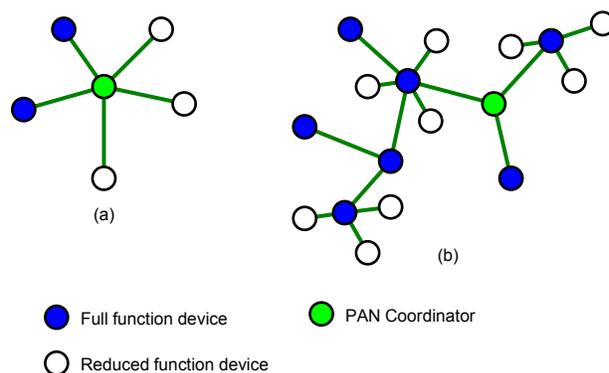


Figure 1: Network topology of a PAN.

Based on these features, a new standard, named IEEE 802.15.4, has been developed recently which focuses on low rate wireless personal area networks (LR-WPAN) [4]. The objectives of the standard are short-range operation, extremely low cost, and a reasonable battery life, while maintaining a simple and flexible protocol. The typical communication range is around 10 meters.

The standard defines three possible roles for a node in a PAN: *PAN coordinator*, *coordinator*, and *device*. Each PAN has a unique PAN coordinator, which determines a 16-bit PAN ID and a logical channel. All the communications in the PAN occur in this logical channel. In order to join a PAN, other devices send association request to the corresponding PAN coordinator. After several nodes have joined the PAN, a star topology is setup as illustrated in Figure 1 (a). Because the transmission range of the PAN coordinator is limited, other nodes can act as coordinators and accept association

This research was supported in part by Hong Kong RGC Grants HKUST6161/03E and AoE/E-01/99. This work was initiated when Jian is a visiting student in MSR Asia.

requests. Therefore, the range of the whole PAN can be extended. A more complicated PAN with one PAN coordinator and several coordinators are shown in Figure 1 (b). Since a PAN coordinator can be regarded as a special kind of coordinators, we use coordinator to refer either PAN coordinator or coordinator in later text unless otherwise specified.

There are two device types defined in the standard: *full function device* (FFD) and *reduced function device* (RFD). An FFD can talk to RFDs and other FFDs, and operate in one of the three possible modes: a PAN coordinator, a coordinator, or a device. RFDs are intended for extremely simple nodes implemented with very low cost so that an RFD can talk only to the coordinator with which it is currently associated.

Every 802.15.4-based sensor node has a unique 64-bit *extended address*. After associated with a PAN, a node will be assigned a 16-bit *short address* by its PAN coordinator. The short address is introduced to reduce the communication overhead in addressing.

A coordinator broadcasts beacons periodically if it is in a beacon-enable mode. Unassociated nodes discover nearby coordinators by listening beacons. All the associated nodes check the beacons and synchronize with their coordinator. When a coordinator wants to send a data frame to one of its associated node, it either forwards the frame directly or adds the frame into its transaction list and indicates the traffic in next beacon. The targeted node checks the incoming beacons and extracts its pending frame from the coordinator using CSMA-CA channel access mechanism. The whole procedure is called *indirect data transmission*. If a node does not have frames to send, it just need turn on its receiver during the period when its coordinator sends beacons. According to the power consumption characteristics shown in Table 1, indirect data transmission is an important mechanism to save power. The duty cycle of transceivers on these nodes is thus minimized. From the view of whole PAN, the average duty cycle is reduced and the battery life is prolonged significantly.

Table 1: 802.15.4 radio parameters [5]

Different mode	Power consumption
Transmit	36.5 mW
Receive	41.4 mW
Idle	41.4 mW
Sleep	42 μ W

Let us consider the scenario in which all sensor nodes in an 802.15.4-based sensor network form a PAN. Naturally the sink node, which acts as a gateway node to communicate with the outside world [1], should act as the PAN coordinator since it usually has powerful computing capability and unlimited power supply. For other FFDs in the sensor network, they act as either a

coordinator or a device. All the coordinators will form a peer-to-peer topology. They always turn on their radio and buffer traffic for their associated nodes. Therefore, other nodes that choose to be devices just turn on their transceiver when they send frames, listen to the beacons, or extract pending frames from their coordinators. There are many possible network topologies for the whole PAN; and the minimum requirement is to keep network connectivity so that every node could send back sensory data to the sink. To fulfill this requirement, a RFD will require one of its neighboring FFDs to become its coordinator. In order to simplify the problem, we just consider the scenario in which all the nodes are FFDs. They can freely select their role without considering their neighboring RFDs except the sink node. In order to lower power consumption, we should choose as few coordinators as possible.

The rest of this paper is organized as follows. Section 2 presents an overview on the closely related work. In Section 3, we describe our topology control algorithms to reduce the number of coordinators in detail. Section 4 shows the simulation results, which demonstrate the effectiveness of our proposed algorithms. Finally, we conclude the paper by giving the conclusion and future work.

2. Related Work

While the IEEE 802.15.4 standard is chartered to investigate a low data rate MAC solution with long battery life and very low complexity, ZigBee [6] takes full advantage of the powerful physical radio specified by the IEEE 802.15.4 standard and adds the logical network, security and application layer.

Several papers focus on the performance evaluation of IEEE 802.15.4 MAC protocols. Lu et al. [7] analyzed the tradeoff between tracking and non-tracking mode in star topology of 802.15.4-based networks. A more complex peer-to-peer topology with 101 nodes was simulated by Zheng and Li in [8]. Association efficiency, collision caused by hidden terminals, and duty cycle under this topology are quantitatively analyzed.

Wu and Li [9] proposed a marking scheme to construct a connected dominating set in ad hoc networks. The nodes in connected dominating set are called *gateway* nodes. Each node that has two directly unconnected neighbors is marked as a gateway node. Two pruning rules are used to reduce the size of the gateway set. According to pruning rule 1, a gateway can become a non-gateway if all of its neighbors are also neighbors of another neighbor that has a higher priority. According to pruning rule 2, a gateway can become a non-gateway if all of its neighbors are also neighbors of either of two other neighbors that are directly connected and have higher priorities. Dai and Wu [10] extended the previous

algorithm to a more general pruning rule called rule k : a gateway becomes a non-gateway if all of its neighbors are also neighbors of any one of k other nodes that are connected and have higher priorities. Two-hop neighborhood information is required to implement rule k .

Chen et al. [11] proposed the Span protocol to construct a set of coordinators based on clustering. A node becomes a coordinator if it has two neighbors that are not directly connected, indirectly connected via one intermediate coordinator, or indirectly connected via two coordinators. The priority of a node is computed from its energy level, node degree, and the number of pairs of its neighbors that are not directly connected. A node changes from a coordinator to a non-coordinator when its energy level is low. However, the network connectivity cannot be guaranteed since two coordinators may simultaneously become non-coordinators.

Stojmenovic et al. [12] extended Dai and Wu's algorithm by assuming that every node knows its accurate location. Every node only maintains a list of its neighbors and their geographic positions. Only 1-hop information is needed to implement the algorithm.

In addition to saving power, the topology control technique based on selecting coordinators can also be applied to avoid the broadcast storm [13]. Wu and Dai [14] proposed a generic broadcast protocol based on a simple self-pruning rule. Broadcast history information is also used in determining the role of nodes.

3. Pruning Algorithms

In this paper, we assume that all wireless links in a sensor network are bidirectional. Since 802.15.4 MAC has ACK mechanism for every frame, this assumption is reasonable. As shown in Figure 2, node A has a route to the sink node E. If node B is just a device, it is associated with either node A or node C. If node B's coordinator is node A, node B cannot synchronize with node C. When a packet for node A comes from the sink node E to node C, node C cannot forward this packet to node B since it does not know when the receiver of node B will be turned on. Symmetrically, if node B is associated with node C, a packet from node A cannot reach the sink node E. The above analysis also applies to other intermediate nodes such as node C and D. Generally, we have the following theorem.

Theorem 1: A route between two nodes is bidirectionally connected if and only if all intermediate nodes along the route are coordinators.

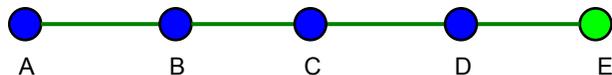


Figure 2: A connected route to the sink.

Based on Theorem 1, we transform our problem to finding a minimal connected coordinator set in a sensor network. Every node either should be a coordinator or associated with one of the coordinators.

Our problem can be formalized as follow. A sensor network with bidirectional links is represented by a graph $G = (V, E)$, where V is the set of sensor nodes and E is the set of all bidirectional links. We denote the sink node as node s . For each node v , $N(v) = \{u \mid (u, v) \in E\}$ denotes its neighbor set. For a node set S , $N(S)$ is the union of $N(v)$ for every node v in S . Furthermore, node v has a unique ID: $id(v)$. In IEEE 802.15.4 protocol, this ID corresponds to a 64-bit extended address since the association processes are not started yet. We want to find a *connected dominating set* in G , which includes node s and has minimum number of coordinators.

We state other assumptions here. First, there are no isolated nodes in sensor networks; otherwise, the connected dominating set does not exist. We also assume that all the nodes are static, which is very common in current sensor networks. During the initial setup stage, the well-known Bellman-Ford algorithm is started from the sink node so that each node can determine its shortest hop distance to the sink node. All the nodes with the same hop distance k forms the *layer k*. The shortest hop distance of node v is denoted as $d(v)$. Obviously $d(s) = 0$, and for every node u in $N(v)$, $d(v) - 1 \leq d(u) \leq d(v) + 1$. For every node v , a neighbor with hop distance smaller than $d(v)$ is called a *precedent node* of v . $P(v) = \{u \in N(v) \text{ and } d(u) = d(v) - 1\}$ denotes the *precedent set* of v . The set $P(v)$ is nonempty for every node except the sink node s . In the network setup stage, every node broadcasts its neighbor list with their hop distance to all its neighbors so that 2-hop information is collected. For a static sensor network, the neighborhood information does not require to be updated. Furthermore, we only rely on the neighborhood information in which two nodes are connected if and only if they can receive the signals from each other, our solutions are thus independent of the physical radio models.

We present a series of localized *pruning* algorithms below. Pruning is the decision that a node is excluded from the dominating set since the default role is coordinator. Localized feature is desired in designing algorithms for sensor networks to reduce the communication overhead [15].

3.1. Self Pruning

In a sensor network, we observe that the nodes near the sink node always need forward the traffic between the sink and nodes far from the sink. At first, we define the priority of a node to be a coordinator as follows. If two nodes have different hop distances, the one with

smaller hop distance has a higher priority; otherwise, the node with a smaller node ID has a higher priority.

Our first pruning algorithm in sensor networks is a variation of the pruning rule k [10]. For each node v , let the set $S(v) = N(v) \cap \{u \mid u \text{ has a higher priority than } v\}$. Node v will not be a coordinator if

1. $S(v)$ is nonempty and connected;
2. $N(v) \subseteq S(v) \cup N(S(v))$.

The first condition is called *connectivity test*; and the second condition is called *coverage test*. In other words, the set $S(v)$ is a connected dominating set for the subgraph consists of all the neighbors of node v if the two tests are passed. Since each node decides its role independently, this algorithm is called *self-pruning* (SP). The SP algorithm is demonstrated in Figure 3. In order to explain our algorithm easily and clearly, the node IDs are assigned to follow our priority rule in the network so that nodes with higher priorities have smaller IDs. Node 0 is the sink node, and all nodes with hop distance 1 has smaller node IDs than nodes with hop distance 2. For node 3, the set $S(3) = \{0, 1, 2\}$ is connected. Node 7 is a neighbor of node 1; and node 8 is a neighbor of node 2. Based on our SP algorithm, node 3 acts as a device.

For low rate PAN, the end-to-end delay is mainly due to forwarding delay at intermediate nodes along a route rather than backoff and collision. An important estimation for the delay is the shortest path from the sender to the receiver. In order to analyze the length of shortest paths, let us give Lemma 1 first.

Lemma 1: Using self-pruning, for any node v except the sink, there exists a neighboring coordinator $u \in N(v)$ such that $d(u) = d(v) - 1$.

Proof: As we mentioned before, the precedent set $P(v)$ is nonempty. Let u be the node with minimum node ID in $P(v)$. If u is a coordinator, we reach the conclusion. Otherwise, node u must pass the coverage test. Thus, node v is directly connected to another node x in $S(u)$ which has a higher priority than node u . Based on our priority definition, $d(x)$ should be $d(v) - 1$ and $id(x) < id(u)$. This is a contradiction. ■

Based on Lemma 1, we can easily conclude that the selected dominating set does maintain network connectivity.

Theorem 2: Using self-pruning, each node v is connected to the sink node s through a route with length $d(v)$.

Proof: Let $d(v)=k$. Based on Lemma 1, v can find a route $(v=n_k, n_{k-1}, \dots, n_1, n_0=s)$ to the sink s such that $d(n_i)=i$. The route length is k . ■

Using SP algorithm, the set of selected coordinators not only maintains the network connectivity, but also keeps the length of shortest path for each node to the sink node.

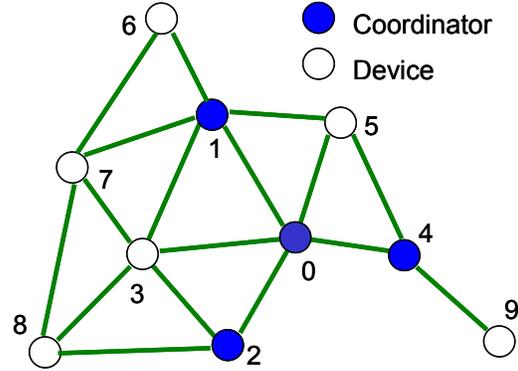


Figure 3: Self-pruning algorithm.

3.2. Ordinal Pruning

For each node, SP algorithm reserves one shortest path to the sink node. However, SP algorithm is not effective in reducing the number of coordinators. Since every node makes its decision independently, it does not consider the possibilities that some neighbors with lower priorities become coordinators. More neighboring coordinators could increase the chance to pass the coverage test. If the set of neighbors with higher priorities is a connected subgraph and covers the whole neighbor set, the expanded set should still be connected after adding other neighboring coordinators.

For each node v , let $S(v) = N(v) \cap \{u \mid u \text{ has a higher priority than } v \text{ or } u \text{ has a lower priority but becomes a coordinator}\}$. Node v will not be a coordinator if

1. $S(v)$ is nonempty and connected;
2. $N(v) \subseteq S(v) \cup N(S(v))$.

Since every node decides its role only after all its neighbors with lower priorities have determined their roles, this algorithm is called *ordinal pruning* (OP).

The comparison between SP and OP algorithms is demonstrated in Figure 4. The node IDs are assigned to follow our priority rule so that a node with smaller ID has a higher priority. Node 0 is the sink node. Each node with a circle around is selected as a coordinator. The result of SP algorithm is shown in Figure 4 (a). Figure 4 (b) depicts the generated topology of OP algorithm. The real lines represent the association relationship in which the node with a lower priority will be associated with the node with a higher priority. All the other communication links are shown as dash lines. Because the priority of node 15 is lower than that of all its neighbors, node 15 will decide its role first in OP algorithm. The S set of node 15, $\{7, 8, 14\}$, passes connectivity and coverage test. Node 15 acts as a device. After node 15 makes its decision, node 14 can decide its role because node 15 is the only node with a lower priority in its

neighborhood. Following the same rule, the running order of OP algorithm is sequentially from node 15 to node 1. To show the difference between SP and OP algorithms, we take node 5 as an example. In SP algorithm, the S set of node 5 is $\{1, 2, 4\}$. Since node 11 is not covered by this set, node 5 acts as a coordinator. In OP algorithm, the S set of node 5 is extended to $\{1, 2, 4, 6, 7\}$ because node 6 and 7 are both coordinators. Now the coverage test can be passed and node 5 acts as a device. Furthermore, we give the following theorem.

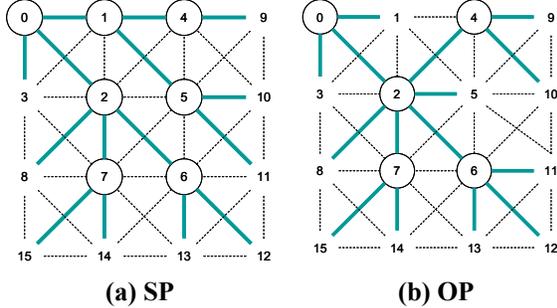


Figure 4: Comparison of SP and OP algorithms

Theorem 3: The coordinator set selected by OP algorithm is a subset of that of SP algorithm.

Proof: We just need prove that if a node v is not a coordinator using SP algorithm, it will not be a coordinator using OP algorithm. We denote $S_1(v)$ defined in SP algorithm as $S_1(v)$, and $S_2(v)$ defined in OP algorithm as $S_2(v)$. Obviously, we have $S_1(v) \subseteq S_2(v)$. Since $S_1(v)$ is nonempty, $S_2(v)$ should also be nonempty. The coverage test is also satisfied because

$$N(v) \subseteq S_1(v) \cup N(S_1(v)) \subseteq S_2(v) \cup N(S_2(v)).$$

Based on the coverage test, any node in $S_2(v) - S_1(v)$ could find a neighbor in $S_1(v)$. We know $S_2(v)$ is connected and the connectivity test is satisfied. We reach the conclusion. ■

Next, we prove that the selected coordinator set is a dominating set. A lemma is given first.

Lemma 2: Using ordinal pruning, for any node v except the sink, there exists a neighboring coordinator $u \in N(v)$ such that $d(u) \leq d(v)$.

Proof: As we mentioned before, the precedent set $P(v)$ is nonempty. Let w be the node with minimum node ID in $P(v)$. If w is a coordinator, we reach the conclusion. Otherwise, w must pass connectivity and coverage test.

If node v is not a coordinator, it will be directly connected by a node $u \in S(w)$ according to coverage test. Since w has the minimum node ID in $P(v)$, node u has a lower priority than node w . Based on the definition of S set in OP algorithm, node u is a coordinator.

If node v is a coordinator, then $v \in S(w)$. Because node w is not a coordinator, node w is not the sink node.

The precedent set of w is nonempty. Let x be one of the precedent nodes of w . Based on the connectivity test, node v and x is connected via a route in which all the nodes belong to $S(w)$. Let node u be the node next to node v in this route. Since w has the minimum node ID in $P(v)$, node u must be a coordinator. ■

Theorem 4: Using ordinal pruning, each node v connects to the sink node s .

Proof: We use induction to prove this theorem.

For each node v , we denote $\min P(v)$ as the node with minimum node ID in $P(v)$.

For each node with $d(v) = 1$, the conclusion is obviously true since v can directly connect to the sink s . If the conclusion is true for all the nodes with $d(v) \leq k$, then we consider node v with $d(v) = k+1$.

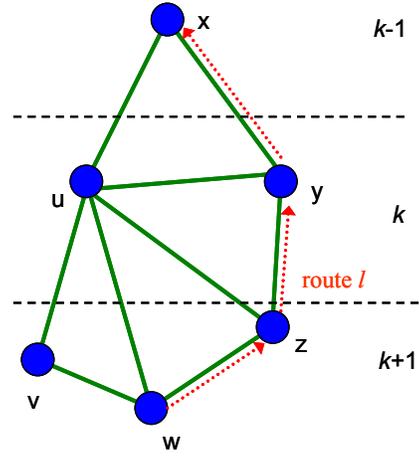


Figure 5: Illustration of the proof of Theorem 4

We assume that node v cannot find a route to the sink s . For each precedent node in $P(v)$, it is not a coordinator; otherwise, a route from v to sink s is found. Let node $u = \min P(v)$ and node $x = \min P(u)$. According to Lemma 2, there exists a coordinator $w \in N(v)$. Furthermore we know $d(w) = k+1$ since any node in $P(v)$ cannot act as a coordinator. Furthermore, based on the proof of Lemma 2, we get $w \in S(u)$.

According to connectivity test, $S(u)$ should be a connected set. Thus, node w is connected to x along a route l in $S(u)$ as illustrated in Figure 5. Because $d(x) = k-1$ and $d(w) = k+1$, the route l must go through one node with hop distance equal to k . Let node y be the first node with $d(y) = k$ along route l . All the intermediate nodes along the part of route l , which starts from node w to node y , are coordinators. If y is a coordinator, node v connects to sink s via node w and that part of route l to y . If y is not a coordinator, then y has a higher priority than u . According to our priority definition, we have $id(u) > id(y)$. We denote the previous node of y along

route l as node z . Then $d(z) = k+1$ and $y \in N(z)$. We obtain the inequality:

$$id(\min P(z)) \leq id(y) < id(u) = id(\min P(v))$$

If z has a route to sink s , v can also use the part of route l and the route of z to construct its own route to sink s . Otherwise, node z could find another node p such that $id(\min P(z)) > id(\min P(p))$ following the above analysis. This process can repeat infinitely, but the minimum node ID for nodes in layer k is finite. This is a contradiction. ■

3.3. Layered Pruning

OP algorithm reduces more coordinators than SP algorithm since all nodes sequentially decide their roles in OP algorithm, but the running time in OP algorithm is much longer. Let n denote the total number of nodes (sensors). The running time of the OP algorithm is $O(n)$ while that of SP algorithm is just $O(1)$. In order to get a tradeoff between the number of selected coordinators and running time, we propose another pruning algorithm called *layered pruning* (LP). Instead of waiting for the decisions of all the neighbors with lower priorities, each node just takes into account all neighbors with higher hop distance. Therefore, all the nodes in the same layer decide their roles simultaneously.

For each node v , let $S(v) = N(v) \cap \{u \mid u \text{ has a higher priority than } v \text{ or } d(u) > d(v) \text{ but } u \text{ becomes a coordinator}\}$. Node v will not be a coordinator if

1. $S(v)$ is nonempty and connected;
2. $N(v) \subseteq S(v) \cup N(S(v))$.

We use the same topology used in Figure 4 to demonstrate LP algorithm. Figure 6 depicts the generated topology of LP algorithm. Because the hop distance of node 15 is no less than that of all its neighbors, node 15 will make its decision first in LP algorithm. The same rule is applied to all nodes in layer 3. Nodes 9 to 15 compute their roles simultaneously in the first step. After receiving decisions from nodes 14 and 15, node 8 can decide its role because nodes 14 and 15 are the only two nodes with higher hop distances in its neighborhood. Following the same reason, all the nodes in layer 2 compute their roles in the second step. The running order of LP algorithm is from the furthest layer to the layer 0 and one layer is computed in each step. To show the difference between the three pruning algorithms, we take node 2 as an example. As shown in Figure 4 (a), the S set of node 2 is $\{0, 1\}$ in SP algorithm. Since node 6 is not covered by this set, the coverage test is failed and node 2 acts as a coordinator. As shown in Figure 4 (b), the S set of node 2 is $\{0, 1, 4, 6, 7\}$ in OP algorithm. Connectivity test is failed and node 2 acts as a coordi-

nator in OP algorithm. In LP algorithm, the S set of node 2 is $\{0, 1, 4, 5, 6, 7\}$. The two tests can be passed and node 2 acts as a device.

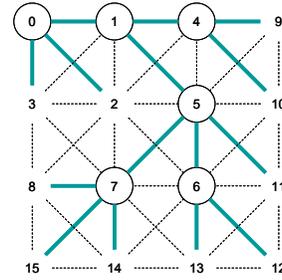


Figure 6: LP algorithm

The running time of LP algorithm depends on the hop distance of the farthest nodes. In a grid topology, the running time is $O(\sqrt{n})$. It is much shorter than the running time of OP algorithm especially when the scales of sensor networks are large. Furthermore, the performance of LP algorithm is given by the following theorem.

Theorem 5: The coordinator set selected by LP algorithm is a subset of that of SP algorithm.

Proof: We denote $S_1(v)$ defined in SP algorithm as $S_1(v)$, and $S(v)$ defined in OP algorithm as $S_2(v)$. Since the condition $S_1(v) \subseteq S_2(v)$ is still satisfied, we can use the exact same process as the proof of Theorem 3 to prove Theorem 5. We do not duplicate the proof here. ■

Based on Theorems 3 and 5, we know OP and LP algorithms are superior over SP algorithm in terms of power saving. Since the conditions to construct the S set in OP algorithm is less rigid than that of LP algorithm, the chance to pass the coverage test in OP algorithm is higher. Intuitively the size of dominating set selected by OP algorithm is no more than that of LP algorithm. However, there is not a subset relationship between the two sets. Comparing the resultant topologies shown in Figure 4 (b) and Figure 6, node 2 is selected as a coordinator by OP algorithm but is not selected by LP algorithm.

The correctness of LP algorithm is given in the following theorem.

Theorem 6: Using layered pruning, each node v connects to the sink node s .

Proof: The proof just applies the same method used in the proof of Lemma 2 and Theorem 4. The only difference is that the two pruning algorithms choose different S sets, but the process of deduction is not affected. We do not duplicate the proof here. ■

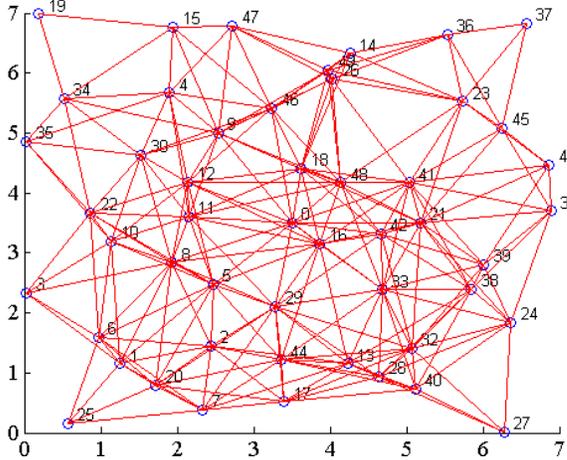


Figure 7: An example network with 49 nodes

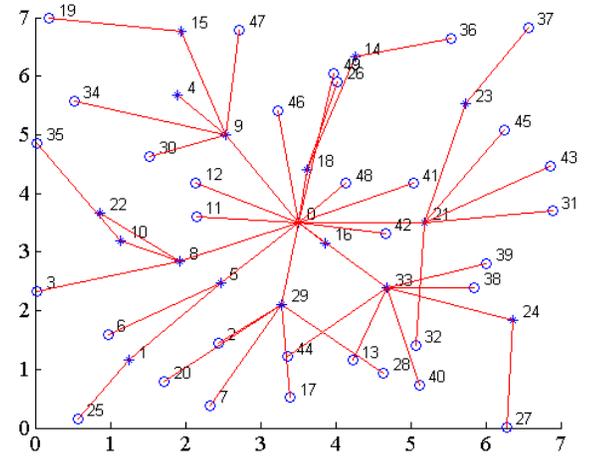


Figure 8: 17 coordinators selected by SP

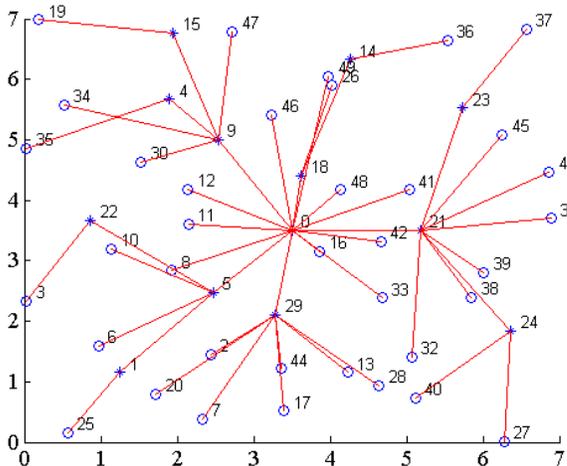


Figure 9: 13 coordinators selected by OP

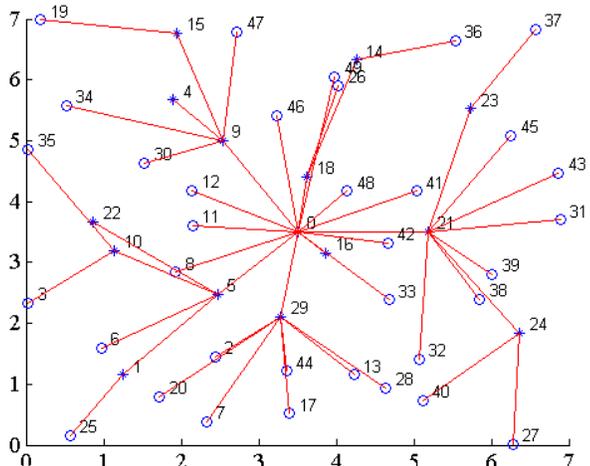


Figure 10: 15 coordinators selected by LP

4. Simulation Experiments

In our simulations, sensor nodes are distributed in a square terrain in accordance with predefined side length. The sink node is placed in the center of the square, marked as node 0. Two popular node placements are investigated, namely uniform and random:

- Uniform placement: the terrain is partitioned into unit grids. All the nodes are evenly divided amongst these grids (random distribution inside each grid).
- Random placement: it distributes all nodes randomly throughout the terrain.

In our network topology, a node directly connects to all the nodes that reside in its communication range. If the generated placement cannot maintain network connectivity, this placement will be skipped.

To demonstrate the effectiveness of our algorithms, a uniform distributed sensor network with 49 nodes is shown in Figure 7. The sink node 0 is placed in the center. A line represents that the two end nodes are connected with each other. The generated topologies for this sensor network are separately demonstrated in Figure 8, Figure 9 and Figure 10, respectively.

In these figures a node represented by a star acts as a coordinator; otherwise, the node acts as a device. The lines in these figures represent the association relationship.

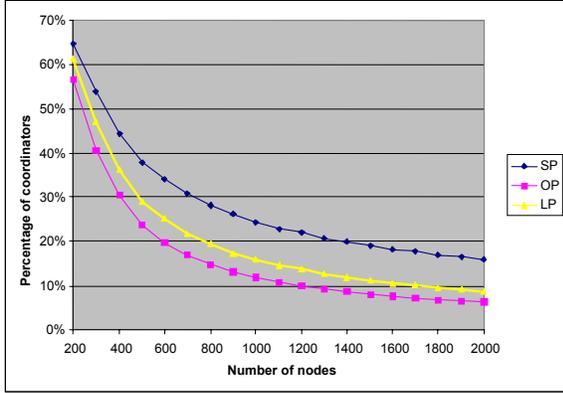


Figure 11: Percentage of coordinators (uniform)

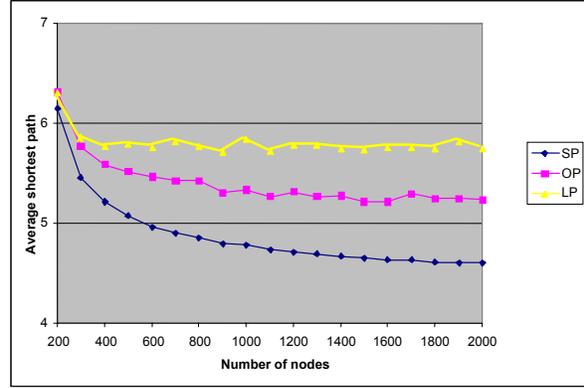


Figure 12: Average shortest path (uniform)

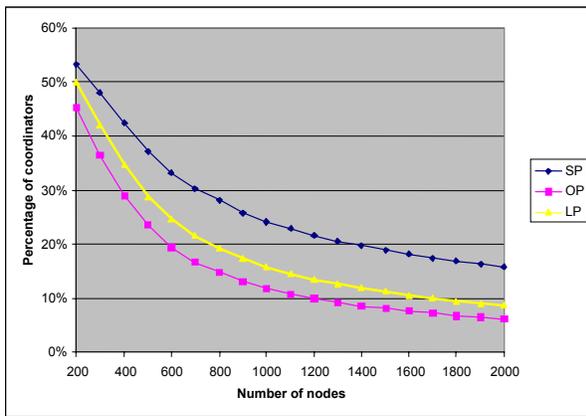


Figure 13: Percentage of coordinators (random)

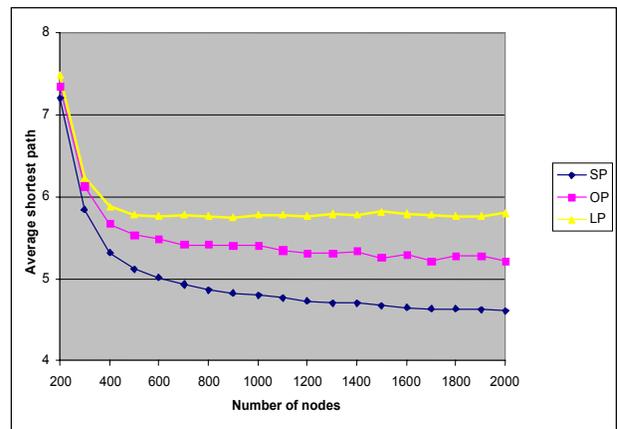


Figure 14: Average shortest path (random)

The terrain used in our simulations is 100 * 100 square meters. The communication range of each sensor node is 10m. To consider different density, the number of deployed sensor nodes is changed from 200 to 2000 gradually. For each density condition, the experiments are repeated 50 times and the average values are taken.

The percentage of selected coordinators for different pruning algorithms under uniform and random placement are shown in Figure 11 and Figure 13, respectively. The corresponding average lengths of shortest paths to the sink node are depicted in Figure 12 and Figure 14, respectively. The difference between the two placements is quite small if the number of deployed nodes is more than 600. In terms of power saving, OP algorithm is definitely the best one. The performance of LP algorithm is close to OP algorithm, and both of them outperform SP algorithm. In terms of end-to-end delay, the average lengths of shortest paths of all the algorithms are quite steady under high density. As we analyzed before, SP algo-

gorithm is the best one, and surprisingly OP algorithm outperforms LP algorithm with smaller number of selected coordinators.

The average length of shortest paths in LP algorithm is longer than that of OP algorithm because of their different way to handle the decisions of the neighbors in the same layer. As OP algorithm does wait for the decisions of neighbors in the same layer, there is some level of cooperation among the nodes in the same layer. However, all nodes in the same layer make their decisions independently in LP algorithm. When there is a sequence of nodes with increasing node IDs in one layer, the shortest paths in generated topology of LP algorithm are longer. An example is shown in Figure 15 and Figure 16. The result of OP algorithm is shown in Figure 15. Figure 16 depicts the generated topology of LP algorithm. The association relationship is represented by real lines and connectivity relationship is represented by dashed lines in these figures. Nodes 10 to 14 are deployed as a sequence of nodes with increasing node IDs in layer

4. Using LP algorithm, these nodes make their decision independently and all become coordinators. Then only node 6 is selected as a coordinator in layer 3. The route from node 20 to the sink node 0 under this topology is 8 hops rather than 5 hops in Figure 15.

In order to compare these pruning algorithms precisely, we use SP algorithm as the reference and show the relative performance of other pruning algorithms. The comparison of the percentage of selected coordinators is shown in Figure 17. OP and LP algorithms both perform better when the network density is higher. LP algorithm can save nearly 45% power and OP algorithm can even save more than 60% power compared with SP algorithm when the network is dense. The difference between OP and LP algorithm is less than 15%. The comparison of average length of shortest paths is shown in Figure 18. In high-density networks, the additional length for LP algorithm is from 15% to 26% compared with SP algorithm; and the additional length for OP algorithm is from 10% to 15%.

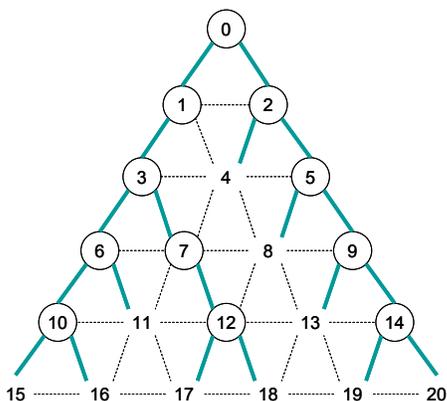


Figure 15: Shortest path in OP

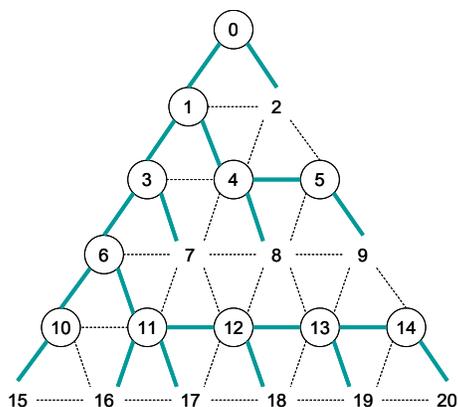


Figure 16: Shortest path in LP

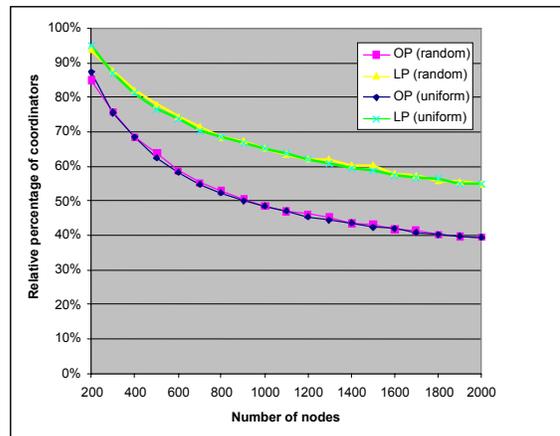


Figure 17: Comparison of coordinator sets

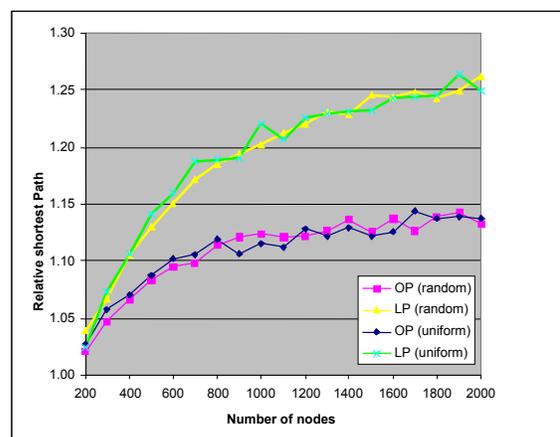


Figure 18: Comparison of shortest paths

5. Conclusion and Future Work

In this paper, three localized pruning algorithms are proposed to construct low-power network topologies in 802.15.4-based sensor networks.

Self-pruning is a variation of the pruning rule k . We prove that for each node, the generated topology maintains one shortest path to the sink node. Thus, self-pruning provides the shortest average end-to-end delay among three algorithms. Ordinal pruning significantly improves self-pruning in terms of power saving. Up to 60% power consumption can be reduced in dense sensor networks. The average end-to-end delay is increased around 10% to 15%. The major drawback of ordinal pruning is its long running time $O(n)$. Layered pruning can substantially reduce the running time to $O(\sqrt{n})$. The power consumption and average end-to-end delay both increase just less than 15% compared with ordinal pruning.

With their different characteristics in power saving, average end-to-end delay and running time, we provide a flexible set of topology control solutions. Different sensor networks can choose different algorithms based on their unique application requirements. Furthermore, all the algorithms just rely on neighborhood information so that they are independent of the physical radio propagation characteristics.

Our algorithms can also be applied to other kinds of networks that support global ID and the association operation in 802.15.4 MAC protocol.

Our future research includes analyzing the shortest path generated in OP and LP algorithms. In order to balance the power consumption among all the nodes and extend the lifetime of the whole sensor network, some role rotating policies should be defined. To increase system reliability, we may consider to provide an additional redundant to the sink node. Dynamic reconstruction of coordinators due to the failure of some coordinators should be considered. The radio model used in our simulation is the popular disk model, but in real world, the coverage areas are irregular due to multipath interference. We will incorporate some irregular radio models and show the performance of our algorithms. The priority among the nodes in same layer is determined by their node IDs. The priority rule could be more related to physical meaning if we add the remaining power into consideration. One of the models is the one used by He et al [16]. In this case, the solution will be more complicated since the priority relations are dynamic.

Reference

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *Communications Magazine, IEEE*, vol. 40, pp. 102-114, f2002.
- [2] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Commun. ACM*, vol. 43, pp. 51--58, 2000.
- [3] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*: ACM Press, 2002, pp. 88--97.
- [4] "IEEE standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks specific requirements part 15.4: wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (LR-WPANs)," in *IEEE Std 802.15.4-2003*, 2003, pp. 0_1-670.
- [5] Data Sheet for CC2420 2.4GHz IEEE 802.15.4/ZigBee Transceiver, http://www.chipcon.com/files/CC2420_Data_Sheet_1_2.pdf
- [6] ZigBee Alliance, www.zigbee.com
- [7] G. Lu, B. Krishnamachari, and C. Raghavendra, "Performance Evaluation of the IEEE 802.15.4 MAC for Low-Rate Low-Power Wireless Networks," presented at Workshop on Energy-Efficient Wireless Communications and Networks (EWCN '04), held in conjunction with the IEEE International Performance Computing and Communications Conference (IPCCC), 2004.
- [8] J. Zheng and M. J. Lee, "Will IEEE 802.15.4 Make Ubiquitous Networking a Reality?: A Discussion on a Potential Low Power, Low Bit Rate Standard," *IEEE Communications Magazine*, vol. 42, 2004.
- [9] J. Wu and H. Li, "On Calculating Connected Dominating Set for Efficient Routing in Ad Hoc Wireless Networks," presented at the Third International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, 1999.
- [10] F. Dai and J. Wu, "Distributed dominant pruning in ad hoc networks," presented at IEEE International Conference on Communications, 2003.
- [11] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An energy-efficient coordination algorithm for topology maintenance in Ad Hoc wireless networks," in *Proceedings of the 7th annual international conference on Mobile computing and networking*: ACM Press, 2001, pp. 85--96.
- [12] I. Stojmenovic, M. Seddigh, and J. Zunic, "Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, pp. 14-25, 2002.
- [13] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*: ACM Press, 1999, pp. 151--162.
- [14] J. Wu and F. Dai, "A generic broadcast protocol in ad hoc networks based on self-pruning," presented at Parallel and Distributed Processing Symposium, 2003.
- [15] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: scalable coordination in sensor networks," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*: ACM Press, 1999, pp. 263--270.
- [16] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *Proceedings of the 9th annual international conference on Mobile computing and networking*: ACM Press, 2003, pp. 81--95.