Neighbor Node Discovery in Asynchronous Sensor Networks

Sri Chandhana Pothula[1], R.Suguna[2], A.Chrispin Jiji[3]
Student1, Asst. Professor2, Asst.Professor3
Department of Electronics and Communication1,2
The Oxford College of Engineering1, 2, Bangalore-560068, Karnataka, India

Abstract

A sensor network may contain a huge number of simple sensor nodes that are deployed at some inspected site. In large areas, such a network usually has a mesh structure .In most sensor networks the nodes are static .Nevertheless ,node connectivity is subject to changes because of disruptions in wireless communication, transmission power changes, or loss of synchronization between neighbouring nodes. Hence, even after a sensor is aware of its immediate neighbours, it must continuously maintain its view, a process we call continuous neighbour discovery. In this work we distinguish between neighbour Discovery during sensor network initialization and continuous neighbour discovery. We focus on the latter and view it as a joint task of all the nodes in every connected segment. Each sensor employs a simple protocol in a coordinate effort to reduce power consumption without increasing the time required to detect hidden sensors.

I. INTRODUCTION

In the sensor network model considered in this work, the nodes are placed randomly over the area of interest and their first step is to detect their immediate neighbors - the nodes with which they have a direct wireless communication - and to establish routes to the gateway. In networks with continuously heavy traffic, the sensors need not invoke any special neighbor discovery protocol during normal operation. This is because any new node, or a node that has lost connectivity to its neighbors, can hear its neighbors simply by listening to the channel for a short time. However, for sensor networks with low and irregular traffic, a special neighbor discovery scheme should be used.

Despite the static nature of the sensors in many sensor networks, connectivity is still subject to changes even after the network has been established. The sensors must continuously look for new neighbors in order to accommodate the following situations:

- Loss of local synchronization due to accumulated clock drifts.
- Disruption of wireless connectivity between adjacent nodes by a temporary event, such as a passing car or animal, a dust storm, rain or fog. When these events are over, the hidden nodes must be rediscovered.
- The ongoing addition of new nodes, in some networks to compensate for nodes which have ceased to function because their energy has been exhausted.
- The increase in transmission power of some nodes, in response to certain events, such as detection of emergent situations.

For these reasons, detecting new links and nodes in sensor networks must be considered as an ongoing process. This work focuses on the continuous neighbour discovery and views it as a joint task of all the nodes in every connected segment. Each sensor employs a simple protocol in a coordinate effort to reduce power consumption without increasing the time required to detect neighbor sensors.

The main idea behind the detection of hidden neighbour sensor nodes continuously is that the task of finding a new node u is divided among all the nodes that can help v to detect u. These nodes are characterized as follows: (a) they are also neighbours of u; (b) they belong to a connected segment of nodes that have already detected each other; (c) node v also belongs to this segment. Let $deg_S(u)$ be the number of these nodes. This variable indicates the in-segment degree of a hidden neighbour u. In order to take advantage of the proposed discovery scheme, node v must estimate the value of $deg_S(u)$.

The remainder of the paper is organized as follows. In Section 2, the basic schemes are provided. Estimation values of the indegree of the neighbor nodes are presented in Section 3. Section 4 presents an algorithm for our findings in section 3 and section 5 describes the simulation results. The paper is concluded in Section 6.

II. BASIC SCHEMES

There are some particular schemes to be followed in the detection of hidden nodes in a sensor network.Detection of these hidden nodes depend on how they are located and also how they are connected.Two nodes are said to be neighbouring nodes if they have direct wireless connectivity. We assume that all nodes have the same transmission range, which means that connectivity is always bidirectional. During some parts of our analysis, we also assume that the network is a unit disk graph; namely, any pair of nodes that are within transmission range are neighbouring nodes.Consider a pair of neighbouring nodes that belong to the same segment but are not aware that they have direct wireless connectivity. See, for example, nodes a and c in Figure 2(a). These two nodes can learn about their hidden wireless link using the following simple scheme, which uses two message types: (a) SYNC messages for synchronization between all segment nodes, transmitted over known wireless links; (b) HELLO messages for detecting new neighbours.

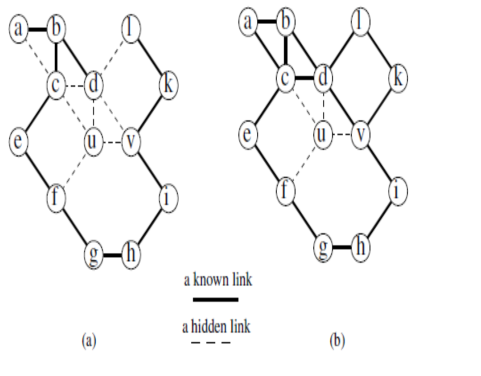

Fig- 2(a) & (b) Segments with hidden nodes and links

Scheme 1 (detecting all hidden links inside a segment):

This scheme is invoked when a new node is discovered by one of the segment nodes. The discovering node issues a special SYNC message to all segment members, asking them to wake up and periodically broadcast a bunch of HELLO messages. This SYNC message is distributed over the already known wireless links of the segment. Thus, it is guaranteed to be received by every segment node.

When scheme 1 is used,a hidden node is discovered by all of its in-segment neighbours as soon as it is discovered by the first of them. If scheme 1 is not used ,node u discovers all its in-segment hidden neighbours one by one. The expected delay in this case is the expected delay until the first discovery in a set of m neighbours plus the expected delay until the first discovery in a set of m-1 neighbours and so on.

Scheme 1 allows two neighbouring nodes u and v to discover each other if they belong to a connected segment. However, in order for two neighbours not yet connected to the same segment to detect each other, each node should also execute the following scheme:

Scheme 2 (detecting a hidden link outside a segment):Node u wakes up randomly, every T (u) seconds on the average, for a fixed period of time H.During this time it broadcasts several HELLO messages, and listens for possible HELLO messages sent by new neighbours. The value of T (u) is as follows:

- T (u) = T_I , if node u is in the Init state.
- T (u) = T_N(u), if node u is in the Normal state

By Scheme 1, the discovery of an individual node by any node in a segment leads to the discovery of this node by all of its neighbours that are part of this segment. Therefore, discovering a node that is not yet in the segment can be considered a joint task of all the neighbours of this node in the segment. As an example, consider Figure 2(a), which shows a segment S and a hidden node u. In this figure, a dashed line indicates a hidden wireless link, namely, a link between two nodes that have not yet discovered each other. A thick solid line indicates a known wireless link. After execution of Scheme 1, all hidden links in S are detected (see Figure 2(b)). The links connecting nodes in S to u are not detected because u does not belong to the segment. Node u has 4 hidden links to nodes in S. Hence, we say that the degree of u in S is deg_S (u) = 4. When u is discovered by one of its four neighbours in S, it will also be discovered by the rest of its neighbours in S as soon as Scheme 1 is reinvoked. Consider one of the four segment members that are within range of u, node v say. Although it may know about the segment members within its own transmission range, it does not know how many in-segment neighbours participate in discovering u. In the next section we study three methods that allow v to estimate the value of deg_S (u) for a hidden node u, and compare their accuracy and applicability.

III. ESTIMATING THE DEGREE

We consider the discovery of hidden neighbors as a joint task to be performed by all segment nodes. To determine the discovery load to be imposed on every segment node, namely, how often such a node should become active and send HELLO messages, we need to estimate the number of in-segment neighbors of every hidden node u,denoted by deg_S(u). In this section we present methods that can be used by node v in the continuous neighbour discovery state to estimate this value. Node u is assumed to not yet be connected to the segment, and it is in the initial neighbor discovery state.Three methods are presented:

- Node v measures the average in-segment degree of the segment's nodes, and uses this number as an estimate of the in-segment degree of u. The average in-segment degree of the segment's nodes can be calculated by the segment leader. To this end, it gets from every node in the segment a message indicating the in-segment degree of the sending node, which is known due to Scheme 1.

- Node v discovers, using Scheme 1, the number of its in-segment neighbors, $de_{gS}(v)$, and views this number as an estimate of $de_{gS}(u)$. This approach is expected to yield better results than the previous one when the degrees of neighboring nodes are strongly correlated.
- Node v uses the average in-segment degree of its segment's nodes and its own in-segment degree $deg_S(v)$ to estimate the number of node u's neighbors. This approach is expected to yield the best results if the correlation between the in-segment degrees of neighbouring nodes is known.

The in-segment degree of v and u depends on how the various nodes are distributed in the network. The above three methods for finding the estimations results in some MSEs which are tabularized as shown below with the corresponding $deg_S(u)$ value.

Method	$degs(u)$	MSE
1	μ	$\mathrm{Var}(X)$
2	X	$(2-2C)\,\mathrm{Var}(X)$
3	$CX+(1-C)\mu$	$(1-C^2)\,\mathrm{Var}(X)$

Table-1 : The three methods and their MSEs

Further we have proposed an algorithm which will be using one of the three methods presented above.

IV. ALGORITHM

In this section this work presents an algorithm for assigning HELLO message frequency to the nodes of the same segment. This algorithm is based on detecting all hidden links inside a segment. Namely, if a hidden node is discovered by one of its segment neighbors, it is discovered by all its other segment neighbors after a very short time. Hence, the discovery of a new neighbor is viewed as a joint effort of the whole segment. One of the three methods presented in Section 3 is used to estimate the number of nodes participating in this effort.

Suppose that node u is in initial neighbor discovery state, where it wakes up every T_I seconds for a period of time equal to H, and broadcasts HELLO messages. Suppose that the nodes of segment S should discover u within a time period T with probability P. Each node v in the segment S is in continuous neighbor discovery state, where it wakes up every $T_N(v)$ seconds for a period of time equal to H and broadcasts HELLO messages.

We assume that, in order to discover each other, nodes u and v should have an active period that overlaps by at least a portion δ, 0 < δ < 1, of their size H. Thus, if node u wakes up at time t for a period of H, node v should wake up between t - H(1 - δ) and t + H(1 - δ). The length of this valid time interval is 2H(1 - δ). Since the average time interval between two wake-up periods of v is $T_N(v)$, the probability that u and v discover each other during a specific HELLO interval of u is $2H(1 - \delta) / T_N(v)$.

Let n be the number of in-segment neighbors of u. When u wakes up and sends HELLO messages, the probability that at least one of its n neighbors is awake during a sufficiently long time interval is $1 - (1 - 2H(1 - \delta) /T_N(v))^n$.

Consider a division of the time axis of u into time slots of length H. The probability that u is awake in a given time slot is H / T_I, and the probability that u is discovered during this time slot is $P_1 = H / T_I(1 - (1 - 2H(1 - \delta)/T_N(v))^n)$. Denote by D the value of T/H . Then, the probability that u is discovered within at most D slots is $P_2 = 1- (1 - p_1)^D$. Therefore, we seek the value of $T_N(v)$ that satisfies the following equation:

$$1-(1-P_1)^D \geq P,$$

This can also be stated as

$$\text{Since } P_1 = \frac{H}{T_I}\left(1-\left(1-\frac{2H(1-\delta)}{T_N(v)}\right)^n\right), \text{ we get}$$

$$P_1 \geq 1-\sqrt[D]{1-P}.$$

$$\frac{H}{T_I}\left(1-\left(1-\frac{2H(1-\delta)}{T_N(v)}\right)^n\right) \geq 1-\sqrt[D]{1-P}$$

And therefore,

$$T_N(v) \leq \frac{2H(1-\delta)}{1-\sqrt[n]{1-\frac{T_I}{H}(1-\sqrt[D]{1-P})}}.$$

V. SIMULATION RESULTS

The simulation results in this works shows the discovered neighbours in the network within the transmission range of 200m.The Overall packets delivered, the packet delivery ratio and the energy consumed by the network is also shown in these results. Below fig:3 shows the neighbour node detection , fig:4 shows the achieved packet delivery ratio of the entire network and fig:5 shows the energy consumption by each node in the network.

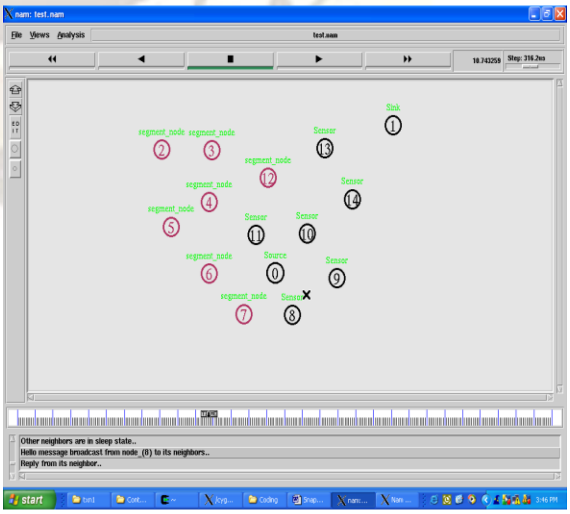

Fig 3: continuous neighbor discovery process

TRANSMISSION RANGE: (200m)

The transmission power is 2.0 watts.

Output:

Generated data Packets = 3100

Received data Packets = 2905

Packet Delivery Ratio = 93.7097%

Total Dropped Packets = 195

Routing Overhead = 3600

Forwarded Packets = 66

Average End-to-End Delay per packet = 1.33643 ms

Overall energy consumption by the network:

223.81 Joules

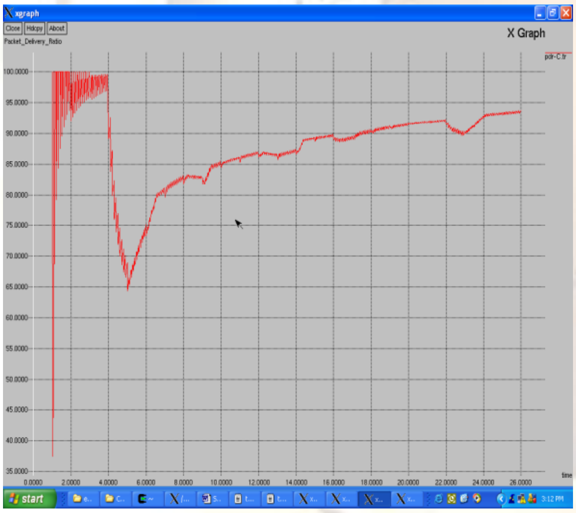

Fig 4: Packet delivery ratio is achieved to 93%

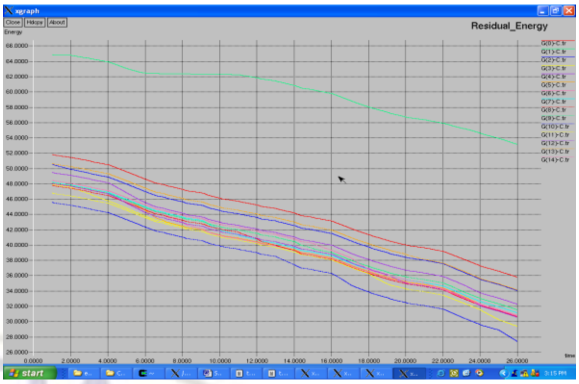

Fig 5: Energy consumption by each node

VI. CONCLUSION

The concern paper work is doing an investigation of finding neighbour nodes continuously in wireless sensor networks, referred to as ongoing continuous neighbor discovery. It argue that continuous neighbour discovery is crucial even if the sensor nodes are static. This paper presented the analytical as well as simulation results which will give clear insight for a practical design wireless sensor network.

REFERENCES

[1] S. Vasudevan, J. Kurose, and D. Towsley, .On neighbor discovery in wireless networks with directional antennas,. in *INFOCOM*, vol.4,2005.

[2] R. Madan and S. Lall, .An energy-optimal algorithm for neighbour discovery in wireless sensor networks,. *Mob. Netw. Appl.*, vol. 11, no. 3,pp. 317.326, 2006.

[3] M. J. McGlynn and S. A. Borbash, .Birthday protocols for low energy deployment and _exible neighbor discovery in ad hoc wireless networks,. in *MobiHoc: Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing*. New York,NY, USA: ACM Press, 2001, pp. 137.145.

[4] D. Baker and A. Ephremides, .The architectural organization of a mobile radio network via a distributed algorithm,. in *IEEE Transactions on Communications*, vol. 29, Nov. 1981, pp. 1694.1701.

[5] A. Keshavarzian and E. Uysal-Biyikoglu, .Energy-ef_cient link assessment in wireless sensor networks,. in *INFOCOM*, 2004.

[6] E. B. Hamida, G. Chelius, and E. Fleury, Revisiting neighbor discoverywith interferences consideration,. in *PE-WASUN*, 2006, pp. 74.81.