ISSN 2249-5460

INTERNATIONAL JOURNAL OF MATHEMATICAL SCIENCES, TECHNOLOGY AND HUMANITIES

www.internationalejournals.com

Available online at ww.internationaleJournals.com

International eJournals

International Journal of Mathematical Sciences, Technology and Humanities 71 (2012) 798 – 804

Detection of Hidden Nodes in Sensor Networks

Venkata Durgarao.Matta[#1], Ch.MuraliKrishna[#2]

[#]Computer Science and Engineering, JNTU, Kakinada. Email: durgaraomatta@gmail.com
[#2]Gokul Institute of Technology and Science, Piridi, Andhra Pradesh, India.
Email: chinnimuralikrishna.ssis@gmail.com

Abstract:

In networks with continuously heavy traffic, the sensors need not invoke any neighbor discovery protocol during operation. Most of the sensor nodes are used to send data, but some of the nodes are in inactive state. The data transmission in sensor nodes takes high bandwidth, energy and power. In this work showed how the scheme works well if every node connected to a segment estimates the in-degree of its possible hidden neighbors and then presented a neighbor discovery algorithm to determine the frequency of hidden nodes.

Keywords: Hidden neighbors, Sensor nodes.

1. INTRODUCTION

The sensor nodes can sense various events very sensitively. The sensor network contains very large number of these sensor nodes. These sensor nodes may be connected to each other inside a network by any mesh structure. Some of the sensor nodes act as routers and gateways to pass the message from one particular sensor node to another sensor node. In order to pass the data there will be high consumption of bandwidth, energy and even power.

Therefore we design this project in such a way that we can minimize these three critical issues. These issues can be overcome by alternatively putting the sensor nodes inactive state and passive state. In this paper the sensor nodes are randomly distributed over particular area and each sensor nodes have certain transmission area to cover. The first step is to detect the immediate neighbors. The sensor nodes should have direct wireless communication between them. Then the sensor nodes should establish the particular roots through which they can communicate with the other sensor nodes via any router or gateway in between. In order to communicate we need to first create communication between two sensor nodes.

The sensor nodes will be awake for a very short period of time. Therefore there can be heavy traffic in the channel or in the particular transmission area. This paper presents a special neighbor discovery scheme that can be used to reduce the traffic that is being caused by the sensor nodes. Another important issue in the sensor network is that the sensor nodes despite of being static can change due to the following situation.

1) Loss of local synchronization due to accumulated clock drifts.
2) 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.
3) The ongoing addition of new nodes, in some networks to compensate for nodes which have ceased to function because their energy has been exhausted.
4) The increase in transmission power of some nodes, in response to certain events, such as detection of emergent situations.

After resolving the four issues the sensor nodes can be there in two states. One is the Init state and the second is the normal state. Now in this discussion a main idea is to discover the links during the normal operation, and this is referred to as Continuous Neighbor Discovery. Now we have to discuss about how the nodes are being discovered by Continuous Neighbor Discovery algorithm. At first when the sensor nodes is in Init state, it remains.

Active for a very short period of time let's say from T1 second to T2 second. Now for this Particular period of time this sensor node will search for any other sensor node which is active in between that specific time period. If any sensor node is active at that period, the first sensor node repeatedly transmits HELLO packets to the next active sensor node. The other sensor node replies back by sending the ACK packet to the previous sensor nodes and therefore the two way communication between the sensor nodes is being established.

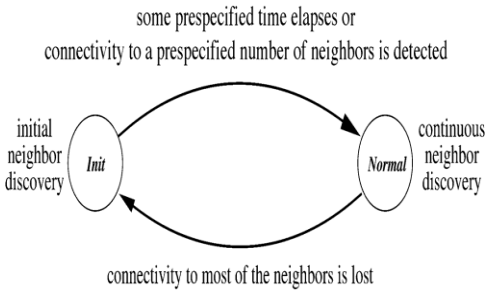

Fig: Continuous neighbor discovery vs.Intial neighbor discovery

In this protocol, a node becomes active according to its duty cycle. Let this duty cycle be in Init state and in Normal state. When a node becomes active, it transmits periodical HELLO messages and listens for similar messages from possible neighbors. A node that receives a HELLO message immediately responds and the two nodes can invoke another procedure to finalize the setup of their joint wireless link.

II. RELATED WORK

The neighbor discovery is the process of having a new node detected by the base station. Since energy consumption is not a concern for the base station, discovering new nodes is rather easy. The base station periodically broadcasts a special HELLO message.

.A regular node that hears this message can initiate a registration process. The regular node can switch frequencies/channels in order to find the best HELLO message for its needs. Which message is the best might depend on the identity of the broadcasting base station, on security consideration. Problems related to possible collisions of registration messages in such a network are addressed .Other works try to minimize neighbor discovery time by optimizing the broadcast rate of the HELLO messages. In addition, the hidden nodes are assumed to be able to hear the HELLO messages broadcast by the central node.

In contrast, detection of hidden nodes in sensor networks is performed by every node, and hidden nodes cannot hear the HELLO messages when they sleep. In mobile ad-hocnetworks (Manets), nodes usually do not switch to a special sleep state. Therefore, two neighboring nodes can send messages to each other whenever their physical distance allows communication. As in Wi-Fi, the process of neighbor discovery in Bluetooth is also asymmetric. Anode that wants to be discovered switches to an inquiry scan mode, whereas a node that wants to discover its neighbors enters the inquiry mode. In the inquiry scan mode, the node listens for a certain period on each of the 32 frequencies dedicated to neighbor discovery, while the discovering node passes through these frequencies one by one and broadcasts HELLO in each of them. This process is considered to be energy consuming and slow. Asymmetric neighbor discovery scheme for Bluetooth is proposed. The idea is to allow each node to switch between the inquiry scan mode and the inquiry mode. Under this mode, a newly deployed node should transmit a beacon request on each available channel. A network coordinator that hears such a request should immediately answer with a beacon of its own. However, this scheme does not supply any bound on the hidden neighbor discovery time.

Detection of Hidden nodes in sensor networks is addressed in [2]. Here propose a policy for determining the transmission power of every node, in order to guarantee that each node detects at least one of its neighbors using as little power as possible.

Detection of Hidden nodes is studied for general ad-hoc wireless networks. The authors propose a random HELLO protocol, inspired by ALOHA. Each node can be in one of two states: listening or talking. A node decides randomly when to initiate the transmission of a HELLO message. If its message does not collide with another HELLO, the node is considered to be discovered. The goal is to determine the HELLO transmission frequency, and the duration of the neighbor discovery process.

The sensor nodes are supposed to determine, for every time slot, whether to transmit HELLO, to listen, or to sleep. The optimal transition rate between the three states is determined using a priori knowledge of the maximum possible number of neighbors.

III. PROBLEM DEFINITION

In this discussion we assume that the network is a unit disk graph; namely, any pair of nodes that are within transmission range are neighboring nodes. Two nodes are said to be directly connected if they have discovered each other and are aware of each other's wake-up times. Two nodes are said to be connected if there is a path of directly connected nodes between them. A set of connected nodes is referred to as a segment. Consider a pair of neighboring nodes that belong to the same segment but are not aware that they have direct wireless connectivity. These two nodes can learn about their hidden wireless link using the following simple scheme, which uses two message types:1) SYNC messages for synchronization between all segment nodes, transmitted over known wireless links; 2) HELLO messages for detecting new neighbors.

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 wire-less links of the segment. Thus, it is guaranteed to be received by every segment node. By having all the nodes wake up "almost at the same time" for a short period, we can ensure that every wire- less link between the segment's members will be detected.

Scheme 2 (Detecting a Hidden Link outside a Segment): Node wakes up randomly, every second on the average, for a fixed period of time. During this time, it broadcasts several HELLO messages and listens for possible HELLO messages sent by new neighbors.

A random wake-up approach is used to minimize the possibility of repeating collisions between the HELLO messages of nodes in the same segment. Theoretically, another scheme may be used, where segment nodes coordinate their wake-up periods to prevent collisions and speed up the discovery of hidden nodes. However, finding an efficient time division is equivalent to the well-known node-coloring problem, which is NP-hard and also cannot be well approximated. Since the time period during which every node wakes up is very short, and the HELLO transmission time is even shorter, the probability that two neigh- boring nodes will be active at the same time is practically 0. In the rare case of collisions, CSMA/CD can be used to schedule retransmissions.

IV.DETECTION OF HIDDEN NODES IN-DEGREE SEGMENT

As already we know that 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 , denoted by degs (u). In this section, we present methods that can be used by node in the Normal (continuous neighbor discovery) state to estimate this value. Node is assumed to not yet be connected to the segment, and it is in the Init (initial neighbor discovery) state. Three methods are presented.

1) Node measures the average in-segment degree of the segment nodes and uses this number as an estimate of the in-segment degree of . The average in-segment degree 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 e of the sending node, which is known due to Scheme 1. We assume that the segment size is big enough for the received value to be of neighbors considered equal to the expected number of every node.
2) Node n discovers, using Scheme 1, the number of its in-segment neighbors, and views this number as an estimate of . This approach is expected to yield better results than the previous one when the degrees of neighboring nodes are correlated.

3) Node n uses the average in-segment degree of its segment's nodes and its own in-segment degree to estimate the number of node's neighbors. This approach is expected to yield the best results if the correlation between the in-segment degrees of neighboring nodes is known. An interesting special case is when the in-segment nodes are uniformly distributed.

V. EFFICIENT NEIGHBOR DISCOVERY ALGORITHM

In this section we present 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 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 TI 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.

The Neighbor Discovery Algorithm uses a protocol called Neighbor Discovery Protocol.

Neighbor Discovery Protocol:

The Neighbor Discovery protocol is detailed in a document put forward by the Internet Society, an organization responsible for developing standards to be used in the global Internet. This document is called"RFC2461" and is currently in its draft standard, as is RFC2460which specifies theIPv6 protocol. Draft standard implies that working implementations are available and that they have been thoroughly testate Neighbor Discovery protocol manages interactions between nodes via message exchanges.

These messages provide the data necessary for the processes of host auto configuration and packet transmission on a local link, Host auto configuration involves

- Parameter Discovery
- Address Auto configuration
- Duplicate Address Detection
- Router Discovery

- Prefix Discovery
- Address Resolution
- Neighbor Unreachability Detection
- Redirect

VI. SIMULATION

A simulator tool is used to imitate selected parts of the behavior of the real world and is normally used as a tool for research and development. Depending on the intended usage of the simulator, different parts of the real-world system are modeled and imitated. The parts that are modeled can also be of varying abstraction level. Earlier simulators especially designed for WSN imitates the wireless media and the constraints nodes in the network but currently sensor network simulators have a detailed model of the wireless media including effects of obstacles between nodes, while other simulators have a more abstract model.

This application is a simulation of the wireless sensor network described hereinabove. The network may be deployed based on a wide range of parameters: network size (number of nodes), communications distance, energy costs for transmitting and receiving packets, etc. The network can then be used to simulate the detection of vectors traveling across the sensor network field. In this simulation, when a vector trips the sensor of a network node, the node generates a data packet and sends it to a downstream network node. The packets are routed appropriately until they reach a sensor within the "uplink zone" (the right side of the map, designated with a striped pattern.) Each node also simulates an energy store, which is depleted by sending receiving packets, and by detecting vectors. Since the nodes have finite energy, they will eventually power down and drop out of the communications network, causing network failure.

The application has the ability to run successive tests on a network and report the mean network lifetime across 1,000 trials. The network routing parameters can be tweaked to allow testing of different network configurations.

VII. CONCLUSION

Here we exposed a new problem in wireless sensor networks, referred to as Detection of Hidden Nodes in Sensor Networks. We argue that Continuous neighbor Discovery is crucial even if the sensor nodes are static. if the nodes in a connected segment work together on this task, hidden nodes are guaranteed to be detected within a certain probability ‘p’ and a certain time period ‘t’ with reduced expanded on the detection.

we showed that my scheme works well if every node connected to a segment estimates the in-segment degree of its possible hidden neighbors’. I presented a continuous neighbor discovery algorithm that determines the frequency with which every node enters the HELLO period. We simulated a sensor network to analyze our algorithm and showed that when the hidden nodes are uniformly distributed in the area.

VIII. REFRENCES

[1] S. Vasudevan, J. Kurose, and D. Towsley, "On neighbor discovery in wireless networks with directional antennas," in Proc. IEEE INFOCOM, 2005, vol. 4, pp. 2502–2512.

[2]R. Madan and S.Lall, "An energy-optimal algorithm for neighbor discovery in wireless sensor networks," Mobile 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 flexible neighbor discovery in ad hoc wireless networks," in Proc. 2nd ACMMobiHoc, New York, 2001, pp. 137–145.

[4] D. Baker and A. Ephremides, "The architectural organization of a mobile radio network via a distributed algorithm," IEEE Trans. Commun., vol. COMM-29, no. 11, pp. 1694–1701, Nov. 1981.

[5] A. Keshavarzian, E. Uysal-Biyikoglu, F. Herrmann, and A. Manjeshwar,"Energy-efficient link assessment in wireless sensor networks,"in Proc. IEEE INFOCOM, 2004, vol. 3, pp. 1751–1761.

[6] E. B. Hamida, G. Chelius, and E. Fleury, "Revisiting neighbor discovery with interferences consideration," in Proc. PE-WASUN, 2006,pp. 74–81.

[7] S. A. Borbash, "Design considerations in wireless sensor networks,"Ph.D. dissertation, ISR, August 2004.

[8] G. Alonso, E. Kranakis, R. Wattenhofer, and P. Widmayer, "Probabilistic protocols for node discovery in ad-hoc, single broadcast channel networks," in Proc. IPDPS, 2003, p. 218.

[9] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," RFC 3561, Jul. 2003.

[10] J. Haartsen, Bluetooth Baseband Specification v. 1.0.

[11] T. Salonidis, P. Bhagwat, L. Tassiulas, and R. O. LaMaire, "Distributed topology construction of Bluetooth personal area networks," in Proc.IEEE INFOCOM, 2001, pp. 1577–1586.

[12] IEEE 802.15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs), IEEE 802.15 WPAN Task Group 4 (TG4), 2006.

[13] P. Dutta and D. Culler, "Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications," in Proc. 6th ACMSenSys,New York, 2008,pp.71–84.