Distributed Neighbor Discovery in Ad Hoc Networks Using Directional Antennas *

Rully Adrian Santosa, Bu-Sung Lee, Chai Kiat Yeo and Teck Meng Lim School of Computer Engineering, Nanyang Technological University, Singapore {rull0001,ebslee,asckyeo,limtm}@ntu.edu.sg

Abstract

In this paper, we propose a distributed neighbor discovery algorithm in ad hoc networks using directional antennas. A newly joining node obtains the information about its neighbors in both its omnidirectional and directional range by querying a subset of its omnidirectional neighbors. The main contribution of the proposed algorithm is that it does not require all neighbor nodes to be involved in the discovery process. Simulation results show that the algorithm performance closely tracks the theoretical analysis and only requires a fraction of the neighbors to be involved in the discovery process.

1. Introduction

Neighbor discovery plays an important role in wireless ad hoc networking. Knowledge about the existence and the location of the neighbors facilitates a node in making a better routing decision. Neighbor discovery becomes more significant when directional antennas are used.

Directional antennas offer many benefits over omnidirectional antennas. Using directional antennas, the transmission can be focused only at the direction of the receiver, minimizing the interference with other nodes within the omnidirectional transmission range. The transmission range is also extended, allowing nodes outside the omnidirectional transmission range to be reached in one hop. The usual transmission range for directional antenna can be as high as twice the transmission range for omnidirectional antenna.

Traditional neighbor discovery algorithm operates by periodically transmitting beacon (HELLO message) omnidirectionally. The beacon contains the identifier of the node and possibly the location of the node. All nodes that receive the beacon will update their respective neighbor lists to include the sender of the beacon. Using traditional neighbor discovery algorithm, a node is able to obtain a complete knowledge of the neighbors inside its omnidirectional transmission range. However, with the use of directional antenna, there is a need to know the neighbors outside the omnidirectional transmission range to benefit from the extended transmission range of directional antennas.

In this paper, we propose a distributed approach to neighbor discovery using directional antennas. The main idea is to minimize the number of neighbor nodes involved in the discovery process. Some neighbor nodes are chosen as representatives of other neighbor nodes in the discovery process. The decision to choose the representative nodes is done independently at each node in a distributed manner.

2. Related work

Some neighbor discovery algorithms that exploit the benefit of directional antennas have been proposed. They are implemented at the MAC layer. A polling-based neighbor discovery algorithm is proposed in [6]. A node sends polling message in a specific direction and waits for replies from the neighbors in that direction. Polling can be contention-based or contention-free. The space is successively scanned until the entire space has been covered. The node will have complete knowledge of its neighbors after it finishes polling in all directions.

Similarly in [7], nodes are synchronized with each other and periodically perform neighbor discovery. During neighbor discovery, a node can either be in scanning or in listening mode. The selection of the mode can be stochastic or deterministic. A scanning node scans the space in a predetermined sequence and transmits advertisements. A listening node listens in the directions opposite to the scanning directions and discovers a scanning node when both nodes form beams towards each other.

In [8], a node transmits its identifier and location in random direction with probability p and listens for transmissions with probability 1 - p. A neighbor node is discovered upon receiving a successful transmission from that neighbor node. Depending on p, the algorithm needs varying

^{*}This project is supported by A*STAR SERC grant: Pervasive Network for Seamless Mobile Communication (Project number: 042 101 0012).



Figure 1. Sets of neighbors of node *i*.

time duration to obtain complete knowledge of the neighbors. Neighbor discovery can be synchronous (time-slotted) or asynchronous (non time-slotted). An enhancement based on gossip is also proposed. In this algorithm, a node also transmits the identifiers and locations of the neighbors it has accumulated so far.

An integrated neighbor discovery and MAC protocol is proposed in [3]. Time is segmented into consecutive frames and nodes are synchronized with each other. Neighbor discovery is performed in the search slot at the start of each frame. The search slot is divided into two sub-slots. In the first sub-slot, each node randomly chooses whether to transmit pilot tone or to receive, together with the direction for transmission/reception. If the node chooses to transmit pilot tone in the first sub-slot, it will receive in the second subslot, and vice versa. Two nodes discover each other upon successful exchange of pilot tones.

In [5], a synchronized directional neighbor discovery is proposed. It aims to get two nodes that do not know of each other's existence to beamform towards each other simultaneously. A direction is chosen based on the synchronized time. Each node alternates randomly between transmitting in that direction and listening in the opposite direction. When the transmit beamform of a node aligns with the receive beamform of another node, the two nodes can discover each other. A complete knowledge of the neighbors is available after one such cycle covering all directions.

3. Proposed algorithm

In typical antennas with omnidirectional transmission range of d, the directional transmission range is usually much longer [4][2]. For simplicity of analysis, we assume a directional transmission range of 2d. Let N_i denote the set of neighbors of node i in its directional transmission range 2d. Let N_i^o denote the set of neighbors of node i in its omnidirectional transmission range d. N_i^o is a subset of N_i . Let $N_i^d = N_i - N_i^o$, that is the set of neighbors of node i in its directional transmission range 2d, but outside its omnidirectional transmission range d. Fig. 1 shows the neighbors sets of node i. We assume an ideal communication channel.

Consider the case when a new node joins the system and begins the neighbor discovery process. All existing nodes

in the system are assumed to have a complete knowledge of the neighbors in their respective directional transmission ranges 2d. Given a new node x joining the system, x can obtain complete knowledge of the neighbors in its directional transmission range 2d (i.e. N_x) using the information provided by only a subset of the neighbors in its omnidirectional transmission range d (i.e. N_x°).

When x joins the system, it broadcasts information about itself via omnidirectional transmission. This information includes the node identifier and its location. The broadcast message from x is received by all nodes inside the omnidirectional transmission range of x (i.e. N_x^o). Upon receiving the broadcast message, all nodes in N_x^o update their respective neighbor lists with information about x.

Note that each node i in N_x^o has complete knowledge of other nodes in N_x^o . In addition, each node i in N_x^o also has partial knowledge of some nodes in N_x^d . These relationships are shown in Fig. 2. Hence, some nodes in N_x^o can be chosen as representative nodes to provide x with the information about the neighbors in its directional transmission range 2d (i.e. N_x). Let S denote the set of the representative nodes. Each node in S sends the neighbor information to x using directional transmission.

The nodes included in set S should be chosen such that

• the union of the directional transmission range 2d of all nodes in S covers the directional transmission range 2d of node x, i.e.

$$N_x \subset \bigcup_{i \in S} N_i$$

• the number of nodes in S is minimized. Intuitively, the best case for the number of nodes in S is 1, i.e. when the representative node is located extremely close to x.

As it has first been assumed that all nodes in N_x^o have complete knowledge of other nodes in N_x^o , they can compute and determine independently which nodes should be included in the set S in order to meet the two goals above. Although each node in N_x^o computes the set S independently, they will agree on the same set S because each node runs the same selection algorithm to determine the set S.

After set S has been determined, each node checks whether it belongs to set S, i.e. whether it is one of the representative nodes that is responsible for sending the neighbor information to x. If it does not belong to set S, it ceases to be further involved in the neighbor discovery process of node x. Hence it does not need to transmit any information to node x. However, if it belongs to set S, it will be required to transmit some neighbor information to node x.

The neighbor information a representative node needs to send to node x depends on the order of the node in S. Let i_k denote the k^{th} representative node in S. The first representative node, i_1 , sends the information about all neighbor

nodes it knows that also fall in the directional transmission range 2d of node x. The second representative node, i_2 , sends the information about all neighbor nodes it knows that also fall in the directional transmission range 2d of node x, minus all neighbor nodes that has been covered by the first representative node, i_1 . Similarly, the third representative node, i_3 , sends the information about all neighbor nodes it knows that also fall in the directional transmission range 2dof node x, minus all neighbor nodes that has been covered by the first and the second representative nodes, i_1 and i_2 . In general, the k^{th} representative node sends the information about all neighbors it knows that also fall in the directional transmission range 2d of node x, minus all neighbor nodes that has been covered by the previous representative nodes:

$$\left(N_{i_k}\bigcap N_x\right) - \bigcup_{j=1}^{k-1} N_{i_j}$$

Note that the information about a particular neighbor node of x will be sent once by exactly one representative node. The neighbor information sent by different nodes in S does not overlap. Thus there is no redundant neighbor information being sent, and hence less traffic is generated. The calculation to determine which node information to send by a particular representative node is based on the locations of the representative node itself, the previous representative nodes, node x and the neighbor nodes.

A simple selection algorithm based on heuristic is used to choose the representative nodes to be included in set Sand to determine the order of the representative nodes in S. To choose the first representative node i_1 in S, pick the node in N_x^o closest to x. By doing this, N_x will overlap substantially with N_{i_1} . A tie between two nodes with the same distance from x is resolved using the additional rule; i.e. choose the first node in clockwise direction starting from the north. By choosing the first representative node, i_1 , a minimum of 68.4% of x's directional transmission range 2d area would have been covered. The worst case scenario occurs when i_1 lies on the border of x's omnidirectional range, i.e. the distance from x to i_1 equals to d (see Fig. 2). Note that i_1 is able to provide x with the information about all nodes in N_x^o , plus the nodes in $N_x^d \cap N_{i_1}$:

$$N_x \cap N_{i_1} = (N_x^o \cup N_x^d) \cap N_{i_1}$$

= $(N_x^o \cap N_{i_1}) \cup (N_x^d \cap N_{i_1})$
= $N_x^o \cup (N_x^d \cap N_{i_1})$

To cover the remaining area $N_x - N_{i_1}$, the second representative node, i_2 , is chosen from N_x^o to send additional information to x. i_2 is the next node in N_x^o closest to x. As mentioned before, i_2 needs to send only the information about the nodes it knows which have not been covered by i_1 . This step is repeated by choosing i_3 , i_4 and so on



Figure 2. Worst case scenario in choosing the first representative node, i_1 .

Algorithm 1 Noc	le x joining	the system
-----------------	--------------	------------

- 1: Let N_x be the neighbor list of node x
- N_x ← Ø
 2: broadcast request message using omnidirectional transmission to initiate neighbor discovery process
- wait for response messages containing neighbor information from the representative nodes
- 4: for all neighbor information R contained in the received response messages do
- 5: $N_x \leftarrow N_x \cup R$ 6: end for

until no more node in N_x^o is able to send additional neighbor information to x that has not been covered by previous representative nodes.

The pseudocode of the proposed neighbor discovery algorithm is presented in Algorithm 1 and Algorithm 2.

To keep the neighbor information updated, every node x periodically requests for updates by broadcasting the information about itself using omnidirectional transmission to the nodes in N_x^o , similar to the procedure performed when the node joins the system for the first time. Upon receiving the broadcast message, all nodes in N_x^o can update their respective neighbor lists with the information about x. The representative nodes in S will then send the update information to x using directional transmission.

When node x leaves the system, other nodes in N_x^o will not receive any more update requests from x. After a timeout period, the nodes in N_x^o will drop the information about x from their respective neighbor lists. Alternatively, the leaving node x can explicitly broadcast a leave message to other nodes in N_x^o before leaving the system. This will speed up the leaving procedure without having to wait for the timeout period.

3.1. Example

Consider the scenario in Fig. 3. 7 nodes are currently active (nodes 1 to 7). A new node x joins the system and starts the neighbor discovery process. Firstly, x broadcasts the information about itself using omnidirectional transmission. Nodes 1, 2, 3 and 4 in the omnidirectional transmission.



Algorithm 2 Each node i in the omnidirectional transmission range of x

1: receive broadcast request message from x2: Let N_i be the neighbor list of node *i* $N_i \leftarrow N_i \cup \{x\}$ 3: $W \leftarrow (N_i \cup \{i\}) - \{x\}$ 4: for all $j \in W$ do if distance(x, j) > 2d then 5: $W \leftarrow W - \{j\}$ 6: end if 7: 8: end for 9: $Z \leftarrow W$ 10: sort all $j \in W$ in ascending order based on distance(x, j), then based on angle(x, j)11: for all $j \in W$ do 12: if j = i then 13: exit for 14: end if $Z \leftarrow Z - \{j\}$ 15: for all $k \in W, k \neq j$ do 16: if distance(j,k) <= 2d then 17: 18: $Z \leftarrow Z - \{k\}$ 19: end if 20end for 21: end for 22: if $Z \neq \emptyset$ then send Z to x using directional transmission 23: 24: else 25: do nothing 26: end if



Figure 3. Scenario of a new node x starting the neighbor discovery process.

sion range d of x receive the broadcast message and update their respective neighbor lists with information about x.

Next, nodes 1, 2, 3 and 4 compute and determine independently whether they are representative nodes who provide x with the neighbor information. Node 1 finds that it is closest to x among other nodes in the omnidirectional transmission range d of x. Hence it becomes the first representative node. Node 1 sends the information about all nodes in the omnidirectional transmission range d of x (nodes 1, 2, 3 and 4) plus information about a node in the directional transmission range 2d of x that it knows (node 5).

Node 2 finds that it is second closest to x. Node 2 has the information about all nodes in the omnidirectional transmission range d of x (nodes 1, 2, 3, and 4), plus information

about some nodes in the directional transmission range 2d of x (nodes 5 and 6). However, node 2 computes that information about nodes 1, 2, 3, 4, and 5 has been sent to x by node 1. Thus, node 2 only sends information about the node that has not been covered by node 1 (node 6).

Similarly, node 3 finds that it is the third closest to x. Node 3 has information about nodes 1, 2, 3, 4, 5 and 6, all of which have been covered by previous representative nodes (nodes 1 and 2). Thus, node 3 will not send any neighbor information to node x.

Finally, the last closest node to x is node 4. Among all nodes that are known to node 4, only node 7 has not been covered by previous representative nodes (nodes 1, 2 and 3). Thus, node 4 sends the information about node 7 to x.

At this point, no more nodes in the omnidirectional transmission range d of x have additional neighbor information that has not been covered by previous representative nodes.

4. Analysis

We focus our analysis on the coverage of the proposed neighbor discovery algorithm, i.e. how many percent of its neighbors a newly joining node can discover by performing the proposed neighbor discovery algorithm. When a new node joins the system and performs the proposed neighbor discovery algorithm, it may be able to obtain a complete or partial knowledge of the neighbors in its directional transmission range, depending on the number and the placement of neighbors residing in its omnidirectional transmission range. Having more neighbors in the omnidirectional transmission range leads to a higher chance of covering the whole directional transmission range, provided that the neighbors are distributed enough in the omnidirectional transmission range. Having some neighbors clustered in one spot within the omnidirectional transmission range is as good as having only one neighbor in the same spot.

We begin the analysis by finding how much of the directional transmission range is covered on average when there are n neighbors in the omnidirectional transmission range. We notice that calculating the area of intersecting circles purely by geometry is difficult. Therefore, we use the monte carlo method to approximate the expected coverage.

A large number of sample points (≈ 10000 points) are scattered randomly inside the directional transmission range. *n* neighbor nodes are placed randomly inside the omnidirectional transmission range. Then, each sample point is evaluated on whether it falls inside the directional transmission range of any neighbor nodes. If it is, the sample point is marked as covered. Otherwise, it is marked as not covered. After all sample points have been evaluated, the percentage of the covered sample points is calculated. The procedure is repeated a large number of times (≈ 10000 times) with different placements of sample points



Figure 4. Expected coverage c(n) against the number of neighbor nodes inside the omnidirectional transmission range.

and neighbor nodes. Averaging over all runs, the expected coverage when the number of neighbor nodes inside the omnidirectional transmission range is n can be found. We denote this expected coverage as c(n). We calculate the expected coverage c(n) for n from 0 to 20. The directional transmission ranges used are twice (2d) and four times (4d) the omnidirectional transmission range (d). The result is shown in Fig. 4. This result is used to obtain the expected coverage of the proposed neighbor discovery algorithm:

We assume the nodes are uniformly distributed in a large area. Let ρ denote the node density, i.e. the average number of nodes in a unit area. The expected number of nodes in the omnidirectional transmission range, denoted as λ , can be found as follows:

$$\lambda = \rho \pi d^2 \tag{1}$$

The probability of finding n nodes in the omnidirectional transmission range follows the Poisson distribution with mean λ :

$$p(n) = \frac{\lambda^n e^{-\lambda}}{n!} \tag{2}$$

Thus, the expected coverage of the proposed neighbor discovery algorithm, E[c(n)], performed by a newly joining node can be calculated as follows:

$$E[c(n)] = \sum_{k=0}^{\infty} p(k)c(k)$$
(3)

We compute the expected area not covered by the proposed algorithm, 1 - E[c(n)], for node density from 10 to 100 nodes/km². The omnidirectional transmission range d used is 200 m. The directional transmission ranges used are 2d = 400 m and 4d = 800 m the omnidirectional transmission range. Fig. 5 shows the result.

It can be seen from the graph that in a sparse population, a considerable percentage of the directional transmission



Figure 5. Expected area not covered by the proposed neighbor discovery algorithm against the node density.

range is not covered by the proposed neighbor discovery algorithm. However, as the population density increases, the expected not-covered area decreases rapidly. When the node density is greater than 50 nodes/km², the percentage of the not-covered area drops to less than 1% and hence becomes insignificant. We conclude that the proposed neighbor discovery algorithm works better in a dense population.

It is crucial for a newly joining node x to obtain as complete knowledge as possible about neighbors in its directional transmission range. This is due to the assumption used by the proposed algorithm that all existing nodes in the system have complete knowledge of neighbors in their respective directional transmission ranges. Thus, in order for the algorithm to be applicable for subsequent new joining nodes, the assumption has to be still valid after the neighbor discovery process of node x. Hence, node x shall obtain a complete knowledge of the neighbors in its directional transmission range, which can be accomplished in a dense population where the percentage of the not-covered area is considerably small.

5. Simulation

We perform simulation to evaluate the actual coverage of the algorithm and compare the result with the expected coverage derived in Section 4. n nodes are randomly scattered inside a square area of 1500 m × 1500 m according to the uniform distribution, such that the node density equals to ρ . We use the omnidirectional transmission range of 200 m, and directional transmission range of 400 m. The coverage simulation was performed using a custom-tailored C++ program. In addition, we repeat the simulation using QualNet Network Simulator 3.8 [1].

A newly joining node x is placed at a random position inside the square area. Then, the proposed neighbor discovery





Figure 6. Actual not-covered area obtained from the simulation vs expected theoretical not-covered area.

algorithm is started for node x. At the end of the neighbor discovery process, the number of undiscovered neighbors of x is compared with the number of actual neighbors of x.

We ran the simulation with different values of ρ , from 4 nodes/km² to 44 nodes/km². For each ρ , the simulation is repeated 1000 times. By averaging over all simulation runs for each ρ , we obtain the average not-covered area. The simulation results are compared with the theoretical analysis as shown in Fig. 6. It can be seen from the graphs that the simulation results closely track the theoretical analysis.

In addition, we also record the number of neighbor nodes inside the omnidirectional transmission range of xthat transmit neighbor information to x, i.e. the representative nodes of x. By averaging over all simulation runs for each ρ , we obtain the average number of representative nodes. Fig. 7 shows the result. It can be seen from the graph that the average number of representative nodes is lower than the average number of neighbor nodes inside the omnidirectional transmission range. The difference increases dramatically with node density. Thus, we conclude that the proposed neighbor discovery algorithm is more efficient since only a small subset of the neighbor nodes inside the omnidirectional transmission range are involved in the discovery process.

6. Conclusion

We propose a distributed neighbor discovery algorithm in ad hoc networks using directional antenna. A newly joining node obtains information about neighbors in both its omnidirectional and directional transmission ranges by querying a subset of its omnidirectional-range neighbors.

The proposed algorithm possesses a number of benefits. Firstly, it does not require all neighbor nodes to be involved in the discovery process. Only some nodes, chosen as repre-



Figure 7. Average number of representative nodes against the node density.

sentatives nodes, are required to send neighbor information to the newly joining node. Secondly, there is no overlapping of information being sent. Each representative node only sends the neighbor information that has not been covered by previous representative nodes. As a consequence, the neighbor discovery process is faster and more efficient.

Future works will investigate the performance of the algorithm under dynamic network condition, where nodes join and leave the system dynamically.

References

- [1] Scalable network technologies. [Online]. Available: http://www.scalable-networks.com/.
- [2] A. Alexiou and M. Haardt. Smart antenna technologies for future wireless systems: Trends and challenges. *IEEE Communications Magazine*, pages 90–97, Sept. 2004.
- [3] G. Jakllari, W. Luo, and S. V. Krishnamurthy. An integrated neighbor discovery and MAC protocol for ad hoc networks using directional antennas. In *Proc. IEEE WoWMoM*, pages 11–21, June 2005.
- [4] P. H. Lehne and M. Pettersen. An overview of smart antenna technology for mobile communications systems. *IEEE Communications Surveys*, 2(4):2–13, Fourth Quarter 1999.
- [5] R. Ramanathan, J. Redi, C. Santivanez, D. Wiggins, and S. Polit. Ad hoc networking with directional antennas: A complete system solution. *IEEE J. Select. Areas Commun.*, 23:496–506, Mar. 2005.
- [6] T. Ren, I. Koutsopoulos, and L. Tassiulas. Efficient media access protocols for wireless LANs with smart antennas. In *Wireless Communications and Networking*, 2003, volume 2, pages 1286–1290, Mar. 2003.
- [7] M. E. Steenstrup. Neighbor discovery among mobile nodes equipped with smart antennas. In *Proc. 3rd Scandinavian Workshop on Wireless Ad-hoc Networks*, May 2003.
- [8] S. Vasudevan, J. Kurose, and D. Towsley. On neighbor discovery in wireless networks with directional antennas. In *Proc. IEEE INFOCOM 2005*, volume 4, pages 2502–2512, Mar. 2005.

