Agricultural Monitoring System

Ye Li, Raimundo Daniel Becerra, Tian Zheng
Department of Electrical and Computer Engineering
Technical University of Munich

***Abstract*—The self-organizing network is a collection of wireless nodes, forming a temporary network, without the help of any established infrastructure or centralized management. In such an environment, due to the limited wireless transmission range of each node, a node may need to forward data packets to its destination with the help of other nodes. In this report, an agricultural monitoring system based on this kind of network is presented. Specifically, this project's network organization is based on the Ad hoc On-Demand Distance Vector (AODV) routing protocol. When the location of the node moves, the protocol can quickly adapt to routing changes, with almost no additional overhead when the node moves relatively little. The node collects local temperature and brightness information and sends it to the terminal through this routing protocol. The temperature and brightness information at the required node location can be viewed on the terminal, thereby realizing the expected agricultural information communication system.**

I. Introduction

Wireless network nodes and sensors are becoming widely available. Although nodes and sensors can also work in wired networks, in many scenarios it is more desirable to be able to communicate without fixed wired infrastructure because the necessary infrastructure is not economically feasible or physically possible. For example, a group of students may want to interact with each other during a lecture, to share some files. In this case, a collection of nodes with wireless network interfaces will form a temporary network without any established infrastructure or centralized management. This type of wireless network is called an ad hoc network.

If there are only two nodes close together in the ad hoc network, there is no need for routing protocols or routing decisions, they can communicate directly. However, in many ad hoc networks, two hosts that want to communicate may not be within each other's wireless transmission range. In this case, if other nodes are also participating in the self-organizing network, they can forward data packets for them, so they can communicate. For example, in the network shown in Figure 1, node C is not within node A's wireless transmission range (represented by the circle around A), and node A is not within node C's wireless transmission range. If A and C want to exchange data packets, in this case, they can enlist node B to forward the data packets for them, because B is in the overlapping range between A's range and C's range. In reality, due to the inherent non-uniform propagation characteristics of wireless transmission and any or all involved hosts may move at any time, the routing problem in a true self-organizing network may be more complicated than shown in this example.

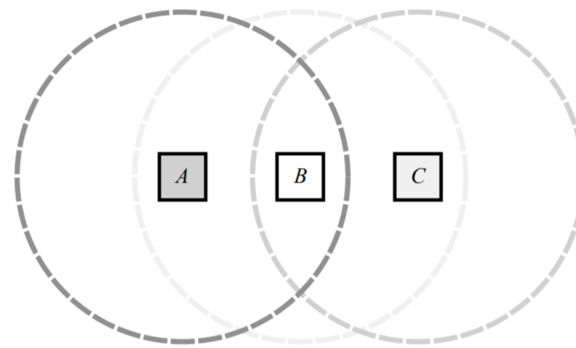

Fig. 1. A simple ad hoc network of three wireless nodes [5]

Path-based routing protocols in traditional wireless networks usually use distance vector or link state routing algorithms, both of which require each router to periodically broadcast routing advertisements. In distance vector routing [1], [2], each node broadcasts to each of its neighbour nodes its view of the distance to all hosts, and each node computes the shortest path to each host based on the information advertised by each of its neighbours. In link state routing [3], [4], each node instead broadcasts to all other nodes in the network its view of the status of each of its adjacent network links, and each node then computes the shortest distance to each host based on the complete picture of the network formed from the most recent link information from all nodes. [5]

This project uses the Ad hoc On-Demand Distance Vector (AODV) routing protocol instead. It offers quick adaptation to dynamic link conditions, low processing and memory overhead, low network utilization, and determines unicast routes to destinations within the ad hoc network. [6]

There are no regular router advertisements in the AODV protocol. By not sending advertisements and not needing to receive advertisements, mobile nodes also save battery power. On the other hand, because other mobile nodes are required to continue to think that these routes or network links are valid, in the distance vector and link state routing, the advertisements must be sent continuously even if there is no change in the network. When a node needs a route to another node, the AODV protocol will quickly and dynamically determine a route based on cached information and the results of the route discovery protocol. Further, it does not require nodes to maintain routes to destinations that are not in active communication. AODV allows mobile nodes to respond to link breakages and changes in network topology in a timely manner. Also, when links break, AODV causes the affected set of nodes to be notified so that they are able to invalidate the routes using the lost link. It uses destination sequence numbers to ensure loop freedom at all times, avoiding problems (such as ”counting to infinity”) associated with classical distance vector protocols.

Section 2 of this article details the design of the network and mobile nodes and sensors of this project. The basic operation of the Ad hoc On-Demand Distance Vector (AODV) routing protocol of this project is described in Section 3. In Section 4, conclusions and possible improvements will be presented.

II. Design

A. Network Design

There are a maximum of seven nodes in the network of this project at the same given time. One of these seven nodes is selected to act as a terminal for receiving data (the receiving node). The temperature and luminosity(brightness) at the other nodes (the sending nodes) can be observed intuitively through the GUI interface. Each sending node collects new local temperature and brightness data every twenty seconds, and uses the stored route or finds a new route to send to the receiving node. Every intermediate node forwards the data received from the last node, and add its own address at the end of the route vector in the data packet. Then it sends the new data packet forward. During the generation and forwarding of data, the LED will flash once green to show the process. The flash of yellow LED is to show that the node is now sending a request to its neighbourhood. Once the receiver node (here in the network, there is only one node that is set to be the receiver) successfully received the data packet, it will show by the flash of blue LED. Once the routing table has exceeded the setting timer, the node will flash once red LED to notify.

B. Nodes and Sensors Design

1) Temperature Sensor: The on-board CC2538 SoC has a built-in Temperature Sensor, which can be selected as an input to the Analog to Digital Coverter (ADC) for temperature measurements. In this way, temperature data can be collected from each sender nodes.

2) Brightness Sensor: The brightness sensor used in this project is an external sensor. The RE-Mote has available 2 ADC ports providing the normally used GND and VCC pins to connect external analogue sensors. Depending on the power operation requirement, the ADC1 (3.3V) or the ADC3 (5V) can be selected. The brightness sensor used in this project can use both the 3.3 and 5V ADC. The luminosity(brightness) can be obtained from the sensor using the following formula:

$$Luminosity(lux) = m * \frac{raw * V_{ref}}{4096} * 200 + b \qquad (1)$$

in which V_{ref} means the selected sensor power (ADC1: 3.3V; ADC2: 5V). The brightness sensor requires either a 3V or 5V phidget connection. The raw value is read directly from the sensor.

C. GUI Design

If a user needs to know the information in the network, they must press an ”Open” button to read the data in the USB.

What the user sees in the middle of the GUI interface is the topology of the network. The topology shows in real time which nodes have established routes. If the source node finds a route that can transmit data with higher quality, the original route will be cleared and replaced by the new route.

If the user chooses to view the data of ”ALL” node, then whenever the receiver receives the temperature and brightness data sent by any node, it will update the weather and brightness data in the two bar graphs on the right side of the GUI interface. If the user chooses to view only the data of the specified node, the display of the bar graph will be updated only when the receiver receives a new set of data from the selected source node.

In order to make the received data more obvious and intuitively displayed on the GUI interface, the bar graph is displayed in different colours according to the range of temperature and brightness values (for temperature(in Celsius): 0-20 with blue, 20-25 with green, 25-30 with orange, 30-50 with red; for brightness/luminosity: 0-50 with blue, 50-100 with green, 100-150 with orange, 150-200 with red)

The detailed data content received by the receiver node is displayed at the bottom of the GUI interface. The format of the data content is: DATA; data ID; source node address; target node address; Temperature: temperature data Luminance: brightness data; source node address, first hop node address, second hop node address, ..., fifth Hop node address; the number of hops passed between the source node and the target node. An address of zero means that no node has passed through [Fig. 2].

The last updated set of data will be stored in a .txt file, so that the next time the user opens the GUI interface, they can see the data received at each node last time.

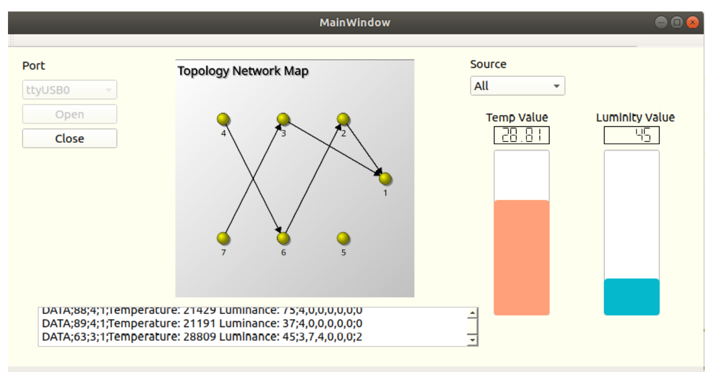

Fig. 2. GUI interface

III. Operations

The Ad hoc On-Demand Distance Vector (AODV) routing protocol, used on multi-hop Ad Hoc wireless networks, is a simple and efficient routing protocol. This protocol is an on-demand routing protocol. As long as the sender node of a communication connection has a valid route to the destination node, AODV does not play any role. When a route to a new destination is needed, or the old route is not valid any more, the node broadcasts a route request to find a route to the destination. A route can be determined when the route request reaches the destination itself, or when the route request reaches the node that has already formed a route to the same destination. The route is made available by unicasting a route reply back to the origination of the route request. Each node receiving the request caches a route back to the originator of the request, so that the route reply can be unicast along the route path to that originator.

AODV is a routing protocol, which deals with route table management. Route table information must be kept for a certain time period, to let the next data package with same destination in this time period go along this route. All states of the AODV routing protocol are ”soft states”, that is, the joining or leaving of any node does not have a great impact on the entire network. The loss of any state will not affect the correct operation of the routing protocol, unless it is part of the current valid route. Nodes monitor the link status of next hops in active routes. A destination becomes unreachable when a link breaks or is deactivated. When these conditions occur, the route is invalidated by operations involving the sequence number and marking the route table entry state as invalid. When a link break in a valid route is detected, a new route request will be broadcast to find a new route from the current node before the invalid one to the original destination node.

While waiting for the route discovery to complete, the node continues normal processing. It still sends and receives data packets with other nodes. Once a route is learned from route discovery, the route will be stored in the route table. If the original data packet is still within the expiration time, the node will cache the original data packet and transmit it according to the newly discovered route. If the original data packet exceeds its expiration time while waiting for the route to be discovered, the data packet will be discarded. The route table is associated with another specified route expiration time, after which the route will be deleted from the route table.

The following subsections describe how nodes generate Route Request (RREQ), Route Reply (RREP) messages to get the best route towards the destination, and how the data packets are handled in this project.

A. Route Discovery

AODV uses a broadcast route discovery mechanism, which relies on intermediate nodes to dynamically establish a route table for packet transmission. Every source node collects data every 20 seconds and needs a route to send the collected data to the destination, if there is no available route, it will broadcast a route request (RREQ) to all neighbours and wait for 15 seconds in the waiting table for a route. The RREQ packet contains the address of this node as source, the address of the destination node, and the number of each new request as the ID.

If the node which received a unique RREQ message is not the destination node defined in this RREQ message, there is also no existing route towards to that destination node, it should store the information in the discovery table of this node. Each discovery table line should contain the ID, source node address and destination node address from RREQ message, and sender node address of this RREQ message. The age in the discovery table indicates the maximum time to obtain a route to the destination. If it is reduced to zero, this discovery table line should be set as invalid. After establishing the discovery table, the node should forward the broadcast of RREQ message.

If the node which received a unique RREQ message is not the destination node defined in this RREQ message, but there is already an existing route for the same destination stored in this node, it will directly send a RREP message back to the sending node. This RREP message will contain the information of the previous existing route, adding one to the hop counter associated with it.

Each time the RREQ packet arrives at the destination node, the destination node will generate the route reply (RREP) packet and use the established discovery table to forward the RREP packet. The ID, source and destination node address of RREP packet stay the same as the ones in RREQ packet. The hop counter in the RREP packet is set to zero. Every time the RREP packet is forwarded through an intermediate node, one will be added to the hop count. The RSSI(Received signal strength indication) value in RREP packet is set to be 999 at the destination node.

Every time the RREP packet is forwarded, the average RSSI value should be calculated based on the current RSSI value and the hops in the RREP packet. The RSSI value is the criterium to select the best route. Once the new calculated average RSSI value is smaller than the old one, that means the data transmission qualification is better, the route table should then be updated. If the node that received the RREP packet is not the source, it should unicast a new RREP with new calculated average RSSI value and new hops to the next node recorded in the discovery table. The RREP packet will keep being forwarded, until it has reached the source node.

After the source node has received the RREP packet and generated the routing table, the sensed data at the source node in the waiting table is sent in accordance to the route table, one node after the other, if the maximum waiting time is not exceeded.

The route table of the node also stores a timer, the purpose is to clear the route that has not been used within a certain period of time. Once the 60 seconds specified by the timer is exceeded, the node's route is deemed invalid and a new RREQ needs to be sent to find a new route.

B. Route Maintenance

If the movement of the node has nothing to do with the currently active path, it will not affect the established route.

After the source node sends out RREQ, it receives the RREP and establishes the route table. If the row corresponding to the RREP in the source node's waiting table is still valid within the valid time, then the data packet in the waiting table will be called out and be sent through the established route table and the data packet will then be stored in the waiting acknowledgement table. The function of waiting acknowledgement table is to actively discover whether the intermediate nodes that make up the route in the network can successfully transmit a data packet or drop out of the network.

The waiting acknowledgement table has a valid time limit, which means that the data in the table will wait 10 seconds to receive the acknowledgement information returned by the next hop node that should receive the data packet. If the node receives the acknowledgement message within the effective time limit, it means that the data package sent by the node is successfully received by the next hop, which means the normal working ability of the receiving node. If the node does not receive the acknowledgement message within the effective time limit, it means that the data packet sent by the node has not been successfully received, and the node that plans to receive the data is deemed to be unable to work normally. In order to ensure that the original data packet will not be lost, the node will re-use itself as the source node, store the original data in its waiting table, and rebroadcast RREQ to all neighbours to find a new route to the destination node.

IV. Conclusion

In short, the routing protocol based on AODV implemented in this project has the following characteristics and advantages:

- Based on the traditional distance vector routing mechanism, the idea of AODV routing protocol is simple and easy to understand.
- AODV routing protocol support intermediate node response, which enables the source node to quickly obtain routing, effectively reducing the number of broadcasts.
- The node only stores the required routes, reducing memory requirements and unnecessary duplication.
- Uses IDs to avoid routing loops.
- AODV routing protocol can handle the scalability of the network.
- Link failure can be reported immediately so that routing can be re-established.
- Inactive routes will be deleted after a certain period of time.
- Intermediate nodes return the RREP directly if they have a route with same destination node, so that the route of this node could be shared. This saves broadcasts and reduces time delay.

References

[1] Hedrick C. Routing information protocol[R]. RFC 1058, Rutgers University, 1988.

[2] Kurose J, Ross K. Computer networks: A top down approach featuring the internet[J]. Peorsoim Addison Wesley, 2010.

[3] ISO D P. 10589[J]. Intermediate System to Intermediate System Intra-Domain Routing Exchange Protocol for Use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service (ISO 8473), 1990.

[4] McQuillan J, Richer I, Rosen E. The new routing algorithm for the ARPANET[J]. IEEE transactions on communications, 1980, 28(5): 711-719.

[5] Johnson D B, Maltz D A. Dynamic source routing in ad hoc wireless networks[M]//Mobile computing. Springer, Boston, MA, 1996: 153-181.

[6] Perkins C E, Royer E M. Ad-hoc on-demand distance vector routing[C]//Proceedings WMCSA'99. Second IEEE Workshop on Mobile Computing Systems and Applications. IEEE, 1999: 90-100.