A Reliable Anycast Protocol Based on Recovery Point in ZigBee Networks

Tsung-Long Chen, Chih-Yung Cheng, Shyr-Kuen Chen, Pi-Chung Wang

Department of Computer Science and Engineering

National Chung Hsing University Taichung, Taiwan 402

{s9556039, s9556058, phd9609, pcwang}@cs.nchu.edu.tw

***Abstract*—ZigBee network supports low data-rates; low power consumption and simple route. Aim at searching service; it usually relies on message broadcasting which tends to result in large traffic overhead. Other major challenge is unstable forwarding path. This work presents an anycast scheme for searching service in ZigBee network. That has the capability to choose the best one of servers in an anycast group as a destination, and reduce the volume of query messages as well as the reply messages. It also reduces the query latency and increases the accuracy of service discovery. In addition to basic anycasting, our scheme also increases the reliability of packet transmission by providing Recovery Point (RP). The RP scheme keeps copies of data packets of the source for recovering lost packets for its downstream node. The experimental results and demonstrate that our scheme is efficient and feasibile for ZigBee network.**

***Index Terms*—anycasting; recovery point; ZigBee.**

I. INTRODUCTION

There are a multitude of wireless standards, like Bluetooth and WiFi. That address mid to high data rates for services, such as voice, PC LANs and video. However, the wireless network has not been standardized to meet the unique needs of sensors and control devices. Sensors and controls do not need high bandwidth, but low latency, very low energy consumption for long battery life and large device arrays. ZigBee uses direct sequence spread spectrum (DSSS) modulation in mixed mesh, star, and peer-to-peer topologies (including cluster-free) to deliver a reliable data service with optional acknowledgments. The radio range is a nominal 10m which differs from popular implementations that normally use a single-hop range of up to 100m per node line of sight. ZigBee employs 64-bit IEEE addresses and shorter 16-bit ones for local addressing which accommodates networks with thousands of nodes. ZigBee is currently being built into millions of day-to-day devices for monitoring and controlling the real world environment. It is created to address the need for connecting a large number of devices (up to 65,000) in a network, running on batteries to last for many years. The initial markets for ZigBee include home, building and industrial automation, remote healthcare, and smart metering.

In the anycast mechanism, service providers are assigned a single anycast address within an anycast group. When a client sends packets to an anycast address, routers will attempt to deliver the packets to the closest server which matches the anycast address. Figure 1 illustrates the anycast packet flow. Three servers are configured with an anycast address "S" and located in different areas of the network. When any client want to search anycast server, it send a request message on the network, the routing system automatically delivers the request packets to the closest destination server which matches the anycast address. In 1993, the Internet Engineering Task Force (IETF) defines the basic role of IP anycast in RFC 1546 [4] as "*the host transmits a datagram to an anycast address and the Internetwork is responsible for providing best-effort delivery of the datagram to at least one, and preferably only one.*" Several years later, the

anycast address portion of the IPv6 addressing architecture has been defined in RFC 2373 [1] . In this work, we propose an anycasting scheme for ZigBee networks. In this scheme, an anycast tree is established and several nodes are selected as gateways (outgoing path greater than two) to reduce the control overhead. When a client node sends out a service request message to the anycast tree, the client is responded by its nearest or best anycast server. The responses from the other service providers will be discarded by the control-gate in gateway nodes. Therefore, our anycast scheme reduces the control overhead and is suitable for large-scale wireless sensor networks.

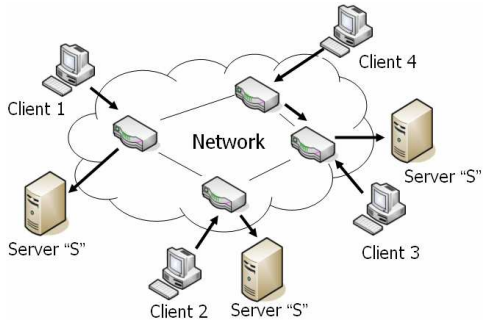

Fig. 1 Anycast Packet Flow

Although our anycast can reduce the control overhead, but had a major challenge is unstable forwarding path in the ZigBee networks. Thus, we present a recovery point scheme to increase the reliability of packet transmission, the retransmit packet work will hand over to the nearest RP node by receiver, and the sender needn't retransmit the lost packets.

The rest of the paper is organized as follows. In Section II the related works are presented. Section III gives a description of the system architecture. Section IV presents the experiment environment and the simulation result. Finally, Section V concludes this work.

II. RELATED WORK

In this section, we describe one of the existing anycast schemes in wireless sensor networks and present several different approaches of reliable protocols with recovery scheme.

(1) Anycast in WSN

Anycast also has applications in wireless sensor networks, where it could be used for distributed controlling or data gathering. Wireless sensor networks, a novel paradigm in distributed wireless communication technology, has been proposed for using in various applications including military and environmental monitoring. These networks consist of small sensor nodes that can be ran on battery power, have limited memory and processing power, and are capable for wireless communication. The nodes collect data by sensing the environment, process it locally, and then send the information back to the user. For this purpose, a protocol called Sink-based Anycast Routing Protocol (SARP) for Ad-hoc Wireless Sensor Networks was proposed [2] . The goal of this protocol is to reduce power and bandwidth consumption while packets delivery. The idea behind this protocol is that the data is delivered to the closest sink, so there is no precise destination for a packet.

(2) Reliable Protocols with Recovery Scheme

In this section, we introduce two reliable protocols with recovery scheme. These protocols include ACK-based protocols, NAK-based protocols.

1) ACK-based Protocol [5] : support for end-to-end data control. Data packets are sent from sender to the receivers via connection path. ACKs from the receivers are sent back to the sender along the same path. The sender must keep each data packets is stored in a history buffer, until the sender receiver ACK from receiver. The ACK support retransmission control can be either sender-driven, where the retransmission timepiece is managed at the sender (if the sender doesn't receive ACK within a certain time) or receiver-driven, where the retransmission timepiece is managed at the receivers (if receiver detects a missing message, send a negative ACK to sender). Although ACK-based Protocol provided guarantee and reliability, but will increase control overhead to process acknowledgment message from receivers.
2) NAK-based Protocol [3] : improve the ACK-based protocol implosion problem, this protocol only sends non-acknowledgment (NAK) to the sender when a retransmission is necessary. Since the sender only receives

feedback from the receivers when packets are lost, the sender is unable to ascertain when data can safely be released from memory. In order to ensure reliability, an infinite buffer space would be required.

The two type of protocol has its own advantages and limitations. ACK-based protocols provide reliability and low memory requirement and low implementation complexity, but suffer from the ACK implosion problem. NAK-based protocols alleviate the ACK implosion problem but require an infinite memory size or other mechanisms to ensure reliability, the implementation complexity is high than ACK-based. We present anycast protocol with recovery scheme will adopt ACK-based protocol.

III. SYSTEM ARCHITECTURE

In this section, we describe our system framework. Include construction of an anycast tree topology, anycast routing algorithms, and recovery scheme.

(1) Tree Topology and Neighbor Table Construction

Each device executes the following operations to discover and join an existing Wireless Personal Area Network (WPAN). These operations are listed as follows: i) search for available WPANs; ii) select the WPAN to join; iii) start the association procedure with the Personal Area Network (PAN) coordinator or with another Full-Function Device (FFD) device, which has already joined the WPAN. The discovery of available WPANs is performed by scanning channels and by searching available coordinators. We adopt passive scan for nodes-forming tree network automatically. For example, when those nodes being placed one hop away from the PAN coordinator receive their first beacon from the PAN coordinator, they initiate the association process. After a successful association, if those nodes are not leaf nodes, they will act as a coordinator and start transmitting beacons to those nodes placed at the next level in the tree. At the same time, those intermediate node (Router) constructs neighbor table which includes its parent and children nodes, as shown in Table I. PAN ID indicates Network ID, devices make the decision on what radio networks to join based on their PAN ID. ShortAddress indicates the network address for each node. DeviceType can be categorized in three devices: ‘0’ indicates the coordinator; ‘1’ indicates the router; and ‘2’ indicates the end device. The relationship can be deduced as a node’s parent (node 0000) by going upwards or children (node 1430) by going downwards. When all leaf nodes are associated, the network-forming phase is complete.

Table I: The Neighbor Tables of Nodes

PAN ID	Short Address	DeviceType	Relationship
12345	0000	0	0
12345	1430	2	1

(2) Anycast routing algorithm

We propose a novel routing protocol called AAODV (Anycast AODV) to reduce the control overhead. There is an example is shown in Figure 2 and the detailed algorithm is described as follows.

1) When a sender (node 1) requests for a service, it directly floods the request service (SREQ) message to its parent (node 3). The SREQ message also includes the identifier of the service instance.

2) When a router/forward (node 3) receives a service request message (SREQ) message, it will copy a SREQ message and forward this SREQ message to the other outgoing paths (node 0, 4, 9) according to its neighbor table. IF a router node’s outgoing path more than 2 will becomes a control-gate node. In addition, a router also creates a flag, *ControlGate*, to decide which reply message should be returned to the sender node. Each *ControlGate* is associated with a service instance identifier. Initially, the value of *ControlGate* is set to **FALSE**.

3) When a service (node 2, 5, 7, 8, 10, 11) receives a SREQ message, it will reply a SREP message to its parent (node 4, 6, 9) on the reverse path of SREQ message. The SREP message also indicates the identifier of the corresponding service instance.

4) When a router node receives a service reply (SREP) message, it checks its *ControlGate of* the corresponding service instance. If it is

TRUE, it means that the node has already received and forwarded a SREP message to the sender node. Therefore, it discards the subsequently received SREP message. Otherwise, the SREP message is sent to its parent node based on the reverse path of SREQ message until the SREP message reaches the source node. After sending the SREP message, the value of *ControlGate* in the router node is set to **TRUE** to avoid transmitting any further SREP messages for the same service instance.

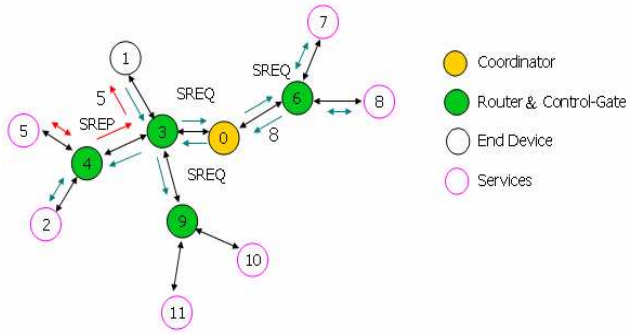

Fig. 2 Anycast AODV Scheme

(3) Maintain of Neighbor Table

Entries in the neighbor table are created when a node joins an existing network. When a joining node requests a NLMENETWORK-DISCOVERY, it receives response beacons from nodes which have already joined. The newly joined node also stores its neighbor information which is contained in the beacon packets. When a node leaves network, it will send NLME-LEAVE indication message to its neighbors. Therefore, the nodes receiving this message could remove the entry of leaving node from their neighbor tables. Since the information on the neighbor table is updated every time a device receives any frame from the some neighbor node, the information of the neighbor table can be kept up-to-date all the time.

(4) RP Establishment and Retransmission Scheme

In order to increase the reliability of anycast, we propose a recovery point scheme based on the control-gate node. There is an example show in Figure 3.

1) First, the sender (node 1) send request message to searching closest service provider (node 2) by anycast scheme, then send request messages (1-10) to this provider. After the establishing connection path, at the same time the control-gate node will become a recovery point in this path. The RP (node 3) is responsible for keeping data packets (1-6) sent from the sender node during the data delivery process.

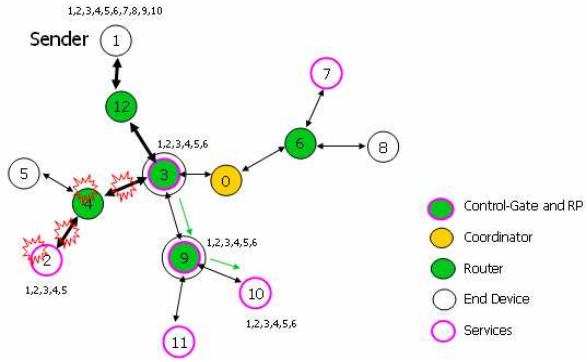

Fig. 3 RP Establishment and Retransmission Scheme

2) When the RP node doesn't receive ACK within a certain time, the RP node will be selection next service provider (node 10), then retransmit data packets (1-6) in memory to next service provider, at the same time establishment other RP (node 9) and keeping data packets in memory.

3) The RP node releases the data packets kept in memory only after positive all ACK message from receiver for data packets are received.

IV. EXPERIMENTAL RESULTS

This session evaluates the efficiency of the anycast scheme by OMNet++ [6]. Experiments are conducted for ZigBee Network, and implementation of the anycast services scheme in ZigBee networks.

(1) Simulation Results in ZigBee Networks.

Our scheme is a development from the IEEE 802.15.4 model in the INET framework. The architecture of the 802.15.4 model there are three sub models, traffic, MAC and PHY, each of which is a independent module and inherited from the basic C++ class cSimpleModule in OMNeT++. The simulation parameters are shown in Table II. In the simulation, two important performance metrics are evaluated:

1) Control overhead: the total number of forward packets and reply packets.

2) Average Hop Count: the average hop count from the sending services request packet to receiving reply service per request.

Table II: Simulation Parameters

Parameters	Value
Number of nodes	50
-Coord	1
-Router (Control-Gate)	39
-End Devices (Services Provider)	10
Routing	AAODV
PlaygroundSize	1000 * 600

Case 1: Number of Nodes-Services

In the first simulation, we evaluate the performance of our scheme by varying the number of nodes (from 30 to 50) with different number of service providers (from 5 to 15). As shown in Figure 4, when the number of nodes increases, the number of control overhead increases significantly in the traditional flooding mechanism. However, it is not the case for our scheme. The numerical results show that our scheme only needs about half control packets as compared to the flooding mechanism. In addition, the control overhead is less relevant to the number of service providers since our scheme only needs the closest service provider. Therefore, our scheme has superior scalability as compared to the flooding mechanism.

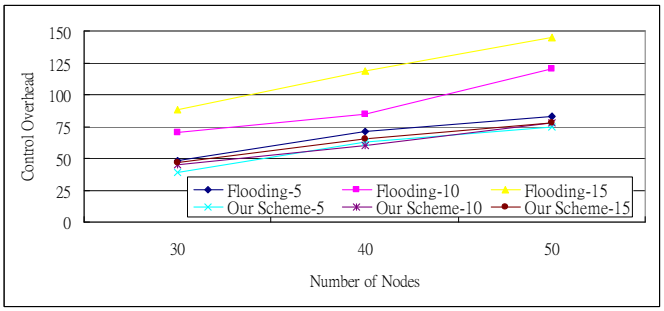

Fig. 4 Control Overhead vs. Number of Nodes

Case 2: Number of Services

Next, we measure the average round trip hop count for each service request. As shown in Figure 5, the average hop count is about 3-7 hops for our scheme, whereas the average hop count is about 8-9 hops for the flooding scheme. Because we use the anycast algorithm to find the closest service provider, the average hop count decreases as the number of service providers increases. As a result, our scheme could shorten the latency of service requests while maintaining the control overhead in the same level.

Fig. 5 Average Hop Count vs. Number of Services

In the last simulation, we compare the number of control packets for various numbers of service providers in the circumstance with link or service provider failure. In which case, our recovery point scheme will search for a new service provider by using the Anycast AODV algorithm for packet retransmission. As shown in Figure 6, the simulation results of our scheme (Our Scheme with RP) show that the control overhead of our scheme is consistently lower than the traditional flooding scheme with a retransmission mechanism (Flooding with Retransmission). Figure 6 shows that both schemes had high control overhead when there are only a few service providers, which might be caused by the long distance between the service provider and the sender. As the number of service providers increases, our scheme could keep the control overhead in the same level, which cannot be achieved by the flooding with retransmission scheme.

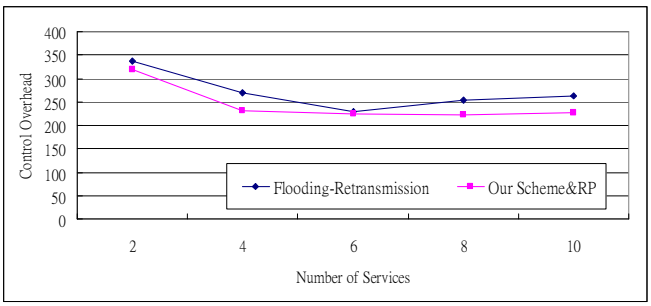

Fig. 6 Control overhead for different number of service providers with link and service provider failure.

(2) AAODV Implementation

We implement our AAODV scheme with the ZigBee modules from Jennic [7]. We use six modules to emulate a small environment for AAODV, as shown in Figure 7. When a source (node C) sends a service request message, it directly floods service request message to the tree-based network topology. With our AAODV scheme, only the modules of the closest node (E1) send service reply messages to the source. The experimental results in Figure 8 show that our scheme could save 23% of the control overhead in the traditional flooding scheme, and in Figure 9 show that our scheme could save 30%(average of ten simulation results) of the control overhead in the without RP scheme.

Fig. 7 Emulate a small environment for AAODV s

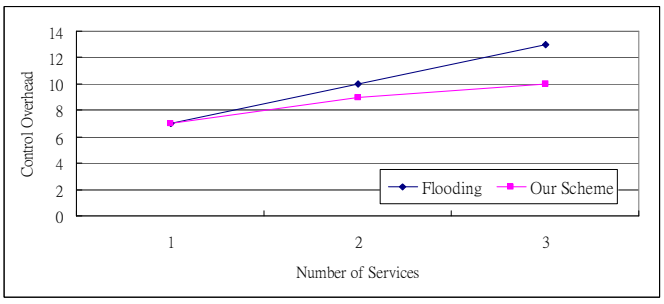

Fig. 8 Control Overhead vs. Number of Services

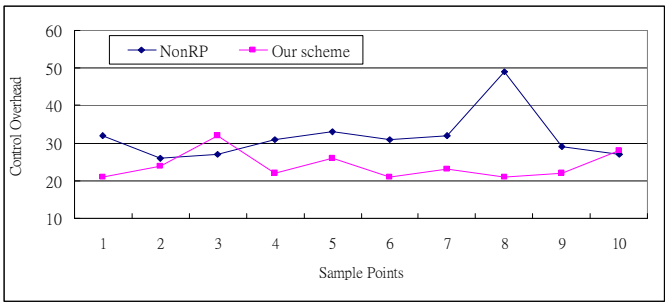

Fig. 9 Control Overhead for with RP and without RP

V. CONCLUSION

This work presents an efficient anycasting scheme with recovery scheme in the ZigBee network. We propose a technique to effectively anycast service in order to attain the reliability of deliver paths and to reduce the control overhead. The simulation results demonstrate that our anycasting scheme can be used to reduce the number of deliver packets and improve the performance of service discovery. In order to increase the reliability of anycast, we propose a recovery point scheme. The simulation results demonstrate that our recovery scheme is consistently lower than the flooding.

REFERENCES

[1] S. Deering and R. Hinden, "IP Version 6 Addressing Architecture," *IETF RFC 2373*, Jul. 1998.

[2] C. Intanagonwiwat and D. D. Lucia, "The Sink-based Anycast Routing Protocol for Ad Hoc Wireless Sensor Networks," *Technical Report 99-698, Department of Computer Science, University of Southern California*, Jan. 1999.

[3] A. Koifman, S. Zabele, "RAMP: A reliable adaptive multicast protocol, " *in Proceedings of IEEE INFOCOM*, March 1996, pp. 1442–1451.

[4] T. Mendez, W. Milliken, and C. Partridge, "Host Anycasting Service," *IETF RFC 1546*, Nov. 1993.

[5] S. Paul, K.K. Sabnani, J.C. Lin and S. Bhattacharyya, "Reliable multicast transport protocol RMTP," *IEEE Journal on Selected Areas in Communications 15* (1997) (3).

[6] "Omnet++ community site."
http://www.omnetpp.org/,Accessed 2005.

[7] Jennic, “JN5139 datasheet”,
http://www.jennic.com.