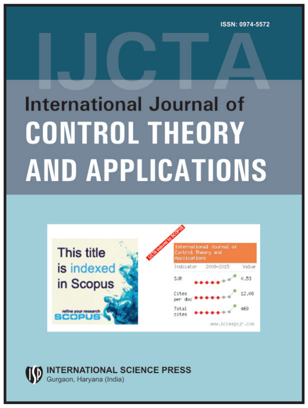


International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 10 • Number 24 • 2017

Energy Efficient Routing for Spherical Shaped Wireless Sensor Networks

Nancy Hopper.V[a], Murugaanandam.S[b] and Ganapathy.V[c]

[a]*M.Tech Student , Department of Information Technology, SRM University, Chennai, India, nancyhopper93@gmail.com*
[b]*Assistant Professor Department of Information Technology SRM University, Chennai, India, murugaanandam.s@ktr.srmuniv.ac.in.*
[c]*Professor Department of Information Technology SRM University, Chennai, India, ganapathy.v@ktr.srmuniv.ac.in*

Abstract: Broadcasting is an effective data dissemination mechanism in a route discovery. The broadcast storm problem is without knowing the route from source to destination, a mobile node blindly rebroadcasts the first received route request packets. For minimizing the routing overhead in WSN, neighbor coverage-based probabilistic rebroadcast protocol is proposed. To find the rebroadcast order, a rebroadcast delay is proposed. A reasonable rebroadcast probability is set by combining the additional coverage ratio and connectivity factor. This method is combined with the M2M network, which is in frame-by-frame basis. Each frame has four parts: notification period, contention only period, announcement period and Transmission only period**.** The BS broadcasts notification message to all devices for notifying the beginning of the contention during NP. The active devices will contend the channel during COP. BS broadcasts the beginning of the transmission period during AP. During the left over time of a frame, the devices that contend and succeed is allowed to transmit data packet. Advantages in this scheme: generates less rebroadcast traffic than the flooding and some other optimized scheme, mitigates the network collision and contention because of less redundant rebroadcast, so as to increase the packet delivery ratio and decrease the average end-to-end delay.

1. INTRODUCTION

The main scope of the proposed system is to make the wireless sensor network more energy efficient. For that first in network layer, the routing is made more efficient by avoiding rebroadcasting, find common neighbors, find uncovered neighbors, reduce time delay, calculate additional coverage ratio, calculate connectivity factor. Next step in improving efficiency is by reducing power or energy consumption in transferring data from one node to another. This is done by following frame by frame method in MAC layer. This method follows four steps. Notification period, contention only period, announcement period and transmission only period.

Notification message is sent to all devices by BS for contention during NP. Then during COP, the contention happens within all active devices. The transmission period will be broadcasted by BS during AP. During the left over time of a frame, the data packet can be transmitted by the devices that succeed in the contention that is mentioned as the TOP. A TDMA is one of the type communication type for the devices is provided by the TOP.

2. RELATED WORK

To find the best route in a network, the best mechanism is broadcasting. But, in high dynamic networks, the routing overhead caused by broadcasting will be very large. The rebroadcast is not cost effective and the network resource is also consumed more and this has been found by studies of the broadcasting protocol, both by analytical and experimental. High routing overhead and many problems such as redundant retransmissions, contentions, and collisions are caused due to broadcasting. The best solution to improve the routing performance is to optimize the broadcasting in discovering the route. Every node forwards a packet along with a probability is a gossip approach proposed by Haas [1]. Compared to the flooding, up to 35 percent overhead can be saved by gossip-based approach proposed by Haas [1]. The gossip-based approach is improved yet limited when the network density is increased or the traffic load is heavy. Based on combination of coverage area and neighbor confirmation, a probabilistic broadcasting scheme is developed by Kim [2]. To guarantee reachability, use the neighbor confirmation and to set the rebroadcast probability, the coverage area is used in this scheme. A neighbor knowledge scheme named Scalable Broadcast Algorithm (SBA) was introduced by Peng and Lu [3]. The packet rebroadcast would reach additional nodes is determined in this scheme. A Dynamic Probabilistic Route Discovery (DPR) scheme based on neighbor coverage is proposed by Abdulai [4]. According to the set of neighbors which are covered by the previous broadcast, each node determines the forwarding probability in this approach. The coverage ratio by the previous node is only considered, excluding the neighbors receiving the duplicate RREQ packet in this scheme. An AODV protocol with Directional Forward Routing (AODV-DFR) proposed by Chen [5] that takes the directional forwarding used by geographic routing into AODV protocol. The next-hop node for packet forwarding can be found automatically by this protocol if a route breaks. Two timer-based broadcast schemes are proposed by Keshavarz-Haddad [6]: Dynamic Reflector Broadcast (DRB) and Dynamic Connector-Connector Broadcast (DCCB). Entire reachability can be achieved over a lossless MAC layer by their schemes, and they are robustness for the environment of node failure and mobility. A Robust Broadcast Propagation (RBP) protocol is proposed by Stann [7]. To provide very perfect reliability for flooding in wireless networks, and a good efficiency is achieved by this protocol. A new perspective for broadcasting are presented: not to make a single broadcast more efficient instead make them more reliable, which means to enhance the overall performance of flooding by reducing the frequency of upper layer that causes flooding. The aim is to make the dissemination of neighbor knowledge much faster and a deterministic rebroadcast delay is set in our protocol.

3. PROPOSED WORK

Neighbor Coverage - Based Probabilistic Rebroadcast Protocol

The rebroadcast delay and rebroadcast probability of the proposed protocol are calculated. To estimate the rebroadcast delay, we use the upstream coverage ratio of an RREQ packet received from the previous node. To estimate the rebroadcast probability in our protocol, we utilize the additional coverage ratio of the RREQ packet and the connectivity factor, which requires that each node needs its 1-hop neighborhood information.

Uncovered Neighbor Set and Rebroadcast Delay

When an RREQ packet is sent from its previous node s to node ai, to calculate how many of its neighbors have not been included by the RREQ packet from r, it can use the neighbor list in the RREQ packet. The RREQ packet can include more additional neighbor nodes, if node ai rebroadcasts the RREQ packet only if node ai has more neighbors uncovered by the RREQ packet from r. To quantify this, we define the UnCovered Neighbors set $U(ai)$ of node ai as follows:

$$U(ai, Ps\,.\,id) = A(ai) - [A(ai) \cap A(r)] - \{r\}$$

where, $A(r)$ and $A(ai)$ are the neighbors sets of node s and ai, respectively. r is the node which sends an RREQ packet to node ai. We obtain the initial UCN set according from above formula. Node ai can receive the duplicate

RREQ packets from its neighbors due to broadcast characteristics of an RREQ packet. With the neighbor knowledge, node *ai* could adjust the U(*ai*). Avoid channel collisions to sufficiently exploit the neighbor knowledge. Each node should set a rebroadcast delay. The key success for this protocol is the selection of a correct delay because the dissemination of neighbor coverage knowledge is affected by the scheme used to find the delay time. The rebroadcast delay is calculated according to the neighbor list in the RREQ packet and its own neighbor list when a neighbor receives an RREQ packet. The rebroadcast delay Rd(*ai*) of node *ai* is defined as follows:

$$Rp(ai) = [1 - |A(s) \cap A(ai)|]/|A(s)|$$

$$Rd(ai) = MaxDelay \times Rp(ai)$$

where, R*p*(*ai*) is the delay ratio of node *ai*, and MaxDelay is a small constant delay. |.| is the number of elements in a set. The rebroadcast delay mentioned above is defined with the following reasons: First, to determine the node transmission order, the delay time is used. It should be disseminated as quickly as possible to effectively exploit the neighbor coverage knowledge. All its neighbors *ai*; I = 1, 2, ..., |A(*r*)| receive and process the RREQ packet when node *r* sends an RREQ packet. We assume that the largest number of common neighbors with node *r* is for node *nk*. According to above formula, node *nk* has the lowest delay. Since node *nk* has the largest number of common neighbors, there are more nodes to receive RREQ packet once node *nk* rebroadcasts the RREQ packet. Then, to adjust their UCN sets, there are more nodes which can exploit the neighbor knowledge. Of course, depending on its rebroadcast probability, it is decided whether node *nk* rebroadcasts the RREQ packet. The RREQ packet must not be rebroadcasted to more nodes, but to disseminate the neighbor coverage knowledge more quickly is the objective of this rebroadcast delay. The node can set its own timer, after determining the rebroadcast delay.

Neighbor Knowledge and Rebroadcast Probability

RREQ packets from the nodes which have lower rebroadcast delay, will be observed by the node which has a larger rebroadcast delay. For example, if neighbor node *aj* sends a duplicate RREQ packet to node *ai*, it knows that RREQ packet from *aj*, covers how many of its neighbors. Thus, the UCN set can be adjusted by node *ai* to the neighbor list in the RREQ packet from *aj*. Then, the U(*ai*) can be adjusted as follows:

$$U(ai, Ps\,.\,id) = U(ai, Ps\,.\,id) - [U(ai, Ps\,.\,id) \cap A(aj)]$$

The RREQ packet received from aj is discarded after adjusting the U(*ai*). The rebroadcast delay need not be adjusted because, to determine the order of disseminating neighbor coverage knowledge, the nodes which receive the same RREQ packet from the upstream node, the rebroadcast delay is used. Thus, it is determined by the neighbors of upstream nodes and its own. The node obtains the final UCN set, when the timer of the rebroadcast delay of node *ai* expires. The nodes that need to receive and process the RREQ packet are the nodes that belong to the final UCN set. Note that, the UCN set is not changed, which is the initial UCN set, if a node does not sense any duplicate RREQ packets from its neighborhood. We define the additional coverage ratio P*a*(*ai*) of node *ai* as

$$Pa(ai) = [|U(ai, Ps.\,id)|/|A(ai)|]$$

The ratio of the number of nodes that are additionally covered by this rebroadcast to the total number of neighbors of node *ai* is indicated by this metrics. Receiving and processing of the RREQ packet must be done for the nodes that are covered additionally. The rebroadcast probability should be set to be high since more nodes will be covered by this rebroadcast as Ra becomes bigger and more nodes need to receive and process the RREQ packet. The probability of the network being connected is approaching 1 as n increases, where n is the number of nodes in the network if each node connects to more than 5:1774 log n of its nearest neighbors, which was derived by Xue and Kumar. Then, for the connectivity metric of the network, we can use 5:1774 log n. Fc(*ai*) is the ratio of the number of nodes that need to receive the RREQ packet to the total number of neighbors of node *ni*. We have a heuristic formula to keep the probability of network connectivity approaching 1: $|A(ai)|. Fc(ai) \leq 5{:}1774 \log n$. Then, we define the minimum Fc(*ai*) as a connectivity factor, which is

$$Fc(ai) = Nc/|A(ai)|$$

where, $Nc = 5{:}1774 \log n$, and n is the number of nodes in the network. When $|A(ai)|$ is greater than Nc, Fc(*ai*) is less than 1. Then, only part of neighbors of node ai forward the RREQ packet could keep the network connectivity which means node *ai* is in the dense area of the network. And Fc(*ai*) is greater than 1 when $|A(ai)|$ is less than Nc. Then node *ai* should forward the RREQ packet in order to approach network connectivity which means node *ai* is in the sparse area of the network. We obtain the rebroadcast probability Pre(*ai*) of node *ai* by combining the additional coverage ratio and connectivity factor:

$Pre(ai) = Fc(ai). Pa(ai)$ where, if the Pre(*ai*) = is greater than 1, we set the Pre(*ai*) = to 1.

With the following reason, the above rebroadcast probability is defined. The relationship of the local node density and the overall network connectivity is not considered, although the parameter Pa reflects how many next-hop nodes should receive and process the RREQ packet. The parameter Fc increases the rebroadcast probability if the local node density is low, then increases the reliability of the NCPR in the sparse area. The parameter Fc could further decrease the rebroadcast probability if the local node density is high and then further increases the efficiency of NCPR in the dense area. The calculated rebroadcast probability Pre(*ai*) may be greater than 1, but it does not impact the behavior of the protocol. It just shows that the node must forward the RREQ packet since the local density of the node is so low. Then, to rebroadcast the RREQ packet, node *ai* is needed which is received from *s* with probability Pre(*ai*).

4. ALGORITHM DESCRIPTION

NCPR

Definitions:

RREQv: RREQ packet received from node *v*.

Pv.*id*: the unique identifier (*id*) of RREQv.

A(*ui*): Neighbor set of node *u*.

U(*u*, *x*): Uncovered neighbors set of node *u* for RREQ whose *id* is *x*.

Timer(*u*, *x*): Timer of node *u* for RREQ packet whose *id* is *x*. {Note that, in the actual implementation of NCPR protocol, every different RREQ needs a UCN set and a Timer.}

{Compute initial uncovered neighbors set U(*ai*, Ps.*id*) for RREQs:}

$U(ai, Ps.id) = A(ai) - [A(ai) \cap A(s)] - \{s\}$

{Compute the rebroadcast delay Rd(*ai*):}

$Rp(ai) = [1 - |A(s) \cap A(ai)|]/|A(s)|$

$Rd(ai) = MaxDelay \times Rp(ai)$

Set a Timer(*ai*, Ps.*id*) according to Rd(*ai*)

end if 9:

while *ai* receives a duplicate RREQj from *aj* before Timer(*ai*, Ps.*id*) expires do

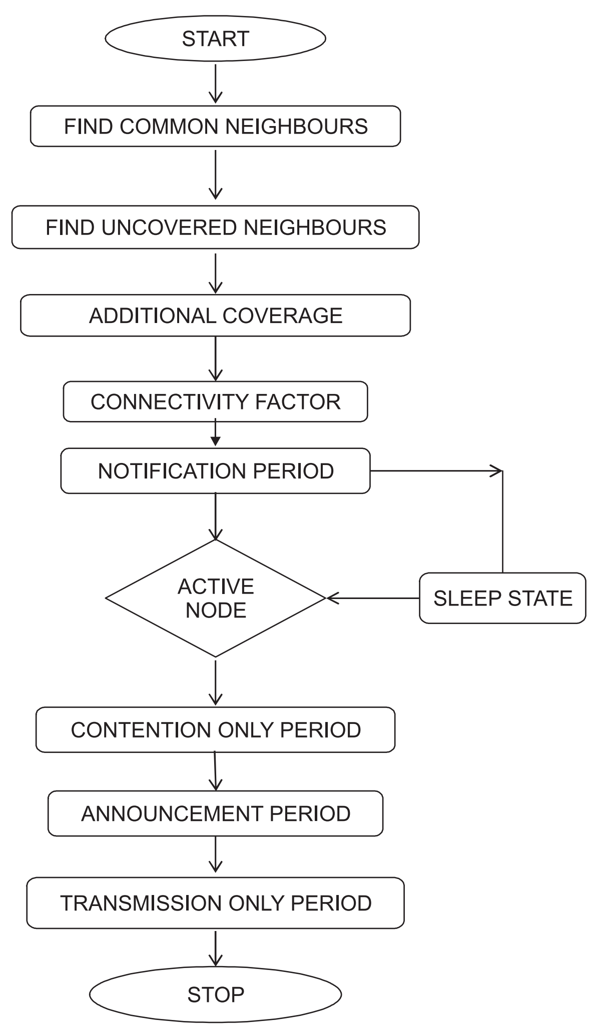
{Adjust U(*ai*, Ps.*id*)}

$U(ai, Ps.id) = U(ai, Ps.id) - [U(ai, Ps.id) \cap A(aj)]$

discard(RREQj)

```
end while
if Timer(ai, Ps.id) expires then
{Compute the rebroadcast probability Pre(ai)}
Pa(ai) = [|U(ai, Ps.id)|/|A(ai)|]
Fc(ai) = Nc/|A(ai)|
Pre(ai) = Fc(ai) . Pa(ai)
if Random(0,1) ≤ Pre(ai) then
broadcast(RREQs)
else
discard(RREQs)
end if
end if
```

5. WORK FLOW DIAGRAM



6. PROTOCOL IMPLEMENTATION AND PERFORMANCE EVALUATION

1. Protocol Implementation

To implement our proposed protocol, we modify the source code of AODV in NS-2 (v2.30). To obtain the neighbor information, the proposed NCPR protocol needs Hello packets and also needs to carry the neighbor list in the RREQ packet. Therefore, some techniques are used to reduce the overhead of Hello packets and neighbor list in the RREQ packet in our implementation, which are described as follows:

- The broadcasting packets such as RREQ and route error (RERR) can play a role of Hello packets, since a node sending any broadcasting packets can inform its neighbors that its existing. The node needs to send a Hello packet, only when the time elapsed from the last broadcasting packet (RREQ, RERR, or some other broadcasting packets) is greater than the value of HelloInterval. The value of HelloInterval is equal to that of the original AODV.
- Each node needs to monitor the variation of its neighbor table and maintain a cache of the neighbor list in the received RREQ packet in order to reduce the overhead of neighbor list in the RREQ packet. The size of neighbor list in the RREQ packet and following the num_neighbors is the dynamic neighbor list is represented by modifying the RREQ header of AODV, and adding a fixed field num_neighbors. The neighbor table of any node *ai* has the following three cases in the interval of two close followed sending or forwarding of RREQ packets:
  - then node *ai* sets the num_- neighbors to a positive integer, which is the number of listed neighbors, and then fills its complete neighbor list after the num_neighbors field in the RREQ packet if the neighbor table of node *ai* adds at least one new neighbor *nj*. It is because that node $_{nj}$ needs the complete neighbor list of node *ai* because node *nj* may not have cached the neighbor information of node *ai*;
  - node *ai* does not need to list its neighbors, and set the num_neighbors to 0 if the neighbor table of node *ai* does not vary. According to the value of num_neighbors in the received RREQ packet, the nodes which receive the RREQ packet from node *ai* can take their actions.
  - according to the neighbor list in the received RREQ packet, the node substitutes its neighbor cache of node *ai* if the num_neighbors is a positive integer.
  - the node updates its neighbor cache of node *ai* and deletes the deleted neighbors in the received RREQ packet if the num_neighbors is a negative integer.
  - the node does nothing if the num_neighbors is 0.

  This technique can reduce the overhead of neighbor list listed in the RREQ packet because of the two cases 2 and 3.

2. Simulation Environment

We compare the performance of the proposed NCPR protocol with some other protocols using the NS-2 simulator to find the performance of our protocol. One of the applications in this paper is route request in route discovery. We choose the Dynamic Probabilistic Route Discovery protocol which is an optimization scheme for reducing the overhead of RREQ packet incurred in route discovery and the conventional AODV protocol to compare the routing performance of the proposed NCPR protocol. Simulation parameters are as follows: For MAC layer protocol, the Distributed Coordination Function (DCF) of the IEEE 802.11 protocol is used. The radio channel model follows bit rate of 2 Mbps of a Lucent's Wave LAN and 250 meters of transmission range. We randomly choose different source-destination connections by considering constant bit rate (CBR) data traffic. With size is 512 bytes per second, every source sends four CBR packets. The random waypoint model in a field of 1;000 m – 1;000 m is used in the mobility model. With a random speed from a uniform distribution [1, max-speed], each node moves to a random selected destination in this mobility model. The node stops for a pause time interval and chooses a new destination and speed after the node reaches its destination. We set the max-speed to 5 m/s and set the pause time to 0 to reflect the network mobility. In the default implementation of AODV in NS-2, the MaxDelay which is used to determine the rebroadcast delay is set to 0.01 s, which is equal to the upper limit of the random jitter time of sending broadcast packets. Thus, in the route discovery it could not induce extra delay. For each simulation scenario, the simulation time is set to 300 seconds. The performance of routing protocols using the following performance metrics are evaluated:

- MAC collision rate: the collisions at the MAC layer per second results the average number of packets (including RREQ, route reply (RREP), RERR, and CBR data packets) dropped.
- Normalized routing overhead: the ratio of the total packet size of control packets (include RREQ, RREP, RERR, and Hello) to the total packet size of data packets delivered to the destinations. Each single hop is counted as one transmission for the control packets sent over multiple hops. We use the size of RREQ packets instead of the number of RREQ packets to preserve fairness, because the DPR and NCPR protocols include a neighbor list in the RREQ packet and its size is bigger than that of the original AODV.
- Packet delivery ratio: the ratio of the number of data packets successfully received by the CBR destinations to the number of data packets generated by the CBR sources.
- Average end-to-end delay: from source to destination node, the average delay of successfully delivered CBR packets. All possible delays from the CBR sources to destinations are included in it. There are three parts in the experiments, and we evaluate the impact of one of the following parameters on the performance of routing protocols in each part:
  - Number of nodes. In a fixed field, to evaluate the impact of different network density we vary the number of nodes from 50 to 300. We set the number of CBR connections to 15, and do not introduce extra packet loss in this part.
  - Number of CBR connections. To evaluate the impact of different traffic load, the number of randomly chosen CBR connections from 10 to 20 will be varied with a fixed packet rate. We set the number of nodes to 150 in this part, and also do not introduce extra packet loss.
  - Random packet loss rate. To introduce packet loss to evaluate the impact of random packet loss, we use the Error Model provided in the NS-2 simulator. The packet loss rate whose range is from 0 to 0.1 is uniformly distributed. We set the number of nodes to 150 and set the number of connections to 15 in this part.

3. Performance with Varied Number of Nodes

The data and control packets share the same physical channel in the IEEE 802.11 protocol. The massive redundant rebroadcast causes many collisions and interference in the conventional AODV protocol, which leads to excessive packets drop. This will be more severe with an increase in the number of nodes. The number of retransmissions in MAC layer is affected by packet drops in MAC layer and will also affect the packet delivery ratio of CBR packets in the application layer. To reduce the redundant rebroadcast and packet drop caused by collisions, routing performance has to be improved which is very important. The NCPR protocol reduces the MAC collision rate by about 92.8 percent on the average compared to conventional AODV protocol. The MAC collision rate is reduced by about 61.6 percent under the same network conditions when the DPR protocol is compared to NCPR protocol. Thus, the NCPR protocol could improve the routing performance is the main reason. In dense network, the NCPR protocol can significantly reduce the routing overhead. The NCPR protocol reduces the number of RREQ packets more even though the NCPR protocol increases the packet size of RREQ packets. Then, the RREQ traffic is still reduced. Additionally, the statistics of normalized routing overhead includes Hello traffic for fairness. The improvement of normalized routing overhead is considerable since the NCPR protocol still yields the best performance. The overhead is reduced by about 45.9 percent in the NCPR protocol on average compared to the conventional AODV protocol. The overhead is reduced by about 30.8 percent when the NCPR protocol is compared to the DPR protocol under the same network conditions. The NCPR protocol reduces overhead by about 74.9 and 49.1 percent when compared to the AODV and DPR protocols, respectively when network is dense. The NCPR protocol is the most efficient among the three protocols can be understood.

The packet delivery ratio can be increased by the NCPR protocol because it significantly reduces the number of collisions. So, because of collision, it reduces the number of packet drops. The packet delivery ratio is improved by about 11.9 percent in the NCPR protocol on average when compared with the conventional AODV protocol. The NCPR protocol improves the packet delivery ratio by about 3.7 percent when compared with the DPR protocol in the same situation. The NCPR protocol increases the packet delivery ratio about 21.8 and 6.3 percent when compared with the AODV and DPR protocols, respectively when network is dense. The average end-to-end delay of CBR packets received at the destinations with increasing network density is measured. Due to a decrease in the number of redundant rebroadcasting packets, the NCPR protocol decreases the average end-to-end delay. Because of the redundant rebroadcast increases delay, it causes too many collisions and interference, which not only leads to excessive packet drops, yet will increases the number of retransmissions in MAC layer so as to increase the delay and the back off timer in MAC layer is increased due to too many channel contentions, so as to increase the delay. Thus, the delay can be decreased by reducing the redundant rebroadcast. The end-to-end delay is reduced by about 60.8 percent in the NCPR protocol on average when compared with the conventional AODV protocol. To replace the random delay in the AODV protocol, a rebroadcast delay based on coverage ratio is used by the NCPR protocol. The MaxDelay in the NCPR protocol is equal to the upper limit random delay in the AODV protocol, so the NCPR protocol does not cause extra delay cost.

4. Performance with Varied Number of CBR Connections

As the number of CBR connections increases, the physical channel will be busier and then the collision of the MAC layer will be more severe because the data and control packets share the same physical channel in the IEEE 802.11 protocol. Load balance is not considered by DPR and NCPR protocols, but to reduce the packet drops caused by collisions, they can reduce the redundant rebroadcast and alleviate the channel congestion. The NCPR protocol reduces the MAC collision rate by about 95.2 percent on the average by comparing with the conventional AODV protocol.

The NCPR protocol reduces more MAC collision rate than the DPR protocol as network density increases. But, the NCPR reduce nearly the same scale of MAC collision rate than the DPR protocol in the same node density and different traffic load. Therefore, the NCPR protocol can also improve the routing performance at different traffic load. Both the DPR and NCPR protocols have more routing overhead than the conventional AODV protocol at very light traffic load (10 CBR connections). This is because that the extra overhead is added by Hello packets and neighbor list in the RREQ packet. The routing overhead of the conventional AODV protocol significantly increases as the traffic load increases, but the overhead of the DPR and NCPR protocols are relatively smooth. Both the DPR and NCPR protocols by contrast, decrease the routing overhead. The packet drops of the conventional AODV protocol without any optimization for redundant rebroadcast are more severe as the traffic load increases. The DPR and NCPR protocols increase the packet delivery ratio compared to the conventional AODV protocol, because they significantly reduce the number of collisions and then reduce the number of packet drops caused by collisions. There is a significant increase in the end-to-end delay of the conventional AODV protocol with the increase of traffic load, which is the same as the MAC collision rate and routing overhead. By reducing the redundant rebroadcast, DPR and NCPR protocols alleviate the channel congestion and reduce the retransmissions at MAC layer when the traffic load is heavy, thus, both of them reduce the end-to-end delay.

7. A SCALABLE HYBRID MAC PROTOCOL DESIGNOPERATION OF BS AND DEVICES

M2M Hierarchical network is deployed internet, Gate Way, Base station, devices and sensors. Basic IoT setup is executed in the network layer to establish M2M communication.

Mac Layer

Time frame is divided into following frames,

1. Notification period,
2. Contention only period,
3. Announcement period,
4. Transmission only period

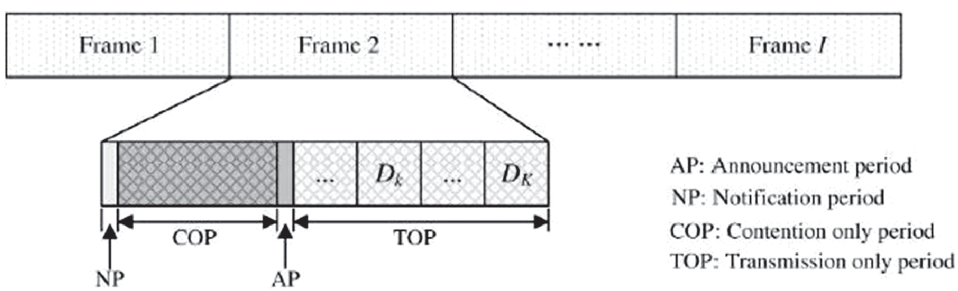


Figure 1: Frame Structure of M2M

Consider the operation of the M2M network on a frame-by-frame basis. Each frame is composed of four portions as depicted in Figure 1.

1. **Notification period:** Nodes broadcast notification message to all devices in the network to find number of devices to contend for data slot. To transmit data, devices receives the notification message and check for the data. Prepare to contend the slot, if it has data, otherwise goes to sleep state. Estimating the optimal contending parameter such as contention probability, duration and incremental indicator during this notification after the nodes identifies the packet arrival rate of each active device. To save the energy, the devices that does not participate in the data transmission go to the sleep state

2. **Contention only period:** Using the persistent problem and based on their own contention problem estimated in notification period, the devices that need to contend the slot will perform the contention process. The transmission request is to the nearby node by these devices. When the device sends the TranREQ, using the persistent problem and based on their own contention problem estimated in notification period. Collision problem is estimated if more devices contend the same slot. The acknowledge timer is started once the nodes receives this Tran-REQ message to compute the optimal contention period and then send the ACK Message. The device stops sending the Request message once the device receives the ACK Message and waits for the AP frame. The slot index and time period of slot required for the data communication is in ACK Message. Using TDMA setup to be used in TOP, the total slot is divided equally for all devices.

3. **Announcement period:** To all devices, nodes initiate the announcement messages. In TOP, devices check for the successful contention when they receive this announcement and will start sending own data.

   Remaining nodes go to the sleep state.

4. **TOP:** The devices switch the state to transmission mode and wake up from the sleep state those who are ready to transmit the data. The allocated TDMA slot for the own by the BS is checked. The packet duration which need to be completed in the current slot itself is validated. It performs the transmission need to be filled in this slot, else moves to the next frame to transmit the data if it does not fit in the slot.

8. RESULT

Packet Delivery Ratio: The packet delivery ratio is improved by about 11.9 percent in the NCPR protocol on average when compared with the conventional AODV protocol.

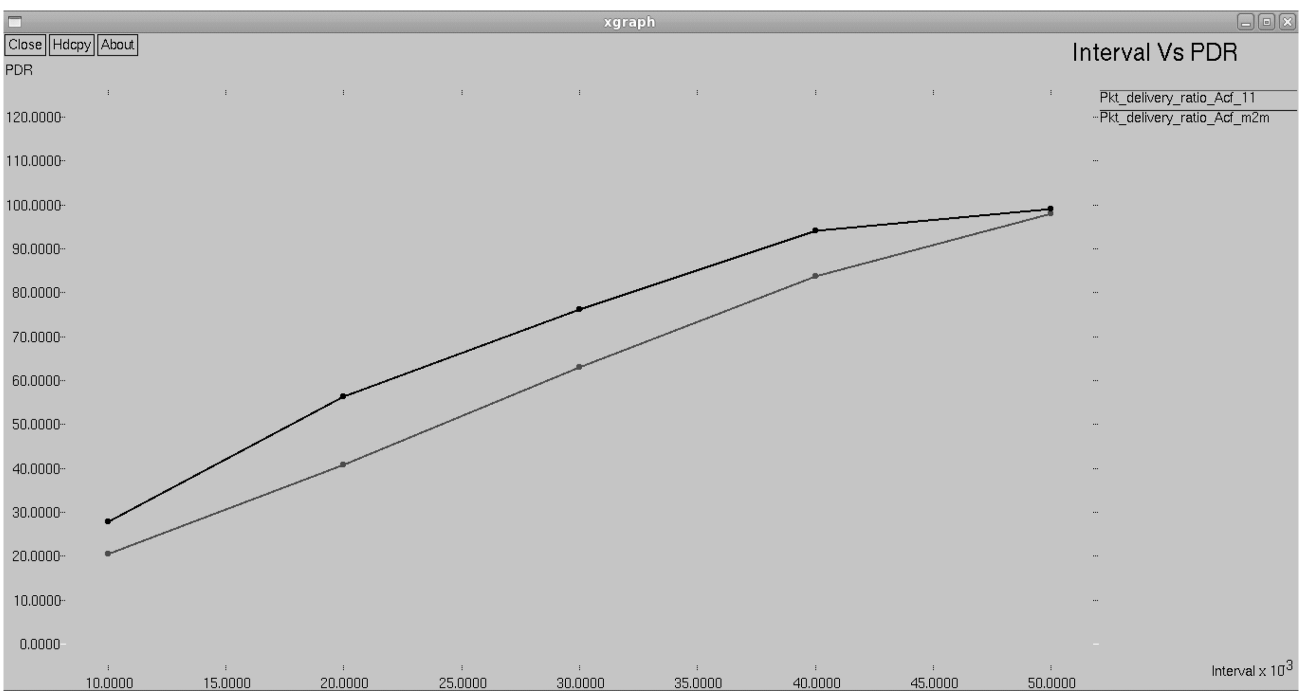


Figure 2: Interval Vs PDR

Delay: The NCPR protocol increases the packet delivery ratio about 21.8 and 6.3 percent when compared with the AODV and DPR protocols, respectively when network is dense.

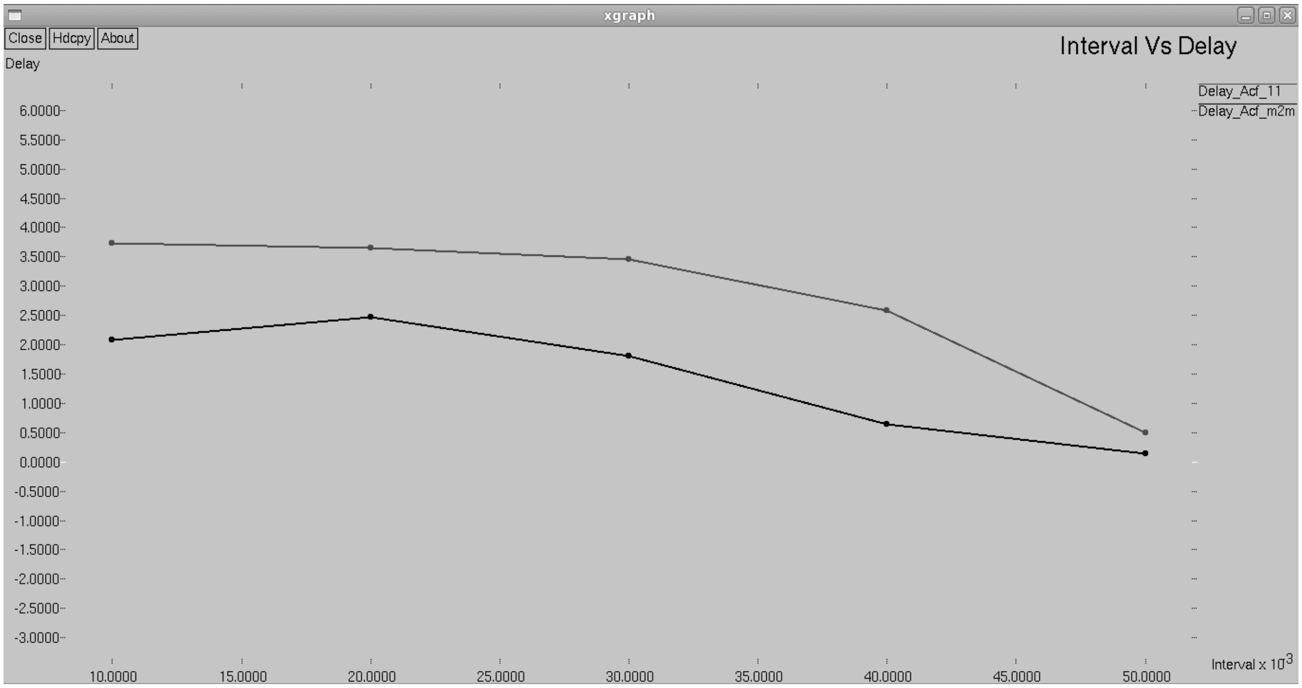


Figure 3: Interval Vs Delay

Dropping_ratio: To reduce the redundant rebroadcast and packet drop caused by collisions, routing performance has to be improved which is very important.

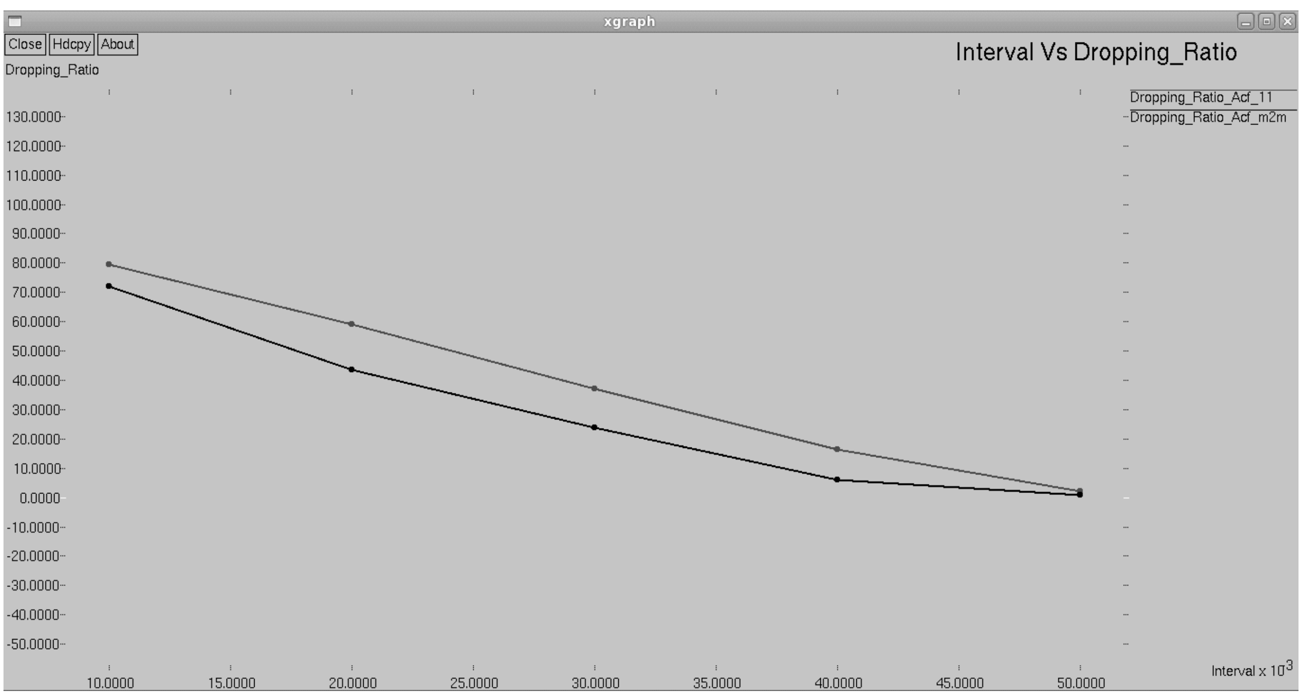


Figure 4: Interval Vs Dropping Ratio

Average_Energy_Consumption:

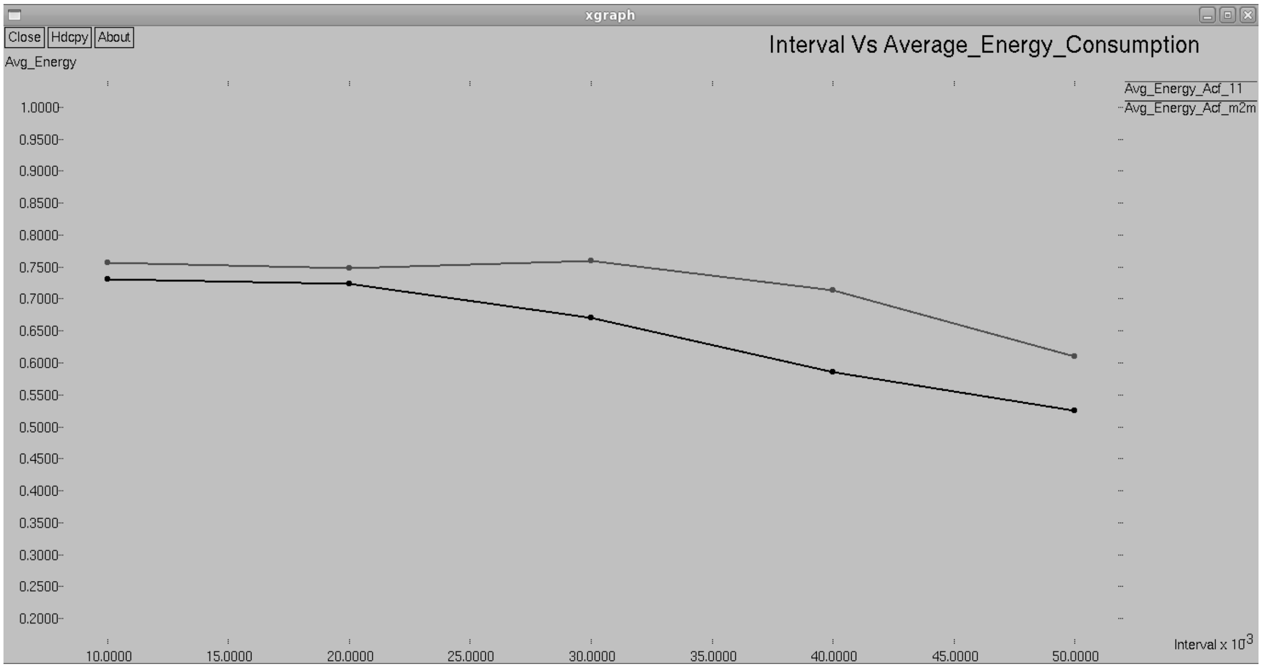


Figure 5: Interval Vs Average_Energy_Consumption

Throughput:

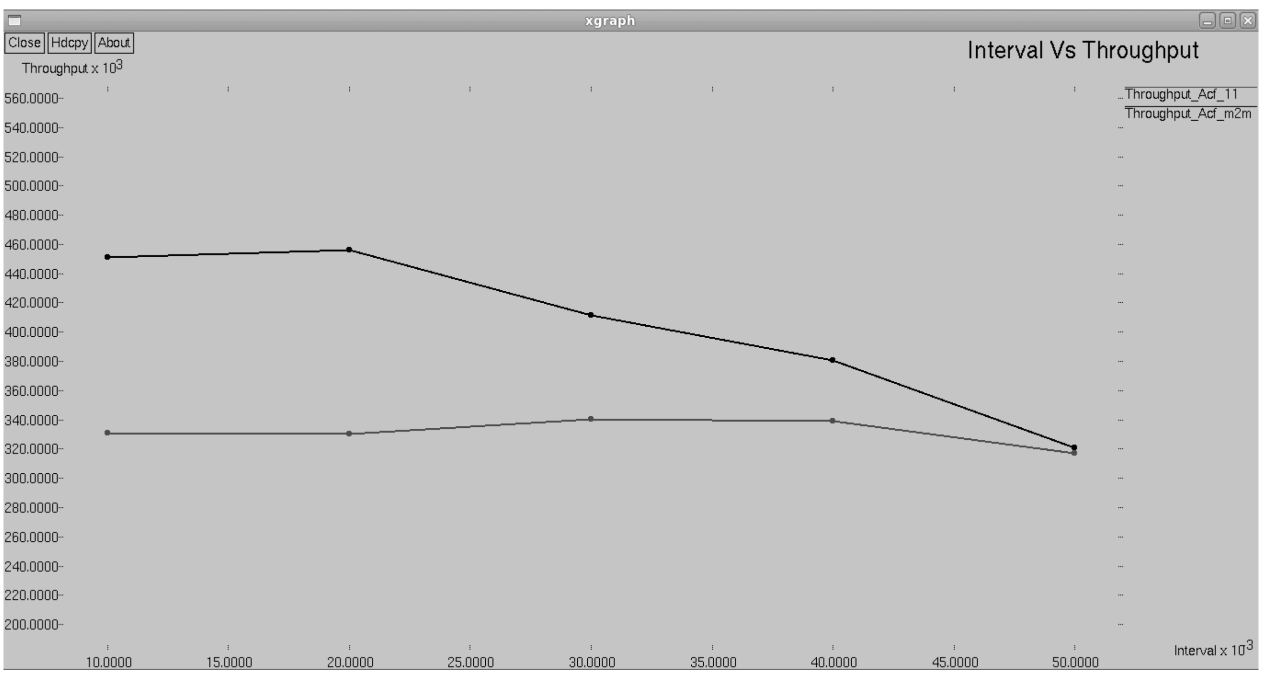


Figure 8: Interval Vs Throughput

9. CONCLUSION

To reduce the routing overhead in wireless sensor networks, we proposed a probabilistic rebroadcast protocol based on neighbor coverage. Additional coverage ratio and connectivity factor is included in the neighbor coverage knowledge. To dynamically calculate the rebroadcast delay, which is used to determine the forwarding order and more efficiently exploit the neighbor coverage knowledge, we proposed a new scheme. This scheme is combined with MAC protocol and following are the advantages of it. The proposed protocol generates less rebroadcast traffic than the flooding and some other optimized scheme in literatures are shown in simulation results. The proposed protocol mitigates the network collision and contention because of less redundant rebroadcast, so as to increase the packet delivery ratio and decrease the average end-to-end delay. The proposed protocol has good performance when the network is in high density or the traffic is in heavy load is also shown in the simulation results.

REFERENCES

[1] Gossip-Based Ad Hoc Routing Zygmunt J. Haas Joseph Y. Halpern Li Li School of Electrical and Computer Engineering/ Department of Computer Science Cornell University.

[2] J. Kim, Q. Zhang, and D.P. Agrawal, "Probabilistic Broadcasting Based on Coverage Area and Neighbor Confirmation in Mobile Ad Hoc Networks," Proc. IEEE GlobeCom, 2004.

[3] W. Peng and X. Lu, "On the Reduction of Redundancy in Mobile Ad Hoc Network" Proc. ACM MobiHoc, pp. 29-130, 2000.

[4] J.D. Abdulai, M. Ould-Khaoua, L.M. Mackenzie, and A. Mohammed, "Neighbor Coverage: A Dynamic Probabilistic Route Discovery for Mobile Ad Hoc Networks," Proc. Int'l Symp. Performance Evaluation of Computer and Telecomm. Systems (SPECTS '08), pp. 165-172, 2008.

[5] J. Chen, Y.Z. Lee, H. Zhou, M. Gerla, and Y. Shu, "Robust Ad Hoc Routing for Lossy Wireless Environment," Proc. IEEE Conf. Military Comm. (MILCOM '06), pp. 1-7, 2006.

[6] A. Keshavarz-Haddady, V. Ribeirox, and R. Riedi, "DRB and DCCB: Efficient and Robust Dynamic Broadcast for Ad Hoc and Sensor Networks," Proc. IEEE Comm. Soc. Conf. Sensor, Mesh, and Ad Hoc Comm. and Networks (SECON '07), pp. 253-262, 2007.

[7] RBP: Robust Broadcast Propagation in Wireless Networks Fred Stann John Heidemann Rajesh Shroff Muhammad Zaki Murtaza USC/ISI 4676 Admiralty Way, Marina Del Rey, CA, USA.