VOL. 10, NO. 10, JUNE 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

COST BASED MULTIPATH ROUTING IN WIRELESS SENSOR NETWORK FOR MULTIMEDIA DATA

Vairam T.[1] and Kalaiarasan C.[2]
[1]Department of Information Technology, PSG College of Technology, India
[2]Department of Computer Science and Engineering,Tamilnadu College of Engineering, Anna University, Coimbatore, India
E-Mail: tvm@ity.psgtech.ac.in

ABSTRACT

A Wireless Sensor Network (WSN) essentially consists of data sensor nodes and video sensor nodes, which senses both sound and motion of events. Routing is the process of moving packets across a network from one node to another. It is usually performed by dedicated devices called routers. Single path routing protocol is used for route discovery and resource utilization. There are some drawbacks like reduced network throughput, network performance, increased traffic load and delay in data delivery. To overcome these problems, multipath routing is preferred. It is the method of using multiple paths in a network. A new system called Cost Based Multipath Routing (CBMR) for multimedia data is proposed based on multipath routing to improve the reliability of data transmission. The need for the proposed system is to provide multipath routing for multimedia data in an efficient manner where the best path is chosen based on the cost. . The proposed system incorporates Real-time Transmission Protocol (RTP) and Multiple Description Coding (MDC) for transmission of multimedia data. Using these techniques, the multimedia data is encoded into trace files at the sender side and the decoding of the trace files is performed at the receiver side. The generated trace files contain information about frames, pixels and headers. This system is applied between the source and sink in a wireless sensor network. The proposed protocol is simulated using Network Simulator2.

Keywords: wireless sensor network, routing overhead ETX, routing, cbmr, throughput, delay.

1 INTRODUCTION

A Wireless Sensor Network (WSN) [1] [2] is a large collection of sensor nodes with limited power supply and constrained computational capability. Due to the restricted communication range and high density of sensor nodes, packet forwarding in sensor network is usually performed through multi-hop data transmission. Therefore, routing in WSN has been considered an important field of research over the past decade. The main responsibility of the sensor nodes in each application is to sense the target area and transmit their collected information to the sink node for further operations. Resource limitations of the sensor node and unreliability of low-power wireless links, in combination with various performance demands of different applications impose many challenges in designing efficient communication protocols for WSN. Meanwhile, designing suitable routing protocols to fulfill different performance demands of various applications is considered as an important issue in WSN. The sensor nodes perform desired measurements, process the measured data and transmit it to a base station, commonly referred to as the sink node, over a wireless channel. The base stations collect data from all the nodes and analyze those data to draw conclusions about the activity. Sink which is a powerful data processor or access point for human interface can also act as gateway to other networks. WSN [3] have restrictions in supporting the video/audio streaming applications because of the lack of raw bandwidth, poor link characteristics and limited power supply. Multimedia data processing in WSN has become promising technology. The data sensor node finds and locates the target in order to sense the motion, sound, heat and light. On the other hand, it triggers the video sensor nodes in order to transmit the video. When multiple paths are being used concurrently, the broadcast nature of wireless channels result in inter-path interference which significantly degrades end-to-end throughput.

Recently, various routing protocols have been proposed for WSNs. Most of them use a single path [4] to transmit data. The optimal path is selected based on the metrics, such as the gradient of information, distance to the destination or the node residual energy level. Various other routing protocols [5] [6] which use multiple path will choose the network reliability as their design priority.

The objective of the proposed system is to transmit multimedia data between source and sink efficiently in wireless medium with increased throughput, reduced delay, minimum routing overhead and energy consumption. Multipath routing approach is used to implement the system with improved network performance through efficient utilization of available network resources.

Work presented here is organized as follows: Section 2 presents the related work with respect to multipath routing and multimedia data. Section 3 presents a description of the proposed system. Section 4 discusses the implementation environment and the result analysis. Finally, the conclusions and the future work are provided on Section 5.

2. RELATED WORKS

The comparisons of multipath routing protocols were done in [7]. Ad hoc On Demand Multipath Distance Vector Routing [8] extends AODV [4] to provide the multipath. Here each Route Request and Route Reply defines an alternative path to the source or destination. Node-disjointness is achieved by suppressing duplicated Route Request at intermediate nodes. Routing entries contains the list of next hops in which the multiple paths are maintained. AOMDV introduces the maximum hop count value which is the advertised hop count.

Since sensor nodes have limited energy capacity, the quality of some applications is influenced by the network lifetime and the energy consumption. The multipath routing protocol utilizes a multipath routing approach to provide energy-efficient communications through balancing of the network traffic over multiple paths. To this aim, the residual battery lives in the nodes are the most important metric considered in the route discovery phase. Nevertheless, as this protocol neglects the effects of wireless interference and assumes error-free links, it cannot achieve significant performance improvement in throughput and data delivery ratio. An Interference Aware multipath routing protocol (IAMR) [5] to support high rate streaming in wireless sensor networks is proposed. This protocol constructs link-disjoint paths by assuming a specific network model and localization support. It is assumed that there are several gateway nodes connected directly to the command center using non-interfering and high capacity links.

The source node constructs three link-disjoint paths towards the three distinct gateway nodes. After that, the source node utilizes the primary and secondary paths for data transmission and preserves the third path for prompt packet recovery from path failures. Although IAMR shows higher performance compared to the standard AOMDV. Moreover, to reduce the negative effects of intra path interference, IAMR constructs shortest paths towards the gateway nodes. Since the longest hops should be used to create the shortest paths, the time varying properties of wireless links highly affect the performance achieved by this protocol.

Energy Efficiency QoS Assurance Routing in Wireless Multimedia Sensor Networks (EEQAR) [9] introduces a social network analysis to optimize network performance. EEQAR focuses on how to build energy-efficient QoS assurance routing for WMSNs. However, EEQAR does not use a link quality estimator to select reliable routes, generating an extra overhead for route discovery for intra-cluster communication. It does not evaluate the video quality level.

A Power Efficient Multimedia Routing (PEMuR) [6] aims to provide an efficient video communication based on a combination of hierarchical routing protocol and video packet scheduling models. The protocol creates clusters in a centralized way by using a combination of beacon, schedule, advertise, identifier and join messages. The main drawback of PEMuR is that it only uses the remaining energy to find routes (not link quality), which makes the proposal unreliable. Thus, PEMuR does not assure the transmission of videos with QoS/QoE support. The proposal assumes that the BS can communicate with all nodes by using single-hop communication, which is not realistic for large sensor networks. The protocol uses a centralized scheme to create clusters, which increases signaling overhead and decreases network lifetime.

3. PROPOSED WORK

The proposed CBMR is designed for transmission of multimedia data in an efficient manner. It is implemented using RTP and MDC [10]. The main operations involved here is generation of trace files and transmission of multimedia data using RTP and CBMR. The Figure-1 shows overview of the system

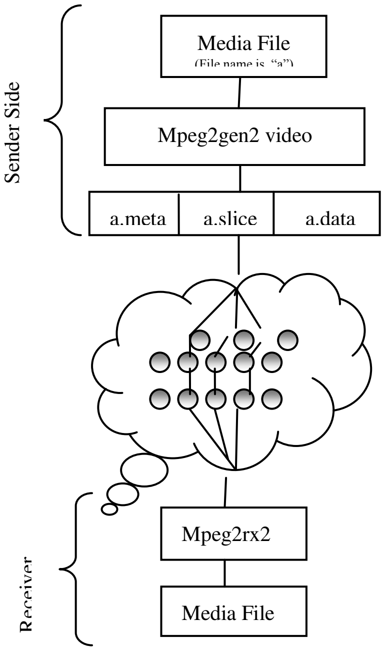


Figure-1. System overview.

The media file is divided into three trace files in the format of data, meta, slice in sender side using Mpeg2gen2 video tracer. Data file is the actual video data consists of each pixel information. Meta file is the raw data file consists of header information. Slice file consists of frame positions and its details. Frames are used for encoding in the sender side where as in the receiver side; the slice counter is used for decoding. Generated trace files are sent to the receiver by means of RTP. The audio and video packets are delivered in a standardized format with the help of RTP. On the receiver side Mpeg2Rx2 is used to receive the video files.

For progressive sequences, all pictures are considered as frame pictures as shown in Figure-2. The output of the decoding process for a progressive sequence is a series of reconstructed frames. Encoded pictures are classified into 3 types: I, P, and B. I Pictures are Intra

Coded Pictures that means all macroblocks coded without prediction. Needed to allow receiver to have a "starting point" for prediction after a channel change and to recover from errors. P Pictures are Predicted Pictures. Here, macroblocks may be coded with forward prediction from references made from previous I and P pictures or may be intra coded. B Pictures are Bi-directionally predicted pictures. Macroblocks may be coded with forward prediction from previous I or P references otherwise macroblocks may be coded with backward prediction from next I or P reference. They also coded with interpolated prediction from past and future I or P references. Note that in P and B pictures, macroblocks may be skipped and not sent at all. The decoder then uses the anchor reference pictures for prediction with no error. B pictures are never used as prediction references.

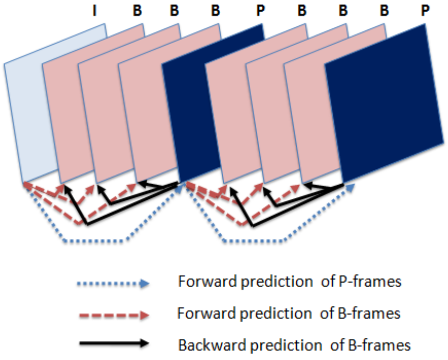


Figure-2. Representation of IPB frames.

For an example, the frame sequence shown in Figure-2 is transferred in the following order: I P B B B P B B B. The only task of the decoder is to reorder the reconstructed frames. To support this, an ascending frame number comes with each frame. Thus, the reconstruction of inter coded frames goes ahead in two steps:

Step-1: Application of the motion vector to the referred frame

Step-2: Adding the prediction error compensation to the result

The prediction error compensation requires less byte than the whole frame because the white parts are zero and can be discarded from MPEG stream. Furthermore the DCT compression is applied to the prediction error which decreases its memory size.

Routing process encompasses of three phases. In initialization phase, each node acquires its neighborhood information. This information will be used in the route discovery and establishment phase to find the next best next-hop node towards the sink. The route discovery and establishment phase is triggered whenever an event is detected. The outcome of this phase is multiple-interference-minimized paths between the source and the sinks. Finally, the route maintenance phase handles path failures during data transmission. The load balancing algorithm is taken here by distributing the traffic over the multiple established paths. Once the route is established then the node starts to transmit the data.

3.1 Initialization phase

In CBMR protocol, each node obtains its neighborhood information, which also includes the ETX (Estimated Transmission Cost) of its neighbor towards the sink. The ETX value of a link indicates the required number of transmissions for successful packet reception at the receiver. Thus, ETX is affected by the link loss ratio, the difference between forward and backward reception rates, and the interference level of successive links (i.e., intra- path interference) [11]. The ETX value of the link is defined as follows:

$$ETX = \frac{1}{p*q} \tag{1}$$

Where p and q are the probabilities of forward and backward packet reception over that link. Each of the node calculates its own accumulated ETX value to the sink node as follows:

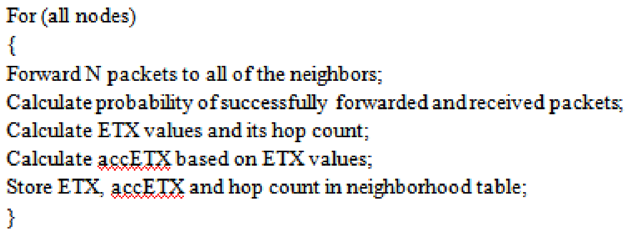
$$accETX_i = accETX_j + \frac{1}{p_{ij}*q_{ij}} \tag{2}$$

Node i receives a broadcast packet from node j, it saves the cost included in this packet as the accumulated ETX cost of node j to the sink. The path cost is calculated here for each path and the transmission of data packet is based on that cost.

In the beginning, each node will forward a set of data packets to other nodes and records the number of successfully received packets from its neighbors. So that it will record p and q value in the neighborhood Table. Then, the sink node sets its cost as zero and broadcasts to the neighbors. Whenever a node receives packet with cost, it records the retrieved cost as the accETX cost of the neighbor node. For example, when node i receives a broadcast packet from node j, it saves the cost included in this packet as the accETX cost of node j to the sink.

Node i should broadcast the newly calculated accumulated ETX cost, if it is lower than the current cost of node i towards the sink. In fact, whenever a node receives a broadcast packet from one of its neighbors, it should calculate its accumulated ETX by equation (2), cost through that node and broadcast that value if it is lower than its current ETX cost towards the sink. In addition to the initialization phase, the cost update process should also be performed during normal network operation whenever a node finds a new transmission cost to the sink. The pseudo code for initialization phase is given in Table-1.

Table-1. Psuedo code intialization phase.



```
For (all nodes)
{
Forward N packets to all of the neighbors;
Calculate probability of successfully forwarded and received packets;
Calculate ETX values and its hop count;
Calculate accETX based on ETX values;
Store ETX, accETX and hop count in neighborhood table;
}
```

3.2 Route discovery and establishment phase

Whenever the event is detected the route discovery phase is triggered. After the discovery, it uses the neighborhood Table and update it. The source node starts data transmission by sending Route Request to the neighbor node. After receiving the route request packet, the node will calculate its transmission cost for the neighboring nodes which are not included in any path from the current source to the sink. This avoid the same node included in the different path. The node with minimum cost is included in the path selection. Node ID is recorded by each path after receiving the request packet which is used to form the reverse path. Transmission of packets is through the node which having lower cost.

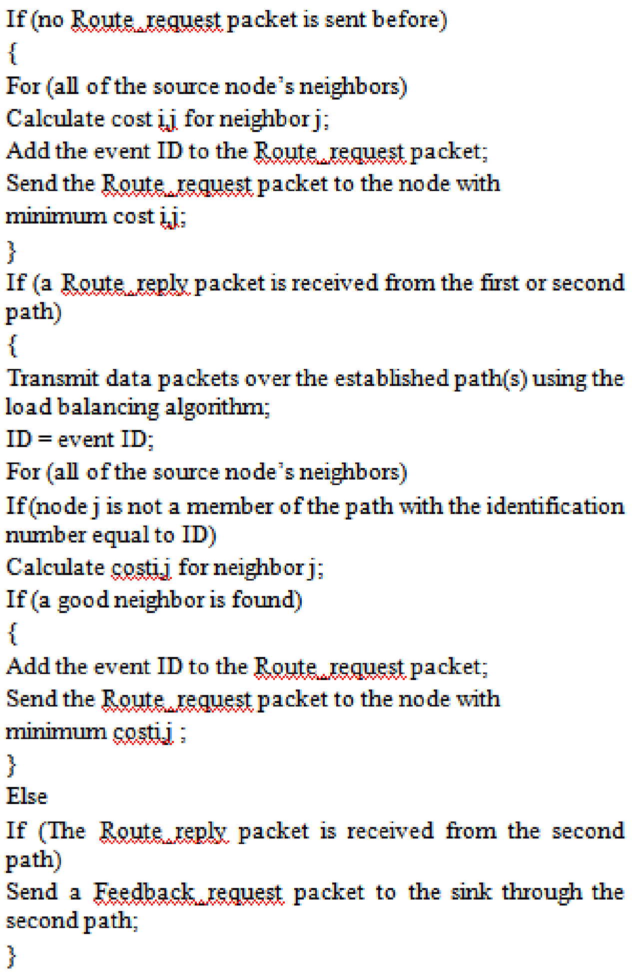
$$cost_{i,j} = \left(accETX_j + \frac{1}{p_{i,j}q_{i,j}}\right) \cdot \left(\frac{1}{resBatt_j}\right) \cdot \left(1 + ilevel_j\right) \qquad (3)$$

In equation (2), accETX*j* is the ETX cost of node *j* to the sink, which is contained in the neighborhood table of node *i*. *pi,j*and *qi,j* are the forward and backward packet reception rates between node *i* and node *j*, respectively. resBatt*j* is the remaining battery level of node *j* expressed in percentage .Interference Level *j* is the maximum interference level that node *j* has experienced. Path memebership variable is used where it sets 1 for Route_Request, it will show that the particular node is already used by another path. Automatic Repeat_Request is used for ensuring packet delivery. Receiving a Route_request packet at the source node indicates that the algorithm cannot establish another node-disjoint path. When the sink receives a Route_request packet, it replies by transmitting a Route_reply packet along the reverse path. Here the Path membership variable for Route_reply is set as 2 to indicate an active path passing through this node.

In order to reduce the end-to-end latency, it starts packet transmission immediately after the first path is established. To initiate the second path, route discovery will send Route_request to another path. The source node distributes data packets over the first and second path using the load balancing algorithm.

Received Packet Throughput (RPT) is used at the sink node to measure the performance improvement. It is calculated for each data packets from source node separately. Positive feedback is send to the path which is having higher RPT and sends negative feedback which has lower RPT. Suppose if there are n numbers of paths are established and data packets are being transmitted through n paths, then it compares RPT of n paths with the RPT of n-1 paths and decides if transmitting over the n paths results in higher performance.

Table-2. Pseudo code for route discovery process.



```
If (no Route_request packet is sent before)
{
For (all of the source node's neighbors)
Calculate cost i,j for neighbor j;
Add the event ID to the Route_request packet;
Send the Route_request packet to the node with minimum cost i,j;
}
If (a Route_reply packet is received from the first or second path)
{
Transmit data packets over the established path(s) using the load balancing algorithm;
ID = event ID;
For (all of the source node's neighbors)
If(node j is not a member of the path with the identification number equal to ID)
Calculate costi,j for neighbor j;
If (a good neighbor is found)
{
Add the event ID to the Route_request packet;
Send the Route_request packet to the node with minimum costi,j ;
}
Else
If (The Route_reply packet is received from the second path)
Send a Feedback_request packet to the sink through the second path;
}
```

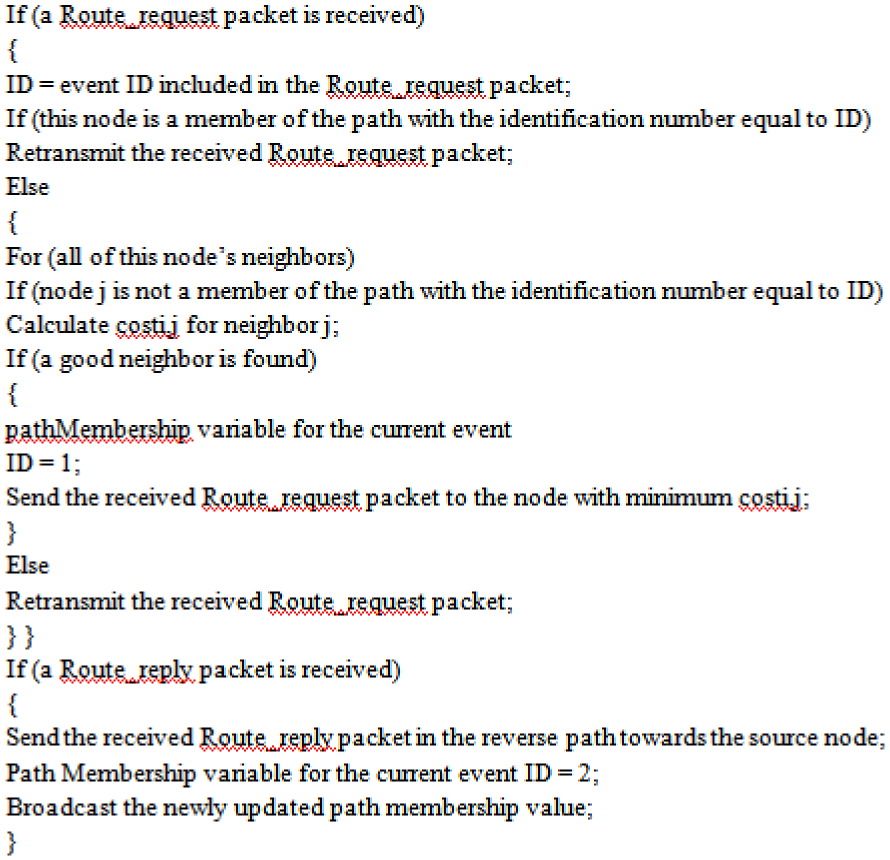
3.3 Route maintenance phase

In event-driven applications, the data packets should be transmitted after the event detected.Therefore the chance of node failure are lower than that of monitoring applications in which there is frequent data transmission from all the nodes to the sink. According to the packet transmission mechanism at the data link layer, if a node on an active path does not receive the ACK packet from the nexthop node after k efforts, it notifies the network layer about the link failure.

After link failure detection, an error message will be transmitted to the source node through the reverse path. Upon reception of this message at the source node, it disables the path from which this message has been received and redistributes traffic over the remaining paths.

Furthermore, to prevent performance degradation, the source node initiates a new route discovery and establishment process.

Table-3. Pseudo code for route establishment and maintenance phase.



```
If (a Route_request packet is received)
{
ID = event ID included in the Route_request packet;
If (this node is a member of the path with the identification number equal to ID)
Retransmit the received Route_request packet;
Else
{
For (all of this node's neighbors)
If (node j is not a member of the path with the identification number equal to ID)
Calculate costij for neighbor j;
If (a good neighbor is found)
{
pathMembership variable for the current event
ID = 1;
Send the received Route_request packet to the node with minimum costij;
}
Else
Retransmit the received Route_request packet;
} }
If (a Route_reply packet is received)
{
Send the received Route_reply packet in the reverse path towards the source node;
Path Membership variable for the current event ID = 2;
Broadcast the newly updated path membership value;
}
```

4. IMPLEMENTATION ENVIRONMENT

In this section we describe the performance metrics, simulation environment and simulation results. We used NS-2 [12] to implement and conduct a set of simulation experiments for our algorithms and did a comparative study with the EECA and AOMDV protocol.

Our simulation environment consist of various set of nodes starting from 30 nodes to 250 nodes which are randomly deployed in a field of 1000mx1000m all nodes are identical with a transmission range 250m [13] [14]. Table-1 shows the simulation parameters.

Table-4. Simulation parameters.

Parameters	Value
Transmission range	250 m
Interference range	550 m
Simulation Time	>800s
Topology Size	1000m x 1000m
Number of Sensors	100
Number of sinks	1
Traffic type	Constant Bit Rate
Packet size	512 bytes
Bandwidth	2Mb/s
Transmission range	250m
Interference range	550m
Initial energy in batteries	10 Joules
Energy Threshold	0.001mJ
Node distance	200 m

5. RESULT ANALYSIS

We compare our method CBMR with EECA and AOMDV over different network sizes of 30, 60, 90, 120, 150, 180, 210 and 250. Figure-3 shows the average end to end delay of CBMR, EECA and AOMDV. End to end delay refers to the time taken for a packet to be transmitted across a network from source to sink. It is the sum of transmission delay, propagation delay and processing delay. End to end delay calculation also includes number of routers. The average End to end delay was calculated for each set of nodes. From the result we can observe that AOMDV and EECA have longest delay. On the other hand, CBMR has less delay since it maintains route details, will not take much time to reroute the packets in a new path when one of the multiple route is disconnected, and hence no route acquirement latency is required.

Figure-4 shows the average throughput of CBMR, EECA and AOMDV. Throughput is the average rate of successful message delivery over a network. This data may be delivered over a physical or logical link, or pass through a certain network node. The throughput is usually measured in bits per second (bit/s or bps), and sometimes in data packets per second or data packets per time slot. Here the throughput was measured as data packets per second. The average throughput was calculated for each set of nodes. When the number of nodes are minimum we can achieve a better throughput since there is a less traffic between source and sink, whereas the number of node increases, observably the performance of the protocol will be decreased.

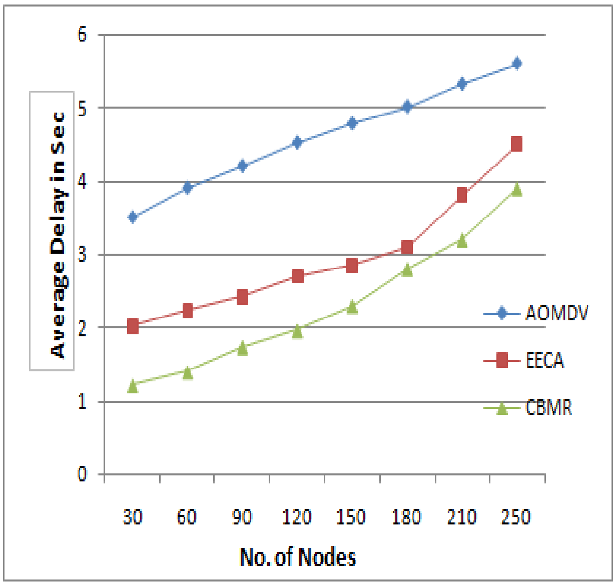


Figure-3. Average end to end delay.

Figure-5 shows the routing overhead of CBMR, EECA and AOMDV. Data bits added to user-transmitted data, for carrying routing information and error correcting and operational instructions. Routing overhead refers to the time it takes to transmit data on a network. Each packet requires extra bytes of that are stored in the packet header. Information will be combined during assembly and disassembly of packets which leads to reducing the overall transmission speed of the raw data. The Figure-5 shows the experimental results of average routing overhead. Again this metric also affected when the number of nodes increases. From the Figure-5 we could easily interpret that CBMR achieves better results than the EECA and AOMDV.

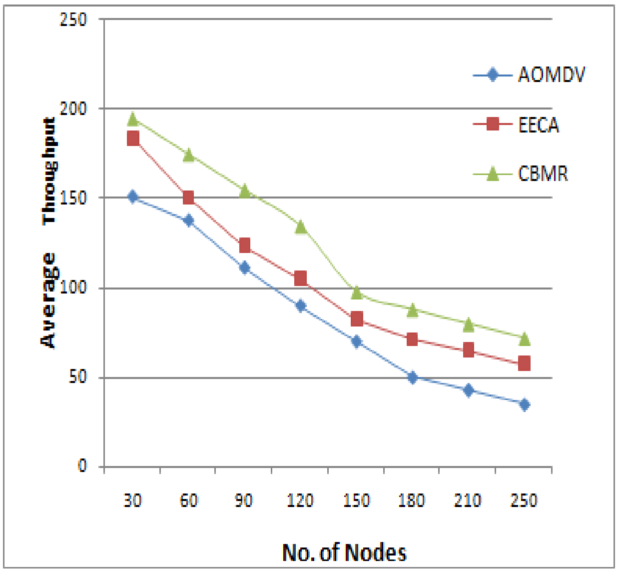


Figure-4. Average throughput.

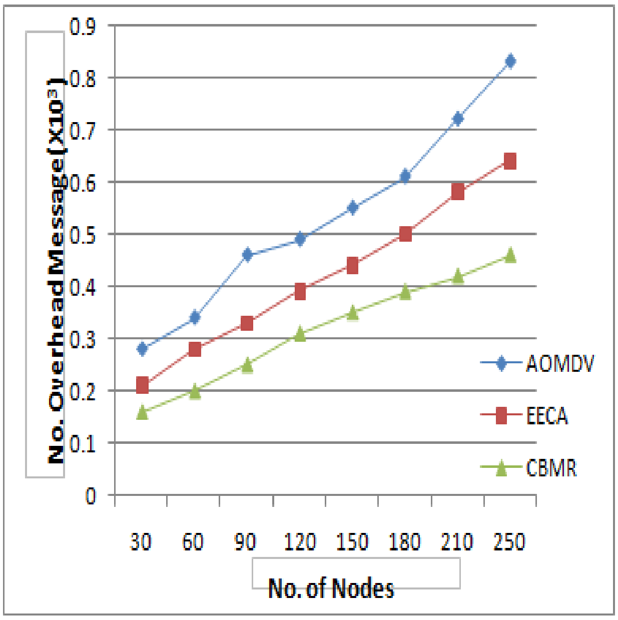


Figure-5. Average routing overhead message.

Energy is one of the important QoS factors in WSNs. To increase network lifetime, energy must be saved in every hardware and software solution composing the network architecture. Therefore, it is desirable to use short -range instead of long-range communication between sensor nodes because of the transmission power required. We also considered the energy [15] as one of the evaluation metric. In Figure-6 shows the analysis of average energy consumption and our proposed protocols consumes less energy compared than the other.

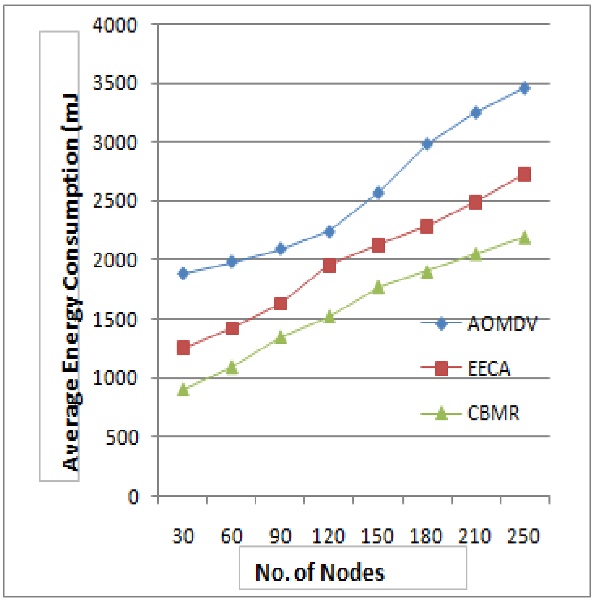


Figure-6. Average energy consumption.

6. CONCLUSIONS

Cost Based Multipath Routing protocol is implemented to improve QoS demands in Wireless Sensor Network. The proposed system includes phases such as trace file generation, initialization, routing and maintenance. In this system, at the sender side the multimedia data is encoded into trace files using MDC and transmitted to the receiver side using RTP. CBMR guarantees efficient transmission of multimedia data in which the routing phase plays important role in selecting best path. The packet transmission takes place based on the cost calculated for each node. At the receiver side, the successfully received multimedia data is decoded using MDC. CBMR, the cost based technique is analyzed with AOMDV and EECA in a graphical manner. In this analysis, the system with cost based technique gives more efficient results. In the proposed work CBMR is applied to static nodes which can be further extended for mobile nodes in the future enhancement.

REFERENCES

[1] Shuang Li, Raghu KisoreNeelisetti, Cong Liu and Alvin Lim. 2010. Efficient Multipath protocol for Wireless Sensor Networks. In: Proc. International Journal of Wireless and Mobile Networks (IJWMN). 2(1).

[2] Akkava K and Younis M. 2005. A Survey on Routing Protocols for Wireless Sensor Networks. In: Proc. Ad Hoc netw. J. 3: 325-349.

[3] Kai Lin, Rodrigues J.J.P.C., Hongwei Ge, Naixue Xiong. 2011. Energy Efficiency QoS Assurance Routing in Wireless Multimedia Sensor Networks. Systems Journal, IEEE Volume: 5 issue: 4.

[4] C.E Perkins and E.M Royer. 1999. Ad Hoc On-Demand Distance Vector Routing. Proc. Second IEEE Workshop Mobile Computing Systems and Application. pp. 90-100.

[5] Vairam T, Kalaiarasan C. 2013. Interference Aware Multi-path Routing Wireless Sensor Network", Internatioanl journal of Emerging Science and Engineering.

[6] Denis Rosário *et al.* 2012. A Hierarchical Multi-hop Multimedia Routing Protocol for Wireless Multimedia Sensor Networks. Network Protocols and Algorithms ISSN 1943-3581, vol. 4.

[7] Yamuna Devi, Vairam T, Kalaiarasan C. 2012. Efficient Comparison of Multipath Routing Protocols in WSN. Proceedings of International Conference on Computing, Electronics and Electrical Technologies.

[8] Mahesh K. Marina and Samir R. Das. 2006. Ad hoc on-demand multipath distance vector routing. Wireless communications and mobile computing. 6: 969-988.

[9] Zijian Wang, Eyuphan Bulut and Boleslaw K.Szymanski. 2009. Energy Efficient Collision Aware Multipath Routing for Wireless Sensor Networks. In: Proc. International Conference on Communication. pp. 1-5.

[10] Y. Wang and S.Lin. 2001. Error resilient Video Coding using Multiple Description motion compensation. In: Proc. IEEE Fourth Workshop on Multimedia Signal Processing. pp. 441-446.

[11] MarjanRadi, BehnamDezfouli, Kamalurlnizam Abu Bakar, Shukor Abd Razak and Mohammad Ali Nematbakhsh. 2011. Interference-aware multipath routing protocol for Qos improvement in event-driven wireless sensor networks. In: Proc. Tsinghua Science and Technology. 16(5).

[12] Ns-2.34: Network Simulator (http://www.isi.edu/nsnam/ns/.).

[13] Jonathan L. Bredin, Erik D. Demaine, Mohammad Taghi Hajiaghayi and Daniela Rus. 2010. Deploying Sensor Networks with Guaranteed Fault Tolerance. In Proc IEEE/ACM Transactions on Networking. 18(1).

[14] Bettstetter C, Reseta G, Santi P. 2003. The node distribution of the random way point mobility model for wireless ad hoc networks. IEEE Transactions on Mobile Computing. 2(3): 257-269.

[15] S. Lindsey, C.S. Raghavendra and K.M Sivalingam. 2002. Data Gathering algorithms in sensor networks using energy metrics. IEEE Transaction on Parallel and Distributed Systems. 13(9): 924-935.