International Journal of Electrical and Computer Engineering (IJECE)
Vol. 8, No. 6, December 2018, pp. 4745~4754
ISSN: 2088-8708, DOI: 10.11591/ijece.v8i6.pp4745-4754

Cross Layer Solution for Energy and Delay Optimization in MANETs

Bhagyashri R Hanji[1], Rajashree Shettar[2]
[1]Department of Computer Science Engineering, Global Academy of Technology, India
[2]Department of Computer Science Engineering, R V College of Engineering, India

Article Info

Article history:

Received Jan 1, 2018
Revised Jun 18, 2018
Accepted Jul 28, 2018

Keyword:

AODV
Energy management
MANET
Overhead reduction
Pairing function
Route optimization

ABSTRACT

A novel method for packet forwarding in MANETs has been proposed in this paper. A node in the network acts as both host and router. Energy utilization of the node increases as all nodes in MANET operate as source, destination, and router to forward packets to the next hop ultimately to reach destination. Routers execute a variety of functions from simple packet classification for forwarding to complex payload revision. As the number of tasks and complexity increases, processing time required also increases resulting in significant processing delay in routers. The proposed work optimizes packet header at transport and network layer by calculating Unique Identifier using pairing function for the fields which do not change for a source–destination pair. This technique optimizes the processing cost of each packet header thereby conserving energy and reducing delay. It also simplifies the task of system administration. This paper elucidates an extension to basic AODV protocol, allowing routing of most packets without an explicit header, reducing the overhead of the protocol while still conserving its basic properties. The proposed method improves the network performance significantly compared to AODV, MTPR, and S-AODV protocol.

Corresponding Author:

Bhagyashri R Hanji,
Department of Computer Science and Engineering,
Global Academy of Technology,
Bengaluru, Karnataka, India.
Email: bhagyashri@gat.ac.in

1. INTRODUCTION

Mobile Ad Hoc NETwork (MANET) is a self-governing group of scattered mobile users, where every node operates as source, destination, and also passes on packets for other nodes, hence acts as a router also. The idea of Mobile Ad Hoc Networking is to maintain robust and proficient operation in mobile wireless networks by including routing functionality into mobile nodes [1], [2]. MANETs have gained much attention in research aimed to improve their basic performance and providing Quality of Service (QoS). QoS support in MANETs requires more harmonious cooperation between layers, necessitating exchange of information between layers rather than conventional independent layered network architecture. There is current and future need for dynamic Ad Hoc networking technology. Internet Protocol (IP) is the first choice of transport layer protocol in wired and wireless networks. Generally it has been observed that the header size of the packet is about the same size of the packet or bigger than the payload data. The protocol headers are significant for end-to-end connections involving multiple hops whereas the protocol headers are insignificant and do not serve any constructive purpose when there are no intermediate nodes and communicating nodes are connected directly [3]. Optimizing the packet header size leads to improvement in efficient resource utilization [4].

In Rahul Desai *et. al*. authors perform a comparative analysis of various existing routing protocols such as DSDV, AODV, AOMDV, OLSR and DSR [5]. Analysis shows that AODV and DSR are better

suitable for high traffic network at the cost of large delay. In S. Syes *et. al*. authors propose a method to improve the performance of AODV and DSR reactive routing protocol in terms of delay and overhead [6]. The link residual life is estimated to reduce link failure and velocity of moving node is considered while choosing the next node to forward data. The authors propose a method to find better routing path having more energy than other routes through the analysis of average energy and minimum energy of paths extending the network lifetime in [7]. An algorithm to effectively manage the consumption of bandwidth and energy by applying mobile agents to carry the data is put forward by authors in [8].

The mobile agent will move to the neighbor node with the route request packet. It will look for node's route cache to find the path to the destination node. During this search process the mobile agent will operate in the disconnected mode and hence will not use bandwidth. Thus the bandwidth and power are optimized. In Rahul *et. al*. authors describe a new optimization technique based on reinforcement learning where routing tables are replaced by estimation tables called as Q Values [9]. Q values are based on link delay. The results show that the number of packet drops and delay is even low when the network size is increased. In [10], authors present a new identifier based multihop IP header compression design. Furthermore it also resolves problems with pure IP based adhoc networks emerged due to IP address auto configuration service, distributed naming and name resolution and at application layer the role of an IP address as an identifier. An approach to lower networking overhead and investigate the consequence of larger packet sizes is focused in [11].

The article reviews different methods to reduce overhead by having large frame size, interrupt coalescing, and copy avoidance by page remapping, integrated copy/checksum and hardware computation. High bandwidth in the system can be achieved by reducing host overheads by applying optimizations above and below the required layer, identifying the network interface support. Internet Engineering Task Force (IETF) has developed two IPv6 applicable header compression schemes and their applicability to Global Information Grid (GIG) infrastructure is discussed in article [12]. Several challenges necessary to be thought of before applying header compression to a particular device is also discussed. A novel packet forwarding key identification scheme in optical packet switched networks is presented in [13].

Neither header modification nor label distribution protocol is required, making network management simple and also reduces the component cost. In Hooshiar authors put forward tunnelling based route optimization, which reduces packet overhead and communication delay [14]. Bytes consumed to establish mobile communication is the key comparison metric used. The results shows that more payloads are transmitted as packets sent contain less packet overhead. In Fahim and Vilas authors aim to speed up the data transmission, reducing the packet processing time by optimizing the current IP packet structure and reducing header size [15]. In Bow-Nan *et. al*. authors introduce MANET IP Header Compression (MIPHC) protocol that reduces 20 bytes header to 8 bytes [16]. Various header compression and decompression techniques are introduced earlier, but compressing and decompressing involves various steps to be performed increasing the processing time and energy consumption.

In M. Tamilarasi *et. al.* authors recommend to conserve battery power by transmitting packets with minimum required energy generating less overhead and delay compared to standard DSR [17]. This performance is achieved by modifying packet header format which excludes details of intermediate nodes retaining only source and destination address. Minimum Total Transmission Power Routing (MTPR) which concentrates on end-to-end energy efficiency selecting the minimum hop path has been addressed in [18]-[19] while remaining energy of node is not taken into account for decision making. Location and the angular displacement of nodes while selecting the intermediate nodes has been considered in [20]. This ensures stable path between source and destination in turn reducing the frequent link-breaks and is developed over AODV protocol. The method called stable and reduced link break routing protocol (S-AODV) reduces the process of finding the new path oftenly.

Selecting minimum hop path, sending larger packet size, using header compression and decompression are the different methods used to reduce header overhead. Concentrating on minimum hop paths always attempts to select nodes lying at the center and their involvement in forwarding data is also considerably high. Involving larger or variable packet size increases the percentage of data sent per attempt. In a faulty environment, retransmission of larger packets comes with cost and recomputation. Compressing and decompressing reduces the overall size to be transmitted but the overhead involved with compression and decompression still continues.

The proposed method gives a simple and effective solution by calculating unique number based on the fields of header which remains unchanged for communicating pairs involving very less computation. This number is incorporated in first data packet and is noted down at all intermediate nodes that will be used throughout the session. All the secondary nodes will be identified by this number and accordingly forwarding decision is made. The proposed method gains information through interlayer communication and provides

optimized path with respect two parameters namely energy and delay and is built over AODV reactive routing protocol, named as Optimized –AODV (O-AODV).

2. PROPOSED METHOD

Faster networks transmit more packets and at the same time transmit more payload bytes of specified packet size per unit time, increasing both per-packet and per-byte overhead on the host. Huge portion of energy and time is consumed during the transmission or reception of packet which is used only for the data administration purpose i.e. for processing of headers and trailers [21]. To prolong the service time of devices efficient usage of energy is a requisite. A series of packets exchanged between the same pair of source destination IP addresses, port addresses and using the same transport layer protocol can be treated in the similar manner by a intermediate node for forwarding once the decision is made based on first packet [22]. So instead of sending the same repeated data fields, those fields can be replaced by an identifier field. The proposed method, O-AODV works in two steps:

Firstly when the data is received from higher layer, primary packet is created with all the necessary calculations done and sent to lower layer for further processing and transmission. Next, all data received will fit into secondary packets with compact header added and sent to lower layer for transmission as shown in Figure 1. The header of primary packet is greater than secondary packet header. Every intermediate node which first receives the primary packet stores all the details in its routing table to further identify the flow. The secondary packet received next will be checked with the unique identifier if matched it will be forwarded further to next node. This method leads to less processing time and the header size is also compact reducing the overhead data which is carried with every packet ultimately reducing the energy consumption.

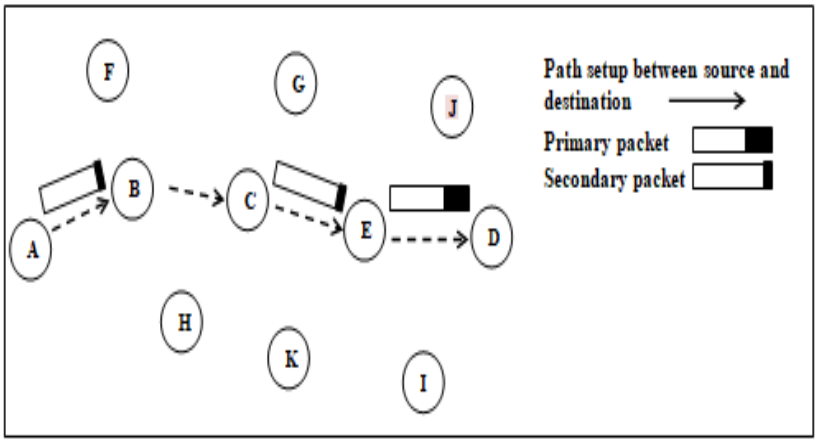

Figure 1. Primary and Secondary Packet transmission

Primary and secondary packet formats are detailed in Figure 2. Few fields remain constant for a communication session between two nodes. For example, the source and destination port in UDP header remains constant. These two fields are taken as input and unique identifier is calculated. Source and destination IP Addresses remain same in IP header for a communicating pair. These two fields are taken as input and Unique Address Pair Identification Number is calculated and stored. Version, Internet Header Length, Type of service, Identification, Flags, Time to Live, Protocol remains unchanged. Every communication starts with primary packet which contains all the fields along with unique identifiers. Every intermediate node forming the path stores the details in routing table which will be further used to identify all secondary packets. Secondary packets follow the primary packet containing compacted packet header and increased payload maintaining same fragment size at the lower layers. M bit when 1 is used to indicate the first/middle fragments and last fragment when set to 0. First field P/S bit is used to indicate Primary Packet (P/S=1) and Secondary Packet (P/S=0).

Packet header structure is modified as shown in Figure 3, so as to work according to the requirements. The amount of data stored in routing table also differs to support the requirements. The incoming packet is first checked for primary or secondary with the first field (P/S) and the packet is processed accordingly. For every primary packet received most of the details along with identification

number for the session are stored in each tuple and the secondary incoming packets are only checked for matching identifier and forwarded accordingly.

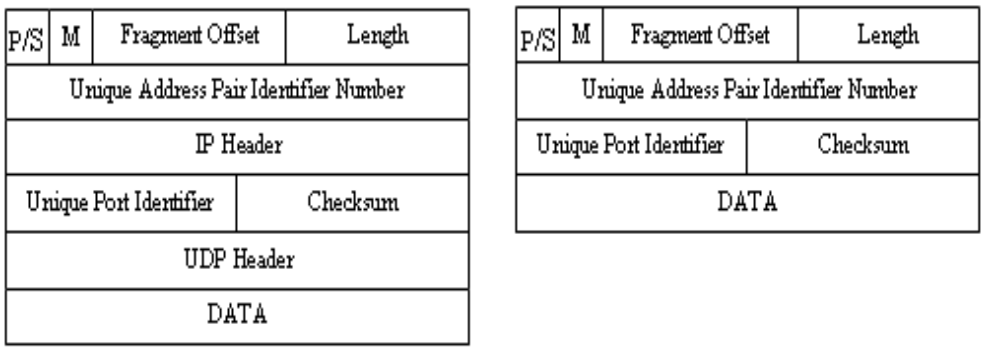

Figure 2. Primary and secondary Packet Header Fields

```
class  rt_entry {
nsaddr_t rt_dst;
nsaddr_t  rt_src;
u_int32_t  rt_uid;
u_int16_t  rt_pid;
u_int8_t  rt_ps;
…….
}
```

```
recv-pkt  (Packet *p)
{
Header_LS*  rh  = Header_LS :: access(p);
Header_IP*   ih  = Header_IP :: access(p);
Header_PP*  pp = Header_PP :: access(p);
Header_SP*  pp = Header_SP :: access(p);
……
If(pp_ps  == 1)
  Process the received packet as primary packet
Else
  Process the received packet as secondary packet
……..
}
```

Figure 3. Modified packet header structure and its processing

The mathematical calculation (theoretical) for standard protocol header overhead is as shown in Table 1. In IPv4 version, fields like version, Internet Header Length, Type of Service, Identification, flags, protocols remains unchanged. The values depict that for the primary packet the overhead is increased while for secondary packets the overhead is decreased. At the end the overall calculation shows that the proposed method performs better. For example if system transmits 50, 100 and 200 IPv4/UDP packets in a session which will have one primary Packet and 49, 99,199 secondary packets respectively.

Table 1. Overhead calculation for standard protocol header without options & padding

Protocol Header	Total Packet Header Size	Packet Header Size		Overhead		Overall efficiency calculation		
		Primary	Secondary	Primary	Secondary	Header Overhead		Average
IPv4/TCP	40 bytes	67 bytes	27 bytes	67.5% (↑)	67.5% (↓)	10 packets 25 packets 50 packets 100 packets 200 packets	77.5% 71.5% 69.5% 68.5% 68%	65%-70% (↑)
IPv4/UDP	28 bytes	40 bytes	12 bytes	42.8 % (↑)	42.8% (↓)	10 packets 25 packets 50 packets 100 packets 200 packets	52.85% 46.85% 44.85% 43.85% 43.35%	35%-40% (↑)

Pairing function [23] is a reversible process to uniquely encode two non-negative numbers into a single unique number. Consider two positive integers a and b, PairFunc [a, b] yields a single positive integer c uniquely associated with the input pair. Inverse function UnPairFunc[c] outputs the positive integer pair [a, b] related to generation of c [24], [25]. Pairing function are bijections and is important in logic, computation and mathematics on the whole [26]. Basic properties of a pairing functions are Injection, Induction and onto as given in equation (1), (2) and (3) respectively. Injective function maps distinct input arguments to distinct output values. Two pairs generating the same identifier is possible only when both pairs are identical. Onto or surjective function maps every possible value to at least one argument.

$$\langle a_1, a_2 \rangle = \langle b_{1,} b_2 \rangle \rightarrow a_1 = b_1 \wedge a_2 = b_2 \quad (1)$$

$$a < \langle a,b \rangle \wedge b < \langle a,b \rangle \quad (2)$$

$$a = 0 \vee \exists b \exists a = \langle b,c \rangle \quad (3)$$

Equation (4) states that the function generates a unique number as output for any given pair of integer numbers ($Inum_1$, $Inum_2$) as input. The output generated is equal only if two pairs of input are equal. Equation (5) states that given a number K as input, then the two integers ($Inum_1$, $Inum_2$) can be generated as output. The pairing function satisfies equation (4) and (5).

$$\Pi(Inum_1, Inum_2) = (Inum_3, Inum_4), then \langle Inum_1, Inum_2 \rangle = \langle Inum_3, Inum_4 \rangle \quad (4)$$

$$\begin{aligned} &for\ each\ K \in N\ there\ is\ a\ pair \langle Inum_1, Inum_2 \rangle \in NXN \\ &such\ that\ \Pi(Inum_1, Inum_2) = K \end{aligned} \quad (5)$$

The Cantor Pairing function is a primitive recursive pairing function defined by equation (6) and (7) below. Both equations describe the function used to produce Unique source destination Pair Identifier (UPI) taking $Inum_1$, $Inum_2$ as input values and N as the set of values.

$$\prod : NXN \rightarrow N : \langle Inum_1, Inum_2 \rangle \quad (6)$$

$$\Pi(Inum_1, Inum_2) := \frac{1}{2}(Inum_1 + Inum_2)(Inum_1 + Inum_2 + 1) + Inum_2 \quad (7)$$

The limitation of Cantor pairing function is that the range of encoded result doesn't always stay within the limits of $2N$ bit integer. If the inputs are two N-bit integers, then there are $2^N * (2^N-1)$ combination of possible inputs. With Pigeon Hole Principle we need an output of size at least $2^N * (2^N-1)$ which is equal to $2^{2N}-2^N$ in other words a size equal to $2N$ bit number. Malthew Szudzik method used in [18] overcomes the above limitation and is defined by equation (8) below. $Inum_1$ and $Inum_2$ are two key input integers based on which a unique number is calculated and returned.

$$\begin{aligned} &Inum_1 \geq Inum_2\ ?\ Inum_1 * Inum_1 + Inum_1 + Inum2 : Inum_1 + Inum_2 * Inum_2 \\ &where\ Inum_1, Inum_2 \geq 0 \end{aligned} \quad (8)$$

Figure 3, illustrate different unique number key generated represented by dot on the graph for around 500 pair of numbers taken as key1 and key2. The numbers included in pairs which are tested are as small as two digit number till twelve digit numbers. At the other end when given a key value, then key1 and key2 may also be generated easily.

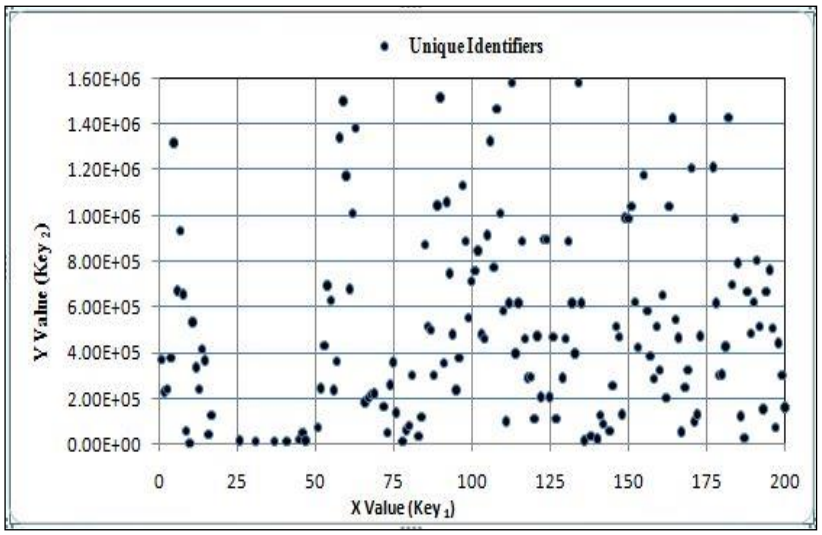

Figure 3. Unique Identifier generated for 500 different source-destination pairs

3. RESULTS AND DISCUSSION

The experiment is carried out in NS-2.35[27]. Simulation area taken is 1500m x 1500m consisting of 50 mobile nodes. All the nodes are initialized to 100J and nodes are arbitrarily positioned with Random Waypoint Mobility Model with mobility in the range of 0m/s to 5m/s. 250m is taken as communication range and data rate as 2Mbps. AODV is used as fundamental routing protocol and IEEE 802.11 as medium access control protocol. The method discussed is measured in terms of packet overhead, packet processing time, energy saved during this process and percentage of extra payload sent. The method is compared with AODV, S-AODV and proves positively good in terms of the three metrics packet overhead, packet processing time and energy consumption considered for evaluation. Packet overhead is defined as the proportion of number of bytes in header to payload length. Packet processing time is measured as the time the node enters at input interface of a node to the time when the packet is put out at the output interface. Total energy consumed is calculated as the difference between remaining energy and initial energy.

The graphs in Figure 4 and 5 shows that the number of payload transmitted are more by 3 % to 5% in AODV with IPv4/UDP and IPv4/TCP. Graph result is shown for 50 packets to 200 packets along X-axis and number of bytes of payload sent along Y-axis. Payload carried in primary packet will be 1460bytes (1518-40-18) and secondary packet will be 1488 bytes (1518-12-18) for IPv4 packets. Payload carried in primary packet will be 1433 bytes (1518-67-18) and secondary packet will contain 1473 bytes (1518-27-18) for IPv4/TCP.

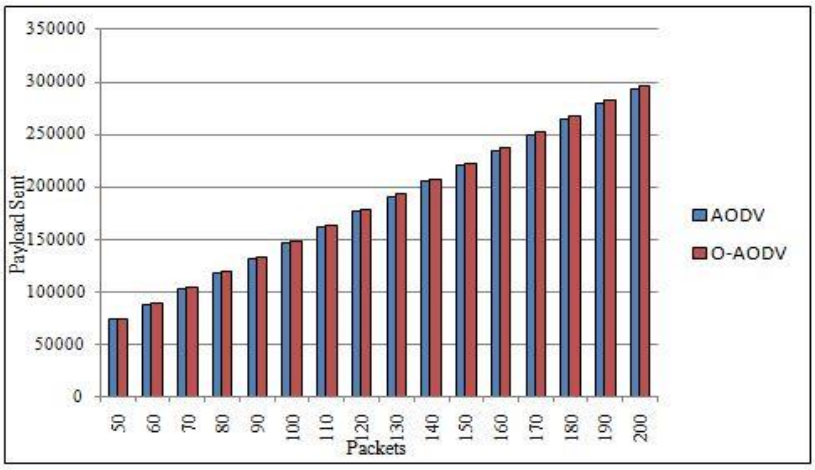

Figure 4. Payload Sent in AODV and O-AODV with IPv4/UDP

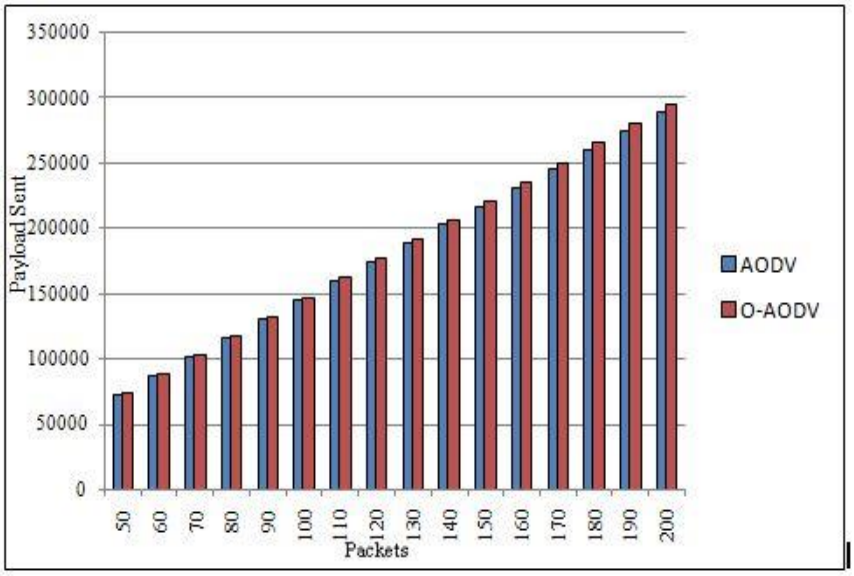

Figure 5. Payload Sent in AODV and O-AODV with IPv4/TCP

Figure 6, 7 and 8 shows the graphical results obtained for Routing load, Energy Consumption and End to End delay of the proposed method, O-AODV compared to AODV, MTPR, and S-AODV. The primary packet header for IP/UDP is increased to 40 bytes which is higher then regular packet header of 28 bytes, at the same time the secondary packet header is reduced to 12 bytes. As the ratio of primary to secondary packets in general is taken as 1: (N-1), where N is the total number of packets sent, (N-1) is the number of secondary packets following a primary packet the routing overhead is reduced reasonably as depicted in Figure 6.

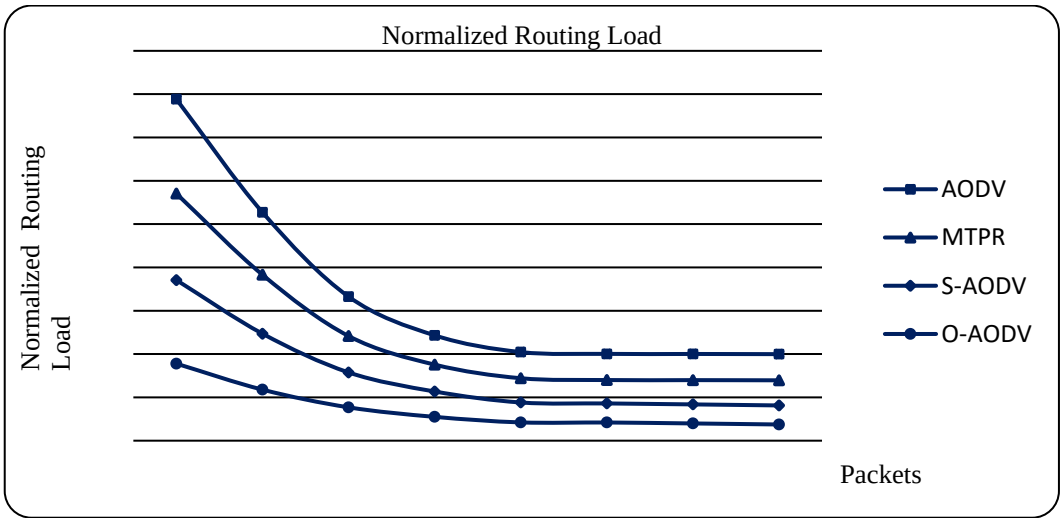

Figure 6. Normalized Routing Load Vs Number of Packets

Energy consumption is also reduced appropriately as shown in Figure 7. Average energy consumption at the end of simulation is calculated and found to be improved by 6% to 8% less energy consumption than S-AODV method. Figure 8 shows that end to end delay is also reduced as the packet processing time is reduced as the number of fields to be checked is minimal. On an average around 3% to 5% improvement is shown with respect to delay. The processing time is reduced which adds to the improvement in overall delay.

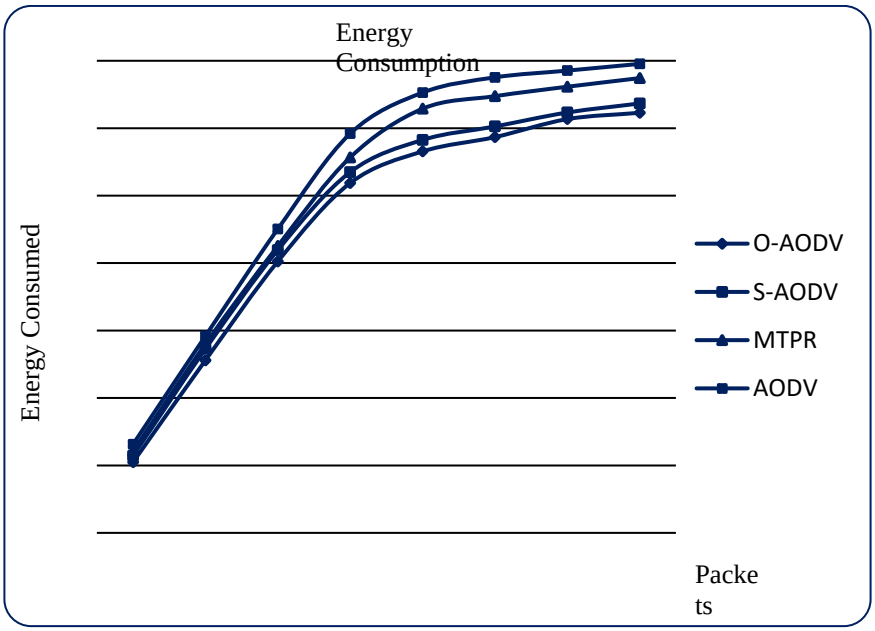

Figure 7. Energy Consumption Vs Number of Packets

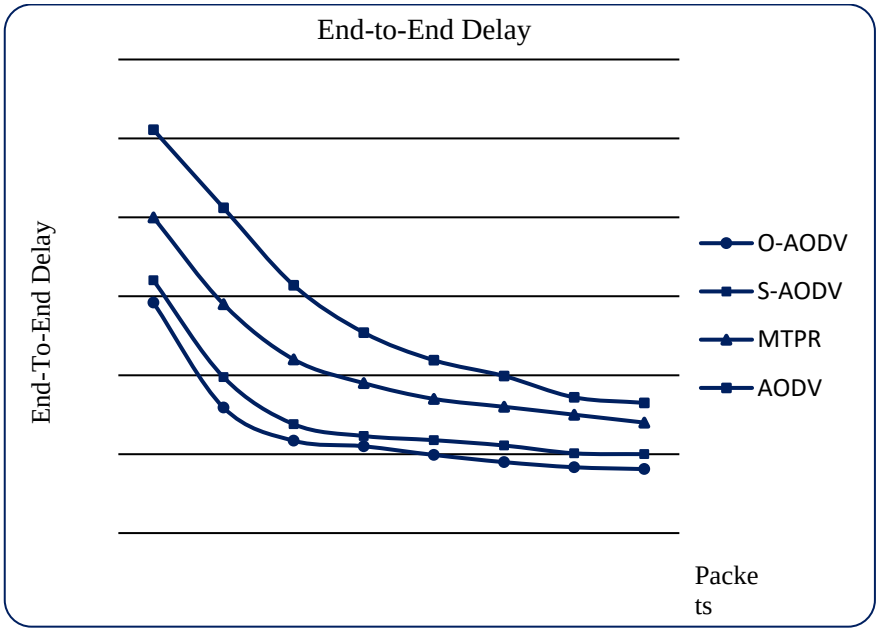

Figure 8. End to End DelayVs Number of Packets

4. CONCLUSION

The work in this paper focusses on a novel method to lessen packet processing time. Delay is becoming more and more noteworthy as networks put into practice complex protocols running on routers. Minimizing host overhead in handling packets and their data achieve better performance. Optimized packet header transmits more payloads with smaller headers and reduced processing time increasing the effectiveness of the existing network. The primary packet is sent with regular header and few additional fields, which will be processed by all intermediate nodes store the necessary details required to further forward all the secondary packets belonging to the same communication session. The proposed work shows improved performance with respect to the three metrics routing overhead, energy consumed and end to end delay. The work is carried out with IPv4 addressing and can be extended to IPv6 addressing. The idea behind

calculating Unique Address Pair Identification Number using the IP addresses can be modified to include the Source IP Address and the Start time of the session when primary packet is generated. This adds security as the Unique Number generated will be valid for only one session communication.

REFERENCES

[1] S. Corson,J. Macker,"Mobile Ad Hoc Networking(MANET) : Rouitng protocol Performance Issues and evaluation Considerations," Network Working Group,RFC: 2501,1999.

[2] Lei Chen,"Protocols for Supporting Quality of Service in Mobile Ad Hoc Networks," Ph.d Thesis,University of Rochester,New York,2006.

[3] Sassan Ahamadi,"The IEEE 802.16m Convergence Sub-Layer,"Mobile WiMAX,2011.

[4] EFFNET AB, "An Introduction to IP Header Compression," White Paper,2004.

[5] Rahul Desai, B P Patil, Davinder Pal Sharma, "Routing Protocols for Mobile Ad Hoc Network: A Survey and Analysis". Indonesian Journal of Electrical Engineering and Computer Science. vol. 7, (3), 2017, pp. 795-801.

[6] S. Syes Abdul Syed,T. Senthil Kumaran,"Extremely Vibrant Routing Scheme for Mobile Adhoc Network", Indonesian Journal of Electrical Engineering and Computer Science,vol. 9(2), 2018,pp. 306-310.

[7] Hasan Awni Shakir,"Enhancement of Energy Control Routing Protocol for Mobile Ad Hoc Network Based on Hybrid Particle Swarm Optimization with Ant Colony-based Energy Control Routing", Indonesian Journal of Electrical Engineering and Computer Science,vol. 8,(2),2017, pp. 308-314.

[8] Bindiya Bhatia, M. K. Soni, Parul Tomar,"Extended Bandwidth Optimized and Energy Efficient Dynamic Source Routing Protocol in Mobile Ad-hoc Networks", International Journal of Electrical and Computer Engineering (IJECE), vol. 8, (3), 2018, pp.1460-1466.

[9] Rahul Desai, B P Patil, Davinder Pal Sharma. Learning Based Route Management in Mobile Ad Hoc Networks. Indonesian Journal of Electrical Engineering and Computer Science. vol. 7, (3), 2017, pp. 718-723.

[10] Shahrukh Khalid,Athar Mahboob,Choudhry Fahad Azim,Aqeel Ur Rehman,"Design and Realization of a Novel Header Compression Scheme for Ad Hoc Networks," ETRI Journal,vol. 38, (5), 2016,pp. 922-933.

[11] Jeffrey S. Chase, Andrew J. Gallatin,Kenneth G. Yocum,"End System Optimizations for High Speed TCP," IEEE Communications Magazine,2001,pp. 68-74.

[12] Emre Ertekin,Chris Christou,Booz Allen Hamilton,"Internet Protocol Header Compression,Robust Header Compression, and Their Applicability in the Globl Information Grid," IEEE Communications Magazine,2004,pp.106-116.

[13] Henrik Wessing,Henrik Christiansen,Tina Fjelde,Lars Dittmann,"Novel Scheme for Packet Forwarding Without Header Modifications in Optical Netwoks," Journal of Lightwave Technology,vol. 20,(8),2002,pp. 1277-1288.

[14] Hooshiar Zolfagharnasab,"Reducing Packet Overhead in Mobile IPv6," International Journal of Distributed and Parallel Systems,vol. 3,(3),2012,pp. 1-8.

[15] Fahim A Ahmed Ghanem,Vilas M. Thakare,"Optimization Of IPv4 Packet's Headers," International Journal of Computer Science Issues,vol. 10,(1),2013,pp. 294-302.

[16] Bow-Nan Cheng,John Zuena,Jim Wheeler,Scott Moore,Brian Hung,"MANET IP Header Compression," *Proceedings of the IEEE Military Communications Conference*,2013,pp.494-503.

[17] M. Tamilarasi,S. Chandramathi,T. G. Palanivelu,"Overhead Reduction and Energy Management in DSR for MANETs*" proceedings of Third international Conference on COMmunication, COMSWARE*,Bangalore,2008.

[18] Scott D, Toh C, Cobb H," Performance Evaluation of battery life aware routing schemes for wireless ad hoc networks," *Proceedings of IEEE ICC*, vol. 9, 2001,pp.2824-2829.

[19] W. Su, S. J. Lee, M Gerla,"Mobility Prediction and Routing in Ad Hoc Wireless Networks," International Journal of Network Management, vol. 11,(1),2001,pp.3-30.

[20] Bhagyashri R Hanji,Rajashree Shettar,"Stable Reduced Link Break Routing Technique in Mobile Ad Hoc Network," *Proceeding of the Second International Conference ICACDS*,Dehradun,2018.

[21] Mobin Iftekharul,"Energy Efficient Packet Size Optimization for wireless Ad Hoc Networks,"Ph.d Thesis,Queen Marry University of London,United Kingdom,2013.

[22] "Network Routing: Algorithms,Protocols, and Architectures,"2nd Edition,Elsevier,2010.

[23] Steven Pigeon," Pairing Functions," FromMathworld-A Wolfram Web Resource,created byvEric W. Weisstein.

[24] Matthew Szudzik,"An Elegant Pairing Function," *Proceedings of the Wolfram Science NKS 2006 Conference*,2006. Available from http://szudzik.com/ElegantPairing.pdf.

[25] Dovier,Agostino,Enrico Pontelli, and Gianfranco Rossi,"Set unification", Theory and Practice of Logic Programming,2006.

[26] Kenneth W Regan,"Minimum-Complexity Pairing Functions," Journal of Computer and System Sciences,vol. 45,2006,pp. 285-295.

[27] Kevin Fall,Annan Varadhan, The NS Manual, The VINT Project, 2011.

BIOGRAPHIES OF AUTHORS

Bhagyashri R Hanji received her B.E and Mtech degree from Visvesvaraya Technological University, Belagavi. She is currently working towards her Ph. D. degree dealing with improving the Quality of Service in Mobile Ad Hoc networks. She is currently working as Assistant professor at Global Academy of Technology, Bengaluru.

Dr. Rajashree Shettar is currently working as Professor, Dept. of Computer Science, R. V College of Engineering, Bengaluru. Her research work focuses on "Knowledge Discovery in Semi-structured Data". She has around 45 publications in various International Journals and Conferences. She has authored a book on "Sequential Pattern Mining from Web Log Data: Concepts, Techniques and Applications of Web Usage Mining", published by LAMBERT Academic Publishing company, Germany and co-authored book chapters