Communications on Applied Electronics (CAE) – ISSN : 2394-4714
Foundation of Computer Science FCS, New York, USA
Volume 3– No.5, November 2015 – www.caeaccess.org

Optimized Routing by Excluding Selfish Nodes for MANET

Smita Rukhande
Student
K.J.Somaiya College of Engineering
Vidyavihar, Mumbai

Prasanna Shete
Associate Professor
K.J.Somaiya College of Engineering
Vidyavihar, Mumbai

ABSTRACT
The Mobile nodes in MANET's are free to move randomly. Therefore, the network topology may change rapidly. Routing protocols for MANET are used for delivery of data packets from source to the desired destination. Routing protocols AODV are also designed based on the assumption that all the participating nodes are fully cooperative. However, due to the scarcity of available battery, node misbehaviors may exist. One such routing misbehaviors is that some nodes may be selfish as they take assistance of other nodes in route discovery and maintenance process, but refuse to forward the packet in order to save its resources. To solve this problem we propose a selfish node detection algorithm SIAODV that finds selfish node and calculates risk factor to find the optimum path (minimum risk path) for data forwarding. Most of the existing work focused on identification of selfish nodes but not on the selection of optimum path (minimum risk path) for data forwarding because shortest path may not be suitable in all the cases. The proposed solution presents a technique that identifies selfish nodes and finds the minimum risk path for data forwarding in Ad hoc On-demand Distance Vector (AODV) routing protocol for MANETs.

Keywords
Mobile Ad-Hoc Networks (MANET), Selfish nodes, Ad-Hoc on Demand Distance Vector Routing Protocol (AODV), RREQ.

1. INTRODUCTION
A mobile ad hoc network (MANET) is a collection of mobile nodes with no pre-established infrastructure forming a temporary network. Each device in a MANET is free to move independently in any direction, and will therefore change its links to other devices frequently. Because of the limited transmitter range of the nodes, multiple hops may be needed to reach other nodes. Due to the mobility of the nodes, the structure of the network changes dynamically [1]. In MANET, each node participates in routing by forwarding data for other nodes, and so the determination of which nodes forward data is made dynamically based on the network connectivity. The efficient node-energy utilization in mobile ad-hoc networks is essential as ad-hoc nodes operate with limited battery power. In ad-hoc networks, nodes perform the function of hosts as well as router as there is no existing infrastructure. Thus, failure of node in ad hoc network leads to loss of communication in the network. Mobile Ad Hoc networks find its application in many areas and are useful for many cases.

Routing algorithms designed for MANET such as DSR and AODV are based on the assumption that every node forwards every packet. But some of the nodes may act selfishly to conserve their resources e.g. battery capacity. These nodes use the network and its services but they do not cooperate with other nodes. Such selfish nodes do not consume power, memory and also bandwidth for retransmitting the data of other nodes and they reserve them only for themselves [2]. Different techniques are suggested in the literature to detect and overcome the problem of node misbehavior. These techniques can be classified broadly into two categories: Credit- based Schemes and Reputation based Schemes.

The basic idea of credit based techniques [3], [4] and [5] is to provide incentives for nodes to faithfully perform networking functions. In order to achieve this goal, virtual (electronic) currency or similar payment system may be set up. Nodes get paid for providing services to other nodes. When they request other nodes to help them for packet forwarding, they use the same payment system to pay for such services. In a reputation based technique [6], [7], [8] and [9], each node is responsible for monitoring the transmission of a packet to neighbour node, or obtaining the status of other nodes from a centralized node on the network. If a node successfully contributes in the transmission of data by forwarding data packets, the reputation of the node is increased, or if the node discards the packet by dropping it, the reputation is decreased. After the nodes reputation drops below a threshold set by the developer, the node is either punished or ignored. Also energy-efficient routing is an effective mechanism for reducing energy cost of data communication in wireless ad hoc networks. [10], [11], [12] [13] presents energy-efficient routing protocols. In [10], AODV protocol is extended with SARSA (i.e. State-Action-Reward-State-Action) RL mechanism [14], in which the mobile nodes use learning based adaptive energy efficient RREQ forwarding policy. The residual lifetime of mobile node in seconds, at time step t is considered as a "state" st calculated by taking the ratio of residual energy (REt) and energy drain rate (DRatet) at time interval t. Each node monitors its energy consumption for T second interval from which energy drain rate is calculated. High state value means the node has a high expected lifetime and low state value corresponds to node with poor expected lifetime. Based on the state value, the node performs action i.e. forwarding of RREQ packets.

Different from the previous work, a scheme is presented which finds the selfish nodes as well as minimum risk path for data forwarding. The proposed algorithm is implemented as SIAODV routing protocol by modifying existing AODV routing protocol in Qualnet simulator.

2. RELATIVE TECHNIQUES
In order to understand the operation procedure of AODV routing protocol and the definition of the selfish node. In this section the introduction about the working of AODV routing protocol and the behaviors of selfish nodes is given.

2.1 AODV Routing Protocol

In AODV four control messages are defined to establish and maintain the routes to the destination. These control messages [15] include RREQ (Route Request) message, Hello message, RERR (Route Error) message and RREP (Route Reply). Periodically a hello message is broadcasted by every node in the network to all its neighbors to tell that it is alive. Whenever a neighboring node receives a Hello message, the neighbor node includes the data about the node which sends a Hello message into its routing table. If a node wants to communicate with some other node, the source node will check destination node in its routing table. Route Request (RREQ) packet is broadcasted by the source node to all its neighbors in case if the routing table does not contain destination node. Every neighboring node likewise rebroadcasts the gained route request (RREQ) messages to its neighbors. Through along these lines over and over until the destination node is reached.

AODV Message format

Type	**J**	**R**	**G**	**D**	**U**	**Reserved**	**Hop Count**
RREQ ID							
Destination IP Address							
Destination Sequence Number							
Originator IP Address							
Originator Sequence Number							

Fig 1: AODV Route Request (RREQ) Message Format

If the neighbor node accepts the route reply packet (RREP), it likewise replies conversely the Route reply packet to the former neighbor node as per the data in its routing table. The transmission path can be created at the point when the route reply (RREP) message is sent again to the originating node.

Throughout the information transmission, if in this transmission way a node is not able to communicate with the neighbor nodes, then a route error(RERR) message is sent by this node to the source node and the data that belongs to this transmission way is deleted from its routing table. The source node will retransmit RREQ packet for building a new transmission path when it receives a route error (RRER) message considering that the transmission path to the desired destination node has broken.

Type	**R**	**A**	**Reserved**	**Prefix SZ**	**Hop Count**
Destination IP Address					
Destination Sequence Number					
Originator IP Address					
Life Time					

Fig 2: AODV Route Reply (RREP) Message Format

2.2 Selfish Node Behaviors

Selfish nodes are inclined to get the greatest profits from the networks and at the same time these nodes trying to conserve their own resources like bandwidth, battery life [16]. A selfish node only communicates to other nodes if its data packet is required to send to some other node and refuses to cooperate other nodes whenever some data packets or routing packets are received by it that it has no interest in. Hence selfish node drop data packets refuse to retransmit routing packets that they have no interest in it.

The selfish nodes behaviors in AODV routing protocols can be as follows:

Nodes which do not send Hello packet: The principle target of this sort of selfish node is hiding itself and to abstain from being included in the others transmission way Nodes which do not forward RREP messages: Because of this kind of selfish behavior whole network will be paralyzed. In AODV, the source node will get a RREP message from the destination node through some intermediate nodes to establish a complete transmission path, but here the communication path will not be established because this kind of selfish nodes will not forward the RREP message. Hence the source node will broadcast Route Request (RREQ) message continuously.

Nodes which do not forward Data messages: The misbehavior of this type of selfish node impacts the performance of MANET by dropping all the data messages that are received by these nodes. Instead of relaying these data messages these will be dropped.

Nodes forwarding RREQ messages with delay: When this kind of selfish node gets a Route Request(RREQ) message it forwards this RREQ message after some lag near the upper bound of time out for not to participate in a route.

Nodes which do not forward RREQ messages : In MANET, if this type of selfish nodes receives some RREQ messages, then instead of forwarding these RREQ messages, these messages are dropped and thus these kind of selfish nodes skips being the route member for other nodes. Thus avoiding forwarding these messages for others as a result more nodes are required for building a transmission path.

3. PROPOSED STRATEGY

In proposed SIAODV scheme, every node maintains neighbour table that contains information about the neighbour nodes which are forwarded RREQ packets as shown in table I. Neighbour table contains information such as residual battery and credit risk of Neighbour nodes. The node extracts the individual's residual battery and the credit risk of previous node. Whenever an intermediate node hears a request from its neighboring node to forward a RREQ packet, it will check the neighbor table that whether requested node has forwarded the RREQ packets. If the request is from selfish node it will discard the packet. Else if the request is from a non-selfish node then it add forwards the incoming packet to the neighbouring nodes.

Table 1. Neighbour Table

Neighbour Address	Number of RREQ Packet forwarded	Battery Life	Selfishness alarm P	Total packet forwarded by node	Credit Risk

At the destination node, it checks the credit risk of incoming RREQ packets and select the path with minimum credit risk for RREP packet. And select this optimum path to forward the data packet. Proposed scheme consists of two parts: 1) Detecting selfish nodes, 2) Finding optimum path

Each node detects the selfish nodes based on credit risk score and by excluding selfish nodes forward data packet along minimum credit risk path. The CR score is updated during the RREQ processing phase.

3.1 Detecting Selfish Node

The notion of credit risk can be described by the following equation:

$$\text{Credit Risk} = (\text{Expected risk})/(\text{Expected value}) \quad (1)$$

In this proposed strategy, each node calculates the selfishness alarm (P) for its neighboring nodes and calculate CR score for its neighboring nodes based on selfishness alarm (P). Selfish features are divided into two categories: node- specific and RREQ processing-specific. Node-specific features can be explained by considering the following case:

A node may share or consume its battery for forwarding traffic of other nodes. Amount of battery or number of RREQ forwarded by node can be used to represent the degree of selfishness. When node Ni observes that node N_k has sufficient battery and forwards the RREQ packet then node N_k will be treated as a valuable node by node Ni. At the RREQ processing-specific feature, the ratio of selfishness alarm of N_k on N_i is utilized which is denoted as P_{ki} , which is the ratio of N_i's request being not served by the expected node N_K due to selfishness in battery consumed.

$$P_{ki} = \frac{\text{Ni's RREQ packets not forwarded by node } N_k}{\text{Total no. of RREQ packets transmitted by Ni}} \quad (2)$$

That is Ni's RREQ packet not forwarded by the expected node N_K due to selfishness in battery consumed.

$$P_{ki} = \frac{\text{Ni's RREQ packets not forwarded by node } N_k}{\text{Total no. of RREQ packets transmitted by Ni}} \quad (3)$$

Thus, the RREQ processing-specific feature can represent the expected risk of a node. For instance, when Pki gets larger, node I will treat K as a risky node because a large Pki means that K cannot serve I's requests due to selfishness in battery consumed.

To effectively identify the expected node (s), node I should know the (expected) status of other nodes' residual battery.

Using the described features, modify (1) into (4):

$$\text{CRki} = \text{Pki} + \frac{(\text{residual battery} - \text{average battery})}{\text{average battery}} \quad (4)$$

Each node has its own threshold of CRki. If the measured CRki exceeds, node will be detected as a selfish node. The value of Pki and remaining battery of nodes is updated at every RREQ processing.

Algorithm 1. Pseudo code to detect selfish nodes

00: / Ni detects selfish nodes with this algorithm /

01: detection () {

02: for (each connected node Nk) {

03: if (CRik>= ð) Nk is marked as selfish node;

04: else Nk is marked as non-selfish node ;}}

Algorithm 2. Pseudo code to update selfish features

At every RREQ processing time

/ When Ni issues a RREQ /

01: update credit risk score () {

02: while (during RREQ interval time) {

03: if (an expected node Nk forwards RREQ packets

04: decrease Pki;

05: if (an unexpected node Nj forwards RREQ) {

06: Pij =0; } }

07: if (an expected node Nk does not forward

RREQ) {

08: increase Pki;}}

As described in Algorithm 2, Ni maintains credit risk score during each RREQ processing phase. When Ni receives RREQ packet, Ni checks whether it is from selfish node or non-selfish node. If RREQ is from selfish node, then Ni does not forward RREQ otherwise Ni flood RREQ to neighbor's nodes. Before flooding RREQ, node Ni adds residual battery and credit risk in RREQ packet. Whenever Ni detects the selfish behavior of Nk, it modifies Pki. If Nk forwards the RREQ as expected, however, only Pki will be decreased. Note that, in case an unexpected node Nj forward RREQ, Ni will set Pj=0. That is, when unexpected nodes do not forwards RREQ it will not affect the selfish features of expected nodes. Note also that Ni may receive multiple RREQ from unexpected and/or expected nodes. When expected node Nk does not forward RREQ, Ni observes Nk's selfish behavior and modifies Pki. Also the residual battery threshold is introduced where the battery threshold is calculated as the 20% of the initial battery.

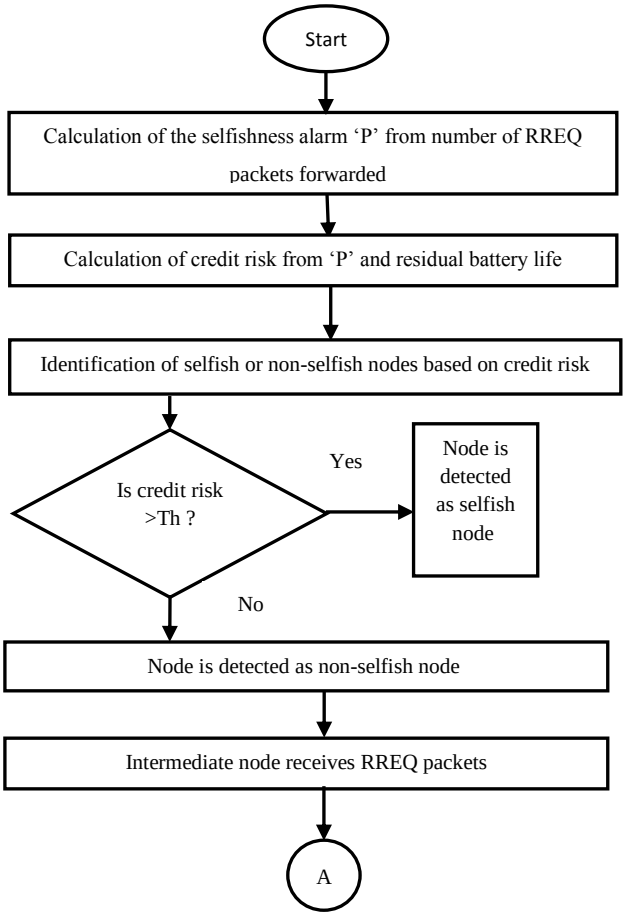
Battery Threshold = 0.20 * Initial Battery (5)

Then this threshold value is compared with the residual battery. If the residual battery is greater than threshold, it can be concluded that intermediate node has sufficient energy to broadcast RREQ message and will successfully reach destination node, so the intermediate rebroadcasts RREQ message. If the residual battery is less than threshold, concludes that the intermediate node do not have sufficient

battery to forward RREQ message and hence the intermediate node will drop the RREQ message and will save its energy. Such critical nodes are not considered as selfish nodes.

3.2 Find optimum path

At destination node, stores multiple RREQs and extract the credit risk from incoming RREQ packets and select path which has less credit risk for RREP packet to complete transmission path and forward data packet along this selected path. Flowchart given in figure 3 describes the working of SIAODV scheme.



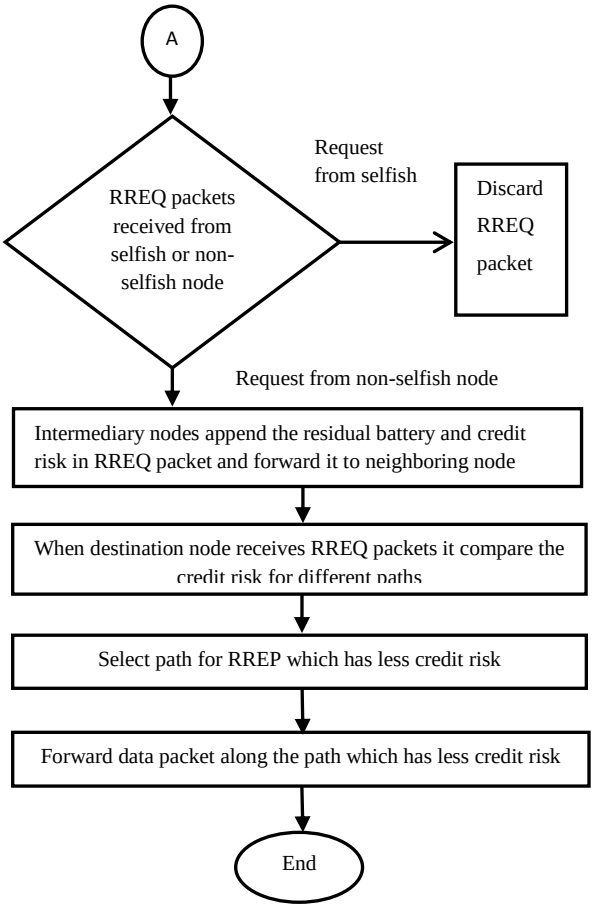


Fig 3: Flowchart of SIAODV scheme

4. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed SIAODV scheme [10] on the top of AODV [14] routing protocol using the Qualnet Simulator is implemented. The performance of SIAODV with SARSA-AODV routing protocol is compared. SARSA-AODV is also implemented on the top of the AODV routing protocol and it is explained in section 1.

The following performance metrics are measured:

1) Average end to end delay

2) Throughput

3) Average Delivery Ratio of data packets (DR).

4) Critical nodes battery dead time

4.1 Simulation for 25 nodes

Simulation parameters setting and simulation results for 25 nodes are described below.

Table 2: Simulation Parameters Setting For 25 Nodes

Parameters	Value
Simulation Time	600 seconds
Number of Nodes	25
Protocols	SIAODV, SARSA, AODV

Node placement strategy	Grid [Grid unit= 300m]
Packet Size	1024 bytes
Network Area	1500m x 1500m
Communication patterns	Constant bit rate (CBR)
Number of sources	1
Data rate	4 packets / second
Critical nodes	5
Selfish nodes	4
Initial energy of non-selfish	12 mAh
Battery threshold for non-	2 mAh
Initial energy of selfish	12 mAh
Battery threshold for selfish	11.9 mAh
Initial energy of selfish	12 mAh
Battery threshold for selfish	11.9 mAh
Initial energy of critical nodes	2.4 mAh
Battery threshold for selfish nodes	2 mAh

Simulation Results for 25 nodes

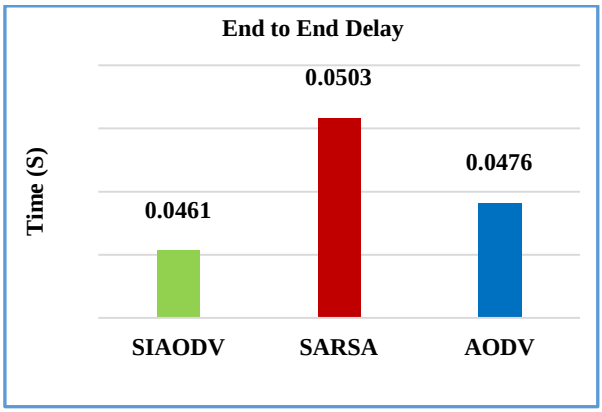


Fig: 4 End to end delay

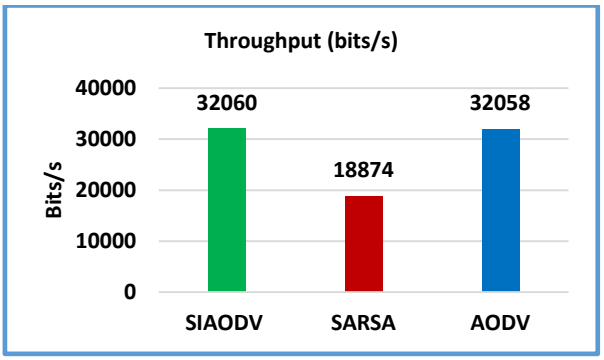


Fig 5: Throughput

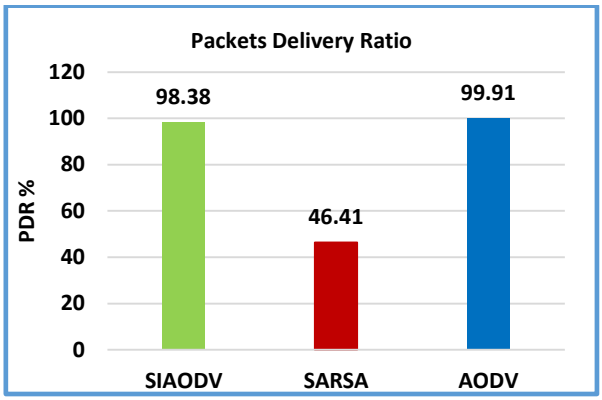


Fig 6: Packet delivery ratio

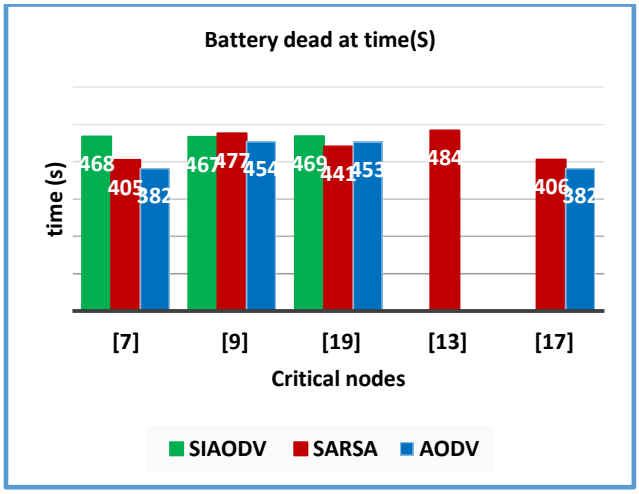


Fig 7: Critical nodes battery dead time

4.2 Simulation for 30 nodes

Simulation parameters setting and simulation results for 30 nodes are described below.

Table 3. Simulation Parameters Setting For 30 Nodes

Parameters	Value
Simulation Time	600 seconds
Number of Nodes	30
Protocols	SIAODV, SARSA, AODV
Node placement strategy	Grid [Grid unit= 300m]
Packet Size	1024 bytes
Network Area	1500m x 1500m
Communication patterns	Constant bit rate (CBR)
Number of sources	1
Data rate	4 packets / second
Critical nodes	3
Selfish nodes	4
Initial energy of non-selfish nodes	12 mAh

Battery threshold for non-selfish nodes	2 mAh
Initial energy of selfish nodes	12 mAh
Battery threshold for selfish nodes	11.9 mAh
Initial energy of selfish nodes	12 mAh
Battery threshold for selfish nodes	11.9 mAh
Initial energy of critical nodes	2.4 mAh
Battery threshold for selfish nodes	2 mAh

Simulation results for 30 nodes

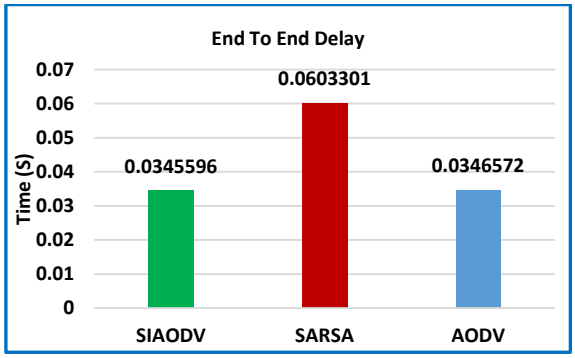


Fig 8: End to end delay

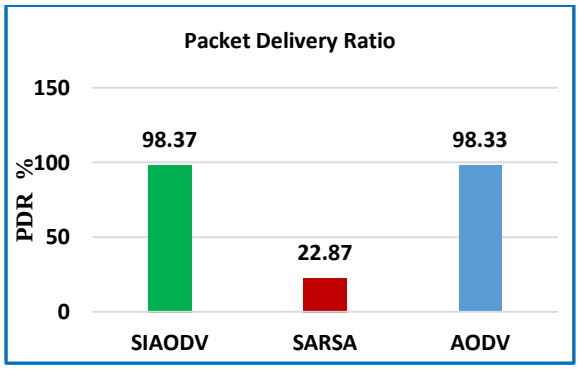


Fig 9: Packet delivery ratio

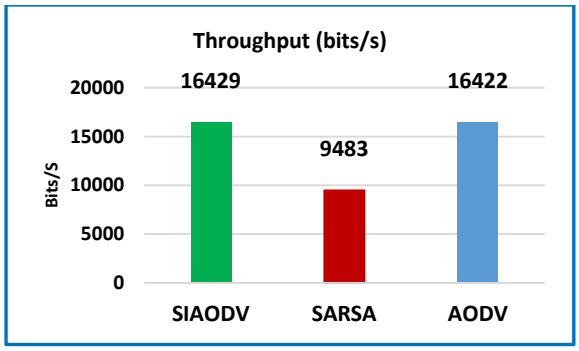


Fig 10: Throughput

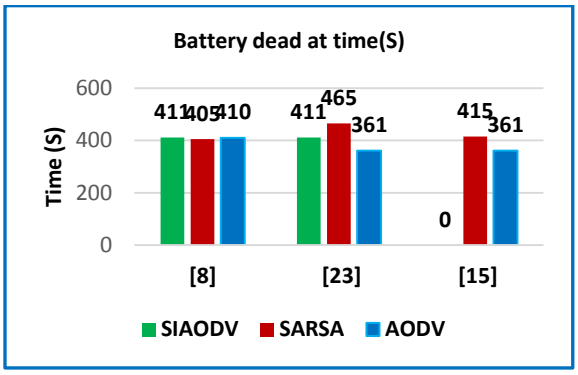


Fig 11: Critical nodes battery dead time

5. CONCLUSION

Multi-hop wireless Ad Hoc Networks have gained acceptance due to their ad hoc capabilities. These networks are characterized by dynamically changing topology due to high mobility of the nodes. Hence routing is a crucial issue in these networks. In this optimized routing scheme, the report described the issues of selfish nodes in ad hoc networks which degrades the performance of AODV routing protocol and provide methods to improve AODV performance.

In this Optimized routing scheme firstly identify selfish nodes based on number of RREQ packet forwarded and residual battery life of neighboring nodes. Secondly intermediate nodes calculate risk factor from its own number of RREQ packets forwarded and residual battery life. Lastly it finds a minimum risk path for RREP and forwards data packet along this path.

In AODV, nodes keep on losing their energy irrespective whether the node is critical or not until the simulation is not over. In contrast, nodes in SIAODV will first check its residual energy then it makes a forwarding decision. If it has sufficient energy, it forwards the RREQ to the next node; else it will simply drop any further incoming RREQ's. With this routing scheme the critical nodes are saved and data packets are forwarded along minimum credit risk path. In addition, there is also a strong decrease in the average number of RREQ's forwarded by each node especially by the critical nodes, also simulation results shows that the method increases the end-to-end delay and packet delivery ratio.

6. REFERENCES

[1] S. Corson and J. Macker, “Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations” RFC 2501, January 1999.

[2] Shin Yokoyama, Yoshikazu Nakane, Osamu Takahashi, Eiichi Miyamoto, “Evaluation of the Impact of Selfish Nodes in Ad-Hoc Networks and Detection and Countermeasure Methods” 7th International Conference on Mobile Data Management, IEEE 2006.

[3] L. Buttyan and J.P. Hubaux, “Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks,” ACM/KUler Mobile Networks and Applications, Vol. 8 No. 5 2003

[4] S. Zhong, J. Chen and Y. R. Yang, “Sprite: A Simple Cheat-Proof, Credit Based System for Mobile Ad hoc Networks”, Proc. INFOCOM, Mar-Apr 2003.

[5] Jae-Ho Choi, Kyn-Sun Shim, SangKeun Lee and Kun-Lung Wu, "Handling Selfishness in Replica Allocation over a Mobile Ad-Hoc Network", IEEE Transaction on Mobile Computing, February 2012.

[6] Sergio Marti, T.J. Giuli, Kevin Lai, and Mary Baker "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks", ACM 2000.

[7] Sonja Buchegger, Jean-Yves Le Boudec, "Performance Analysis of the CONFIDANT Protocol (Cooperation Of Nodes: Fairness In Dynamic Ad-Hoc Networks)" ACM June 2002.

[8] Tiranuch Anantvalee and Jie, "Reputation-Based System for Encouraging the Cooperation of Nodes in Mobile Ad Hoc Networks" , 2007 IEEE

[9] S. Bansal and M. Baker, "Observation-Based Cooperation Enforcement in Ad Hoc Networks", Research Report NI/0307012, Stanford University, 2003.

[10] Saloua Chettibi, Salim Chikhi, "An Adaptive Energy-Aware Routing Protocol for MANETs Using The SARSA Reinforcement Learning Algorithm", 978-1-4673-1727 ©2013 IEEE.

[11] Shivashankar, Hosahalli Narayanagowda Suresh, Golla Varaprasad, And Guruswamy Jayanthi, "Designing Energy Routing Protocol With Power Consumption Optimization in MANET", IEEE Transactions on emerging topics in computing, volume 2, no. 2, June 2014.

[12] Tanmoy Kanti, Halder, Chandreyee Chowdhury, Sarmistha Neogy, "Power Aware AODV Routing Protocol for MANET", 978-1-4799-4363-0/14 2014 IEEE.

[13] Xu Zhen, Xiao Juan , "Energy-aware and Delay-aware QoS Routing in Mobile Ad-Hoc Networks", 978-1-4673-1697-2 2012 IEEE and ICCP 2012 Proceeding.

[14] R. Sutton and A. Barto, "Reinforcement leaming," MIT Press, Cambridge, 1998.

[15] C. E. Perkins and E. M. Royer, "Ad Hoc On Demand Distance Vector (AODV) Routing", Proceedings of 2nd IEEE Workshop on Mobile Computing Systems and Applications, February 1999.

[16] Wu, Lien-Wen, and Rui-Feng Yu. "A threshold-based method for selfish nodes detection in MANET." Computer Symposium (ICS), 2010 International. IEEE, 2010.