Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 01, January-2016

Mobile Ad-Hoc Networks: Analysis of the Impact of Selfish Users in the Performance OLSR Protocol

Diógenes Antonio M. José
Department of Computer Science
Mato Grosso State University (UNEMAT)
Barra do Bugres - MT, Brazil

Diego De Lima Nascimento
Department of Computer Science
Mato Grosso State University (UNEMAT)
Barra do Bugres - MT, Brazil

Fahim Elias C. Rihbane
Institute of Physics
Mato Grosso Federal University (UFMT)
Cuiabá - MT, Brazil

Judith Abi Rached Cruz
Department of Mathematics
Mato Grosso State University (UNEMAT)
Barra do Bugres - MT, Brazil

***Abstract — Mobile Ad-Hoc Networks (or MANETs)* have been widely discussed because of their characteristics such as dynamic topology, autonomous routing and limited energy. It has been challenging to overcome the selfish behavior of *MANETs* participants, who refuse to route packets on behalf of other nodes. However, most papers about selfish nodes conducted this research using reactive protocols. In this paper, we measured the impact of selfish behavior in the *OLSR* protocol and its extensions *OLSR-ML*, *OLSR-ETX* and *OLSR-MD*. The evaluation was made using *Network Simulator-2 (NS-2) version 2.34*, taking into account performance metrics such as throughput, packet loss rate and energy consumption. The results have shown that the selfish strategy is efficient in energy saving, affects the performance of the tested protocols and it can also lead to network collapse.**

Keywords — MANETs; Selfish Nodes; OLSR; Proactive Routing.

I. INTRODUCTION

With the advent of ubiquitous/pervasive computing there is the promise of the Mobile Ad-Hoc Networks (MANETs), to meet the increased demand for ubiquitous access to various types of services, such as healthcare networks [1]. MANETs are described in the literature as a no infrastructure configuration with multi-hop routing, dynamic topology, power limitation and mobile nodes [2]. These features allow each node in the network functions as a router capable of maintaining and participate in the route discovery process and, in addition, forwarding packets on behalf of others.

One of the main drawbacks of the MANETs is the selfish behavior of nodes and this drawback together with the possibility of changes in network card settings may stop its redirection capacity (e.g., disable the packets IP forwarding). According to Athanasiou et al., [3], the main cause of selfish behavior consists of battery limitations of mobile devices. According to Buttyá and Hubaux [4], selfish behavior is different from the malicious behavior as it seeks the network benefits that can be measured quantitatively, such as throughput (Kbps) and power (Joules). In the context of MANETs, it is considered selfish node a mobile device managed by a user who has no interest in forward packets on behalf of other users on the network, acting against the standard protocol. Thus, the selfish node drops packets which does not have it as final destination [5].

There are several studies in the literature that quantify the impact of selfish nodes in MANETs, however most papers conducted this research using reactive protocols (e,g., AODV and DSR). In this sense, the objective of this research is to assess the impact of selfish nodes, with emphasis on those that drop data packets, in the Optimized Link State Routing (OLSR) and its extensions: OLSR Expected Transmission Count (OLSR-ETX) [6], OLSR Minimum Loss (OLSR-ML) [7] and OLSR Minimum Delay (OLSR-MD) [8]. Thus, for the purpose of clarification, the term "extensions" refers to variations OLSR a criterion choice of route different from the original one (hop-count) RFC 3626 [9]. The evaluation was done by simulation using discrete event simulator NS-2 (version 2.34). Several simulations were performed in order to verify the negative impact of selfish action in the following performance metrics: throughput, packet loss rate and power consumption. The results showed that the selfish strategy is an efficient energy saving method to decrease the throughput and increase the packet loss rate.

The rest of this paper is organized as follows. Section II presents the related work. Section III presents a brief description of OLSR and extensions used in carrying out the simulations. Section IV describes the methodology used in conducting simulations. Section V presents the results obtained. Finally, in Section VI, the conclusions and the possibilities for future works are presented.

II. RELATED WORK

Many researchers have investigated the impact of selfish nodes in MANETs, as Yokoyama et al., [10] which classifies this behavior into three categories: Deny of Service (DoS), careless and greedy. In addition, the authors

This research was funded by Mato Grosso State Foundation (FAPEMAT) in partnership with Mato Grosso State University (UNEMAT).

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 01, January-2016

state that the most damaging kind of selfish node to the network is the one who participates in the routes construction process, however, this type of node drops the packet without forwarding them. The work in question proposes methods to detect selfish behavior patterns, however, some of the proposed methods make false positives, for example, the proposed method for detecting selfish nodes which drops data packets had no effect.

The study by Kothari and Chaturvedi [11] identifies two types of selfish behavior in MANETs, which does not forward packets to other nodes and which disables the device wireless function. The authors measured the selfishness impact in Dynamic Source Routing (DSR) protocol in both types of behavior. The second quoted, saves more energy than the first mentioned, it is noteworthy that the research did not consider any metric performance but the impact of selfish action on the residual energy of the node.

The work presented by Babakhouya et al., [12] describes two types of selfish behavior, the one which participates in routing functions dropping data packets silently and the one that can discard all messages Relay Route Request (RREQ), or do not forward a message Relay Route Reply (RREP) to the destination. The authors found that the selfish node that drops packets silently is the most damaging to packet delivery rate. The experiments did not take into account the throughput.

The work of Gupta et al., [13], argues that to thwart the selfish action is necessary to use a mechanism to encourage cooperation, as described in Robert et al., [14], which should motivate and encourage the packet forwarding on the network and at the same time, detect and delete nodes from a selfish route. It was observed in experiments that 100% of selfish nodes the throughput is not completely empty because there will still be direct links. To measure the DSR performance in the presence of selfish nodes the research used the Dijkstra algorithm and did not take into account the nodes mobility.

Other researches that also address the selfish behavior, not only quantifying the impact of this but proposing mechanisms to force cooperation and isolate selfish nodes, can be found on [14] [15].

III. OLSR PROTOCOL

The OLSR protocol, RFC 3626 [9], consists of a routing protocol developed for use in Ad-Hoc networks, MANETs and Mesh [16]. The OLSR is proactive which implies periodicity in the control messages flooding, for instance: Topology Control (TC) and HELLO. The main advantage of the OLSR is the use of MultiPoint Relays (MPR), Figure 1.

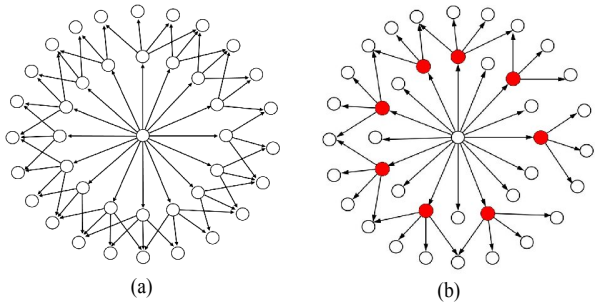


Fig.1. Message flooding without MPRs nodes (a) and with MPRs nodes (b) [17].

The MPRs are nodes selected by their one hop neighbors in order to spread messages about network information (e.g., network topology by TC message), a process called flooding. Using MPR the amount of control messages broadcasted on the network is reduced. The OLSR routing table uses the *hop-count* metric, which uses a graph algorithm, *Breadth-First Search*, to count the hops between the source and destination.

A. OLSR Extensions

The OLSR can be a protocol that prioritizes best effort traffic, and could end up selecting paths with low quality (e.g., low throughput, high delay, high packet loss rate, etc.) [18] [8]. As a result, a variety of extensions have been proposed to improve the QoS in OLSR.

B. OLSR Expected Transmission Count (OLSR-ETX)

The OLSR-ETX consists of a variation of the OLSR which uses the ETX routing metric, proposed by De Couto et al., [6], whose goal is to find routes with the best throughput. This metric predicts the expected number of transmissions (including retransmissions) so that a frame arrives at the destination successfully. The calculation of ETX is obtained from the ratio of total frames sent in the link divided by the total frames that were successfully confirmed (e.g., $d_f * d_r$), as in (1). Thus, the total ETX path is the sum of all links that are part of the same path, Figure 2 (e.g., $ETX_{Total} = ETX_{A\rightarrow B}+ETX_{B\rightarrow C} +ETX_{C\rightarrow D}+ETX_{D\rightarrow E} +ETX_{E\rightarrow F}$).

$$ETX = \frac{1}{d_f * d_r} \quad (1)$$

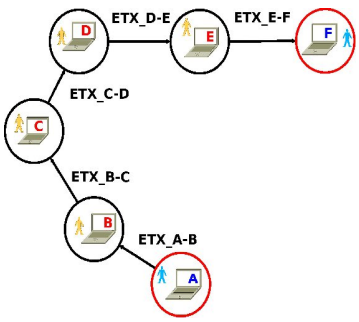


Fig.2. ETX links from A to F nodes.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 01, January-2016

C. OLSR Minimum Loss (OLSR-ML)

OLSR-ML is a OLSR variant that makes use of the ML metric proposed by Passos et al., [7]. This metric aims to find paths with the highest probability of success in the packet delivery rate. Thus, the route calculation is performed using the link delivery rates in both directions, back and forth, (2).

$$ML = d_f * d_r \quad (2)$$

Figure 3 shows a situation in which there are two routes to the same ETX, (e.g., $ETX_{Total} = 4$). Thus, the OLSR-ML chooses the highest number of hops path, which has 100% success probability (e.g., $ML_{Total} = P_{(A-B)} \times P_{(B-C)} \times D_{(C-E)} \times P_{(E-F)} \rightarrow 1$), since the path with the least number of hops (e.g., Figure 3, link 1 A-D and link 2 D-F) has 25% success probability. This means that half of the packets sent on each link in the A-D-F path has not been confirmed by the data link layer.

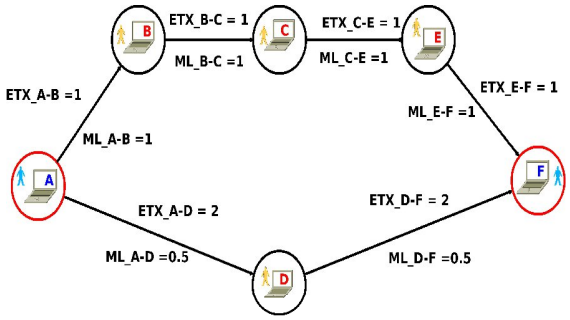


Fig.3. ML links from A to F nodes.

D. OLSR Minimum Delay (OLSR-MD)

The MD metric implemented in OLSR was proposed by Cordeiro et al., [8] and aims to select paths with the lowest transmission delay. The metric in question combines a AdHoc Probe variation, used to measure the packet pair dispersion. The AdHoc Probe uses packet pairs to measure the delay of a link in one way, One Way Delay (OWD). Thus, both the routing table calculation as the MPRs choice may use as criterion the transmission delay. The OWD calculation is as shown in (3). TABLE I shows the (3) component meaning.

$$T = (T_{recv\,2,i} - T_{send,i} - \delta) - (T_{recv\,1,i} - T_{send,i} - \delta) \rightarrow T_{recv\,2,i} - T_{recv\,1,i} \quad (3)$$

TABLE I. EQUATION (3) COMPONENT MEANING.

Component	Meaning
δ	Node clock compensation.
$T_{send,i}$	Packet sending time signed by *i* emitter.
$T_{recv1,i}$ and $T_{recv2,i}$	Packet pair receiving time signed in the receiver.

Furthermore, each node should calculate, adaptively, in order to maintain its neighbors delay information, Smoothed Transmission Delay (STD), as in (4), TABLE II.

$$STD_{n,x} = \sum_{i=0}^{n} \alpha (1-\alpha)^{n-i} D_{i,x} \quad (4)$$

TABLE II. EQUATION (4) COMPONENT MEANING.

Component	Meaning
α	Indicates delay sensibility. α is in the]0,1[interval. The bigger the value the bigger the delay sensibility.
D	Delay in the *x* node direction in relation to the current node.
n	*Nth* pair of packet sent by the *x* node.

Thus, just as the OLSR-ETX and OLSR-ML, the OLSR-MD broadcasts OWD through the network by means of the HELLO and TC modified messages.

IV. METHODOLOGY

For evaluating impact of selfish nodes in the proposed scenarios, it was taken into account the difference in the performance metrics between the best case (without selfish nodes) and worst case (with 60% of selfish nodes). The purpose of this is to verify which of the extensions has the highest performance loss facing selfish nodes. Thus, the methodology for conducting the simulations was determined by:

- Simulator: it was used the network simulator NS-2 version 2.34, for being one of the simulators commonly used in the MANETs evaluation;
- Mobility Model: it was used the Random Walk with Reflection (RWR) because in the real world nodes (users) deviate from obstacles and this model offers this feature [19];
- Willingness: the willingness OLSR parameter was set to WILL ALWAYS (7). The purpose of this is to keep nodes always cooperative. This parameter specifies how a node is willing to forward packets on behalf of other nodes [9];
- Flow: in order to obtain a packet delivery rate above 90% (in RFC 3626 OLSR), scenario with 10 nodes, five traffic flows of 512Kbps was used. For the scenario with 30 nodes, in order to obtain a packet delivery rate above 80% (OLSR in RFC 3626) fifteen traffic flows of 256Kbps was used;
- Simulations Number: the results were obtained from the arithmetic average of 10 simulations with random seed distribution and 95% confidence level. The simulations were made sequentially, one after other, so that a simulation result did not interfere with each other. Two scenarios were used (S) (10 and 30 nodes),

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 01, January-2016

five environments (E) with different selfish nodes rates (0%, 10%, 20%, 40% and 60%), four routing metrics (RM) (hop-count, ML, ETX and MD) and ten runs of simulation (R). Thus, the number of simulations (Q) is equal to $Q = S*E*RM*R \rightarrow 2*5*4*10 = 400$;

- Hardware and Software: processor Intel (R) CORE (TM) i5 CPU M 450 2.40GHz, 6GB of RAM, 500GB HD, operating system Ubuntu Linux 12.04 LTS (Precise Pangolin) 64Bit;
- Performance Metrics:

o Throughput – described by Hossain and Issaraiyakul [20] as the amount of bits sent from a source node to a destination node, divided by the observation time, as in (5);

$$Throughput = \frac{Bits\ Sent_{Source \rightarrow Destination}}{Observation\ Time_{(s)}} \quad (5)$$

o Packet Loss Rate – in this paper, is the amount of packets generated at the source node minus the amount of packets received at the destination node [21], as in (6);

$$PLR = Packets_{Sent} - Packets_{Received} \quad (6)$$

o Power Consumption – is the amount of energy used by a node for maintenance of routes, receiving and sending packets. The power consumption can be measured in joules (J) or milliwatts (mW).

- Description Selfish Node: the kind of selfish node used in this research is one that drops data packets and relays control packets [22] (Figure 4) .

Algorithm 1: Behavior Selfish - OLSR Function recv(Packet* p, Handler* h).

```
Input: Data Packet or Control Packet
Output: Forward Packet or Drop Packet or Process Packet
if (Data Packet) then
    if (Data packet was sourced in current node) then
        Forwarding(Packet);
        else
            Drop(Packet);
        end if
    end if
    else
        Pass the packet to OLSR for processing;
    end if
end if
```

Fig.4. Algorithm selfish behavior implemented in OLSR.

The scenarios considered, from which were obtained all the results have been configured with the parameters shown in TABLE III.

TABLE III. SIMULATION PARAMETERS SUMMARY.

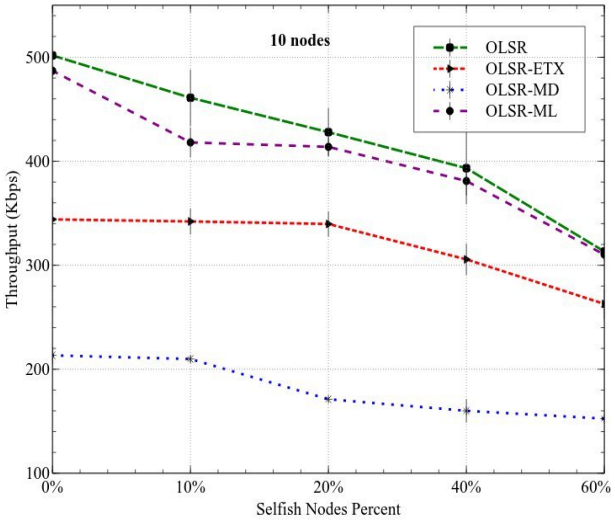
Parameters	Value
Routing Protocol and Extensions	*OLSR RFC* 3626, *OLSR-ETX, OLSR-ML* and *OLSR-MD*
Simulation Area	600*m* × 600*m*
Number of nodes	10 node (Scenario 1) and 30 nodes (Scenario 2)
Traffic Type	*CBR*
Packet Size	1000 *Bytes*
Node Power	100 *Joules*
Transmitter Power	TX = 1.2*W and* RX = 0.6*W,* range 250m [27]
Signal Propagation Model	*TwoRayGround*
MAC Type	*IEEE* 802.11b
Node Speed	Minimum 1.39*ms* – maximum 5.0*ms* no pause (human speed between walking and running)
Selfish Nodes Percentages for each Environment	10%, 20%, 40% and 60%
Simulation Time	200*s*
Average Generated Packets for Scenario	Scenarios*: 1* → 42110 and *2* → 41470

V. EXPERIMENTAL RESULTS

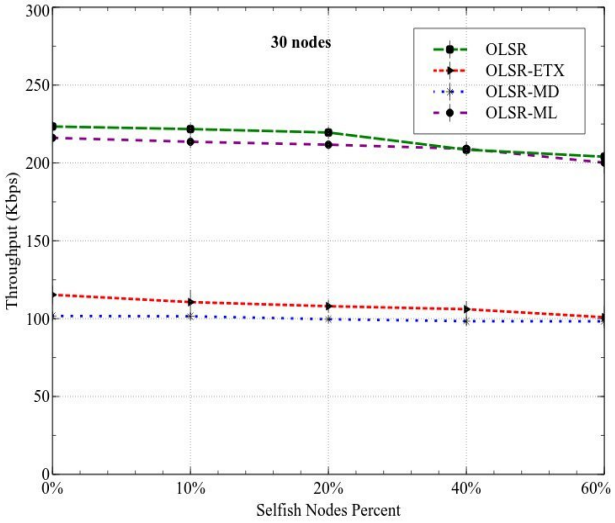
The throughput graphics show in general (Figure 5) in environments without selfish nodes, that OLSR has better performance. This can be explained by the fact that it selects relay nodes (MPRs) with better connectivity, other extensions use the algorithm MPR selection described in [18], according to Rohal et al., [23] if connectivity is good throughput increases. The OLSR-ML has the second best performance and the explanation for this lies in the fact that the metric ML chooses routes with the lowest loss probability, hence better throughput [7]. The OLSR-ETX extension has the third best throughput, this happens due to the fact that the metric ETX chooses paths that mask high packet loss rates [6]. Finally, with the worst throughput, OLSR-MD, the explanation for this phenomenon lies in the fact that the metric MD generates high control message overhead, twice as many other extensions, which causes congestion, collisions and results in packet loss [24].

With regard to scenario 1 (10 nodes) (Figure 5a), in terms of losses from the selfish action, comparing the throughput difference obtained between the best case (without selfish nodes) and worst case (60% of selfish nodes) OLSR-ETX was the one which had the least impaired throughput loss 81.3Kbps (23.6%), followed by OLSR-MD with 60.9Kbps (28.5%) loss, the OLSR-ML with 176.7Kbps (36.3%) loss, and at last the OLSR with 188.5Kbps (37.5%) loss. In scenario 2 (30 nodes) (Figure 5b), the OLSR-MD had the best performance with the least throughput loss 3.5Kbps (3.4%), second OLSR-ML with 15.9Kbps (7.4%) loss, followed by OLSR with 19.4Kbps (8.7%) loss and, finally, the OLSR-ETX with 14.4Kbps (12.5%) loss. It is observed, based on Figure 5, that the impact of selfish behavior is more harmful in scenario 1 (10 nodes), due to the fact that there is few routes, most of which are more likely to be made by selfish nodes. It is observed in Figure 5, that regardless the metric used by OLSR, the selfish nodes hinder the throughput, changes in this impact can be explained by the fact that the metrics select different routes that have different throughput.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 01, January-2016



(a)

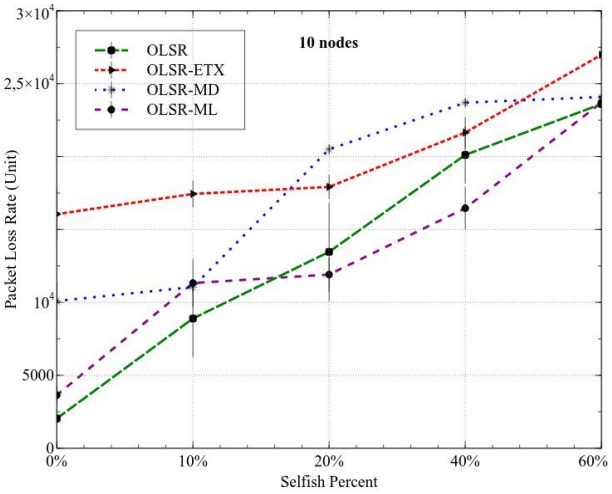


(b)

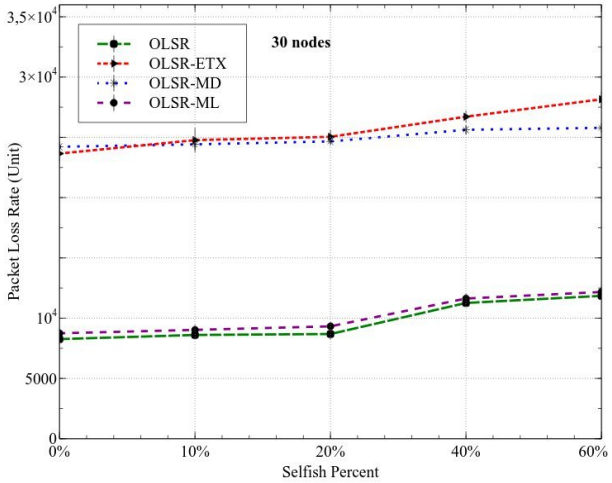
Fig.5. Global throughput.

Regarding packet loss, no selfish nodes environment, OLSR has the best performance due to the use of the hop-count metric, choosing paths with fewer hops, routes with this feature while not having good quality have a low probability packet loss [8]. The OLSR-ML has the second lowest packet loss, and the explanation for this lies in the fact the metric ML select routes with the lowest loss probability [7]. The high rate of OLSR-ETX packet loss is due to the excessive forwarding amount, according to Cordeiro [25], this can occur because of loops on the route (i.e., there were many packet droppings per loop event on this metric), this phenomenon was observed not only on simulations, but also experimental testbeds and production networks using olsrd. Regarding the OLSR-MD, high packet loss rate is related to the high overhead of control messages, this metric sends packet pairs to calculate the delay [8], the excessive amount of control packets increases network congestion and collisions that generate delays and packet losses.

In the case of packet loss due to selfish action, in scenario 1 (10 nodes) Figure 6a, taking into account the difference between the best and the worst case, the OLSR-ETX had its packet loss increased by 27.58% (best Performance), followed by OLSR-MD 32.90% loss, OLSR-ML 47.64% loss, and finally OLSR 51.24% loss. In scenario 2 (30 nodes) (Figure 6b), considering the difference between the best and the worst case, the OLSR-MD had the best performance with increased loss in just 3.54%, second OLSR-ML with 26.8% loss followed by OLSR with 8.66% loss and, finally, OLSR-ETX with 10.86% loss.



(a)



(b)

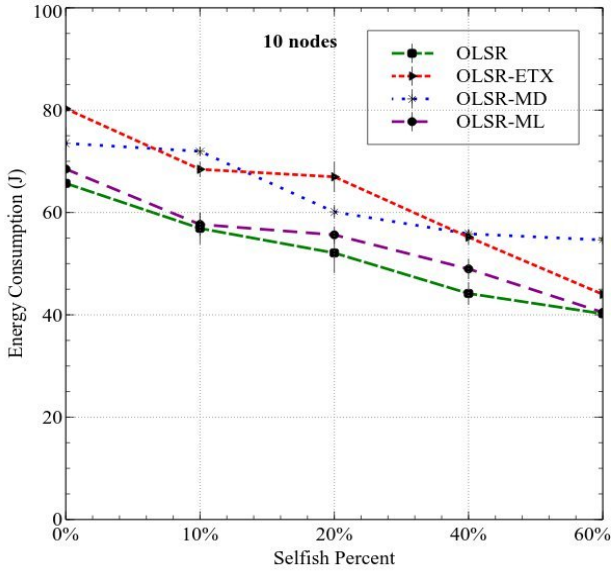
Fig.6. Global packet loss rate.

Thus, just as in the throughput (Figure 5), the impact of selfish action is more harmful in scenario 1 (10 nodes). In addition, it is observed even with variations, that the selfish actions have less impact in scenario 2 (30 nodes) and the explanation for this lies in the fact that scenario have more route options, due to the fact that mentioned scenario has more nodes. Accordingly, the probability of selecting a route which is not comprised by selfish nodes is higher.

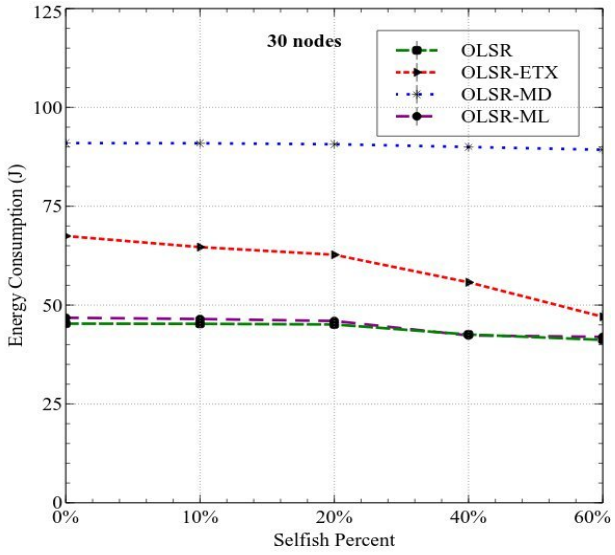
Regarding energy efficiency (Figure 7) the OLSR is observed to be more efficient (lower power consumption) and this is because it consumes less processing time for route choice, since its heuristic routing does not use additional fields in their control messages or probe message

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 01, January-2016

usage, as does the OLSR-ML, OLSR-ETX and OLSR-MD. Moreover, the algorithm used in the OLSR hop-count metric has a computational cost O(E+V) lower than the tested extensions which use Dijkstra's algorithm with $O(V^2)$ complexity.



(a)

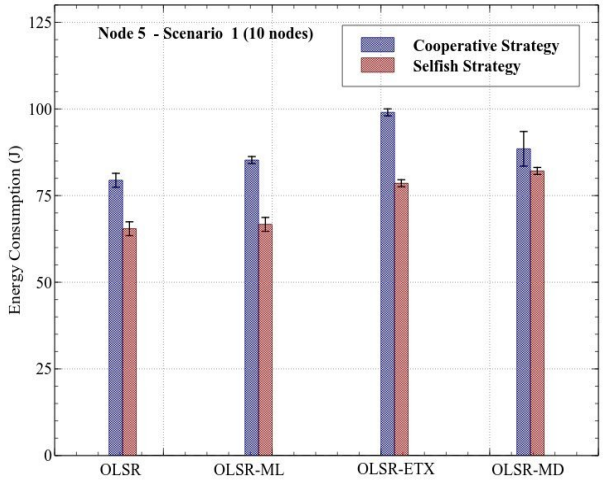


(b)

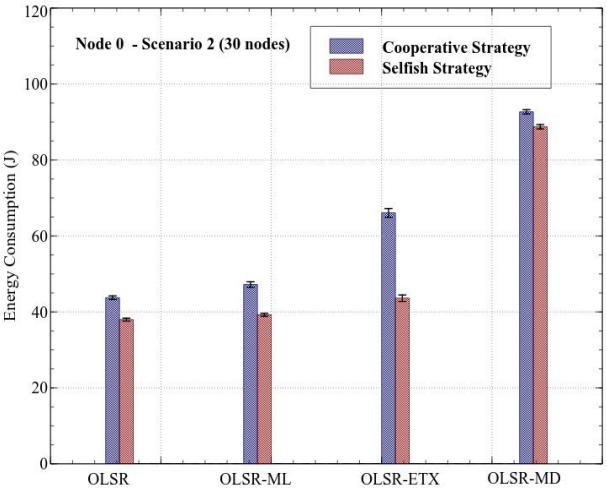
Fig.7. Global energy consumption.

It is also verified that the global power consumption decreases as the selfish nodes percentage increases in the system, and this is because the selfish nodes drop packets that should be forwarded consequently the other nodes that are part of the route fail to forward these packets to the destination, which implies in energy savings [11]. Moreover, Figure 7 shows that the average overall energy consumption in scenario 2 (30 nodes) is smaller than the scenario 1 (10 nodes), and this occurs because the number of hops increases, since the greater the number of hops in a route the less energy is expended to forward a packet to the destination because the cost is divided by nodes that make up the route [26]. The exception in scenario 2 (30 nodes) is the OLSR-MD, which has an average power consumption higher than in scenario 1 (10 nodes), and this is because there are more routes to be evaluated, which requires the OLSR-MD to send more probe messages in scenario 2 (30 nodes) than in scenario 1 (10 nodes), twice the other metrics [24].

To verify if the selfish strategy saves energy, two nodes were randomly chosen within the selfish node set, one in each scenario (10 and 30 nodes). Thus, the nodes power consumption were compared before and after becoming selfish (Figure 8). As presented in Figure 8, it is observed that in both cases the selfish strategy, in fact, saves energy.



(a)



(b)

Fig.8. Energy consumption with strategy change, cooperative to selfish.

VI. CONCLUSIONS AND FUTURE WORK

One of the main MANETs limitation is the power constraint, since a mobile node is completely dependent on the battery use. In this context, the selfish nodes that drop data packets are a problem on those networks as their primary objective is to maximize their battery lives by denying data packet forwarding for other nodes. So one of the biggest challenges in MANETs is to develop routing protocols that consume few energy resources and are also able to work around the problem of selfish nodes.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 01, January-2016

In this paper we presented the impact of selfish behavior, which drops data packets in the OLSR protocol and extensions (OLSR-ML, OLSR-ETX and OLSR-MD). It was found with experiments that selfish action has a negative impact on throughput and packet loss rate. It was also noted that this impact is more visible in sparse environments, low node density as scenario 1 (10 nodes). Thus, the methodology employed in carrying out simulations aimed to assess the impact of selfish behavior in performance metrics, throughput and packet loss. In addition, simulations were performed to see if, in fact, the selfish strategy takes advantage regarding to energy savings, the results showed in the proposed scenarios, that the selfish strategy obtains significant energy savings in both OLSR as well as their extensions.

According to this, for future MANETs routing protocol designs, routing metrics should take into account not only the paths with better throughput or minimum delay, but also the choice of reliable paths without selfish nodes and better energy capacity. In this sense, as future work, we intend to make an assessment of selfish behavior impact in various routing protocols, such as OLSR, DSDV, AODV, DSR and ZRP. The goal is also to verify in which protocols this selfish behavior is more harmful.

REFERENCES

[1] C. P. Agrawal, M. K. Tiwari, and O. P. Vyas, O. P, "Evaluation of AODV Protocol for Varying Mobility Models of MANET for Ubiquitous Computing," Convergence and Hybrid Information Technology, 2008. ICCIT'08. Third International Conference on IEEE, p. 769-774, 2008.

[2] I. Chlamtac, M. Conti, and J. J. -N. Liu, "Mobile ad hoc networking: imperatives and challenges," Ad hoc networks, vol. 1, n. 1, p. 13-64, 2003.

[3] G. Athanasiou, L. Tassiulas, and G. S. Yovanof, "Overcoming misbehavior in mobile ad hoc networks: An overview," Crossroads The ACM Student Magazine, vol. 11, n. 4, p. 5-5, 2005.

[4] L. Buttyá, J-Pierre Hubaux, "Security and Cooperation in Wireless Networks – Thwarting Malicious and Selfish Behavior in the Age of Ubiquitous Computing", Cambridge University Press, July 2007(draft).

[5] C. Toh, D. Kim, S. Oh, and H. Yoo, "The Controversy of Selfish Nodes in Ad Hoc Networks," Advanced Communication Technology (ICACT), 2010 The 12th International Conference on IEEE, p. 1087-1092, 2010.

[6] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A High-Throughput Path Metric for Multi-Hop Wireless Routing," Wireless Networks. vol. 11, n. 4, p. 419-434, 2005.

[7] D. Passos, D. V. Teixeira, D. C. Muchaluat-Saade, L. C. S. Magalhaes, and C. V .N Albuquerque, "Mesh Network Performance Measurements", Institute for Computing of the Fluminense Federal University (IC/UFF) and Telecommunications Engineering Department of the Fluminense Federal University (TET/UFF), 2006.

[8] W. Cordeiro, E. Aguiar, W. M. Junior, and A. A. M. Stanton, "Providing Quality of Service for Mesh Networks Using Link Delay Measurements," ICCCN 2007. Proceedings of 16th International Conference on IEEE, p. 991-996, 2007.

[9] T. H. Clausen, P. Jacquet, "RFC 3626 OLSR - Optimized Link State Routing Protocol," Available in: http://www.ietf.org/rfc/rfc3626.txt, Accessed on June 10, 2014.

[10] S. Yokoyama, Y. Nakane, O. Takahashi E. Miyamoto, "Evaluation of the Impact of Selfish Nodes in Ad Hoc Networks and Detection and Countermeasure Methods," MDM 2006. 7th International Conference on IEEE, p. 95-95, 2006.

[11] H. Kothari, M. Chatuverdi, "Effect of Selfish Behavior on Power Consumption in Mobile Ad Hoc Network" Proceedings of the Asia-Pacific Advanced Network. vol. 32, p. 91-100, 2011.

[12] A. Babakhouya, Y. Challal, and A. Bouabdallah, A, "A simulation analysis of routing misbehaviour in mobile ad hoc networks," NGMAST'08. The Second International Conference on IEEE, p. 592-597, 2008.

[13] S. Gupta, C. K. Nagpal, and C. Singla, "Impact of selfish node concentration in manets," International Journal of Wireless & Mobile Networks (IJWMN). vol. 3, p. 29-37, 2011.

[14] J.-M. Robert, H. Otrok, and A. Chriqi, "RBCOLSR: Reputation-based clustering OLSR protocol for wireless ad hoc networks," Computer Communications, vol. 35, n. 4, p. 487-499, 2012.

[15] J. Sengathir, R. Manoharan, "Exponential Reliability Coefficient based Reputation Mechanism for isolating selfish nodes in MANETs. Egyptian Informatics Journal, Elsevier, p. vol. 16, n. 2, :231–241, 2015.

[16] A. Tønnesen, T. Lopatic, H. Gredler, B. Petrovitsch, A. Kaplan, and S.-O. Tucke, "Optimized Link State Routing Protocol," Available in: http://www.olsr.org/?q=about, Accessed on February 10, 2015.

[17] F. Cuppens, N. Cuppens-Boulahia, T. Ramard, and J. Thomas, "Misbehaviors Detection to Ensure Availability in OLSR, "In: Mobile Ad-Hoc and Sensor Networks Lecture Notes in Computer Science, vol. 4864, p. 799–813, December 2007.

[18] Y. Ge, T. Kunz, and L. Lamont, "Quality of Service Routing in Ad-Hoc Networks Using OLSR," IEEE Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS03), p. 487–499, 2003.

[19] J.-Y L. Boudec, M, Vojnovc, "Perfect Simulation and Stationarity of a Class of Mobility Models," INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. p. 2743-2754, 2005.

[20] T. Issaraiyakul, E. Hossain, "Network Simulator 2 Ultimate: post processing throughput calculation," Available in: http://www.ns2ultimate.com/post/3442965938/post-processing-ns2-result-using-ns2-trace-ex1-link, Accessed on August 8, 2015.

[21] A. U. Salleh, Z. Ishak, N. M. Din, and M. Z. Jamaludin, "Trace Analyzer for NS-2," Research and Development, 4th Student Conference on IEEE, p. 29-32, 2006.

[22] S. Yokoyama, Y. Nakane, O. Takahashi, and E. Miyamoto, "Evaluation of the Impact of Selfish Nodes in Ad Hoc Networks and Detection and Countermeasure Methods," MDM 2006. 7th International Conference on IEEE, p. 95-95, 2006.

[23] P. Rohal, R. Dahiya, and P. Dahiya, "Study and Analysis of Throughput, Delay and Packet Delivery Ratio in MANET for Topology Based Routing Protocols (AODV, DSR and DSDV)," International Journal For Advance Research In Engineering And Technology (IJARET) India, vol.1, n. 2, 54–58, Mar 2013.

[24] D. A. M José, "OLSR Fuzzy Cost (OLSR-FC): an extension to OLSR protocol based on fuzzy logic and applied to prevent selfish nodes," Goiânia, Goiás Federal University (UFG) – Institute of Informatics (INF), Available in: http://repositorio.bc.ufg.br/handle/ri/10452, Accessed on August 05, 2015.

[25] W. L. C. Cordeiro, "OLSR modules for NS2 (OLSR-MD, OLSR-ETX, OLSR-ML, OLSR)," Available in: http://www.inf.ufrgs.br/~wlccordeiro/resources/olsr/, Accessed on June 10, 2015.

[26] P. S. Hiremath, M. S. Joshi, "Energy efficient routing protocol with adaptive fuzzy threshold energy for manets," International Journal of Computer Networks and Wireless Communications (IJCNWC). ISSN: 2250-3501, vol. 2, 2012.

[27] H. K. H. Y. Wardi, Y. S. Kobayashi, "Re-olsr: Residual energy-based olsr protocol in mobile ad hoc networks," International Journal of Multimedia Technology, vol. 1, n. 2, 2011.