Journal Homepage: -www.journalijar.com

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)

ISSN NO. 2320-5407

Article DOI:10.21474/IJAR01/12396
DOI URL: http://dx.doi.org/10.21474/IJAR01/12396

RESEARCH ARTICLE

ASSESSMENT OF THE PERFORMANCE OF A BUFFER MANAGEMENT SYSTEM TO IMPROVE HIGH PRIORITY MESSAGE DELIVERY TIME LIMIT

Gballou Yao Théophile[1,2], Touré Kidjegbo Augustin[1] and Tiecoura Yves[1,2]

1. Ecole Doctorale Polytechnique de l'INP-HB Yamoussoukro, Côte d'Ivoire.
2. Laboratoire de Recherche en Informatique et Télécommunication (LARIT).

Manuscript Info

Manuscript History
Received: 27 November 2020
Final Accepted: 30 December 2020
Published: January 2021

Key words:-
Vehicular Delay-Tolerant Networks, Message Weight, Priority Class of Service, High-Priority Message

Abstract

In vehicular delay-tolerant networks, buffer management systems are developed to improve overall performance. However, these buffer memory management systems cannot simultaneously reduce network overload, reduce high priority message delivery time limit, and improve all priority class message delivery rates. As a result, quality of service is not guaranteed. In this paper, we propose a drop policy based on the constitution of two queues according to message weight, the position of the node in relation to the destination and the comparison of the oldness between the high-priority message and the messages in the low-priority queue. The results of the simulations show that compared to the existing buffer management policy based on time-to-live and priority, our strategy simultaneously reduces network overload, reduces the delivery time limit of high-priority messages and allows for an increase in the delivery rate of messages regardless of their priority.

Introduction:-

The DTN (Delay Tolerant Network) proposed by the DTNRG (Delay Tolerant Network Research Group), appears as a solution to the problem of networks characterized by intermittent connectivity, long or variable time limits, asymmetric data rates and high error rates. In addition, these networks use the Store-Carry-and-Forward (SCF) paradigm due to the absence of a connection from one end to another between source and destination [1] [2]. VDTN (Vehicular delay-tolerant Networks) extend VANET (Vehicular Ad-Hoc Network) with DTN technologies to support long disconnections in a sparse environment where there is no connection from one to another between source and destination. Thus, VDTNs not only support the different applications (information and entertainment applications, traffic management applications, road safety applications....) supported by VANETs, but by using the SCF mechanism, they enable non-real-time communication, connectivity in remote areas without any conventional communication infrastructure [3] [4] [5]. In addition, VDTNs simultaneously support multiple asynchronous applications whose generated traffic have different requirements in terms of delivery time limit and delivered message ratios. Thus, in order to guarantee the quality of service, the traffic differentiation is elaborated at the bundle protocol level, which classifies the message traffic into three CoS (Class of Service) of priority: high priority for accident messages, medium priority for traffic management messages and low priority for information and entertainment messages [6] [7].

In addition, in order to increase the message delivery ratio on the one hand, and to reduce the message delivery delay on the other hand, VDTNs use multiple copy routing protocols. However, the combination of excessive

Corresponding Author:- Gballou Yao Théophile
Address:- Ecole Doctorale Polytechnique de l'INP-HB Yamoussoukro, Côte d'Ivoire.

replication of these routing protocols in a limited buffer space and storage for varying lengths of time results in rapid buffer saturation. Thus, the overall performance of the network in terms of delivery ratio, delivery time limit and network overload is fully affected [4] [8] [9]. Consequently, it results in the need for the development of an efficient buffer memory management system in order to determine the message(s) to be transmitted during opportunistic encounters, on the one hand, and the message(s) to be dropped in case of buffer congestion, on the other hand [10] [11] [12]. Many strategies for buffer memory management (scheduling, dropping) have been developed to help improve the overall performance of the network. Some strategies are based either on global knowledge of the network (information on the whole network for all messages), however they are practically impossible to implement (partitioned network, highly dynamic topology...), while other strategies, preferably used, are based on local information (remaining time-to-live, size,…) [10]. The latter strategies fall into two categories depending on whether or not they use the CoS of priority required for the different applications carried in the VDTN.

In [13], the authors proposed a buffer memory management policy based on message weight and the distance between the destination and the intermediate node with high delivery predictability [14]. The message weight is based on local message information. The local message information is: number of hops, number of replications, size, remaining time-to-live, and buffer time and priority class of service. The weight calculation allows messages to be split into two queues, one for high and one for low weight messages. The distance between the destination and the intermediate node with high predictability of delivery, allows determining the closest node to the destination in order to avoid the deletion of a message close to its destination. However, when the size of the new message is less than the size of the high weight queue and the node is close to the destination ($d < d_{min}$), then this message is ignored. Therefore, the message delivery rate will be affected. In order to determine the factors that enable an improvement of the overall network performance and quality of service, we formulate the following research question: What buffer management system should be developed to simultaneously enable a reduction of network overload, an increase of the message delivery rate regardless of the priority class and a reduction of the delivery delay of high priority messages?

In order to address this concern, we propose in this article a buffer memory management system based on the dynamic weight of the message and the particular conditions for inserting a new high-priority message into the buffer.

This paper is summarized in the following points:
A state of the art of the most relevant buffer management systems, reducing network overload and delivery time on the one hand, and on the other hand allowing an improvement of the delivery rate.

The proposal of a buffer abandonment strategy based on the weight of the message, the position of the current node in relation to the destination. The position of the current node in relation to the destination depends on the age of the high-priority message.

The rest of the article is organized as follows. Section 2 presents previous work. Section 3 describes in detail the algorithm of the proposed buffer management strategy. The simulation area, simulation parameters, and results are presented in section 4. The conclusion and perspectives are presented in section 5.

State of the art:-
In delay-tolerant networks, nodes have limited bandwidth and buffer space. To this end, scheduling and dropping policies are developed to determine which message to abandon in the event of buffer congestion and which message to transfer during opportunistic encounters.

In [15][16], the authors proposed a routing algorithm and a buffer management system based on message weight. In [15][16], the message weight is a function of the estimated delivery time of the message and the estimated number of hops at the destination. In the routing algorithm, the current node transfers the message either to the message destination, if the destination is in its neighborhood. Otherwise, it transfers the message to the neighbor with the lowest weight. The drop policy deletes the message with the lowest number of copies first, then the messages with the highest weight if the space left by the message with the lowest number of copies is insufficient. This strategy shows that limiting the number of message copies, estimates of the number of message hops and delivery time limit to the destination reduce network overhead and delivery time and improve the delivery rate. However, this strategy does not take into account the priority class of service. In [16], the authors proposed two expressions of weight, one of which is used for message scheduling and the other for drop policy. For planning, weight depends on the number of copies, age and size of the message. The message with the highest weight is transmitted first. For the drop policy, the weight depends on the number of copies, the time-to-live and the size of the message. In case of buffer congestion, the message with a higher weight than the weight of the incoming message is deleted first. This strategy reduces network overhead, increases the delivery rate, but does not reduce the delivery delay. In [17], the authors proposed a management system that divides messages into two queues according to message weight. In case of congestion, the messages in the high weight queue are deleted to insert the new message. If the buffer cannot accommodate the new message after the deletion of the messages in the high weight queue, then all messages in the high weight queue are deleted and then those in the low weight queue are deleted according to the increasing time-to-live if the node is the destination of the message. Otherwise the new message is ignored. This strategy provides protection for recent messages, reduces network overhead, and increases the delivery rate. However, it does not offer any results on message delivery time limit.

The authors in [18][19] proposed a buffer memory management strategy to improve the performance of the PRoPHET routing protocol [14]. Buffer memory management is based on the message utility function. This utility function takes into account parameters such as the time-to-live, the duration of the message in the buffer, the number of hops and the number of replications of the message. In case of buffer congestion, the message with the highest utility value is discarded. These strategies improve the message delivery rate and reduce network overload. However, these strategies have a message delivery time limit almost close to the traditional time-to-live based management strategy, SHLI (Evict Shortest Life Time First) [20].

In [21] [22], the authors proposed buffer management strategies that take into account traffic differentiation in order to highlight the different applications supported by delay-tolerant networks. In [21], the authors proposed two buffer management algorithms named Priority and Factor. Priority divides the messages in the buffer into a high-priority queue whose messages are transmitted using the First-Contact routing protocol [23], a medium-priority queue whose messages are transmitted using the Epidemic routing protocol [24], and a low-priority queue whose messages are transmitted using the Spray-and-Wait routing protocol [25]. Factor first transfers the high priority and highest TTL message. In case of buffer congestion, Factor drops the low TTL message first. However, if two messages have the same TTL value, then Factor abandons the lower priority message. Both of these strategies increase the rate of delivered messages and reduce network overload as the number of nodes increases. However, not only are the graphical representations (delivery rate, delivery time) independent of priority classes, but Factor also favors high priority messages over lower priority messages. In [22], the authors proposed a hybrid routing mechanism allowing the transfer of high, medium and low priority messages with the routing protocols Epidemic, Spray-and-Wait and First-Contact respectively. In case of buffer congestion, messages with a lifetime below a threshold are deleted. This strategy reduces network overload and increases the message delivery rate regardless of priority. However, not only is there an increase in delivery time limit compared to traditional strategies, but the proposed strategy uses an empirical threshold of the number of copies.

The above buffer management systems have advantages and limitations. Thus, taking into account the advantages of systems based on priority service class and the advantages of systems based on message weight and utility functions, we propose a new buffer management model.

The proposed buffer management strategy:-

As mentioned in the previous section, the buffer management strategy proposed in [13] is composed of, on the one hand, the message weight based on local network information, and on the other hand, the combination of the current node's position with respect to the destination and the performance of the PRoPHET routing protocol [14].

In this part, we present the system model, the constitution of queues, the conditions for inserting a high-priority message, the details of the drop policy and the scheduling policy.

System model:-

In VDTNs, the nodes are characterized by high mobility and relatively high speeds that result in a very dynamic network topology. This highly dynamic topology results in short contact times between nodes and intermittent connectivity. Consequently, message transfers are impacted by the mobility model [26] [27]. Thus, for message transfer, nodes must be within communication range. Furthermore, we assume that all nodes in the network have the same limited buffer memory capacity.

Queue building:-

The weight of a message

In [13], the weight of a message in the buffer is determined from the following local information: number of jumps, number of copies, size, lifetime, time in buffer and priority class of the message.

This weight is given by the following expression:

$$P_d[i] = HC_i + RC_i + \frac{1}{S_i} + \frac{1}{TTL_i} + \frac{1}{Tbuf_i} + \frac{1}{P_i} \quad (1)$$

Where HC_i is the average number of hops of message i, RC_i is the number of replications of message i, Si is the size of message i, TTL_i is the remaining lifetime of message i, $Tbuf_i$ is the time put in the buffer by message i and P_i is the priority class of message i.

The Low Weight Queue and the High Weight Queue

The low weight queue consists of messages whose weight is less than the average weight of all messages in the buffer. And the high weight queue consists of messages whose weight is greater than the average weight. The average weight of the messages in the buffer is given by :

$$P_M = \frac{1}{N(t)} \sum_{i=1}^{N(t)} P_d[i] \quad (2)$$

Where N(t) is the total number of messages in the node at date t and P_d[i] is the weight of a message in the buffer.

Special conditions for inserting ahigh-priority message in a low weight queue of an intermediate node:-

Comparison of number of hops and number of copies

In [13], the low weight queue consists of messages that are recent because these messages have low hop counts and low copy counts. In our model, when the current node is an intermediate node and the size of the incoming message is greater than the sum of the high-weight queue size and free space, then we decide to delete all messages from the high-weight queue and then delete the messages from the low-weight queue according to their increasing TTL under a particular condition.

The incoming message must be a high priority message and more recent than the message(s) to be deleted in the low weight queue. Thus, a message P is more recent than the message L, if the sum of the number of hops and the number of replications of P is less than the sum of the number of hops and the number of replications of L

$$HC_P + RC_P < HC_L + RC_L \quad (3)$$

Where HC_P and RC_P are the hop numbers and replicate numbers of the incoming high priority message (P). And HC_L and RC_L are the hop and replicate numbers of a message in the low priority queue (L).

The following activity diagram shows the mechanism for inserting a high-priority message into a low-priority queue when the message size is greater than the sum of the high-priority queue size and free space.

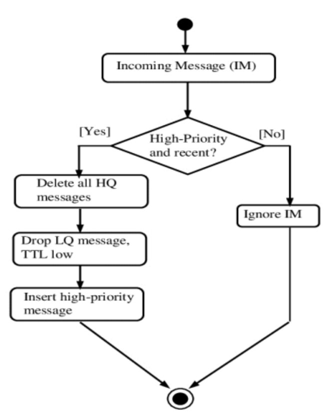


Figure 1:- High priority message insertion activity diagram.

Determination of the high predictability and near-destination node

In [13], the authors determined the node of high delivery predictability and close to destination. The characteristics of this node combine the performance of the PRoPHET routing protocol and the characteristics of the node movement (speed, position, direction). This node is selected from a set of nodes that are candidates as next relay.

This node is defined by the following expressions:

$$P(N;D) > P(M;D) \qquad (4)$$

$$d(N;D) \le d_0 \qquad (5)$$

Where P(N;D) and P(M;D) represent the delivery predictability of node N and node M, respectively, d(N;D) is the distance from node N to destination D, and d_o is the minimum distance to the destination given by :

$$d_0 = \min\{d(N;D)\} = ND\sin\theta \qquad (6)$$

With θ the angle formed by [ND) and [NH) is given by the figure below

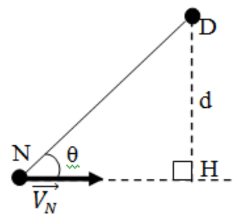


Figure 2:- Node movement in relation to the destination.

In our model, we determine this node with high delivery predictability and close to destination to avoid the deletion of high-priority messages that have arrived at their destination. We can then improve the delivery time of these high-priority messages.

The operations of the proposed strategy:-

In [17], when a message arrives, messages in the high weight queue are deleted. The condition for deleting messages from the low weight queue is based on the size of the incoming message and whether the node is the destination of the new message. Taking into account this condition, and the conditions on the high-priority message age and on the node's position relative to the destination, we propose a new buffer management strategy.

When a new message arrives, then the messages in the high weight queue are deleted according to their decreasing weight, then the message is inserted. However, if the size of the incoming message is greater than the sum of the high-weight queue size and free space, then all messages in the high-weight queue are deleted, then those in the low-weight queue are deleted according to their increasing TTL if the node is the destination of the message.However, if the current node is an intermediate node, then the messages in the low weight queue are deleted if these conditions are met. The current node is far from the destination, the message to be inserted is a high priority message and it is more recent than the message(s) to be deleted.

The following figure shows the activity diagram of the proposed drop strategy

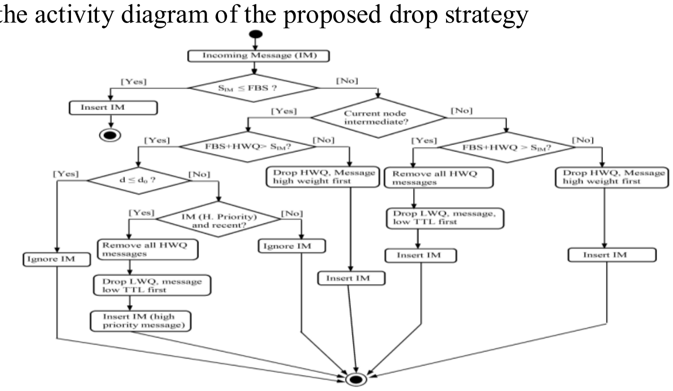


Figure 3:- Drop stratégy activity diagram.

The proposed planning policy:-

The purpose of the buffer scheduling policy is to determine the order in which messages should be sent to other nodes in the network because of the short contact time between nodes and the limited bandwidth [6] [28]. In the proposed buffer management strategy, we adopt a planning policy that takes into account the policy developed in [13] and the priority class of service. Thus, in opportunistic contacts, messages from the low weight queue are transmitted first, followed by those from the high weight queue. However, if two messages have the same weight value, then the message with the highest priority is transmitted first.

Performance evaluation:-

In this work, in order to evaluate the performance of the proposed strategy, we use the PRoPHET routing protocol [14] and the Factor buffer management strategy [21]. Factor transfers first the high-priority and highest TTL message. In case of buffer congestion, Factor drops the low TTL message first. However, if two messages have the same TTL value, then Factor drops the low priority message. The simulation tool used to perform buffer management system evaluations is the ONE (Opportunistic Network Environment) simulator [29]. It is a discrete event simulator that is written with the JAVA language. It is designed to simulate delay-tolerant networks. This simulator can be run under Windows or Linux.

The following subsections present performance measures, simulation configurations, results and performance analyses in scenarios of 25 vehicles and 150 vehicles.

Performance measures:-

We used three performance measures to evaluate our management policy: the delivery rate by priority class, the average delivery time by priority class and the rate of network overload.

The delivery ratio by priority class is determined by the ratio of the number of messages delivered in a priority class to the total number of messages in that priority class.

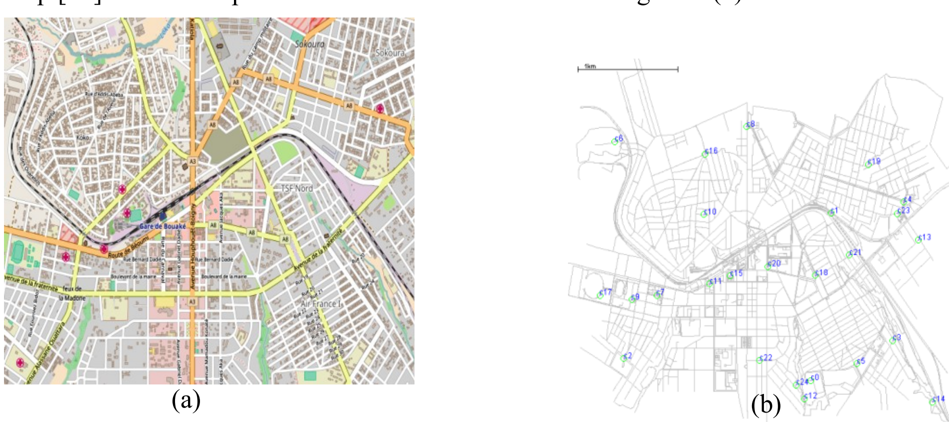
The average delivery time limit per priority class represents the average time required by the message in relation to the priority class from the time of its creation to the time of its delivery to the destination.

The network overload ratio represents the number of replications relayed for the delivery of a message. It is the ratio of the difference between the number of messages relayed and the number of messages delivered by the number of messages delivered.

Simulation settings:-

The Simulation Area

The simulation area is a 4000m×4000m section of the city center of Bouaké in Ivory Coast. The actual data of the map of this city was imported from Open Street Map [30] (see Figure 4 (a)). Then these data obtained in .osm format were converted to .wkt [31] format for use in ONE. In addition, vehicle routes were separated from railways using OpenJump [32]. The initial positions of the nodes are shown in Figure 4 (b).



(a) (b)

Figure 4:- (a) Map of Bouaké city center in Ivory Coast-(b) Initial positions of 25 nodes.

Simulation Parameters
We simulate scenarios with a number of vehicles ranging from 25 vehicles to 150 vehicles. The vehicles move with a Shortest-Path Map-Based Movement model. Once they reach their destination, the vehicles pause for 600 to 900 seconds before choosing a new destination with speeds between 30 and 50 km/h. In addition, the vehicles each have a constant buffer capacity that is set at 25 MB and 50 MB for each scenario. The vehicles use a wireless communication link (WiFi) with a data transmission rate of 6 Mbps and a transmission range of 30m. In addition, the traffic volumes generated by the high-priority messages are large. Thus, a high-priority message has a priority value of P=3 and a size in the range [750KB, 1.5MB]. A medium priority message has a priority value of P=2 and a size in the range [250KB, 750KB]. A low priority message has a priority value of P=1 and a size within the range [100MB, 250MB] [33]. In addition, all messages of each priority class are generated by three event generators, with a creation time in the range [15.30] seconds and a TTL of 120 minutes. All the parameters of the settings are given in table 1 below.

Table 1:- Simulation parameters and their values.

Parameters	Values
Simulation time:	43200 s (12 h)
Simulation are :a	4000m × 4000m
Number of nodes :	25-50
Buffer size :	25 MB - 50 MB
Transmission rate :	6 Mbps
Transmission range :	30 m
Random speed :	30-50 km/h
Message TTL:	120 min
Waiting time:	600s -900 s
Interval of creation:	15s -30s
Message size:	100KB-1.5MB
Mobility model:	Shortest-path map-based movement

The Results:-

1. Variation of Buffer Size from 25M to 50M
2. Performance evaluation with 25 vehicles

Figure 5 shows the delivery ratio for high priority, medium priority and low priority messages when the buffer size ranges from 25M to 50M for the proposed policy and Factor.

In general, whatever the buffer size (25M or 50M), the proposed policy is more efficient than the Factor policy since it provides significantly higher message delivery ratios per priority class (rates are between 58% and 64%) than the Factor policy (rates are between 6% and 61%). However, if we consider the competitiveness according to the buffer size, we notice that with the 50 M buffer size, the Factor policy is more efficient than the 25 M buffer size. Concerning the proposed policy, the difference in competitiveness is less perceptible with high rates of low amplitude between 58% and 64% for the respective buffer sizes 25M and 50M. This is explained by the fact that in the proposed policy, messages of all priority classes have the same chance of being delivered. Unlike Factor, this favors the high priority message to the detriment of messages from other priority classes.

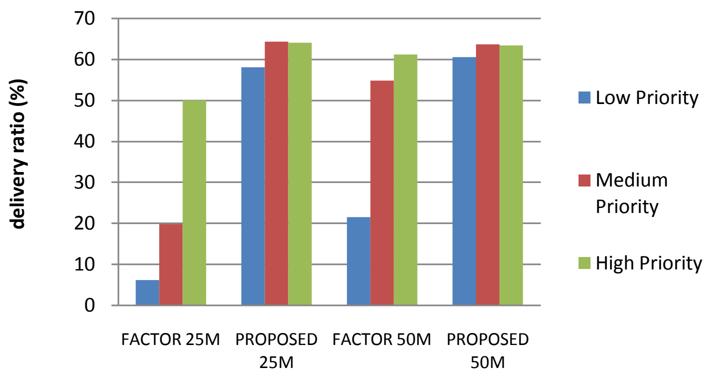


Figure 5:- Message delivery ratio by priority class according to buffer size for 25 vehicles.

Figure 6 shows the impact of buffer size on the network overload ratio for the proposed policy and the Factor policy.

This figure shows that whatever the nature of the buffer management policies, buffers with a buffer size of 50 M allow for a reduction in network overload unlike buffers with a buffer size of 25 M. In addition, the proposed buffer management policy better reduces network overload compared to the Factor policy. In fact, the network overload ratio of the proposed policy ranges from 12.3 to 10 for the 25M and 50M capacity buffers respectively. Conversely, in the case of Factor, the overload rate varies from 36.31 for 25M to 23.9 for 50M. All this confirms the results of the previous Figure 6, which shows that with the 50M capacity buffers compared to the 25M capacity buffers, the proposed buffer management policy is better than Factor's for the 25 vehicle sample. In addition, this reduction in network overload in the policy is explained by the fact that in the proposed policy, replications are reduced by the use of the message weighting function.

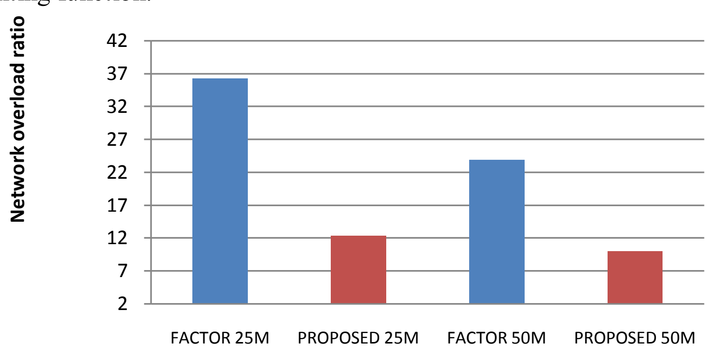


Figure 6:- Network overload ratio according to buffer size for 25 vehicles.

Performance evaluation with 50 vehicles

Contrary to Figure 5, Figure 7 is based on a sample of 50 vehicles even though both figures show the same performance indicators. Thus, did doubling the sample of vehicles change the trends in message delivery ratios for each of the priority classes of the two buffer management policies? The answer is no at the general level. That is, there is also confirmation for this figure, confirming the key results of Figure 6 presented during the interpretation of Figure 7. However, in this particular case, the Factor policy despite the buffer size (25M or 50M) in Figure 8 differs mainly from that of Figure 6 when comparing the variations in the average priority ratio: the magnitude of Figure 6 (20%, 55%) is higher than that of Figure 8 (14%, 40%) for each of the buffer sizes (25M or 50M).

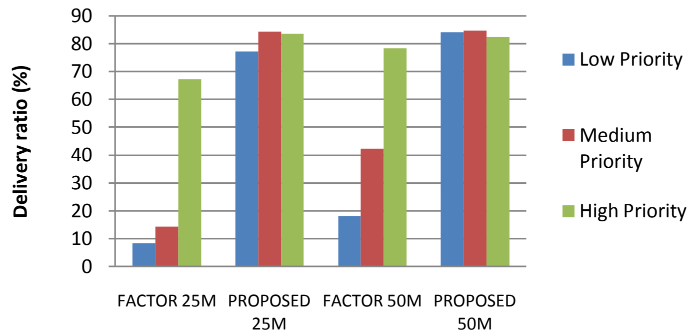


Figure 7:- Message delivery ratio by priority class according to buffer size for 50 vehicles.

Figure 8 is also characterized for a sample of 50 vehicles and follows the same statistical objectives as those of the previous Figure 6. Thus, since we obtain a quasi-similarity of the 2 curves, we can deduce that compared to the Factor policy, the proposed buffer management policy reduces network overload more. In fact, the proposed policy based on the weight of the messages reduces replications and consequently the consumption of resources than the Factor policy.

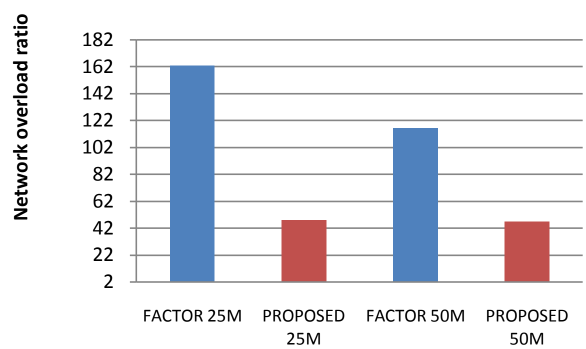


Figure 8:- Network overload ratio according to buffer size for 50 vehicles.

Variation in the number of Vehicles

In this section, we vary the number of vehicles from 25 vehicles to 150 vehicles. All vehicles have the same buffer capacity set at 50M. We evaluate the performance of the proposed buffer management policy and Factor's policy and then compare the performance of the two buffer management policies. Figures 9, 10, and 11 shows the impact of the number of vehicles on the delivery ratio for the proposed policy and Factor for low, medium, and high priority messages respectively.

Figures 9 and 10 show that for an increase in sample size from 25 vehicles to 150 vehicles, we observe an increase in delivery ratios for low and medium priority messages with the proposed buffer management policy ($\tau_{pf9} = +54.9\%$ and $\tau_{pf10} = +51.05\%$) while we observe a decrease in these rates with the Factor policy ($\tau_{Ff9} = -68.8\%$ and $\tau_{Ff10} = -79.8\%$). However, we obtain identical observations when the number of vehicles varies between 25 and 100 ($\tau_{pf9} = +58.4\%$ and $\tau_{pf10} = +48.5\%$) for the proposed policy and ($\tau_{Ff9} = -54\%$ and $\tau_{Ff10} = -70.35\%$) for Factor.

Thus, performance in terms of delivery of low and medium priority messages is held by the proposed buffer management policy relative to that of Factor. Contrary to Figures 9 and 10, Figure 11 shows an increase in the delivery ratios of high priority messages with both the proposed buffer management policy ($\tau_{pf11} = +51{,}54\%$) and Factor ($\tau_{Ff11} = +39.91\%$) for a sample of vehicles ranging from 25 to 150. However, identical observations are obtained when the number of vehicles ranges from 25 to 100 ($\tau_{pf11} = +47.24\%$) for the proposed policy and ($\tau_{Ff11} = +39.94\%$) for Factor. Thus, the Factor policy is only effective for the delivery of high priority messages.

Ultimately, it can be concluded that the proposed buffer management policy is efficient in terms of message delivery ratios whatever their priority class, in a low node density and high node density environment.

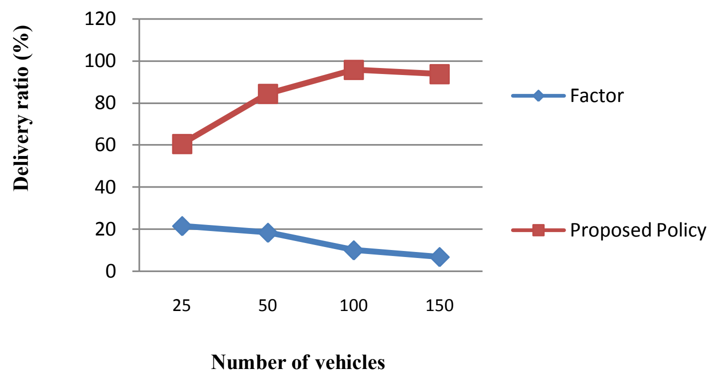


Figure 9:- Delivery ratio of low priority messages according to the number of vehicles.

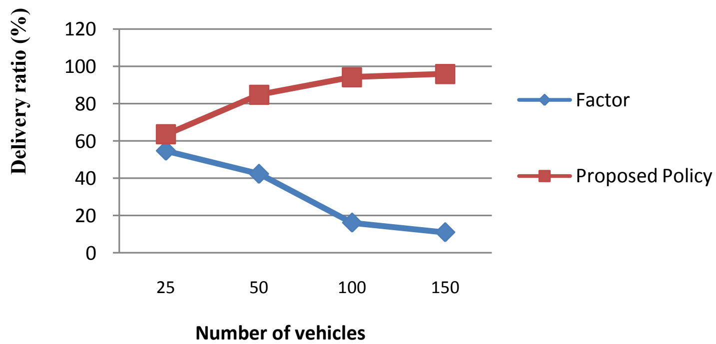


Figure 10:- Delivery ratio of medium priority messages according to the number of vehicles.

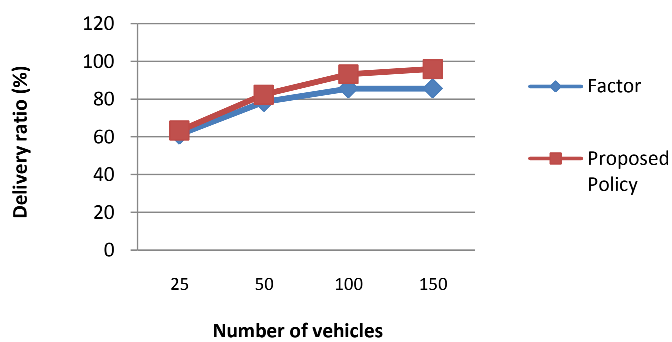


Figure 11:- Delivery ratio of high priority messages according to the number of vehicles

Figure 12 shows the impact of the number of vehicles on the delivery delay of high priority messages for the proposed policy and Factor.

This figure shows a low rate of variation in the average delivery delay of high priority messages both with the proposed buffer management policy ($\tau_{pf12} = -2.5\%$) and with Factor ($\tau_{Ff12} = -2.44\%$) for a sample of vehicles ranging from 25 to 50. In contrast, for a sample of vehicles ranging from 50 to 150, the rate of variation in the average delivery delay for high priority messages in the proposed policy ($\tau_{pf12} = -39.26\%$) is greater than that of Factor ($\tau_{Ff12} = -29.39\%$). Therefore, the proposed policy favors' a low delivery time relative to Factor. This is because in the proposed policy, when the current node is an intermediate node that is close to the destination, then high priority messages are not dropped. As a result, these messages arrive faster at the destination.

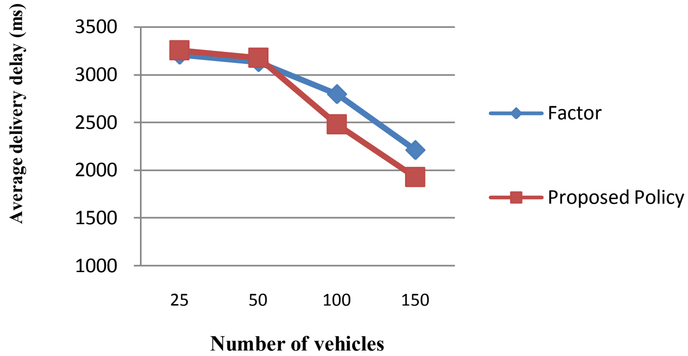


Figure 12:- Average delivery delay for high priority messages according to the number of vehicles.

Figure 13 shows the impact of the number of vehicles on network overload for the proposed policy and Factor where the number of vehicles ranges from 25 to 150.

Figure 13 indicates that unlike the Factor policy (network overload rate 23.9 to 977.5), the proposed policy (network overload ratio 10 to 274.9) better reduces network overload for a sample of vehicles ranging from 25 to 150. This is due to the weight of the message used in the proposed policy allowing for a reduction in the number of replications

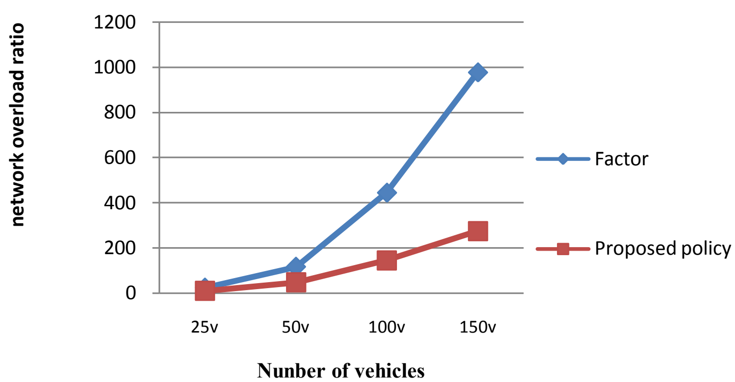


Figure 13:- Network overload ratio according to number of vehicles.

Conclusion and perspectives:-

In this paper, we evaluated the performance of our buffer management policy, which takes into account the distribution of messages according to their weight in the queues, the mechanism for determining the node with high delivery predictability and close to destination, and the most recent high-priority message. Thus, by using message weight in the drop policy, we can observe a reduction in network overload. Moreover, the determination of the node of high predictability and close to the destination avoids the deletion of high priority messages that have almost reached the destination. Evaluating the performance of our policy with a Factor buffer management system shows a significant reduction in network overload, a reduction in the time limit of high priority messages and an increase in the delivery ratio of messages of all priority classes.

When we take a sample of 25 vehicles, the delivery time is virtually equal to Factor. Therefore, in our future work, by integrating other mechanisms into the proposed model, we could further reduce the delivery time of high priority messages when the number of nodes is low, and thus improve road safety.

References:-

1. Kevin Fall. A Delay-Tolerant Network Architecture for Challenged Internets [C]. In Proceedings of the ACM SIGCOMM 2003. Karlsruhe, Germany, August 25-29, 2003:27-34
2. CAO, Yue and SUN, Zhili. Routing in delay/disruption tolerant networks: A taxonomy, survey and challenges. *IEEE Communicationssurveys & tutorials*, 2013, vol. 15, no 2, p. 654-677.
3. Isento, J. N., Rodrigues, J. J., Dias, J. A., Paula, M. C., & Vinel, A. Vehicular delay-tolerant networks - A novel solution for vehicular communications. *IEEE Intelligent Transportation Systems Magazine*, 2013, vol. 5, no 4, p. 10-19.
4. N. Benamar et al., Routing protocols in Vehicular Delay Tolerant Networks: A comprehensive survey, Comput. Commun. (2014), http://dx.doi.org/10.1016/j.comcom.2014.03.024
5. Soares, V. N.G.J., Farid Farahmand, and Joel Jose PC Rodrigues. "Impact of vehicle movement models on VDTN routing strategies for rural connectivity." *International Journal of Mobile Network Design and Innovation* 3.2 (2009c): 103.
6. Soares, V. N., Farahmand, F., & Rodrigues, J. J. (2009, September). Scheduling and drop policies for traffic differentiation on vehicular delay-tolerant networks. In *SoftCOM 2009-17th International Conference on Software, Telecommunications & Computer Networks* (pp. 353-357). IEEE.
7. Soares Vasco NGJ, Farid Farahmand, and Joel JPC Rodrigues. "Traffic differentiation support in vehicular delay-tolerant networks." *Telecommunication Systems* 48.1-2 (2011) : 151-162.
8. Spaho Evjola, et al. "Evaluation of Single-Copy and Multiple-Copy Routing Protocols in a Realistic VDTN Scenario." *2016 10th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS)*. IEEE, 2016.
9. Kang, H., Ahmed, S. H., Kim, D., & Chung, Y. S. (2015). Routing protocols for vehicular delay tolerant networks: a survey. *International Journal of Distributed Sensor Networks*, *11*(3), 325027.
10. Jain Sweta et Meenu Chawla. "Survey of buffer management policies for delay tolerant networks." *The Journal of Engineering*2014.3 (2014): 117-123.
11. Ahmed, Kawakib K., Mohd Hasbullah Omar, and Suhaidi Hassan. "Routing strategies and buffer management in delay tolerant networks." *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 8.10 (2016): 139-143.
12. Rahmatullah, Mohammad, and Priyanka Tripathi. "A new approach of enhanced buffer management policy in delay tolerant network (DTN)." *network* 11 (2014): 12.
13. Théophile, GBALLOU Yao, GOORE Bi Tra, and Brou Konan Marcelin. "Dynamic Weight Dropping Policy for Improve High-Priority Message Delivery Delay in Vehicular Delay-Tolerant Network." **in** International Journal of Advanced Computer Science and Applications 9(10) · January 2018 DOI: 10.14569/IJACSA.2018.091010
14. LINDGREN, Anders, DORIA, Avri, and SCHELEN, Olov. Probabilistic routing in intermittently connected networks. In : *Service assurance with partial and intermittent resources*. Springer, Berlin, Heidelberg, 2004. p. 239-254.
15. Wang, H., Feng, G., Wang, H., Lv, H., & Zhou, R. (2018). RABP: Delay/disruption tolerant network routing and buffer management algorithm based on weight. *International Journal of Distributed Sensor Networks*, *14*(3), 1550147718757874.
16. Sreekanth, G. R., Suganthe, R. C., & Latha, R. S. Performance Improvement of DTN Routing Protocols with Enhanced Buffer Management Policy. International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-4, November 2019.
17. Rashid, S., Ayub, Q., & Abdullah, A. H. (2014). Reactive Weight Based Buffer Management Policy for DTN Routing Protocols. *Wireless Personal Communications, 80*(3), 993-1010.
18. Alhutaba, A., Sati, S., & Eshtawie, M. Dynamic Weight Dropping Policy For Improve PROPHET Routing Performance, 1st Conference of Industrial Technology (CIT2017).
19. Sati, S., Probst, C., & Graffi, K. (2016, December). Implementing Forward and Drop Policies for Improving PRoPHET's Routing Performance. In *2016 12th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN)* (pp. 236-242). IEEE.
20. Rashid S., Ayub Q., Soperi Mohd Zahid M., Abdullah A.H.: 'E-Drop: an effective drop buffer management policy for DTN routing protocols', Int. J. Comput. Appl., 2011, 13, (7), pp. 8–13
21. More, Ruchira & Penurkar, Milind. (2015). Scheduling and Dropping Policies in Delay Tolerant Network. IARJSET. 2. 68-70. 10.17148/IARJSET.2015.21014.

22. Penurkar, M. R., & Deshpande, U. A. (2016, December). Priority-based scheduling policy for a hybrid routing algorithm in a Vehicular Delay Toleant Network. In *2016 International Conference on Computing, Analytics and Security Trends (CAST)* (pp. 578-583). IEEE
23. S. Jain, K. Fall, and R. Patra, “Routing in a Delay Tolerant Network,” in Proceedings of ACM SIGCOMM 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Portland, Oregon, USA, August 30-September 3, 2004, pp. 145-158, 2004.
24. Vahdat, D. Becker “Epidemic Routing for Partially Connected Ad Hoc Networks,” Technical Report CS-200006, Duke University, April 2000.
25. T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks," in ACM SIGCOMM 2005 - Workshop on Delay Tolerant Networking and Related Networks (WDTN-05), Philadelphia, PA, USA, August 22-26, 2005.
26. Soares, Vasco NGJ, Joel JPC Rodrigues, and Farid Farahmand. "Impact analysis of the shortest path movement model on routing strategies for VDTNs in a Rural region." *7th Conference on Telecommunications. CONFTELE 2009.*. 2009.
27. Soares, V. N.G.J., Farid Farahmand, and Joel Jose PC Rodrigues. "Impact of vehicle movement models on VDTN routing strategies for rural connectivity." *International Journal of Mobile Network Design and Innovation* 3.2 (2009): 103.
28. De Jesus, V. N. D. G., Rodrigues, J. J. P. C., Ferreira, P. S., & Nogueira, A. M. D. (2009, September). Improvement of messages delivery time on vehicular delay-tolerant networks. In *2009 International Conference on Parallel Processing Workshops* (pp. 344-349). IEEE.
29. Keränen A, Ott J, Kärkkäinen T. (2009) The ONE simulator for DTN protocol evaluation[C]//proceedings of the 2nd international conference on simulation tools and techniques. ICST (Institute for Computer Sciences, social-informatics and telecommunications engineering), 55-64
30. "Open street map", http://www.openstreetmap.org
31. mayer2010osm, Christoph P. Mayer,osm2wkt - OpenStreetMap to WKT Conversion, http://www.chrismc.de/osm2wkt,2010
32. "OPEN JUMP" Simulator, http://openjump.org/
33. Sadreddini, Z., & Afshord, M. M. (2013). Impact of using several criteria for buffer management in vehicular delay tolerant networks. *World Applied Sciences Journal*, *22*(9), 1204-1209.