International Journal of Emerging Science and Engineering (IJESE)
ISSN: 2319–6378, Volume-4 Issue-1, November 2015

Implementation of Rate Adaptation Algorithms for Vehicular Simulations in OMNeT++

Kenneth Sorle Nwizege, Harry, Inye H, Irimiagha Paul Gibson

Abstract - Rate adaptation (RA) is the process of dynamically switching data rates to match the channel conditions, with the goal of selecting the rate that will give a good throughput for the given channel condition. The two fundamental issues when designing a rate adaptation scheme are "when to increase and decrease the transmission rate". Rate Adaptation is a technique that enhances network performance. It is a process of alerting network nodes to change rate in respect to channel condition. Constant Bit Rate (CBR), Adaptive Auto Rate Fallback (AARF) , and Auto Rate Fallback (ARF) are the only Rate Adaptation Algorithms (RAAs) implemented in OMNeT++. In this paper, we implement, Onoe as new RAAs in OMNeT++ simulator. This is credit-based RAA that changes rate based on credit threshold accumulated.

Keywords: rate adaptation; wireless networks; OMNeT++;vehicular communication; IEEE802.11p; propagation phenomena; inetmanet

I. INTRODUCTION

Vehicular communication networks are a type of mobile network. In this network type, the users connect to existing networks directly or via an Access Point (AP). The IEEE 802.11p standard consists of a multi-channel operation, with seven 10 MHz channels. The allocation of this spectrum has been implemented both in European Union (EU), and the United States (US) and currently being implemented in other countries as each country reviews the allocated spectrum available to their region of operation. The Physical (PHY) layer handles message sending and receiving, collision detection, and bit error calculation [5]. The PHY layer also informs the Media Access Control (MAC) layer about signal detected on the channel. The function of MAC layer is to coordinate the use of the communication medium. MAC layer protocol decides which node should access the shared medium at any given time. The information received in the MAC layer; it is used to compute signal that represents the packet [6]. IEEE 802.11p physical layer is identical to IEEE 802.11a. IEEE 802.11a PHY layer employs 64-sub-carrier OFDM, out of which only 52 is used for actual transmission consisting of 48 data sub-carriers and four pilot sub-carriers. Moreover, the transmission power may be higher (up to 44 dBm) in 802.11p compared to that in 802.11a. IEEE 802.11p MAC layer is derived from the basic IEEE-802.11 Distributed Coordination Function (DCF).

Revised Version Manuscript Received on November 24, 2015.

Kenneth Sorle Nwizege, School of Engineering Elect/Elect Engineering Dept. Ken Saro-Wiwa Polytechnic Bori, Nigeria.

Harry, Inye H, School of Engineering Elect/Elect Engineering Dept. Ken Saro-Wiwa Polytechnic Bori, Nigeria.

Irimiagha Paul Gibson, School of Engineering Elect/Elect Engineering Dept. Ken Saro-Wiwa Polytechnic Bori, Nigeria.

The rest of the paper is organized as follows: Section 2 is related works, while Section 3 is overview of OMNeT++ simulator, Section 4 deals with rate adaptation in vehicular networks, and Section 5 is proposed results. Finally, Section 6 concludes this paper.

II. RELATED WORKS

The rapid trend in wireless technology applications, and especially in vehicular communications have given many researchers the interest in the subject of rate adaption. Some of the existing RAs are: Adaptive Auto Rate Fallback (AARF) [1], Context-Aware Rate Selection (CARS), SampleRate [2], Onoe [7], Context-Aware Rate Algorithm (CARA) [8], and Robust Rate Adaptation Algorithm (RRAA) [10] have been proposed in the literature. However, only AARF among the mentioned RAAs has been implemented in OMNeT++.

III. OVERVIEW OF OMNET++ SIMULATOR

OMNeT++ has standard tool to study protocols for both wired and wireless networks. There are various tools in OMNeT++ that enable users to achieve reliable results, depending on the choice of tool in OMNeT++ simulator. Some of these tools are MIXIM, SUMO, and inetmanet. In this research, we have chosen to use inetmanet because it contains the features required in implementing ACARS. It also provides detailed models and protocols, as well as supporting infrastructures. Proper choice of mobility model is very vital when dealing with vehicular communications. This tool has already implemented mobility models, and it is also flexible in when it comes to creating your own model.

A. Simulation model

The interest of researchers in wireless and mobile communication has been on the increase and many wireless networks are evaluated using OMNeT++. It is flexible to use because of its Graphical User Interface (GUI). You can see what you have coded and edit code. One of the challenging tasks in vehicular communications is how to combat with high mobility of vehicles to meet the requirement of Dedicated Short Range Communications (DSRC). It is important that proper choice should be made when considering mobility model to be used. There are several mobility models such as Circle, Linear, Rectangle, Random Walk model, Freeway, ConstSpeed, RAndomWaypoint, Manhattan. In this research, the implemented mobility type that determines how the nodes move around is Linear mobility.

B. Linear mobility

This mobility model allows node to reflect upon reaching the end. The key features of this mobility type are: speed,

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

angle, and acceleration. With equation 3, the positions of the vehicles are updated. The vehicle co-ordinates are calculated using figure 1.0.

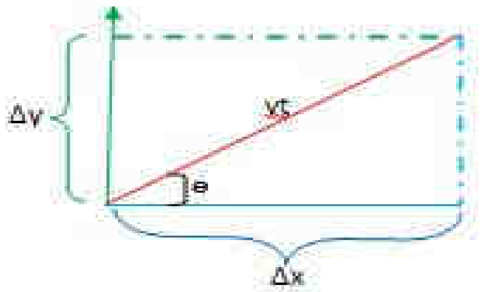


Figure 1.0: How to Calculate Vehicle Position from co-ordinates.

$\Delta x = vt \cos\theta$ (1)

$\Delta y = vt \sin\theta$ (2)

To get the movement of the node we use these equations:

$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + \cdots + (x_n - y_n)^2}$ (3)

where **d** is distance in meters, **x** and **y** are vehicle positions in the **x** and **y** coordinates, **t** is time, and **θ** is angle.

ῦ= đ * v (4)

where **ῦ** is last speed, **đ** is the direction, and **v** is speed.

ţ = ŧ - ŋ (5)

where **ţ** is elapsed time, **ŧ** is simulation time, and **ŋ** is last update.

ρ = ῦ * ţ (6)

where **ρ** is last position , **ῦ** is last speed, and **ţ** is elapsed time .

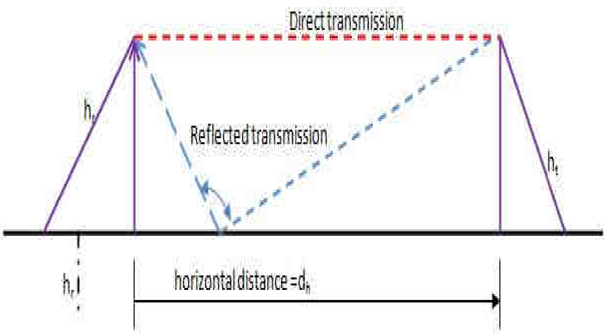
C. Radio model in inetmanet

Propagation model in inetmanet is handled by the radio model. Some of the implemented models relevant to our research in this simulator include, Friis free space loss, log normal shadowing, and two ray reflection model. These three implemented models are shown in equations 7-9 below. Figure 3.1 shows the two-ray model.

$L = P_t * g_r * g_r * (\lambda^2)$ (7)

$P_r = \frac{P_t * g_t * g_r * (h_t^2 * h_r^2)}{D^4 * L}$ (8)

$P_r = P_t - L$ (9)



where L= path loss, P_t transmit power, P_r received power, g_r is received gain, g_t is transmit gain, h_r is received height, h_t is transmit height, D is distance, and λ is wavelength.

Figure 2: Two ray diagram.

IV. RATE ADAPATION IN VEHICULAR NETWORKS

The study of rate adaptation is important because of the frame losses that occurs during movement of vehicles, and a good rate algorithm should be able to differentiate frame losses. This process dynamically switches data rates to match the channel conditions, with the goal of selecting the best transmission rate that will yield a maximum throughput at all times. The effectiveness of a rate adaptation scheme depends greatly on how fast it may respond to the wireless channel variation [8]. It is a critical component to ensure optimal system performance in a dynamic mobile environment. One of the goals of rate adaptation is to maximize throughput via exploiting the multiple transmission rates available for 802.11 devices by adjusting their transmission rates dynamically based on to the time-varying and location dependent wireless channel conditions. It is very important to be aware that transmission failure in rate adaptation occurs for two reasons, collisions and poor channels.

A. Implementation of Onoe rate algorithm

Onoe is a very slowly adapting algorithm whose implementation is available in the MADWifi driver code [3]. It tries to change rate after one second interval. It is a credit-based algorithm that maintains the credit score of the current rate for every destination and after the end of a second; it calculates the credit and makes the rate change decision. Onoe [4] was developed by MadWifi organization for Wi-Fi adapters with Atheros chips .It is a credit based algorithm. As a slowly adapting algorithm it tries to change the rate after 1s interval, by maintaining the credit score of the current rate for every destination and after the end of a second it calculates the credit and makes the rate decision [1][7][9]. For each individual destination, the Onoe algorithm keeps track of the current bit-rate for the link and the number of credits that bit-rate has accumulated. It only keeps track of these credits for the current bit-rate and increments the credit if it is performing with very little packet loss. Once a bit-rate has accumulated a threshold value of credits, Onoe will increase the bit-rate. If a few error conditions occur, the credits will be reset and the bit-rate will decrease [4]. It calculates the credit for the current transmission rate based on the packet loss ratio. The initial rate credit is set to be 0. Onoe observes received ACKs and increases the credit by 1 and then increased to 1 when less than 10% of frames in a time window of 1 second need retransmission (and the total frame transmissions are at least 10) otherwise it decrement it by 1. When the credit for the current rate reaches 10, it will increase the rate to the next higher level, else it decrements it to the next lower level if 10 or more frame transmission have been sent and more than 50% of the frame transmissions failed during the last period. Credit is reset to 0 when the rate changes. Onoe tries to find the highest bit-rate with less than a 50% loss rate, so it will step down from 11 megabits and will settle on 5.5 megabits. It decreases the bit-rate when the packets need at least 1 retry on average and increases the bit-rate

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

when less than 10% of packets require a retry. Onoe is said to be a conservative algorithm because it does not increase the current transmission rate when it detects good channel quality, but waits until the credit values threshold is reached. Once it observes that a bit-rate will not work, it will not attempt to step up again until at least 10 seconds have gone by. The Onoe algorithm can also take time to stabilize [4]. This algorithm is summarised in Algorithm 1.

B. Overview of Onoe algorithm

In this paper, we implement Onoe as a new RAA in inetmanet. Summary of this implementation is shown in Algorithm 1.

Algorithm 1 Onoe algorithm

1: **if** (Ђ == 0)
{
2: Я = $\delta_{\mathbf{Rate}}$
reset()
}
3: **else if**(Ђ > 10 && (ή OR Ђ) > 1)
{
4: Я = $\delta_{\mathbf{Rate}}$
reset()
}
5: **else if** (**ɱ**$_{Retry}$ OR Ђ) > 0.1)
6: $\delta_{\mathbf{Credit}}$ }
7:**else if** (**ɱ**$_{Retry}$ OR Ђ) < 0.1) }
8: $\Delta_{\mathbf{Credit}}$
}
9: **if** (Ђ == *NONE* && credits >= 10)
{
10: Ђ = $\Delta_{\mathbf{Rate}}$
11: reset() }
12: **return** Я

From Algorithm 1, Ђ / ή (success/failedCounters) give the output of success or failure of each decision from Onoe algorithm, while **Я** (result decision) finally gives the result of increasing /decreasing credit or bit-rate. $\boldsymbol{\delta}_{\mathbf{Rate}}$ Is for decrease rate, $\boldsymbol{\Delta}_{\mathbf{Rate}}$ for increase rate, $\boldsymbol{\Delta}_{\mathbf{Credit}}$ for increase credit, $\boldsymbol{\delta}_{\mathbf{Credit}}$ for decreasing credit, and **ɱ**$_{Retry}$ represents message retry.

C. Simulation configuration and set up

In this scenario, all vehicles act as clients while the Road Side Unit (RSU) is acts as the receiver. Vehicles generates their positions randomly, while distance between vehicles and AP was calculated from equation (18). Our scenario consists of a road length of 200 m with a communication range of 300 m, as shown in figure 4.1. Figure 4.0 shows vehicular communication and interaction with aid of RSU. Other parameters for this scenario are shown in table 1. All vehicles start at the same time at the beginning of the road, and establishes connection once in range with the server (AP) which is located at the middle of the road, known as RSU. Vehicles selects speed uniformly over this range and to ensure that all vehicles were within 0 and x_max (length of high way).

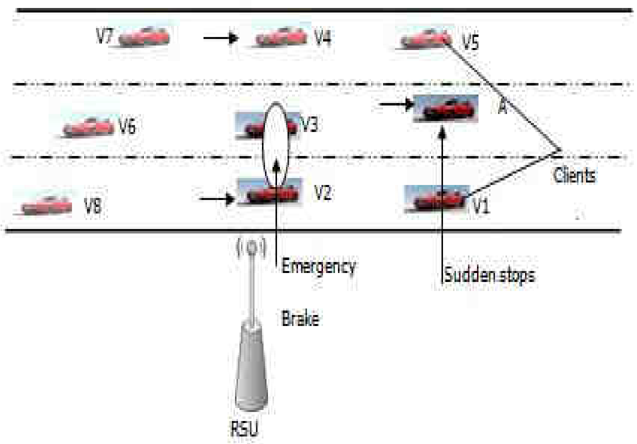


Figure 3: Vehicular scenario.

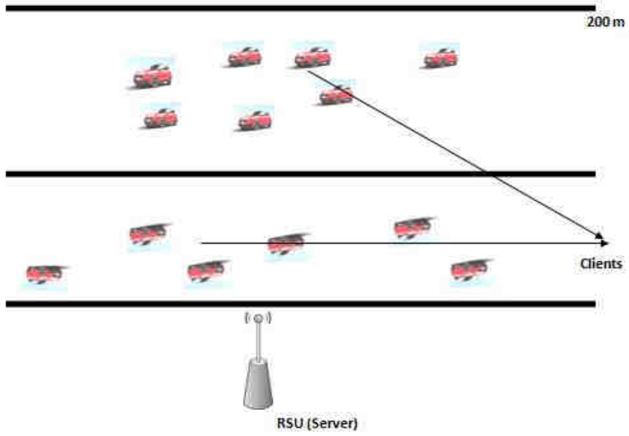


Figure 4: Vehicle- to infrastructure communication.

V. PROPOSED RESULTS

In this section, we present some results obtained in our simulation using Onoe algorithm. These results are called proposed because we are still making analysis on them, and also trying to verify results with theories and results obtained from other simulation platforms. Simulation parameters are shown in table 1.

TABLE 1. CONFIGURATION PARAMETERS.

Parameters (Units)	Values
Length of Road (m)	200
Number of Vehicles	150
Position of AP (m)	100
PHY and MAC Protocol	802.11p
Frequency (GHz)	5.89
Carrier frequency (GHz)	2.4
Normalized Transmit Power (mW)	20
Maximum retry	7
Data rate (Mbps)	3, 4.5, 6,9,12,24,27
Simulation time (s)	3600
Update interval (ms)	100
Mobility speed (mps)	30
cwMulticast	31
cwMinData	31
rtsThresholdBytes (B)	3000
maxQueueSize	14

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

A. Analysis of proposed results

Results obtained from our simulations are shown in this section. Although we have not arrived at a presentable outcome, but we present results obtained so far from our simulations. Figure 5 and 6 show the simulation run for Onoe algorithm.

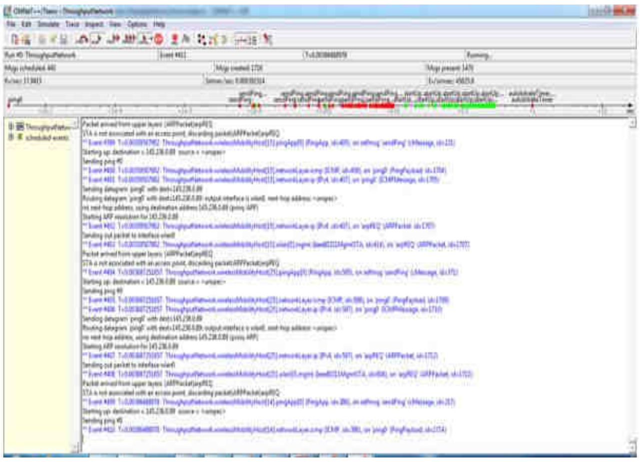


Figure 5: Simulation for Onoe algorithm.

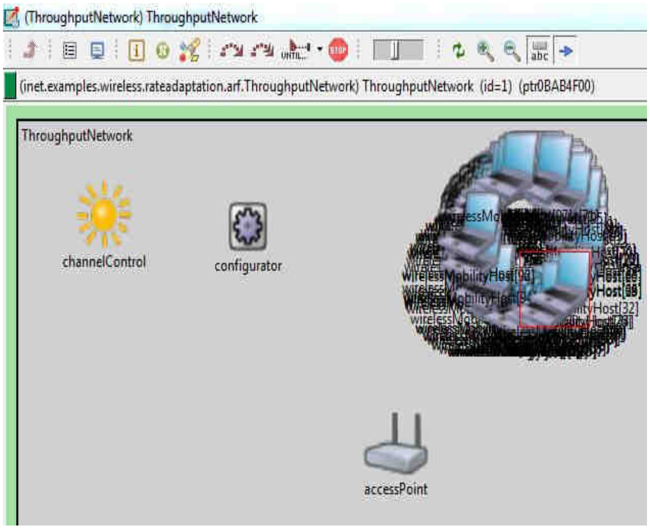


Figure 6: Network set up.

Also from result captured from inetmanet, we can see some parameters that would be analysed in future displayed in figure 7.

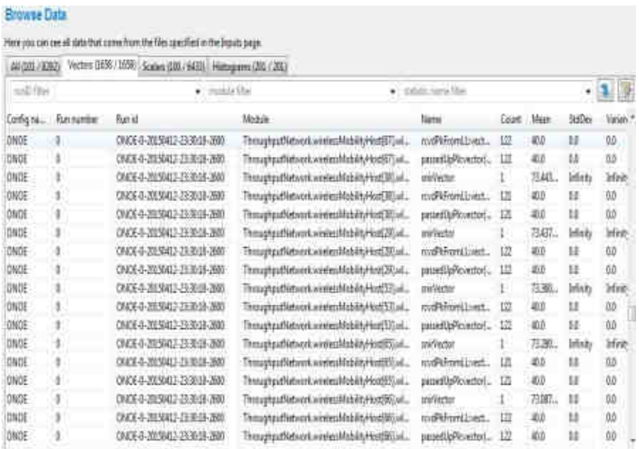


Figure 7: Network analysis.

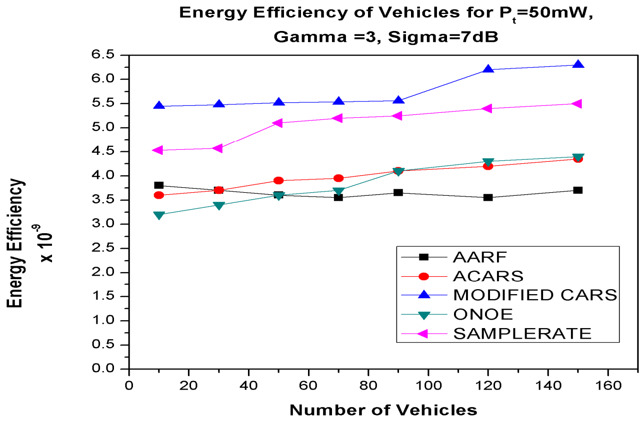


Figure 8: Energy Efficiency for σ= 7dB, P_t = 50mW, ϒ = 3.

VI. CONCLUSION AND FUTURE WORKS

From simulation results, we can only demonstrate implementation of Onoe algorithm, and since analysis is still in progress, we cannot conclude on the performance of Onoe algorithm compared to what we had in MATLAB or other experiments and tools by other researcher. Our first success is the implementation of this new algorithm in inetmanet which does not exist in the current version. Results will be published when we are done with our analysis as shown in Figure 8.

A. Future works

We are still making analysis on this simulation result, and also comparing results obtained in our experiments and simulations for authentication. We have implemented this algorithm in MATLAB, and this is a bench mark for out initial testing and authentication of results obtained using inetmanet.

REFERENCES

[1] P. Shankar, T. Nadeem. J. Rosca , and I. Iftode. Context Aware Rate Selection for Vehicular Networks. Department of Computer Science, Rutgers University, IEEE, 2008, pp 1-10.

[2] S. Wu. High Performance Rate Adaptation on IEEE 802.11 Networks. PhD Thesis, Auburn University, 2008, pp 34-60.

[3] A. Kamerman , and L. Monteban. A High-Performance Wireless LAN for the Unlicensed Band. AT&T Bell Laboratories Technical Journal, 1997, pp.118-133.

[4] J. C. Bicket. "Bit-rate selection in wireless networks," *Master's Thesis, MIT*, 2005, pp. 39-31.

[5] K. S. Nwizege, J. He, and M. Shedrack,. Optimizing Rate Algorithms in Wireless Networks. European Modelling and Simulation (EMS) , Madrid, 16-18 November, 2011, pp 2-6.

[6] S . R. Saunders, and A. A. Zavala. Antennas and Propagation for Wireless Communication systems. John Wiley and Sons , Ltd, second edition, 2007, pp 172-178.

[7] K. S. Nwizege, and J. He. ACARS. Adaptive Context Aware Rate Selection Algorithms in Vehicular Networks. International Journal of Convergence Information Technology , April 2013, pp 1-5.

[8] S. C. J. Kim, S. Kim, and D. Qiao. CARA: Collision-Aware Rate Adaptation for IEEE 801.11 WLANs. IEEE Communication Society, 2006, pp 1-3.

[9] K. S. Nwizege, and M. MacMammah. Performance Evaluation of Path Loss Exponents on Rate Algorithms in Vehicular Networks International Journal of Emerging Science and Engineering (IJESE), August 2013, vol.1, pp 103-108.

[10] F. Karnadi, Z. H. Mo, and K.C. Lan. Rapid Generation of Realistic Mobility Models for VANET. March 2007, pp 2509-2513.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.