A Wireless Peer-to-Peer Broadcast Model for Emergency Vehicles Using Automotive Networking

Yilang Wu*, William Putnam*, Junbo Wang†, Zixue Cheng‡
*Graduate School of Computer Science and Engineering, University of Aizu, Fukushima, Japan
†Promotion Office for Top Global University, University of Aizu, Fukushima, Japan
‡School of Computer Science and Engineering, University of Aizu, Fukushima, Japan
E-mail: {y-wu, m5201152, j-wang, z-cheng}@u-aizu.ac.jp

***Abstract*—Automotive networks are simple, real-time networks with very low error rates. In-vehicle infotainment (IVI) systems include GPS-based navigation and wireless hotspots for cellular communication. In the event of a power outage or a major catastrophe, such as an earthquake, existing network mainframes may either shut down completely or become overloaded with traffic. In that regard, it is important for emergency vehicles to spread the word of certain emergencies, such as road closures and traffic accidents, to vehicles within the area. In this paper, we propose a method using the peer-to-peer broadcast model to transmit emergency messages between vehicles using IVI and GPS systems. A distributed peer-to-peer algorithm is presented to deliver a high priority message and avoid acceptance of duplicated ones to enlarge the broadcasting coverage of messages with higher priority. We also discussed the usability, feasibility and future improvement of this proposed model.**

***Index Terms*—automotive networks, in-vehicle infotainment, emergency systems, peer-to-peer**

I. Introduction

Technology has been able to be integrated into many things thanks to concepts such as embedded systems and the Internet of Things (IoT). Such technology has also been extended into the vehicles that transport us.

Many modern-day vehicles offer in-vehicle infotainment (IVI) systems, with some system features included as standard features at the point of sale. IVI systems combine many features of a vehicle that were once separate, such as the radio, navigation system, and climate control. These systems even allow users to sync their mobile devices, or use the system as a wireless hotspot that can detect wireless and cellular signals.

However, despite this connectivity, many useful features are still missing, as shown in the scenario on the left side of Figure 1. One is the ability to be made aware of issues within the surrounding environment. Emergency warnings that may be deployed to a separate device like a cellphone are of little use because the driver compromises safe operation of the vehicle by checking the device while moving. At the same time, various environment factors such as population density, alert priority, and the scale and nature of disasters (for example, a simple car accident versus a major earthquake) can make it difficult for drivers to receive word of an issue otherwise.

The purpose of this paper is to develop a model that would be useful for emergency vehicles (as shown on the right side of Figure 1) to broadcast messages to different vehicles in the surrounding area. These vehicles will also pass on received messages to surrounding vehicles within a given distance to warn drivers of area calamities.

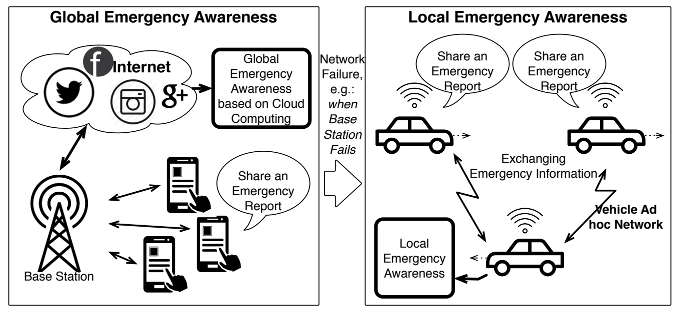

Fig. 1. Scenario of IVI-based Local Emergency Awareness

The rest of this paper is outlined as follows. Section II discusses previous research conducted related to the issue. Section III introduces the underlying framework of our model. Section IV outlines the equations and pseudocode required to achieve this framework. The overall results are discussed in Section V, and finally, the conclusion and plans for future development are discussed in Section VI.

II. Related Work

There is a bevy of different research into automotive networks within the past five years. This research has covered everything from the automotive network itself to the feasibility of large-scale systems. For this paper, we focused on four different fields: the automotive networks themselves, the various types of IVI models available, existing protocols for emergency communication, and systems for monitoring surrounding environments.

A. Automotive Networks

Some automotive networks, such as the Vehicular ad-hoc network (VANET), automatically establish a mobile network so that vehicles within a particular communication range can exchange various pieces of information such as speed and location obtained via GPS and sensors. [1] VANET has already been implemented in some cases, mainly due to its easy integration into existing architectures. [2]

The connectivity of automotive networks to external sources remains a critical issue, [3] with signal strength and barriers such as building layouts in urban areas playing important factors to total connectivity between vehicles in the same neighborhood, or even the same block. Depending on the type of message being sent, especially of a higher priority, said message may not be able to reach other drivers in time, even if said drivers are theoretically in a particular broadcast range.

Other automotive networks are called opportunistic networks, in which a source node originates a message that is forwarded to an intermediate node which serves as a bridge to the destination node. The intermediate node stores the message and carries it while the destination node is not available. [4]

B. IVI Connectivity Models

IVI systems have become central to a driver's automotive experience in recent years. Thanks to developments in mobile devices and embedded systems, [5] it is possible for these systems to perform multiple functions for multimedia applications, such as controlling traditional and streaming radio, and operating the GPS-based navigation systems. Such systems can also control various features of the car itself, such as the suspension and climate control.

Currently there are several challenging network connectivity issues in opportunistic automotive networks, such as sparse connectivity, long or variable delay, intermittent connectivity, asymmetric data rate, high latency, high error rates and even no end-to-end. [6] In addition, previous research [7] has highlighted the lack of upgradability for IVI systems with respect to synced mobile devices, which makes it harder to make newer technologies and methods backwards compatible.

C. Emergency Communication Network

Big data analytics in a disaster area provides solutions to understand the various situations occurring in disaster areas. [8] Content analytics, such as support vector machines and hidden Markov models, help the understanding of emergent crowd-sourced contents collected through both social and physical sensing networks. Spatial analytics, such as spatial clustering and spatial co-location detection, help the understanding of spatial and temporal distributions of emergencies. However, such big data analytics are usually in a large spatial range, and computations can not be easily performed due to insufficient data collection or computation resources. Plus, from a local point of view, drivers can also share the emergency information by passing through an automotive ad-hoc network.

D. Environment Monitoring Systems

There has been plenty of research performed regarding the monitoring and detection of automotive activity. A good example is research performed by Jose, Prasad, and Sridhar [9], where they designed a model in both hardware and software that can collect data and store it at a third-party site, such as a cloud server, for further analysis.

Many different models have been designed to detect various environmental features that may be of significant consequence, all while being adapted to fault-tolerant networks. For example, a model detailing the monitoring of road conditions [10] adheres to the Delay Tolerant Networking (DTN) configuration for areas with sparse populations or limited network coverage due to disasters affecting access points. Another example [11] involves using the IoT to collect visual data on actual road conditions.

The above-mentioned models and more must have their data successfully transmitted to surrounding vehicles in a given area. However, in addition to the issues stated in Section II-B, there are issues with these particular models as well. Djahel, Doolan, Muntean, and Murphy [12] have determined that some factors, such as special social events and traffic ettiquete, are hard to mitigate and can interfere with the desired outcomes of a particular monitoring system, even at a large infrastructure level.

III. Model Design

This section considers two important features of the model: the message itself, and how said message is actually passed.

A. Message Protocol

VEHICLE_ID	TIMESTAMP	LOCATION (COORD)	EMERGENCY CONTENT	
			EMG_STATUS	PRIORITY

Fig. 2. Message Unit of Wireless Peer-to-Peer Broadcast Model

Figure 2 shows a message protocol for an emergency message. The various parts of the message are as follows:

- *VEHICLE_ID* - This identifier references the vehicle ID that sent the information to surrounding vehicles. It is possible for security reasons to overwrite this field with zeroes to improve message security.
- *TIMESTAMP* - This identifier references the date and time that the message was originally sent from the vehicle.
- *LOCATION* - This identifier references the location at which an incident, be it a car accident or natural disaster, has taken place. This data could theoretically be passed onto the vehicle's navigation system to be displayed using its local maps.
- *Emergency Content* - This identifier is broken up into two pieces:
  - *EMG_STATUS* - This piece stores a hexadecimal code that would serve as a general emergency message. This configuration would reduce the hassle and complexity of sending a message with actual text.
  - *PRIORITY* - This piece highlights the priority, or emergency, level of the message to be transferred. A priority level of one would relate to a relatively trivial event, such as a minor fender bender, while a priority level of five would relate to a serious disaster, such as an earthquake or a landslide. In the case of the model, the higher this level, the farther that the message is transmitted.

B. Message Passing Protocol

Figure 3 shows an series of connectivity cases for a given ad-hoc network. Vehicles are represented as moving nodes,

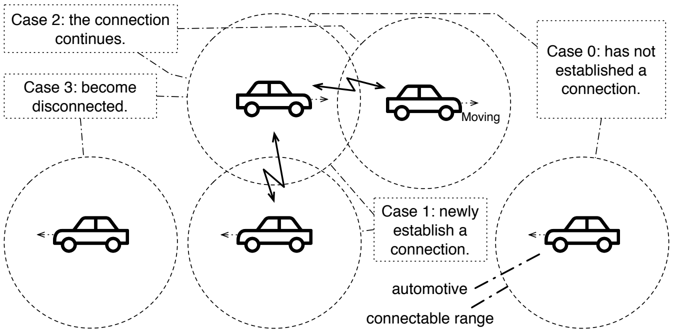

Fig. 3. Connection Cases of Wireless Peer-to-Peer Broadcast Model

and each of them has a limited distance of wireless network communication, which is denoted by the dotted circles. A networking scanning module searches for neighboring nodes that are within the communication spatial range or distance.

There are four connection cases (as shown in the figure):

1) When two nodes are still very far away from each other, no connection is established.
2) When two nodes get within communication distance of each other, a new connection is established. Both nodes are added to each other's connected node list.
3) Connection between the nodes continue so long as both nodes remain within each other's range.
4) When two connected nodes distance themselves too far apart, they will lose connection if they are out of communication distance for longer than a specified threshold period.

IV. Algorithm Design

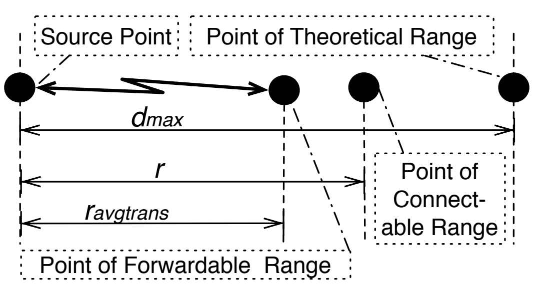

Fig. 4. Message Passing through Wireless Peer-to-Peer Broadcast Model

Figure 4 shows a diagram of the various points of a given ad-hoc network as defined by variables. Here, there are four major variables:

- d_{max} - the maximum distance that any message can theoretically travel. This is a constant value.
- *r* - the broadcasting radius from a given creation node.
- $r_{avgtrans}$ - the average radius of successful message transmission.
- *R* - the total distance that a message has travelled in its lifetime.

In the case of the above-mentioned figure, the leftmost point is the *source point*, where the message is created. It is then sent to another node, which will check to make sure that the message is not duplicated, and then check to see if it is still within the range of r and, to an extent, d_{max}. These respective nodes are the right two nodes in the figure.

A. Equations

There are two equations to represent the model:

- Equation 1 determines the lifetime for a given message to be broadcasted. Here, the length of time that a message is broadcasted depends on its priority factor *p*. The t_{max} variable represents the default time that a message should be broadcasted with no priority. A higher priority factor increases the message's broadcast lifetime.
- Equation 2 determines the broadcast radius for the message from the source point. This result could be used as a variable upper limit for the broadcasting of a message. Again, the priority level determines the radius that the message should travel.
- Equation 3 determines the total travelled distance of a message across all nodes that said message has tried to access. The s_i variable represents a binary value (0 or 1) determining if a message attempt was truly successful at each peer-to-peer transmission. We also assume that the number of entries to be added to the range total *R* is related to the message's lifetime over its average transmission time between two nodes.

$$t_{life} = t_{max} \times p \tag{1}$$

$$r = \frac{p}{p_{max}} \times d_{max} \tag{2}$$

$$R = max \sum_{i=1}^{t_{life}/t_{avgtrans}} r_{avgtrans} \times s_i, \quad \text{where } s_i \in 0, 1 \text{ and } r_{avgtrans} \leq r \tag{3}$$

B. Pseudocode

Algorithms 1 and 2 describe five different functions for the handling of data:

- *create_message* - This function, provided exclusively for emergency vehicles, creates a struct containing message data and passes it off to the *send_data* function.
- *send_data* - This function searches for all of the vehicle in a given area that it may or may not yet have connections to. After proper connections are established or disestablished, the message is sent out to each vehicle.
- *rec_data* - This function loops until a message is received, and then determines is not a duplicate. If so, it then checks to see if the message is still worthy of further transmission. If that is the case, it then calls the *send_data* function.
- *calculate_location* - This is a helper function for the *rec_data* function. It calculates the difference in distance travelled between two nodes and returns a one-dimensional distance instead of a two-dimensional coordinate pair.

- *calculate_time* - This is also a helper function for the *rec_data* function. It calculates the difference in time between the receiving vehicle's system time and the timestamp of the original message. We assume in this model that the sending and receiving vehicle's system time are properly synchronized with each other.

Algorithm 1 Pseudocode for Message Creation and Sending

```
data := Struct{vehicle_id, timestamp, coord, emg_status, emg_level, life}   ▷ Let
  Data be the data of message.
t_max, p_max, and d_max are constants

procedure CREATE_MESSAGE(status, level)                    ▷ Create a message
    life := level * t_max
    data.vehicle_id := get_vehicle_id()
    data.timestamp := get_system_time()
    data.emg_status := status
    data.emg_level := level
    data.life := life
    send_data(data)
end procedure

procedure SEND_DATA(data)                                   ▷ Send a message
    vehicles := array of vehicles in r
    vehicles := search_for_vehicles()
    vehicles[data.vehicle_id] := null
    for i in vehicles do
        send(i, data)
    end for
end procedure
```

There are a few factors to consider in this pseudocode. One is that the class type for the *vehicles* array allows for the storage of an average signal strength, which could be used to determine which nodes to send a message to first. Another factor is that the *search_for_vehicles* function, while searching for surrounding vehicles, will automatically sort and return the array based on the signal strength. That way, nodes that are closer to the broadcasting vehicle stand a better chance of receiving a message.

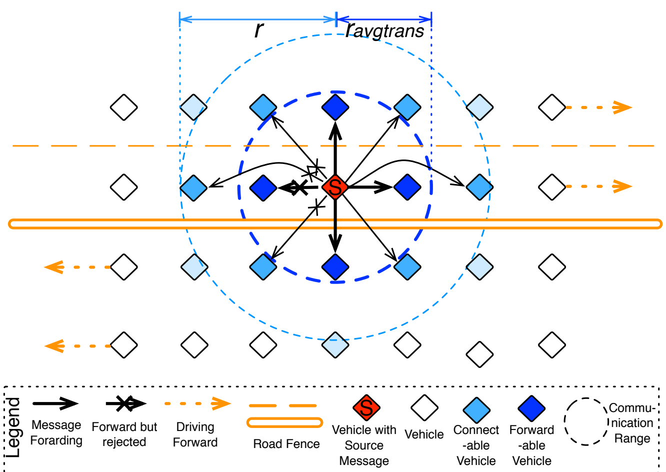

Fig. 5. Message Forwarding through Wireless Peer-to-Peer Broadcast Model

Figure 5 shows a rough outline of how the data passage should work. The red node is the emergency vehicle creating and initially sending the message. It sends the message to all of the vehicles within its immediate range. The different shades of blue on the surrounding nodes indicate the various degrees to which the message could be sent by the emergency vehicle. The dark blue nodes are perfect candidates because they are the closest to the emergency vehicle. The regular blue nodes share intersections with other ranges. The light blue and white nodes would be considered unnecessary to send to, as closer nodes would be able to send the message on behalf of the emergency vehicle provided they remain within range.

Algorithm 2 Pseudocode for Message Receipt

```
procedure REC_DATA( )                                       ▷ Receive a message
    coord := vehicle location
    data := message data
    t_diff := elapsed message time
    r := distance for message to travel
    while data := null do
        data := Listen()
    end while
    if verify_msg(data.timestamp, data.vehicle_id) := true then
        t_diff := calculate_time(data.timestamp)
        d_tr := calculate_location(data.coord)
        r := p / p_max * d_max
        if d_tr < r and t_diff < t_life then
            send_data(data)                   ▷ Forward the received message.
        end if
    end if
end procedure

procedure CALCULATE_LOCATION(data.coord)  ▷ Calculate the distance travelled
  from the origin coord := host car's current location
    diff := coord - data.coord return conv_coord_to_distance(diff)
end procedure

procedure CALCULATE_TIME(data.timestamp)     ▷ Calculate the elapsed message
  time
    diff := get_system_time() - data.timestamp

    return conv_systime_to_time(diff)
end procedure
```

The pseudocode defined in this model does not explicitly define any technical details or how a message is verified. With regards to the latter, there are many ways to organize the message data to check for duplicate messages. For example, a local database could be created for each car that holds the message data. Regardless of the actual method, the message's origin vehicle ID and the timestamp of the original message must be provided to accurately determine if a message is a copy, regardless of the number of hops that message has travelled before being sent back to the sender.

V. Discussion

In this section, we review the model's novelty and feasibility. We also discuss the various existing mediums in how the model could technically be implemented, both for existing and future implementations.

A. Relevance of the Model

Compared to a centralized emergency communication network, an IVI-based peer-to-peer network is more advanced in a local range. It has three major characteristics:

- *Promptness* - prompt awareness of local emergency based on the broadcast of emergency message.
- *Reliability* - a complementary to traditional wireless network, and is important when other network fails.

- *Extensibility*: can be extended to wearable devices or in-bicycle infotainment when it achieves low power consumption.

However, there are also some potential downsides:

- *Security* - messages only rely on a message ID instead of a network address, which could lead to poor security
- *Limited bandwidth* - the throughput is relatively low due to the short connection times of pair-wise connections among moving vehicles, and is mainly for transferring high-priority messages

B. Implementation Mediums

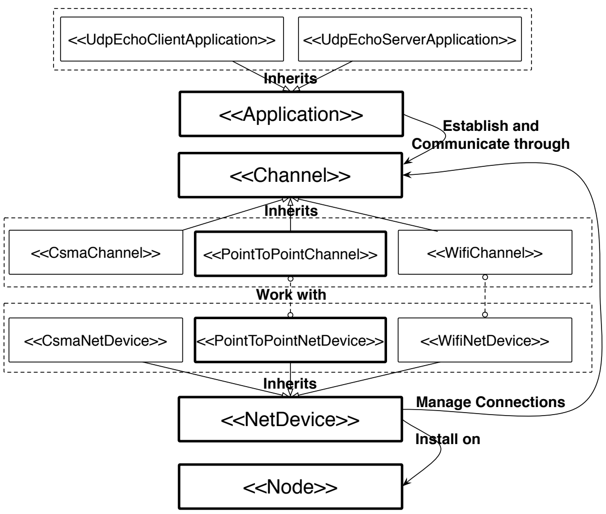

Fig. 6. Commonly used NS-3 (C++) Classes for Network Simulation

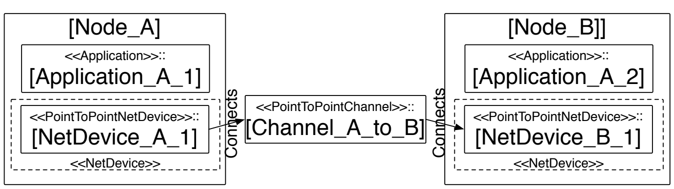

Fig. 7. Initialized Objects in A Sample Network

Figure 6 shows the general class structure in using an ns-3 based application. Our experiments will use the UdpEchoClient and UdpEchoServer helpers to construct a network for communication between wireless nodes. These applications rely on Channel-type object tied to NetDevice-type objects that are installed onto the nodes. Figure 7 shows the interactions between two initialized nodes over a pair of PointToPointChannels. (These channels are configured as one-way.)

Finally, Figure 8 is a sequential diagram of the message broadcasting process outlined in Section III. For this diagram, four waypoints are considered:

- the emergency that warrants the message creation (*Emergency*)
- the creator of the message (*Node A*)
- the first recipient of the message (*Node B*)
- the second recipient of the message (*Node X*)

The message msg_{e1} related to emergency *e1* is generated by *Node A* and sent to *Node B*. After *Node B* receives, verifies, and processes the message, it sends the message to *Node X*.

Implementation of the model can be relatively flexible. Because there is no large infrastructure involved with implementation, such as cellphone towers or cloud servers, only the actual vehicles need to be considered. Emergency vehicles, such as police vehicles, firetrucks, and ambulances, and fixed access points, such as local antenna towers, would have the ability to broadcast emergency messages. Theoretically, radio frequency (RF) would be the most efficient means of broadcasting the signal, with Bluetooth Low Energy (BLE) being a decent alternative.

In addition, a common protocol would have to be developed for all road-legal vehicles to handle the message passing. For vehicles developed within the past three years, this would be relatively simple, as a firmware update of the IVI system would be able to utilize the existing hardware on board a vehicle already equipped with the proper hardware. However, expanding this coverage to older vehicles or newer vehicles without the proper hardware will be difficult. The best way to approach this particular issue would be to implement the new protocol into aftermarket IVI systems so that older vehicles would be able to pick up and send emergency messages.

This model also doesn't necessarily have to be restricted to passenger vehicles. Said protocol could also be implemented with billboards and LED display signs. This method would make it easier for the above disadvantaged type of vehicles, as well as pedestrians, motorcyclists, and bicyclists, to be aware of higher priority events within the area.

C. Demonstration of Peer-to-peer Broadcasting

TABLE I
TEST SYSTEM INTEGRATED DEVELOPMENT ENVIRONMENT (IDE)

Category	Specification
Virtual Machine	Virtualbox v5.0.26 r108824
OS	Ubuntu 16.04.1 LTS x64, Linux kernel v4.4.0-38-generic
Software	Python 3.5.2 :: Anaconda 4.1.1 (64-bit), ns-3.25, g++ (Ubuntu 5.4.0-6ubuntu1 16.04.2) 5.4.0 20160609

TABLE II
A SAMPLE MESSAGE CONTENTS

VEHICLE_ID	TIMESTAMP	LOCATION (COORD)	EMERGENCY CONTENT	
			EMG_STATUS	PRIORITY
abc1234567890	Mon Jul 04 15:31:15 JST 2016	37.4948 139.9298	0x1126	2

TABLE III
PERFORMANCE OF SIMULATING A PEER-TO-PEER BROADCASTING

Attribute	Performance in NS-3
Total simulation time (s)	8
Message transfer time (s)	0.01504
Echo of packet contents (s)	0.01278
Byte count of message	58
Simulation memory consumption (kB)	75.9453125

A simple demo system has been implemented based on the IDE outlined in Table I. After initializing a peer-to-peer network (see Figure 7), the system sets a test message based on the message protocol outlined in Figure 2. The message contents are displayed in Table II. This message is broadcasted

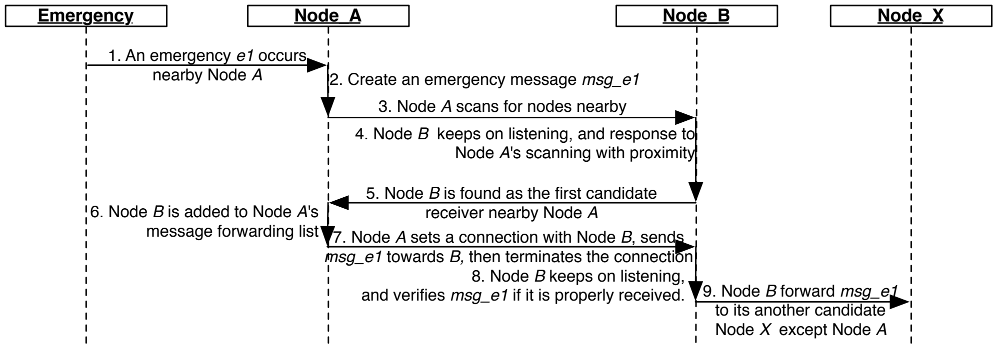

Fig. 8. Sequential Diagram of Message Broadcasting Process

from *Node A* to *Node B*. The performance of the peer-to-peer simulation is given in Table III.

VI. Conclusion

The model presented in this paper seeks to solve the issue regarding peer-to-peer message passing for emergency vehicles. We have presented some basic equations and pseudocode to serve as a baseline for any future systems that derive from this model. In times of regular network outages (e.g. no cellular reception or wireless Internet), this model will be a key player in helping civilians stay alert to ongoing emergencies in their community.

For the current stage, simulation of the pseudocode has not yet been performed. This shall be done using the ns-3 simulator [13] for networking simulation. Once the simulation is complete, we will have simulated results for the variables for range and average travel time between nodes.

One possible improvement to the model brought forth in this paper is the mitigation of traffic congestion in networks with low bandwidth. The model presented in this paper assumes that there are no bandwidth issues with individual receivers, and that said receiver can accept multiple messages at the same time. However, in the case where this is not true, message collision would be a serious issue that would prevent important messages from getting to the vehicle in time. Therefore, a method we need to be developed, mostly in code, that would be able to prioritize multiple incoming messages based on a number of factors such as signal strength, priority factor, and duplication.

An additional possible improvement is an increased layer of security to the overall model. For example, the current version of this model does not account for spoofing, which is this case is the act of a hacker trying to pass along a message under a false identity. Message spoofing can lead to false information being passed to the driver. This flaw can be especially dangerous if a method like the above improvement were to be implemented, as a hacker could successfully manipulate a vehicle to perform an action against the wishes of the driver.

Acknowledgment

This research was supported by JST-NSF joint funding, Strategic International Collaborative Research Program, SICORP, entitled “Dynamic Evolution of Smartphone-Based Emergency Communications Network”, from 2015 to 2018. The authors would like to thank the professors, staff, and other students of the Computer Network Systems Laboratory at the University of Aizu for their continuous and gracious support.

References

[1] T. ETSI, “Intelligent transport systems (its); vehicular communications; basic set of applications,” Definitions. Technical Report 102 638, Tech. Rep., 2009.

[2] M. C. Surugiu and R. V. Alexandrescu, “Analysis of the development and implementation of vanet network intervehicular communication systems,” in *Electronics, Computers and Artificial Intelligence (ECAI), 2013 International Conference on*, June 2013, pp. 1–6.

[3] A. Pentland, R. Fletcher, and A. Hasson, “Daknet: Rethinking connectivity in developing nations,” *Computer*, vol. 37, no. 1, pp. 78–83, 2004.

[4] L. Pelusi, A. Passarella, and M. Conti, “Opportunistic networking: data forwarding in disconnected mobile ad hoc networks,” *IEEE Communications Magazine*, vol. 44, no. 11, pp. 134–141, 2006.

[5] M. Faezipour, M. Nourani, A. Saeed, and S. Addepalli, “Progress and challenges in intelligent vehicle area networks,” *Commun. ACM*, vol. 55, no. 2, pp. 90–100, Feb. 2012. [Online]. Available: http://doi.acm.org/10.1145/2076450.2076470

[6] M. P. Singh, P. K. Shukla, and A. J. Deen, “Relay assisted epidemic routing scheme for vehicular ad hoc network,” *International Journal of Computer Science and Information Security*, vol. 11, no. 10, p. 22, 2013.

[7] S. Diewald, A. Mller, L. Roalter, M. Kranz, and T. U. M. Lehrstuhl, “Mobile device integration and interaction in the automotive domain,” in *In AutoNUI: Automotive Natural User Interfaces Workshop at the 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI 2011) (Nov.Dec*, 2011.

[8] J. Wang, Y. Wu, N. Yen, S. Guo, and Z. Cheng, “Big data analytics for emergency communication networks: A survey,” *IEEE Communications Surveys Tutorials*, vol. PP, no. 99, pp. 1–1, 2016.

[9] V. Vijayakumar, V. Neelanarayanan, D. Jose, S. Prasad, and V. Sridhar, “Big data, cloud and computing challenges intelligent vehicle monitoring using global positioning system and cloud computing,” *Procedia Computer Science*, vol. 50, pp. 440 – 446, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S187705091500513X

[10] K. Ito, G. Hirakawa, Y. Arai, and Y. Shibata, “A road condition monitoring system using various sensor data in challenged communication network environment,” in *Advanced Information Networking and Applications Workshops (WAINA), 2015 IEEE 29th International Conference on*, March 2015, pp. 518–523.

[11] Q. Li, H. Cheng, Y. Zhou, and G. Huo, “Road vehicle monitoring system based on intelligent visual internet of things,” *Journal of Sensors*, p. 16, 2015.

[12] S. Djahel, R. Doolan, G. M. Muntean, and J. Murphy, “A communications-oriented perspective on traffic management systems for smart cities: Challenges and innovative approaches,” *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 125–151, Firstquarter 2015.

[13] N. S. Foundation and P. group at INRIA Sophia Antipolis. (2016) ns-3. [Online]. Available: https://www.nsnam.org/