Balancing Safety and Routing Efficiency with VANET Beaconing Messages

Scott E. Carpenter
Student Member, IEEE
Department of Computer Science
North Carolina State University
Raleigh, NC, USA
scarpen@ncsu.edu

In Partial Fulfillment of the Requirements of
Computer Networking (CSC 570, fall, 2013),
David Thuente, Ph.D.

Abstract—Wireless communications between vehicles enables both safety applications, such as accident avoidance, and non-safety applications, such as traffic congestion alerts [1] with the intent of improving safety in driving conditions. The most promising technologies in the U.S. that will enable such a vehicular ad hoc network (VANET) are collectively referred to as Dedicated Short Range Communications (DSRC). However, the standards are weak on specifications regarding message size, transmission rates, retransmission strategies, and network routing. While studies using parametric strategies address these concerns, results are often quantified to either safety or routing efficiency, but rarely both. VANET literature often discusses beaconing as a solution to the information dissemination challenges. However, in VANET literature, the term beaconing is used with ambiguity, which could hinder the solutions that VANET application developers and network architects propose. A beacon represents a small packet transmitted with some regularity. We attempt to clarify further the term beaconing by adopting the terms safety beacon to mean the SAE standard Basic Safety Message (BSM) and beaconcasting to mean the transmissions involved in route discovery functionality. Clarifying beaconing motivates the following research objective: The goal of this research is to improve VANET application and routing design by precisely defining safety and routing beacon data so that transmission characteristics of their shared elements can maximize both safety and routing efficiency. Nonetheless, models exist that relate probability of packet reception (PPR) to driver awareness and traffic safety measures [2]. While altogether these models do not provide a consistent use of beaconing, they provide the modules that may support a methodology to do so. By decomposing beacon messages into contextual elements, we put forth the concept of contextual beaconing in which we aim to group related elements into the minimal messages with common requirements so as to maximize their delivery probability. Thoughtful adjustment of transmission parameters based on beacon data characteristics improves their chance of successful receipt by neighboring vehicles, and this in turn alleviates network overload and more importantly improves safety. Typical modeling within the literature commonly assumes fixed beacon packets on the order of hundreds of bytes. Reducing beacon message sizes leads to marked improvements in reception probability. Our work suggests an alternate beaconing policy which may be described as follows: To improve safety through maximal reception probability, frequently-transmitted safety beacons should be of minimal size and broadcast within distance-limited communication range, while additional safety and non-safety data elements should be scheduled with frequency, size, and intended range based upon their contextual beaconing requirements. We find that minimizing beacon size increases reception probability, improves VANET operational efficiency, and strengthens safety, thus supporting one of the primary goals of connected vehicle systems.

Keywords—VANET; Safety; Beacon; BSM; PPR; reception probability

I. INTRODUCTION

Collisions among moving vehicles lead all causes of traffic fatalities, injuries, and property damage[1]. By exchanging information using wireless communications, cars and trucks enable both safety applications, such as accident avoidance, and non-safety applications, such as traffic congestion alerts[1]. The US Intelligent Transportation Systems (ITS) Joint Program Office (JPO) suggests that vehicle safety applications and supporting technologies will prevent tens of thousands of automobile crashes

every year\textsuperscript{2}. The most promising technologies in the U.S. that will enable such a vehicular ad hoc network (VANET) are collectively referred to as Dedicated Short Range Communications (DSRC). However, the standards are weak on specifications regarding: message size; transmission rates; retransmission strategies; and network routing. While studies using parametric strategies address these concerns, results are often quantified to either safety or routing efficiency, but rarely both.

The information dissemination requirements within VANETs vary based upon application needs. For example, a safety-critical application such as forward collision warning system (FCWS) requires rapid information dispersal of the SAE J2735 Basic Safety Message (BSM) to nearby traffic \textsuperscript{3}. Other non-safety, broadcast-oriented applications, such as traffic-jam alert or vehicle-to-vehicle video chat require efficient information delivery to large or targeted areas with reduced reliability and delay requirements as compared to safety applications \textsuperscript{3}.

VANET literature often discusses beaconing as a solution to the information dissemination challenges. However, in VANET literature, the term beaconing is used with ambiguity, leading to confusion and conflicting claims. For example, some characterize several broadcast routing protocols as reliable \textsuperscript{4}, whereas others claim that beaconing is entirely unnecessary to routing performance \textsuperscript{5}. The Merriam-Webster Online dictionary defines beacon as a noun, meaning “a radio transmitter emitting signals to guide (aircraft)\textsuperscript{6},” whereas The Free Dictionary defines beaconing as a verb, meaning “to guide or warn”.\textsuperscript{7} It is precisely the partial adoption of the latter dual meaning that leads to the ambiguity. Inconsistent use of the term beacon/beaconing could hinder the solutions that VANET application developers and network architects propose. Nonetheless, analytical models for beacon traffic impact analysis exist that estimate BSM collision and expiration probability \textsuperscript{6}, relate congestion window (CW) size to probability for reception \textsuperscript{7} \textsuperscript{8}, and associate probability of packet reception (PPR) to driver awareness and traffic safety measures \textsuperscript{2}. While altogether these models do not provide a consistent use of beaconing, they provide the modules that may support a methodology to do so.

Especially in safety applications, both network routing requirements and application data requirements expect recurring data transmissions. A beacon, therefore, represents a small packet transmitted with some regularity. We attempt to clarify further the term beaconing by adopting the terms safety beacon to mean the BSM and beaconcasting to mean the transmissions involved in route discovery functionality. Clarifying beaconing motivates the following research objective: The goal of this research is to improve VANET application and routing design by precisely defining safety and routing beacon data so that transmission characteristics of their shared elements can maximize both safety and routing efficiency. Our investigations are guided by the following research questions (RQ):

RQ1 Can safety beacon and routing beacon data be categorized and described in a way that clarifies both functional and contextual meaning to their common elements?

A decomposition of beacon data beyond simple terms into exact data elements explains their use and allows analysis of their commonalities. For example, we know that the BSM safety beacon broadcasts a vehicle’s location, and we know that stateful VANET routing protocols desire to make local routing decisions using information from a neighbor information table. After routing protocols fail to find an existing route, they begin a neighbor discovery process that adds overhead in the transmission of routing beacons. However, a routing algorithm reduces beaconcasting overhead if it queries directly a neighbor information table populated from safety beacon data. Various routing protocols prefer different elements to exist in the neighbor information table, yet may not be guaranteed that BSM safety beacons provide the necessary information. Safety and routing processes thus require a functional coordination of data elements to achieve optimality. Such coordination requires a common and consistent vernacular.

RQ2 Can we relate how modifications to beacon data parameters impact transmission characteristics?

Adjustment of BSM parameters such as generation rate, transmission power, and contention window size impacts network characteristics, including reception probability \textsuperscript{4}. For safety and routing, an understanding of the relationship between parameters and the resulting efficiency is vital. For example, a routing protocol may need certain data elements to efficiently make routing decisions that the BSM may not already be transmitting. Even if a routing algorithm could request additional routing beacon data to be included in safety beacons, the impact of such a change needs to be measured. Similarly, outdated or expired BSMs should be removed from the transmission queue (e.g. MAC queue) \textsuperscript{6}, yet such action may adversely affect the route discovery needs of neighboring nodes. Changing any transmission characteristics necessitates an impact assessment of overhead, efficiency, and safety.

RQ3 Can we modify or develop analytical models that measure quantitatively the safety and routing efficiency when beacon data transmission parameters vary?

A model allows us to evaluate the impact to the safety and routing efficiency through sensitivity studies of parameters such as the beacon message size and transmission characteristics. For example, it is important to measure how additional data

\textsuperscript{2} U.S. Department of Transportation, Connected Vehicle Research, Online: \url{http://www.its.dot.gov/connected_vehicle/connected_vehicle.htm}, Accessed: [October 15, 2013]

\textsuperscript{3} Merriam-Webster Online Dictionary, beacon, Online: \url{http://www.merriam-webster.com/dictionary/beacon}, Accessed: [November 11, 2013]

requirements for routing beacons effects safety applications. In VANETs, the objective should be to maximize driver awareness of surroundings, so key metrics should include beacon delivery ratio, expired BSMs, number of consecutive messages lost between pairs of vehicles, reliability, overhead, speed of data dissemination, and reception probability.

Using a decomposition of safety beacon and beaconcasting requirements into common contextually-related data elements, we present a model that relates beacon PPR in terms of several parameters including beacon size, traffic density, communication range, congestion window size, and bit error rate (BER). We parameterize the model and perform numerical evaluations that classify beaconing solution boundaries.

We extend the works of others by decomposing beacon messages into contextually grouped elements so as to maximize their reception probability in a VANET. Our contributions include:

- An analytical model that relates beaconing PPR to beaconing transmission characteristics such as data rate, number of nodes, beacon size, communications range, and congestion window size.
- Evidence that suggests how minimizing beacon size increases reception probability.

This paper is organized as follows: Section II provides background information and discusses related work. Section III details the methodology and Section IV discusses results. Section V discusses threats to validity. Section VI concludes the paper and discusses future work.

II. BACKGROUND AND RELATED WORK

A. VANET Communications

A vehicular ad hoc network (VANET) forms spontaneously with an uncoordinated association of a set of vehicles that each change position dynamically. Organization of nodes within a VANET does not require external fixed infrastructure, but it does not preclude it, either. Regardless, specific network topology patterns affect the performance and feasibility of vehicular applications. Devices communicate through wireless links with the desire to attain the shortest communication time while utilizing low system resources. VANETs typically employ a multi-channel concept for the exchange of both safety and infotainment messages. Single hop transmissions are usually short-range – typically 300m or less for safety messages – and, thus, the term Dedicated Short-Range Communications (DSRC) is often applied when discussing VANETs. Broader information dissemination over two or more transmissions (i.e. multihop) greatly enhances application effectiveness [9].

Sporadic connectivity often characterizes a sparse VANET. Recent work [10] suggests using a store-carry-and-forward approach in which essential data packets are forwarded within each local group of vehicles, stored within the fringe vehicles which carry them until they come within range of other local vehicular groups, at which time the stored packets are forwarded once connectivity establishes. This paradigm describes the so-called Vehicular Delay-Tolerant Network (VDTN).

Vehicular communication architectures include native support for IPv6 in their protocol stacks, as defined by international standardization bodies including: IEEE 1609 Wireless Access in Vehicular Environments (WAVE), ISO TC 204 (CALM), the Car2Car Communication Consortium (C2C-CC) and ETSI TC ITS. DSRC standards include: an application message set, SAE J2735 [11]; network-layer and security services, IEEE standard 1609 [12], [13], [14], and [15]; and supporting modifications to the MAC and PHY layers, IEEE standard 802.11p [16]. These standards describe networking capabilities, such as: channel allocation and multi-channel access; priority queueing and channel routing; congestion control; security and privacy mechanisms; and application message data elements. Figure 1 shows the communication reference model for the IEEE 1609/WAVE-based VANET (i.e. DSRC).

<table>
<thead>
<tr>
<th>Application</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DSRC Message Set Dictionary SAE J2735</strong></td>
<td></td>
</tr>
<tr>
<td>Transport</td>
<td></td>
</tr>
<tr>
<td>Internet</td>
<td></td>
</tr>
<tr>
<td>Data Link</td>
<td></td>
</tr>
<tr>
<td>Physical</td>
<td></td>
</tr>
<tr>
<td>TCP</td>
<td>UDP</td>
</tr>
<tr>
<td>IPv6</td>
<td>LLC 802.2</td>
</tr>
<tr>
<td>1609.3</td>
<td>1609.4</td>
</tr>
<tr>
<td>Security 1609.2</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1 – DSRC communications reference model
IEEE Std 802.11p handles the MAC and PHY layers of the reference model and provides extensions intended to address the dynamic nature of potentially fast-moving vehicles. IEEE 802.11p allows a STA that is not a member of a Basic Service Set (BSS) to transmit data frames. Devices transmit such data frames “outside the context of a BSS,” or, more commonly, OCB. As such, this defines a new type of 802.11 communications. On top of the 802.11p MAC and PHY layers sits the IEEE 1609/WAVE stack. WAVE supports IP-based data transfers and also non-IP-based traffic through the WAVE Short Message Protocol (WSMP). IEEE 1609.3 specifies the WAVE Management Entity (WME) and corresponding network services, as well as WSMP. Channel coordination is a collection of enhancements to the IEEE 802.11 MAC, and interacts with the IEEE 802.2 LLC and IEEE 802.11 PHY; IEEE 1609.4 describes multi-channel operations. WAVE security services are specified in IEEE 1609.2. Atop the IEEE 1609 networking services lie the applications. While IEEE 1609 does not define application structure, another organization, SAE, has defined a DSRC message data dictionary which has been studied and implemented using IEEE 1609 below it.

The design of IEEE 1609/WAVE supports a single control channel (CCH) and multiple service channels (SCH), and it is generally assumed that the CCH will be mostly dedicated to safety applications. Channels operate in the 5.9GHz range and typically occupy 10MHz each, although the potential exists for 5MHz and/or 20MHz channels. The WAVE standard supports periodic channel switching, but does not preclude multiple channel-dedicated radios. Sync intervals split the channel at a rate of 10 Hz, which CCH and SCH intervals then split equally to support channel switching. Thus, a CCH interval is assumed to be $\frac{1}{2} \times 1/10 s = 50 \text{ ms}$. Vehicles operate autonomously to synchronize clocks, requiring a guard interval before accessing the channel. Typically, the guard interval is 4ms, leaving only 46ms per CCH channel access for transmission.

The WAVE standard addresses message priority using four different Access Classes, AC0 (lowest priority) to AC3 (highest priority), with the MAC layer maintaining separate queues and channel access for each AC. During packet transmission selection, the four ACs first contend internally with the winning packet then contending for the channel externally using its contention parameters. The WAVE contention mechanism is similar to the one used in conventional Wireless Local Area Network (WLAN) and the IEEE 802.11e Enhanced Distributed Channel Access (EDCA) Quality of Service (QoS) enhancements [17]. Each AC specifies a number of Arbitration Inter-Frame Spacing (AIFS) and Contention Window (CW) slots and each AC waits at least its AIFS slots, plus additional slots determined by the selected CW value. The number of slots times 16µs determines the wait time.

While the 802.11 family is well-understood, 802.11 unicasting is not well-suited to the VANET environment [7]. The IEEE standard 802.11p supports beacon transmission using the Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) broadcast method. Sensing of the network delays a transmission which would otherwise cause a collision, but CSMA/CA does not prevent all such collisions, especially in the hidden node scenario in which one vehicle cannot sense that a transmission would collide with another nearby vehicle. Transmitters employ a back-off scheme when collisions occur to re-attempt a transmission at a randomly selected future time controlled with the CW parameter.

### B. Data Transmission Requirements

Data transfers within a VANET result from application and networking requirements. Generally, applications describe data exchanges between vehicles while network routing requires local neighbor information for efficient transmission paths. Nodes broadcast beacons to nearby neighbors in conditions where the supplied load per node is small but the number of nodes can become large. The message properties (e.g. size, frequency, and lifetime) of regularly broadcast beacons directly impact MAC protocol performance. Some applications require transmission beyond the immediate communications range, thus requiring multi-hop communications. It is when multi-hop is required that networking layers work to discover and maintain routing and forwarding tables.

The data transmission needs of applications and routing serve dual purposes but contend for the same network resources. Furthering this apparent conflict, the term beaconing is often used in the literature in both application and networking contexts. In 802.11a wireless communications, beaconing typically refers to an access point broadcasting a beacon frame to announce its presence. However, the 802.11p VANET can communicate in OCB mode without an associated BSS, and so this definition is not applicable to our discourse. In VANETs, beaconing more typically has meaning in either an application context or in a routing context. In an application context, every vehicle transmits with regularity a beacon to alert other vehicles, avoid collisions and promote safety. In the networking context, the term beacon refers to routing / forwarding protocols using local knowledge gained from small presence messages, called beacons, to proactively or reactively maintain routing availability.

### C. Safety Beaconing

SAE standard J2735 describes an application data message set. Among these, the Basic Safety Message (BSM) represents the “here I am” message and announces vehicular information regularly. Two parts categorize the data elements of the BSM, as follows [8]: 1) BSM Part 1 contains core data elements, including vehicle position, heading, speed, acceleration, steering wheel angle, and vehicle size, which are transmitted at an adjustable rate of about 10 times per second; and 2) BSM Part 2 contains a variable set of data elements which are selected based on event triggers, e.g., ABS activated. BSM Part 2 data elements are added to Part 1 and sent as part of the BSM message, but are transmitted less frequently. In high vehicular density scenarios (e.g. urban intersections, traffic jams), safety beacon traffic can approach or even exceed channel capacity [6].

The information contained in an offered safety beacon supersedes prior information. Thus, prior, unsent beacons should be removed from the transmission queue [6]. Because beacons with current information are generated frequently, little time is spent...
Beacon-based routing protocols can be stateful, in which case a routing table is maintained, or stateless, in which case the routing protocol transmits beacon messages to exchange neighbor node information. However, aimlessly retransmitting broadcast packets obligates all nodes to rebroadcast newly-received messages. We define beaconcasting to describe when a broadcast routing protocol transmits beacon messages to exchange neighbor node information. However, aimlessly retransmitting broadcast packets can lead to broadcast storm in which collision and resulting congestion from traffic proliferation cripples performance [3]. Inefficient beaconcasting can tax limited network resources and reduce reliability. Typical metrics that evaluate broadcast routing performance in a VANET scenario include: reliability, or maximizing broadcast message delivery; overhead, or minimizing the number of redundant message transmissions; and speed of dissemination, or minimizing the delay incurred in delivering broadcast messages to appropriate vehicles in the VANET [4].

Broadcast protocols customarily contain 1) a preferred node selection algorithm [3] [4], 2) a waiting timeout calculation [4], and 3) an overhead messaging system [3] [4]. The node selection algorithm uses some metric (e.g. 1-hop distance) to discriminate from available nodes to select the ones to rebroadcast messages. When the right vehicles are selected, overhead is reduced and protocol efficiency increases. The waiting timeout calculation greatly affects the balance between reliability and delay. The requirements of the node selection algorithm and timeout calculation determine the frequency and content of overhead messaging, which communicates topological information between nodes using a beaconing mechanism.

Numerous broadcast routing protocols have been proposed, many of which with VANET in mind. Self-pruning protocols decide autonomously to rebroadcast a message by testing if all neighbors are already covered by other rebroadcasting nodes [3]. These protocols typically include their neighbor lists in header messages, which others use to compare to their own neighbors. Dominant-pruning topological algorithms, such as Multi-Point Relay (MPR), use a greedy approach to select a subset of its neighbors to rebroadcast messages [3]. The following sources contain excellent overviews of VANET routing protocols: [4], [19], and [6].

Varying mechanisms differentiate among broadcast routing protocols. As already mentioned, these include preferred node selection algorithm and waiting timeout, but also include the beaconing interval and beacon message content. For example, the waiting timeout may depend on distance from the precursor node, number of neighbors, or received signal strength indicator (RSSI) level. When nodes are mobile, care must be taken to select a beacon interval so that beacons do not expire over the course of the interval. The beaconing interval may be fixed, or may be adaptive based on network load or number of neighbors. Routing beacons typically contain current location and neighbor lists. In dynamically mobile VANET, it may be beneficial to include velocity and location history (or intended future position) to assist with proactive hand-off of forwarding neighbor selection. Beacon-based routing protocols can be stateful, in which case a routing table is maintained, or stateless, in which case the routing table is not maintained. Additionally, broadcast beacons may piggyback message ACKs.

E. Beaconless Routing

Beaconless routing protocols differ from broadcast routing protocols primarily in that they do not utilize an overhead messaging system to learn neighbor information and subsequently perform node selection efficiently; they instead extract required information from network packets that are successfully received. While the beaconless routing approach thus reduces communications overhead, it is constrained to the information content existing in the network. Beaconless algorithms may have only limited receiver information from which to rely on, such as: transmission range, distance from transmitter, and transmitter speed [5]. The receiver may also have available a source (i.e. on-board equipment) from which it can derive its own speed, i.e. the receiver speed. Together, this information can be used to calculate performance characteristics such as first transmission attempt delay and forwarding probability [5]. Furthermore, the transmitter may have control over transmission power adjustments, which can be leveraged to affect its own data transmission performance.
A. Research Goals

Both network routing requirements and application data requirements, especially safety applications, expect recurring data transmissions. A beacon, therefore, represents a small packet transmitted with some regularity. We attempt to clarify further the term beaconing by adopting the terms safety beacon to mean the BSM and beaconcasting to mean the transmissions involved in route discovery functionality. Clarifying beaconing motivates the following research objective: The goal of this research is to improve VANET application and routing design by precisely defining safety and routing beacon data so that transmission characteristics of their shared elements can maximize both safety and routing efficiency. Our investigations are guided by the following research questions:

RQ1 Can safety beacon and routing beacon data be categorized and described in a way that clarifies both functional and contextual meaning to their common elements?

A decomposition of beacon data beyond simple terms into exact data elements explains their use and allows analysis of their commonalities. For example, we know that the BSM safety beacon broadcasts a vehicle’s location, and we know that stateful VANET routing protocols desire to make local routing decisions using information from a neighbor information table. After routing protocols fail to find an existing route, they begin a neighbor discovery process that adds overhead in the transmission of routing beacons. However, a routing algorithm reduces beaconcasting overhead if it queries directly a neighbor information table populated from safety beacon data. Various routing protocols prefer different elements to exist in the neighbor information table, yet may not be guaranteed that BSM safety beacons provide the necessary information. Safety and routing processes thus require a functional coordination of data elements to achieve optimality. Such coordination requires a common and consistent vernacular.

RQ2 Can we relate how modifications to beacon data parameters impact transmission characteristics?

Adjustment of BSM parameters such as generation rate, transmission power, and contention window size impacts network characteristics, including reception probability [4]. For safety and routing, an understanding of the relationship between parameters and the resulting efficiency is vital. For example, a routing protocol may need certain data elements to efficiently make routing decisions that the BSM may not already be transmitting. Even if a routing algorithm could request additional routing beacon data to be included in safety beacons, the impact of such a change needs to be measured. Similarly, outdated or expired BSMs should be removed from the transmission queue (e.g., MAC queue) [6], yet such action may adversely affect the route discovery needs of neighboring nodes. Changing any transmission characteristics necessitates an impact assessment of overhead, efficiency, and safety.

RQ3 Can we modify or develop analytical models that measure quantitatively the safety and routing efficiency when beacon data transmission parameters vary?

A model allows us to evaluate the impact to the safety and routing efficiency through sensitivity studies of parameters such as the beacon message size and transmission characteristics. For example, it is important to measure how additional data requirements for routing beacons affect safety applications. In VANETs, the objective should be to maximize driver awareness of surroundings, so key metrics should include beacon delivery ratio, expired BSMs, number of consecutive message lost between pairs of vehicles, reliability, overhead, speed of data dissemination, and reception probability.

B. Beaconing Data

In a VANET, vehicles regularly transmit beacons, whether required for safety or beaconcasting. We define a beacon to be a message composed of one or more primitive data elements. SAE standard J2735 defines a DSRC message set using a hierarchical information design paradigm as follows: Data Elements compose Data Frames which in turn compose Messages. We follow this hierarchy, and focus on the data element as the lowest level datagram common to all beacons. While many message types explicitly define the mandatory data elements they contain, SAE standard J2735 also allows message definition using a la carte composition of individual data frames. By considering the context of individual beacon message data elements, we put forth the concept of contextual beaconing in which the data elements for a beacon message may be assembled and transmitted based on their common context. For example, elements in the context of basic safety may be transmitted with a certain frequency, whereas a beacon message with different contexts such as vehicle event or position history may require a different transmission frequency. Each data element possesses attributes such as:

- **Context** - The situation or events which most directly influence its meaning or effect. For example, basic safety elements such as position are contextually different from advisory information, such as a vehicle’s positional history.
- **Size** - The amount of data, usually expressed in bytes, including additional packet preamble overhead.
- **Frequency** - How often the data elements must be transmitted, usually expressed in terms of a beacon interval. For example, Basic Safety Messages are normally sent 10 times per second, but more often in hard-braking conditions, whereas routing neighboring discovery beacons may occur less frequently, or even on demand.
• **Priority** - A classification that differentiates the message type and contents. For example, the information of an imminent crash, may have higher priority than under non-emergency operations, even though the content may be identical.

• **Range** - The coverage area over which the contents are applicable. This could be limited to the range of the transmitter, or extended further (e.g. to a target geographic area further), requiring a multihop capability.

• **Lifetime** - Lifetime represents how long a beacon may live before expiring. Safety beacons typically live only as long as one time interval, after which a new beacon effectively replaces the expiring one. Distance, coverage area, number of hops, and/or length of time often express beacon lifetimes.

• **Retransmission strategy** - When multiple vehicles transmit beacons nearly simultaneously, collisions occur, initiating a back-off before subsequent re-transmission attempts. The back-off time is typically parameterized by a congestion window (CW). Beacons which are about to expire may not need to attempt retransmission.

• **Delay tolerance** - In the case of sparsely populated and disconnected VANETs, some beacons may tolerate delay in which nodes on the periphery of a local group may store and carry a message, forwarding it as soon as coming within range of another group. This concept is different from lifetime as lifetime generally assumes a connected network which does not tolerate delay. As an example, relaying beacons to the edge of a local group of vehicles may still hold some benefit when the local group comes within transmission range of another local group and the beacons are then forwarded (as opposed to waiting for new beacons to disseminate through the entire, recently connected network).

SAE standard J2735 decomposes the Basic Safety Message (BSM) into two parts consisting of Part I (required) and Part II (optional) data frames. The mandatory Part I elements contain standard vehicle elements, such as location, speed, heading, acceleration, braking status, and vehicle dimensions. The optional Part II contains the data elements necessary to define the vehicle’s past and/or predicted path history; support the exchange of GPS information for relative inter-vehicle positioning; and event trigger information that supports applications such as Emergency Electronic Brake Lights (EEBL) and Control Loss Warning (CLW).

Traffic density, message size, frequency, and priority most greatly affect delivery probability. All vehicles within communications range of one another share a limited bandwidth. In high vehicular density scenarios (e.g. urban intersections, traffic jams), safety beacon traffic can approach or even exceed channel capacity [6]. Thus, transmission characteristics require balance in order to achieve optimal efficiency. SAE standard J2735 provides guidance on differentiation of application priority. Table 1 shows a representative example for safety of life messages (i.e. the most important safety application class). IEEE standard 802.11p defines a transmission priority scheme that relates user priorities to access categories in the range of 0 to 3, with 3 being highest [16]. Table 2 shows the suggested mapping of transmission priorities to access categories. Regardless, SAE standard J2735 cautions that, despite such priority guidelines, urgent safety messages (i.e. all BSMs) will likely always be transmitted using the highest priority Access Category.

<table>
<thead>
<tr>
<th>J2735 Message Set</th>
<th>Default Message Priority</th>
<th>Latency for Reception</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crash-Pending Notification</td>
<td>7</td>
<td>&lt; 10 ms</td>
</tr>
<tr>
<td>Pre-Crash</td>
<td>7</td>
<td>&lt; 10 ms</td>
</tr>
<tr>
<td>Basic Safety + Hard-Brake (e.g. Collision Warning, Emergency Electronic Brake Lights, Anti-Lock, etc.)</td>
<td>7</td>
<td>&lt; 10 ms</td>
</tr>
<tr>
<td>Basic Safety Message</td>
<td>5</td>
<td>10 to 20 ms</td>
</tr>
<tr>
<td>Emergency Vehicle Alert</td>
<td>5</td>
<td>10 to 20 ms</td>
</tr>
<tr>
<td>ATIS Roadside Alerts (e.g. Accident)</td>
<td>5</td>
<td>10 to 20 ms</td>
</tr>
<tr>
<td>ATIS Probable-situation (e.g. Rapidly deteriorating dangerous conditions)</td>
<td>3</td>
<td>&gt; 20 ms</td>
</tr>
</tbody>
</table>

Table 1 - Representative application priorities (e.g. Safety of Life)

<table>
<thead>
<tr>
<th>SAE J2735 Priority</th>
<th>IEEE 802.11 Access Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Highest AC3</td>
</tr>
<tr>
<td>6</td>
<td>AC2</td>
</tr>
<tr>
<td>5</td>
<td>AC1</td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Lowest AC0</td>
</tr>
</tbody>
</table>

Table 2 - Priority and Access Category

Differing message priority intends to alleviate the chance of collisions from overlapping transmissions. Since there is no centralized coordination and control within an IEEE 1609 / WAVE VANET, different message priorities alone cannot prevent such collisions. In IEEE standard 802.11p, beacons are transmitted using the CSMA/CA broadcast method and transmitters employ a back-off scheme when collisions occur to re-attempt a transmission at a randomly selected future time controlled with a CW parameter. Previous work [20] [6] shows the importance of CW specification and the effects on successful delivery.
Aside from message priority, smaller and/or less frequent messages also reduce bandwidth burdens. Therefore, in contextual beaconing, we aim to decompose beaconing message data elements so that they are grouped into the minimal messages with common requirements so as to maximize their delivery probability.

C. Model

In an IEEE 1609 / WAVE-based VANET, vehicles contend on recurring intervals to broadcast unacknowledged safety beacons and other messages. Packet delivery occurs, with some probability, $P_r$, when collision-free successful receipt of all bits occurs before the current interval ends. Conceptually, WAVE employs a Pure ALOHA within a Slotted ALOHA approach. Slotted ALOHA delays fixed-length packet transmission until a known slot start time, while Pure ALOHA permits stations to transmit at will. WAVE uses recurring intervals, similar to slots in Slotted ALOHA. However, within each interval, all vehicles then freely contend for access and multiple transmissions occur at will during each interval. WAVE supports variable packet sizes and the context of required data elements can determine beacon message sizes. The 802.11p MAC layer intends to avoid collisions using CSMA/CA with channel sensing before sending and priority-based exponential backoff after collisions [21]. The maximum throughput for Pure ALOHA is 18% and for Slotted ALOHA it is 37% (with 26% collisions) [21], neither of which is nearly sufficient for VANET safety applications [2]. WAVE attempts improvements through several mechanisms, including: rate adaptation, power adaptation, and EDCA techniques that allow priority-based CW and AIFS parameters. However, modulation is less robust for higher data rates reducing frame reception likelihood [21]. While increasing transmission power may improve chance of delivery, it also increases communications range and thus collision probability to nearby nodes.

Several previous studies model the relationships between VANET data elements and the probability of their successful delivery. The authors of [22] present VANET beaconing in terms of channel utilization and reception probability. Packet loss that is acceptable for WLAN applications remains too high for safety-oriented applications [23]. The authors of [2] relate application effectiveness with driver situational awareness, and conclude that a PPR of at least 95.9% satisfies safety scenarios involving an awareness range of 100m. The authors of [17] show that collision probability relates to CW size and becomes high as soon as only a few nodes all using the highest-priority default AC parameters begin transmitting. The authors of [20] demonstrate the relationship between CW and the number of contending stations in saturated unicast traffic and prove the optimal value of $CW = \bar{n}\sqrt{2T}$, where $T$ is the total packet transmission time (including headers, SIFS, AIFS, etc.) and $\bar{n}$ is the average number of contending stations, but they do not take into account characteristics specific to VANET beacons. The authors of [6] account for beacon expiration and collision probabilities in high traffic density scenarios (e.g. highway traffic jams and urban intersections), but assert that collisions remain constant until CW increases sufficiently to cause beacons to expire.

While these models approach beacon PPR by fixing the beacon size, they do not preclude a use which exploits beacon variety. Thus, we combine the concepts of these analytical models [17] [6] [22] [20] and extend them through variation of beacon message size and frequency. We place ourselves in the context of a vehicle along some assumed roadway consisting of $i$ lanes. Each lane, $l_i$, has some density of vehicles $\rho_i$ km of $\rho$. Thus, the total vehicle density we perceive, $\rho$ is given by Equation 1:

$$\rho = \sum \rho_i$$

All vehicles regularly try to transmit beacons, and collisions may occur when other nodes transmit within either the Carrier Sense (CS) range, which represents the range up to which other nodes may sense the medium is busy (i.e., they sense an ongoing transmission), and also the Interference Range (IR), which includes the range up to which transmissions from hidden nodes will interfere with ongoing transmissions, (i.e., acts as noise). Standard modeling values for CS and IR remains an open issue in the literature, and choices abound. For example, safe driver situational awareness requires at least a 100m communications range [2]. While IEEE 802.11p allows for communication range up to 1000m, modeling gives a maximum attainable communication range of 386m with transmit power of 23dB [24]. We parameterize the CS and IR ranges, and nominally choose $c_s = 250m$ and $i_r = 350m$. The average number of transmitting neighbors within CS range, $N_{CS}$, the average number of additional nodes inside the IR, $N_{IR}$, and the total nodes, $N$, are given by Equations 2 - 4:

$$N_{CS} = 2\rho c_s$$

$$N_{IR} = 2\rho (i_r - c_s)$$

$$N = N_{CS} + N_{IR}$$
Equation 5 expresses the time to transmit a beacon of size $S$ in bits, at a data rate, $R$, in b/s, including $T_h$ which summarizes the time interval between frames (AIFS), the time to transmit the PHY layer convergence procedure (PLCP) preamble and header, and the contention window (CW):

$$T_s = T_h + \frac{S}{R}$$

(5)

The beacon size, $S$, includes an IEEE 1609 WAVE Short Message Protocol (WSMP) header, $S_W$, an IEEE 802.11p header $S_p$, and a variable length beacon payload, $S_b$ with contents that depend upon beaconing requirements.

$$S = S_W + S_p + S_b$$

(6)

Data elements further decompose a SAE J2735 BSM beacon payload into: DSRC Msg ID ($b_D$), the mandatory Part I BSM elements ($b_1$) and the optional Part II BSM element ($b_2$). By definition, $b_D$ is 1 byte (8 bits) and $b_1 = 38 \times 8 = 304$ bits. When only Part I BSMs are send, $b_2 = 0$.

$$S_b = b_D + b_1 + b_2$$

(7)

A common strategy to increase the chance of undamaged frame delivery is to shorten frame length [21] thus reducing the number of potential individual bit errors. BER defines the number of unsuccessfully received bits as a percent of the total bits attempted. BER for IEEE 802.11p is difficult to quantify; simulation gives IEEE 802.11p BER around $10^{-2}$ for SNR in the range of 20-25 dB [25]. Equation 8 gives the probability of being received entirely correctly, $P_{bf}$, for $S$ bits and a bit error, $B$. For example, for $B=10^{-4}$, the probability of receiving a bit-error-free beacon of 428 bytes is 71%, but the probability increases to 91% if the beacon frame size is reduced to 116 bytes, the minimum size of a Part I BSM with headers.

$$P_{bf} = (1 - B)^S$$

(8)

Models differ in how they model collision probabilities, especially the effects of CW size [22] [17] [20] [6]. We adopt the Eichler model [17], which states that the probabilities of collision, $P_c$, and collision-free, $P_f$, transmissions for N nodes using congestion window size of CW slots is given by:

$$P_f = \frac{N}{CW^N} \sum_{i=1}^{CW-1} (CW - i)^{N-1}$$

(9)

$$P_c = 1 - P_f$$

(10)

We now turn our attention to success measurements of channel utilization and beacon PPR.

A beacon is successfully received if the packet does not collide with other senders’ transmission attempts and if all the bits are successfully received by the intended recipient. Equation 11 relates collision probability and BER to beacon reception probability, $P_s$:

$$P_s = (1 - P_c)P_{bf}$$

(11)

We note the reliance on CW and recall Bianchi’s results which show that CW can be maximized as $CW = N\sqrt{2T_s}$. This leads to an adaptive congestion windowing concept which can therefore be employed to minimize collisions as the number of nodes varies, and we allow this optimization in our model [20]. Further, it has been shown that at high traffic densities requiring large CW due to the dependence on $N$, beacons fail to be delivered within their allotted interval and thus expire [6]. While this introduces another term which prevents packet reception, the probability of expiration remains low until $N$ becomes large, and we thus explicitly exclude this term from our model.
Utilization is the amount of time that transmissions are successful as a percentage of the total time. Normalized channel load models channel busy time, which we give in Equation 12.

\[
\mu = N \lambda g T_S
\]

Clearly, channel utilization cannot exceed 1. Yet an upper bound on the optimal value remains elusive. In Slotted ALOHA, the maximum channel utilization is 37%, whereas CSMA/CA in IEEE 802.11 has typical maximum utilization of 54-66%, absent hidden terminals [7]. The beacon generation rate, \( \lambda_g \), determines the frequency that a node generates a beacon. When we assume \( N \) and \( T_d \) are fixed, then an upper bound on \( \lambda_g \) maximizes \( \mu \) when \( \mu = P_s \). Beacon generation rate modeling often assumes \( \lambda_g = 10 \text{ Hz} \). When higher beacon generation rates overload the channel, utilization suffers, and reducing \( \lambda_g \) addresses this situation. However, a beacon generation rate of less than 10 Hz jeopardizes safety. Thus, we assume that only non-safety beacon data element generation occurs at rates lower than 10 Hz.

D. Data Collection and Approach

To answer RQ1, we decompose all safety beacon and beaconcasting requirements into common contextually-related data elements, which we have described above. By treating beacon data in terms of their individual data elements instead of holistically as entire beacons, we separate out the mandatory elements (e.g. BSM Part I) from other elements (e.g. BSM Part II elements or additional non-safety data required by broadcast routing protocols). This decomposition allows beacon data elements to be individualized by context, which we can then model.

To answer RQ2 and RQ3, we present a model that relates beacon PPR in terms of several parameters including beacon size, traffic density, communication range, congestion window size, and BER. We parameterize the model to understand the impact of parameter sensitivity, run several numerical simulation scenarios, and collect the results, which numerical evaluations classify. We examine the impact of each parameter independently (by fixing the others). Using beacon data element decomposition, we can examine more precisely the impact of beacon size on reception probability. A numerical procedure projects beaconing solution boundaries for which we depict the results graphically.

Beaconing solutions relate the following primary parameters:

- **Beacon transmission time (\( T_s \))** – The beacon transmission time depends on the data rate and influences the optimal congestion window size (i.e. CW) when adaptive windowing is employed. Thus, we model varying rates with and without adaptive windowing, to examine the impact of beacon transmission time on PPR.

- **Beacon size (\( S \))** – The beacon size directly impacts the reception probability. We set a minimum beacon size and vary the size and frequency of additional beacon data elements to assess the impact of beacon size on PPR.

- **Data rate (\( R \))** – The data rate also directly influences reception probability, although the modulation of higher data rates often reduces PPR.

- **Number of nodes (\( N \))** – The number of nodes depends on traffic density and communication range, and influences greatly the reception probability. Furthermore, number of nodes influences optimal congestion window size. We model the impact of node number by varying density and communication range while fixing other parameters.

- **Congestion Window size (\( CW \))** – The congestion window size effects collision probability. When adaptive congestion windowing is allowed, CW increases with the number of nodes and/or time to transmit a beacon. We model the effects on PPR with and without adaptive congestion windowing.

- **Bit error rate (BER)** – The bit error rate effects PPR, but is hard to quantify in an IEEE 802.11p VANET. We model beacon reception probability using an assumed BER of 10^{-4}.

IV. STUDY RESULTS

We define four study scenarios to give results, and Table 3 describes parametric values. Several values remain constant across all modeled scenarios and we focus primarily on assessing the impact of varying data rate, number of nodes, beacon size, and communications range.
A description of each scenario and analysis of results follows:

**A. Data Rate**

Increasing data rate proportionally reduces the time to send a message, thus improving throughput. For example, when the data rate is doubled, the time to transmit is cut in half. If, however, the contention window slot time is also cut in half, and the CW parameter is unchanged, then collision probability remains unchanged because, while packets will be transmitted in less time, the channel contention will also be reduced proportionally. Therefore, we model a scenario with increasing data transmission rate from 3 to 54 Mbps, assuming all attempts use Access Category 3 (CW=0-3) and we hold the contention window time slot constant as transmission time decreases. Figure 2 shows reception probability of 25% at an initial data rate of 3 Mbps, increasing as the data rate increases. Furthermore, we also chart the adaptive CW mechanism in which the optimal value of $CW = \frac{n}{\sqrt{T}}$ is used and this shows marked improvement over the fixed default AC3 values, achieving a PPR of 73%. Thus, our results support those of [20] [6] and demonstrate that increasing data rate improves reception probability which an adaptive CW maximizes.

**B. Number of Nodes**

As the number of transmitting nodes increases collision probabilities increase, thus degrading reception likelihood. We vary the vehicular density from 4 to 50 vehicles / km. The results charted in Figure 3 show that for CW values assigned to the AC3 defaults, the PPR is only 55% at low vehicular density and degrades rapidly, dropping below 10% with a density of 18 vehicles / km. However, by again utilizing adaptive CW, we see that optimal CW selection maximizes PPR at 71%.

---

**Table 3 – Model scenarios and their parameters**

<table>
<thead>
<tr>
<th>Term</th>
<th>Model Variable</th>
<th>Data Rate</th>
<th>Number of Nodes</th>
<th>Beacon Size</th>
<th>Comm. Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lanes of traffic</td>
<td>$i$</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Traffic Density</td>
<td>$\rho$</td>
<td>10</td>
<td>10</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>Carrier Sense (CR) range</td>
<td>$c$</td>
<td>250 m</td>
<td>250 m</td>
<td>250 m</td>
<td>75-875</td>
</tr>
<tr>
<td>Interference Range (IR)</td>
<td>$\lambda$</td>
<td>350 m</td>
<td>350 m</td>
<td>350 m</td>
<td>1.5 x CR</td>
</tr>
<tr>
<td>PLCP Preamble time</td>
<td>$-T_h$</td>
<td>32 us</td>
<td>32 us</td>
<td>32 us</td>
<td>32 us</td>
</tr>
<tr>
<td>PLCP header time</td>
<td>$-T_h$</td>
<td>8 us</td>
<td>8 us</td>
<td>8 us</td>
<td>8 us</td>
</tr>
<tr>
<td>DIFS</td>
<td>$-T_h$</td>
<td>64 us</td>
<td>64 us</td>
<td>64 us</td>
<td>64 us</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>$-T_h$</td>
<td>1 us</td>
<td>1 us</td>
<td>1 us</td>
<td>1 us</td>
</tr>
<tr>
<td>Beacon frequency</td>
<td>$\lambda$</td>
<td>10 Hz</td>
<td>10 Hz</td>
<td>10 Hz</td>
<td>10 Hz</td>
</tr>
<tr>
<td>Data Rate</td>
<td>$R$</td>
<td>3-54 Mbps</td>
<td>3 Mbps</td>
<td>3 Mbps</td>
<td>3 Mbps</td>
</tr>
<tr>
<td>WSMP header</td>
<td>$S_w$</td>
<td>64 bytes</td>
<td>64 bytes</td>
<td>64 bytes</td>
<td>64 bytes</td>
</tr>
<tr>
<td>802.11p header</td>
<td>$S_p$</td>
<td>26 bytes</td>
<td>26 bytes</td>
<td>26 bytes</td>
<td>26 bytes</td>
</tr>
<tr>
<td>Beacon DSRC Msg ID</td>
<td>$b_D$</td>
<td>1 byte</td>
<td>1 byte</td>
<td>1 byte</td>
<td>1 byte</td>
</tr>
<tr>
<td>BSM Part I</td>
<td>$b_I$</td>
<td>38 bytes</td>
<td>38 bytes</td>
<td>38 bytes</td>
<td>38 bytes</td>
</tr>
<tr>
<td>BSM Part II / other beacon data</td>
<td>$b_z$</td>
<td>250</td>
<td>250</td>
<td>0-500</td>
<td>0, 250, 500</td>
</tr>
<tr>
<td>Congestion window size</td>
<td>$CW$</td>
<td>0-3, adaptive</td>
<td>0-3, adaptive</td>
<td>0-3, adaptive</td>
<td>0-3, adaptive</td>
</tr>
<tr>
<td>Bit error rate</td>
<td>$BER$</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

**Figure 2 – Beaconing data rates and PPR**

**Figure 3 – Vehicle density and beacon PPR**
C. Beacon Size

Beacon message size is usually assumed fixed and often ignored in the literature, yet control of beacon size has significant impact on reception probability. Recall that safety beacons consist of mandatory and optional elements. Beacon size has a nearly proportional relationship with time to transmit the message, as expressed in Equation 5. Thus, we expect to observe that an increasing beacon size will nearly linearly decrease reception probability. Figure 4 shows this to be the case. When default AC3 parameters are used and optional beacon data elements are varied from 0 to 500 bytes, we see reception probability at 25% when only the mandatory safety beacon elements are sent, dropping to 17% when the optional beacon element size increases to 500 bytes. Here, the large range of optional beacon data seems to have little impact on reception probability. However, when we further include adaptive CW, the impact becomes significant, as reception probability rises for a 0-byte optional beaconing element from 25% with default AC3 parameters to 84% when adaptive CW is used. The reason for this is a combination of minimized collision probability along with the improved error-free receipt of the smaller number of bits transmitted. Minimizing safety beacon messages to only required elements thus improves the odds of their reception. Required safety beacon elements (i.e. $b_1=39$, $b_2=0$ per Equation 7) require a sending rate of 10 Hz and the transmission rate of any additional data elements depends solely on application and networking requirements. For example, a beaconing policy may send minimal safety beacons (i.e. $b_2=0$) at a rate of 10 Hz, while the additionally requested data elements (i.e. $b_2>0$) transmit at a rate of, say, 2 Hz. Regularly sending additional non-required data degrades overall reception quality, thus jeopardizing the intended safety requirements.

D. Communications Range

A larger communications range potentially increases the number of nodes contenting for the channel, which, as we have already seen, may significantly degrade the chance of successful packet delivery. Broadly speaking, communications range is related to transmitting power output, and a smaller communications range implies lower power. We therefore vary CR from 75m to 875m and set IR to an additional 50% of the CR value. The results of Figure 5 show that for communications ranges of 375m or more, reception probability is similar regardless of beacon packet size. However, for the case where a communications range is short (i.e. CR=75m), we observe marked improvements in reception probability as packet size decreases. For example, when 500 bytes of additional (i.e. $b_2=500$) beacon data is sent with communications range of 75m, the reception probability is 40%, but increases to 60% if the optional data is omitted (i.e. $b_2=0$). Improved error-free receipt of fewer transmitted bits primarily gives this result. Additionally, when adaptive CW is employed, the reception probability rises to 84%. Better reception likelihood within smaller communications range implies that there is benefit from a beaconing policy of reduced power (i.e. leading to reduced range) for higher priority data.

V. Threats to Validity

While we have identified limitations within our study, we believe that these limitations could not invalidate our results. First, the model has not been validated by simulation or in a real-world communications setting. To support our analytical model, however, we have leveraged the previous models of others for 802.11p transmission collision probability, (adaptive) contention windowing impacts, and bit error rate, while also extending them to also assess the impact of beacon size on reception probability. Furthermore, to validate our model, we have isolated beaconing characteristics (e.g. data rate, number of nodes, etc.) and confirmed that our models output aligns with previous works which assess the same. Second, it remains unclear if beacon delivery rates will support requisite safety efficiency. For example, our best modeled results were an 84% reception probability, which falls short of the 95.9% safety requirement in [2]. While safety and beacon reception requires continued study, we have focused on maximizing packet delivery probability, which is primary pathway to realizing safety requirements.
Thoughtful adjustment of transmission parameters based on beacon data characteristics improves their chance of successful receipt by neighboring vehicles, and this in turn alleviates network overload and more importantly improves safety. Our model shows that reducing beacon message sizes leads to marked improvements in reception probability, especially when combined with reduced communications range (i.e., implied lower power) and using adaptive congestion windowing. Typical modeling within the literature commonly assumes fixed beacon packets on the order of hundreds of bytes. Our work suggests an alternate beaconing policy which may be described as follows: To improve safety through maximal reception probability, frequently-transmitted safety beacons should be of minimal size and broadcast within distance-limited communication range, while additional safety and non-safety data elements should be scheduled with frequency, size, and intended range based upon their contextual beaconing requirements. Future work will explore this further through simulation and trade study analysis of various beaconing policies applied to several scenarios in which data size, transmission frequency, and communications range parameters may be modeled. For example, the concept of piggybacking the BSM message from others into one’s own transmission is often presented as a way to disseminate safety information through a VANET. However, the balance must be further explored between the tradeoff in minimizing the safety requirements of an individual vehicle, which may require small packets with short-range transmission, against the needs to include additional information (e.g., gathered from neighboring vehicles) within each beacon. Regardless, minimizing beacon size increases reception probability, leading to a more efficiently operating VANET. An improved VANET communications system improves safety, thus supporting one of the primary goals of connected vehicle systems.

REFERENCES


**APPENDIX I – SOURCE CODE**

Merely for completeness, we include here Java code which implements the analytical model of this paper.

```java
package VANET_BeaconingModel;

/**
 * The Class VANET_BeaconingModel.
 * Implementation of the analytical model from:
 * Balancing Safety and Routing Efficiency with VANET Beaconing Messages
 * Scott E. Carpenter
 * Department of Computer Science
 * North Carolina State University
 * Raleigh, NC, USA
 *scarpen@ncsu.edu
 */
public class VANET_BeaconingModel {

    /**
     * Instantiates a new vANET beaconing model.
     */
    public VANET_BeaconingModel() {
    }

    /** The scenario name. */
    private String strScenarioName = "";

    /** Sets the scenario name. */
    public void setScenarioName(String strScenarioName) {
        this.strScenarioName = strScenarioName;
        // output scenario name and header info to console
        System.out.println();
        System.out.println("Scenario: " + this.strScenarioName);
        printlnHeader();
    }

    /** The max lanes. */
    private int MAX_LANES = 4;

    /** The number of lanes. */
    private int nLanes = MAX_LANES;

    /** The densities_per_km. */
    // default = 12 vehicles / km (over 4 lanes)
    private double[] densities_per_km = {3.0, 3.0, 3.0, 3.0};

    /**
     * Sets the traffic lane densities.
     */
    public void setTrafficLaneDensities(double lane1, double lane2,
                                         double lane3, double lane4) {
        densities_per_km[0] = lane1;
        densities_per_km[1] = lane2;
        densities_per_km[2] = lane3;
        densities_per_km[3] = lane4;
    }

    /** The traffic density, rho. */
    private int rho = 0;
}
```
/**
 * Calculate traffic density.
 */
private void calculateTrafficDensity() {
    double totalDensity = 0;
    // sum all lanes
    for (int i = 0; i < nLanes; i++) {
        totalDensity += densities_per_km[i];
    }
    rho = (int) Math.floor(totalDensity);
}

/** The Carrier Sense range, in meters, cs_m. */
private int cs_m = 250;

/**
 * Sets the cs_m.
 * @param cs_m the new cs_m
 */
public void setCs_m(int cs_m) {
    this.cs_m = cs_m;
}

/** The Interference Range, in meters, ir_m. */
private int ir_m = 350;

/**
 * Sets the ir_m.
 * @param ir_m the new ir_m
 */
public void setIr_m(int ir_m) {
    this.ir_m = ir_m;
}

/** The number of vehicles within CS, Ncs. */
private int Ncs = 0;

/** The number of additional vehicles within IR, Nir. */
private int Nir = 0;

/** The total number of all vehicles, N. */
private int N = 0;

/** The number of meters per km. */
private double M_PER_KM = 1000.0;

/**
 * Calculate number of nodes.
 */
private void calculateNumberOfNodes() {
    Ncs = (int) Math.floor(2.0 * (double) rho * (double) cs_m / M_PER_KM);
    Nir = (int) Math.floor(2.0 * (double) rho
          * ((double) ir_m - (double) cs_m) / M_PER_KM);
    N = Ncs + Nir;
}

/** The data rate in Mbps, R_mbps. */
private double R_Mbps = 3.0;

/**
 * Sets the r_mbps.
 * @param r_Mbps the new r_mbps
 */
public void setR_Mbps(double r_Mbps) {
    R_Mbps = r_Mbps;
}
/** The beacon DSRC Msg ID size in bytes, bD. */
private int bD = 1;

/** The beacon mandatory BSM minimum size, in bytes, b1. */
private int b1 = 38;

/** The beacon optional data size, in bytes, b2. */
private int b2 = 350;

/** Sets the b2.
 * @param b2 the new b2 */
public void setB2(int b2) {
    this.b2 = b2;
}

/** The size in bytes of the WSMP header, SW. */
private int SW = 64;

/** The size in bytes of the 802.11p header, Sp. */
private int Sp = 26;

/** The beacon message size in bytes, Sb. */
private int Sb = bD + b1 + b2;

/** The total beacon size, including headers, in bytes, S. */
private int S = SW + Sp + Sb;

/** Setup beacon size. */
private void setupBeaconSize() {
    Sb = bD + b1 + b2;
    S = SW + Sp + Sb;
}

/** The PLCP preamble time in us. */
private int PLCP_preamble_time_us = 32;

/** The time to transmit the PLCP header, in us. */
private int PLCP_header_time_us = 8;

/** The DIFS in us. */
private int DIFS_us = 64;

/** The sigma in us. */
private int sigma_us = 1;

/** The time to transmit the beacon header in us, Th. */
private double Th = (PLCP_preamble_time_us + PLCP_header_time_us + DIFS_us + sigma_us) * Math.pow(10, -6);

/** The time to transmit the beacon data including headers in us, Ts. */
private double Ts = 0;

/** Calculating beacon transmission time. */
private void calculateBeaconTransmissionTime() {
    Th = (PLCP_preamble_time_us + PLCP_header_time_us + DIFS_us + sigma_us) * Math.pow(10, -6);
    int Sbits = S * 8;
    Ts = (double) Th + (double) Sbits / ((double) R_Mbps * Math.pow(10, 6));
}

/** The probability of error-free (i.e. successful) bit transmission, Pbf. */
private double Pbf = 0;
/** The bit error rate (BER), B. */
private double B = Math.pow(10, -4);
/** Calculate error free bit transmission. */
private void calculateErrorFreeBitTransmission() {
    int Sbits = S * 8;
    Pbf = Math.pow(1 - B, Sbits);
}
/** The number of contention window slots for AC3. */
public static int AC3_CW_slots = 4;
/** The contention window value in slots. */
private int CW = AC3_CW_slots;
/** Sets the CW size. */
/** @param cW the new CW */
public void setCW(int cW) {
    CW = cW;
}
/** The flag to use adaptive conegration windowing (or not). */
private boolean adaptiveCW = false;
/** Sets the adaptive cw. */
/** @param adaptiveCW the new adaptive cw */
public void setAdaptiveCW(boolean adaptiveCW) {
    this.adaptiveCW = adaptiveCW;
}
/** The slot time for one slot, in us. */
private int slot_time_us = 16;
/** Calculate the value for CW. */
private void calculateCW() {
    if (adaptiveCW == true) {
        int Tslots = (int) Math.floor(Ts / ((double) slot_time_us) * Math.pow(10, -6));
        int factor = (int) (R_Mbps / 3.0);
        int newCW = (int) Math.floor((double) N * Math.sqrt(2.0 * Tslots)) * factor;

        // do not go below default
        if (newCW > AC3_CW_slots) {
            CW = newCW;
        } else {
            CW = AC3_CW_slots;
        }
    } else {
    // CW = AC3_CW_slots;
    }
}
/** The probability of no collisions, Pf. */
private double Pf = 0;
/** The probability of a collision, Pc. */
private double Pc = 0;
/** Calculate collision probability. */
private void calculateCollisionProbability() {
    int nc = CW;
    double ntN = Math.pow(nc, N);
    double ssum = 0;
    for (int i = 1; i <= (nc - 1); i++) {
        double term = Math.pow((nc - i), N - 1);
        ssum += term;
    }
    Pf = (double) N / ntN * ssum;
    Pc = 1.0 - Pf;
}

/** The beacon reception probability = probability of packet reception (PPR), Ps. */
private double Ps = 0;

/** * Calculate beacon reception probability. */
private void calculateBeaconReceptionProbability() {
    Ps = Pf * Pbf;
}

/** The maximal beacon generation rate, lambda_g. */
private double lambda_g = 0;

/** * Calculate optimal beacon generation rate. */
private void calculateOptimalBeaconGenerationRate() {
    lambda_g = Ps / ((double) N * Ts / Math.pow(10, -6));
}

/** * Calculate all that is necessary for beacon reception probability. */
public void calculate() {
    calculateTrafficDensity();
    calculateNumberOfNodes();
    setupBeaconSize();
    calculateBeaconTransmissionTime();
    calculateErrorFreeBitTransmission();
    calculateCW();
    calculateCollisionProbability();
    calculateBeaconReceptionProbability();
    calculateOptimalBeaconGenerationRate();
}

/** * Prints the header. */
private void printHeader() {
    System.out.println("rho, cs_m, ir_m, Ncs, Nir, N, Sb, S, Mbps, Th, Ts, Pbf, CW, lambda, Pf, Pc, Ps");
}

/** * Report the results. */
public void report() {
    String strLine = String.format("%4d, %4d, %4d, %4d, %4d, %4d, %4d, %4d, %010.4f, %08.6f, %08.6f, %06.4f, %06.4f, %06.4f, %06.4f, %06.4f", rho, cs_m, ir_m, Ncs, Nir, N, Sb, S, R_Mbps, Th, Ts, Pbf, CW, lambda_g, Pf, Pc, Ps);
    System.out.println(strLine);
}
/**
 * The main method.
 * @param args the arguments
 */
public static void main(String[] args) {
    VANETBeaconingModel m = new VANETBeaconingModel();
    // defaults
    m.setTrafficLaneDensities(10, 0, 0, 0);
    m.setB2(250);
    // Scenario 1: Vary data rate (increasing)
    m.setScenarioName("Data Rate, CW=AC3");
    for (int R_Mbps = 3; R_Mbps <= 54; R_Mbps += 3) {
        int factor = R_Mbps / 3;
        m.setR_Mbps(R_Mbps);
        m.setCW(VANETBeaconingModel.AC3_CW_slots * factor);
        m.calculate();
        m.report();
    }
    m.setScenarioName("Data Rate, CW=adaptive");
    m.setAdaptiveCW(true);
    for (int R_Mbps = 3; R_Mbps <= 54; R_Mbps += 3) {
        m.setR_Mbps(R_Mbps);
        m.calculate();
        m.report();
    }
    // Scenario 2: Vary the number of nodes (i.e. increasing vehicular density)
    m.setR_Mbps(3);
    m.setScenarioName("Number of Nodes, CW=03 (AC3)");
    m.setAdaptiveCW(false);
    m.setCW(VANETBeaconingModel.AC3_CW_slots);
    for (int nodes = 4; nodes <= 50; nodes += 2) {
        m.setTrafficLaneDensities(nodes, 0, 0, 0);
        m.calculate();
        m.report();
    }
    // Scenario 3: Vary the beacon size
    m.setScenarioName("Beacon size, CW=03 (AC3)");
    m.setTrafficLaneDensities(3, 3, 3, 3);
    m.setR_Mbps(3);
    m.setAdaptiveCW(false);
    m.setCW(VANETBeaconingModel.AC3_CW_slots);
    for (int beaconsize = 0; beaconsize <= 500; beaconsize += 50) {
        m.setB2(beaconsize);
        m.calculate();
        m.report();
    }
    m.setScenarioName("Beacon size, CW=adaptive");
    m.setTrafficLaneDensities(3, 3, 3, 3);
    m.setR_Mbps(3);
    m.setAdaptiveCW(true);
    m.setCW(VANETBeaconingModel.AC3_CW_slots);
    for (int beaconsize = 0; beaconsize <= 500; beaconsize += 50) {
        m.setB2(beaconsize);
        m.calculate();
        m.report();
    }
    // Scenario 4: Vary the Communications Range and beacon size
    m.setScenarioName("Beacon size=0, cr=75-875, CW=0-3 (AC3)");
    m.setTrafficLaneDensities(4, 4, 3, 3);
    m.setR_Mbps(3);
    m.setAdaptiveCW(false);
    m.setB2(0);
m.setCW(VANETBeaconingModel.AC3_CW_slots);
for (int cs = 75; cs <= 875; cs += 100) {
    int ir = cs + cs / 2;
    m.setCs_m(cs);
    m.setIr_m(ir);
    m.calculate();
    m.report();
}

m.setScenarioName("Beacon size=250, cr=75-875, CW=0-3 (AC3)");
m.setTrafficLaneDensities(4, 4, 3, 3);
m.setR_Mbps(3);
m.setAdaptiveCW(false);
m.setB2(250);
m.setCW(VANETBeaconingModel.AC3_CW_slots);
for (int cs = 75; cs <= 875; cs += 100) {
    int ir = cs + cs / 2;
    m.setCs_m(cs);
    m.setIr_m(ir);
    m.calculate();
    m.report();
}

m.setScenarioName("Beacon size=500, cr=75-875, CW=0-3 (AC3)");
m.setTrafficLaneDensities(4, 4, 3, 3);
m.setR_Mbps(3);
m.setAdaptiveCW(false);
m.setB2(500);
m.setCW(VANETBeaconingModel.AC3_CW_slots);
for (int cs = 75; cs <= 875; cs += 100) {
    int ir = cs + cs / 2;
    m.setCs_m(cs);
    m.setIr_m(ir);
    m.calculate();
    m.report();
}

m.setScenarioName("Beacon size=0, cr=75-875, CW=adaptive");
m.setTrafficLaneDensities(4, 4, 3, 3);
m.setR_Mbps(3);
m.setAdaptiveCW(true);
m.setB2(0);
m.setCW(VANETBeaconingModel.AC3_CW_slots);
for (int cs = 75; cs <= 875; cs += 100) {
    int ir = cs + cs / 2;
    m.setCs_m(cs);
    m.setIr_m(ir);
    m.calculate();
    m.report();
}