Repetition-based Broadcast in Vehicular Ad Hoc Networks in Rician Channel with Capture

Farzad Farnoud, Shahrokh Valaee

Abstract—**In this paper we study the performance of different vehicular wireless broadcast schemes that rely on repetition as a means for providing reliable communications in Rician environment with capture effect. We investigate different patterns for retransmission and show that the one based on Optical Orthogonal Codes (OOC) performs better than others in terms of probability of success and delay. We propose using different numbers of repetitions for providing different Quality of Service (QoS) priority levels and show this method can effectively provide different QoS classes for different types of data without throughput loss. Probability of success and delay are obtained via simulation for three broadcast schemes in the presence of capture effect in Rician fading environment. Furthermore, analytical solutions are compared to simulation for transmission with no capture as a special case.**

I. Introduction

VEHICULAR ad hoc networks (VANETs) are becoming an increasingly important type of mobile ad hoc networks because of their application in cooperative safety systems. VANETs have the potential to provide communication with lower delay as compared to infrastructure-based communication, and are available in areas where dedicated infrastructure is not available. One of the major challenges in VANETs is designing the medium access control (MAC). The unlimited battery power of vehicles, high mobility, and harsh channel conditions produce an interference-limited environment. Safety packets are of broadcast type [1] [2] and should be transmitted by each vehicle to all other vehicles in its neighborhood. In 802.11, broadcast transmission does not use RTS/CTS signaling, hence hidden terminals can cause collision and data loss. Furthermore, in 802.11, acknowledgment is not required for broadcast communications, which results in less reliable communications. Using acknowledgements on the other hand causes significant overhead since the original data packets are usually small and there are many receivers.

We have recently used Optical Orthogonal Codes (OOC) [3] to create a repetition pattern which mitigates the interference caused by multiple repetitions [4]. Optical orthogonal codes produce a number of binary sequences with small cross-correlation. Using these codes as the repetition patterns guarantees that the maximum number of times that two wireless nodes simultaneously transmit is limited. Simulation results show that the proposed OOC technique outperforms the other repetition-based techniques.

This work was supported in part by an in-kind contribution from Mark IV Industries.

F. Farnoud and S. Valaee are with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto (email:{farnoud,valaee}@comm.utoronto.ca)

In the present paper, we extend our previous work to include the capture mode. The previous work assumes that all simultaneous transmissions will result in packet loss. Here, we assume that the wireless channel has a Rician fading, and a transmitted packet is successfully received if the signal-to-interference and noise ratio (SINR) at the receiver is above the detection threshold. We also provide lower bounds for the probability of successful transmissions, and propose a means for providing different priorities in the network.

The rest of this paper is organized as follows. Section II explains the basics of repetition-based broadcast and different methods studied in this paper. In Section III, we present the system model, analysis, and metrics used to evaluate the performance. Section IV presents the simulation results. Finally Section V presents the concluding remarks.

II. Repetition-based Broadcast

In repetition-based broadcast schemes, time is divided into frames. Each frame, in turn, is divided into L timeslots with length equal to the transmission time of a single packet. Each packet is transmitted a number of times inside the frame. In each timeslot, if the node is not transmitting, it moves to the receive mode. The transmission patterns of wireless nodes should be selected in a distributed fashion.

Each pattern can be shown with a binary vector of length L in which '1' denotes a transmission and '0' represents an idle timeslot. Each of these vectors is called a codeword and the set of all codewords is called a code.

Here, we study two random schemes proposed in [5] and a one based on optical orthogonal codes previously introduced in [4]:

a) Synchronous Fixed Retransmission (SFR): In SFR, each packet is transmitted w times in each frame, i.e., w timeslots are randomly chosen out of the L available timeslots for repeated transmissions of the packet.

b) Synchronous p-Persistent Retransmission (SPR): In SPR, the node transmits the packet in each timeslot in a frame with probability p and remains idle with probability $1-p$. Note that in this approach a packet may be transmitted L times or not transmitted at all. Therefore, it is expected that this scheme result in a less desirable performance.

c) Retransmission based on Synchronous Optical Orthogonal Codes (OOC): If transmission patterns are chosen more carefully, we will be able to mitigate the interference among users. Assuming vectors $\mathbf{x}$ and $\mathbf{y}$ are two codewords, their cross-correlation, represented by the inner product $\langle \mathbf{x}, \mathbf{y} \rangle$, is the number of collisions that occur if two users transmit with the pattern indicated by $\mathbf{x}$ and $\mathbf{y}$. Therefore, limiting

978-1-4244-2219-7/08/$25.00 (c)2008 IEEE

the correlation of two codewords is equivalent to limiting the possibility of collisions in transmission. A synchronous optical orthogonal code, C, is a code whose codewords satisfy the following condition:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^{L} x_i y_i \leq \lambda \qquad \forall \mathbf{x}, \mathbf{y} \in C \tag{1}$$

where λ is a fixed integer usually taken to be 1.

A Synchronous Optical Orthogonal Code, C, with length L, weight w, and maximum correlation λ is equivalent to a constant weight code with minimum Hamming distance $2\delta = 2(w - \lambda)$ and same length and weight. The size of the largest constant-weight code with given values for L, w, and 2δ is unknown in the general case [6]. Johnson provides an upper bound for the number of codewords in such code [7]

$$\|C\| \leq \left\lfloor \frac{L}{w} \left\lfloor \frac{L-1}{w-1} \cdots \left\lfloor \frac{L-w+\delta}{\delta} \right\rfloor \cdots \right\rfloor \right\rfloor \tag{2}$$

where $\lfloor x \rfloor$ is the largest integer less than or equal to x. Lower bounds are usually obtained by constructing a code with given parameters. An example code with $L = 7$, $w = 3$, and $\lambda = 1$ is shown below,

$$C = \{1100001, 0110010, 0101100, \\ 1011000, 0010101, 1000110, 0001011\} \tag{3}$$

One can construct a code by building a graph with $\binom{L}{w}$ vertices, each corresponding to a codeword, connected if their Hamming distance is larger than or equal to 2δ, and finding the maximum clique [8]. However, in our case finding a large maximal clique results in a sufficiently large code and finding the maximum clique is not necessary. Therefore, we use a greedy depth first search (DFS) algorithm [9] to find a maximal clique (code). In this algorithm, in each step a '1' is added to the current codeword in such a way that it does not cause the hamming distance between the current codeword and previous codewords to decrease below the given number. When a codeword with weight w is obtained, it is added to the code. The algorithm ends when it cannot find a new codeword. The pseudocode of the algorithm is presented in the Appendix.

We assume that the set of codewords (patterns) is decomposed into two subsets. A subset of codewords is only reserved for network association. Once a vehicle enters a cluster, it randomly selects a codeword from the subset reserved for network association. The first few transmissions by the new node is performed with this temporary codeword, until the node learns about the available codewords from its neighbors and moves to a more permanent codeword selected from the complementary subset. The node releases the codeword once it leaves the cluster. We do not discuss the code distribution protocol in this paper and leave it to a future work.

III. Modeling and Performance

A. Probability of Success

In each timeslot, the desired receiver, denoted by u_0, receives the power S_i^m from user u_i where m denotes the timeslot number. In an interference limited network, the desired transmitter, u_j, is successful in sending its packet to u_0 in the mth timeslot if

$$\begin{cases} u_j \in T^m \\ u_0 \notin T^m \\ S_j^m > \frac{1}{\beta} \sum_{i:u_i \in T^m - \{u_j\}} S_i^m \end{cases} \tag{4}$$

where T^m is the set of transmitting users in the mth timeslot and β is the capture ratio.

In a Rician fading channel with Rice factor K, the pdf of the received power, S_i, from u_i transmitting at distance d_i from u_0, is (see e.g. [10])

$$f_{\mathbf{S}_i}(S_i) = \frac{2K}{A_i^2} exp\left(-K(1 + \frac{2S_i}{A_i^2})\right) I_0 \left(\sqrt{\frac{8K^2 S_i}{A_i^2}}\right) \tag{5}$$

where $I_0(.)$ is the modified Bessel function of the first kind and zeroth order. A_i is the amplitude of the line-of-sight component equal to $A_t / d_i^{\frac{n}{2}}$, where A_t is the transmitted power, and n a constant between 2 and 4. For simplicity the superscript m is dropped in (5). This pdf can be transformed by $U_i = 4KS_i/A_i^2$ to a noncentral chi-square distribution with 2 degrees of freedom and noncentrality parameter $\lambda_x = 2K$.

A transmission by u_j is successful *in a frame* if (4) is satisfied at least for one timeslot m in that frame. The probability of success, P_s, is defined as the number of such successful transmissions divided by the number of frames in which transmission is attempted.

To obtain an analytical expression for the probability of success, the pdf of $\sum S_i^m$ is required. This summation is a weighted sum of noncentral chi-square random variables with weights $A_i^2/4K$. In a special case, when all users are placed at equal distances from the receiver, all A_i are the same and therefore $\sum S_i^m$ could be written as $A^2 U^*/4K$ where $U^* = \sum U_i$ is a higher order noncentral chi-square random variable.[1] In the general case, however, the pdf of this sum can be obtained in terms of the Laguerre expansions (see e.g. [11]). Furthermore, the use of the central limit theorem (CLT) or Lyapunov's CLT is not accurate as their conditions are not satisfied. For our purpose, to avoid unnecessary complications, we resort to the Monte Carlo simulation method.

Probability of success also depends on the number of transmitting users in a frame. We assume that all nodes broadcast data (e.g. their location) when they see a need for update. In practice, the location update frequency is related to the velocity of the vehicle. Higher velocity requires more frequent updates. In this paper, we assume that each vehicle independently makes a local decision, whether or not to transmit its location to the neighboring vehicles. Furthermore, we assume these periodical updates are generated according to a Bernoulli model in each frame with probability μ_p. Since the decisions for data transmission are independent, the number of nodes with an active packet in each frame is a Binomial random variable with parameters N and μ_p, where N is the total number of cars in the cluster, which is loosely defined here as the vehicles located in the reception range of u_0. Simulation results for P_s are presented in Section IV.

[1] Sum of two noncentral chi-square random variables is again a noncentral chi-square random variable with a higher order.

978-1-4244-2219-7/08/$25.00 (c)2008 IEEE

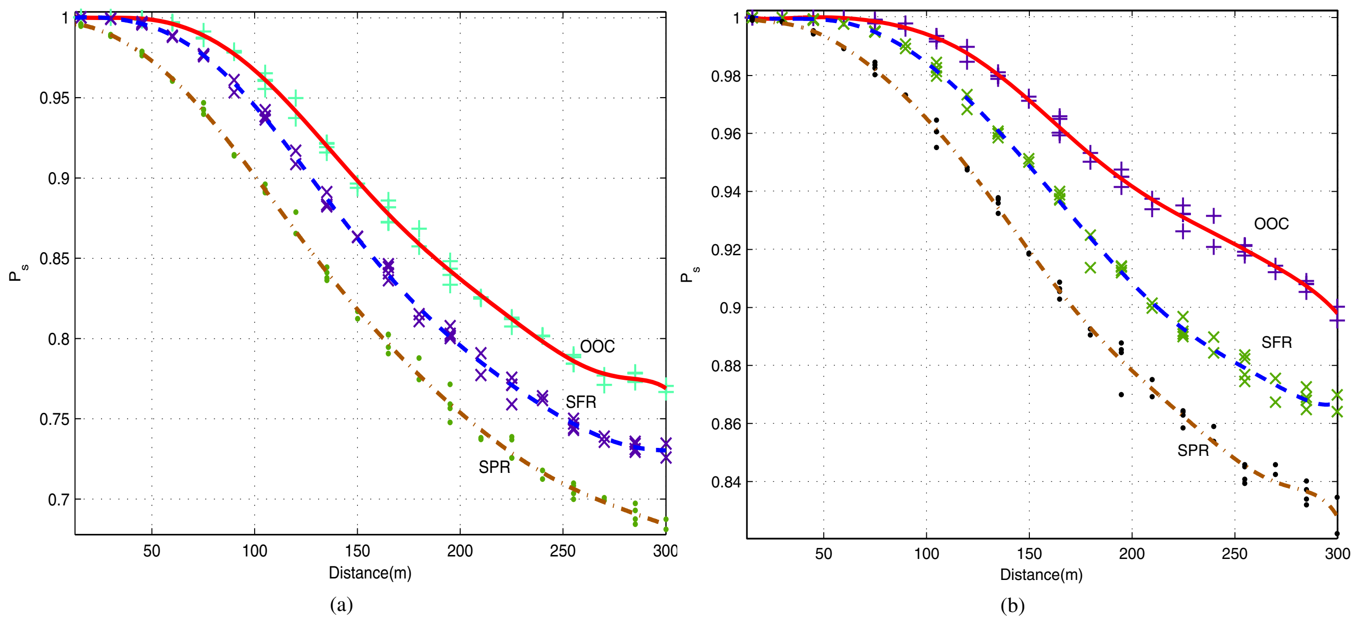


Fig. 1: Simulation results for $\mu_p = 0.3$ and $N = 61$: (a) and (b) show probability of success vs. distance from the receiver for $(L, w) = (64, 6)$ and $(94, 8)$ respectively

B. Lower Bound to Probability of Success

To obtain a lower bound on the system performance, we consider the limiting case of $\beta = 0$, when the capture effect is not present. This means (4) is satisfied only if $T^m = \{u_j\}$, otherwise a collision results in the loss of all colliding packets. This obviously results in worse performance as compared to reception with capture.

Let M be the number of interfering users that intend to transmit in the current frame. Noting that M has a binomial distribution, by using the total probability theorem, the probability that a desired receiver successfully receives a packet from the desired transmitter, P_s, can be written as

$$P_s = \sum_{M=0}^{N-1} \binom{N-1}{M} \mu_p^M (1-\mu_p)^{N-M-1} P_s(M) \qquad (6)$$

where $P_s(M)$ is the probability of successful transmission in a frame with M active interfering users.

For OOC and SFR, a general formula, which is valid for repetition based schemes with certain number of retransmissions, can be used to calculate $P_s(M)$ [4]:

$$P_s(M) = \sum_{k=1}^{w} (-1)^{k+1} \binom{w}{k} \left(1 - \sum_{j=1}^{w} p_j \left[\binom{w}{j} - \binom{w-k}{j}\right]\right)^M \qquad (7)$$

where p_j is the probability that a randomly chosen interferer (possibly the receiver itself since it cannot transmit and receive at the same time) interfere with the desired transmitter in j given slots out of the w slots in which the desired transmitter transmits. For SFR, since codewords are randomly generated, there are $\binom{L-w}{w-j}$ codewords in which a given set of j timeslots are occupied, and $\binom{L}{w}$ codewords in total. Therefore,

$$p_j = \binom{L-w}{w-j} \binom{L}{w}^{-1}. \qquad (8)$$

By substituting (8) in (7) and then in (6), probability of success for SFR is obtained.

For OOC with $\lambda = 1$, because only p_1 is nonzero, (7) reduces to

$$P_s(M) = \sum_{k=1}^{w} (-1)^{k+1} \binom{w}{k} (1 - kp_1)^M \qquad (9)$$

in which p_1 can be empirically obtained for a given code. The final result is achieved by substituting (9) in (6). $P_s(M)$ for SPR can be calculated as

$$P_s(M) = 1 - \left(1 - (\frac{w}{L})(1 - \frac{w}{L})^M\right)^L \qquad (10)$$

In Section IV it is shown that these expressions are in agreement with numerical results with $\beta = 0$ and indeed provide a lower bound for $\beta \neq 0$.

C. Delay

The delay of a successful transmission, denoted by D_s, is defined as the first timeslot in which the power of the corresponding transmitter satisfies (4). When (4) is not satisfied for any timeslot in a frame, i.e. unsuccessful transmission, we consider the delay to be the time from the start of the frame in which transmission has failed until an update is successfully received. [2]

[2] Note that a transmitter has no way of knowing that its transmission has been unsuccessful. Therefore, it will not start retransmission in the following frames and will only transmit a new packet if it has new information to be sent.

978-1-4244-2219-7/08/$25.00 (c)2008 IEEE

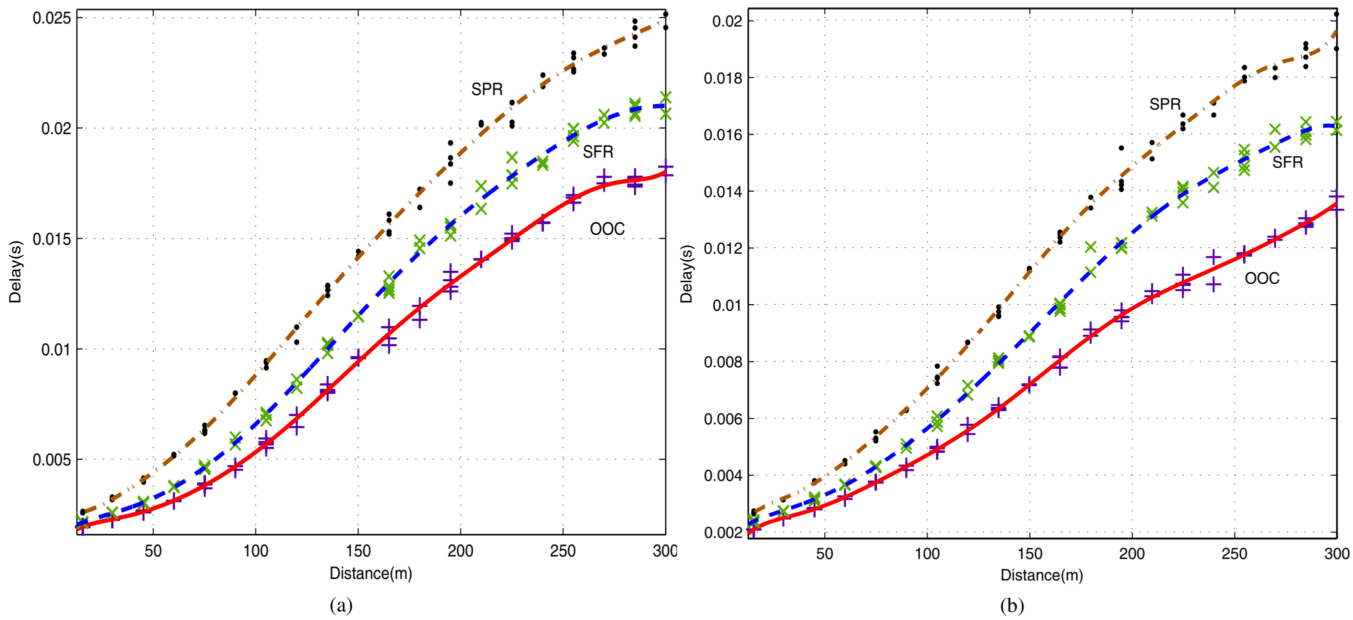


Fig. 2: Simulation results for $\mu_p = 0.3$ and $N = 61$: (a) and (b) show delay vs. distance from the receiver for $(L, w) = (64, 6)$ and $(94, 8)$ respectively

Assuming that the update packet is transmitted in each of the next frames with probability μ_p, the average delay, D can be calculated as

$$\begin{aligned} D &= P_s D_s \\ &+(1-P_s)\sum_{i=0}^{\infty}(1-P_s\mu_p)^i P_s\mu_p(L+iL+D_s) \\ &= D_s + \frac{L}{\mu_p}\left(\frac{1}{P_s}-1\right). \end{aligned} \quad (11)$$

IV. Simulation Results

A. Setup

To simulate a vehicular environment, we consider a cluster of $N = 61$ cars placed on a 3-lane road with 30m separation between cars. Each lane is 4m wide. The middle car is the desired receiver and the probability of success and the delay is obtained for transmissions originating from all other cars and plotted versus their distance from the middle car. As mentioned earlier in our data traffic model, in each frame, each car transmits a packet with probability μ_p. Data rate is assumed to be 10Mbps, packet size is 250B, and each timeslot is $200\mu s$. Path loss exponent, n, is equal to 2, Rice K factor is 3, and $\beta = 0.5$ unless otherwise stated.

In each timeslot, a sample for the power received from each user to the desired receiver is drawn according to the distribution in (5). Then, the successful transmitter, if any, is found by applying (4). Note that the desired receiver may transmit as well. Since the users are not able to receive and transmit at the same time, in such timeslots no packet is received from other nodes. As mentioned earlier, all users with at least one successful timeslot in a frame are considered to be successful in that frame. After a sufficient number of runs, P_s for each user is calculated by dividing the number of successful frames to the number of frames in which the user has transmitted.

For each successful transmission, the delay is found and then averaged to obtain D_s. Using (11), the delay for each method is calculated.

B. Results

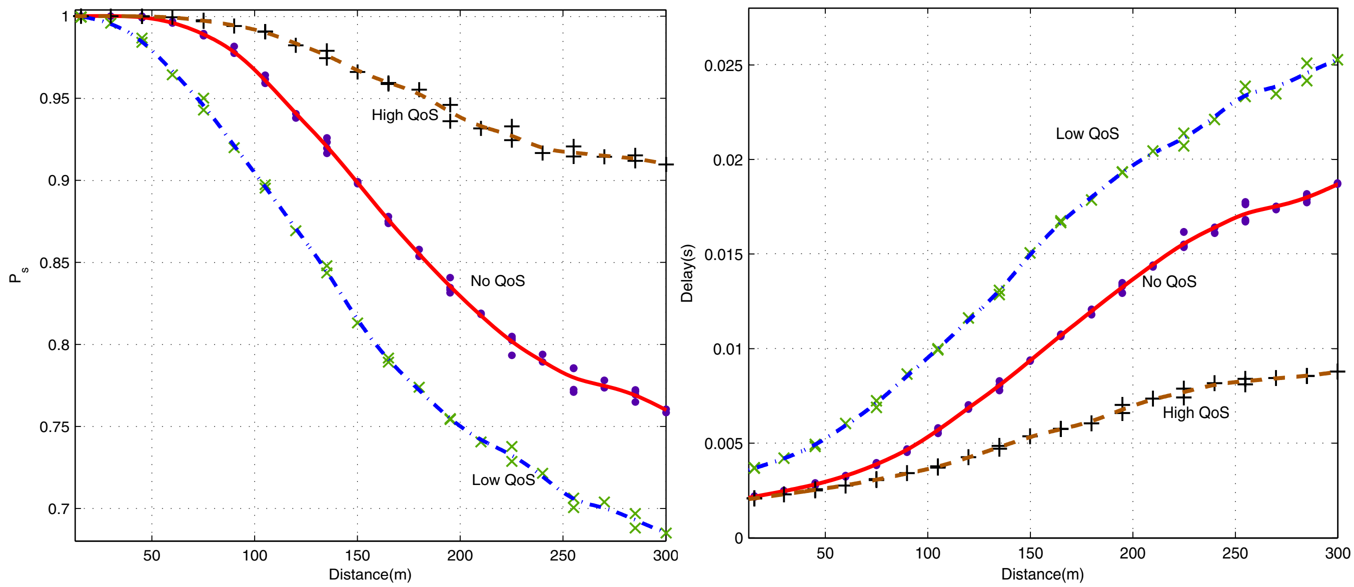
The simulation is performed for $N = 61$, $\mu_p = 0.3$ and two different pairs of values for (L, w), namely $(64, 6)$ and $(94, 8)$. For each (L, w) pair the simulation is run for 20000 frames. Results are shown in Fig. 1a-2b. As observed in these figures, performance improves for higher L and w, for all schemes. This is because larger frames result in fewer collisions and a larger w results in more transmissions, while L is sufficiently large, increases the probability of success. This, however, comes at the cost of lower efficiency. In addition, mean delay averaged across users with different distances is lower for higher (L, w). The standard deviation of the mean delay across users with different distances decreases with increasing (L, w) which means that jitter with the change of location is less.

In Fig.1a-2b, it is observed that OOC performs better than SFR and SPR both in terms of probability of success and delay. SFR, in turn, is better than SPR. However, OOC codes should be carefully chosen and duplicate codes in a network should be avoided while SFR and SPR do not have such limitations. SPR performs worse than other methods, but it does not need frame synchronization.

C. Providing Different QoS Levels

Since different types of messages in vehicular environment have different importance with respect to the safety of motorists, providing different Quality of Service is a much needed feature in vehicular networks. Authors in [12] use a priority scheme based on the Enhanced Distribution Channel Access (EDCA) of IEEE 802.11e [13] to provide different users with

978-1-4244-2219-7/08/$25.00 (c)2008 IEEE



(a) (b)

Fig. 3: Simulation results for $\mu_p = 0.3$ and $N = 61$: (a) and (b) show probability of success and delay vs. distance, respectively, for $L = 64$, $w_{high} = 6$, and $w_{low} = 4$

different classes of QoS. However, it only considers one user to be of high priority, therefore it is not apparent how the system would perform with many users with high priority. In all the methods studied here, we can provide different QoS for different users or for different types of messages. This can be done by allowing different users (or messages) to have different number of repetitions. To demonstrate this effect we apply the following changes to the simulation setup. In each pair of the cars, which have equal distance from the desired receiver, one has a high number of repetitions and the other one has a low number of repetitions. A code with weight 6 is used, but for low-priority users, two of the repetitions are randomly eliminated. As observed in Fig. 3a nodes with higher repetitions have higher P_s compared to users in no-priority setup in which all users have 6 retransmissions, while nodes with lower repetitions have lower P_s. Results for delay are presented in Fig. 3b.

The result of the simulation with $\beta = 0$ is plotted alongside the results obtained from (6) and (10) for each of the three methods for $N = 31$, $L = 64$, $w = 6$, $\mu_p = 0.3$ in Fig. 4. It can be observed that simulation and analytical results are in good agreement. However, for OOC the simulation and (6) differ more than others which is due to the fact that OOC codewords are not as random as other schemes.

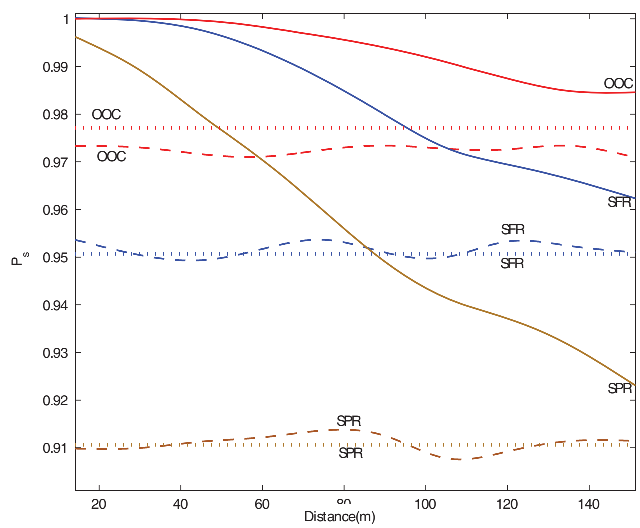


Fig. 4: Simulation results for $\beta = 0.5$ (solid lines), $\beta = 0$ (dashed lines), and analytical results for $\beta = 0$ (dotted lines); $N = 31$, $L = 64$, $w = 6$, $\mu_p = 0.3$

V. Conclusion

In this paper, we have shown that among repetition-based broadcast schemes, OOC-based broadcast systems outperform others, both in terms of probability of success and delay. We have studied the effect of capture on system performance in a typical Rician fading channel and shown that utilizing capture effect can improve the performance of repetition-based systems. The effect of different number of repetitions and its application for providing different priories in the network is also studied. We have shown that implementing different number of repetitions is an effective method for providing different QoS classes with no loss of efficiency.

Appendix
Generation of Optical Orthogonal Codes

In this appendix, we present the pseudocode of the algorithm used to generate optical orthogonal codes. As mentioned before, this algorithm finds a maximal clique as opposed to the maximum clique. Global variables are initialized outside the procedures. Procedure FindCode(L, w) starts by randomly choosing the first timeslot for the current codeword from a set of available positions and calls procedure DFS until there is no possibility for DFS to find another codeword. Procedure DFS searches for available timeslots to add to the current codeword and calls itself recursively to complete the current codeword or become unable to find another available timeslot.

In Fig. 5 a histogram of the position of '1's in a sample code is shown. It can be observed that '1's are roughly distributed in a uniform manner. Therefore, all codewords must, more or less, have equal probability of success. Furthermore, codewords whose first '1' is at the beginning of the codeword have better delay characteristics and can be used for applications with a more stringent delay.

$Code \leftarrow \{\}$
$depth \leftarrow 0$
$found \leftarrow false$
$Codeword \leftarrow \{\}$
procedure FINDCODE(L, w)
  $bits = \{1, \ldots, L\}$ ▷ All bits are available
  while $bits \neq \{\}$ **do**
    $b \leftarrow$ random choice from $bits$
    $Codeword \leftarrow \{b\}$
    $depth \leftarrow 1$
    $found \leftarrow true$
    while $found = true$ **do**
      $found \leftarrow false$
      DFS
    end while
    remove b from $bits$
  end while
end procedure
procedure DFS
  $tmp = Codeword$
  $bits \leftarrow \{1, \ldots, L\} - Codeword$
  Exclude bits that cause low distance with previous codes
  while $bits \neq \{\}$ **do**
    $b \leftarrow$ random choice from $bits$
    $Codeword \leftarrow \{b\}$
    $depth \leftarrow 1$
    if $depth = w - 1$ **then**
      $found \leftarrow true$
      Add $Codeword$ to $Code$
    else
      $depth \leftarrow depth + 1$
      DFS
      if $found = true$ **then**
        return
      end if
    end if
  end while
  $depth \leftarrow depth - 1$
  $Codeword \leftarrow tmp$
end procedure

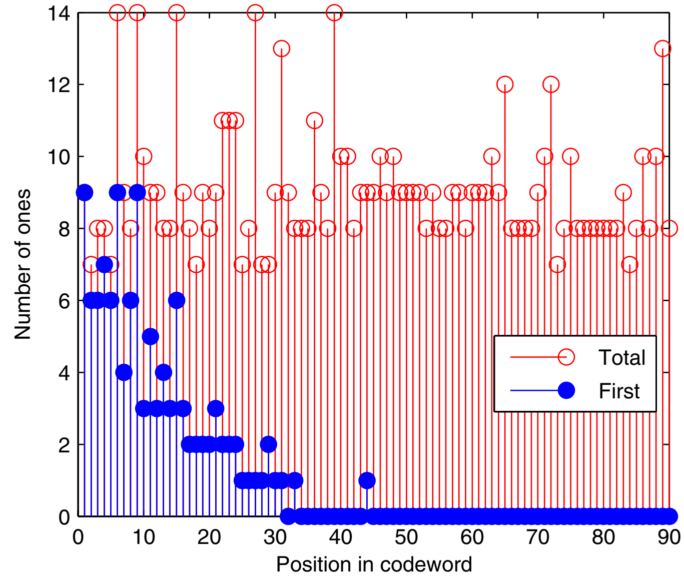


Fig. 5: Distributions of '1's in a sample code and the first occurrence of a '1' in codewords with $(L, w) = (90, 7)$

REFERENCES

[1] D. Jiang, V. Taliwal, A. Meier, W. Holfelder, and R. Herrtwich, "Design of 5.9 ghz dsrc-based vehicular safety communication," *Wireless Communications, IEEE [see also IEEE Personal Communications]*, vol. 13, no. 5, pp. 36–43, 2006.

[2] S. Biswas, R. Tatchikou, and F. Dion, "Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety," *Communications Magazine, IEEE*, vol. 44, no. 1, pp. 74–82, 2006.

[3] F. Chung, J. Salehi, V. Wei, B. Res, and N. Morristown, "Optical orthogonal codes: design, analysis and applications," *Information Theory, IEEE Transactions*, vol. 35, no. 3, pp. 595–604, 1989.

[4] F. Farnoud, B. Hassanabadi, and S. Valaee, "Message Broadcast using Optical Orthogonal Codes in Vehicular Communication Systems," *Wireless Networking for Intelligent Transportation Systems, ICST QSHINE Workshop*, Aug. 2007.

[5] Q. Xu, T. Mak, and R. Sengupta, "Medium access control protocol design for vehicle-vehicle safety messages," *Vehicular Technology, IEEE Transactions*, vol. 56, no. 2, pp. 499–518, 2007.

[6] E. Agrell, A. Vardy, and K. Zeger, "Upper bounds for constant-weight codes," *Information Theory, IEEE Transactions*, vol. 46, no. 7, pp. 2373–2395, 2000.

[7] S. Johnson, "A new upper bound for error-correcting codes," *Information Theory, IEEE Transactions*, vol. 8, no. 3, pp. 203–207, 1962.

[8] A. E. Brouwer, J. B. Shearer, N. J. A. Sloane, and W. D. Smith, "A new table of constant weight codes," *Information Theory, IEEE Transactions*, vol. 36, no. 6, pp. 1334–1380, 1990.

[9] T. Cormen, C. Leiserson, R. Rivest *et al.*, *Introduction to algorithms*. Cambridge, Mass.: MIT Press, 2001.

[10] R. Prasad and C. Liu, "Throughput analysis of some mobile packet radio protocols in Ricianfading channels," *Communications, Speech and Vision, IEE Proceedings I*, vol. 139, no. 3, pp. 297–302, 1992.

[11] A. Castaño-Martínez and F. López-Blázquez, "Distribution of a sum of weighted noncentral chi-square variables," *TEST*, vol. 14, no. 2, pp. 397–415, 2005.

[12] M. Torrent-Moreno, D. Jiang, and H. Hartenstein, "Broadcast reception rates and effects of priority access in 802.11-based vehicular ad-hoc networks," *Proceedings of the first ACM workshop on Vehicular ad hoc networks*, pp. 10–18, 2004.

[13] "IEEE 802.11e Std., Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements," Sept. 2005.

978-1-4244-2219-7/08/$25.00 (c)2008 IEEE