International Journal of Reliable Information and Assurance
Vol.8, No.1 (2020), pp.1-6
http://dx.doi.org/10.21742/ijria.2020.8.1.01

A Performance Model of Controller Area Networks for Vehicles under Unsaturated Traffic

JinYeong Um

*Dept. of Computer Science & Engineering, Dongguk Univ. 30, Pildong-ro 1-gil,
Jung-gu, Seoul, Korea
mog07@dongguk.edu*

Abstract

This paper proposes a performance model of Controller Area Networks (CAN) to measure the effect of bit-error, especially on an identifier field of CAN messages over erroneous channels. CAN were introduced to allow various devices embedded inside cars to communicate in real-time over a shared channel. For resolving collisions in a real-time way, CAN adopt a contention-free protocol, namely a bit-map algorithm where messages are bitwise arbitrated based on their priority, precisely the identifier in CAN's messages when multiple nodes contend for the shared channel. Contamination on the identifier, therefore, can severely degrade the performance of CAN since all senders recognizing errors immediately terminate their transmission and wait for a certain amount of time to resume competition. Furthermore, when the number of errors experienced exceeds the predetermined threshold, CAN stays at the bus-off mode where no messages are sent, requiring outside intervention to be back to the normal operational mode. To predict the performance deterioration of CAN over noisy channels, this paper builds a Markov chain model of CAN.

Keywords: *CAN, Identifier, Arbitration field, Analytical model, Markov chain, Throughput*

1. Introduction

Controller Area Networks (CAN) were designed to run real-time distributed applications which need to exchange data in real-time among sensors, actuators, controllers, and other nodes [1][2]. For real-time transmission, CAN adopts a contention-free algorithm known as a bits-map algorithm. In Bit-map algorithms [3][4], each sender contends to grab the shared channel by at first sending the priority information namely bit-map ahead of data to reduce the collision time. To find out the winner, each sender listens to the channel while sending its bit-map. Here CAN assumes that the channel performs a bitwise AND-operation on each binary bit of the messages' bit-map. When a sender knows that it sends a low-priority message, it immediately stops sending its bit-map while the other senders keep contention until only one sender with the highest priority message is chosen.

Before sending the bit-map, CAN carries out a carrier sense multiple access protocol (CSMA) which allows each node to observe the bus before transmitting data on it. If a sender detects a message's transmission on bus, it waits for some predefined time and then continues to try again until it successfully sends the message. This proscribes corrupting an ongoing transfer on bus. If more than two nodes try to send a message at the same moment, CAN uses collision detection and bit-wise arbitration. One of the sending nodes detects that the message is distorted

Article history:
Received (February 6, 2020), Review Result (March 13, 2020), Accepted (April 16, 2020)

Print ISSN: 2288-8950, eISSN: 2207-5143 IJRIA

into the outside environment of the arbitration field. If any of the fields starting from control field to CRC field there is bit mismatch then there will be bit error which will result in error frames. The nodes then use the error handling of CAN, in which event one of the transmitting nodes being switched off, bus-off mode. On CAN bus controller failure or an extreme accumulation of errors there is a state transition to bus off.

This paper proposes a performance model of CAN to include the effect of bit-error on the identifier acted as a bit-map to measure the effective throughput of CAN under unsaturated traffic. Our model evaluates the performance deterioration due to the error on the identifier and also collision caused by identical identifiers. Note that collision can occur when malfunctioned or malicious devices in CAN would use the same identifier even though each device is allocated with a unique identifier.

This paper is organized as follows. Section 2 introduces the CAN protocol. Section 3 explains the analytical model for evaluating CAN's throughput on the noisy channel. Section 4 presents various results of the numerical analysis based on our model and Section 5, finally, describes conclusions.

2. CAN

CAN are a serial fieldbus communication network in which nodes transmit their messages through a shared bus [5][6]. Note that fieldbus is the name of a family of industrial computer network protocols used for real-time distributed control. For providing real-time communications in vehicles, CAN consist of two protocols such as a CSMA protocol with collision detection (CD) and arbitration based on message priority (AMP).

AMP enables each node to determine the highest priority message among contending messages by comparing identifiers. While sending the identifier, each node monitors whether a message with the higher priority is being delivered. For the priority comparison and also error detection, the physical layer of CAN is assumed to bitwise-AND all bits transferred including identifiers synchronously transmitted and makes the result of bitwise AND operation to all attached nodes to the CAN network.

When identifiers happen to be not unique, messages with the same identifier will collide with each other so that the transmission stops somewhere in the middle of their payload with a one-bit negative acknowledgment. Even though identifiers are uniquely assigned to each node to avoid this problem, this kind of collisions can occur due to malfunctioned or hacked nodes.

3. A performance model for CAN

[Figure 1] shows a proposed Markov chain model [7][8][9] of CAN to evaluate the effective CAN's throughput on erroneous channels and under unsaturated traffic. Precisely [Figure 1] plots a model abstracting how nodes operate in CAN based on their pre-assigned priority. The model of [Figure 1] illustrates all the states in sequence that a node of CAN undergoes to send a message under unsaturated traffic and transitions among them. The unsaturated traffic is assumed to be generated from each node by a Poisson process with the average arrival rate λ, the average number of messages per second, which is denoted a circle on the right top of the model in [Figure 1]. Here, the size of a message is assumed to be fixed. p_{data} is the average message generation rate during the average transmission time of a message.

Each state in the Markov model of [Figure 1] is denoted by two variables *(i, j)* where $I \in [0,2]$ and $j \in [0, w]$. Specifically *0, 1,* and *2* for *i* indicate the data-ready state, on-transmission state, and finally no-data state respectively. The value of *j* means the remaining bits of the outstanding w-bit identifier to be transmitted later. If the first bit of the identifier is sent, for example, the state changes from *(0, w)* to *(0, w-1)*. If a node is the highest priority among current competitors, its state ultimately advances to *(0,0)* with the transition probability $p_{i,j}$. From *(0,0)* CAN can jump to either *(1,0)* or *(2,0)* with the priority of p_s or $(1-p_s)$ respectively depending on collision priority $(1-p_s)$ where p_s is the priority of a successful transmission without collision due to the same identifier intentionally or mistakenly assigned to multiple nodes.

In Figure 1, the transition probability of the node with the *i-th* priority from state *(0, j)* to *(0, j-1)* is denoted as $p_{i,j}$, where *i*∈ *[1, N]* and *j* ∈ *[1, w]*. Here *N* and *w* are the total number of nodes attached to CAN and the size of the identifier field in terms of bits respectively. Then since $p_{i,j}$ is the probability that no nodes with higher probability should not send and at the same time no error occur, for $p_{i,j}$ Equation (1) subtracts one with the product of two probabilities, the chance of no transmission of *(i-1)* nodes with higher probability $(1-p_{i,j})^{i-1}$ and the chance of no error $(1-p_{error})$.

$$p_{i,j} = 1 - p_{data}(1-p_{data})^{i-1}(1-p_{error}) \tag{1}$$

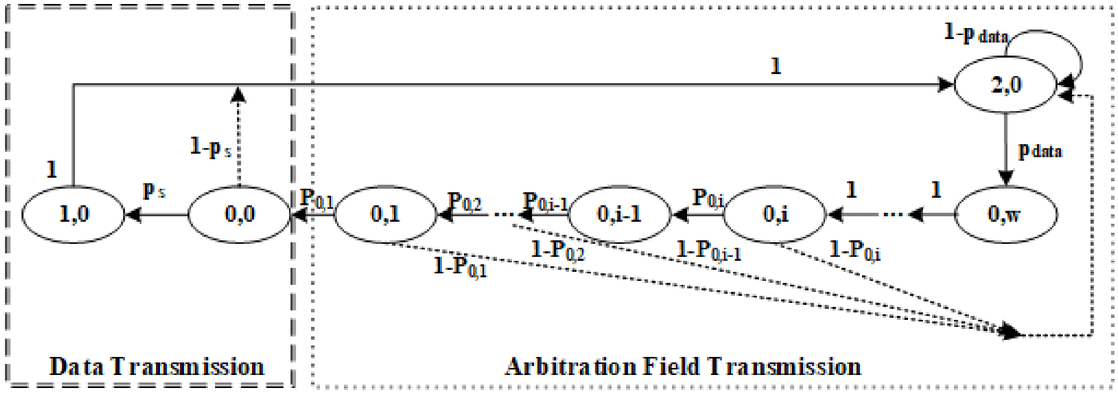


Figure 1. Markov chain models for the node of the i-th priority

Since all the state probabilities must sum to *1*, we have Equation (2) where $p_{0,j}$ is the probability of staying at state *(0, j)* in Figure 1 at a given time.

$$b_{i,0} + b_{0,0} + \sum_{j=1}^{w} b_{0,j} + b_{2,0} = 1 \tag{2}$$

The third term in Equation (2) will be expressed in terms of $b_{0,0}$ since $b_{1,0}$ and $b_{2,0}$ can be also replaced as a function of $b_{0,0}$. $b_{0,0}$ denotes the priority visiting the state *(0, 0)* in [Figure 1] before the node of the *i-th* priority attempts to transmit the payload after the identifier bits are compared. For more details of deriving Equation (3) please refer to [10].

$$\sum_{j=1}^{w} b_{0,j} = \frac{(1-p^w)}{p^w(1-p)} b_{0,0} \tag{3}$$

Based on Equation (1)-(3) Equation (4) now presents $b_{0,0}$ with the probabilities shown in [Figure 1] such as p_{data}, p_{error}, and etc.

$$b_{0,0} = \left[1 + p_s + \frac{1-p^i}{p^i(1-p)} + \frac{w-i}{p^i} + \frac{1}{p^i p_{data}}\right]^{-1} \tag{4}$$

Equation (5) computes the probability τ_i. that the i-th node finally accesses the bus after being picked as a node of the highest priority among current contenders.

$$\tau_i = \left[1 + p_s + \frac{1-p^w}{p^w(1-p)} + \frac{p^w p_s + (1-p^w)}{p^w p_{data}}\right]^{-1} \tag{5}$$

Equation (6) shows S_i which is the normalized throughput of the *i-th* node, defined as the fraction of time during which the *i-th* node delivers a message successfully without preemption from the higher-priority nodes and bit-errors on both its identifier and payload fields. Note that p_{tr-i} is the probability of a successful transmission, which equals to τ_i in Equation (5).

$$S_i = \frac{E[payload\ information\ transmitted\ in\ a\ slot\ time\ by\ the\ i-th\ node]}{E[length\ of\ a\ slot\ time]}$$

$$= \frac{p_s p_{tr\_i} E[P]}{(1-p_{tr\_i})\sigma + p_{tr\_i} p_s + p_{tr_i}(1-p_s)T_c} \tag{6}$$

4. Throughput analyses

For the numerical analysis of the model of [Figure 1], we use the Matlab. [Table 1] summarizes the values of the parameters for these analyses. The length of the payload is set to the longest one that CAN can send for these experiments [3].

Table 1. Values of parameter for CAN simulations

Parameter	Value
Packet payload	8 bytes
ID	12 bits
CRC	16 bits
ACK	2 bits
EOF	7 bits
IFS	7 bits
Channel Bit Rate	1Mbps

[Figure 2] shows the variations of throughput of CAN according to p_{data} by fixing the number of nodes, *N* to *4* and p_{error} to 10^{-4}. I in Priority*I* of [Figure 2] denotes the rank of the priority assigned to each node. At first, [Figure 3] evidences that the higher priority nodes achieve the better throughput regardless of the value of p_{data}. In detail when p_{data} is low (< 0.3) these four nodes of CAN tend to share the channel fairly. However, as p_{data} approaches 1 implying that the highest node has always messages to send, the lower priority nodes rarely take chances to send messages.

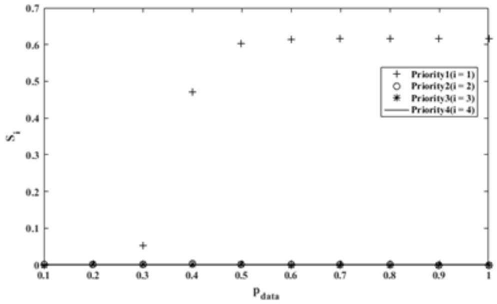
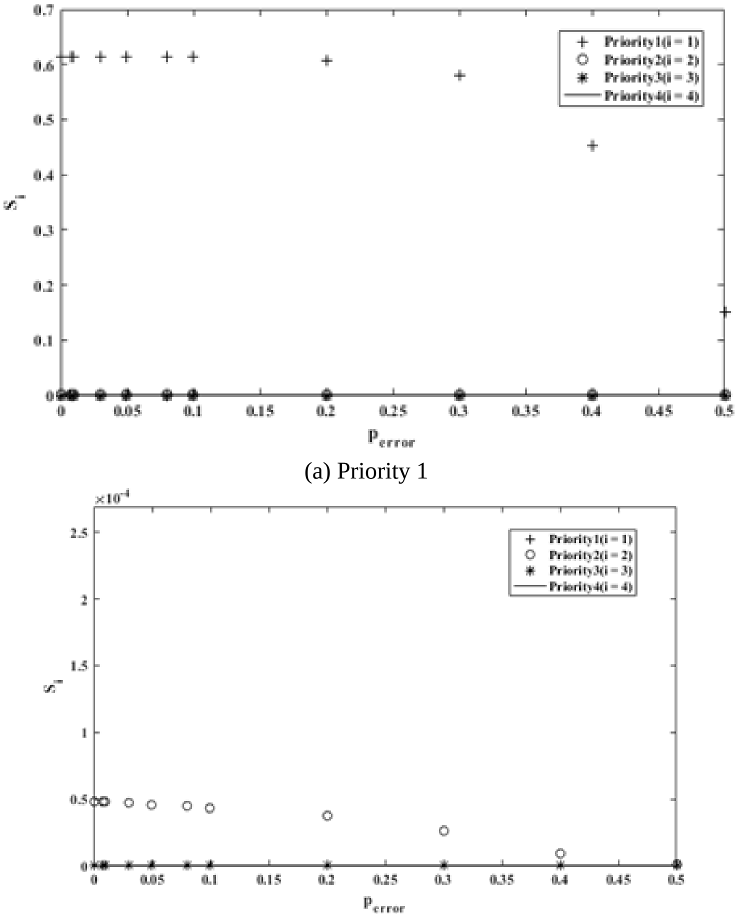


Figure 2. S_i versus P_{data}

[Figure 3] plots the throughput of CAN as a function of p_{error} under the same conditions as those for [Figure 2]. As shown in [Figure 3], at near 0 of p_{error} these four nodes accomplish the fairly good throughput as witnessed in Fig. 3(a) and (b). However as p_{error} becomes larger, the throughput of these four nodes all together steeply fall down to 0 since they repeat competition without delivering messages due to bit-error regardless of the priority.



(a) Priority 1

(b) Priority 2

Figure 3. S_i versus P_{error}

5. Conclusions

This paper presents an analytical model to compute the effective throughput of CAN over noisy channels. Our model takes into account two unsuccessful transmission cases such as collision and bit-error on identifiers to see the performance deterioration of CAN. The numerical results show that identifier field errors severely affect the throughput of CAN. As our future work, we plan to design a reliable data transfer method that mitigates the bit error impact on CAN.

In our future work we plan to design a reliable data transmission method in CAN to solve the bit error problems

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. 2017R1D1A1B03032736).

References

[1] Navet N., Song Y. Q., and Simonot F., “Worst-case deadline failure probability in real-time applications distributed over controller area network,” Journal of systems Architecture, vol.46, no.7 pp.607-617, **(2000)** DOI: 10.1016/s1383-7621(99)00016-8

[2] Jeonghoon Kwak and Yunsick Sung, “Beacon-based indoor location measurement method to enhanced common chord-based trilateration,” Journal of Information Processing Systems, vol.13, no.6, pp.1640-1651, **(2017)**

[3] CAN specification version 2.0 Robert Bosch GmbH, published by copper hill Technologies Corporation.

[4] Texas Instruments, “Introduction to the controller area network (CAN)” Application Report, SLOA101A-August 2002-Revised July, **(2008)**

[5] A. D. Paret, “Multiplexed networks for embedded systems,” Wiley, **(2007)**

[6] Um J. Y. and Ahn J. S, “A performance model for can over noisy channels,” The 2018 World Congress on Information Technology Applications and Services, February, **(2018)**

[7] bianchi G., “Performance analysis of the IEEE 802.11 distributed coordination function,” IEEE Journal on selected areas in communications, pp.535-547, **(2000)** DOI: 10.1109/49.840210

[8] Um J. Y., Ahn J. S., and Lee K. W., “Evaluation of the effects of a grouping algorithm on IEEE 802.15. 4 networks with hidden nodes,” Journal of Communications and Networks, vol.16, no.1, pp.81-91, **(2014)** DOI: 10.1109/JCN.2014.000011

[9] Pollin S., Ergen M., Ergen S.C., Bougard B., Van der Perre L., Moerman I., and Catthoor F., “Performance analysis of slotted carrier sense IEEE 802.15.4 medium access layer,” IEEE Transactions on wireless communications, pp.3359-3371, **(2008)** DOI: 10.1109/TWC.2008.060057

[10] bianchi G., “Performance analysis of the IEEE 802.11 distributed coordination function,” IEEE Journal on selected areas in communications, pp.535-547, **(2000)** DOI: 10.1109/49.840210