# PAPER Worst Case Response Time Analysis for Messages in Controller Area Network with Gateway

# Yong XIE<sup>†\*a)</sup>, Gang ZENG<sup>††b)</sup>, Yang CHEN<sup>†††c)</sup>, Ryo KURACHI<sup>†††d)</sup>, Nonmembers, Hiroaki TAKADA<sup>†††e)</sup>, Member, and Renfa LI<sup>††</sup>, Nonmember

SUMMARY In modern automobiles, Controller Area Network (CAN) has been widely used in different sub systems that are connected by using gateway. While a gateway is necessary to integrate different electronic sub systems, it brings challenges for the analysis of Worst Case Response Time (WCRT) for CAN messages, which is critical from the safety point of view. In this paper, we first analyzed the challenges for WCRT analysis of messages in gateway-interconnected CANs. Then, based on the existing WCRT analysis method proposed for one single CAN, a new WCRT analysis method that uses two new definitions to analyze the interfering delay of sporadically arriving gateway messages is proposed for non-gateway messages. Furthermore, a division approach, where the end-to-end WCRT analysis of gateway messages is transformed into the similar situation with that of non-gateway messages, is adopted for gateway messages. Finally, the proposed method is extended to include CANs with different bandwidths. The proposed method is proved to be safe, and experimental results demonstrated its effectiveness by comparing it with a full space searching based simulator and applying it to a real message set.

key words: CAN, gateway, busy sequence, worst case response time, the minimum distance constraint

#### 1. Introduction

## 1.1 Background and Motivation

To meet the requirements of safety, energy efficiency and infotainment, more and more sensors, actuators and ECUs are added into the automotive electronic system, which increases the complexity of the automotive networks to a large extent [1]. CAN is currently the most widely used network technology inside the automobiles. To reduce design complexity and cost, several CANs are utilized in different sub systems, such as the body system, powertrain system and infotainment system. Therefore, gateway is employed to enable the communication between them [1], [2]. The basic

\*Presently, with the Graduate School of Information Science in Nagoya University.

DOI: 10.1587/transinf.E96.D.1467

function of a gateway is to realize the message exchange between different CANs, where messages from one CAN are first stored in queue inside the gateway and then forwarded into another CAN when they win the arbitration. But other complex functions, such as jitter reduction and message filtering which can reduce the WCRT for messages and the bus load for CAN, can also be implemented inside the gateway [3], [4].

As automotive electronic system is generally a hard real time system, we must analyze the WCRT of CAN message as accurate as possible to get a safe upper bound of the response time. By comparing the analyzed WCRT with the correponding deadline, we can validate the schedulability of messages, otherwise it may result in a catastrophic result. Thus, WCRT analysis is a must for CAN messages. The WCRT analysis for messages in one single CAN has attracted much attention since 1994 [5], [6], but the adding of gateway brings new challenges. Therefore a new WCRT analysis method for messages in gateway-interconnected CANs is desirable.

## 1.2 Related Work and Contributions

There are very few works about the gateway-interconnected CANs: Sommer and Andblind [3] proposed a CAN-CAN gateway embedded system, where the resource dimensioning problem like the gateway processing time and buffer capacity are investigated; Davis and Navet [4] proposed a method to reduce the jitter for gateway messages; Sojka et al. [7] proposed a measurement based method to analyze the latency introduced by gateway. And other related works, such as FlexRay/CAN gateway [8]-[10] and Ethernet/CAN gateway [11], their main contributions are the gateway impelentation methods, and their performance evaluation like the gateway processing delay, reliability and the protocol head overhead are all measurement based. No formal WCRT analysis method is proposed for messages in the above works. For switched-Ethernet and Ethernet AVB (Audio Video Bridging), although their network topologies are similar with gateway-interconnected CANs, the formal WCRT analysis methods proposed for them [12], [13] cannot be directly reused due to the difference of the employed message scheduling algorithms. The full duplex based method is used to eliminate the frame collision for switched Ethernet and Ethernet AVB, while non-preemptive fixed-priority based method is used for CAN. There are in-

Manuscript received September 24, 2012.

Manuscript revised January 24, 2013.

<sup>&</sup>lt;sup>†</sup>The authors are with the Laboratory of Embedded Systems and Networking, Hunan University, Changsha, P. R. China.

<sup>&</sup>lt;sup>††</sup>The author is with the Graduate School of Engineering, Nagoya University, Nagoya-shi, 464–8603 Japan.

<sup>&</sup>lt;sup>†††</sup>The authors are with the Graduate School of Information Science, Nagoya University, Nagoya-shi, 464–8603 Japan.

a) E-mail: yongxie@hnu.edu.cn

b) E-mail: sogo@ertl.jp

c) E-mail: chenyang@ertl.jp

d) E-mail: kurachi@ertl.jp

e) E-mail: hiro@ertl.jp

f) E-mail: lirenfa@vip.sina.com

dustry tools that also claim to support WCRT analysis for messages in gateway-interconnected CANs, but their methods are confidential. To the best of our knowledge, there is no such publication that gives a complete WCRT analysis method for messages in gateway-interconnected CANs.

The main contributions of this work are as follows: (1) it analyzed the challenges for WCRT analysis of messages in gateway-interconnected CANs; (2) it proposed two analysis methods, which target message sets of different size and with different analysis accuracy, to solve such challenges and can get the safe upper bound of the response time for messages, where the situations that CANs with the same or different bandwidths are both included; (3) the correctness of the proposed method is proved; (4) the effectiveness of the proposed method is validated by comparing it with a full space searching based simulator and using a real message set.

### 1.3 Organisation

The remainder of this paper is organised as follows: In Sect. 2, we introduce the message model and the assumptions. In Sect. 3, we analyze the new challenges for WCRT analysis of messages in gateway-interconnected CANs. Then, as the basis of WCRT analysis, Sect. 4 gives two new definitions to analyze the interfering delay for sporadically arriving gateway messages. Section 5 proposes the WCRT analysis method for non-gateway and gateway messages, and then extends it to the case when the gateway-interconnected CANs have different bandwidths. Section 6 presents the experimental results for a simulated and a real message set. Finally, this paper is concluded with section 7.

### 2. System Model and Assumptions

We assume that automotive electronic system consists of two sub systems as shown in Fig. 1, and each sub system includes several ECUs that are connected by CAN. The included two CANs have the same bandwidth and are interconnected by gateway, where messages are exchanged to realize the communication between them (the proposed method will be extended to CANs with different bandwidths as shown in Sect. 5.3). Each ECU  $ECU_N$  includes a message set  $S_N$  that needs to be transmitted on CAN. We use  $m_i$ to represents the message, each CAN message has a unique priority and the subscript of  $m_i$  is used to indicate its priority. Thus,  $S_N = \{m_i \mid m_i \in S_N\}$ .  $m_i$  is indicated by a



Fig.1 System architecture for automotive electronic system.

4-tuple:  $\langle T_i, P_i, C_i, D_i \rangle$ , which represent the period, priority, transmission time and deadline, respectively, and we assume that  $T_i = D_i$ . If i < j, it means that  $m_i$  has higher priority than  $m_i$ . For two communicating ECUs that belong to different sub systems, the communicating messages will be transmitted on the included CAN of its belonging sub system first, and then go through the gateway and be transmitted on the other CAN. We define this kind of message as gateway message, the included CAN of its belonging sub system as its source CAN CANsou and the other CAN as its destination CAN  $CAN_{des}$ . Such as  $m_1$  in  $ECU_1$  is a gateway message, its  $CAN_{sou}$  is  $CAN_1$  and its  $CAN_{des}$  is  $CAN_2$ . For two communicating ECUs that belong to the same sub system, the communicating messages will be only transmitted on its CAN<sub>sou</sub>, and we define this kind of message as non-gateway message. But to keep the consistency of the notation with gateway messages, we also define the other CAN as the  $CAN_{des}$  of non-gateway messages. Such as  $m_2$ in  $ECU_1$  is a non-gateway message, its  $CAN_{sou}$  is  $CAN_1$  and its  $CAN_{des}$  is  $CAN_2$ .

To simplify the WCRT analysis for messages, the following assumptions are made about the gateway [3]: in each transmission direction, for example from  $CAN_1$  to  $CAN_2$ , there is a set of queues (include the input and output queue) to realize the store and forward operation for messages, hence the messages from different transmission directions will not interfere with each other inside the gateway; queues are managed with the fixed-priority based policy; the processing time of the gateway is ignored by assuming that messages are forwarded to their destination CAN as soon as they arrive at the gateway. Actually, the gateway processing time will affect the response time of messages, here for the purpose of simplicity we ignore it. Gateway's effect can be considered by extending the Busy Sequence defined in Sect. 4, and we will try to include it in future work.

#### 3. Problem Analysis

Compared with the WCRT analysis for messages in the single CAN, several new challenges exist for the WCRT analysis of messages in gateway-interconnected CANs. Thus in this section, we will try to clarify the new challenges. For the following parts, we assume that the object message for WCRT analysis is  $m_i$ . We define the WCRT of  $m_i$  as the maximal interval between the release time in its host ECU and the finish time in its destination ECU. However, as the transmission path is different for non-gateway message and gateway message, we have to differentiate the WCRT for these two kinds of message. If  $m_i$  is a non-gateway message, it will be transmitted on its CAN<sub>sou</sub> only, thus we indicate its WCRT as  $r_{s,i}$ . But if  $m_i$  is a gateway message, the transmission path will includes its CAN<sub>sou</sub>, gateway and its CAN<sub>des</sub>, and the corresponding WCRT is usually called end-to-end WCRT, thus we indicate it as  $r_{e2e,i}$ .

For  $m_i$ 's WCRT analysis, the key problem is to analyze all the possible kinds of messages that would cause the interfering delay on  $m_i$ , therefore next we will try to solve

this problem for non-gateway message and gateway message separately. If  $m_i$  is a non-gateway message, three different types of message will interfere with  $m_i$  as shown in Fig. 2 (a). The first type is the shp(i) that represents the set of messages belonging to m<sub>i</sub>'s CAN<sub>sou</sub> and having higher priority than  $m_i$ . The second type is the  $dhp_{GW}(i)$  that represents the set of gateway messages belonging to  $m_i$ 's  $CAN_{des}$ and having higher priority than  $m_i$ . The subscript GW indicates that the included messages of this message set are gateway messages. The third type is the slp(i) and  $dlp_{GW}(i)$  that represent two message sets both having lower priority than  $m_i$  and belonging to  $m_i$ 's  $CAN_{sou}$  and  $m_i$ 's  $CAN_{des}$ , respectively. As CAN messages are scheduled non-preemptively, they can cause the interfering delay to  $m_i$  due to the priority inversion. For example for  $m_2$  in Fig. 1,  $shp(2) = \{m_1\}$ ,  $dhp_{GW}(2) = \{\}, slp(2) = \{m_3, m_4, m_8\}, dlp_{GW}(2) = \{m_5, m_6\}.$ The arriving pattern of the shp(i) messages is periodic as they belong to the same CAN with  $m_i$ , thus we can reuse the existing method proposed for one single CAN to analyze their interfering delay [6]. slp(i) and  $dlp_{GW}(i)$  messages as a whole can only cause the priority inversion to  $m_i$  once, therefore we can include their interfering delay by choosing the message with the maximal  $C_k$  [6]. But for  $dhp_{GW}(i)$ messages, they need to be scheduled in  $m_i$ 's  $CAN_{des}$  first, and then arrive at  $m_i$ 's  $CAN_{sou}$  and cause interference on  $m_i$ . The arriving time of  $dhp_{GW}(i)$  messages in  $m_i$ 's  $CAN_{sou}$  is equal to its finishing time in  $m_i$ 's  $CAN_{des}$  when ignoring the gateway processing time. But because the response time for different instances of  $dhp_{GW}(i)$  message are different in  $m_i$ 's  $CAN_{des}$ , the arriving pattern of the  $dhp_{GW}(i)$  messages in  $m_i$ 's  $CAN_{sou}$  is sporadic. Thus, how to define the interfering delay from the sporadically arriving  $dhp_{GW}(i)$  messages is a challenge for  $r_{s,i}$ 's analysis. Furthermore, as shown in Fig. 1, gateway messages from  $CAN_1$  such as  $\{m_1, m_3, m_4\}$  will interfere with messages that belong to  $CAN_2$ , and conversely gateway messages from  $CAN_2$  such as  $\{m_5, m_6\}$  will also interfere with messages that belong to  $CAN_1$ . Consequently, another challenge for  $r_{s,i}$ 's analysis that is also brought by gateway messages is the inter-dependency between the interfering delay's analysis of messages belonging to two different CANs.

If  $m_i$  is a gateway message, the complexity for  $r_{e2e,i}$ 's analysis will be much higher compared with that of  $r_{s,i}$ 's analysis, because it needs to analyze all the possible kinds of messages that would cause the interfering delay to  $m_i$  in two CANs. When  $m_i$  is transmitted on its  $CAN_{sou}$ , all the possible kinds of interfering messages are the same as that of  $r_{s,i}$ 's analysis for non-gateway messages as shown in Fig. 2(a), please refer to the above paragraph for more details. When  $m_i$  is transmitted on its  $CAN_{des}$ , another three types of messages will interfere with it as shown in Fig. 2 (b). dhp(i)represents the set of messages that belongs to  $m_i$ 's  $CAN_{des}$ and has higher priority than  $m_i$ .  $shp_{GW}(i)$  represents the set of gateway messages that belongs to  $m_i$ 's  $CAN_{sou}$  and has higher priority than  $m_i$ . dlp(i) represents the set of messages that belongs to  $m_i$ 's  $CAN_{des}$  and has lower priority than  $m_i$ . As CAN messages are scheduled non-preemptively,





 $slp_{GW}(i)$  messages cannot arrive at  $m_i$ 's  $CAN_{des}$  at the same time as  $m_i$ , thus they cannot cause the priority inversion to  $m_i$ . Take  $m_3$  in Fig. 1 for example, when  $m_3$  is transmitted on its  $CAN_{sou} CAN_1$ ,  $shp(3) = \{m_1, m_2\}$ ,  $slp(3) = \{m_4, m_8\}$ ,  $dhp_{GW}(3) = \{\}, dlp_{GW}(3) = \{m_5, m_6\}; \text{ when } m_3 \text{ is transmit-}$ ted on its  $CAN_{des} CAN_2$ ,  $shp_{GW}(3) = \{m_1\}, dhp(3) = \{\},\$  $dlp(3) = \{m_5, m_6, m_7\}$ . For  $dlp_{GW}(i)$  and slp(i) messages as a whole when  $m_i$  is transmitted on its  $CAN_{sou}$ , as they can only cause priority inversion to  $m_i$  once, we can easily include their interfering delay by choosing the message with the maximal  $C_k$ . And for dlp(i) messages when  $m_i$  is transmitted on its  $CAN_{des}$ , we can take a similar approach to analyze their interfereing delay. When  $m_i$  is transmitted on its  $CAN_{sou}$ , the arriving pattern of shp(i) messages is periodic. But part of shp(i) messages are gateway messages, which are the  $shp_{GW}(i)$  messages when  $m_i$  is transmitted on its  $CAN_{des}$ , and their arriving pattern is sporadic. Thus first, the arriving patterns of  $shp_{GW}(i)$  messages are different in  $m_i$ 's  $CAN_{sou}$  and  $CAN_{des}$ , and second, the interfering delays caused by them in these two CANs are inter-dependent. The same situation also happens to  $dhp_{GW}(i)$  messages when  $m_i$ is transmitted on its CAN<sub>sou</sub>. Take m<sub>6</sub> in Fig. 1 for example, when  $m_6$  is transmitted on its  $CAN_{sou}$   $CAN_2$ ,  $m_5$  belongs to shp(i) and its arriving pattern is periodic,  $m_1$  belongs to  $dhp_{GW}(i)$  and its arriving pattern is sporadic. But after  $m_6$  going through the gateway and arriving at its  $CAN_{des} CAN_1$ ,  $m_5$  belongs to  $shp_{GW}(i)$  and its arriving pattern is changed to be sporadic,  $m_1$  belongs to dhp(i) and its arriving pattern is changed to be periodic. As a result, the challenges for  $r_{s,i}$ 's analysis of non-gateway messages are also happen to  $r_{e2e,i}$ 's analysis. Furthermore, another challenge is to define in what kind of situation the higher priority gateway messages like  $dhp_{GW}(i)$  and  $shp_{GW}(i)$  messages will cause the maximal interfering delay on  $m_i$  from the end-to-end's point of view.

### 4. Interference Analysis for Gateway Messages

From the above analysis, we can find that the challenges for WCRT analysis are brought by gateway, and the common source of those challenges is gateway messages. As a result, we will focus on how to analyze the interfering delay from gateway messages in this section, which is the basis for WCRT analysis. To clarify the description, we assume that  $m_i$  is a non-gateway message, and the analysis objects are  $dhp_{GW}(i)$  messages,  $m_k \in dhp_{GW}(i)$ . As explained before, the arriving pattern of  $m_k$  in  $m_i$ 's  $CAN_{sou}$  is sporadic. Thus, one typical approach that is used in real-time system is to treat  $m_k$  as a sporadic message in  $m_i$ 's  $CAN_{sou}$ , and set the closest distance between the arriving time of two continuous instances as its period just like [4], [5] did. By doing this, the sporadically arriving  $m_k$  is transformed into a periodically arriving message and we can reuse the existing method to analyze the interfering delay that would be caused by it. But this approach will bring much pessimism, as the response time of  $m_k$ 's instances is variable between  $C_k$  and  $r_{s,k}$ , therefore the variation range of the distance between the arriving time of two continuous instances is very large. We propose a new definition Busy Sequence to capture the characteristic of the sporadically arriving  $m_k$ , which can get a tighter analysis of the interfering delay that would be caused by it.

**Definition 1:** The instance sequence that includes the maximal number of instances of  $m_k$ , which can finish their transmission in  $m_i$ 's  $CAN_{des}$  and arrive at  $m_i$ 's  $CAN_{sou}$  in any time period of t, is defined as the busy sequence  $BS_k$  of  $m_k$ .

Figure 3 describes the  $BS_k$  of  $m_k$ , which starts from  $T_0$ .  $T_0$  indicates the arriving time of  $m_k$ 's first instance in  $m_i$ 's  $CAN_{sou}$ , which equals to its finishing time in  $m_i$ 's  $CAN_{des}$ . It shows that when the transmission of  $m_k$ 's first instance is finished at its WCRT and the transmission of the following other instances are finished as soon as they are arrived in  $m_i$ 's  $CAN_{des}$ , the number of the arrived instances in  $m_i$ 's  $CAN_{sou}$  will be maximized for  $m_k$  in any time period of t



**Fig. 3** The busy sequence definition for  $dhp_{GW}(i)$  messages

that starts from  $T_0$ . For  $BS_k$ , only the distance between the arriving time of the first and the second instance equals to the closest distance between the arriving time of two continuous instances in  $m_i$ 's  $CAN_{sou}$ :  $(T_k - r_{s,k} + C_k)$ . The distance between the arriving time of any other two continuous instances of  $m_k$  in  $m_i$ 's  $CAN_{sou}$  is constrained by  $m_k$ 's period in  $m_i$ 's  $CAN_{des}$ , thus it equals to  $T_k$ . Consequently, compared with the generally used sporadic message model, the definition of busy sequence can get a tighter analysis of the interfering delay that would be caused by  $m_k$ . Equation (1) shows how to calculate the maximal number of arrived instances for  $BS_k$  during any time period of t that starts from  $T_0$ . For periodically arriving messages such as shp(i) messages in  $m_i$ 's  $CAN_{sou}$ , their busy sequences are corresponding to the periodically arriving instance sequences.

$$Num_k(t) = \lceil \frac{t + r_{s,k} - C_k}{T_k} \rceil \tag{1}$$

However, definition of the busy sequence can only define the maximal interfering delay that would be caused by each  $dhp_{GW}(i)$  message, how to define the maximal total interfering delay that would be caused by all  $dhp_{GW}(i)$  messages is quite another matter. The direct and intuitive assumption is that all  $dhp_{GW}(i)$  messages arrive at  $m_i$ 's  $CAN_{sou}$ at the same time, thus the maximal total interfering delay equals to the sum of the maximal interfering delays that are caused by  $dhp_{GW}(i)$  messages. But CAN messages are scheduled non-preemptively, hence for different  $dhp_{GW}(i)$ messages, there is distance constraint between their arriving time in  $m_i$ 's  $CAN_{sou}$ . Next, we give the Definition 2 to capture this fact.

**Definition 2:** For  $dhp_{GW}(i)$  message  $m_k$ , before its arriving time in  $m_i$ 's  $CAN_{sou}$ , there is an interval where no other  $dhp_{GW}(i)$  messages can arrive. The theoretical lower bound of this interval is equal to its  $C_k$ , thus we define  $C_k$  as the minimum distance constraint  $MDC_k$  of  $m_k$ .

We only consider the  $MDC_k$  for the first instances of  $dhp_{GW}(i)$  messages. It is impossible to consider the  $MDC_k$  for all instances of them, considering their interleaving transmissions. Under this assumption, the analyzed total interfering delay for all  $dhp_{GW}(i)$  messages is conservative as their instances will arrive with the busy sequence pattern. And in spite of this, there is already n! different scenarios if there are  $n \ dhp_{GW}(i)$  messages. However,  $MDC_k$ can only defines the relative distance relation between  $m_k$ and other  $dhp_{GW}(i)$  messages. To get the upper bound of the maximal total interfering delay that would be caused by all  $dhp_{GW}(i)$  messages, we need to determine the arriving order of all  $dhp_{GW}(i)$  messages in  $m_i$ 's  $CAN_{sou}$ , so that the absolute distance relation among them can be decided. As a result, we need to search the possible arriving orders of  $dhp_{GW}(i)$  messages. And the objective is to find the arriving order of  $dhp_{GW}(i)$  messages that corresponds to the maximal interfering delay to  $m_i$ . To clarify the description, we use  $ADC_k$  to indicate the absolute distance constraint between  $m_k$  and the first arrived  $dhp_{GW}(i)$  message.



Fig. 4 The searching trace for the exhaustive searching algorithm.

During the searching process, only those messages with decided arriving order have determined  $ADC_k$ .  $ADC_k$ of other messages not only depend on their own  $C_k$ , but also on the arriving order of the messages with decided arriving order. Therefore,  $ADC_k$  of all  $dhp_{GW}(i)$  messages can only be determined after their arriving orders are all decided, which means that the generation of all possible arriving orders of  $dhp_{GW}(i)$  messages is a must to find the arriving order that corresponds to the maximal interfering delay on  $m_i$ . Based on the depth-first searching [14], we implemented an exhaustive searching algorithm that can generate all the possible arriving orders of  $dhp_{GW}(i)$  messages, and then by using Eq. (2) to Eq. (4), we can find the arriving order that corresponds to the maximal interfering delay on  $m_i$ . The complexity of this algorithm is O(n!). The execution trace of the exhaustive searching algorithm will form a searching tree as shown in Fig. 4 (if there are  $n dhp_{GW}(i)$  messages). Each path that starts from the node message of level 1 to the leaf node message of level *n* represents a possible arriving order of  $dhp_{GW}(i)$  messages. For each arriving order,  $ADC_k$ of each  $dhp_{GW}(i)$  message can be calculated with Eq. (2). It means that  $ADC_k$  of  $m_k$  equals to the sum of its own  $C_k$  and ADC<sub>i</sub> of  $m_i$  that is located just before  $m_k$  in the arriving order. For the first arrived  $dhp_{GW}(i)$  message, its  $ADC_k = 0$ . After  $ADC_k$  is determined, the interfering delay  $INF_k(t)$  that would be caused by  $m_k$  can be calculated with Eq. (3), where t represents any time period that begins from the first arrived  $dhp_{GW}(i)$  message. Please refer to Fig. 5 in Sect. 5.1 for a concrete example. Consequently, the total interfering delay  $INF_{sum}$  that would be caused by all  $dhp_{GW}(i)$  messages can be calculated with Eq. (4).

$$ADC_k = C_k + ADC_j, level(m_j) = level(m_k) - 1$$
 (2)

$$t' = t + r_{s,k} - C_k - ADC_k$$
$$INF_k(t) = \begin{cases} 0 & \text{if } t' < 0\\ C_k & \text{if } t' = 0\\ \lceil \frac{t'}{T_k} \rceil C_k & \text{else} \end{cases}$$
(3)

$$INF_{sum}(t) = \sum_{k \in dhp_{GW}(i)} INF_k(t)$$
(4)

**Theorem 1:** For  $m_i$ , the total interfering delay caused by

Algorithm 1 The simplified searching algorithm

 $GW \leftarrow the \ object \ gateway \ message \ set$  $num \leftarrow number \ of \ gateway \ messages \ in \ GW$  $inf_{max} \leftarrow 0 // the maximal total interfering delay caused by GW messages$  $i \leftarrow 1$ while  $(i \le num)$  do  $ADC_i \leftarrow 0$  //set i-th message in GW as the first arrived message and its ADC as 0  $i \leftarrow 1$ while  $(j \le num)$  do if (i! = i) then  $ADC_j = C_j$  //set ADC of other messages in GW as its own transmission time end if  $i \leftarrow i+1$ end while  $inf_{sum} \leftarrow calculate \ the \ INF_{sum} \ caused \ by \ GW \ messages \ with \ Equa$ tion 2 to Equation 4 **if**  $(inf_{sum} > inf_{max})$  **then**  $inf_{max} \leftarrow inf_{sum}$ end if  $i \leftarrow i+1$ end while **return** the maximal total interfering delay from GW messages: in  $f_{max}$ 

all  $dhp_{GW}(i)$  messages can be upper bounded by  $ADC_k$  correlated busy sequences of  $dhp_{GW}(i)$  messages.

**Proof**: First,  $dhp_{GW}(i)$  messages cannot arrive at  $m_i$ 's  $CAN_{sou}$  at the same time, the  $ADC_k$  that is considered for their first instances conservatively captures this fact. Second,  $ADC_k$  is defined based on  $C_k$  of  $dhp_{GW}(i)$  messages, and the proposed searching algorithm is used to find the arriving order of  $dhp_{GW}(i)$  messages that corresponds to the maximal interfering delay on  $m_i$ . And after the arriving order is determined, instances of each  $dhp_{GW}(i)$  message arrive with the corresponding busy sequence pattern, thus its interfering delay on  $m_i$  can still be upper bounded. Consequently, the  $ADC_k$  correlated busy sequences of all  $dhp_{GW}(i)$  messages can upper bound their total interfering delay on  $m_i$ .

As the complexity of the exhaustive searching algorithm is O(n!), it cannot be used when the number of  $dhp_{GW}(i)$  messages is big. Therefore, another simplified searching algorithm that is inspired from [15] is proposed to define the minimum distance relation among  $dhp_{GW}(i)$ messages in Algorithm 1. That is we only consider the  $MDC_k$  between the first arrived  $dhp_{GW}(i)$  message and all other  $dhp_{GW}(i)$  messages, and we ignore the  $MDC_k$  among all other  $dhp_{GW}(i)$  messages. As this assumption will bring more pessimism into the total interfering delay's analysis of all  $dhp_{GW}(i)$  messages, the finally calculated WCRT of  $m_i$ will still upper bound its exact WCRT. Complexity of this simplified searching algorithm is only O(n).

#### 5. The Proposed WCRT Analysis Method

After the above analysis, we solved the challenge about how to define the interfering delay for sporadically arriving gateway messages. To tackle the challenge about the interdependency between messages in two CANs, we propose the following general processes for WCRT analysis:

- First, sort all messages inside the whole message set in order of decreasing priority.
- Second, calculate the WCRT for messages according to the order of decreasing priority. If the current  $m_i$  is a non-gateway message, calculate its  $r_{s,i}$ ; if the current  $m_i$  is a gateway message, calculate its  $r_{e2e,i}$ .

since when we try to calculate the WCRT of  $m_i$ , WCRT of all other messages with priority higher than  $m_i$  in both two CANs are already analyzed. Hence, the interfering delay that would happen to  $m_i$  can be determined. Next, we will show how to calculate the  $r_{s,i}$  and  $r_{e2e,i}$  in detail.

## 5.1 The WCRT Analysis for Non-gateway Messages

The definition of the busy period is fundamental to the WCRT analysis, which represents the maximal interfering delay that would be caused by other messages. And message's WCRT equals to the experienced maximal interfering delay plus the  $C_k$  of itself. For  $r_{s,i}$ 's analysis of non-gateway message  $m_i$ , its level-*i* busy period is defined similarly with [6] as follows:

**Definition 3:** level-*i* busy period of  $m_i$ .

- It starts at some time *t<sub>s</sub>* when a message with priority *i* or higher is queued ready to be transmitted, and no messages with priority *i* or higher waiting for transmission were queued strictly before *t<sub>s</sub>*.
- It is a contiguous period of time during which no message with priority lower than *i* can win arbitration and start transmission.
- It ends at the earliest time *t<sub>e</sub>* when CAN becomes idle, ready for the next round of arbitration and transmission, yet no messages with priority *i* or higher waiting for transmission were queued strictly before *t<sub>e</sub>*.

This time interval  $[t_s, t_e)$  is the level-*i* busy period of  $m_i$ , and  $r_{s,i}$  is corresponding to the maximal level-*i* busy period  $w_{s,i}$  that begins with the so called critical instant [16].

**Definition 4:** The critical instant for the analysis of  $r_{s,i}$  for  $m_i$ .

- The arriving time of  $m_i$  is synchronized with each shp(i) messages.
- The arriving time of  $m_i$  is synchronized with the  $dhp_{GW}(i)$  message set.
- $m_i$  experience the maximal blocking time from  $B_{s,i}$ , where  $B_{s,i} = max(C_i, C_m, C_l), m \in slp(i), l \in dlp_{GW}(i)$ .

**Theorem 2:** When  $m_i$  meets the critical instant conditions, the corresponding level-*i* busy period will be the maximal.

**Proof**: According to the sufficient schedulability test condition proposed for one single CAN [6], when  $m_i$  experiences the maximal blocking from  $B_{s,i}$ , and the arriving of  $m_i$  is synchronized with all other messages with higher priority than  $m_i$ , the corresponding level-*i* busy period will be

the maximal. Based on the given analysis about the possible kinds of interfering messages in Sect. 3, we can extend this sufficient test condition to non-gateway messages in gateway-interconnected CANs, where the interfering delays caused by  $dhp_{GW}(i)$  messages need to be included in the level-*i* busy period, and  $B_{s,i}$  also needs to be extended to include the  $dlp_{GW}(i)$  messages. As  $dhp_{GW}(i)$  message set is asynchronous with  $m_i$ , thus when it also synchronized with the arriving of  $m_i$ , the level-*i* busy period of  $m_i$  will be the maximal  $w_{s,i}$ . According to the Theorem 1, the synchronization between  $m_i$  and the  $dhp_{GW}(i)$  message set means  $m_i$  is synchronized with the first arrived  $dhp_{GW}(i)$  message.

Inside the  $w_{s,i}$ , all messages with priority higher than  $m_i$  will arrive with their busy sequence pattern. Therefore, according to the definition of the critical instant, the maximal level-*i* busy period  $w_{s,i}$  can be calculated iteratively as follows:

$$w_{s,i}^{n+1} = B_{s,i} + \sum_{j \in shp(i)} \lceil \frac{w_{s,i}^n}{T_j} \rceil C_j + \sum_{k \in dhp_{GW}(i)} INF_k(w_{s,i}^n)$$
(5)

$$B_{s,i} = max(C_i, C_m, C_l), m \in slp(i), l \in dlp_{GW}(i)$$
(6)

$$w_{s,i}^0 = C_i \tag{7}$$

In Eq. (5), the first part indicates the maximal blocking time, the second part indicates the interfering delay caused by shp(i) messages, and the third part indicates the interfering delay caused by  $dhp_{GW}(i)$  messages. If the number of  $dhp_{GW}(i)$  messages is small, the exhaustive searching algorithm is used for their interfering delay's analysis, otherwise the simplified searching algorithm is preferable. As we only considered about the sufficient schedulability test condition proposed in [6], the starting value of Eq. (5) is  $w_{s,i}^0 = C_i$ , and it will iterates until  $w_{s,i}^{n+1} = w_{s,i}^n$ . For each possible arriving order of  $dhp_{GW}(i)$  messages, there will be a maximal level-*i* busy period  $w_{s,i}$ . Thus,  $r_{s,i}$  corresponds to the arriving order of  $dhp_{GW}(i)$  messages that contributes to the maximal  $w_{s,i}$ , and it can be calculated as follows:

$$r_{s,i} = max(w_{s,i}^n) + C_i \tag{8}$$

For example, when we try to calculate the  $r_{s,6}$  for  $m_6$ in Fig. 1, all kinds of messages that would contribute to the  $w_{s,6}$  of  $m_6$  are:  $dhp_{GW}(6) = \{m_1, m_3, m_4\}$ ,  $shp(6) = \{m_5\}$ ,  $slp(6) = \{m_7\}$ . Thus, we need to find the arriving order of the three  $dhp_{GW}(6)$  messages that would cause the maximal interfering delay on  $m_6$ . As  $r_{s,i}$  of the three  $dhp_{GW}(6)$ messages are already calculated when we try to analyze the  $r_{s,6}$ , thus busy sequences of them are known. The searching trace for the arriving order searching of the  $dhp_{GW}(6)$  messages is shown in Fig. 5 (a). When the arriving order of the  $dhp_{GW}(6)$  messages is  $m_4 \rightarrow m_3 \rightarrow m_1$ , they will cause the maximal interfering delay on  $m_6$  as shown in Fig. 5 (c), and  $r_{s,6} = 9$ . But if we assume that all  $dhp_{GW}(6)$  messages arrive at  $m_6$ 's  $CAN_{sou}$  at the same time as shown in Fig. 5 (b),  $r_{s,6} = 11$ . As a result, by considering the  $MDC_k$  among



**Fig. 5** An example to show the analysis of  $r_{s,i}$ .

 $dhp_{GW}(i)$  messages, the analysis about the  $r_{s,i}$  of  $m_i$  can be improved. For arriving order of  $m_4 \rightarrow m_3 \rightarrow m_1$ , *level* and  $ADC_k$  of the  $dhp_{GW}(6)$  messages are:  $level(m_4) = 1$ ,  $ADC_4 = 0$ ;  $level(m_3) = 2$ ,  $ADC_3 = 2$ ;  $level(m_1) = 3$ ,  $ADC_1 = 4$ . Figure 5 (d) described the analysis scenario when the simplified searching algorithm is taken to determine the  $MDC_k$  for  $dhp_{GW}(i)$  messages. In this example, when set  $m_4$  as the first arrived message,  $dhp_{GW}(6)$  message set will cause the maximal interfering delay on  $m_6$ . Consequently, only the  $MDC_k$  between  $m_4$  and the arriving order undecided  $m_1$  and  $m_3$  is considered, the  $MDC_k$  between  $m_1$ and  $m_3$  is ignored.

#### 5.2 The WCRT Analysis for Gateway Messages

Considering the intractability for  $r_{e2e,i}$ 's analysis as dis-



**Fig. 6** The division approach for the analysis of  $r_{e2e,i}$ .

cussed in Sect. 3, we take a division approach by dividing it into two separate parts as shown in Fig. 6, where the interdependency between the interfering delays caused by higher priority gateway messages in two CANs is ignored. The first part represents the WCRT of  $m_i$  in its  $CAN_{sou}$ , as it represents the same meaning as  $r_{s,i}$  of non-gateway messages, we also indicate it as  $r_{s,i}$ . The second part represents the WCRT of  $m_i$  in its  $CAN_{des}$ , we indicate it as  $r_{d,i}$ . Since we ignore the gateway processing time for gateway messages,  $r_{e2e,i}$  can be calculated as follows:

$$r_{e2e,i} = r_{s,i} + r_{d,i} (9)$$

For both  $r_{s,i}$  and  $r_{d,i}$ , they will correspond to a leveli busy period as shown in Fig. 6. In Sect. 5.1, we already explained how to calculate  $w_{s,i}$  and  $r_{s,i}$ , next we will explain how to calculate  $w_{d,i}$  and  $r_{d,i}$ . Figure 2(b) illustrates all kinds of messages that would cause the interfering delay to  $m_i$  when  $m_i$  is transmitted on its  $CAN_{des}$ . The arriving pattern of  $shp_{GW}(i)$  and dhp(i) messages for  $r_{d,i}$ 's analysis is the same as that of  $dhp_{GW}(i)$  and shp(i) messages for  $r_{s,i}$ 's analysis, respectively. The only difference is that  $m_i$  is a gateway message that belongs to the same CAN with  $shp_{GW}(i)$ messages, thus there is the  $MDC_k$  between  $m_i$  and  $shp_{GW}(i)$ messages. But for  $r_{s,i}$ 's analysis,  $MDC_k$  only exists among  $dhp_{GW}(i)$  messages. This difference addes much complexity to the  $r_{d,i}$ 's analysis, because when we try to define the total interfering delay that would be caused by  $shp_{GW}(i)$  messages, we need to determine the arriving order of both  $m_i$  and  $shp_{GW}(i)$  messages. And depending on the specific arriving order of  $m_i$ , there will be several candidate positions for dhp(i) messages to start their interference. As a result, the critical instant of the maximal level-*i* busy period  $w_{d,i}$  cannot be uniquely determined. The complexity for  $r_{d,i}$ 's analysis of  $m_i$  is increased to O(n \* n!). In this paper, we take a simplified but more pessimistic approach by ignoring the  $MDC_k$ between  $m_i$  and  $shp_{GW}(i)$  messages. Under this assumption, all kinds of interferences are the same for the analysis of  $r_{d,i}$ and  $r_{s,i}$ , thus  $r_{d,i}$ 's analysis is transformed into the same situation with  $r_{s,i}$ 's analysis. Consequently, we can reuse the analysis method proposed for  $r_{s,i}$ . The maximal level-*i* busy period  $w_{d,i}$  of  $m_i$  can be calculated as follows accordingly for each possible arriving order of  $shp_{GW}(i)$  messages:

$$w_{d,i}^{n+1} = B_{d,i} + \sum_{j \in dhp(i)} \left[ \frac{w_{d,i}^{n}}{T_{j}} \right] C_{j} + \sum_{k \in shp_{GW}(i)} INF_{k}(w_{d,i}^{n}) \quad (10)$$

$$B_{d,i} = max(C_i, C_m), m \in dlp(i)$$
(11)

$$w_{d,i}^0 = C_i \tag{12}$$

In Eq. (10), the first part indicates the maximal blocking time, the second part indicates the interfering delay caused by dhp(i) messages, and the third part indicates the interfering delay caused by  $shp_{GW}(i)$  messages(which searching algorithm should be used for determination of the arriving order of  $shp_{GW}(i)$  messages also depends on the number of the messages). Equation (10) will iterates until  $w_{d,i}^{n+1} = w_{d,i}^{n}$ . And  $r_{d,i}$  can also be calculated accordingly as follows:

$$r_{d,i} = max(w_{d,i}^n) + C_i \tag{13}$$

#### 5.3 Extension to CAN with Different Bandwidths

For different sub systems in automotive electronic system, CANs with different bandwidths such as 125 kbps, 250 kbps and 500 kbps are usually employed. The proposed method of this paper can be extended to include this situation. The specific bandwidth of CAN can only affect the transmission time of messages, other properties of messages are irrelevant with it. For gateway message  $m_k$ , its  $BS_k$  and  $ADC_k$  in its  $CAN_{des}$  only depend on the  $r_{s,k}$  and  $C_k$  of itself. Consequently, first for non-gateway messages  $m_i$ , the interfering delay  $INF_k^s(t)$  caused by  $dhp_{GW}(i)$  message  $m_k$  can be calculated as follows:

$$X = \frac{Bandwidth \ of \ m'_i s \ CAN_{sou}}{Bandwidth \ of \ m'_i s \ CAN_{des}}$$
$$t' = t + r_{s,k} - C_k - ADC_k$$
$$INF_k^s(t) = \begin{cases} 0 & \text{if } t' < 0\\ \frac{C_k}{X} & \text{if } t' = 0\\ \lceil \frac{t'}{T_k} \rceil \frac{C_k}{X} & \text{else} \end{cases}$$
(14)

Another update that needs to be considered for  $r_{s,i}$ 's analysis is about the blocking time from  $dl p_{GW}(i)$  messages as Eq. (16) shows. As a consequence, for non gateway messages, the calculation about the maximal level-*i* busy period  $w_{s,i}$  can be updated as follows ( $r_{s,i}$ 's calculation do not need update):

$$w_{s,i}^{n+1} = B_{s,i} + \sum_{j \in shp(i)} \left[ \frac{w_{s,i}^n}{T_j} \right] C_j + \sum_{k \in dhp_{GW}(i)} INF_k^s(w_{s,i}^n)$$
(15)

$$B_{s,i} = max(C_i, C_m, \frac{C_l}{X}), m \in slp(i), l \in dlp_{GW}(i)$$
(16)

$$w_{s,i}^0 = C_i \tag{17}$$

Then for  $r_{d,i}$ 's analysis of gateway message  $m_i$ . As both  $m_i$  and  $shp_{GW}(i)$  messages are belonging to  $m_i$ 's  $CAN_{sou}$ , the interfering delay  $INF_k^d(t)$  caused by  $shp_{GW}(i)$  message  $m_k$  can be calculated as follows:

$$INF_{k}^{d}(t) = \begin{cases} 0 & \text{if } t' < 0\\ X \cdot C_{k} & \text{if } t' = 0\\ \lceil \frac{t'}{T_{k}} \rceil X \cdot C_{k} & \text{else} \end{cases}$$
(18)

Consequently, the calculation about the maximal level-*i* busy period  $w_{d,i}$  and  $r_{d,i}$  can be updated as follows:

$$w_{d,i}^{n+1} = B_{d,i} + \sum_{j \in dhp(i)} \left[ \frac{w_{d,i}^n}{T_j} \right] C_j + \sum_{k \in shp_{GW}(i)} INF_k^d(w_{d,i}^n)$$
(19)

$$B_{d,i} = max(X \cdot C_i, C_m), m \in dlp(i)$$
<sup>(20)</sup>

$$w_{d,i}^0 = C_i \tag{21}$$

$$r_{d,i} = max(w_{d,i}^n) + X \cdot C_i \tag{22}$$

## 6. Experimental Evaluation

In Sect. 5, we proved that the proposed method can get the upper bound of the response time for messages in gatewayinterconnected CANs. We also implemented a simulator by searching all the possible execution scenarios, where both the message combination and the offset relation among messages are considered. And the result of the simulator is used as the reference to validate the correctness of the proposed two analysis methods. As the running time of the simulator grows exponentially, we use the small message set shown in Fig. 1 as the experimental object, and parameters of the messages are shown in Table 1. We assume that the two CANs are with the same bandwidth. In this experiment, to analyze the interfering delays from sporadically arriving gateway messages, both the exhaustive and the simplified searching algorithm are used for interfering delay's analysis of gateway messages. As the result is the same for the exhaustive searching based WCRT analysis method and the simplified searching based WCRT analysis method, we only show one of them in Table 2. We can find that for  $r_{s,i}$  and  $r_{d,i}$ , the analysis result of the proposed method is close to the simulator's result. But for  $r_{e2e,i}$ , the pessimism is relatively large. The reason is that for the former, the main pessimism comes from the definition of busy sequence. But for the latter, as we took a division approach and ignored the inter-dependency between the interfering delays caused by higher priority gateway messages in two CANs, it brings another main source of pessimism. For this message set, the simulator and the proposed method took about 4 minutes and about 1 second to get the result, respectively.

Because the given real message set provided by automaker is for the single CAN, we divide it into two sub

Table 1 Parameters of the small message set.

| Message's Affiliation |         |         |     |   | $T_i$ | $C_k$ |
|-----------------------|---------|---------|-----|---|-------|-------|
| $m_1$                 | $CAN_1$ | $ECU_1$ | GW  | 1 | 12    | 2     |
| $m_2$                 | $CAN_1$ | $ECU_1$ | NGW | 2 | 12    | 1     |
| $m_3$                 | $CAN_1$ | $ECU_2$ | GW  | 3 | 12    | 2     |
| $m_4$                 | $CAN_1$ | $ECU_2$ | GW  | 4 | 12    | 1     |
| $m_5$                 | $CAN_2$ | $ECU_3$ | GW  | 5 | 12    | 1     |
| $m_6$                 | $CAN_2$ | $ECU_4$ | GW  | 6 | 12    | 1     |
| $m_7$                 | $CAN_2$ | $ECU_4$ | NGW | 7 | 12    | 1     |
| $m_8$                 | $CAN_1$ | $ECU_2$ | NGW | 8 | 12    | 1     |

| Message's Property |     | Simulator's Result |           |             | Analysis Result |           |             |
|--------------------|-----|--------------------|-----------|-------------|-----------------|-----------|-------------|
|                    |     | $r_{s,i}$          | $r_{d,i}$ | $r_{e2e,i}$ | $r_{s,i}$       | $r_{d,i}$ | $r_{e2e,i}$ |
| $m_1$              | GW  | 4                  | 3         | 7           | 4               | 4         | 8           |
| $m_2$              | NGW | 5                  | -         | -           | 5               | -         | -           |
| $m_3$              | GW  | 6                  | 3         | 9           | 7               | 6         | 13          |
| $m_4$              | GW  | 7                  | 6         | 10          | 7               | 6         | 13          |
| $m_5$              | GW  | 7                  | 7         | 9           | 7               | 8         | 15          |
| $m_6$              | GW  | 7                  | 9         | 10          | 9               | 10        | 19          |
| $m_7$              | NGW | 8                  | -         | -           | 12              | -         | -           |
| $m_8$              | NGW | 9                  | -         | -           | 12              | -         | -           |

 Table 2
 Experimental result of the small message set.

message sets with balanced load to imitate the message set employed in two CANs connected by a gateway. The real message set includes 65 messages that are assigned into 14 ECUs, and the bus load is about 53%. And for the given real message set, the message assignment to ECUs is determined. Therefore the division of the real message set into two sub message sets means to divide the ECU set into two sub ECU sets, where the messages included in each ECU are not changed. And in each sub ECU set, there is a message set that needs to be transmitted on the included CAN. We set 50% of the messages in each sub message set as gateway messages. Through this method, we can generate two sub message sets with balanced load for gateway-interconnected two CANs, respectively. We believe they can represent the real gateway message set to some extent. We assume both the two CANs are with bandwidth of 500 kbps, thus bus load of the two CANs are 41.58% and 37.16%, respectively. As the number of the gateway messages is large, the simulator and the exhaustive searching based WCRT analysis method cannot get the result within a reasonable time, thus only the result of the simplified searching algorithm based method is shown. It took about 1 second to get the result and the detail is shown in Fig. 7, where WCRT - SRC represents the  $r_{s,i}$  and WCRT – DES represents the  $r_{d,i}$ , and message with smaller number indicates the message with higher priority.

With the same message set and message assignment, we did another experiment to demonstrate the effectiveness of the proposed method for gateway-interconnected two CANs with different bandwidths. We assume that the two CANs are with bandwidth of 500 kbps and 250 kbps, respectively. Thus bus loads of the two CANs are 41.58% and 74.32%, respectively. It also took about 1 second to get the result and and the detail is shown in Fig. 8. We can find that as bandwidth of one CAN is set as 250 kbps, one consequence is that the  $r_{s,i}$  and  $r_{d,i}$  of some messages are increased to some extent. And another consequence is for non-gateway messages with close priorities but belong to different CANs like message 38 and 39, there are big difference between their WCRT. But if the two CANs are both with bandwidth of 500 kbps, their WCRT are close to each other as shown in Fig. 7. For gateway messages, the situation is similar for Fig. 7 and Fig. 8.

To realize the interfering delay's analysis for sporadically arriving gateway messages, we proposed an exhaustive searching algorithm and a simplified searching algorithm to



Fig. 7 Analysis result: two CANs with the same bandwidths.



Fig. 8 Analysis result: two CANs with different bandwidths.

determine which arriving order of gateway messages will cause the maximal interfering delay on the object message. Thus, we did an experiment to compare the effectiveness of the proposed two searching algorithms. We reused the same big real message set with the same message assignment and message set division as before, but to make the exhaustive searching algorithm usable for WCRT analysis, we only set 30% of the messages in each sub message set as gateway messages. We assume that bandwidth of the two CANs are both 500 kbps. It was found from experiments that when bus loads of the two CANs are relatively low, the analyzed WCRT of the exhaustive searching based analysis method are almost the same as the analyzed WCRT of the simplified searching based analysis method, the advantage of the exhaustive searching algorithm can be shown more clearly only when the bus loads are relatively high. For example when we increase the  $C_k$  of each message by multiplying it with 1.5, bus loads of the two CANs are increased to 69.43% and 62.03%, respectively. It took about 1 minute and 1 second to get the result for the exhaustive searching based analysis method and the simplified searching based analysis method, respectively, and the detail is shown in Fig. 9. WCRT - EXH and WCRT - SIM represent the calculated WCRT when the exhaustive and the simplified searching algorithm is used for interfering delay's analysis of gateway messages, respectively. It was observed that compared with WCRT - SIM, 6 messages' WCRT have been reduced for WCRT – EXH. Thus, compared with the simplified search-



Fig. 9 The comparison for the proposed two searching algorithms.

ing algorithm, the exhaustive searching algorithm that is implemented by considering the  $MDC_k$  for all gateway messages can get a tighter analysis of their interfering delay, although its usability is restricted by its complexity.

#### 7. Conclusion

In modern automobiles, the complexity of automotive networks has been increased greatly, and gateway is commonly utilized to connect different sub networks. WCRT analysis for messages that are transmitted on automotive networks is of great importance to meet the strict safety requirement from automobiles. While the WCRT analysis for messages in one single CAN has been studied intensively so far, there is no WCRT analysis method for messages in gateway-interconnected CANs. For this reason, we proposed a new WCRT analysis method for that in this paper. The proposed method is applicable to CANs with the same or different bandwidths. And differ from the conventional methods, we define two new concepts i.e., busy sequence and the minimum distance constraint to analyze the interfering delay of sporadically arriving gateway messages, which can achieve a tighter bound of the maximal total interfering delay caused by gateway messages. The correctness of the proposed method that it can get a safe upper bound of the response time for messages is proved. Experimental evaluations also demonstrated its effectiveness via comparing it with a full space searching based simulator and applying it to a real message set.

As for future work, we will try to reduce the pessimism for end-to-end WCRT's analysis of gateway messages by considering a holistic approach, and we will also extend our approach to include the effect of gateway processing and to automotive system with more than two CANs.

## Acknowledgements

This work was supported by the National Science Foundation of China (Grant No. 61173036).

#### References

in automotive communication systems," Proc. IEEE, vol.93, no.6, pp.1204–1223, 2005.

- [2] A. Sangiovanni-Vincentelli, and M. Di Natale, "Embedded system design for automotive applications," Computer, vol.40, no.10, pp.42–51, 2007.
- [3] J. Sommer and R. Andblind, "Optimized resource dimensioning in an embedded CAN-CAN gateway," Proc. IEEE SIES, 2007.
- [4] R. Davis and N. Navet, "Traffic shaping to reduce jitter in controller area network (CAN)," Proc. wip of ECRTS, 2012.
- [5] K.W. Tindell, H.A. Hansson, and J. Wellings, "Analysing realtime communications: Controller area network (CAN)," Proc. IEEE RTSS, 1994.
- [6] R.T. Davis, A. Burns, R.J. Bril, and J.J. Lukkien, "Controller area network (CAN) schedulability analysis: Refuted, revisited and revised," Real Time Systems, vol.35, no.3, pp.239–272, 2007.
- [7] M. Sojka, P. Píša, O. Špinka, and Z. Hanzālek, "Measurement automation and result processing in timing analysis of a Linux-based CAN-to-CAN gateway," Proc. IEEE IDAACS, 2011.
- [8] S.H. Seo, J.H. Kim, S.H. Hwang, K.H. Hwon, and J.W. Jeon, "A reliable gateway for in-vehicle networks based on LIN, CAN and FlexRay," ACM Trans. Embedded Systems, vol.11, no.1, article 7, 2012.
- [9] E.G. Schmidt, M. Alkan, K. Schmidt, E. Yuruklu, and U. Karakaya, "Performance evaluation of FlexRay/CAN networks interconnected by a gateway," Proc. IEEE SIES, 2010.
- [10] T. Ishigooka and F. Narisawa, "Message packing algorithms for CAN-based legacy control systems mixed with CAN and FlexRay," Proc. SAE International, 2010.
- [11] A. Kern, D. Reinhard, T. Streichert, and J. Teich, "Gateway strategies for embedding of automotive CAN-frames into Ethernetpackets and vice versa," Proc. ARCS, 2010.
- [12] K.C. Lee, S. Lee, and M.H. Lee, "Worst case communication delay of real-time industrial switched ethernet with multiple levels," IEEE Trans. Ind. Electron., vol.53, no.5, pp.1669–1676, 2006.
- [13] J. Diemer, D. Thiele, and R. Ernst, "Formal worst-case timing analysis of ethernet topologies with strict-priority and AVB switching," Proc. IEEE SIES, 2012.
- [14] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, Introduction to Algorithms, Second ed., MIT Press and McGraw-Hill, 2001.
- [15] J. Rox and R. Ernst, "Exploiting inter-event stream correlations between output event streams of non-preemptively scheduled tasks," Proc. DATE, 2010
- [16] C.L. Liu and J.W. Layland, "Scheduling algorithms for multiprogramming in a hard-real-time environment," IEEE Trans. Comput., vol.54, no.7, pp.813–824, 2005.



Yong Xie was born in 1985, received his B.E. degree and M.E. degree in Computer Science from Hunan University of Science and Technology and Hunan University, China, in 2007 and 2010, respectively. Since 2009, he has been a Ph.D. candidate in computing science of Hunan University. And since 2011, he has been a joint-training Phd student in Nagoya Univeristy. His research interests include embedded communication technologies and real-time performance analysis techniques.



**Gang Zeng** is a lecturer at the Graduate School of Engineering, Nagoya University. He received his Ph.D. degree in Information Science from Chiba University in 2006. From 2006 to 2010, he was a Researcher, and then assistant professor at the Center for Embedded Computing Systems, the Graduate School of Information Science, Nagoya University. His research interests mainly include power-aware computing, real-time embedded system design. He is a member of IEEE and IPSJ.



Yang Chen is a Ph.D student at the Department of Information Engineering, the Graduate School of Information Science, Nagoya University. He received his Master degree in Information Engineering from Xian Jiaotong University in 2007. His research interests include invehicle networks and real-time scheduling theory.



**Ryo Kurachi** received his bachelor and master degrees from Tokyo University of Science in 2000 and 2007. From 2000 to 2006 he was working for AISIN AWCO., LTD. Since 2007, he has been a researcher of the Center for Embedded Computing Systems, Nagoya University (NCES). His research interests include invehicle networks and real-time scheduling theory.



**Hiroaki Takada** is a Professor at the Department of Information Engineering, the Graduate School of Information Science, Nagoya University. He is also the Executive Director of the Center for Embedded Computing Systems (NCES). He received his Ph.D. degree in Information Science from University of Tokyo in 1996. He was a Research Associate at University of Tokyo from 1989 to 1997, and was a Lecturer and then an Associate Professor at Toyohashi University of Technology from 1997

to 2003. His research interests include real-time operating systems, realtime scheduling theory, and embedded system design. He is a member of ACM, IEEE, IPSJ, JSSST, and JSAE.



**Renfa Li** received the B.S. degree, Master degree and PhD degree from TianJin University, China. He is a professor and PH.D. supervisor in Hunan University since 2000. During 1994-1995, he worked as a visiting researcher at the George Washington University. His research interests include embedded systems, cyber-physical systems and wireless sensor networks. He has completed the Project of NHR&D (863 Program) of China in 2010. Currently, he is in charge of research projects of