Torque Compensate Control of 4WID-EV under Non-ideal Network Conditions Based on FTTCAN

Shu-Ju WANG¹,a,* and Ying-Ji LIU²

¹Liaoning Technical University, Fuxin Liaoning123000, China
²Research Institute of Highway, Ministry of Transport, Beijing 100088, China

a wangshuju163@163.com
*Corresponding author

Keywords: Four-wheel-independent-drive Electric Vehicle, Controller Area Network, Non-ideal Network Conditions, Direct Yaw-moment Control, Forecasting Control.

Abstract. Controller Area Network has been widely used in four-wheel-independent-drive electric vehicle control system. However, non-ideal onboard network conditions (e.g. communication delay, data packet missing) could have significant negative influence on the real-time performance and stability of vehicle dynamics control. In order to eliminate the negative influence, flexible time-triggered CAN is introduced, which support the reuse of existing CAN based applications and restrict the communication delay efficiently, through a case study of direct yaw-moment control, a compensate method is proposed to cope with data packet missing and achieve better control results.

Introduction

Controller Area Network (CAN) has been widely used in four-wheel-independent-drive electric vehicle (4WID-EV) control system, the typical distributed control structure of 4WID-EV is shown as in Fig. 1.

![Figure 1. Structure of Distributed Control System of 4WID-EV.](image)

In the distributed real time control networks, many periodic and sporadic real-time messages transport between the vehicle controller and other controllers, those messages are often time-critical and safety-critical, must be properly transported in time so that real time massages meet their temporal requirements. However, non-ideal onboard network conditions (e.g. communication delay, data packet missing) could have significant negative influence on the real-time performance and stability [1], especially in vehicle dynamics control system.

Direct yaw-moment control (DYC) has been proved to be a valid chassis technology for enhancing active security. To eliminate the negative influence on DYC in non-ideal onboard network conditions is the focal issue in this paper, a compensate method to cope with data packet missing is proposed based on flexible time-triggered CAN (FTTCAN), which support the reuse of existing CAN applications and restrict the communication delay efficiently. The paper is structured as follows: In section I, FTTCAN is introduced. In section II, Direct yaw-moment control is
introduced. In Section III, the method to cope with different non-ideal onboard network conditions is introduced, and finally, in section IV, conclusion is presented.

**Flexible Time Triggered Controller Area Network**

Flexible time-triggered CAN makes use of the dual-phase elementary cycle concept to combine time-triggered and event-triggered communication with temporal isolation [2]. Moreover, the time-triggered traffic is scheduled online, facilitating the online admission control of dynamic requests for periodic communication because the respective requirements are held centrally in just one local table. With online admission control, the protocol supports the time-triggered traffic in a flexible way, dynamic planning-based scheduling of FTTCAN is implemented according to consecutive planning based Elementary Cycles (ECs), the bus time of FTTCAN is slotted in consecutive ECs with fixed duration (EC time units) as Fig. 2 [2].

![Figure 2. Elementary Cycle in FTTCAN.](image)

**Definitions of Elementary Cycle**

All nodes are synchronized at the start of each EC by the reception of a particular message known as EC trigger message (TM), which is sent by a particular node called master. The transmission of this message takes $T_{TM}$ constant time units. Following TM within Each EC, two consecutive windows, asynchronous and synchronous, that correspond to two separate phases are defined. The former one is used to convey event-triggered traffic, herein called asynchronous, which has a duration $T_{aw}(n)$, because the respective transmission requirements can be issued at any instant. In asynchronous window, the protocol takes advantage of the CAN arbitration to handle event-triggered traffic in the same way as the original CAN protocol does. The latter one is used to convey time-triggered traffic, called synchronous because it is transmitted synchronously with the ECs. The synchronous window of each EC has a duration $T_{sw}(n)$ that is adjustable according to the traffic planning-based requirements conveyed in the respective EC trigger message and its relative starting instant is also conveyed in the trigger message. There is a strict temporal isolation $T_a$ between both phases to prevent the transmission cannot complete within the respective window [2].

**Synchronous Requirements Table**

In FTTCAN, the time-triggered traffic is dynamic scheduled online according to the data field of each EC trigger message, and supports scheduling centrally in the master node. Synchronous requirements Table (SRT) which resides in the master node, is defined to express the temporal attributes of the synchronous messages. The SRT is organized as follows,

$$SRT \equiv \left\{ SM_i \left( L_{D,i}, T_{C,i}, P_i, T_{D,i}, P_i \right), i=1 \ldots N_s \right\}$$ (1)

$L_{D,i}$ is data length, $T_{C,i}$ is the respective maximum transmission time, $P_i$ stands for the relative phasing, $T_i$ for period, $T_{D,i}$ for deadline, and $P_i$ for fixed priority. Both $P_i$, $P_i$, and $T_{D,i}$ are expressed as integer multiples of the EC duration. $N_s$ is the number of synchronous messages [2].
Implementing of Planning Based Scheduling Messages

According to the SRT in master node, an online scheduler builds the synchronous schedules for each EC. These schedules are then inserted in the data area of the respective EC trigger massage and broadcast with it. Each schedule in the respective EC specifies the synchronous messages scheduled for the current EC in data field of itself as in Fig.3 [2].

Each node needs time to decode the received trigger message for finding out which synchronous messages will be transmitted in the following EC, so the asynchronous window precedes the synchronous one. All nodes who produce synchronous messages have to decode the EC trigger message and check whether they are producers of the specified messages. The checking is implemented by scanning a local table in this node. The specified synchronous messages will transmit when the synchronous window arrives, and eventual collisions on bus access are resolved by the regulations of CAN [2].

Direct Yaw-moment Control

Direct yaw-moment control aim at improving the lateral dynamic performance by imposing drive or brake force on both sides wheels. Take the two wheels linear vehicle model as control object, take the yaw rate and side slip angle as system status, take the front wheel angle and yaw moment torque as control variable, the state space of system can be described as[3],

\[
\begin{bmatrix}
\dot{\beta} \\
\dot{\gamma}
\end{bmatrix} =
\begin{bmatrix}
-2 \frac{C_f + C_r}{mv} & -1 - 2 \frac{l_f C_f + l_r C_r}{mv^2} \\
-2 \frac{l_f C_f - l_r C_r}{I_z} & -2 \frac{l_f^2 C_f + l_r^2 C_r}{I_z v}
\end{bmatrix}
\begin{bmatrix}
\beta \\
\gamma
\end{bmatrix} +
\begin{bmatrix}
\frac{2C_f}{mv} \\
\frac{2l_f C_f}{I_z v}
\end{bmatrix} \delta_f
\]

Where \( m \) is mass of vehicle, \( I_z \) is vehicle yaw moment rotational inertia, \( v \) is vehicle speed, \( l_f \) and \( l_r \) is distance between front axle and rear axle, \( C_f \) and \( C_r \) is cornering stiffness of front wheel and rear wheel, \( \beta \) is side slip angle of vehicle, \( \gamma \) is yaw rate of vehicle, \( \delta_f \) is front wheel steering angle, \( M \) is direct yaw moment.

Take the steering angle as input, \( \gamma \) as control target, \( M \) as control variable, the target yaw rate of vehicle is [4, 5],

\[
\gamma_d = \frac{K_f}{s + T_f} \delta_f
\]

Where \( \gamma_d \) is the target yaw rate, \( K_f \) and \( T_f \) is design parameter of vehicle, \( s \) is Laplace operator.

The DYC control rate [4] is
\[ M = K_5 \left( \frac{K_1}{s+T_g} \delta_j - \frac{K_2}{s+T_g} \gamma - \frac{K_3}{s+T_g} \delta_j - \frac{K_4}{s+T_g} \delta_j - \frac{K_6}{s+T_g} \right) \] (4)

Where \( K_1, K_2, K_3, K_4, K_5, K_6, T_g \) is relevant design parameters.

In FTTCAN, the whole control period of 4WID-EV is divided into EC time, adopt the bilinear transformation to acquire discretization model of DYC,

\[ s = \frac{2(z-1)}{T(z+1)} \] (5)

Where \( T \) is control period of system, \( z \) is \( z \) transformation factor.

To combine Eq. 4 and Eq. 5, the DYC control rate [6] will be,

\[ M = \frac{K_1 \left( b_1T + 2 \right) z + K_2 \left( b_2T - 2 \right) \delta_j - K_3 \left( b_1T + 2 \right) z + K_4 \left( b_2T - 2 \right) \gamma}{(a_1T + 2)z + a_2T - 2} \] (6)

Where \( a_1, a_2, b_1, b_2, k_1, k_2 \) are algebraic combination of \( K_1, K_2, K_3, K_4, K_5, K_6, T_g \) respectively.

**Dynamic Control of 4WID-EV in Non-ideal Network Conditions**

**For Communication Delay Conditions**

For a 4WID-EV, its master node of FTTCAN will be the vehicle controller, all the dynamic control relative messages will be scheduled in vehicle controller. For a given time-triggered message \( i \) in FTTCAN, which is an element of message subset \( S(n) \), its worst transmission delay \( T_{Ri} \) can calculated as[7],

\[ T_{Ri} = T_{C_i} + \sum_{\forall j: hp(j) \geq S(n) \cap S} T_{C_j} \] (7)

Where \( T_{C_i} \) is the transmission time of message \( i \), \( hp(i) \) denotes higher priority, \( S(n) \) is the time triggered message set arranged in EC(n).

When designing the FTTCAN scheduling table in master nodes, the dynamic control messages will be arranged into some \( S(n) \). From Eq. 7 we can see that for a given time-triggered message \( i \), its worst transmission delay in FTTCAN is limited, which will shorter than the duration \( T_{SW}(n) \), will obviously shorter than system control period, so FTTCAN will solve the communication delay problem which happened in traditional CAN.

**For Data Packet Missing Conditions**

There are two situations data packet missing could happen, one is in forward channel, the control message transport from master node to some motor node in network, the other is in backward channel.

When the data packet missing situation happened in forward channel in EC(n), the best plan is the target motor node send a request message to network and make master node known which one missing, the request message have higher priority than other event triggered message and lower priority than EC triggered message, when master node receive the request message, it ask for sending the missing data again right after the EC(n+1) triggered message, from Fig. 2 we can see that the resend message will obtain the use authority of CAN bus because of its higher priority, the transmission delay of the resend message waste a triggered message duration than its original worst transmission delay, and shorter than the system control period.

When the data packet missing situation happened in backward channel, the master node cannot achieve system status data, a forecasting controller could be established in master node.
First, the master node store recent vehicle status and torque data in local buffer, and then based
on the DYC control algorithm and combine the history data storage, the forecasting controller will
create torque control sequence as[6],

$$[M_{EC_n}, M_{EC_{n-1}}, \cdots M_{EC_{n-k}}]^T$$

When data packet missing happened in backward channel, mater node adopt the forecasting data $M_{EC_n}$ as control order send in EC(n), if consecutive data packet missing happened, such as plug-in component loose, mater node adopt the corresponding torque control sequence cope with the serial number of EC.

Conclusion

To cope with non-ideal onboard network conditions in four-wheel-independent-drive electric vehicle control network, flexible time-triggered CAN is introduced in this paper, which support the reuse of existing CAN based applications, for communication delay condition, FTTCAN will restrict the communication delay efficiently. For two data packet missing condition, compensating method based on FTTCAN and forecasting control method based on the direct yaw-moment control algorithm is proposed, those method can achieve better control results.

Acknowledgement

This research was financially supported by the National Natural Science Foundation of China (51405213, 51305181).

References