Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Distributed fault-tolerant strategy for electric swing system of hybrid excavators under communication errors*

Da-hui GAO[1,2], Qing-feng WANG[1], Yong LEI[†‡1]

([1]*State Key Laboratory of Fluid Power and Mechatronic Systems, Zhejiang University, Hangzhou 310027, China*)

([2]*Beijing Electro-Mechanical Engineering Institute, Beijing 100074, China*)

[†]E-mail: ylei@zju.edu.cn

Received Feb. 25, 2016; Revision accepted July 10, 2016; Crosschecked July 14, 2017

Abstract: A distributed fault-tolerant strategy for the controller area network based electric swing system of hybrid excavators is proposed to achieve good performance under communication errors based on the adaptive compensation of the delays and packet dropouts. The adverse impacts of communication errors are effectively reduced by a novel delay compensation scheme, where the feedback signal and the control command are compensated in each control period in the central controller and the swing motor driver, respectively, without requiring additional network bandwidth. The recursive least-squares algorithm with forgetting factor algorithm is employed to identify the time-varying model parameters due to pose variation, and a reverse correction law is embedded into the feedback compensation in consecutive packet dropout scenarios to overcome the impacts of the model error. Simulations and practical experiments are conducted. The results show that the proposed fault-tolerant strategy can effectively reduce the communication-error-induced overshoot and response time variation.

Key words: Fault tolerant; Delay compensation; Controller area network; Communication errors; Electric swing system of hybrid excavator

http://dx.doi.org/10.1631/FITEE.1601021 **CLC number:** TP336

1 Introduction

The swing motion performance of the swing platform (e.g., start–stop performance, speed smoothness, and output tracking performance) is of great importance for the operability of excavators. To achieve good swing motion performance, the swing system of excavators has to achieve good control performance. Compared to conventional hydraulic excavators, hybrid excavators adopt an electric swing system (ESS) to replace the conventional hydraulic-driven swing system. In ESS, the swing platform is driven by an electric swing motor, whose driver is controlled by the central controller through a controller area network (CAN) (Lin *et al*., 2010a; 2010b; Kim *et al*., 2012; Yao and Wang, 2015). Since the electric motor has a smaller damping capacity than the hydraulic motor, ESS is more dependent on the control performance than a hydraulic-driven swing system. Moreover, due to the introduction of CAN, the control loop of ESS is more vulnerable than that of the conventional hydraulic circuit, since CAN is more prone to communication errors than a hydraulic circuit. In practical applications, due to the harsh working conditions, the communication errors caused by electromagnetic interferences and various malfunctions (e.g., loose connections and cable fatigue) are inevitable in the CAN of a hybrid excavator, especially those caused by intermittent faults, which can hardly be avoided by routine maintenance (Gao and Wang, 2014; Lei *et al*., 2014). If communication errors occur in CAN, the delay of the CAN

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 51475414, 51475422, and 51521064) and the National Basic Research Program (973) of China (No. 2013CB035405)

ORCID: Yong LEI, http://orcid.org/0000-0003-0235-5203

messages that carry the control commands and feedback signals of the closed control loop will increase randomly, possibly causing packet dropouts. Ultimately, network-induced delays and packet dropouts degrade ESS's control performance and may even cause damage to ESS's components. Hence, maintaining good control performance for ESS under communication errors is of great significance for the safety and operability of hybrid excavators.

In the research on hybrid excavators, the structures, control strategy, energy management, and energy regeneration have attracted most attention (Wang *et al*., 2005; Kagoshima *et al*., 2007; Katrasnik, 2007; Xiao *et al*., 2008; Wang *et al*., 2009; Lin *et al*., 2010a; 2010b; Kwon *et al*., 2010; Wang and Wang, 2014). Very little attention has been paid to the swing motion performance (Jin *et al*., 2012; Kim *et al*., 2014). To the best of our knowledge, the swing motion performance of ESS under communication errors has not been addressed. Therefore, it is necessary to study the fault-tolerant strategy for ESS to achieve good swing motion performance under communication errors.

In the proposed context, the most important issue is the compensation of communication-error-induced delays and packet dropouts. In the past few decades, a large number of studies on the compensation of delays and packet dropouts in networked control systems (NCSs) have been carried out, and various methods have been proposed (Gupta and Chow, 2010; Zhang *et al*., 2013). Luck and Ray (1990; 1994) reshaped the random network delays to deterministic delays by using two first-in first-out (FIFO) queues such that the NCS becomes time-invariant. Then they used an observer to estimate the plant states and a predictor to implement the predictive control based on past output measurements. Since the size of the FIFO queues is set according to the worst delays, the dynamic model of the plant has to be very precise to achieve accurate predictions. Nilsson (1998) proposed an optimal stochastic control method for NCS with random delay. This method treats the effects of random network delays as a linear quadratic Gaussian (LQG) problem and gives better performance than the queue-based compensation method. However, it requires the statistic characteristics of the network delay be known in advance. Tipsuwan and Chow (2004) developed a gain scheduler middleware (GSM), which does not require the controller be redesigned, to alleviate the network delay. For this method, the distribution model of the delay should be identified and the gain scheduler has to be updated frequently. In addition, the network traffic will increase because of the use of the probing packet. Tian and Levy (2008) proposed three model-free strategies, which use the variation of the past control commands to estimate the lost control command to compensate for the packet dropouts in the forward channel. The application of this method is simple. However, the compensation effect is limited, especially when control commands vary rapidly.

Because of the various limitations of the aforementioned methods, recent related studies have focused on three advanced solutions. One is to use a robust control method in NCSs (Shi and Yu, 2011; Shi *et al*., 2013; Du *et al*., 2014; Shuai *et al*., 2014). The main idea of this solution is to model the random delays and packet dropouts in the system model, and then design a controller to satisfy the proposed stabilization criteria based on the worst-case delay. Since in most cases, the actual delay is smaller than the worst-case delay, the control parameters will be conservative and the system will not work at the optimal conditions most of the time. Another solution is to use adaptive control methods (Kruszewski *et al*., 2012; Rahmani and Markazi, 2013; Wang *et al*., 2014). Usually, the system is modeled as a switched system, and a series of controllers with the gain depending on the size of delays are designed. During each control period, the smart actuator receives a set of control commands and selects one according to the current delay size. The ranges of delays and packet dropouts are prespecified. For this method, multiple controllers are required to be designed, and the network traffic will increase as a result of the transmission of redundant control commands. The third solution is to use predictive control methods, especially the networked predictive control (NPC) (Liu *et al*., 2007; Song *et al*., 2013; Yang *et al*., 2014). NPC predicts the future states of the plant based on the latest received states by using the plant model, and generates a series of future control commands on the controller side. Then, the controller sends all the control commands to the actuator in one packet. The actuator selectively chooses one control command to be implemented in the plant according to the actual delay in the forward channel (controller to actuator). NPC is effective in many applications if an accurate

plant model can be obtained. However, more network bandwidth is required by NPC since more than one control command is transmitted during each control period. In addition, the enlarged packet size will increase the delay of the control commands.

In hybrid excavators, CAN has several limitations, which are nonnegligible in delay compensation: (1) The network bandwidth is limited and the bus load is usually high. The increase of bus load is unacceptable since it will degrade the communication performance, and affect the performance of all the subsystems sharing the same network. (2) Communication errors occur randomly. The network delay and packet dropout vary widely, and their statistic characteristics cannot be obtained in advance. (3) The inertia of the swing platform is time-varying. The plant model cannot be obtained precisely. These limitations make the delay compensation much more difficult. As can be seen from the literature, none of the current delay compensation methods can overcome these limitations simultaneously. Therefore, it is necessary to develop a novel delay compensation method that can adaptively alleviate the adverse impacts of delays and packet dropouts without requiring additional network bandwidth.

This paper aims to develop a fault-tolerant strategy based on a novel delay compensation scheme (DCS) for ESS, so that ESS can retain normal performance under communication errors. The advantages of the proposed method are as follows: First, the bus load of CAN will not be affected by using a distributed compensation structure since no additional signal is required to be transmitted on the CAN bus. Second, it is applicable to various types of communication errors, and the statistic characteristics of communication errors are not required to be given in advance. Third, it is applicable to time-varying systems because of the employment of online model identification and the reverse correction law. The result of this work will improve the reliability of ESS, and may also provide a new solution to NCSs with limited network bandwidth.

2 Analysis of electric swing system

2.1 System description

Fig. 1 shows a typical configuration of the hybrid excavator's ESS. In ESS, the swing platform is driven by an electric swing motor powered by a super capacitor. The actual motor speed is detected by the swing motor driver through an encoder. The reference output of the swing platform, which is set by the operator through the handle, is detected by the central controller. The swing motor driver communicates with the central controller through a CAN shared with other subsystems. During the operation, the swing motor is controlled by the central controller through the swing motor driver to track the reference speed given by the handle. The swing motor works in the torque control mode. To achieve good tracking performance, a closed control loop involving the swing motor driver and the central controller is developed based on CAN. The control structure of ESS is given in Fig. 2, where the parameters are defined in Table 1.

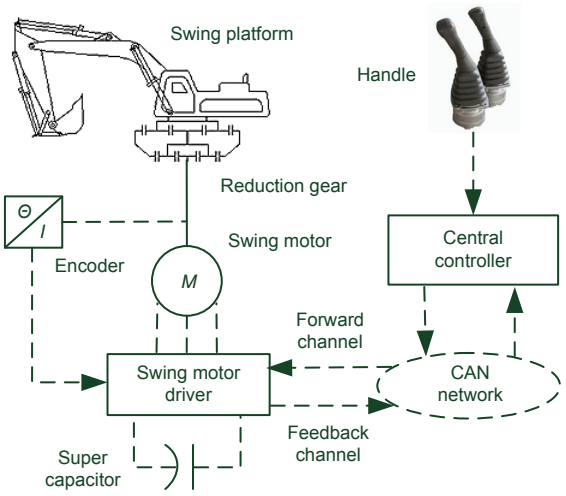

Fig. 1 A typical configuration of a hybrid excavator's electric swing system (CAN: controller area network)

During an arbitrary control period, the central controller first detects the handle input and determines the reference speed $n_r(k)$ according to the input characteristic curve. Then it generates the target motor torque $T_e(k)$ according to the speed tracking error $e(k)$ based on the proportional–integral (PI) control law. Afterward it sends the target motor torque to the swing motor driver. The swing motor driver applies the latest received target motor torque $T(k)$ to the torque controller and conducts a closed-loop torque control by detecting the actual motor torque $T_a(k)$ in real time. Simultaneously, the swing motor driver samples the actual motor speed $n(k)$ and sends it to the central controller.

Table 1 Parameter descriptions

Parameter	Description
$n_r(k)$	Reference motor speed in the kth (k=1, 2, …) control period (r/min)
$n_m(k)$	Newest feedback motor speed received by the central controller in the kth control period (r/min)
$n(k)$	Actual motor speed in the kth control period (r/min)
$e(k)$	Tracking error of the motor speed in the kth control period, where $e(k)=n_r(k)-n_m(k)$ (r/min)
$T_e(k)$	Target motor torque in the kth control period (N·m)
$T(k)$	Newest target motor torque received by the swing motor driver in the kth control period (N·m)
$T_a(k)$	Actual motor torque in the kth control period (N·m)
$\tau_{sc}(k)$	Delay of the feedback motor speed in the kth control period (s)
$\tau_{ca}(k)$	Delay of the target motor torque in the kth control period (s)

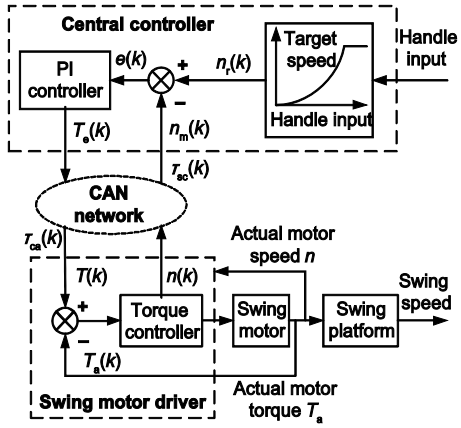

Fig. 2 Control structure of the electric swing system (PI: proportional–integral; CAN: controller area network)

From the above control structure, it can be seen that the network-induced delays lead the target motor torques (or the actual motor speeds) in different CAN nodes to be different. The correlations are as follows:

$$n_m(k)=n(k-\tau_{sc}(k)), \tag{1}$$

$$T(k)=T_e(k-\tau_{ca}(k)). \tag{2}$$

In closed-loop control, the actual motor speed and target motor torque are periodically exchanged. The update rate of the feedback motor speed is equal to that in the control period. The central controller is event-triggered to receive the feedback motor speed and time-triggered to generate the target motor torque. The swing motor driver is event-triggered to implement the received target motor torque and time-triggered to send the actual motor speed. The CAN messages that carry the target motor torque and the actual motor speed are sent in the overwriting mode. Based on the above communication mechanism, the delay ranges of the signals that are successfully transmitted can be inferred as follows:

$$\tau_{sc}(k)\in(0,2t_c), \tag{3}$$

$$\tau_{ca}(k)\in(0,t_c), \tag{4}$$

where t_c denotes the control period. Note that the delay denotes the time between the signal generation and its usage.

In ESS, the update rate of the feedback motor speed and target motor torque is limited because of the high bus load and the limited bandwidth. The highest update rate is designed to be 100 Hz. To achieve satisfactory control performance, the delays have to be guaranteed small. Large delay and packet dropout are unacceptable since they will have major impacts on the control performance of ESS.

2.2 Effects of communication errors

According to the CAN protocol (ISO, 2015), if a communication error occurs when the CAN bus is busy, the current transmission will be interrupted and retransmitted. Hence, the delays will increase under communication errors. If the delays become larger than the corresponding message periods, packet dropouts will appear.

In the feedback channel, if the feedback signal is lost, the newest feedback motor speed in the central controller is the one received in the past:

$$n_m(k)=n_m(k-\lambda_1)=n((k-\lambda_1)t_c-\tau_{sc}(k-\lambda_1)). \tag{5}$$

The corresponding target motor torque generated by the central controller is

$$\begin{aligned}T_e(k)&=[n_r(k)-n_m(k)](k_p+k_1t_c)+R_e(k)\\&=[n_r(k)-n(k)](k_p+k_1t_c)+R_e^*(k)+\varepsilon(k),\end{aligned} \tag{6}$$

where

$$\begin{aligned} R_{\mathrm{e}}(k) &= R_{\mathrm{e}}(k-1)+[n_{\mathrm{r}}(k-1)-n_{\mathrm{m}}(k-1)]k_{\mathrm{I}}t_{\mathrm{c}},\\ R_{\mathrm{e}}^{*}(k) &= R_{\mathrm{e}}^{*}(k-1)+[n_{\mathrm{r}}(k-1)-n(k-1)]k_{\mathrm{I}}t_{\mathrm{c}},\\ \varepsilon(k) &= [n(k)-n((k-\lambda_{1})t_{\mathrm{c}}-\tau_{\mathrm{sc}}(k-\lambda_{1}))]\\ &\quad\cdot(k_{\mathrm{p}}+k_{\mathrm{I}}t_{\mathrm{c}})+\sum_{i=0}^{k-1}[n(i)-n_{\mathrm{m}}(i)]k_{\mathrm{I}}t_{\mathrm{c}}, \end{aligned}$$

where λ_1 is the number of consecutive packet dropouts of the feedback motor speed, $R_{\mathrm{e}}(k)$ and $R_{\mathrm{e}}^{*}(k)$ are the actual and expected cumulative integration before the kth control period on the central controller side, respectively, k_{p} and k_{I} are the proportional gain and integral gain of the PI controller, respectively, and $\varepsilon(k)$ denotes the error of the generated target motor torque.

In the forward channel, if the target motor torque is not lost, the newest target motor torque received by the swing motor driver is

$$T(k)=\begin{cases} T_{\mathrm{e}}(k-1), & kt_{\mathrm{c}}\le t<kt_{\mathrm{c}}+\tau_{\mathrm{ca}}(k),\\ T_{\mathrm{e}}(k), & kt_{\mathrm{c}}+\tau_{\mathrm{ca}}(k)\le t<(k+1)t_{\mathrm{c}}. \end{cases} \tag{7}$$

If the target motor torque is lost, the newest target motor torque received by the swing motor driver is the same as that received in the past:

$$\begin{aligned} T(k) &= T(k-\lambda_{2})\big|_{(k-\lambda_{2})t_{\mathrm{c}}+\tau_{\mathrm{ca}}(k)\le t<(k-\lambda_{2}+1)t_{\mathrm{c}}}\\ &= T_{\mathrm{e}}(k-\lambda_{2}), \end{aligned} \tag{8}$$

where λ_2 is the number of consecutive packet dropouts of the target motor torque.

From Eq. (6), it can be seen that the communication errors in the feedback channel lead the generated target motor torque to be inaccurate, owing to the inaccurate motor speed used in the calculation. In addition to the current packet dropout, delays and packet dropouts that occur in the previous control periods affect the error $\varepsilon(k)$. From Eqs. (7) and (8), the communication errors in the forward channel lead the control action to be inaccurate, owing to the mismatch of control commands. Hence, the adverse impacts of communication errors can be reduced by correcting the feedback motor speed and the control action.

2.3 Plant model

In ESS, the motor speed is related to the dynamics of the plant which consists of the swing motor and its driver, and the swing mechanism. Because of the pose variation of the swing platform, the plant dynamics is time-varying. Therefore, the plant model should be identified online for correcting the feedback motor speed. The plant is a nonlinear system and many factors will affect its model parameters. To simplify the analysis and control design, we model only the main dynamic characteristics of the plant. The negligibility of the neglected dynamics will be examined by practical experiments.

As shown in Fig. 2, the main dynamics of the plant contain the response of motor torque and the dynamics of the swing mechanism which consists of the motor shaft, the reduction gear, and the swing platform. Compared to the swing mechanism, the response of motor torque is very fast since it is equivalent to the response of motor phase current. Therefore, a first-order model is used to describe it:

$$G_{\mathrm{m}}(s)=\frac{T_{\mathrm{a}}(s)}{T(s)}=\frac{1}{\tau_{\mathrm{m}}s+1}, \tag{9}$$

where τ_{m} is the time constant of the torque response.

For swing mechanism, we regard the motor shaft, reduction gear, and swing platform as a whole, which is equivalent to a rotating rigid body with variable inertia. Then we model its dynamics as follows:

$$G_{\mathrm{p}}(s)=\frac{n(s)}{T_{\mathrm{a}}(s)}=\frac{30K_{\mathrm{r}}/\pi}{J_{\mathrm{p}}s+b_{\mathrm{p}}}, \tag{10}$$

where K_{r} is the reduction gear ratio, and J_{p} and b_{p} are the equivalent rotary inertia and equivalent damping coefficient of the swing mechanism, respectively.

Based on the above analysis, the plant model is

$$G(s)=\frac{30K_{\mathrm{r}}/\pi}{J_{\mathrm{p}}\tau_{\mathrm{m}}s^{2}+(b_{\mathrm{p}}\tau_{\mathrm{m}}+J_{\mathrm{p}})s+b_{\mathrm{p}}}. \tag{11}$$

From Eq. (11), it can be seen that the plant can be regarded as a second-order system where the model parameters vary with the pose variation of the swing platform.

3 Distributed fault-tolerant strategy

From the analysis in Section 2, the main idea of DCS is as follows:

1. making the feedback motor speed $n_m(k)$ as close as possible to the real-time actual motor speed $n(k)$;

2. making the control action of the implemented target motor torque $T(k)$ as close as possible to that of the generated target motor torque $T_e(k)$.

Based on the above DCS, the fault-tolerant strategy is developed as shown in Fig. 3. The main idea is to adaptively compensate for the feedback motor speed and the target motor torque in the central controller and the swing motor driver by using the 'feedback compensator' and 'forward compensator', respectively. The compensations are carried out according to the real-time delay and packet dropout detected by the 'delay' and 'packet dropout detector' based on the online delay estimation method proposed by Gao *et al.* (2015). The real-time plant model parameters required by the 'feedback compensator' are identified online by the 'model identifier' based on the recursive least-squares algorithm with a forgetting factor (FFRLS algorithm) (Beza and Bongiorno, 2014). The details of the proposed strategy are explained below.

3.1 Identification of model parameters

The plant model parameters change with the pose variation of the platform. To improve feedback compensation, the plant model parameters should be identified online since the current motor speed is estimated based on the plant model. In addition, since the PI controller is discrete, the discrete-time model equivalent to the plant model given by Eq. (11) is required:

$$G(z)=\frac{b_0+b_1z^{-1}+b_2z^{-2}}{1+a_1z^{-1}+a_2z^{-2}}. \tag{12}$$

The model parameters a_1, a_2, b_0, b_1, and b_2 are identified online by using the FFRLS algorithm:

$$\begin{cases}\hat{\boldsymbol{\theta}}(k)=\hat{\boldsymbol{\theta}}(k-1)+\boldsymbol{K}(k)[n_m(k)\\ \qquad -\boldsymbol{H}^{\mathrm{T}}(k-1)\hat{\boldsymbol{\theta}}(k-1)],\\ \boldsymbol{K}(k)=\boldsymbol{P}(k-1)\boldsymbol{H}(k-1)[\delta(t)\\ \qquad +\boldsymbol{H}^{\mathrm{T}}(k-1)\boldsymbol{P}(k-1)\boldsymbol{H}(k-1)]^{-1},\\ \boldsymbol{P}(k)=\dfrac{1}{\delta(t)}\left[\boldsymbol{I}-\boldsymbol{K}(k)\boldsymbol{H}^{\mathrm{T}}(k-1)\right]\\ \qquad \cdot\boldsymbol{P}(k-1),\end{cases} \tag{13}$$

where $\boldsymbol{\theta}=[a_1, a_2, b_0, b_1, b_2]^{\mathrm{T}}$, $0<\delta\le1$, and

$$\boldsymbol{H}(k)=[-n_m(k-1),-n_m(k-2),\\ T_e(k),T_e(k-1),T_e(k-2)]^{\mathrm{T}},$$

where $\boldsymbol{P}$ is a covariance matrix, $\boldsymbol{\theta}$ is the model parameter vector needed to be identified, $\boldsymbol{H}$ is a vector that consists of the observation data, and $\delta(t)$ is the forgetting factor.

Since the rotary inertia usually abruptly varies in the start and stop stages, but relatively slowly in the swing stage, the forgetting factor during the start and stop stages can be set larger than that in the swing stage to improve the identification accuracy. Therefore, the forgetting factor is determined according to the variation of motor speed:

$$\delta(t)=\begin{cases}\delta_1, & \Delta n_m(k)<\xi,\\ \delta_2, & \Delta n_m(k)\ge\xi,\end{cases} \tag{14}$$

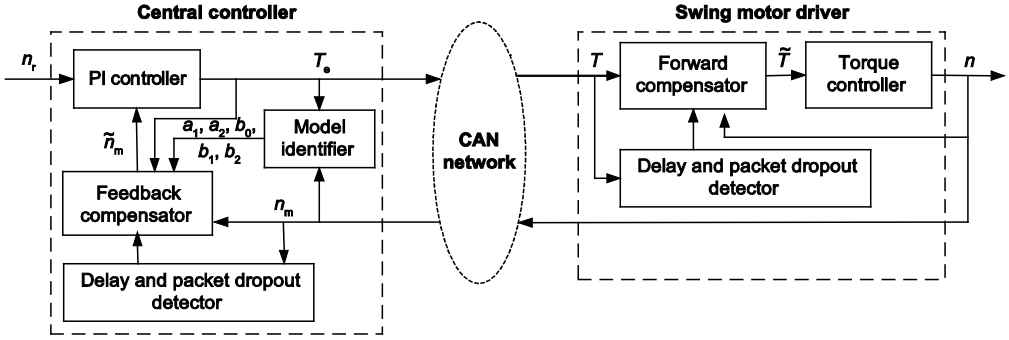

Fig. 3 Distributed fault-tolerant strategy for the electric swing system (PI: proportional–integral; CAN: controller area network)

where $\Delta n_{\mathrm{m}}(k)=n_{\mathrm{m}}(k)-n_{\mathrm{m}}(k-1)$ and $\delta_1>\delta_2$.

The threshold ξ and the value of $\delta(t)$, denoted by δ_1 and δ_2, respectively, are empirically determined.

3.2 Detection of delays and packet dropouts

In previous work, we have studied the online acquisition of communication delays in CAN (Gao *et al.*, 2015). However, since it covers only the cases where no communication error occurs, in this study, we extend this approach to the cases with communication errors. The details of the extension are as follows.

As presented in our previous work (Gao *et al.*, 2015), the message response time (MRT), which is the delay induced by CAN, can be obtained in two ways. The first way is to estimate MRT by using the 'reference message instance', which is a message instance satisfying certain conditions. In this way, the result may be inaccurate when communication errors occur. The second way is to estimate MRT by using 'incremental estimation'. In this way, a relatively accurate result can be obtained under communication errors unless the 'incremental estimation' is consecutively used for a long time. Hence, the MRT of the message instance that carries the feedback motor speed or the target motor torque in the kth control period can be estimated as follows:

$$R_{\mathrm{m}}(k)=\begin{cases}R_{\mathrm{m}_0}(k), & |R_{\mathrm{m}_0}(k)-R_{\mathrm{m}_1}(k)|<\varepsilon,\\ R_{\mathrm{m}_1}(k), & |R_{\mathrm{m}_0}(k)-R_{\mathrm{m}_1}(k)|\geq\varepsilon,\end{cases} \tag{15}$$

where $R_{\mathrm{m}_0}(k)$ and $R_{\mathrm{m}_1}(k)$ are the MRTs estimated in the first way and the second way, respectively, and ε is a threshold empirically determined in advance.

After obtaining MRT, the communication delay of the message instance can be derived as

$$\tau(k)=R_{\mathrm{m}}(k)+\tau_{\mathrm{c}}(k), \tag{16}$$

where $\tau_{\mathrm{c}}(k)$ denotes the computation delay in the central controller or the swing motor driver.

Based on the obtained delay and the recorded receiving timestamp, the start moment of the message instance is

$$t_{\mathrm{s}}(k)=t_{\mathrm{r}}(k)-\tau(k), \tag{17}$$

where $t_{\mathrm{s}}(k)$ and $t_{\mathrm{r}}(k)$ are the start moment and the receiving timestamp, respectively.

Then, based on the obtained start moment, the deadline of the next message instance that carries the feedback motor speed or the target motor torque is

$$t_{\mathrm{d}}(k+1)=t_{\mathrm{s}}(k)+2t_{\mathrm{c}}. \tag{18}$$

If the next message instance has not been received before $t_{\mathrm{d}}(k+1)$, it can be asserted that the message instance is lost. However, by using this criterion, the packet dropout can hardly be compensated for, since the impacts have already been caused when the packet dropout is detected. Therefore, we use a more conservative criterion for the detection of packet dropout by modifying the deadline as

$$t'_{\mathrm{d}}(k+1)=t_{\mathrm{s}}(k)+1.5t_{\mathrm{c}}. \tag{19}$$

Since the compensation of the delay larger than half the control period is similar to that of the packet dropout, the conservative criterion is acceptable.

3.3 Feedback compensation

According to the model given by Eq. (12), the actual motor speed in the kth control period is

$$\begin{aligned}n(k)=&-a_1n(k-1)-a_2n(k-2)\\&+b_0T(k)+b_1T(k-1)+b_2T(k-2)+\omega(k),\end{aligned} \tag{20}$$

where $\omega(k)$ denotes the error caused by the model error. From Eq. (20), it can be seen that the current motor speed can be estimated according to the past motor speed and the past target motor torque if the model error is not considered. Based on this, the feedback compensation is designed as the following steps:

1. Estimate the current motor speed:

$$\begin{aligned}\hat{n}(k)=&-a_1\tilde{n}_{\mathrm{m}}(k-1)-a_2\tilde{n}_{\mathrm{m}}(k-2)\\&+b_0T_{\mathrm{e}}(k)+b_1T_{\mathrm{e}}(k-1)+b_2T_{\mathrm{e}}(k-2),\end{aligned} \tag{21}$$

where $\tilde{n}_{\mathrm{m}}(k-i)$ (i=1, 2) is the previous compensated value.

2. Determine the compensated value of the feedback motor speed used in the calculation of target motor torque:

$$\tilde{n}_{\mathrm{m}}(k)=\begin{cases} n_{\mathrm{m}}(k), & \tau_{\mathrm{sc}}(k)<0.5t_{\mathrm{c}}, \\ \hat{n}(k), & \text{otherwise.} \end{cases} \tag{22}$$

3. Reverse the correction

Because of the model error, the estimation error caused by Eq. (21) is not negligible. To avoid the cumulative estimation error, we introduce a reverse correction to the past estimated values when receiving a new feedback motor speed. Assuming that a new feedback motor speed is received in the (k+1)th control period, the correction is as follows:

If $\tau_{\mathrm{sc}}(k+1)<0.5t_{\mathrm{c}}$,

$$\begin{aligned} &\tilde{n}_{\mathrm{m}}(k+1)=\hat{n}(k+1)=n_{\mathrm{m}}(k+1), \\ \rightarrow\ &\hat{n}(k)=\hat{n}(k)+h[n_{\mathrm{m}}(k+1)-(-a_1\tilde{n}_{\mathrm{m}}(k)-a_2\tilde{n}_{\mathrm{m}}(k-1) \\ &\quad +b_0T_{\mathrm{e}}(k+1)+b_1T_{\mathrm{e}}(k)+b_2T_{\mathrm{e}}(k-1))], \\ \rightarrow\ &\tilde{n}_{\mathrm{m}}(k)=\begin{cases} n_{\mathrm{m}}(k), & \tau_{\mathrm{sc}}(k)<0.5t_{\mathrm{c}}, \\ \hat{n}(k), & \text{otherwise.} \end{cases} \end{aligned}$$

If $0.5t_{\mathrm{c}}\le\tau_{\mathrm{sc}}(k+1)<1.5t_{\mathrm{c}}$,

$$\begin{aligned} &\tilde{n}_{\mathrm{m}}(k)=\hat{n}(k)=n_{\mathrm{m}}(k+1), \\ \rightarrow\ &\hat{n}(k-1)=\hat{n}(k-1)+h[n_{\mathrm{m}}(k+1)-(-a_1\tilde{n}_{\mathrm{m}}(k-1) \\ &\quad -a_2\tilde{n}_{\mathrm{m}}(k-2)+b_0T_{\mathrm{e}}(k)+b_1T_{\mathrm{e}}(k-1) \\ &\quad +b_2T_{\mathrm{e}}(k-2))], \\ \rightarrow\ &\tilde{n}_{\mathrm{m}}(k-1)=\begin{cases} n_{\mathrm{m}}(k-1), & \tau_{\mathrm{sc}}(k-1)<0.5t_{\mathrm{c}}, \\ \hat{n}(k-1), & \text{otherwise.} \end{cases} \\ \rightarrow\ &\tilde{n}_{\mathrm{m}}(k+1)=\hat{n}(k+1) \\ &\quad =-a_1\tilde{n}_{\mathrm{m}}(k)-a_2\tilde{n}_{\mathrm{m}}(k-1)+b_0T_{\mathrm{e}}(k+1) \\ &\quad +b_1T_{\mathrm{e}}(k)+b_2T_{\mathrm{e}}(k-1). \end{aligned}$$

If $1.5t_{\mathrm{c}}\le\tau_{\mathrm{sc}}(k+1)<2t_{\mathrm{c}}$,

$$\begin{aligned} &\tilde{n}_{\mathrm{m}}(k-1)=\hat{n}(k-1)=n_{\mathrm{m}}(k+1), \\ \rightarrow\ &\hat{n}(k-2)=\hat{n}(k-2)+h[n_{\mathrm{m}}(k+1)-(-a_1\tilde{n}_{\mathrm{m}}(k-2) \\ &\quad -a_2\tilde{n}_{\mathrm{m}}(k-3)+b_0T_{\mathrm{e}}(k-1)+b_1T_{\mathrm{e}}(k-2) \\ &\quad +b_2T_{\mathrm{e}}(k-3))], \\ \rightarrow\ &\tilde{n}_{\mathrm{m}}(k-2)=\begin{cases} n_{\mathrm{m}}(k-2), & \tau_{\mathrm{sc}}(k-2)<0.5t_{\mathrm{c}}, \\ \hat{n}(k-2), & \text{otherwise,} \end{cases} \\ \rightarrow\ &\tilde{n}_{\mathrm{m}}(k)=\hat{n}(k) \\ &\quad =-a_1\tilde{n}_{\mathrm{m}}(k-1)-a_2\tilde{n}_{\mathrm{m}}(k-2)+b_0T_{\mathrm{e}}(k) \\ &\quad +b_1T_{\mathrm{e}}(k-1)+b_2T_{\mathrm{e}}(k-2), \\ \rightarrow\ &\tilde{n}_{\mathrm{m}}(k+1)=\hat{n}(k+1) \\ &\quad =-a_1\tilde{n}_{\mathrm{m}}(k)-a_2\tilde{n}_{\mathrm{m}}(k-1)+b_0T_{\mathrm{e}}(k+1) \\ &\quad +b_1T_{\mathrm{e}}(k)+b_2T_{\mathrm{e}}(k-1), \end{aligned}$$

where h is the correction coefficient. It can be simply taken as 1 or optimally determined by simulations.

Note that we use the target motor torque generated by the central controller instead of that implemented by the swing motor driver in feedback compensation. Hence, the compensation accuracy will be affected by that of forward compensation.

3.4 Forward compensation

According to the PI control law, the increment of the target motor torque generated by the central controller is as follows:

$$\begin{aligned} \Delta T_{\mathrm{e}}(k)&=T_{\mathrm{e}}(k)-T_{\mathrm{e}}(k-1) \\ &=[n_{\mathrm{r}}(k)-n_{\mathrm{r}}(k-1)+n_{\mathrm{m}}(k-1) \\ &\quad -n_{\mathrm{m}}(k)](k_{\mathrm{p}}+k_1t_{\mathrm{c}})+\Delta R_{\mathrm{e}}(k), \end{aligned} \tag{23}$$

where

$$\begin{aligned} \Delta R_{\mathrm{e}}(k)&=R_{\mathrm{e}}(k)-R_{\mathrm{e}}(k-1) \\ &=[T_{\mathrm{e}}(k-1)-R_{\mathrm{e}}(k-1)]k_1t_{\mathrm{c}}/(k_{\mathrm{p}}+k_1t_{\mathrm{c}}). \end{aligned}$$

From Eq. (23), it can be inferred that the lost target motor torque may be obtained by the swing motor driver based on the previously received one. Hence, forward compensation can be designed based on Eq. (23). However, since the reference speed is unknown on the swing motor driver side, Eq. (23) cannot be directly used in forward compensation. Therefore, we assume that the reference speed does not change during the packet dropouts, that is, $n_{\mathrm{r}}(k)=n_{\mathrm{r}}(k-1)$. If the number of consecutive packet dropouts is not very large, this assumption is appropriate since the handle input changes slowly compared to the control period, which is 10 ms. Based on the above assumption, the forward compensation is designed as follows:

When the data packet is not lost, the compensated value of the target motor torque is as follows:

If $0<\tau_{\mathrm{ca}}(k)<0.5t_{\mathrm{c}}$,

$$\tilde{T}(k)=T(k). \tag{24}$$

If $0.5t_{\mathrm{c}}<\tau_{\mathrm{ca}}(k)<5t_{\mathrm{c}}$,

$$\begin{aligned} \tilde{T}(k)&=T(k)+[n(k-1)-n(k)](k_{\mathrm{p}}+k_1t_{\mathrm{c}}) \\ &\quad +\Delta R(k), \end{aligned} \tag{25}$$

where

$$\Delta R(k) = R(k) - R(k-1) = [T(k) - R(k-1)]k_1 t_c / (k_p + k_1 t_c),$$

where $R(k)$ is the cumulative integration before the kth control period on the swing motor driver side.

When the data packet is lost, the compensated value of the target motor torque is

$$\tilde{T}(k) = \tilde{T}(k-1) + [n(k-1) - n(k)] \cdot (k_p + k_1 t_c) + \Delta R(k), \quad (26)$$

with

$$\Delta R(k) = R(k) - R(k-1) = [\tilde{T}(k-1) - R(k-1)]k_1 t_c / (k_p + k_1 t_c).$$

It is worth noting that the above forward compensation is also an approximate compensation although it is adequate.

4 Simulation studies

Simulation studies based on MATLAB/Simulink are conducted to validate the proposed DCS. First, the simulation model is presented. Then, the effectiveness of the proposed DCS for different delays and packet dropouts is examined. Whether the model error will degrade the effectiveness of the proposed DCS is also studied.

4.1 Simulation model

The simulation model is shown in Fig. 4. The reference speed is generated by the 'signal input'. Since ESS executes the start–turn–stop action most of the time, the step-function signal with an amplitude of 500 r/min is used as the reference speed. The PI controller is realized by the 'controller', where feedback compensation is also realized. Forward compensation is realized by the 'compensator'. The compensation algorithm can be inactivated or activated. The delays and packet dropouts are generated by DG_f, DPG_f, DG_fd, and DPG_fd, respectively. In different simulations, the delays and the number of consecutive packet dropouts are fixed or set to be random. Unless specifically noted, the delays and packet dropouts in these two channels are set to be the same. The plant of ESS is imitated by the 'ESS model'. The plant model parameters are identified by a test in which the arm cylinder, boom cylinder, and bucket cylinder of the hybrid excavator all stretch out to the end. The identified parameters are given in set 1 of Table 2. The equivalent continuous-time model is

$$G(s) = \frac{-0.0382s^2 - 3.796s + 768.8}{s^2 - 70.13s + 5.403}. \quad (27)$$

The control cycle is t_c=0.01, and the correction coefficient is h=1.

To test whether the model error will degrade the effectiveness of the proposed DCS, another parameter set, given in set 2 of Table 2, is identified from

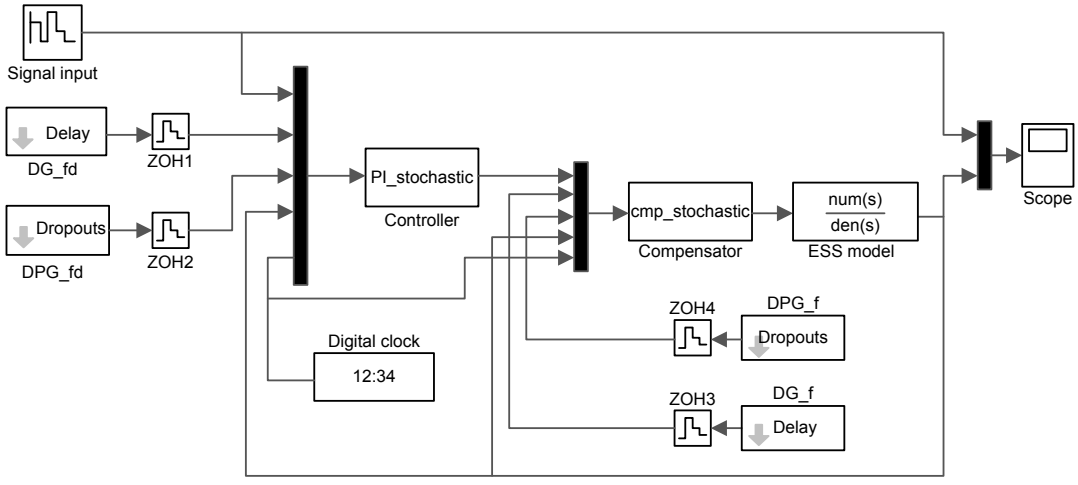

Fig. 4 Simulation model used to validate the proposed delay compensation scheme for the electric swing system (ESS)

Table 2 Identified model parameters of the plant

Set	a_1	a_2	b_0	b_1	b_2
1	–1.4804	0.4808	–0.0281	0.0850	0
2	–1.5804	0.5813	–0.0633	0.1036	0

another gesture where the arm cylinder retracts half of the trip from the gesture used to identify set 1. By using the plant model determined by set 2 in the feedback compensation and using that determined by set 1 in the ESS model, the model error is imitated. The four settings that will be used are given in Table 3. The 'normal condition' used in the following indicates the condition without delays or packet dropouts. The performance under the normal condition is the expected performance.

Table 3 Simulation settings

Setting	Delays and packet dropouts	Status of compensation algorithm	Model used in compensation	Model used in the ESS model
A	Fixed	Inactive	Set 1	Set 1
B	Fixed	Active	Set 1	Set 1
C	Fixed	Active	Set 2	Set 1
D	Random	Active	Set 1	Set 1

ESS: electric swing system

4.2 Evaluation of compensation effects

First, the model is set with setting *A* to examine the impacts of the delays and packet dropouts on ESS. One of the simulation results is given in Fig. 5, from which we can see that both the delays and the packet dropouts will increase the speed overshoot and decrease the speed response time. Since the overshoot and response time can reflect the main performance of interest (operability and comfort), we use them as evaluation indexes in the following studies. The overshoot and response time under the normal condition are 0 and 411 ms, respectively.

The overshoots and response time under different delays and consecutive packet dropouts are given in Figs. 6a and 6b, respectively, from which we can see that the overshoot will increase as the number of consecutive packet dropouts and delay increase, and the response time will increase as the number of consecutive packet dropouts increases and will decrease as the delay increases. From the aspect of overshoot and response time, the greater the difference between the actual value and the expected value, the more serious the performance decline. Hence, the proposed strategy can be claimed to be effective only if the overshoot and response time with compensation become closer to the expected values than those without compensation.

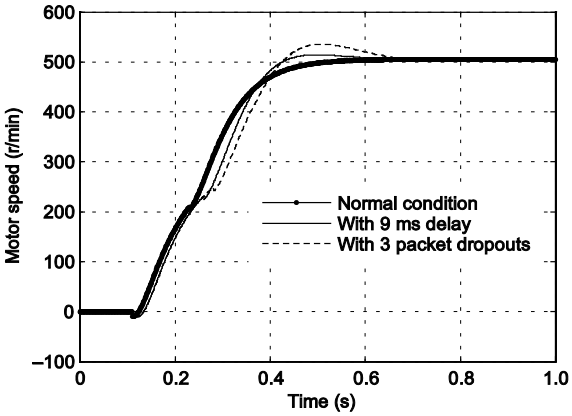

Fig. 5 Effects of delays and packet dropouts on the electric swing system

Second, to evaluate the compensation effects of DCS for varying degrees of delay and packet dropout, the model is set with setting *B*. The overshoots and response time with compensation are shown in Figs. 6c and 6d, respectively. By comparing Fig. 6c with Fig. 6a, we can see that the overshoots under different delays and packet dropouts are effectively suppressed to <2% by the proposed DCS. By comparing Fig. 6d with Fig. 6b, it can be seen that the response time with compensation becomes closer to the expected value (411 ms) if the number of consecutive packet dropouts is not large. However, if the number of consecutive packet dropouts exceeds a threshold (e.g., the five packets in Fig. 6d), the compensation becomes excessive and leads ESS to respond more slowly than in the normal condition.

Third, we set the model with setting *C* to check whether the effectiveness of the proposed DCS is affected by the model error since the online identified plant model has some errors. The results are given in Figs. 6e and 6f. By comparing Figs. 6e and 6f with Figs. 6c and 6d, respectively, we can see that the compensation effects with model error are similar to those without model error except that the threshold, above which the compensation effects degrade, is lightly affected. Hence, the proposed DCS is tolerable to the model error.

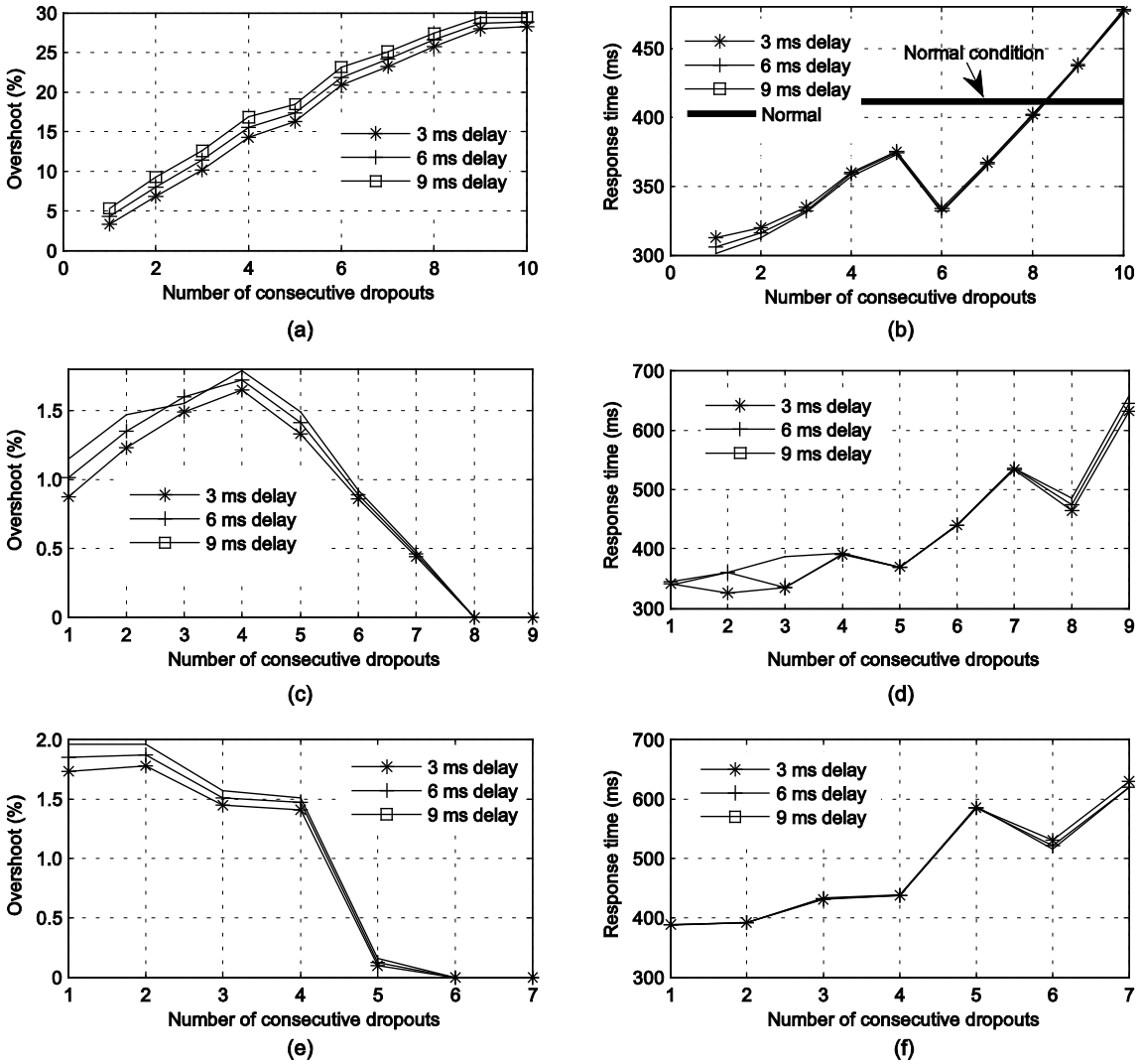

Fig. 6 Overshoots (a) and response time (b) without compensation, overshoots (c) and response time (d) with compensation and no model error, and overshoots (e) and response time (f) with compensation and model error under different delays and consecutive packet dropouts

Finally, the model is set with setting D to validate the proposed DCS to more general communication errors. The random delays and packet dropouts used in the simulation are given in Fig. 7. From the simulation results given in Fig. 8, we can see that the proposed DCS is effective for random communication errors.

Generally, the proposed DCS can improve the ESS's control performance under random communication errors if the number of consecutive packet dropouts is not large. The improvement will degrade when the number of consecutive packet dropouts exceeds a threshold. The model error has little impact on the compensation effects.

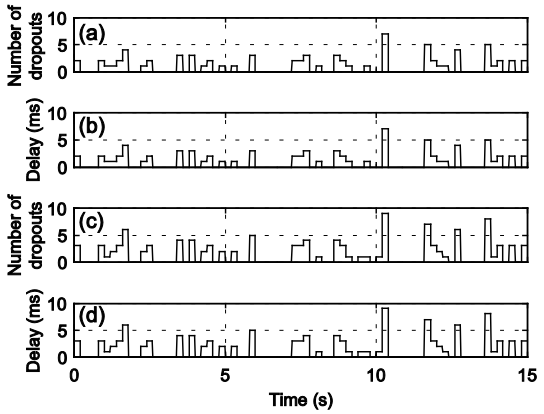

Fig. 7 Packet dropouts (a) and delays (b) in the forward channel, and packet dropouts (c) and delays (d) in the feedback channel

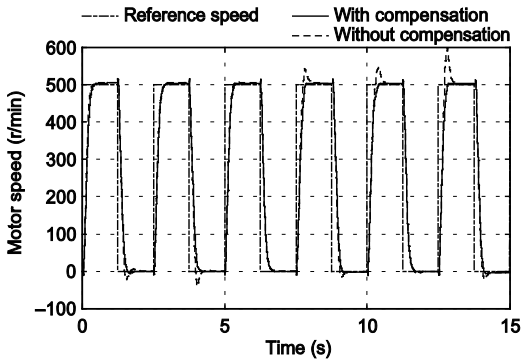

Fig. 8 Compensation effects of the proposed delay compensation scheme for random delays and packet dropouts

5 Experiments and results

In this section, experiments are carried out to validate the proposed fault-tolerant strategy to the actual ESS since nonlinearity and some detailed dynamic characteristics of ESS are neglected in previous simulations.

5.1 Experimental setup

Fig. 9 gives the hybrid excavator used in the experiments. The CAN, based on which ESS is controlled, has six nodes. The messages that carry the target motor torque and feedback motor speed are both sent with a period of 10 ms. The communication errors are imitated by selectively sending or receiving message packets. The numbers of consecutive packet dropouts in the forward and feedback channels are set to be one and two in all tests, respectively.

Fig. 9 Hybrid excavator with the electric swing system

5.2 Experimental results

The experimental results of a step response test are shown in Fig. 10 and Table 4, from which we can see that the overshoot without compensation (19.8%) is much larger than that in the normal condition (5.0%), and the overshoot is reduced to 8.4% by applying the proposed fault-tolerant strategy. The results agree well with the simulation results.

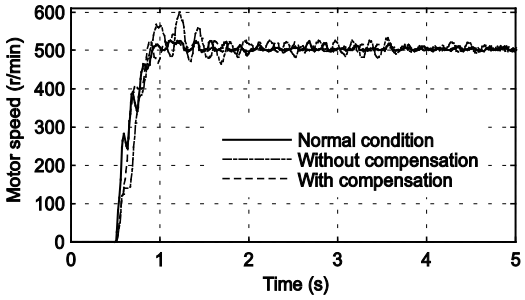

Fig. 10 Step response of the electric swing system with two packet dropouts in the feedback channel and one packet dropout in the forward channel

Table 4 Overshoot of step response in different scenarios

Scenario	Overshoot (%)
Normal condition	5.0
Without compensation	19.8
With compensation	8.4

Fig. 11 shows the speed response of ESS during an arbitrary operation. As can be seen from Fig. 11, if the packet dropouts are not compensated, the motor speed fluctuates obviously during the operation, especially during the period between 3.8 and 6.0 s. During the operation, the operator can obviously feel the wobbles of the swing platform. The wobbles not only degrade the comfort but also may cause damage to the motor shaft. Nevertheless, the wobbles can be effectively suppressed by the proposed strategy so that the

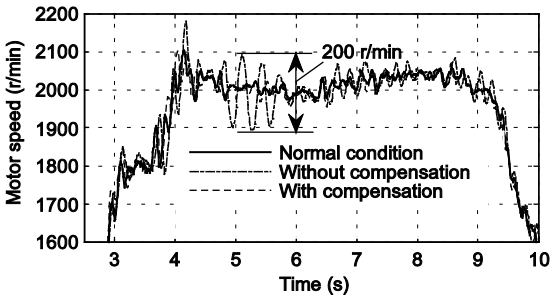

Fig. 11 Speed response of the electric swing system during an arbitrary operation

swing motion performance is almost the same as that in the normal condition. Therefore, the proposed fault-tolerant strategy can help ESS perform well under communication errors.

6 Conclusions

In this paper, we proposed a distributed fault-tolerant strategy to eliminate the adverse impacts of communication errors for the ESS of a hybrid excavator. A novel DCS, where the feedback signal and the control command of the closed control loop are independently compensated for in the central controller and the swing motor driver, respectively, was introduced, so that the communication performance of CAN would not be degraded further by the proposed strategy. By applying DCS, the adverse impacts of the delays and packet dropouts caused by communication errors could be adaptively eliminated. To improve feedback compensation, the FFRLS algorithm was employed to identify the plant model parameters online and a reverse correction law was developed. By conducting simulations and practical experiments, the proposed strategy was verified to be effective for ESS. Future work includes improving the fault-tolerant ability of the proposed strategy, studying the system stability when using the proposed strategy, and extending the proposed DCS to other NCSs.

References

Beza, M., Bongiorno, M., 2014. Application of recursive least squares algorithm with variable forgetting factor for frequency component estimation in a generic input signal. *IEEE Trans. Ind. Appl.*, **50**(2):1168-1176. http://dx.doi.org/10.1109/TIA.2013.2279195

Du, Z., Yue, D., Hu, S., 2014. H-infinity stabilization for singular networked cascade control systems with state delay and disturbance. *IEEE Trans. Ind. Inform.*, **10**(2):882-894. http://dx.doi.org/10.1109/TII.2013.2294114

Gao, D., Wang, Q., 2014. Health monitoring of controller area network in hybrid excavator based on the message response time. IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics, p.1634-1639. http://dx.doi.org/10.1109/AIM.2014.6878318

Gao, D., Wang, Q., Lei, Y., *et al.*, 2015. Online real-time estimation of response time for periodic messages in controller area networks. *Math. Prob. Eng.*, **2015**:1-13. http://dx.doi.org/10.1155/2015/659623

Gupta, R.A., Chow, M.Y., 2010. Networked control system: overview and research trends. *IEEE Trans. Ind. Electron.*, **57**(7):2527-2535. http://dx.doi.org/10.1109/TIE.2009.2035462

ISO, 2015. Road Vehicles—Controller Area Network (CAN)—Part 1: Data Link Layer and Physical Signalling, ISO 11898-1:2015. International Organization for Standardization, Geneva.

Jin, K., Park, T., Lee, H., 2012. A control method to suppress the swing vibration of a hybrid excavator using sliding mode approach. *Proc. Inst. Mech. Eng. C*, **226**(5): 1237-1253. http://dx.doi.org/10.1177/0954406211421260

Kagoshima, M., Komiyama, M., Nanjo, T., *et al.*, 2007. Development of new hybrid excavator. *Kobelco Technol. Rev.*, **27**:39-42.

Katrasnik, T., 2007. Hybridization of powertrain and downsizing of IC ICE—a way to reduce fuel consumption and pollutant emissions—Part 1. *Energy Conv. Manag.*, **48**(5):1411-1423. http://dx.doi.org/10.1016/j.enconman.2006.12.004

Kim, H., Choi, J., Yi, K., 2012. Development of supervisory control strategy for optimized fuel consumption of the compound hybrid excavator. *Proc. Inst. Mech. Eng. D*, **226**(12):1652-1666. http://dx.doi.org/10.1177/0954407012447019

Kim, J.Y., 2014. Anti-Rebounding Control Apparatus and Method in an Electrical Swing System of a Hybrid Excavator. E.P. Patent 2 690 224 A1.

Kruszewski, A., Jiang, W.J., Fridman, E., *et al.*, 2012. A switched system approach to exponential stabilization through communication network. *IEEE Trans. Contr. Syst. Technol.*, **20**(4):887-900. http://dx.doi.org/10.1109/TCST.2011.2159793

Kwon, T., Lee, S., Sul, S., *et al.*, 2010. Power control algorithm for hybrid excavator with super capacitor. *IEEE Trans. Ind. Appl.*, **46**(4):1447-1455. http://dx.doi.org/10.1109/08IAS.2008.193

Lei, Y., Yuan, Y., Zhao, J., 2014. Model-based detection and monitoring of the intermittent connections for CAN networks. *IEEE Trans. Ind. Electron.*, **61**(6):2912-2921. http://dx.doi.org/10.1109/TIE.2013.2272277

Lin, T., Wang, Q., Hu, B., *et al.*, 2010a. Development of hybrid powered hydraulic construction machinery. *Autom. Constr.*, **19**(1):11-19. http://dx.doi.org/10.1016/j.autcon.2009.09.005

Lin, T., Wang, Q., Hu, B., *et al.*, 2010b. Research on the energy regeneration systems for hybrid hydraulic excavators. *Autom. Constr.*, **19**(8):1016-1026. http://dx.doi.org/10.1016/j.autcon.2010.08.002

Liu, G.P., Xia, Y., Chen, J., *et al*., 2007. Networked predictive control of systems with random network delays in both forward and feedback channels. *IEEE Trans. Ind. Electron.*, **54**(3):1282-1297. http://dx.doi.org/10.1109/TIE.2007.893073

Luck, R., Ray, A., 1990. An observer-based compensator for distributed delays. *Automatica*, **26**(5):903-908.

http://dx.doi.org/10.1016/0005-1098(90)90007-5

Luck, R., Ray, A., 1994. Experimental verification of a delay compensation algorithm for integrated communication and control systems. *Int. J. Contr.*, **59**(6):1357-1372. http://dx.doi.org/10.1080/00207179408923135

Nilsson, J., 1998. Real-Time Control Systems with Delays. PhD Thesis, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Rahmani, B., Markazi, A.H.D., 2013. Variable selective control method for networked control systems. *IEEE Trans. Contr. Syst. Technol.*, **21**(3):975-982. http://dx.doi.org/10.1109/TCST.2012.2194739

Shi, Y., Yu, B., 2011. Robust mixed H_2/H_∞ control of networked control systems with random time delays in both forward and backward communication links. *Automatica*, **47**(4):754-760. http://dx.doi.org/10.1016/j.automatica.2011.01.022

Shi, Y., Huang, J., Yu, B., 2013. Robust tracking control of networked control systems application to a networked DC motor. *IEEE Trans. Ind. Electron.*, **60**(12):5864-5874. http://dx.doi.org/10.1109/TIE.2012.2233692

Shuai, Z., Zhang, H., Wang, J., *et al*., 2014. Combined AFS and DYC control of four-wheel-independent-drive electric vehicles over CAN network with time-varying delays. *IEEE Trans. Veh. Technol.*, **63**(2):591-602. http://dx.doi.org/10.1109/TVT.2013.2279843

Song, H., Liu, G.P., Yu, L., 2013. Networked predictive control of uncertain systems with multiple feedback channels. *IEEE Trans. Ind. Electron.*, **60**(11):5228-5238. http://dx.doi.org/10.1109/TIE.2012.2225398

Tian, Y.C., Levy, D., 2008. Compensation for control packet dropout in networked control systems. *Inf. Sci.*, **178**(5):1263-1278. http://dx.doi.org/10.1016/j.ins.2007.10.012

Tipsuwan, Y., Chow, M.Y., 2004. Gain scheduler middleware: a methodology to enable existing controllers for networked control and teleoperation—part I: networked control. *IEEE Trans. Ind. Electron.*, **51**(6):1218-1227. http://dx.doi.org/10.1109/TIE.2004.837866

Wang, D., Guan, C., Pan, S., *et al.*, 2009. Performance analysis of hydraulic excavator powertrain hybridization. *Autom. Constr.*, **18**(3):249-257. http://dx.doi.org/10.1016/j.autcon.2008.10.001

Wang, Q., Zhang, Y., Xiao, Q., 2005. Evaluation for energy saving effect and simulation research on energy saving of hydraulic system in hybrid construction machinery. *Chin. J. Mech. Eng.*, **41**(12):35-140. http://dx.doi.org/10.3901/JME.2005.12.135

Wang, T., Wang, Q., 2014. An energy-saving pressure-compensated hydraulic system with electrical approach. *IEEE/ASME Trans. Mechatron.*, **19**(2):570-578. http://dx.doi.org/10.1109/TMECH.2013.2250296

Wang, Y., Sun, X., Wang, Z., *et al.*, 2014. Construction of Lyapunov–Krasovskii functionals for switched nonlinear systems with input delay. *Automatica*, **50**(4):1249-1253. http://dx.doi.org/10.1016/j.automatica.2014.02.029

Xiao, Q., Wang, Q., Zhang, Y., 2008. Control strategies of power system in hybrid hydraulic excavator. *Autom. Constr.*, **17**(4):361-367. http://dx.doi.org/10.1016/j.autcon.2007.05.014

Yang, R., Liu, G.P., Shi, P., *et al.*, 2014. Predictive output feedback control for networked control systems. *IEEE Trans. Ind. Electron.*, **61**(1):512-520. http://dx.doi.org/10.1109/TIE.2013.2248339

Yao, H., Wang, Q., 2015. The control strategy of improving the stability of powertrain for compound hybrid power excavator. *Proc. Inst. Mech. Eng. D J. Autom. Eng.*, **229**(5):1-15. http://dx.doi.org/10.1177/0954407015574809

Zhang, L., Gao, H., Kaynak, O., 2013. Network-induced constraints in networked control systems—a survey. *IEEE Trans. Ind. Inf.*, **9**(1):403-416. http://dx.doi.org/10.1109/TII.2012.2219540