

Validation of a Seamless Development Process for Real-time ECUs using OSEK-OS Based SILS/RCP

Minchae Lee, Jaehyun Han, Jooyoung Ma,
Jeamyoungh Youn and Myoungcho Sunwoo
Hanyang University

Copyright © 2008 SAE International

ABSTRACT

An efficient development environments such as Software-in-the-Loop Simulation (SILS) and Rapid Control Prototyping (RCP) have been widely used to reduce the development time and cost of real-time ECUs. However, conventional SILS does not consider temporal behaviors caused by computation time, task scheduling, network-induced delays, and so on. As a result, the control performance of ECU is likely to be degraded after implementation. To overcome this problem, SILS/RCP which considers the temporal behaviors was suggested in the previous research. In this study, we validated the proposed SILS/RCP environments which are used to design an Electronic Stability Control (ESC) system which is one of the hard real-time control systems. The proposed SILS/RCP environments make it possible to realize ECUs in the early design phase by considering temporal behaviors.

INTRODUCTION

An efficient development process is an industrial requirement. The automotive industry especially must reduce the development time and cost. There have been several studies to improve the development process, such as Software-in-the-Loop Simulation (SILS), Hardware-in-the-Loop Simulation (HILS) and Rapid Control Prototyping (RCP) that are becoming more popular these days. Typically, the controller designed by SILS can be transferred to HILS by using RCP [1][2]. However, temporal behavior is not considered in the conventional SILS environments, although the performance of the production processor is limited. As a result, control algorithm developed by conventional SILS/RCP should be redesigned and validated again for a production processor which has a limited performance. To improve these problems, target identical SILS/RCP was suggested [3], which makes it possible to apply the control algorithm to the target processor with minimal changes. Moreover, it helps the validation of temporal behaviors to consider the computation of time delays.

Other issues in the automotive industry are in-vehicle networks and OSEK-OS [4], the real-time operating system for automotive software. As various functions are required and more ECUs are added, in-vehicle network systems such as CAN and LIN are being more widely used. As a result, network-induced time delays are caused by an increased number of network nodes [5]. In the domain of a real-time system, OSEK-OS makes it easy to develop and maintain the software by dividing the functions into several tasks, even though the complexity of requirements increases. On the other hand, the scheduling policy and delays should be considered for temporal behaviors. These problems can be solved by the proposed SILS/RCP environments [6]. The proposed SILS/RCP environments provide structured and modular software implementation by using OSEK-OS and distributed control system design by using the CAN and LIN protocols. The proposed SILS/RCP environments also support implementation and simulation by considering task execution delays, scheduling delays and network-induced time delays.

In this study, we validated the proposed SILS/RCP environments with an ESC system. Firstly, the controllers designed for the ESC system were simulated in the SILS environments. Secondly, the controllers were implemented by the RCP and tested in network-based Hardware-in-the-Loop Simulation (Net-HILS) environments, which consist of a CarSim Real-time and dSPACE HIL Simulator [7][8]. Finally, in order to validate the SILS/RCP environments, we compared and analyzed the control performance in the simulation result of SILS environments and in the implementation result of the Net-HILS environments.

THE VALIDATION OF SILS/RCP

THE APPLICATION OF ESC

The ESC is the active safety system that controls the vehicle's dynamic motion in emergency situations. Typically, normal drivers overreact to avoid the emergency situation. A sudden steering maneuver,

however, does not generate sufficient yaw moment. ESC controls the vehicle slip and yaw moment to follow the driver's intention by generating the additional yaw moment [9]. For this reason, ESC should respond accurately within a given time. The proposed SILS/RCP helps the development and validation process of an ESC system by considering temporal behaviors.

The following procedures are required for the development of an ESC system using the proposed SILS/RCP and the Net-HILS environments.

- Step 1. The development of a control algorithm
- Step 2. The application to the SILS/RCP environments
- Step 3. The validation in the Net-HILS environments

Firstly, an ESC control algorithm was developed using Matlab/Simulink [10]. The developed algorithm was simulated with CarSim for a validation. CarSim supports a Matlab/Simulink interface that enables the control of the vehicle model and the retrieval of the vehicle parameters. Secondly, the ESC control algorithm was applied to SILS/RCP environments with several tasks by considering task execution delays, scheduling delays and network induced time delays. Finally, the control algorithm with SILS environments was transferred to Net-HILS environments using RCP, CarSim RT and a dSPACE HIL simulator.

STEP1: THE DEVELOPMENT OF THE ESC CONTROL ALGORITHM

Derivation of an ESC Control Algorithm

The vehicle model used for the ESC has 7 degrees of freedom (DOF). Three of the DOFs are longitudinal velocity, lateral velocity and yaw rate of the vehicle. The others are the wheel velocities of the 4 wheels. The equations of motion of the vehicle are as follows [11].

$$m\ddot{x} = (F_{xfl} + F_{xfr})\cos(\delta) + F_{xfl} + F_{xfr} - (F_{yfl} + F_{yfr})\sin(\delta) + m\dot{\psi}\dot{y} \quad (1)$$

$$m\ddot{y} = F_{yrl} + F_{yrr} + (F_{xfl} + F_{xfr})\sin(\delta) + (F_{yfl} + F_{yfr})\cos(\delta) - m\dot{\psi}\dot{x} \quad (2)$$

$$I_z\ddot{\psi} = l_f(F_{xfl} + F_{xfr})\sin(\delta) + l_f(F_{yfl} + F_{yfr})\cos(\delta) - l_r(F_{yrl} + F_{yrr}) + \frac{l_w}{2}(F_{xfr} - F_{xfl})\cos(\delta) + \frac{l_w}{2}(F_{xrr} - F_{xrl}) + \frac{l_w}{2}(F_{yfl} - F_{yfr})\sin(\delta) \quad (3)$$

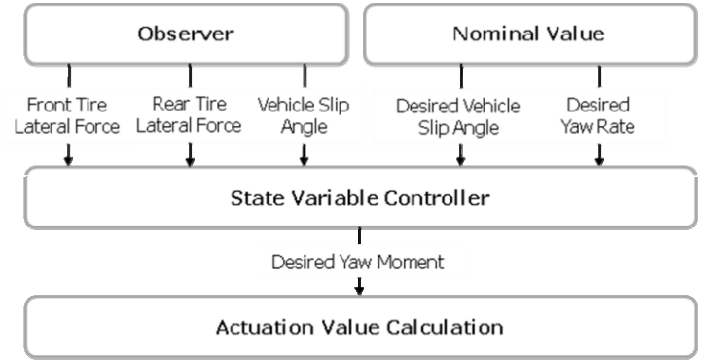


Figure 1. Control Algorithm of the ESC

The ESC stabilizes the vehicle motion by controlling the yaw moment and the slip angle of the vehicle. Figure 1 shows the block diagram of the ESC control algorithm.

The ESC consists of four parts which are: nominal values, observer, state variable controller and actuation value calculation blocks [12]. Observer and nominal value blocks estimate and calculate the values from the sensor signal that are needed to generate the desired yaw moment. The desired yaw rate and slip angle are calculated by Equations (4) and (5)

$$\dot{\psi}_{des} = \frac{\dot{x}}{L + \frac{m\dot{x}^2(l_r C_{ar} - l_f C_{af})}{2C_{af} C_{ar} L}} \delta \quad (4)$$

$$\beta_{des} = \frac{l_r - \frac{l_f m V^2}{2C_{ar} L}}{L + \frac{m\dot{x}^2(l_r C_{ar} - l_f C_{af})}{2C_{af} C_{ar} L}} \delta_{ss} \quad (5)$$

In order to maintain a tire-road adhesion, desired values should be limited. Equations (6) and (7) describe the limitation values.

$$\dot{\psi}_{upper_bound} = 0.85 \frac{\mu g}{\dot{x}} \quad (6)$$

$$\beta_{upper_bound} = \tan^{-1}(0.02\mu g) \quad (7)$$

Additionally, there are some values that should be estimated by the observer, such as vehicle slip angle and the lateral tire force of the front and rear. As a result, estimated values are used for the calculation of the desired yaw moment in the state variable controller. To follow the desired yaw rate and desired vehicle slip, the state variable controller uses the sliding mode control algorithm and calculates the desired yaw moment. The sliding surface and control law are in Equations (8) and (9).

$$\dot{s} = \ddot{\psi} - \ddot{\psi}_{target} + \xi(\beta - \dot{\beta}_{target}) \quad (8)$$

$$\dot{s} = -\eta s \quad (9)$$

Desired yaw moments are needed for the differential braking. In this study, the front tires are used to generate the additional yaw moments. As a result, the desired yaw moment $M_{\psi b}$ is defined as Equation (10)

$$M_{\psi b} = \frac{l_w}{2} (F_{xfr} - F_{xfl}) \quad (10)$$

To calculate the desired yaw moment, Equation (3) is rewritten as follows.

$$I_z \ddot{\psi} = l_f (F_{yfl} + F_{yfr}) \cos(\delta) - l_r (F_{yrl} + F_{yrr}) + M_{\psi b} (\cos(\delta) + \rho) \quad (11)$$

$l_f (F_{yfl} + F_{yfr}) \sin(\delta)$ and $l_w (F_{yfl} - F_{yfr}) \sin(\delta) / 2$ in Equation (3) are ignored by assuming that the steering angle is small. Additionally, the rear brake moment $l_w (F_{xrr} - F_{xrl}) / 2$ is substituted for the proportion of the front brake moment, that is $\rho M_{\psi b}$. ρ is determined by the front-to-back brake proportioning, which is a concern with the proportioning value in the hydraulic system.

To apply the sliding mode control, Equation (8) is differentiated and $\ddot{\psi}$ and $\dot{s}$ is substituted.

$$\dot{s} = \ddot{\psi} - \ddot{\psi}_{target} + \xi(\dot{\beta} - \dot{\beta}_{target}) \quad (12)$$

As a result, the desired yaw moment is derived as follows.

$$M_{\psi b} = \frac{1}{\rho + \cos(\delta)} \begin{bmatrix} -l_f (F_{yfl} + F_{yfr}) \cos(\delta) \\ + l_r (F_{yrl} + F_{yrr}) - I_z \eta s \\ + I_z \ddot{\psi}_{target} - I_z \xi(\dot{\beta} - \dot{\beta}_{target}) \end{bmatrix} \quad (13)$$

Finally, F_{xfr} and F_{xfl} values for differential braking should be determined properly from $M_{\psi b}$ with some gains.

The Simulation Result of Derived ESC Control Algorithm

Simulation environments were designed for the derived ESC control algorithm. In Figure 2, the block diagram shows the controller block of ESC and CarSim S-Function block that simulates the vehicle model.

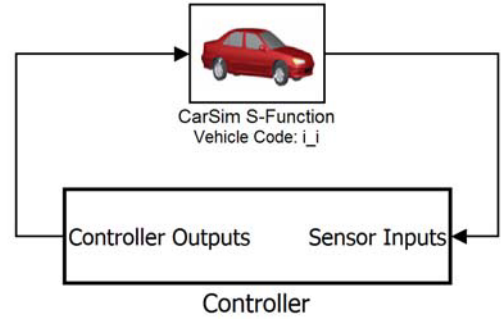


Figure 2. ESC System

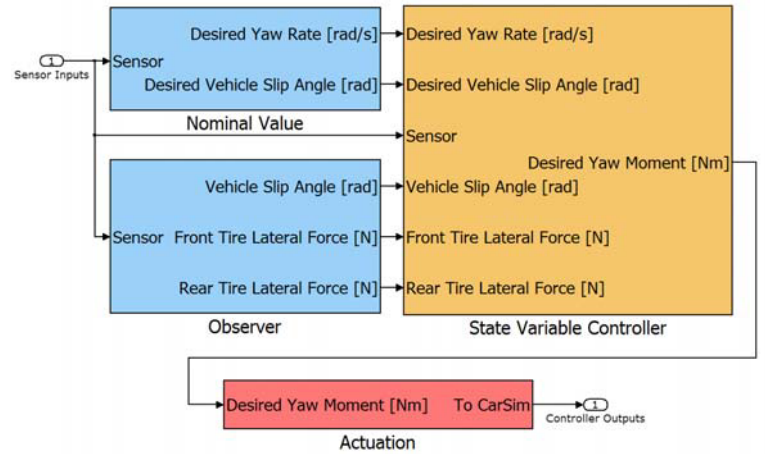


Figure 3. Block Diagram of ESC

Figure 3 shows the top level block diagram of ESC algorithm. The ESC system requires essential sensor information such as wheel speed, steering wheel angle, yaw rate and lateral acceleration and also estimated information from the observer. The Nominal Value block calculates the desired yaw rate and the desired vehicle slip based on sensor values and parameter values. The Observer block calculates the vehicle slip angle and the tire lateral forces. The State Variable Controller block controls the state variables to follow the desired values by using the sliding mode control algorithm and calculates the desired yaw moment for the Actuation block [13][14]. As a result of that, the Actuation block controls the brake differentially to generate the desired yaw moment.

In order to evaluate the ESC system, as shown in Fig 4, the slalom input is applied to the vehicle. Comparisons of the lateral vehicle motions with ESC and without ESC are performed to evaluate the control performance of the ESC system. Fig. 5 and Fig. 6 show the comparison results of the slalom test with ESC and without ESC. As shown in Fig. 5, the vehicle slip angle is limited to less than 1.5 degree by the ESC system. Fig. 6 shows the yaw rate of the two cases. In the case of the vehicle with ESC, the yaw rate is reduced to maintain the stability and controllability of the vehicle.