Cruise Control Development and Hardware-in-the-Loop Validation for Electric Motorcycles

Muhammad Akhimullah Subari
Energy Innovation Centre, WMG
University of Warwick
Coventry, United Kingdom
muhammad-akhimullah-bin.subari@warwick.ac.uk

Truong Minh Ngoc Bui
Energy Innovation Centre, WMG
University of Warwick
Coventry, United Kingdom
truong.bui@warwick.ac.uk

Truong Quang Dinh
Energy Innovation Centre, WMG
University of Warwick
Coventry, United Kingdom
t.dinh@warwick.ac.uk

Jong Il Yoon
Korea Construction Equipment
Technology Institute Korea
jiyoon@koceti.re.kr

***Abstract*—In electric motorcycle applications, speed tracking and speed limiting are important features of the vehicle supervisory control (VSC) system that allows the rider to minimize the control effort and avoid over speeding when riding on highway. In this paper, a cruise control (CC) system is developed allowing the vehicles to follow any desired speed limits within their capability. The proposed system employs a classical Proportional-Integral (PI) controller, which enables the vehicle to match the desired cruise speed set when CC function is active. The developed CC system is verified and validated using Hardware-in-the-Loop (HiL) simulation testing. The HiL results demonstrate that the proposed CC logic function effectively, with a maximum percentage of error between the vehicle speed and the cruise reference speed is less than 7%.**

Keywords—cruise control, PI control, hardware-in-the-loop, electric motorcycles

I. INTRODUCTION

The history of cruise control (CC) system emerged in the late of 1950s when it was developed to hold the throttle in a fixed position and allow the vehicle to travel at a constant target speed [1, 2]. Nowadays, the CC system has become a common feature in most types of vehicle applications such as cars, trucks and even motorcycles [3]. CC system greatly benefits the rider because it reduces the mental and physical fatigue caused by regularly monitoring the vehicle's speed and frequently pressing the gas pedal or throttle to maintain the speed [4].

In line with the development of modern technology and the industrial revolution, cruise control system has evolved to be more robust, reliable, and performed its primary function much more efficiently. Several researchers have improved the conventional cruise control system, such as implementing an adaptive cruise control (ACC) system [5-8] and predictive cruise control (PCC) system [9-11]. In ACC system, if the preceding vehicle is detected, it will calculate and judge whether the vehicle can still travel safely in the rider's predefined cruise speed [12]. Meanwhile, PCC systems are developed to reduce the idle time of the vehicles by using the upcoming traffic signal information and decreasing the use of brakes [13].

Hardware-in-the-loop (HiL) simulation has been intensively employed in different areas such as system control development and verification, product development and assessment, and system performance validation [14, 15]. It enables testing of physical system components in conjunction with virtual computer-based simulation in a real-time testing environment. HiL validation plays a vital role for validating and verification of the systems during the product development process. Not only can it reduce the testing time spent during the development phase, but also it reduces the overall cost required comparing to the conventional physical testing. Hence, one of the novelties in this paper is the validation process of HiL in CC system environment. The validation process will consist of hardware setup and simulation setup. Additionally, this paper will introduce a cruise control strategy which will have three different mode; off, enable and active. The structure of this paper is presented as follows: Section II is about the cruise control development. Section III discusses about the HiL implementation for the developed functionality. Section IV presents the results and discussion. Finally, the conclusion is drawn in Section V.

II. CRUISE CONTROL DEVELOPMENT

A. Cruise Control Strategy Development

Various approaches can be found in the development of cruise control strategies [16-18]. Some researchers utilize accelerating and braking behaviour for their strategies, while others incorporate navigation tools. In this paper, acceleration and braking behaviour-based is chosen as the benchmark for developing the cruise control strategy due to its straightforward functionality and ease of control.

The cruise control strategy developed in this study consists of three modes: off, enable, and active. During the off mode, the cruise control has no function and is entirely disabled. In the enable mode, it only sends a signal to the dashboard indicating that the cruise control is ready. In active mode, the cruise control actively maintains the vehicle's desired travelling speed. The transition between modes must be based on certain conditions. For example, from off mode to the enable mode, the cruise button needs to be pressed for more than 0.5 seconds, and the vehicle must be in the drive state. On the other hand, transitioning from enable mode to the active mode requires the cruise button to be pressed for less than or equal to 0.5 seconds, the bike speed must be more than 20 km/h, and the vehicle must be in drive state.

However, to shift from the active mode to the enabled mode, where the cruise reference speed is disabled, one of the following conditions must be met which the button is pressed for less than or equal to 0.5 seconds, the kill switch is on, the side stand is down, and the brake is applied. Meanwhile, for the transition from the enable mode to the off mode, only one condition needs to be fulfilled: the cruise button should be pressed for more than 0.5 seconds. The output of this cruise control strategy will be fed into the speed controller algorithm, which will then be flashed into the electronic control unit (ECU). Figure 1 shows the transition constraints among different modes while Table 1 presents the definition of the acronyms in the developed cruise control strategy logic.

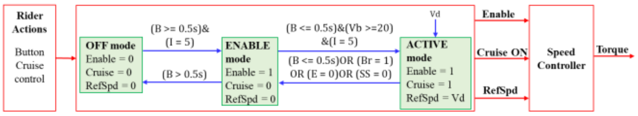

Fig. 1. Cruise control strategy

TABLE I. ACRONYMS DEFINITION

Symbol	Definition
B	Cruise control button
I	Ignition switch
Br	Brake switch
E	Emergency switch
SS	Side stand switch
Vb	Actual bike speed
Vd	Desired speed

B. *PI Controller Design For Cruise Control Strategy*

As aforementioned, a speed controller is used for the developed cruise control strategy. Here, a Proportional-Integral (PI) controller is chosen to control the bike speed following any desired speeds that satisfy the functional constraints. This PI controller is widely used in industry due to its simplicity and fast response to the system behaviour [19]. The detail of the employed PI controller is shown in Figure 2. The formulation of the PI controller used in this study can be defined as follows:

$$U(t) = k_p e + k_i \int_0^t e(t)dt \qquad (1)$$

where $U(t)$ is the controller output, e is the input error between the vehicle speed and the cruise reference speed, k_p and k_i are the controller gains, and t is the time interval.

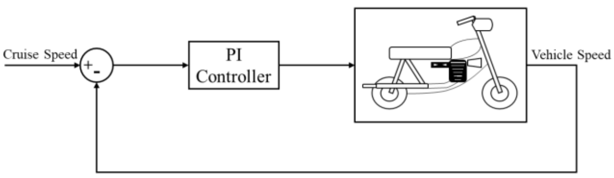

Fig. 2. Block diagram of cruise controller.

III. HARDWARE-IN-THE-LOOP IMPLEMENTATION

The HiL configuration setup and signal communications for the implementation are depicted in Figure 3. A host computer with Intel Core i7 9700 @3.0GHz of CPU and 16GB of RAM running dSpace ControlDesk software, which provides a graphical user interface (GUI) for testing and implementation purposes was used. The GUI comprises two main parts: rider control interface and dashboard. The rider control interface part consists of different buttons to emulate the actual switches and buttons of the real bike, allowing the rider/user to trigger or disable the switches. The dashboard portion is designed to monitor the system behaviors, enabling the rider/user to observe the system responses through virtual indicators and gauges.

Additionally, the implementation employs a Controller Area Network (CAN) data logger VN6310A with CANalyzer software from Vector to monitor the signal transmissions through the CAN network. The CAN bus operates at a fixed baud rate of 500 kbps.

It is noteworthy to mention that, to reduce the complexity of using physical hardware and to satisfy the safety requirements and limitations of real-time testing within the Lab condition, most of the system components are modelled and simulated within HiL simulators except the main vehicle supervisory control (VSC) system containing the developed cruise control functionality, which is finally built into a real electronic control unit (ECU) hardware. As shown in Figure 3, several hardware simulators and components are required to configure the electric motorcycle system for HiL testing, including Microautobox II (MAB), Scalexio, ECU and a telematics unit.

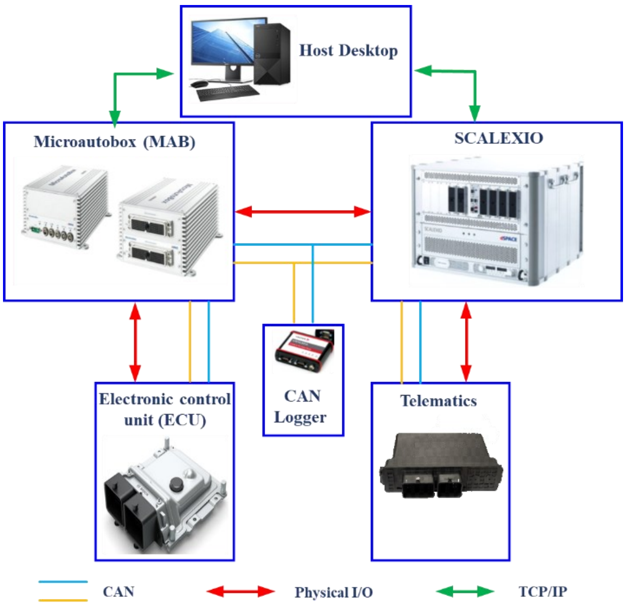

Fig. 3. HiL platform configuration.

The MAB simulates the vehicle's battery and battery management system (BMS), inverter and motor based on their models. On the other hand, Scalexio is used to simulate the rider behavior, vehicle dynamic model including transmission, wheel-brake system models. Due to the limited topic of interest, which is only focused on testing the CC performance, the details of the electric motorcycle model used in this study will be presented in a future work.

The ECU plays a crucial role in this HiL setup, as it contains a complete VSC system algorithm, which is developed and C-code generated using Embedded coder toolbox of Matlab 2021a, to manage the entire operational of the vehicle. The C-code is then finally flashed through the ECU. The VSC has two main functions: low-level VSC and high-level VSC. The low-level VSC is responsible for handling rider commands via buttons and switches on the vehicle control panel, such as turning on/off the lights, requesting cooling system, controlling lighting, cranking, and braking. Meanwhile, high-level VSC is more focused on the vehicle's control performance, control and manage the driving torque during acceleration or braking, alternating current (AC) and direct current (DC) charging, and derating power output based on temperature, battery states, and driving conditions.

The telematics unit is an optional equipment, which is connected to the system CAN bus to log the data during operation and upload such data to the cloud database for back-end analysis purposes.

IV. Results and Discussion

Before carrying out the HiL validation, a set of PI controller parameters was identified using the try-and-error method, as shown in Table II, to ensure the vehicle can accurately track the desired speed. Next, to test the operational and performance of the developed CC system, the reference speed of the vehicle is set to 30mph and 60mph, respectively. Both desired speeds were chosen as they represent the United Kingdom (UK) national speed limit for suburbs area and single-carriageways drive. Figure 4 and Figure 5 depict the cruise control performance of the motorcycle at 30 and 60 mph. From both figures, the red line represents the vehicle speed, the blue line represents the vehicle torque demand where negative torque is a driving torque and positive torque is a regenerative torque, and the green line represents the throttle position. It is noted that the throttle position is automatically controlled during the CC operation. Hence, the rider is only controlling the direction and managing the stability of the vehicle during this time.

The testing procedure and the respond of the bike performance when cruising at 30 mph is explained based on Figure 4 as follows: at point 1, 24% of throttle input was given, which resulted in the vehicle generating a torque of 10.25 Nm. Then, at point 2, CC is enabled by pressing the cruise button for more than 0.5 seconds as required. When the vehicle reaches 30 mph at point 3, the cruise button is pressed once again to indicate that the CC are now in active mode. At point 4, the throttle position is released, allowing the regenerative deceleration to slow down the bike. Here, the bike speed will be reduced until reaching the cruise reference speed. At point 5, the CC is returned to enable mode, reducing the bike speed.

Similarly, Figure 5 shows the result of CC performance at 60 mph. The test procedure is almost the same as discussed before, where the CC is enabled at point 1, the CC is set to active at point 2 at speed of 60 mph, and the throttle is released at point 3 to observe the CC performance. Table III shows the tracking speed error in percentage between the cruise reference speed and the actual vehicle speed, with a maximum of 6.51% recorded at 60 mph and 1.49% recorded at 30 mph.

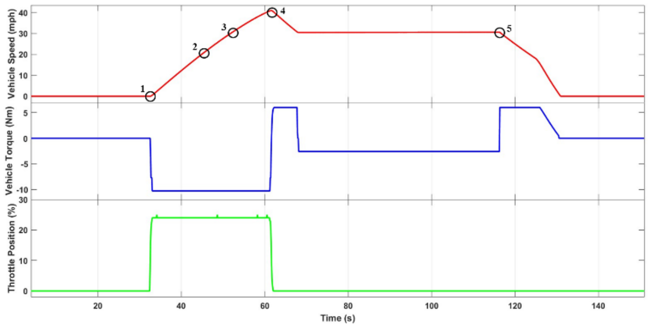

Fig. 4. Cruise control performance at 30 mph.

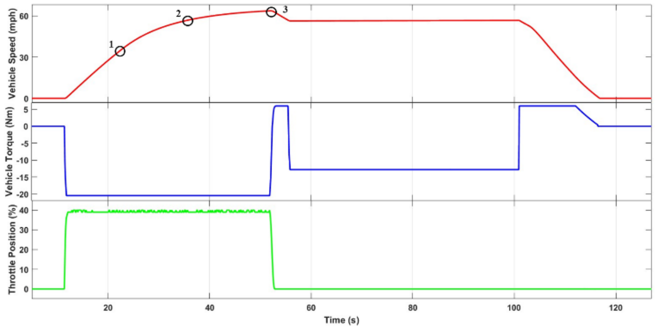

Fig. 5. Cruise control performance at 60 mph.

TABLE II. PID Controller Parameter

Parameters	Value
K_p	50
K_i	0.1

TABLE III. Percentage of differerence of CC system.

Cruise Speed (mph)	Percentage of Difference (%)
30	1.49
60	6.51

V. Conclusions

This paper presents the development and HiL validation of the CC systems for an electric motorcycle application. The results showed that the proposed CC system and HiL validation procedure for this study were successful where the percentage of speed tracking error recorded are less than 7%, which is reasonable for normal drive. However, some improvements can be made for further researcher attention such as to find the optimum value of controller parameters of the PI controller to minimise the error. This can be achieved by using optimization tools such as particle swarm optimization (PSO) or gravitational search algorithm (GSA) in the next study.

Acknowledgment

This work is supported by Innovate UK through the electric BSA (eBSA), Project number: 75281 in collaboration between WMG, BSA, Hypermotive, Dana TM4, Microlise and Vital Auto.

References

[1] Yu, L., & Wang, R. (2022). Researches on Adaptive Cruise Control system: A state of the art review. *Proceedings of the Institution of*

Mechanical Engineers, Part D: Journal of Automobile Engineering, *236*(2-3), 211-240.

[2] Shaout, A., & Jarrah, M. A. (1997). Cruise control technology review. *Computers & electrical engineering*, *23*(4), 259-271.

[3] Abdulnabi, A. (2017). PID controller design for cruise control system using particle swarm optimization. Iraqi Journal for Computers and Informatics, 43(2), 30-35.

[4] Kim, D., Moon, S., Park, J., Kim, H. J., & Yi, K. (2009, August). Design of an adaptive cruise control/collision avoidance with lane change support for vehicle autonomous driving. In 2009 ICCAS-SICE (pp. 2938-2943). IEEE.

[5] Wang, M., Daamen, W., Hoogendoorn, S. P., & van Arem, B. (2014). Rolling horizon control framework for driver assistance systems. Part I: Mathematical formulation and non-cooperative systems. Transportation research part C: emerging technologies, 40, 271-289.

[6] Mehra, A., Ma, W. L., Berg, F., Tabuada, P., Grizzle, J. W., & Ames, A. D. (2015, July). Adaptive cruise control: Experimental validation of advanced controllers on scale-model cars. In 2015 American control conference (ACC) (pp. 1411-1418). IEEE.

[7] Brugnolli, M. M., Pereira, B. S., Angélico, B. A., & Laganá, A. A. M. (2019). Adaptive cruise control with a customized electronic control unit. Journal of Control, Automation and Electrical Systems, 30, 9-15.

[8] Chen, X., Zhai, Y., Lu, C., Gong, J., & Wang, G. (2017, June). A learning model for personalized adaptive cruise control. In 2017 IEEE intelligent vehicles symposium (IV) (pp. 379-384). IEEE.

[9] Chu, H., Guo, L., Gao, B., Chen, H., Bian, N., & Zhou, J. (2018). Predictive cruise control using high-definition map and real vehicle implementation. IEEE Transactions on Vehicular Technology, 67(12), 11377-11389.

[10] Nie, Z., & Farzaneh, H. (2022). Real-time dynamic predictive cruise control for enhancing eco-driving of electric vehicles, considering traffic constraints and signal phase and timing (SPaT) information, using artificial-neural-network-based energy consumption model. Energy, 241, 122888.

[11] Chen, H., Guo, L., Ding, H., Li, Y., & Gao, B. (2018). Real-time predictive cruise control for eco-driving taking into account traffic constraints. IEEE Transactions on Intelligent Transportation Systems, 20(8), 2858-2868.

[12] Xiao, L., & Gao, F. (2010). A comprehensive review of the development of adaptive cruise control systems. Vehicle system dynamics, 48(10), 1167-1192.

[13] Gao, W., Odekunle, A., Chen, Y., & Jiang, Z. P. (2019). Predictive cruise control of connected and autonomous vehicles via reinforcement learning. IET Control Theory & Applications, 13(17), 2849-2855.

[14] Bui, T. M., Niri, M. F., Worwood, D., Dinh, T. Q., & Marco, J. (2019, October). An Advanced Hardware-in-the-Loop Battery Simulation Platform for the Experimental Testing of Battery Management System. In 2019 23rd International Conference on Mechatronics Technology (ICMT) (pp. 1-6). IEEE.

[15] Bui, T. M., Dinh, T. Q., Marco, J., & Watts, C. (2021). Development and real-time performance evaluation of energy management strategy for a dynamic positioning hybrid electric marine vessel. Electronics, 10(11), 1280.

[16] Yi, K., & Moon, I. K. (2004, March). A driver-adaptive stop-and-go cruise control strategy. In IEEE International Conference on Networking, Sensing and Control, 2004 (Vol. 1, pp. 601-606). IEEE.

[17] Lin, Y. C., & Nguyen, H. L. T. (2016). Development of an eco-cruise control system based on digital topographical data. Inventions, 1(3), 19.

[18] Riley, B., Kuo, G., Schwartz, B., Zumberge, J., & Shipp, K. (2000). Development of a controlled braking strategy for vehicle adaptive cruise control (No. 2000-01-0109). SAE Technical Paper.

[19] Subari, M. A., Hudha, K., Kadir, Z. A., Dardin, S. M. F. S. M., & Amer, N. H. (2022). Path following control of tracked vehicle using modified sup controller optimized with particle swarm optimization (PSO). International Journal of Dynamics and Control, 10(5), 1461-1470.