

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter

Knowledge-Based Controller Optimised with Particle Swarm Optimisation for Adaptive Path Tracking Control of an Autonomous Heavy Vehicle

Noor Hafizah Amer, Khisbullah Hudha, Hairi Zamzuri, Vimal Rau Aparow, Amar Faiz Zainal Abidin, Zulkifli Abd Kadir and Muhamad Murrad

Abstract

This chapter discusses the development of an adaptive path tracking controller equipped with a knowledge-based supervisory algorithm for an autonomous heavy vehicle. The controller was developed based on a geometric/kinematic controller, the Stanley controller. One of the mostly known issues with any geometric/kinematic controller is that a properly tuned controller may not be valid in a different operating region than the one it was being tuned/optimised on. Therefore, this study proposes an adaptive algorithm to automatically choose an optimal controller parameter depending on the manoeuvring and vehicle conditions. An optimal knowledge database is developed for an adaptive algorithm to automatically obtain the parameter values based on the vehicle speed, v , and heading error, ϕ . Several simulations are carried out with different trajectories and speeds to evaluate the effectiveness of the controller against its predecessors, namely, Stanley and the non-adaptive modified Stanley (Mod St) controllers. The simulated steering actions are then compared against human driver's experimental data along the predefined paths. It was shown that the proposed adaptive algorithm managed to guide the heavy vehicle successfully and adapt to various trajectories with different vehicle speeds while recording lateral error improvement of up to 82% compared to the original Stanley controller.

Keywords: heavy vehicle, autonomous trajectory tracking, path tracking, Stanley controller, particle swarm optimisation

1. Introduction

This study proposed a new adaptive steering control strategy for trajectory tracking controller of a heavy vehicle. The controller aims to automatically steer the vehicle along the desired trajectory and adapt to various speeds and trajectories

during varied manoeuvres. A path tracking controller is a controller module that is developed to provide electronic actuation to the vehicle system while navigating the vehicle automatically. An effective controller needs to be developed to ensure a functional steering module while autonomously navigating through various paths. Commonly, there are several types of trajectory tracking controllers as reviewed previously [1, 2]. One of the most common types of controllers is kinematic controllers such as Pure Pursuit and Follow-the-Carrot due to the simplicity and stability it can provide. This type of controllers relies on the kinematic properties of the vehicles such as speed and acceleration, as well as the travelled distance for the controller feedback. Compared to other dynamic controllers that require the kinetic properties of the vehicle such as torques, moments, and forces, geometric and kinematic properties are relatively easier to measure.

In geometric/kinematic controllers, one of the most established controllers is the Stanley controller as published in [3, 4]. The controller was developed by the Stanford University racing team on their autonomous vehicle (Stanley) in winning the DARPA Challenge in 2005 [5]. Compared to other geometric/kinematic controllers, the Stanley controller does not include a lookahead distance in its formulation. This enables the controller to be robust enough without depending on what lies ahead. However, the Stanley controller poses different problems, which is common in most of the geometric/kinematic controllers. It was found that the performance of this controller on any given trajectory depends on how well the parameter-tuning process was. A finely tuned Stanley controller will be effective only on the driving conditions that it was tuned for. However, it needs to be re-tuned to work with another road course and speed range. Previous studies have been discussing this issue and stated the same conclusion for most of geometric/kinematic controllers [1, 3, 6].

Therefore, an improvement is proposed to include an adaptive algorithm that will adjust the controller's parameters based on the driving conditions. Adaptive controllers for trajectory tracking controller have been proposed in numerous studies recently [7–10] to improve the adaptability and stability of the controller under varying conditions. These controllers are designed to cater robustness in a specific area, such as for slippery roads [7], unknown slip conditions [9], and unknown skidding conditions [10]. While the adaptiveness of these controllers in the designated area was proven, respectively, it may not be as effective when dealing with multiple types of disturbances other than the ones it was designed for. For example, an adaptive controller designed to cater various skidding conditions may not be able to cater unknown yaw disturbance. Also, most of the studies for an autonomous vehicle are using the linear vehicle model to develop the control structure where most of the nonlinearity in vehicle motions is neglected such as frictions and aerodynamic effects. In addition, some adaptive algorithms consist of algorithms that require high computational capability due to its associated complexity. Therefore, the adaptive controller proposed in this study is aimed to solve these issues by (1) considering a nonlinear vehicle model containing most of the nonlinearity of a vehicle motion in the controller development phase; (2) using adaptive inputs as vehicle speed and heading error, which both are directly dependent on the sharpness of turns and vehicle slips; and (3) using a simple geometric/kinematic controller as the basic controller to be modified.

Overall, the main contribution of this work is on the development of a knowledge-based algorithm using the adaptive mechanism of path tracking controller to accommodate the varying trajectories and vehicle's speed setting.

Two inputs are considered in triggering the adaptive algorithm, which are the heading error, ϕ , and vehicle speed, v . The range of speed region catered in this study is based on the limitations and the expected operating region of the autonomous heavy vehicle, which is up to 72 km/h. Variations in road course trajectory are observed in terms of the instantaneous difference between vehicle heading and trajectory direction, ϕ with a range between 0 and 75 deg. in both directions. This may as well cater sharp turns. The basic controller is modified to increase its sensitivity to disturbance. Then, by optimising the controller parameters for different disturbance input combinations, a knowledge database is developed. With this, a set of optimum parameters can be chosen depending on the instantaneous speed and trajectory experienced by the vehicle. An algorithm has been developed to carry out the selection process. The controller's performance is then evaluated on six different trajectories and four random speed values to evaluate its effectiveness against the basic Stanley controller and the modified controller without an adaptive algorithm. Results show a promising prospect for the proposed controller.

This chapter starts with Introduction section that covers a brief background of the study, followed by the modelling of a nonlinear seven-degree-of-freedom (7DoF) vehicle model used to simulate the vehicle's behaviour. Next, the proposed adaptive controller is explained in Section 3, beginning with the basic controller's and the adaptive algorithm development. The simulation and experiment procedures in evaluating the controllers including the development of road courses used are presented in Section 4, with the findings discussed in Section 5. Conclusions for this work are presented in the final section.

2. Modelling of a 7DoF heavy vehicle

In this study, a full vehicle model is developed with the aim to simulate the vehicle's behaviour in lateral direction. The model, developed based on a heavy vehicle, namely, a High Mobility Multipurpose Wheeled Vehicle (HMMWV), is closely related to an armoured vehicle as shown in **Figure 1**. It consists of several

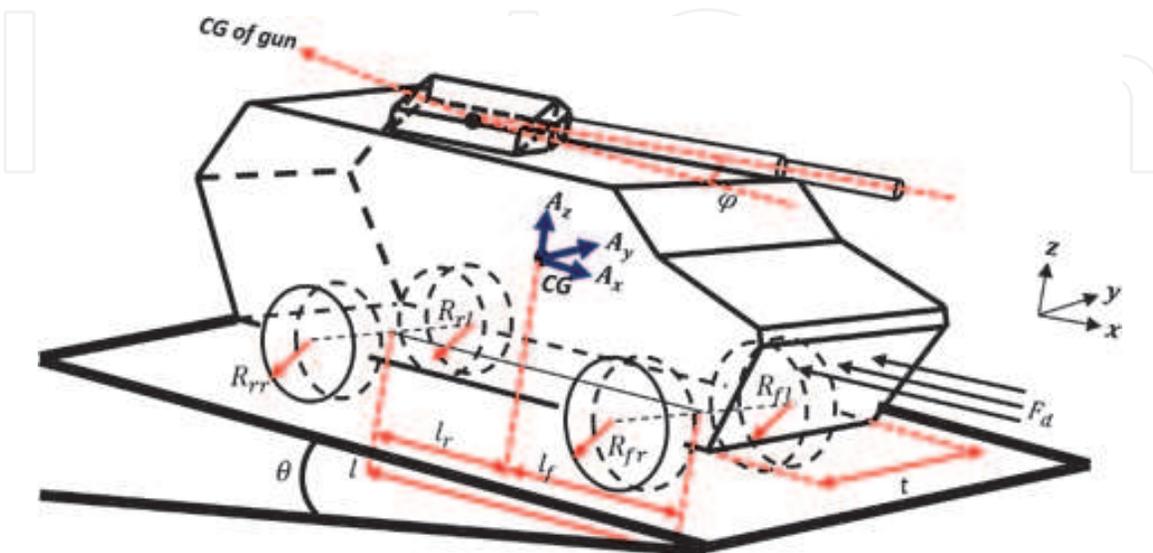


Figure 1.
Heavy vehicle model [11–13].

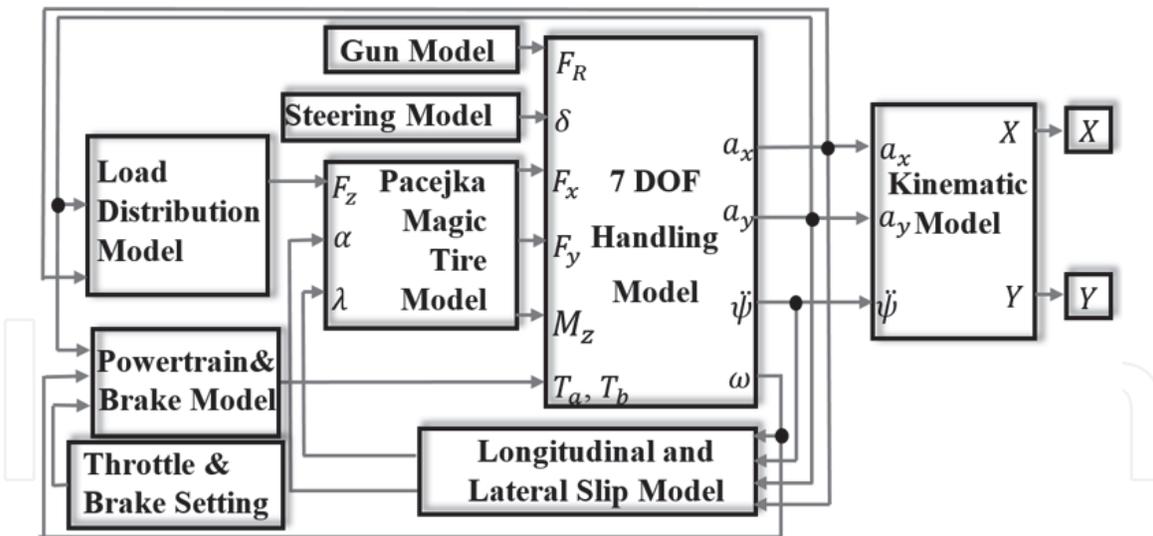


Figure 2.
Configuration of the heavy vehicle model [12, 14, 15].

subsystems in order to simulate the different elements in the vehicle system that contribute to the overall vehicle performance in lateral directions: a 7DoF handling model to simulate the vehicle dynamic responses at its centre of gravity during manoeuvrings; a tyre model in order to simulate all four tyres' behaviour from the road surface interactions; a slip model to calculate the generated slips in lateral and longitudinal directions during driving manoeuvrings, which will be an important input to the tyre model; a load distribution model to estimate the static and dynamic load transfers in longitudinal and lateral directions, which are significant during accelerating, braking, and/or cornering; an engine model; and lastly, a kinematic model to evaluate the vehicle's position relative to the local and global coordinates. The configuration of the overall vehicle model and its respective subsystem is shown in **Figure 2**.

Few assumptions and simplifications are made in developing this model. Firstly, this model focuses mainly on the vehicle responses in lateral and longitudinal directions. Responses and disturbances in vertical direction are considered less significant by assuming an ideal suspension system between vehicles' sprung body, and the vehicle is assumed to travel on even and smooth roads. With these two assumptions, the ride model involving suspension forces and road disturbances is not considered in this model. Next, the vehicle is modelled as a rigid body with concentrated sprung mass at the centre of gravity and four wheels that are connected to each of the vehicle four corners. The vehicle is moving on a level, unbanked, and uninclined road, and therefore, the weight is acting along the z -axis. The vehicle body is represented in three-dimensional x - y - z planes, which is allowed to displace in lateral and longitudinal directions, as well as rotating about the z -axis (yaw). Each of the connected wheels is modelled as a rigid body, which is allowed to rotate about its rotational axis, which is parallel with the vehicle's lateral axis. Only the two front wheels are allowed to steer, which is equipped with an active Pitman arm steering system. Modelling of this system is also considered in this chapter. In calculating the tyre responses, the vertical load on each tyre is calculated using the vehicle's load transfer model by considering the amount of load transfer during manoeuvrings. The load transfer is based on weight distribution in lateral direction during cornering and longitudinal direction during acceleration/braking. To ensure that the model generate response as close as possible to a real condition, rolling and air resistance are considered in this model. In terms of the steering system, the

armoured vehicle is equipped with a Pitman arm steering system where the output steering values were saturated at ± 10 deg. This is due to the consideration of the system's limitation as well as the controller's stability region that has been tested as described previously [3, 14].

The previous work from the authors has demonstrated the derivations and verifications of this model [12, 15]. Also, several previous studies have been evaluating its controller's performance on the same model [11, 14, 16]. **Figure 1** also shows the symbols and vehicle parameters used in developing the model. Perhaps it is worth noting a common confusion between the fixed global coordinate axes (X , Y , and Z), which are commonly associated with the Earth's longitude, latitude, and altitude, respectively, and the moving local coordinate axes (x , y , and z). This model has been validated by verifying the simulated vehicle responses against the HMMWV responses from a CarSim software as described in detail by Aparow et al. [12]. The verification was done using the standard manoeuvring procedures, namely, Slalom tests, double lane change, and step steer manoeuvrings.

3. Knowledge-based adaptive controller development for a path tracking system

3.1 Basic structure of trajectory tracking controller

In this study, the adaptive controller was based on the previously developed controller from the previous publication by Amer et al. [14], namely, the modified Stanley (Mod St) controller. Equation (1) shows the original Stanley controller from Hoffmann et al. [3]. The same publication has extended the controller to include the yaw rate compensation as shown in Eq. (2). A modification has been carried out to these controllers to increase their sensitivity towards the disturbance input as proposed previously in [14]. The Mod St controller used in this work is shown in Eq. (3), where the main parameters are shown in **Figure 3**. Here, e is the lateral error of the vehicle (m), measured between the vehicle and the perpendicular point on path; ϕ is the heading error (rad), which is the difference between the vehicle's and the path's instantaneous directions; v is the vehicle speed (m/s); $\dot{\psi}$ is the instantaneous yaw rate of the vehicle (rad/s); and $\dot{\psi}_{traj}$ is the trajectory yaw rate (rad/s). Meanwhile, k_ϕ , k_1 , k , and k_ψ are the tuneable gains. In this study, the original Stanley controller from Eq. (1) is used as the benchmark, while the Mod St controller from Eq. (3) is used as the base controller for the adaptive control structure.

$$\delta = \phi + \tan^{-1} \left(\frac{ke(t)}{v(t)} \right) \quad (1)$$

$$\delta(t) = \phi + \arctan \left(\frac{ke(t)}{1+v(t)} \right) + k_\psi (\dot{\psi} - \dot{\psi}_{traj}) \quad (2)$$

$$\delta(t) = k_\phi \phi + k_1 \arctan \left(\frac{ke(t)}{1+v(t)} \right) + k_\psi (\dot{\psi} - \dot{\psi}_{traj}) \quad (3)$$

Six trajectories are chosen for this study in order to portray the different kinds of road courses, shown in **Figure 4**. Each was named based on the trajectory shape,

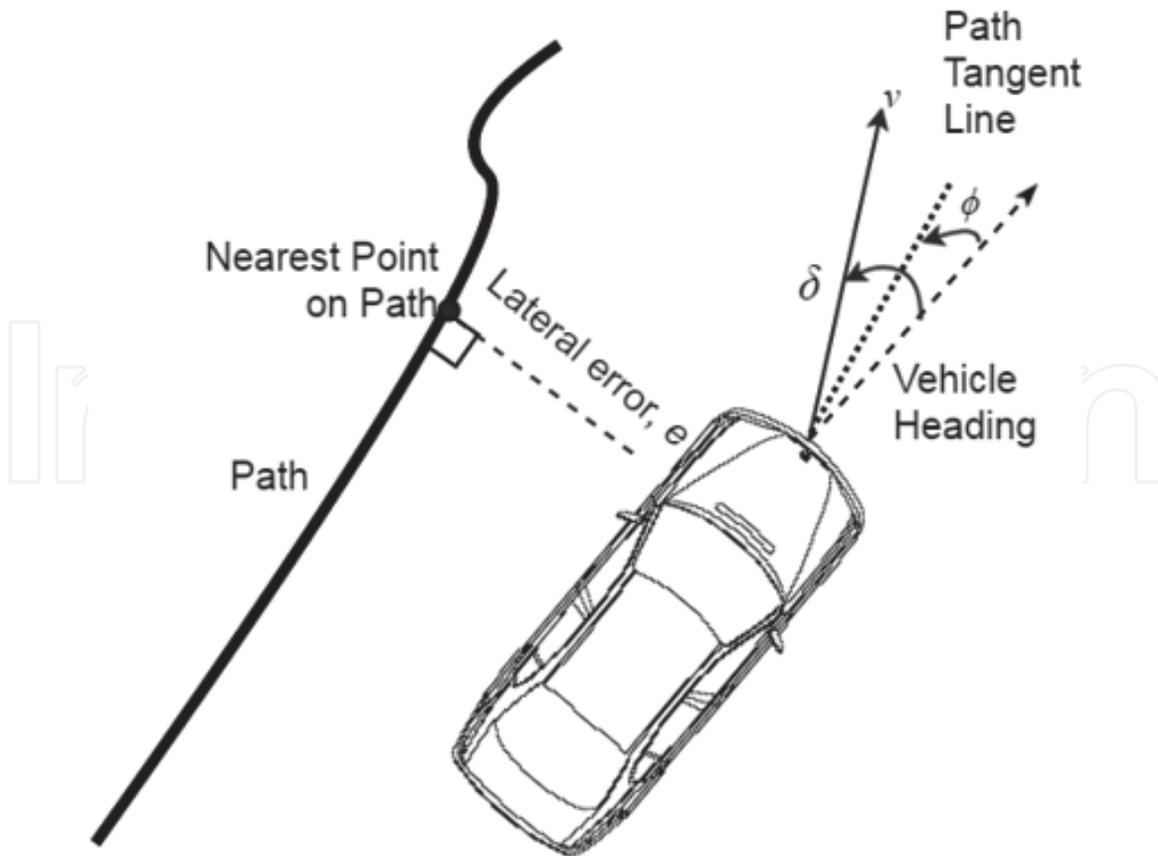


Figure 3.
Main parameters for the Stanley controller and its variants.

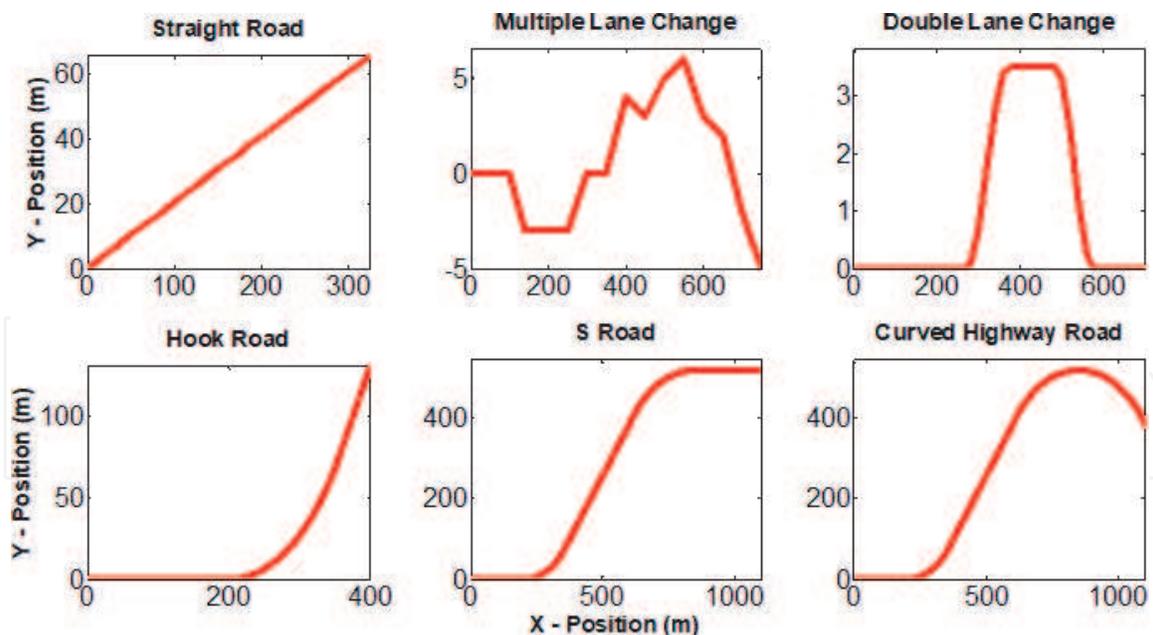


Figure 4.
Road courses for controller testing.

namely, “straight road”, “multiple lane change”, “double lane change”, “curve”, “S”, and “hook”. Each trajectory is defined as a set of points with X-Y coordinates. Therefore, lateral error is obtained based on the current X position for the vehicle, such that $e = Y_{path}(X_{vehicle}) - Y_{vehicle}(X_{vehicle})$.

Further analysis on the application of controller on a heavy vehicle yields an important issue in tuning the parameters as discussed previously by Snider [6] and Shan et al. [17]. A perfectly tuned controller will work properly within a certain

range of vehicle speed, as well as a certain type of trajectory and road courses. Also, the more parameters included in the controller, the more sensitive it is to changes, which will make it less robust. These effects can be shown in **Figure 5** where all the three variants of the Stanley controller were simulated on the heavy vehicle model with different speeds on S road. Each of the controllers was properly tuned for 6 m/s speed on the particular trajectory using particle swarm optimisation (PSO) algorithm with the same approach as presented in [13–15]. **Table 1** shows the parameter values for each controller upon optimisations for each trajectory at 6 m/s speed, which was used throughout the testing for all speed values. From the figures, it can be seen that the properly tuned controller is valid only for a certain range of speed and the number of parameters is one of the main factors that affect the robustness of the controller. The controller with the least parameters showed better robustness and performance in varying speeds. However, significantly large error (0.4–0.8 m) can be noticed. This is agreeable to the finding by Wallace et al. [18], which concludes that it is a conflicting factor in tuning a geometric controller between stability and robustness. Therefore, this study proposes an adaptive controller with the ability to supervise the selection of an optimum set of controller parameters based on the speed and trajectory experienced by the vehicle. With this, the controller can be used regardless of the manoeuvring conditions without the need to be re-tuned.

With the original Stanley controller modified into the Mod St controller in Eq. (3) to increase its sensitivity, an adaptive algorithm is proposed to automatically tune the four parameters in Mod St, namely, k_ϕ , k_1 , k , and k_ψ . An adaptive algorithm using a three-dimensional control surface is used. Since it was known that the main factors affecting the validity of any set of controller parameters are both the vehicle speed and type of trajectory, two inputs are chosen for the algorithm, which are the heading error, ϕ , and vehicle speed, v . Each of the controller parameters is optimised for each input combination to form a knowledge database. With this, the adaptive algorithm is developed and converted into separate control surfaces for each parameter where the adaptive algorithm will be developed to automatically

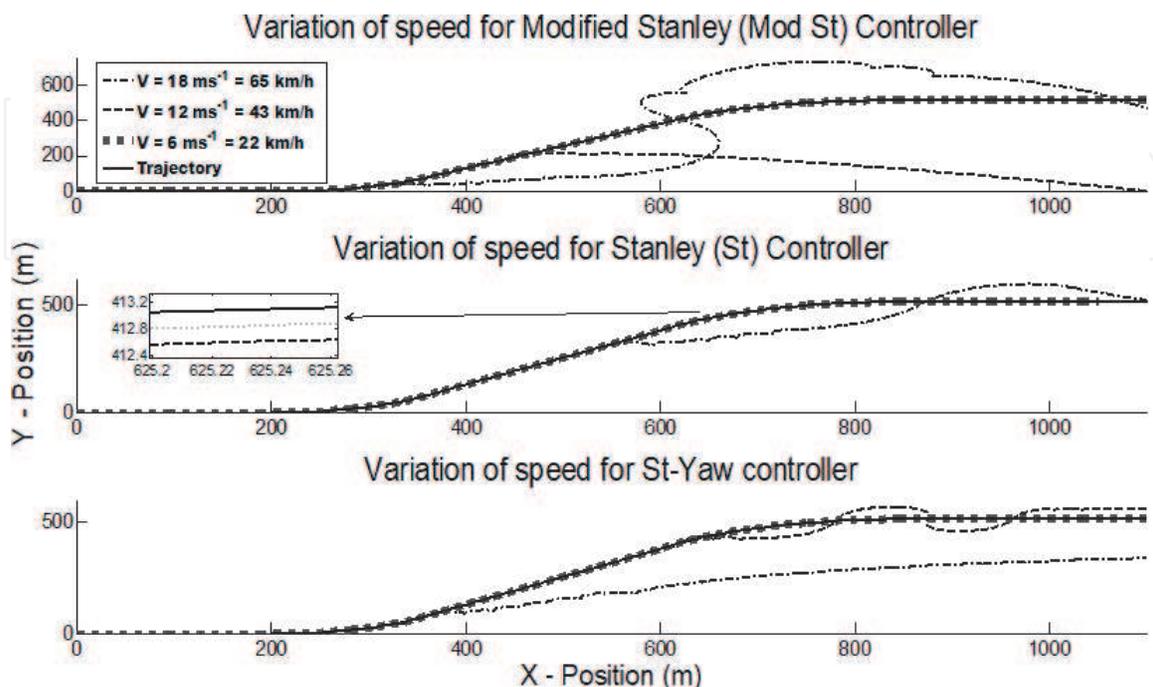


Figure 5. Effect of varying speeds on the S road for the Mod St controller [Eq. (3)], Stanley controller [Eq. (1)], and Stanley controller with yaw compensation [Eq. (2)].

	St	$Mod-St$			
	k	k_l	k	k_ψ	k_ϕ
Straight	10	10	10	-2.964	0.7719
Multiple Lane Change	10	10	10	0.3046	1.892
Double Lane Change	10	10	9.689	0.0901	0.819
Hook	10	10	10	0.0423	-0.058
S	10	10	9.757	0.0642	0.0199
Curve	10	10	10	0.1762	-0.0051

Table 1.
Controller parameters for St and $Mod St$ controllers for each trajectory.

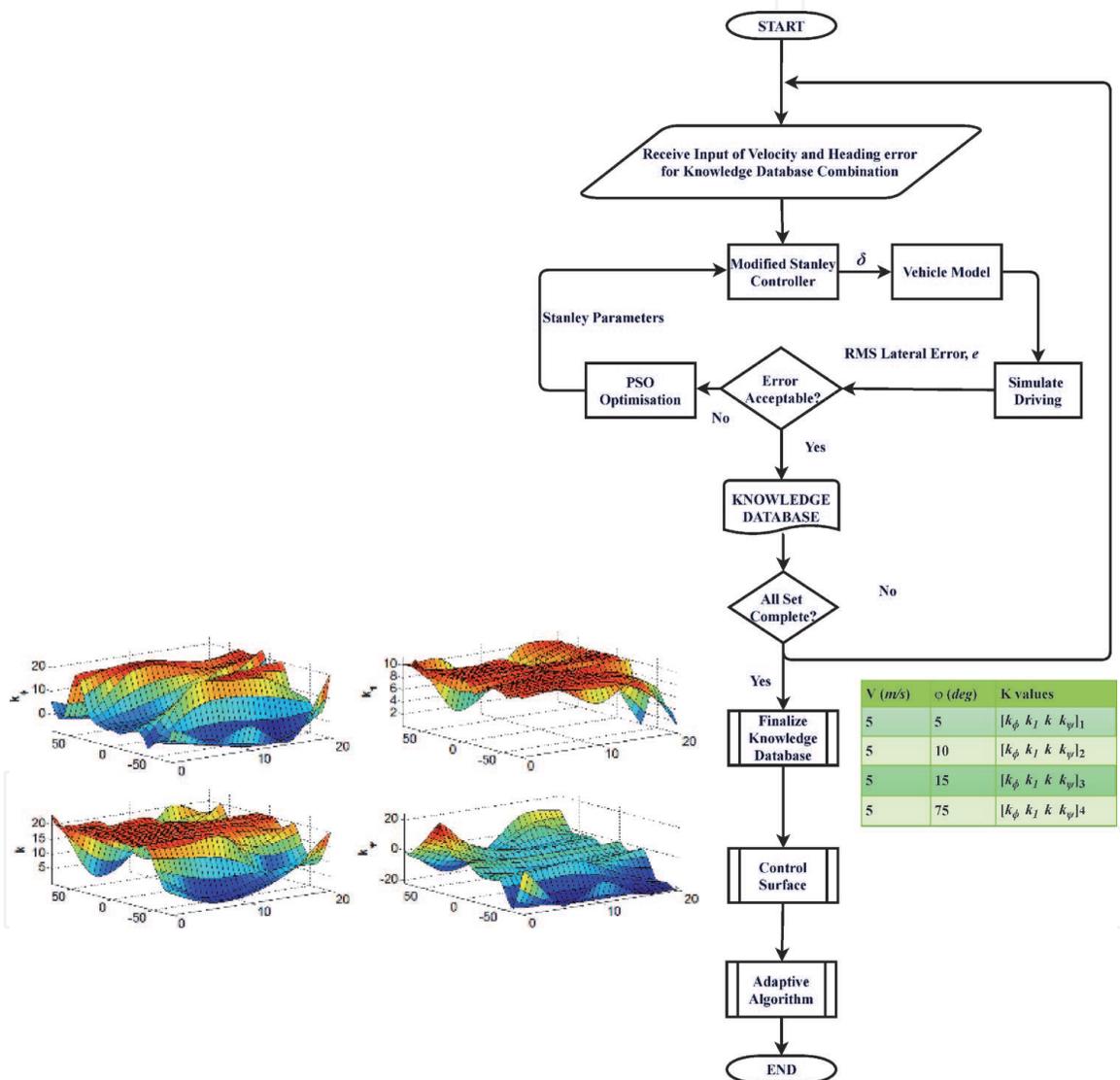


Figure 6.
Procedure in developing an adaptive modified Stanley controller.

choose parameter values from the surfaces. The overall procedure can be illustrated in Figure 6.

3.2 Development of the adaptive controller

An adaptive controller with the ability to supervise the selection of an optimum set of controller parameters depending on the heading error (ϕ) and vehicle speed

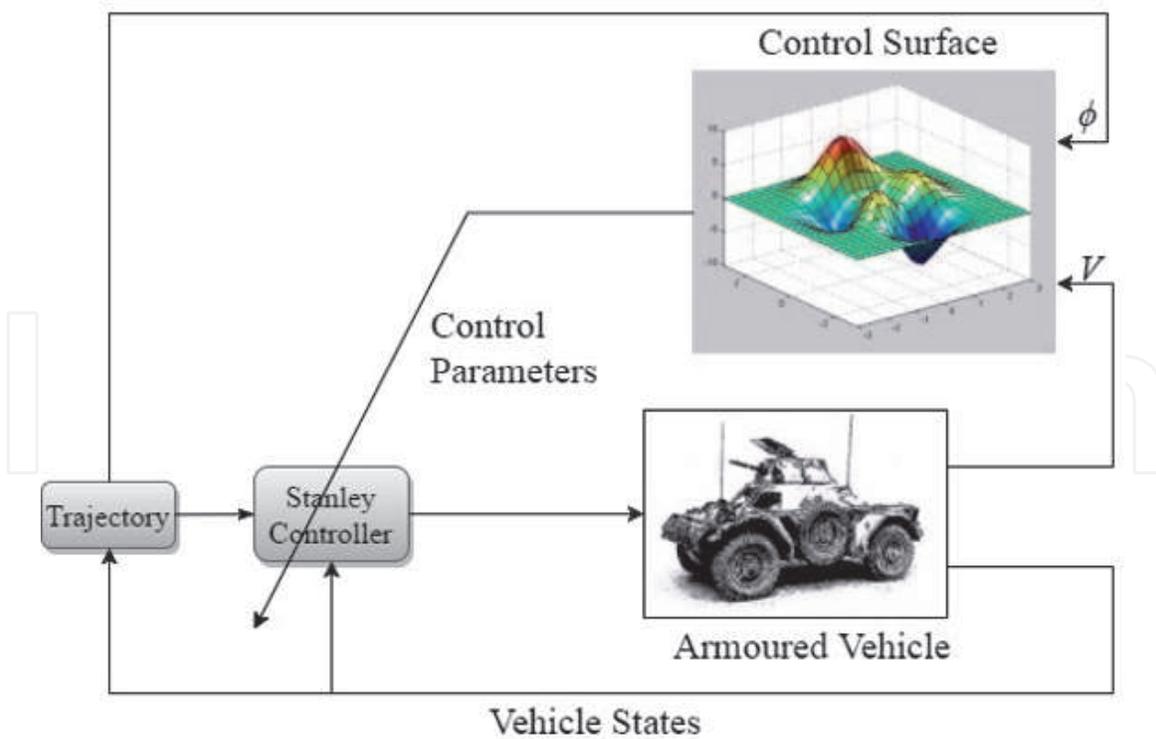


Figure 7.
 Structure of the adaptive Stanley controller.

v (m/s)	1	5	10	15	20									
ϕ (deg)	1	5	15	30	45	60	75	-1	-5	-15	-30	-45	-60	-75

Table 2.
 Interval for knowledge database for each input.

(v) is proposed in this study. These two variables are chosen since they represent the vehicle's instantaneous speed and the state of trajectory heading with respect to vehicle's heading at a particular time. By choosing this as the input for the adaptive mechanism, and optimising the controller at different values of these two states, the controller will be able to react to variable speed and trajectory it operates on. The overall structure for the knowledge-based adaptive Stanley controller is shown in **Figure 7**.

3.2.1 Building knowledge database using particle swarm optimisation

In this study, a knowledge database is built to provide artificial insights for the controller to choose a suitable set of parameter values for the controller based on the vehicle's driving conditions, namely, speed (v) and heading error (ϕ). The database should have a complete set of parameters (k_ϕ , k_1 , k , and k_ψ) that correspond to each range of v and ϕ .

Table 2 shows the interval values for input variables that were chosen for the knowledge database. In choosing the interval values, several sensitivity analyses were carried out to test the suitability of an optimised parameters set in varying speeds. It was found that each optimised parameters will yield satisfactory results within ± 5 m/s and ± 15 deg. Therefore, these intervals are chosen. A minimum of 1 m/s (≈ 3.6 km/h) and a maximum of 20 m/s (≈ 72 km/h) were chosen based on the common and safe operating speed for the vehicle. A minimum of 1 deg. and a

maximum of 75 deg. were chosen for ϕ values, which correspond to the minimum and maximum values of the cornering angle for the vehicle.

A metaheuristic optimisation algorithm, namely, PSO method, is used in building the knowledge database, as outlined in Amer et al. [13, 15] to optimise the four controller parameters for each of the v and ϕ combinations. Each of the four controller parameters is optimised for each of the v and ϕ combinations. PSO is one of the established optimisation algorithms based on a natural phenomenon. First introduced in 1995 [19], it is one of the many optimisation algorithms based on swarm intelligence available as research tools. A comprehensive study on PSO algorithm and other swarm intelligence algorithms with their associated optimisation performance can be found in [20]. The study has found that PSO has better performance in terms of solution consistencies and convergence than the other swarm algorithms. Zhang [21] has also stated that PSO has better convergence in achieving global optimum with better accuracy than conventional methods. Also, with a lesser number of algorithm parameters to tweak than genetic algorithm (GA), PSO is always preferable. Overall, it was found that PSO converged faster and the study concluded that PSO provides better accuracy with a fast convergence for a standard computing capability. The PSO performance has also been studied in various applications for controller parameter tuning such as [22]. Here, PSO was used to tune a model predictive controller, and the optimisation performance was compared against ant colony optimisation (ACO) and gravitational search algorithm (GSA). It was shown that PSO was more superior than the others with up to 90% quicker convergence than ACO and GSA. The PSO-optimised controller also performed better, showing that PSO managed to find better optimum points than its counterparts.

PSO algorithm simulates the behaviour of swarmed particles, which move within the swarm towards an intended position. Each particle will be assigned random positions within the swarm, and its next movement (position and velocity) will be determined based on its relative position to an optimum position. Optimum position will be the position of any particle with an optimum fitness value. Every particle will have its own fitness value and the memory of its own best position, p_{best} , as well as the overall swarm best position, g_{best} , based on the optimum fitness value. The particle's next position will be determined by considering these memories. This will guarantee that the particles will not be moving too quick towards the new position that can turn out to be another local optimum. In other words, any of the i_{th} particle within the swarm will have position coordinates, x_{id} , in each of the d_{th} dimensions and move with velocity v^t . Its next position, $x_{id}^{(t+1)}$, will be determined based on its next velocity $v^{(t+1)}$ shown in Eqs. (4) and (5), respectively. These processes will be iterated over several cycles, and finally, the swarm will converge to an optimum position:

$$x(t+1)_{id} = x(t)_{id} + v(t+1) \quad (4)$$

$$v^{(t+1)} = iw \times v^{(t)} + c \times rand(0,1) \times (p_{best}^{(t)} - x^{(t)}_{id}) + s \times rand(0,1) \times (g_{best}^{(t)} - x^{(t)}_{id}) \quad (5)$$

In this study, the optimisation process of Stanley controllers using PSO can be shown in **Figure 8**. Referring back to **Figure 6**, the upper part of the chart shows the optimisation process of each input combination using PSO, while **Table 3** tabulates the main parameters used in PSO algorithm. In addition, since lateral error is one of the most significant performance indices for a path tracking controller, fitness function for each particle will be evaluated by the lateral error between the vehicle

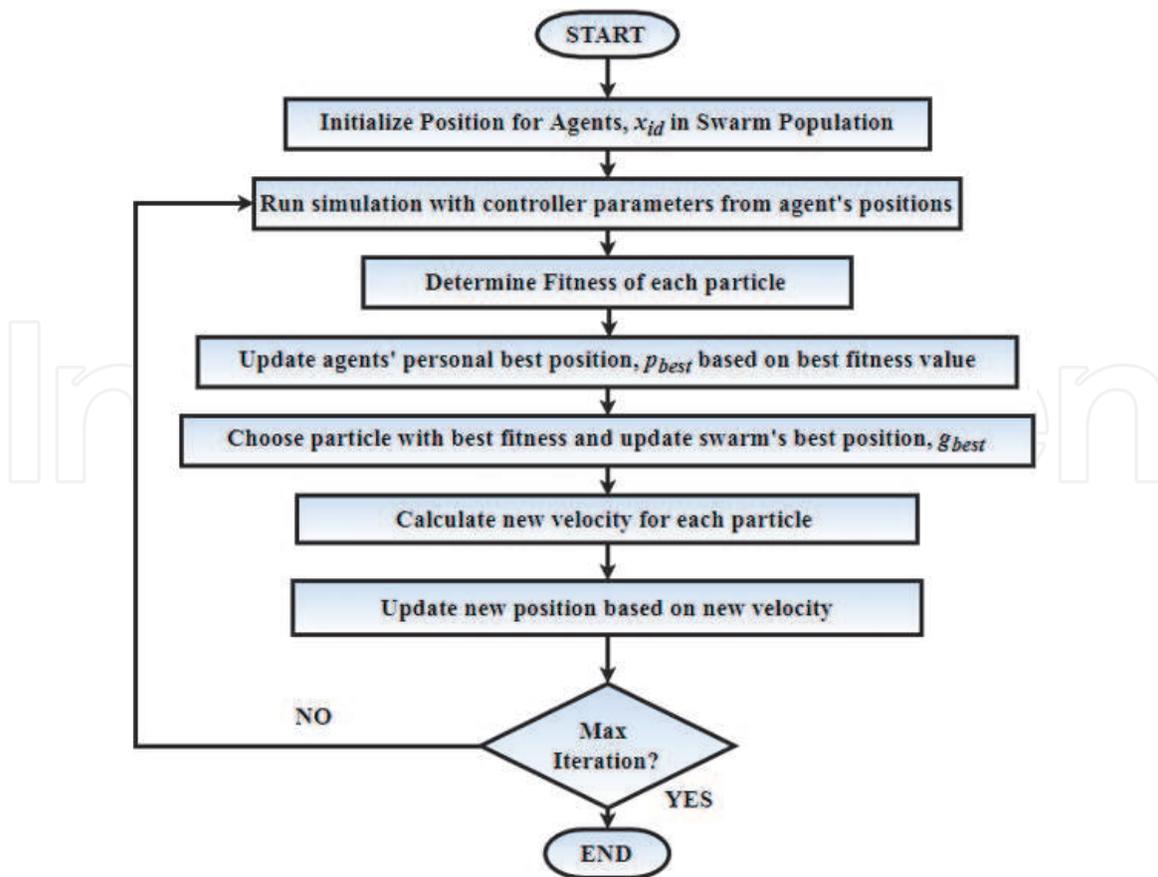


Figure 8.
 Procedure for PSO.

Parameter	Values
Social Coefficient, s	1.42
Cognitive Coefficient, c	1.42
Inertial Weight, iw	0.9
No. of Dimensions, N_d	4 ($k_l, k_\phi, k,$ and k_ψ)
Upper Bound Limit	[10;10;10;10]
Lower Bound Limit	[-10; -10; -10; -10]
No. of Particles, N_p	150
No. of Iterations, N_i	20

Table 3.
 PSO parameters used in building a knowledge database [13, 15].

and trajectory (e), which resulted from the parameter values for each particle's position. Lateral error is quantified by the root mean square (RMS) value, as shown in Eq. (6):

$$\text{Fitness Function, } f(k_f, k_l, k, k_\psi) = \sqrt{\frac{\sum [e(t)]^2}{n}} \quad (6)$$

Upon completion, all sets of parameters are compiled and integrated into a knowledge database consisting all parameters for each combination of ϕ and v

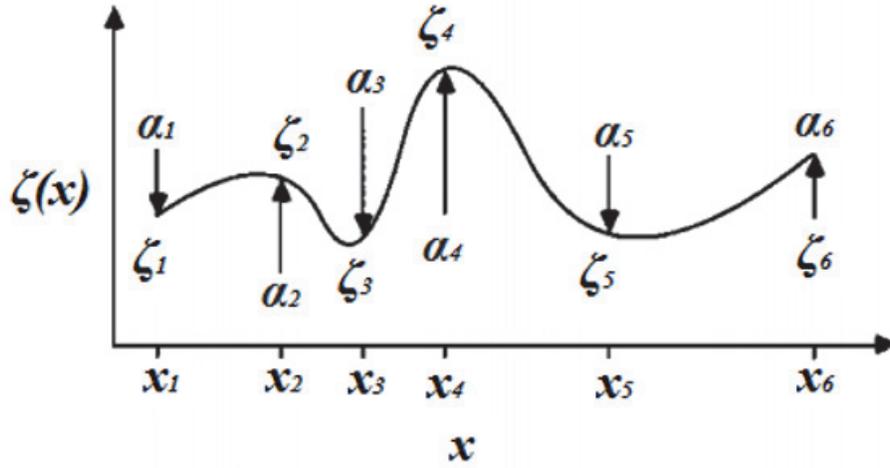


Figure 9.

Wrapping an imaginary spline around data points (x_i, ζ_i) with point forces (α_i) acting at each point.

values. The database is then used to form separate control surfaces for each corresponding parameter. For the proposed basic controller in Eq. (3), there are four parameters, which correspond to four separate control surfaces formed from the knowledge database. For the adaptive method, the controller parameters are adjusted by an algebraic algorithm that will interpolate a suitable set of parameter values based on the developed knowledge database. The interpolation will be on a set of control surfaces that map the optimum set of parameters according to its respective ϕ and v values.

3.2.2 Generating control surface from the knowledge database

In this study, the developed knowledge database in Section 3.2.1 is transformed into a set of control surfaces (Φ^k) that correspond to each of the four controller parameters $k_\phi, k_1, k,$ and k_ψ . The transformation of the database to control surfaces was carried out using the concept named biharmonic spline [23], originally used to deduce geographical contour from a set of altitude data of the topography. It worked by assuming an imaginary spline across a one-dimensional data point and applying a point force (α_j) to each of the data points with coordinates (x_j, ζ_j) . This can be shown in **Figure 9**.

Using the biharmonic spline method, the line in **Figure 9** satisfies a biharmonic partial differential equation (PDE) with $N = 6$ shown in Eq. (7). The solution of this PDE is satisfied with a point-force Green's function that is centred at each point as shown in Eq. (8). Solving for α_j in Eq. (9) will yield the strength of each point force (α_j) :

$$\frac{d^4 \zeta}{dx^4} = \sum_{j=1}^N 6\alpha_j \delta(x - x_j) \quad (7)$$

$$\zeta(x) = \sum_{j=1}^N \alpha_j |x - x_j|^3 \quad (8)$$

$$\zeta(x_i) = \sum_{j=1}^N \alpha_j |x_i - x_j|^3 \quad (9)$$

Extending this method to a three-dimensional space, one can assume spreading an imaginary surface over the points instead of a spline. In this study, the biharmonic interpolation concept is applied on the knowledge database, which consists of data that are scattered in a three-dimensional space. The control surface produced needs to pass through each of the data in the database using this concept

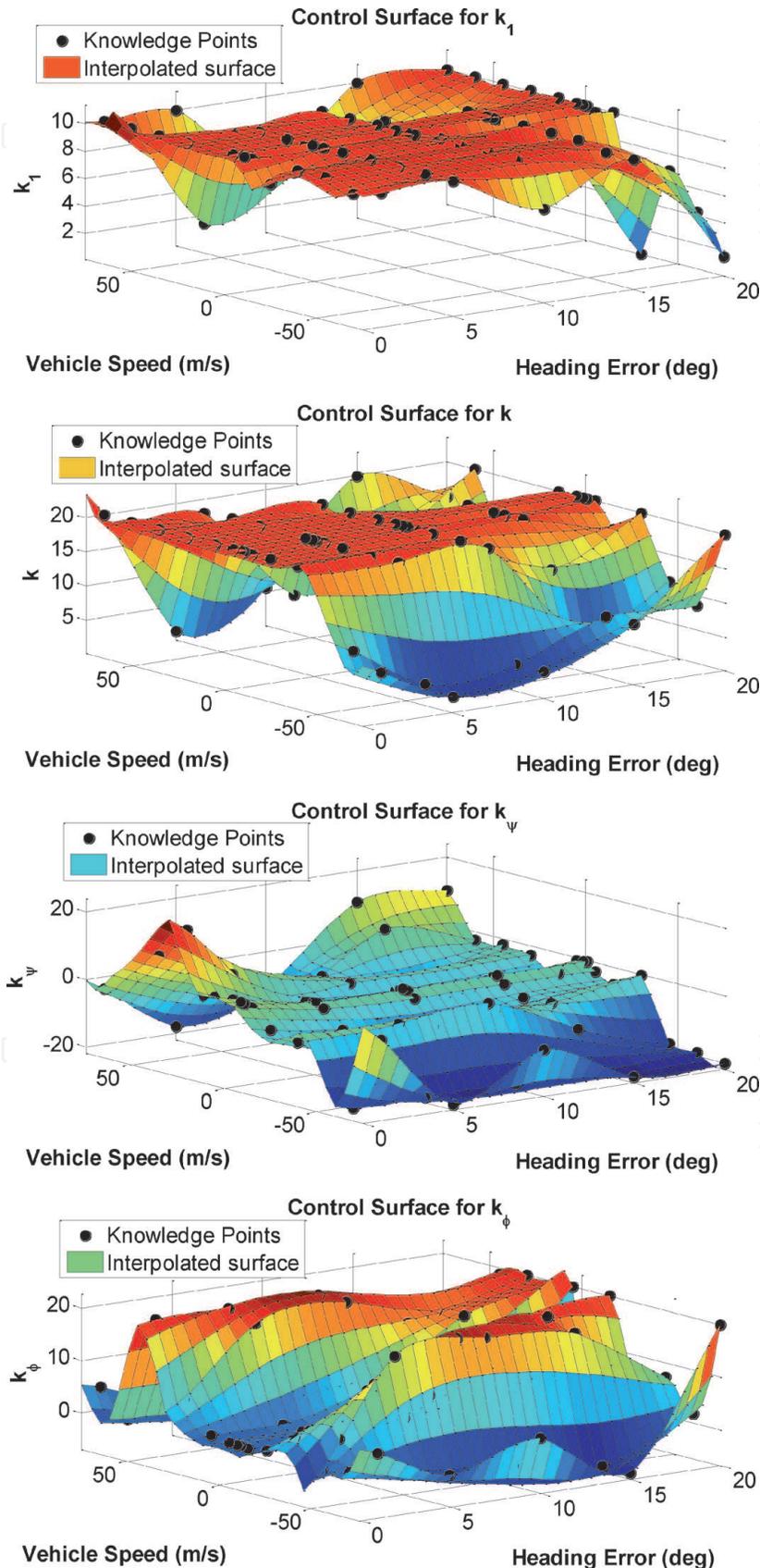


Figure 10.
Control surface for each controller parameter.

by wrapping an imaginary thin lamina around all the data points. The database is separated into four sets of data, which will generate four control surfaces, which corresponds to each controller parameter as shown in **Figure 10**. Each surface shows the interpolated surface relating to the control parameter values to its corresponding vehicle speed and heading error.

3.2.3 Using the control surface to generate an optimum set of parameters

Using the control surface generated previously, the interpolation algorithm will choose corresponding parameter values within the surface based on instantaneous heading error and vehicle's speed. This will adaptively tune the modified Stanley controller, adjusting it to various conditions during manoeuvring regardless of any path and vehicle speed. In cases where the v and ϕ values are outside of the database range from **Table 2** and surfaces from **Figure 10**, no extrapolation method will be used. Instead, parameter values at surface boundaries will be considered. This is to avoid the controller to enter an unstable controller region and ensure that the algorithm will continuously tune the parameters even when the v and ϕ values are outside of the range in the knowledge database. By denoting the k th control surface as $\Phi^k(v, \phi)$ where $k = 1, 2, 3, 4$ and K_k corresponds to the controller parameters k_ϕ , k_s , k , and k_v , the adaptive algorithm to automatically tune the controller parameters has been developed as a set of controller commands as shown in Eq. (10):

$$K_k = \left\{ \begin{array}{ll} \phi^k(\phi, V) & \text{if } \phi_{\min} \leq \phi \leq \phi_{\max}, \text{ AND } v_{\min} \leq v \leq v_{\max} \\ \phi^k(\phi, V_{\min/\max}) & \text{if } \phi_{\min} \leq \phi \leq \phi_{\max}, \text{ AND } v \leq v_{\min} \text{ OR } v \geq v_{\max} \\ \phi^k(\phi_{\min/\max}, V) & \text{if } \phi \leq \phi_{\min} \text{ OR } \phi \geq \phi_{\max}, \text{ AND } v_{\min} \leq v \leq v_{\max} \\ \phi^k(\phi_{\min/\max}, V_{\min/\max}) & \text{if } \phi \leq \phi_{\min} \text{ OR } \phi \geq \phi_{\max}, \text{ AND } v \leq v_{\min} \text{ OR } v \geq v_{\max} \end{array} \right\} \quad (10)$$

4. Simulation and experimental procedures

The effectiveness of the developed controller is evaluated through simulations within the MATLAB/Simulink using ODE solver settings, namely, Heun ODE2 and a fixed-step size of 0.001 s. The controller is implemented on the validated vehicle model from Section 2 that was set to enter each of the trajectories in **Figure 4** with an initial velocity, zero brake, and throttle settings to keep the vehicle at a constant speed of 6 m/s. The adaptive controller is compared against the two of its predecessors, namely, the Mod St from Eq. (3) and the original Stanley controller as shown in Eq. (1). Each of these controllers has been optimised using PSO with similar procedures as described in Section 3.2.1 and simulated using the same simulation settings. In evaluating the controller's performance under varying input disturbances, a set of simulations were carried out using a different trajectory as shown in **Figure 4** as well as different speed settings. The vehicle was set to travel with different speed values that are chosen randomly within and outside of the range of the knowledge database.

4.1 Experimental validation against the human driver

The adaptive path tracking controller developed in this study is validated experimentally against the human-driven responses when navigating similar paths. For this, an instrumented heavy vehicle was driven, and the driver's steering input was recorded. The same manoeuvring was simulated using the developed controller,

and the resulted steering input from the controller was compared against the human's driving input, to observe the validity of the automatic steering input from the controller.

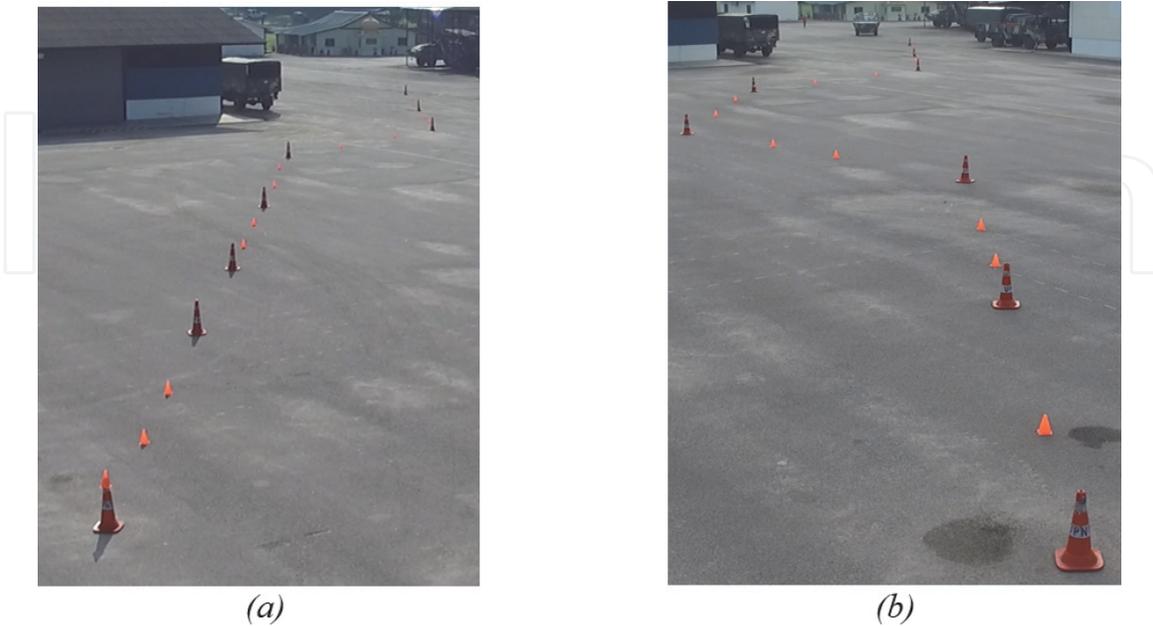


Figure 11.
Road course used in the experiment: (a) single lane change and (b) double lane change.

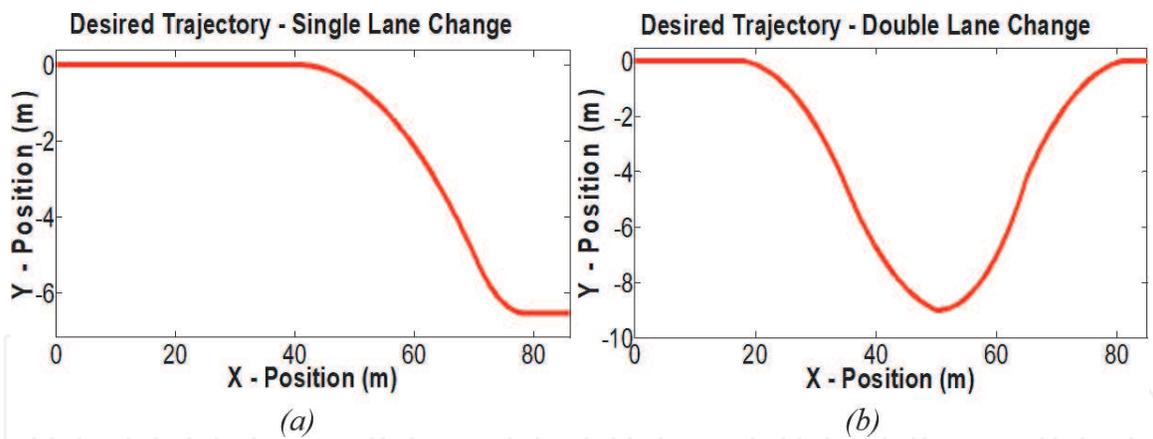


Figure 12.
Road course for simulations taken from real dimensions for (a) single lane change and (b) double lane change.

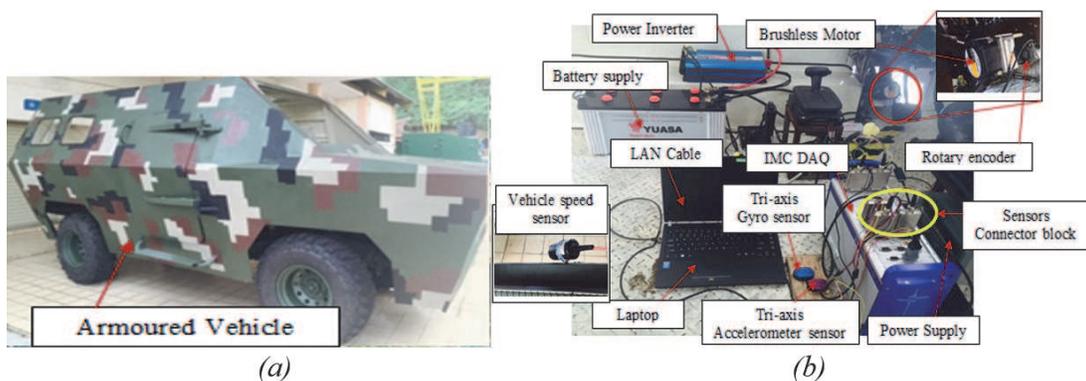


Figure 13.
(a) Prototype of heavy vehicle and (b) sensor configuration in driver's cockpit.

In the experiment, two types of manoeuvrings were conducted, namely, single and double lane changes. The dimensions of the desired road were set on a level, open space as shown in **Figure 11** for both single and double lane change manoeuvrings, based on ISO-3888 [24]. These dimensions were converted into a series of coordinate points for simulation stages as shown in **Figure 12**. As one can see, the dimensions are quite short due to the limited space available. For this, only 20 km/h speed was chosen to ensure good manoeuvrings.

The manoeuvring tests are carried out with an instrumented prototype of a heavy vehicle as shown in **Figure 13**. The prototype was installed with various sensors and actuators as shown. However, for this experiment, the main sensors used are the accelerometer, gyro sensor, and rotary encoder. The rotary encoder will record the driver's steering input from steering column rotation, and from this, the wheel angle data can be deduced. Meanwhile, position data is acquired by a built-in GPS sensor in IMC DAQ that is attached to the roof of the vehicle to ensure good data transmission. Should the GPS fail, one can use the recorded acceleration data from the accelerometer, as well as vehicle speed data from the speed sensor to calculate the vehicle position in local coordinates.

From this experiment, automated manoeuvrings from the controller will be compared against the actual driving inputs from the human driver. Two observations can be deduced due to the nature of the short trajectories. First, the controller's ability in mimicking human driver behaviour can be evaluated. Second, the path tracking performance of the controller in guiding the heavy vehicle along short roads and sharp cornering can be evaluated also. The findings will be detailed in the next section.

5. Results and discussions

In evaluating the performance of the proposed adaptive controller, trajectory tracking performance was observed. Two aspects were compared, namely, (a) the vehicle trajectories and (b) the lateral error while navigating the trajectories with each respected controller. The performance of the proposed knowledge-based adaptive controller was compared against its two predecessors, which are the original St controller from Eq. (1) and Mod St from Eq. (3) as explained in Section 3.1. All simulation results for the six trajectories are shown in **Figure 14** for the straight road, **Figure 15** for the Multiple Lane Change Road, **Figure 16** for the Double Lane Change Road, **Figure 17** for the Hook Road, **Figure 18** for the S Road, and **Figure 19** for the Curved Highway Road.

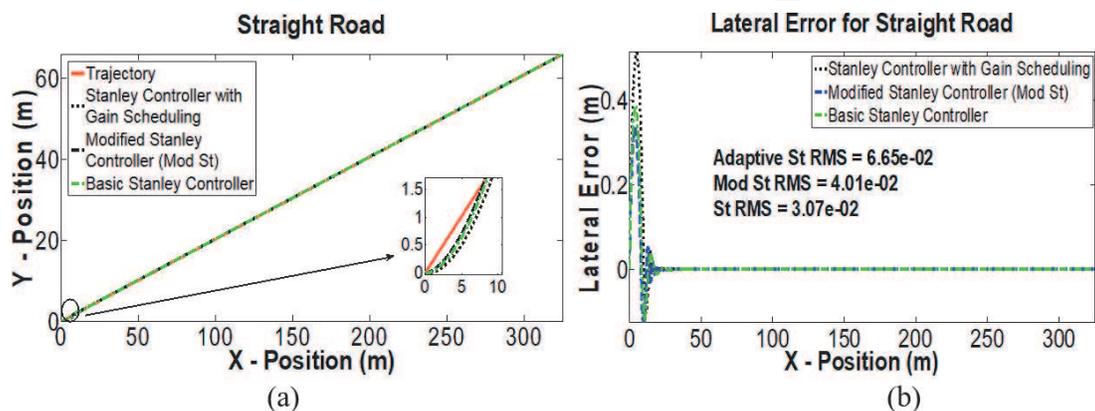


Figure 14. Trajectory tracking performance for straight road: (a) vehicle trajectories, (b) lateral error, e.

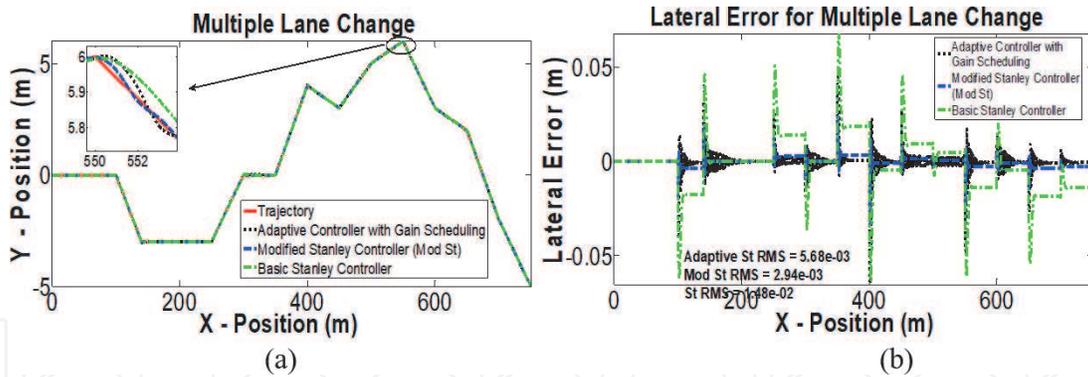


Figure 15. Trajectory tracking performance for curved highway road: (a) vehicle trajectories, (b) lateral error, e.

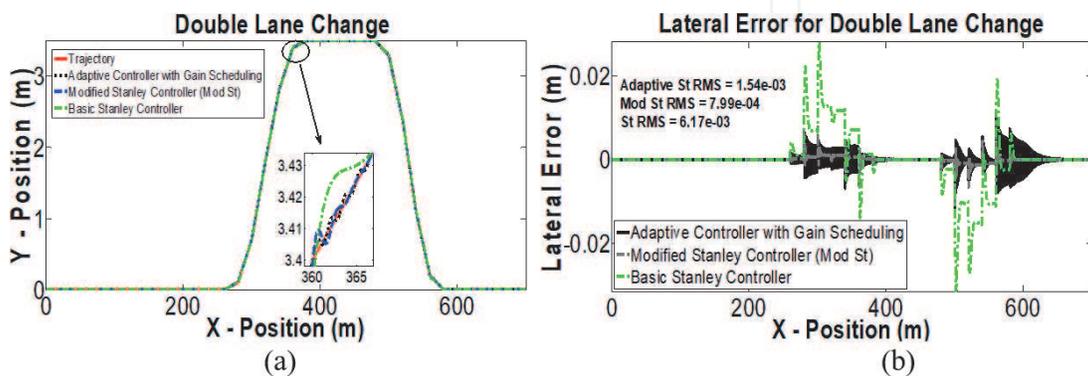


Figure 16. Trajectory tracking performance for multiple lane change road: (a) vehicle trajectories, (b) lateral error, e.

It can be seen that the proposed controller managed to guide the heavy vehicle along the desired trajectories successfully. Overall, the adaptive controller performed significantly better than the St controller. The proposed controller managed to guide the vehicle with better precision closer to the path, which can be indicated further by the lateral error graphs in (b). These improvements are mainly caused by the fact that the adaptive controller adopts Mod St steering command from Eq. (3) as the base controller to be automatically tuned. This controller considers the yaw rate error feedback, which can improve the overall trajectory tracking performance. Also, it has more controller parameters, which made the controller more sensitive to tuning, which, in turn, improve the tracking performance upon parameter selection. This also explains the exceptional performance by the Mod St controller shown in the graphs. For shorter and tight manoeuvring road

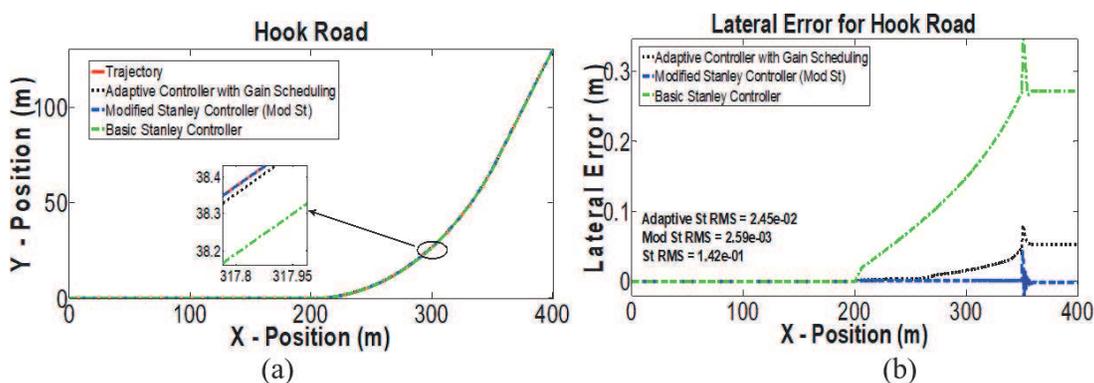


Figure 17. Trajectory tracking performance for double lane change road: (a) vehicle trajectories, (b) lateral error, e.

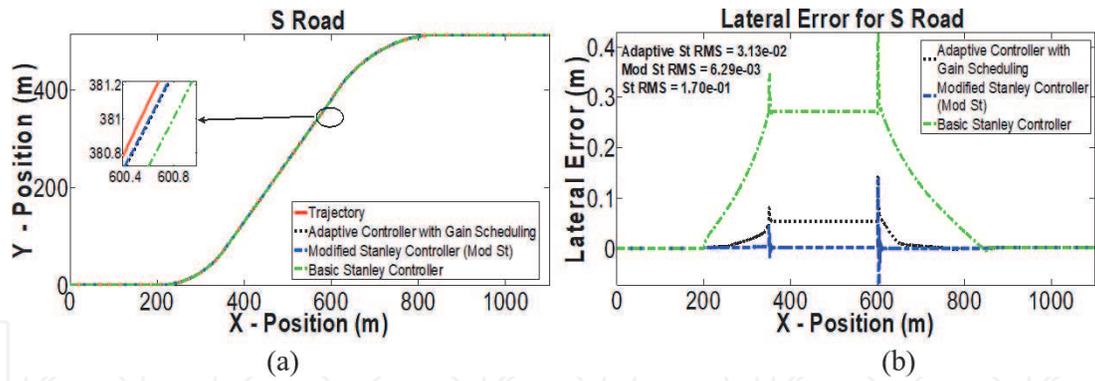


Figure 18.

Trajectory tracking performance for hook road: (a) vehicle trajectories, (b) lateral error, e .

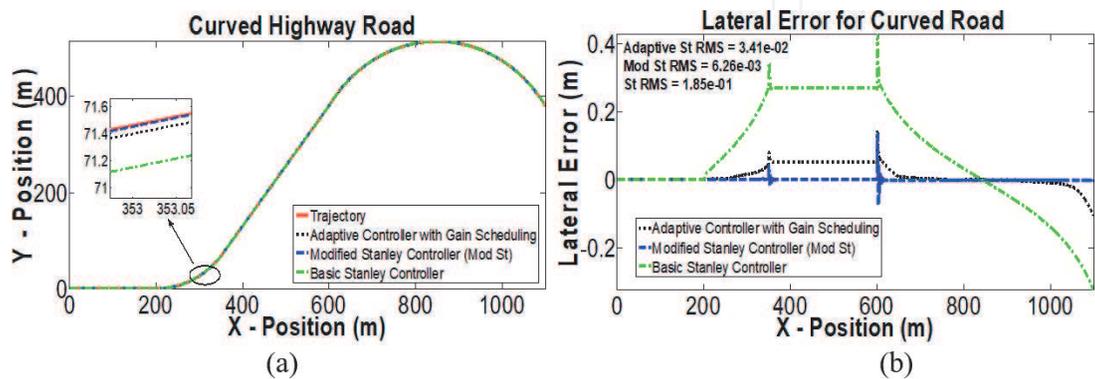


Figure 19.

Trajectory tracking performance for S road: (a) vehicle trajectories, (b) lateral error, e .

courses, namely, straight, multiple lane change, and lane change, shown in **Figures 14–16** respectively, the vehicle was manoeuvred successfully along the intended road courses with significantly better lateral error than the original Stanley controller. However, unwanted oscillations can be observed in **Figures 16** and **17** due to rapid cornering that exists in these courses. This can be minimised by preparing smoother curvature for the vehicle to follow [6].

However, one might notice the inferior performance shown by the proposed controller when compared against the Mod St controller. As stated before, this is the base controller where the adaptive algorithm was built on. In the simulation, the Mod St controller was tuned specifically for each trajectory using a metaheuristic optimisation algorithm, namely, PSO. The procedures are explained by Amer et al. [15]. Since the controller was specifically tuned for each trajectory and the 6 m/s speed, the controller parameter has been chosen to optimise the vehicle performance for each of the roads. This explains the fact that the base controller performed better than the adaptive controller, which was automatically tuned by the adaptive algorithm. However, despite the inferior performance compared to the Mod St controller, the adaptive controller still managed to guide the vehicle with a satisfactory performance. Looking at the RMS values for lateral error, the adaptive controller recorded a lateral error of 0.00154–0.0341 m across all the six trajectories, which are well below the average lateral error of 0.1 m recorded by the Stanley vehicle in the original publication [3]. Therefore, it can be concluded that the proposed adaptive controller performed well in navigating various trajectories. Overall comparison results for the RMS values on lateral error between the evaluated controllers are listed in **Table 4**.

The response of the controller under various vehicle speeds was studied next for hook, S, and curved highway road, as shown in **Figure 20**. These roads were chosen

	Adaptive	Mod St	St	% Diff w.r.t. Mod St	% Diff w.r.t. St
Straight	3.11E-02	5.43E-02	8.85E-02	-74.60	-64.86
Multiple Lane Change	5.68E-03	2.94E-03	1.48E-02	48.24	-61.62
Double Lane Change	1.54E-03	7.99E-04	6.17E-03	48.12	-75.04
Hook	2.45E-02	2.59E-03	1.42E-01	89.43	-82.75
S	3.13E-02	6.29E-03	1.70E-01	79.90	-81.59
Curved	3.41E-02	6.26E-03	1.85E-01	81.64	-81.57

Table 4.
 Comparison of RMS values for lateral error between the controllers.

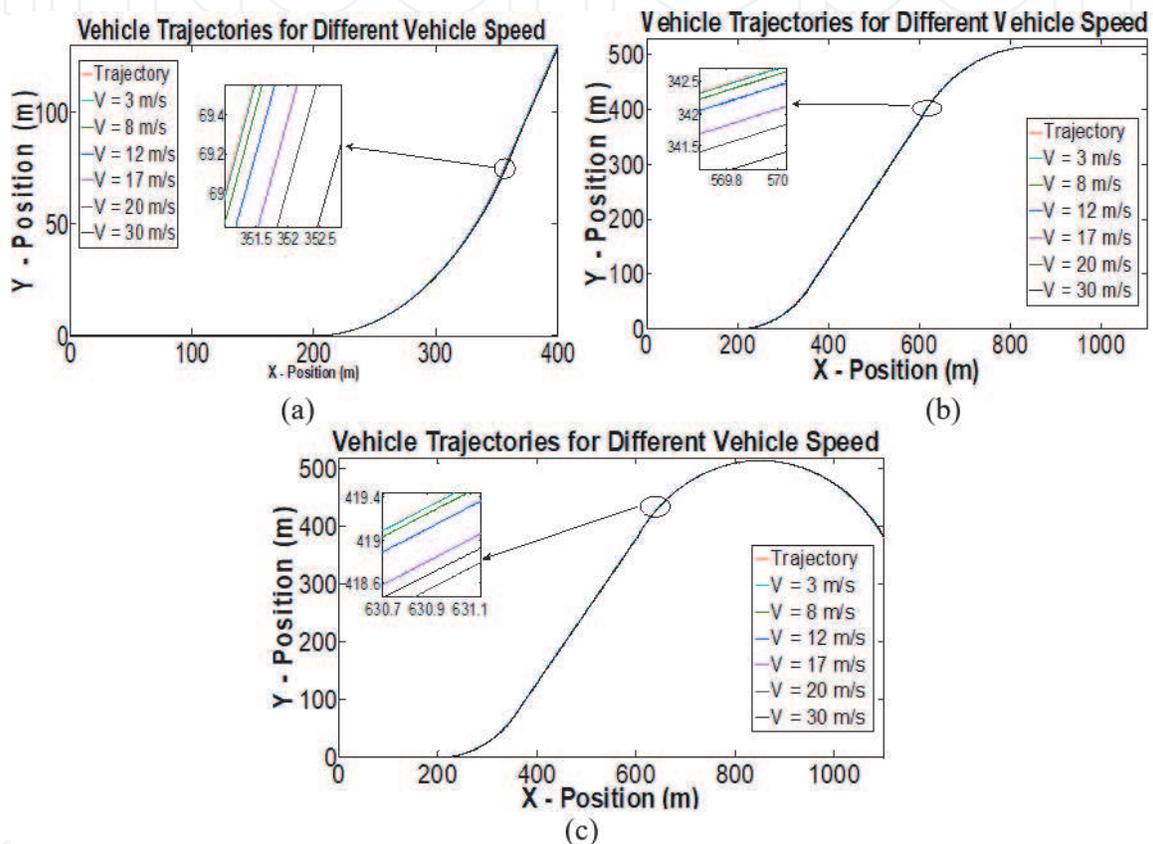


Figure 20.
 Effect of varying speeds for (a) hook road, (b) S road, and (c) curved highway road.

due to its suitability for a high-speed testing. Straight, multiple lane change, and double lane change roads have shorter courses and extreme manoeuvres, which are not suitable for a high-speed testing. In each analysis, six constant speed values were chosen to evaluate the controller, namely, 3, 8, 12, 17, 20, and 30 m/s. Based on knowledge database range and intervals listed in **Table 2**, 20 and 30 m/s were chosen to observe the controller's behaviour on the knowledge database boundary and outside the boundary. Other values were chosen randomly to observe the controller's performance with vehicle speeds well within the knowledge database boundary. From the figures, one can see that the proposed adaptive controller managed to steer the heavy vehicle well along the desired trajectory. However, as the vehicle speeds increase, larger error was observed since the vehicle is moving further than the intended trajectory. This is understandable since an increasing speed means that the vehicle can be diverted faster. Nevertheless, the controller still managed to bring the vehicle back to its intended direction with lateral error of well within 1 m as shown in **Table 5**.

Vehicle Constant Speed (m/s)

	3	8	12	17	20	30
Hook	0.0089	0.0511	0.1448	0.3290	0.4721	0.7547
S	0.0100	0.0625	0.1728	0.3976	0.5706	0.8488
Curve	0.0098	0.0676	0.1870	0.4337	0.6163	0.9322

Table 5.
RMS values of lateral error for various vehicle speeds with the adaptive controller.

5.1 Validation of steering angle against the human driver

For this analysis, three responses were observed, namely, vehicle trajectories, lateral error, and vehicle wheel angle throughout the manoeuvres, which are single lane change and double lane change from **Figure 11**, and described in detail in Section 4.1. The actual data from the instrumented vehicle are compared against the simulation results with the adaptive controller using the same trajectory and road courses in **Figure 12** and constant vehicle speed of 20 km/h. **Figure 21** shows the comparison between experimental results by the human driver and simulation results by the adaptive controller for the single lane change manoeuvring. Meanwhile, **Figure 22** shows the same comparison for the double lane change manoeuvring.

In both figures, graphs in (a) show the vehicle trajectories, which indicate that the proposed controller managed to automatically provide correctional steering input in guiding the vehicle closer to the desired trajectories. There is a noticeable

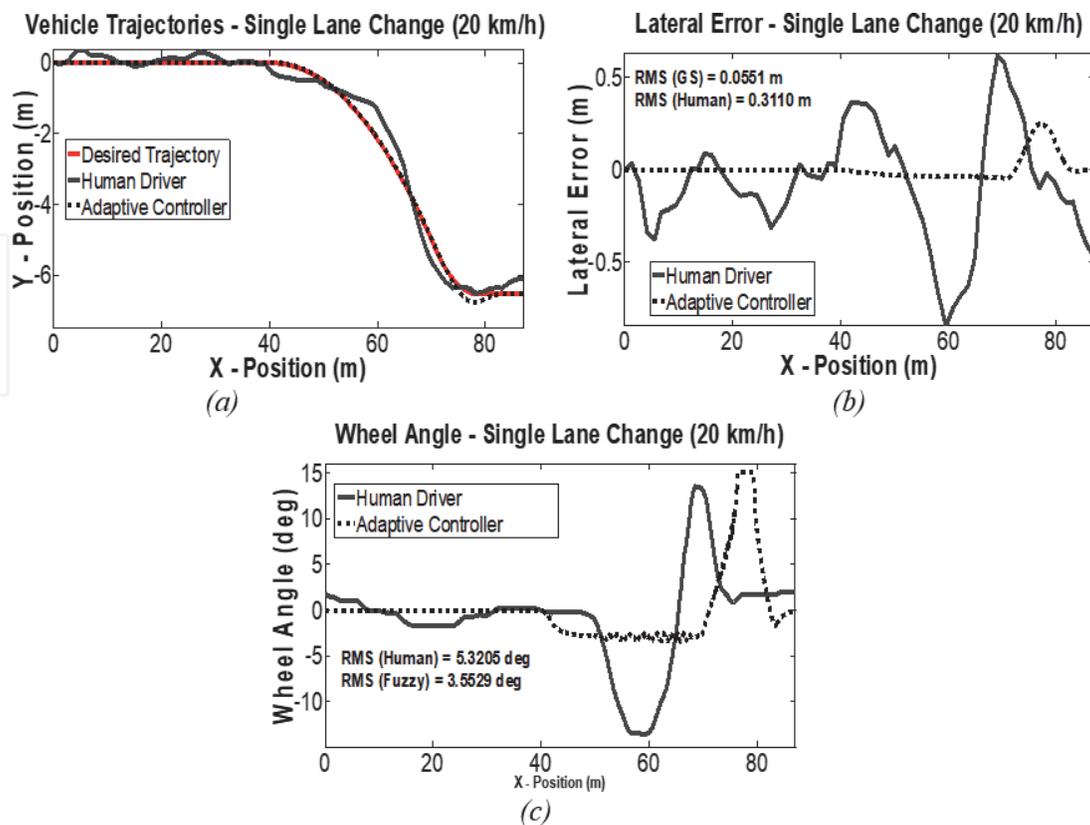


Figure 21.
Comparison between experimental results and simulations for single lane change road: (a) vehicle trajectory, (b) lateral error w.r.t. trajectory, (c) steering input required.

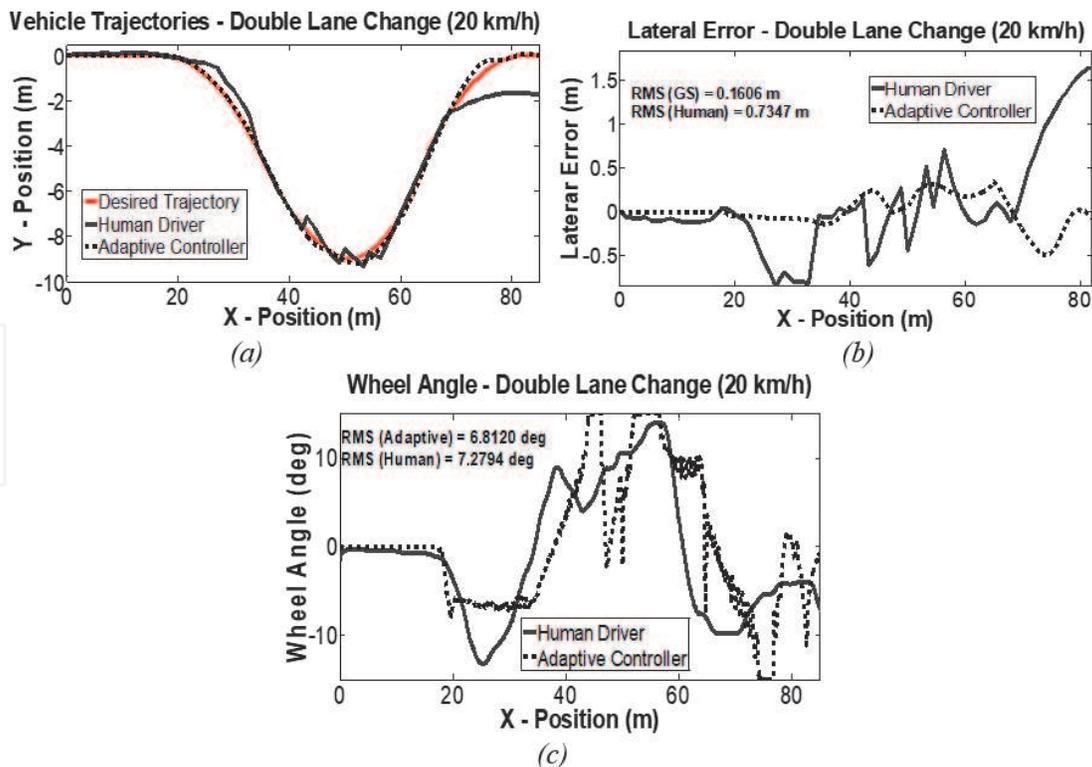


Figure 22. Comparison between experimental results and simulations for double lane change road: (a) vehicle trajectory, (b) lateral error w.r.t. trajectory, (c) steering input required.

overshoot in vehicle trajectory from simulation ($X = 80$ m) for both manoeuvres, which can be attributed to the speed of the vehicle. In the experiment, it was hard to keep the constant speed, and human instinct has caused the driver to slow down the vehicle while navigating sharp corners, whereby the speed was kept constant throughout the simulations. This has caused dissimilarities between the two results. Also, the road courses used here are very short, which may account to the inabilities of the controller, as well as the human driver, to perfectly track the trajectories. Nevertheless, the controller did well in steering the vehicle. This can be further studied through the lateral error results shown in graphs (b). The adaptive controller recorded better lateral error 82 and 78% less RMS values for single lane change and double lane change manoeuvres, respectively.

Looking at the wheel angle from graphs (c), one can observe the correctional steering input provided by the human driver in the experiment, as well as the inputs from the adaptive controller in simulation. Compared against the human driver for the same manoeuvres, the controller provides the steering input to the wheel with the same trend and input shape as the human driver, but with a faster response. Having a fast controller is always advantageous for any unaccountable delays and uncertainties that may happen in real implementation. Also, the controller is able to adapt to various speeds and trajectories while mimicking the human driver actions, which was proven from the results presented here.

6. Conclusions

In this study, an adaptive controller for an autonomous heavy vehicle is presented. The controller was developed based on an established Stanley controller that was modified to increase its sensitivity to the parameter changes. An adaptive algorithm was constructed to automatically tune the controller parameters based on the instantaneous vehicle speeds, v , and heading error, ϕ , between the vehicle and road course

trajectory. In constructing the adaptive algorithm, a knowledge database was built by optimising a set of parameters for the modified Stanley controller that corresponds to various combinations of v and ϕ using particle swarm optimisation.

The developed controller was applied on a validated 7DOF, nonlinear heavy vehicle model. Six trajectories were chosen representing long and short courses as well as courses with large and very small turning curvatures. With these, a series of simulations were carried out, and the performance of the proposed controller was compared against the basic original Stanley and also the modified Stanley controllers.

From the simulations, it was found that the proposed adaptive controller performed well in guiding the vehicle along all trajectories. It recorded significantly lower lateral error RMS between 61 and 82% improvement when compared against the original Stanley controller. However, 48–89% increases in lateral error RMS were observed when comparing the controller against the modified Stanley controller. This can be explained by the fact that the Mod St controller was optimised specifically for the respective courses and, therefore, performed exceptionally well on the roads.

In terms of the controller's ability in adapting to various trajectories and vehicle speeds, the controller was tested with various vehicle speeds within and outside of the knowledge database region. It was capable of navigating the vehicle smoothly regardless of the vehicle speed, with understandably larger error in maximum speed. However, the overall lateral error was well kept below 1 *m*. While the maximum testing speed was 30 m/s, the autonomous heavy vehicle will be operating with speeds much lower than that. Previous studies and implementations of autonomous passenger's vehicles have been recorded to operate with about 10 m/s and lower [4, 25, 26]. In this study, the proposed adaptive controller has managed to perform well without depending on any planner to provide a smooth trajectory. This ensures the applicability of the proposed controller to be operated on a heavy vehicle on various trajectories and vehicle speed values.

Acknowledgements

The authors would like to thank the Malaysian Ministry of Education through research grants LRGS (LRGS/B-U/2013/UPNM/DEFENCE and SECURITY-P1).

IntechOpen

IntechOpen

Author details

Noor Hafizah Amer^{1,2*}, Khisbullah Hudha¹, Hairi Zamzuri², Vimal Rau Aparow³, Amar Faiz Zainal Abidin⁴, Zulkiffli Abd Kadir¹ and Muhamad Murrad¹

1 Faculty of Engineering, Universiti Pertahanan Nasional Malaysia, Kuala Lumpur, Malaysia

2 Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, Kuala Lumpur, Malaysia

3 Faculty of Engineering, The University of Nottingham Malaysia, Selangor, Malaysia

4 Faculty of Engineering Technology, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia

*Address all correspondence to: noorhafizah@upnm.edu.my

IntechOpen

© 2020 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Paden B, Cap M, Yong SZ, Yershov D, Frazzoli E. A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on Intelligent Vehicles*. 2016;**1**(1):33-55
- [2] Amer NH, Zamzuri H, Hudha K, Kadir ZA. Modelling and control strategies in path tracking control for autonomous ground vehicles: A review of state of the art and challenges. *Journal of Intelligent and Robotic Systems*. 2017;**86**(2):225-254
- [3] Hoffmann GM, Tomlin CJ, Montemerlo D, Thrun S. Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing. In: *American Control Conference*, New York, USA. Canada, USA: IEEE; 2007. pp. 2296-2301
- [4] Thrun S et al. Stanley: The robot that won the DARPA Grand Challenge. *Journal of Field Robotics*. 2006;**23**(9): 661-692
- [5] Buehler M, Iagnemma K, Singh S. *The 2005 Darpa Grand Challenge: The Great Robot Race*. Berlin Heidelberg: Springer Science+Business Media, Springer-Verlag; 2007
- [6] Snider JM. *Automatic Steering Methods for Autonomous Automobile Path Tracking*. Pittsburgh, Pennsylvania, USA: Robotics Institute, Carnegie Mellon University, Technical Report CMU-RI-TR-90-17; 2009
- [7] Lucet E, Lenain R, Grand C. Dynamic path tracking control of a vehicle on slippery terrain. *Control Engineering Practice*. 2015;**42**:60-73
- [8] Hima S, Lusseti B, Vanholme B, Glaser S, Mammari S. Trajectory tracking for highly automated passenger vehicles. In: *International Federation of Automatic Control (IFAC) World Congress*, Vol. 44, No. 1, Milan, Italy. Netherlands: Elsevier; 2011. pp. 12958-12963
- [9] Pratama PS et al. Adaptive backstepping control design for trajectory tracking of automatic guided vehicles. In: Duy HV, Dao TT, Zelinka I, Choi H-S, Chadli M, editors. *AETA 2015: Recent Advances in Electrical Engineering and Related Sciences*. Cham: Springer International Publishing; 2016. pp. 589-602
- [10] Li X, Wang Z, Zhu J, Chen Q. Adaptive tracking control for wheeled mobile robots with unknown skidding. In: *IEEE Conference on Control Applications (CCA)*. Sydney, NSW, Australia: IEEE; 2015. pp. 1674-1679
- [11] Kadir ZA, Mazlan SA, Zamzuri H, Hudha K, Amer NH. Adaptive fuzzy-PI control for active front steering system of armoured vehicles: Outer loop control design for firing on the move system. *Strojnicki Vestnik-Journal of Mechanical Engineering*. 2015;**61**(3): 187-195
- [12] Aparow VR, Hudha K, Ahmad MMHM, Jamaluddin H. Development and verification of 9-DOF armored vehicle model in the lateral and longitudinal directions. *Jurnal Teknologi*. 2016;**78**(6):117-137
- [13] Amer NH, Zamzuri H, Hudha K, Aparow VR, Kadir ZA, Abidin AFZ. Modelling and trajectory following of an armoured vehicle. In: *2016 SICE International Symposium on Control Systems (ISCS)*, Nagoya, Japan. Canada, USA: IEEE; 2016. pp. 1-6
- [14] Amer NH, Zamzuri H, Hudha K, Aparow VR, Kadir ZA, Abidin AFZ. Path tracking controller of an autonomous armoured vehicle using modified Stanley controller optimized

with particle swarm optimization. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*. 2018;**40**(2):104

[15] Amer NH, Hudha K, Zamzuri H, Aparow VR, Kadir ZA, Abidin AFZ. Hardware-in-the-loop simulation of trajectory following control for a light armoured vehicle optimised with particle swarm optimisation. *International Journal of Heavy Vehicle Systems*. 2019;**26**:663-691

[16] Amer NH. Adaptive Stanley Controller with Fuzzy Supervisory System for Trajectory Tracking Control of an Armoured Vehicle. Malaysia: Malaysia-Japan International Institute of Technology Universiti Teknologi Malaysia; 2018

[17] Shan Y, Yang W, Chen C, Zhou J, Zheng L, Li B. CF-pursuit: A pursuit method with a clothoid fitting and a fuzzy controller for autonomous vehicles. *International Journal of Advanced Robotic Systems*. 2015; **12**(134):1-13

[18] Wallace R, Stentz A, Thorpe CE, Maravec H, Whittaker W, Kanade T. First results in robot road-following. In: *Proceedings of the 9th International Joint Conference on Artificial Intelligence, Vol. 2, Los Angeles, California, USA*. USA: Morgan Kaufmann Publishers Inc; 1985. pp. 1089-1095

[19] Eberhart RC, Kennedy J. A new optimizer using particle swarm theory. In: *Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Vol. 1, New York, USA*. Canada, USA: IEEE; 1995. pp. 39-43

[20] Ab Wahab MN, Nefti-Meziani S, Atyabi A. A comprehensive review of swarm optimization algorithms. *PLOS One*. 2015;**10**(5):e0122827

[21] Zhang K. A minimum zone method for evaluating straightness errors using PSO algorithm. In: *Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues*. Berlin, Heidelberg: Springer; 2008. pp. 308-314. DOI: 10.1007/978-3-540-87442-3_39

[22] Merabti H, Belarbi K, Bouchemal B. Nonlinear predictive control of a mobile robot: A solution using metaheuristics. *Journal of the Chinese Institute of Engineers*. 2016;**39**(3):282-290

[23] Sandwell DT. Biharmonic spline interpolation of GEOS-3 and SEASAT altimeter data. *Geophysical Research Letters*. 1987;**14**(2):139-142

[24] International Organization for Standardization. Passenger Cars - Test Track for a Severe Lane-Change Manoeuvre. ISO-3888. 1999. Available from: <https://www.iso.org/standard/67973.html>

[25] Buehler M, Iagnemma K, Singh S. *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*. Berlin Heidelberg: Springer-Verlag; 2009

[26] Wang X, Fu M, Ma H, Yang Y. Lateral control of autonomous vehicles based on fuzzy logic. *Control Engineering Practice*. 2015;**34**:1-17