

# An Event-Driven Adaptive Cruise Controller

Jessica Jreijiry and Mohamad Khaldi<sup>(✉)</sup>

Department of Electrical Engineering, Faculty of Engineering,  
University of Balamand, P.O. Box 100, Tripoli, Lebanon  
mohamad.khaldi@balamand.edu.lb

**Abstract.** The proposed Event-Driven Adaptive Cruise Controller (EDACC) serves as longitudinal driver assistant by accelerating, decelerating, and stopping the host vehicle given the readings of various sensors. EDACC uses a simplified non-linear longitudinal vehicle model and a hierarchical control structure of PI and PID controllers integrated with an embedded specific logic. In addition to the adaptability of the ACC, events can be added such as host vehicle entering speed-limit zone and/or having punctured tire. The proposed EDACC is designed and implemented using Matlab and Simulink to simulate multiple scenarios.

**Keywords:** Adaptive Cruise Control · Event-driven systems · Longitudinal vehicle dynamics

## 1 Introduction

Adaptive Cruise Control (ACC) has the purpose of maintaining the speed of the host vehicle at a speed set by the driver as a conventional cruise control system would. ACC also has the ability to detect a slower leading vehicle and follow it at constant speed equal to that of the leading vehicle while maintaining a constant relative time headway policy.

The proposed Event-Driven ACC (EDACC) serves as longitudinal control driver assistant by accelerating, decelerating, and stopping the host vehicle given the measurements of the range sensors that measure the relative speed and distance of the leading vehicle [1]. Pressure sensors that can detect the presence of a tire puncture or leak and pre-entered data combined with GPS technology that transfers the speed limit zone to the EDACC if and when the vehicle enters a speed limit zone [2]. EDACC does not attempt any lateral control, such as lane change. The speed of the host vehicle is specified by the controller depending on the specified mode of operation by the logic incorporated in the Stateflow Chart. If the logic leads to conventional cruise mode, the controller will only attempt to achieve a constant vehicle speed equal to that set by the driver. If the logic leads to vehicle following mode, the controller will attempt to maintain a speed equal to the leading vehicle, i.e. a null relative speed, while reserving an inter-vehicle distance with constant time headway policy, where the constant time headway is specified by the driver. The controller must achieve the desired distance and speed, without colliding with the leading vehicle and without using excessive acceleration and deceleration that may discomfort or endanger passengers. Accelerating and

decelerating the vehicle may lead to discomfort and probably to safety endangerment. The controllers manipulate the throttle and brake and hard braking may lead to swaying and in some cases sliding [3, 4].

There are many controllers which have been proposed for ACC [5–7]. From Proportional-Integral (PI), Proportional-Integral-Derivative (PID), and Constant Time Gap (CTG) controllers in [6], to Sliding Mode Control in [7] and Model-Predictive Control (MPC) in [5]. There are also different inter-vehicle distance policies, such as constant spacing, constant time headway policy, and variable time gap policy [6–8]. Constant spacing policies are deemed to be unsuitable for autonomous control applications [9], since they do not ensure string stability. The main purpose of this study is the introduction of discrete-events to the conventional ACC model.

## 2 Vehicle Model

The vehicle model, introduced in this study, is constituted of several subsystems, divided into two large groups, longitudinal vehicle dynamics and driveline dynamics. Longitudinal vehicle dynamics present all external forces acting on the vehicle; rolling resistance, gravitational, aerodynamic drag, and longitudinal tire forces, while driveline dynamics present the internal forces; engine, transmission and wheel dynamics.

### 2.1 Longitudinal Vehicle Dynamics

The application of Newton's law presents the longitudinal dynamics equation [6],

$$m\ddot{x} = F_x - F_{ad} - R_x - F_g \quad (1)$$

where  $F_x$  is the longitudinal tire force,  $F_{ad}$  is the equivalent longitudinal aerodynamic drag force,  $R_x$  is the force due to the rolling resistance at the tires,  $F_g$  is the gravitational force given by  $mg \sin \theta$ , where  $m$  is the mass of the vehicle,  $g$  is the acceleration due to gravity, and  $\theta$  is the road inclination angle. The angle  $\theta$  is defined to be positive clockwise when the longitudinal direction of motion  $x$  is towards the left. It is defined to be positive counter clockwise when the longitudinal direction of motion  $x$  is towards the right.

The aerodynamic drag force can be determined through the following function [6],

$$F_{ad} = \frac{1}{2} \rho A_F C_d (V_x + V_{wind})^2 = C_{ad} (V_x + V_{wind})^2 \quad (2)$$

where  $\rho$  is the mass air density,  $C_d$  is the aerodynamic drag coefficient,  $A_F$  is the frontal area of the vehicle, which is the projected area of the vehicle in the direction of travel,  $V_x$  is the longitudinal vehicle velocity,  $V_{wind}$  is the wind velocity, in this study wind velocity is considered negligible in comparison to vehicle velocity and will be - omitted when considering  $F_{ad}$ .

Wong developed the following relation between frontal area and car mass [10],

$$A_F = 1.6 + 0.00056(m - 765) \tag{3}$$

where,  $800 \text{ kg} \leq m \leq 2000 \text{ kg}$ , and the drag coefficient  $C_d$  for automobiles ranges between 0.3 and 0.4. The rolling resistance can be obtained from [11],

$$R_x = C_r mg \cos \theta. \tag{4}$$

Where  $C_r$  is the rolling resistance coefficient.

$F_x$  is a function of the slip ratio and the friction coefficient between tire and road [6]. But in this paper, the slip between tires and road is assumed to be zero, i.e. wheel velocity will be considered equal to vehicle velocity; and the longitudinal tire force expression will be extracted from the wheel dynamics.

### 2.2 Driveline Dynamics

The driveline is composed of engine, transmission, wheel, torque converter and final drive [6]. The engine dynamics equation is,

$$I_e \dot{\omega}_e = T_e - T_p \tag{5}$$

where,  $T_e$  is the net engine torque,  $T_p$  is the pump torque,  $I_e$  is the engine’s moment of inertia, and  $\omega_e$  is the engine angular velocity. In the simulation the net engine torque is obtained through a look up table to form the engine map that determines this torque as a function of throttle opening percentage and engine rotational speed.

The torque converter is considered to be locked, and that there is ideal fluid coupling, which yields the following equation [6],

$$T_t = T_p = -6.1 \times 10^{-6} \omega_e^2 \tag{6}$$

where,  $T_t$  is the turbine torque. And the transmission dynamics equation is [6],

$$I_t \dot{\omega}_t = T_t - RT_w \tag{7}$$

where  $T_w$  is the wheel torque,  $I_t$  is the transmission’s moment of inertia,  $\omega_t$  is the transmission angular velocity, and  $R$  the total gear speed transmission ratio.

As for the wheel dynamics equation it is given by [6],

$$I_w \dot{\omega}_w = T_w - T_{br} - r_{eff} F_x \tag{8}$$

where  $T_{br}$  is the brake torque,  $I_w$  is the wheel’s moment of inertia,  $\omega_w$  is the wheel angular velocity, and  $r_{eff}$  is the wheel’s effective radius. Given the previous assumptions that there is zero slip and the transmission is locked; i.e.  $T_t = T_p$ ,  $\omega_w = R\omega_e$  and  $\omega_t = \omega_e$  leads to the following expression of  $F_x$ ,

$$F_x = \frac{-6.1 \times 10^{-6} \omega_e^2 - \dot{\omega}_e [I_t + I_w R^2] - T_{br} R}{r_{eff} R} \tag{9}$$

And the longitudinal vehicle dynamic Eq. (1) becomes;

$$\ddot{x} = \frac{1}{m} \left[ \frac{\dot{\omega}_e [I_t + I_w R^2] + 6.1 \times 10^{-6} \omega_e^2 - T_{br} R_T}{r_{eff} R_T} - C_{ad} V_x^2 - mg(C_r \cos \theta - \sin \theta) \right] \tag{10}$$

### 3 ACC Controllers

The ACC itself is composed of the Lower Level Controller (LLC) and Upper Level Controller (ULC) [5]. The ULC role is to decide on whether the vehicle should be in cruise control mode or in vehicle following mode, and then compute the desired acceleration,  $a_d$ . The desired acceleration is then transferred to the LLC that uses host vehicle dynamics to determine the throttle position or brake input torque,  $T_{br}$  [6].

#### 3.1 System Limitations & Switching Logic

The ACC is restricted by the following physical constraints:

1. An ACC vehicle cannot have a negative velocity during a transitional maneuver [1].
2. For vehicle drag force limitation [6], driving comfort issues [4–6], and minimum jerking purposes [5], an ACC vehicle’s acceleration is limited between  $-4.9 \text{ m/s}^2$  and  $2.45 \text{ m/s}^2$  [12].
3. The inter-vehicular distance between host and leading vehicles must always remain bigger than a minimum safety distance  $d_{min}$  that is needed to avoid a collision if the vehicles applied brakes to the fullest at the same time, and smaller than a distance that can allow another car to cut-in between the two vehicles [6].

The ACC needs to follow a switching strategy that determines whether the ACC should be in regular CC mode or if it should be trailing a vehicle and enter the ACC mode. The following approach is based on the logical operation algorithm suggested in [13], which considers that an ACC vehicle does not need to switch modes, unless the leading vehicle has a lower velocity than that of the host vehicle, and is at distance less or equal to the change distance. Table 1 identifies the ACC’s mode of operation, function of the set vehicle speed  $v_{set}$ , leading vehicle speed  $v_l$ , inter-vehicular distance  $d$ , and the change distance  $d_c$ .

For a constant headway spacing policy,  $d_c$  has the following relationship [13];

$$d_c = l + d_0 + t_h v_x \tag{11}$$

**Table 1.** Switching logic between CC and ACC

	$v_l < v_{set}$	$v_l \geq v_{set}$
$d < d_c$	ACC	CC
$d \geq d_c$	CC	CC

Where,  $l$  is the vehicle’s length,  $d_0$  is an additional safety distance needed to keep the vehicles from colliding,  $t_h$  is the time headway constant and  $v_x$  is the vehicle’s longitudinal speed.

### 3.2 Upper Level Controller

The ULC is responsible for calculating the acceleration at which the vehicle must travel in order to attain the desired speeds and spacing or headway from other vehicles. This study will be based on the simple ACC model, where the control input is a first-order lag [6]. The model for this controller is,

$$\ddot{x}_{des} = \frac{1}{\tau s + 1} u \tag{12}$$

where  $\ddot{x}_{des}$  is the desired acceleration in order to meet the system’s requirements,  $\tau$  is the time lag, and will be considered equal to 0.5 s in this study as analytical and experimental studies show that this is its corresponding value, and  $u$  is the system’s control input.

**Cruise Mode PI Controller.** A typical controller for the cruise control mode is the Proportional-Integral controller [6],

$$u(t) = -k_p(v_x - v_s) - k_I \int_0^t (v_x - v_s) dt \tag{13}$$

where  $v_s$  is the driver’s set vehicle speed,  $k_p = 0.473$  is the proportional gain, and  $k_I = 6.782$  is the integrator’s proportional gain, these values were computed using Matlab optimal design.

### 3.3 Lower Level Controller

The LLC is responsible for interpreting the calculated acceleration value in the upper controller to yield the throttle position and the torque brake inputs. Here, all assumptions that took part of vehicle modeling will be adopted, and the longitudinal vehicle speed it is approximated to be [6],

$$\dot{x} \cong r_{eff} \omega_w. \tag{14}$$

Hence, the longitudinal acceleration can be expressed as,

$$\ddot{x}_{des} = r_{eff} R \dot{\omega}_e. \tag{15}$$

Under these assumptions the dynamics relating  $\omega_e$ ,  $T_e$ ,  $T_{br}$  can be linked with the expression [6],

$$\dot{\omega}_e = \frac{T_e - C_{ad} R^3 r_{eff}^3 \omega_e^2 - R(r_{eff} R_x + r_{eff} mg \sin \theta + T_{br})}{J_e} \tag{16}$$

where,  $J_e = I_e + I_t + R^2 I_w + m R^2 r_{eff}^2$ .

Since using brake and throttle pedals at the same time will not be allowed,  $T_{br}$  will be zero during throttle control and  $T_e$  can be computed using the following relationship,

$$T_e = \frac{J_e}{R r_{eff}} \ddot{x}_{des} + \left( C_{ad} R^3 r_{eff}^3 \omega_e^2 + R(r_{eff} R_x + r_{eff} mg \sin \theta) \right). \tag{17}$$

Then using an inverse engine map, having obtained  $T_e$ , and deriving  $\omega_e$  from the expression of  $\ddot{x}_{des}$ , the percentage of throttle input is obtained [14].

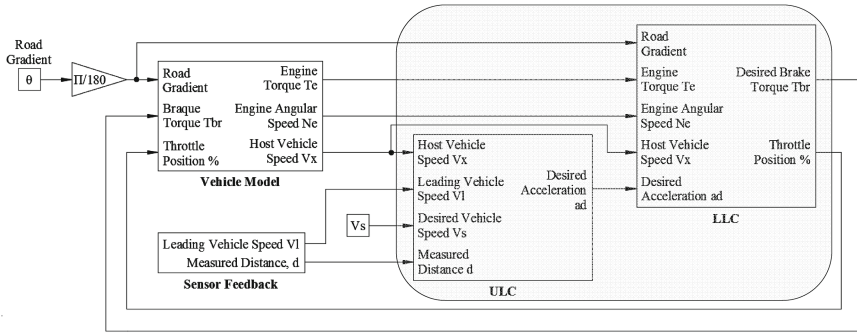
If the ULC decides that braking is needed then the LLC will use the actual engine torque, which is obtained from the vehicle itself, will be used, and  $T_{br}$  will be obtained using the following expression,

$$T_{br} = \frac{T_{e_{act}} - \frac{J_e}{R r_{eff}} \ddot{x}_{des} + \left( C_a R^3 r_{eff}^3 \omega_e^2 + R(r_{eff} R_x + r_{eff} mg \sin \theta) \right)}{R} \tag{18}$$

Connecting both the ULC and LLC will give the overall ACC controller. The inputs to the ULC are the host vehicle speed,  $v_x$ , the leading vehicle speed,  $v_l$ , desired vehicle speed or the set speed,  $v_s$ , and the measured distance,  $d$ . the ULC computes the desired acceleration,  $a_d$ , for the LLC. The LLC also takes in from the vehicle model the host vehicle speed,  $v_x$ , the engine angular speed,  $N_e$ , and the engine torque,  $T_e$ . The LLC can also consider the Road Gradient in determining the throttle position in percentage and the brake torque,  $T_{br}$ , which will be returned to the vehicle, and the overall model of interaction between ACC and vehicle is shown in Fig. 1. for the purpose of simplifying the simulation, the range sensor is considered to give only the leading vehicle’s speed, even though its real function is to give the relative velocity, but since the leading vehicle’s velocity is equal to the relative velocity added to the host vehicle velocity, this simplification can be made. Also the relative position can be obtained through the integration of the difference between the two vehicles velocities.

### 4 Event Driven ACC

A Discrete Event System (DES) is a dynamic system that deals with specific events that may occur abruptly at an unknown instant. In order to relate DES to ACC, one can consider possible events that can occur while driving [15]. The ACC must deal with a

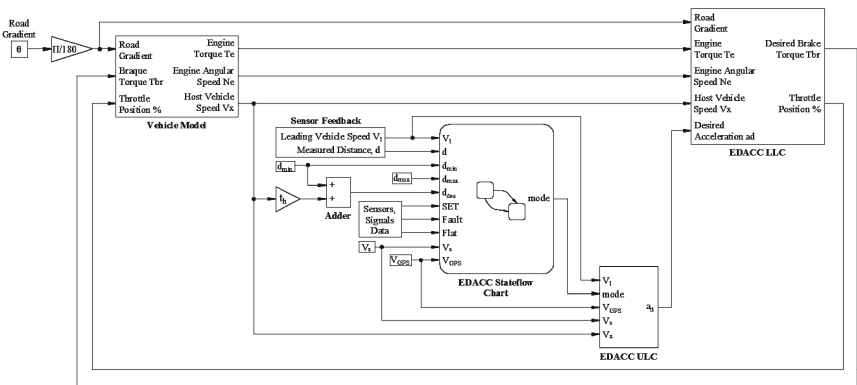


**Fig. 1.** Host vehicle with Adaptive Cruise Control that includes both the upper and lower level controllers

car that cuts in, or a leading car that slows down abruptly, a road block, a car that fully stopped at a red light, etc. A StateFlow Chart (SFC) can represent graphically a Finite State Machine (FSM) which simplifies and facilitates the application of DESs and the inclusion of special events [14]. The use of a SFC to represent transitions between modes facilitates the modeling of an Event-Driven System (EDS). In another words, EDS gives the ACC the ability to adapt and include new conditions to improve its capability of assisting the driver in case of emergencies and abrupt incidents that could lead to an accident, as well as being the milestone that leads to complete vehicle autonomy.

The use of a stateflow chart to represent transitions between modes facilitates the modeling of event-driven systems. In other words, the event-driven system gives the ACC system the ability to adapt and include new conditions to improve its ability to assist the driver in case of emergencies and abrupt incidents that could lead to an accident and as well as being the milestone of the road that leads to complete vehicle autonomy.

The ULC relies on the stateflow chart, Fig. 2, to determine the appropriate mode of operation [15].



**Fig. 2.** The Event-Driven ACC

In order to give the driver full control over the activity of the ACC, the first two evident states of an ACC are whether it is ON or OFF, these states are conveyed to the controller through the SET signal, coming from the ACC push button present on the car's dashboard. After activation, a conventional ACC, goes through four different states [15]:

1. When there are no close vehicles travelling at a speed lower than the driver's set speed  $v_s$ , the ACC's goal is to get the host vehicle to travel at this exact set speed, this state shall be known as: Cruise.
2. When a preceding vehicle is detected, the controller's first goal is to reach the desired inter-vehicle pre-set distance  $d_{des}$ , it must start slowing down the car after it reaches the maximum following headway distance  $d_{max}$ , and then after reaching the desired distance it will move on to the next state, this state shall be known as: Gap\_Reduction.
3. After the inter-vehicular distance is achieved, the controller's next goal is to achieve the leading vehicle's speed  $v_l$  in order to maintain this distance, this state shall be known as: Adaptive\_Speed.
4. If the controller was not able to slow down the vehicle fast enough to keep the vehicle within the desired headway, then a braking operation must occur before the car reaches an unsafe distance  $d_{min}$ , i.e. a distance smaller than that needed to avoid a collision if the two vehicles were to apply maximum braking at the same time, this state shall be known as: Brake.

In order to simplify and lessen the states, the ACC controller can be used to merge the Gap\_Reduction state and the Adaptive\_Speed state, into one state that achieves both their goals, this state shall be known as: Trail.

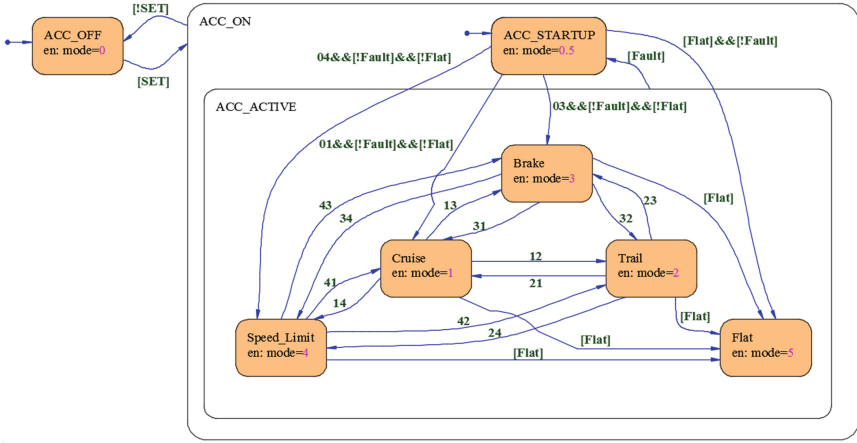
In this paper, an EDACC is introduced, and presents two new events that have been added to the conventional functions of an ACC. The first is the ability to detect a puncture or leak in a tire, Flat\_Tire, through a pressure sensor, and the second uses pre-entered data that uses GPS technology to determine speed limit zones, Speed\_Limit, and transmits it to the ACC in order to calculate the needed throttle or brake actuators position for the vehicle to travel at the speed limit  $v_{GPS}$ , while also notifying the driver that he has entered a speed limit zone with the help of GPS technology [2]. A flat tire requires a special treatment, as a first step the throttle input must be maintained momentarily, and then slowly and gradually must be decreased, and hard braking is disallowed before the car comes to a safe stop [16]. As for the speed limit  $v_{GPS}$ , it can be treated the same as the driver's set speed. The details of the stateflow chart shown in Fig. 2 are exemplified in Fig. 3.

Table 2 shows explicitly how the system transitions from one state to the other.

## 5 Simulation

To simulate the adaptability of the proposed ACC, the desired host vehicle speed,  $v_s$ , is set to 60 km/h. Figure 4 shows the host vehicle speed,  $v_x$  (with CC dashed line and with ACC solid line) against the assumed leading vehicle speed,  $v_l$  (dotted line) with a headway of 8 m. At zero simulation time the host vehicle is traveling at 36 km/h where

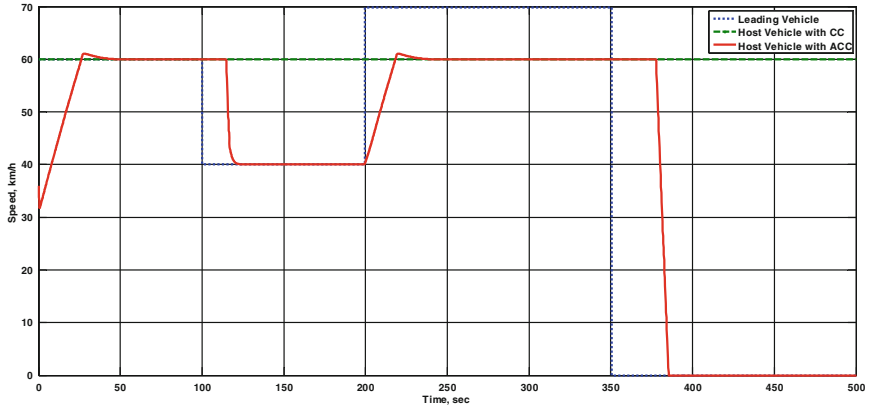




**Fig. 3.** Stateflow chart of the introduced Event-Driven ACC

**Table 2.** EDACC state transitions

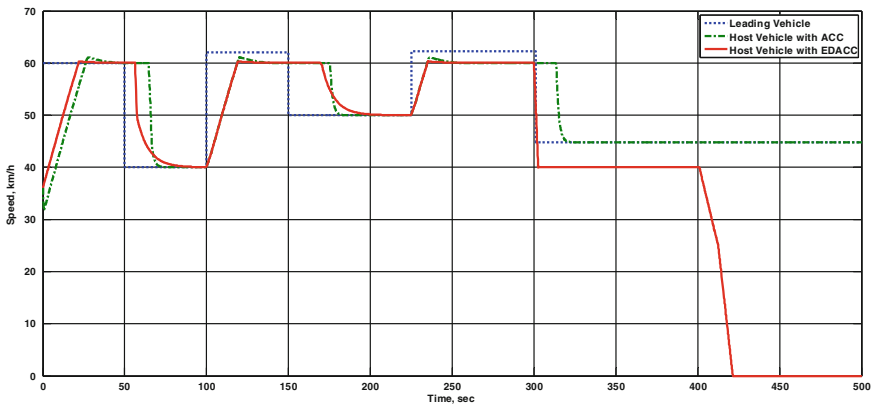
From	To	Condition	State
OFF	ON	SET = 1	[SET]
ON	OFF	SET = 0	[!SET]
Active	Standby	Fault = 1	[Fault]
Standby	Cruise	$v_{GPS} \geq v_s$ and $d \geq d_{max}$	01&&[Flat]&&[!Fault]
Standby	Brake	$d \leq d_{min}$	03&&[Flat]&&[!Fault]
Standby	Speed_Limit	$v_{GPS} < v_s$ and $d \geq d_{max}$	04&&[Flat]&&[!Fault]
Standby	Flat	Flat = 1 and Fault = 0	[Flat]&&[!Fault]
Cruise	Trail	$d < d_{max}$ and $v_1 < v_s$ and $d > d_{des}$	12
Cruise	Brake	$d < d_{min}$	13
Cruise	Speed_Limit	$(v_{GPS} < v_s)$ and $(d \geq d_{max})$ or $(v_s \leq v_1$ and $d > d_{min})$	14
Cruise	Flat	Flat = 1	[Flat]
Trail	Cruise	$(d \geq d_{max})$ or $(v_s \leq v_1$ and $d > d_{min})$	21
Trail	Brake	$d < d_{min}$	23
Trail	Speed_Limit	$(d \geq d_{max})$ or $(v_{GPS} \leq v_1$ and $d > d_{min})$	24
Trail	Flat	Flat = 1	[Flat]
Brake	Cruise	$(d \geq d_{max})$ or $(v_s \leq v_1$ and $d > d_{min})$	31
Brake	Trail	$d_{des} < d$ and $v_1 < v_s$ and $d > d_{min}$	32
Brake	Speed_Limit	$(d \geq d_{max})$ or $(v_{GPS} \leq v_1$ and $d > d_{min})$	34
Brake	Flat	Flat = 1	[Flat]
Speed_Limit	Cruise	$(v_{GPS} \geq v_s)$ and $(d \geq d_{max})$ or $(v_s \leq v_1$ and $d > d_{min})$	41
Speed_Limit	Trail	$d < d_{max}$ and $v_1 < v_s$ and $d > d_{des}$	42
Speed_Limit	Brake	$d < d_{min}$	43
Speed_Limit	Flat	Flat = 1	[Flat]



**Fig. 4.** Comparative performance between two host vehicles one with CC and another with ACC with desired speed was set to 60 km/h

the ACC system was activated at this instant and the driver sets the speed to 60 km/h. Note that when the leading vehicle speed,  $v_l = 40$  km/h at  $100 < t < 200$ , the host vehicle follows the leading vehicle. When the leading vehicle speed,  $v_l = 70$  km/h at  $200 < t < 350$ , the host vehicle speed remains at the set speed of 60 km/h. And eventually, when the leading vehicle stops, the host vehicle simply follows and stops for  $t \geq 350$ . Note that a host vehicle with a regular CC, its speed is equal to the set speed of 60 km/h at all times regardless what the leading vehicle speed is.

The EDACC model was tested through the following scenario, Fig. 5, with a headway of 8 m between the host and leading vehicles: at zero simulation time the host vehicle is traveling at 36 km/h where the EDACC system was activated at this instant and the driver sets the speed to 60 km/h.



**Fig. 5.** Comparative performance between two host vehicles one with ACC and another with EDACC with desired speed was set to 60 km/h

A leading vehicle is moving at a speed,  $v_l = 60$  km/h at  $t \geq 0$ . At  $50 < t < 100$ , the leading vehicle slows down to 40 km/h. Consequently, the host vehicle follows and its speed drops to 40 km/h. When the leading vehicle speed,  $v_l = 65$  km/h at  $100 < t < 150$ , the host vehicle speed remains at the set speed of 60 km/h. Then at  $t = 150$ , a new vehicle traveling at 50 km/h cuts in and then changes lane at  $t = 225$ . The host vehicle adjusts its speed to 50 km/h until the intruder vehicle changes lane at  $t = 225$  and then picks up its speed to the set speed of 60 km/h since the leading vehicle speed is above 60 km/h between  $225 < t < 300$ . Until now the performance of two host vehicles one with ACC and the other with EDACC is the same. However, at  $t = 300$ , the EDACC slows the host vehicle down to 40 km/h because it enters a 40 km/h speed limit zone whereas the host vehicle with ACC kept following the leading vehicle and it did not honor the speed limit. Finally, At  $t = 400$ , the host vehicle with EDACC stops because of a warning message indicating that one of the tires' air pressure is low while the host vehicle with a regular ACC kept going and did not stop for this emergency.

## 6 Conclusion

In this paper, a simplified longitudinal vehicle model was derived. The Adaptive Cruise Controller (ACC) composed of the upper and lower level controllers was also designed and simulated. Also, the importance of Discrete Event System (DES) and how state-flow charts can be used in order to facilitate the design of an Event-Driven Adaptive Cruise Controller (EDACC) was introduced. Two new modes of operation, namely speed-limit zone and flat tire occurrence, have been added to the conventional ACC model to yield the EDACC. Future work can be done on improving the system response in going from one state to another. This work also suggests that well designed EDACC can ultimately lead to an autonomous vehicle.

## References

1. Ali, Z.: Transitional controller design for adaptive cruise control systems. Ph.D. thesis, University of Nottingham, UK (2011)
2. Labbani, O., Rutten, É., Dekeyser, J.L.: Safe design methodology for an Intelligent cruise control system with GPS. In: 64th IEEE Vehicular Technology Conference, Montréal, Québec, Canada, September 2006 (2006)
3. Hiraoka, T., Kunimatsu, T., Nishihara, O., Kumamoto, H.: Modeling of driver following behavior based on minimum-jerk theory. In: 12th World Congress on ITS, San Francisco, USA, November 2005, vol. 113, no. 4, pp. 27–30 (2005)
4. Martinez, J.J., Canudas-de-Wit, C.: A safe longitudinal control for adaptive cruise control and stop-and-go scenarios. *IEEE Trans. Control Syst. Technol.* **15**(2), 246–258 (2007)
5. Luo, L., Liu, H., Li, P., Wang, H.: Model predictive control for adaptive cruise control with multi-objectives: comfort, fuel-economy, safety and car-following. *J. Zhejiang Univ. Sci. A* **I**(3), 191–201 (2010)
6. Rajamani, R.: *Vehicle Dynamics and Control*. Mechanical Engineering Series. Springer, New York (2006)

7. Hedrick, J.K., Gerdes, J.C., Maciucă, D.B., Swaroop, D.: Brake system modeling, control and integrated brake/throttle switching phase I. California PATH Research report, University of California, Berkeley, California (1997)
8. Wang, J., Rajamani, R.: Should adaptive cruise control systems be designed to maintain a constant time gap between vehicles? *IEEE Trans. Veh. Technol.* **53**(5), 1480–1490 (2004)
9. Wang, J., Rajamani, R.: Adaptive cruise control system design and its impact on traffic flow. In: *Proceedings of the American Control Conference*, Anchorage, Alaska, May 2002 (2002)
10. Wong, J.Y.: *Theory of Ground Vehicles*, 4th edn. Wiley, New York (2008)
11. Shakouri, P., Ordys, A., Askari, M., Laila, D.S.: Longitudinal vehicle dynamics using Simulink/Matlab. In: *UKACC International conference on CONTROL*, Coventry, 6–10 September 2010 (2010)
12. Kulkarni, M., Shim, T., Zhang, Y.: Shift dynamics and control of dual-clutch transmissions. *Mech. Mach. Theor.* **42**(2), 168–182 (2007)
13. Shakouri, P., Czczot, J., Ordys, A.: Adaptive cruise control system using balance-based adaptive control techniques. In: *17th International Conference on Methods and Models in Automation and Robotics (MMAR)*, Miedzyzdroje, Poland, August 2012, pp. 510–515 (2012)
14. Güvenç, B.A., Kural, E.: Adaptive cruise control simulator: a low-cost, multiple-driver-in-the-loop simulator. *IEEE Control Syst. Mag.* **26**(3), 42–55 (2006)
15. Breimer, B.: Design of adaptive cruise control model for hybrid systems fault diagnosis. MSc thesis, Department of Computing and Software, McMaster University (2013)
16. How to control your car during a Puncture or Tyre Blowout, June 2014. [roaddriver.co.uk](http://roaddriver.co.uk), online driving community. Article No. 37 RoadDriver 2010