A real-time hardware-in-the-loop vehicle simulator for traction assist

S. Alles*, C. Swick*, M. Hoffman*, S.M. Mahmud †, and Feng Lin †

* Ford Motor Co. Electrical/Electronics Systems Engineering, 20000 Rotunda Dr. Bldg 5, Rm#1020, Dearborn, MI 48121, USA.

† Electrical & Computer Engineering Dept., Wayne State University, Detroit, MI 48202, USA.

Abstract: A computer simulation technique which simulates a vehicle in real-time is presented. The vehicle has been divided into a number of subsystems, and each subsystem is modelled separately. Along with the models, the hardware components of several control units were linked together to provide a generic real-time Hardware-In-The-Loop (HITL) vehicle simulation. Such a model provides the capability to verify analytical and experimental data, and adjust the hardware of certain vehicle components on the test bench. It also provides powerful, complex and dynamic real-time simulations while being portable (unlike large mainframe systems developed for DYNMOD (DeLosh et al.) in non-real-time), reduces on-board vehicle testing time, makes it adaptable to include different types of vehicles, and is cost-effective. This paper outlines the major features of the simulation models, showing a typical application for Traction Assist (TA) development. The simulation model was verified for a TA event by taking measurements from an on-board computer while the vehicle was driven on various types of road surfaces. The results obtained experimentally corresponded well with those obtained through simulation. Due to length limitations, only the results for the dry asphalt and packed snow type surfaces are shown.

Key words: computer simulation, hardware-in-the-loop (HITL), modelling, road surfaces, traction assist (TA), Windows.


1 Introduction

Traction Assist (TA) is a means of reducing slip and achieving better stability, controllability, driveability and optimum traction of the vehicle during acceleration. Its importance is greatly felt when accelerating on a slippery surface like snow or ice and during cornering manoeuvres. If during a slip the driver presses the brake, the TA is disabled and will be resumed only when the brake pedal is released. The Anti-lock/Antiskid Brake System (ABS) is different in that it only prevents wheel lock-up during heavy braking optimizing braking effectiveness while maintaining stability and
steerability. An analysis of the latest and future traction assist trends are discussed in Shiraishi et al., 1992.

Traditionally, most of the mechanical designs in automobiles were done by engineers using the trial and error method. But now due to more complex electronics, stringent safety requirements, continued cost pressures and competitiveness, testing of the total vehicle system must be performed for extreme conditions. Simulation offers this facility with a wide variety of choices incurring a lower fixed cost and shortening the development time immensely. Off-line simulation methods are necessary to verify whether development will be successful before too much cost is incurred. Once this has been established, the real-time simulation is important during the design phase to achieve closer correlation with the actual vehicle and improve the ability to evaluate and resolve complex failure modes. In TA developments, a realistic simulation of the dynamic behaviour of the total vehicle system is needed for the following reasons:

- Control law strategy development.
- Vehicle performance under different driving conditions.
- Suitable testing conditions for various road surfaces (i.e. ice, packed snow, etc.) are not readily available (the correct weather and a dedicated track facility are often required).
- Repeatable conditions (there is no such thing as polished ice, the coefficient of friction changes drastically with temperature).
- Rapid and safe testing scenarios that are potentially hazardous with first pass control strategies, even at the best available facilities.

This simulation provides repeatable runs that can be compared to actual vehicle data. This facilitated fine tuning of the hardware and software before the actual winter testing was performed. Included also was the need to develop the simulation in a generic form thus providing the capability to change vehicle dynamics, driveline or other hardware with minimal down time. The software was designed using an object-oriented (OO) approach so that the modularity of the system could be maintained and also a more user-friendly Windows environment could be used. The OO approach enhances program comprehension for easier development and maintenance during its entire life-cycle, and its role in a distributed computing environment (Yan et al., 1992). As discussed in Section 2, the simulator will need to be regularly updated to encompass more complex models for future development purposes. Therefore, the OO design approach seems to be the most natural approach for a project that needs constant enhancements, as the new design should be able to use most of the previous design and code. Detailed information on the use of OO design in this application can be found in Alles et al., 1993. The type of vehicles modelled are a 4.6-litre, 8-cylinder (VEH4.6) and a 3.8-litre, 6-cylinder (VEH3.8) sequential electronic fuel-injected (SEFI) engine with an automatic transmission. The different strategies used for engine intervention in TA control are explained in greater detail in Section 4. The vehicle simulator was designed for a distributed processing environment, although at present it runs on an IBM PS/2 with a single processor.

Section 2 provides a system description of the models of different components of a vehicle. Section 3 shows the implementation of the entire simulation model. The simulation results are presented in Section 4 together with a comparison between the
simulated data and that obtained from winter testing. Section 5 discusses the validation procedures. Section 6 shows the future direction of the work; and finally the conclusion is given in Section 7.

2 System description

The completion vehicle was divided into the following subsystems, and each subsystem was modelled separately in software:

- Engine model
- Torque converter model
- Driveline model
- Brake model
- Tyre/Road surface model (separate for left and right wheels)
- Vehicle model
- Driver (user interface) model

An IBM PS/2-486 machine was used to run the software models and communicate with other hardware components in the simulation loop. The initial models were developed using the Matrix X software package (June, 1991) as it provided a convenient method of formulating each of the models before it was C-coded for use on an IBM PS/2. At present, due to the real-time constraints, some of the complex subsystems (e.g. Anti-lock Brake Controller (ABS), Powertrain Control Module (PCM)) are connected directly in the simulation loop rather than converting them into software models. Thus, our simulation technique becomes a hardware-in-the-loop simulation. However, for off-line simulation the above mentioned complex subsystems can be modelled and incorporated in software.

When mathematically describing the behaviour of a model, the following should be maintained (Bakker and Pacejka):

- It should be able to describe all steady-state characteristics.
- It should be easily obtainable from measured data.
- Its parameters should characterize the item being modelled.
- It should be compact and sufficiently accurate.

Look-up tables could be used as alternatives to modelling complex items, but because of the limited processing power, the data have been divided into different ranges with variable format regressed equations, so that a better fit could be obtained. Because of the problem of discontinuity at the boundaries, the most convenient method was the use of high order polynomials or special functions (Bakker and Pacejka), e.g. exponential, trigonometric, etc., so as to cover the entire range, but this is not always possible. In cases where an accurate representation is required or the component is too complex to be modelled, regression techniques are used. Especially in high order regressions, the database used should be large enough so that extrapolation is not required, otherwise
there could be large deviations between the physical system and the model. The objective of the models is to represent a simplified concept of the essential features of the physical system necessary from a traction control standpoint. The following subsections present brief descriptions of the models used for the HITL development.

2.1 Engine model

Since the model was developed for a port fuel injected SEFI engine, wall wetting can be neglected (Crossley and Cook, 1991). Unlike carburetted engines, the injectors are in close proximity to the cylinder inlet port and the fuel condensation can be considered insignificant. An important phase of a vehicle is starting on a cold engine, but since TA applies to a vehicle in motion, a warm engine with no initial transients can be assumed for simulation. However, the transients during a throttle tip-in have been modelled with a sufficiently good manifold-filling model. Figure 1 (overleaf) shows the schematic of the engine model with all required inputs/outputs as seen by the PCM. The equivalent throttle angle is a combination of the primary throttle (driver controlled) and a secondary throttle (PCM controlled). The typical characteristics of an induction process are given in Powell and Cook, 1989, who showed how the mass air flow (MAF) rate varied with manifold pressure (MAP) for different engine speeds. When the manifold pressure is less than about half the atmospheric pressure, a choked or sonic condition exists and the mass air flow rate is a function of the equivalent throttle angle only. When the manifold pressure is greater than half the atmospheric pressure, the mass air flow rate is given by a root pressure relation. The mass air flow rate, $m_a$, can be approximately by the following equation (Crossley and Cook, 1991; Powell and Cook, 1989):

$$m_a = F(\theta)_m g(P_m, P_{atm})$$

(1)

where

$$g(P_m, P_{atm}) = \begin{cases} 1 & \text{for } \frac{P_m}{P_{atm}} \leq 0.5 \\ \frac{2}{P_{atm}} \cdot \sqrt{P_m P_{atm} - P_m^2} & \text{otherwise} \end{cases}$$

$P_m$ and $P_{atm}$ are the manifold and atmospheric pressures, respectively, and $F(\theta)_m$ is a function of the discharge coefficient and geometric area of the throttle bore and throttle angle, which can be approximated by a high order polynomial. However, this can become extremely complex, especially with the two-throttle system. Therefore, it was more convenient to use a look-up table or develop variable format regressed equations using actual air flow data for various ranges of throttle angle. The properties of the air flow in the manifold are assumed uniform along with a constant manifold temperature; this greatly simplifies the manifold modelling. The changes in temperature generally influence the condensation and vaporization transients which is important in carburetted engines. The dilution of the air with exhaust gas recirculation (EGR) reduces the overheating, thereby reducing the formation of various nitrogen oxides, increasing the combustion efficiency. The EGR system on the modelled engine consists of an orifice in the exhaust manifold connected back to the intake manifold through a vacuum activated EGR valve. The experimental setup is given in Crossley and Cook, 1991. The EGR flow rate is modelled using the orifice-flow equation similar to that in Equation 1. The gas was assumed to behave ideally, with no manifold leaks.
By the conservation of mass flow into and out of the manifold, the pressure change
within the manifold becomes:

\[
\frac{dP_m}{dt} = k_p \left( \frac{dm_a}{dt} + \frac{dm_{egr}}{dt} - \frac{dM}{dt} \right)
\]

(2)

where \(k_p\), the manifold filling constant, is a function of manifold temperature, pressure,
universal gas constant (R), gas molecular weight (m) and specific heat ratio (γ); M is the
mass air flow out of the manifold into the cylinders. The fuel injectors are turned on by
the PCM according to the engine speed. The PCM also controls the flow of fuel through
each injector by opening the injector for a certain period of time, and by sending a pulse
to the injector to keep it open. The width of this pulse determines the duration the injector
should remain open. Since the injector valve is a mechanical device, there is a certain
time lag between the beginning of the pulse sent by the PCM and the actual opening of
the injector. Therefore, the rate of fuel flow from the injectors can be given by Aquino:

\[
\frac{dm_f}{dt} = A_0 \left( PW - A_1 \right) N_{inj} \cdot N_e
\]

(3)

where \(A_0\) is the fuel flow rate through an injector, \(A_1\) is the time lag between the start of
the pulse and the opening of the injector, PW is the width of the injector pulse, \(N_{inj}\) is the
number of injectors in the engine, and \(N_e\) is the engine speed in revolutions per minute
(rpm). The speed density approach (Aquino; Servati and DeLooh) was used to determine
the air flow rate at the inlet of cylinders. The mass of air, \(m_{cyl}\), inducted per engine cycle
at intake conditions is

\[
m_{cyl} = V_d \cdot \rho
\]

(4)

where \(V_d\) is the engine displacement and \(\rho\) is the density of air. If we assume that the air
in the manifold behaves as a perfect gas, then the density of air can be determined from
the manifold pressure and temperature as follows:

\[
\rho = \frac{P_m}{R T_m}
\]

(5)

where \(P_m\) and \(T_m\) are manifold pressure and temperature, respectively, and \(R\) is the
universal gas constant. Each engine cycle of a 4-stroke engine consists of intake,
compression, power and exhaust strokes. This means that there are two revolutions per
engine cycle. Thus, the mass air flow rate at the inlet of the cylinders at intake conditions
is

\[
m_{in} = m_{cyl} \cdot \frac{N_e}{2}
\]

(6)

But the actual air inducted by the cylinders depends on the volumetric efficiency of the
conducting engine. The volumetric efficiency, \(\eta_v\), is the ratio of the actual mass of air
inducted by the cylinders to the mass of air inducted at intake conditions. Thus, the actual
mass air flow rate into the cylinders is given by
Figure 1  Schematic of engine.
\[
\frac{dM}{dt} = m_{in} \cdot \eta_v
\]
\[
= \frac{N_e}{2} \cdot \frac{V_d \cdot P_m}{RT_m} \cdot \eta_v
\]

For constant intake temperature and exhaust gas pressure, the volumetric efficiency can also be regressed as a high order polynomial in speed and manifold pressure from actual steady state engine mapping data. Thus, the actual mass air flow rate into the cylinders can be obtained using regression (Equation 6). Hence, the actual mass air flow rate can be expressed as a function of \( N_e \) and \( P_m \) as follows:

\[
\frac{dM}{dt} = f(N_e, P_m)
\]

There is always an induction-to-power stroke lag because the engine torque developed at any time is a function of the mass flow rates sampled one induction event earlier. At a speed of \( N_e \) (RPM), the rate of change of crank angle is \( 6N_e \) deg/sec, i.e. \( 1/6N_e \) sec/deg. To turn one combustion cycle (720 degs) it will take \( \frac{120}{N_e} \) secs. For an n cylinder engine, the induction-to-power time lag can be expressed as

\[
\tau = \frac{120}{nN_e}
\]

The torque generated from the combustion process depends on many variables, but the five fundamental variables that provided a significant contribution to the control strategies are: air/fuel ratio, residual gas, EGR rate, engine speed and ignition timing (spark). Thus, the engine torque at time \( t \), \( T_e(t) \), can be expressed as

\[
T_e(t) = f\left(M(t - \tau), nf(t - \tau), spark(t - \tau), egr(t - \tau), N_e(t)\right)
\]

It is the structure of Equation 10 rather than its value that is important, i.e. the sensitivity of the torque variation is important since it determines the effect of the various engine parameters on the output torque. Therefore, the weighting of each variable is obtained by regression with the higher order terms that do not contribute significantly to the torque, neglected. By Newton's laws of motion, the engine speed, \( N_e \) (rpm) is obtained from Equation 11:

\[
J_e \frac{dN_e}{dt} = \frac{30}{\pi} (T_e - T_f)
\]

where \( J_e \) is the moment of inertia of the engine, \( T_e \) is the engine torque, and \( T_f \) represents the friction, altitude loss and torque converter loads.

2.2 Torque converter model

Since the simulation model is for automatic transmission vehicles, the clutch is replaced with a torque converter. The torque converter works on the principle of fluid coupling, providing a smooth transition during drive. It also provides a torque multiplication factor in addition to that provided by the normal mechanical gear transmission. In first gear, the torque converter provides the maximum torque multiplication which progressively
decreases with increase in vehicle speed. The performance of the torque converter is generally specified by a plot of the input capacity factor and torque ratio vs. speed ratio as shown in Figure 2, where

\[
\text{input capacity factor} = \frac{N_r}{\sqrt{T_r}} \quad (12a)
\]
speed ratio (SR) \[= \frac{\text{turbine speed } (N_{gbin})}{\text{impeller speed } (N_e)} \]  

(12b)

torque ratio (TR) \[= \frac{T_c}{T_e} \]  

(12c)

where \(N_{gbin}\) is the turbine speed, which is speed of the input shaft of the gear box, and \(T_e\) is torque output of the torque converter. Energy losses in the torque converter are mainly due to fluid circulation within the drive and expressed as a converter efficiency.

2.3 Driveline model

Assume that the power from the engine is transmitted to the tyres through the shafts and gears with no frictional loss. The automatic transmission is assumed to be in the first gear, which is sufficient for low speed TA. By incorporating the gear shift schedule, an all speed TA could be realized. The engine is assumed to be rigidly mounted in the chassis, as shown in Figure 3. For the left half-shaft (similar for right), applying the principles of mechanics we can write

\[
\omega_{gbout} = \frac{1}{r_{gb}} \cdot \omega_{gbin} \]  

(13a)

\[
T_{gbout} = r_{gb} \cdot T_c \]  

(13b)

\[
T_{lhs} = K_{lhs} \cdot \theta_{lhs} = T_{gbout} \cdot r_{fd} \]  

(13c)

Figure 3  Schematic of driveline.
\[ \omega_{\text{ths}} = \omega_{\text{gout}} \cdot \frac{1}{r_{\text{fl}}} \]  \hspace{1cm} (13d)

\[ \dot{\theta}_{\text{ths}} = \omega_{\text{ths}} - \omega_{\text{fl}} \]  \hspace{1cm} (13e)

where \( \omega_{\text{ehin}} \) and \( \omega_{\text{gout}} \) are the angular speed of the shafts at the input and output of the gearbox, respectively; \( r_{\text{gb}} \) and \( r_{\text{fd}} \) are the gear ratios of the gearbox and the drive, respectively; \( T_{\text{gout}} \) is the torque output of the gearbox; \( T_{\text{ths}} \) and \( \omega_{\text{ths}} \) respectively indicate the torque and angular speed of the left half-shaft; \( K_{\text{ths}} \) and \( \theta_{\text{ths}} \) respectively indicate the stiffness and deflection of the left half-shaft; and \( \omega_{\text{fl}} \) is the angular speed of the front left wheel. Now considering the lumped wheel and half-shaft, we can write

\[ T_{\text{ths}} - T_{\text{flb}} - T_{\text{ftr}} = (J_{\text{ths}} + J_{\text{fl}}) \frac{d\omega_{\text{fl}}}{dt} \]  \hspace{1cm} (14)

where \( T_{\text{ths}}, T_{\text{flb}} \) and \( T_{\text{ftr}} \) are the corresponding torques of the left half-shaft, front left brake and front left traction, respectively. \( J_{\text{ths}} \) and \( J_{\text{fl}} \) respectively indicate the moment of inertia of the left half-shaft and front left wheel.

2.4 Tyre model

It is important to have a realistic tyre model for this simulation as TA studies depend heavily on the road and tyre surfaces. In manoeuvring of the vehicle, the lateral and longitudinal motions of the vehicle are strongly coupled with the tyre forces, and large slips occur simultaneously. Figure 4 shows the dynamics of wheel rotation (e.g. left wheel). The tractive momentum \( (M_t) \) causes the rotation against the inertia and rolling resistance, resulting in a speed difference between the tyre and road. The difference between the wheel's circumferential speed and the vehicle speed is modelled as a continuous deformation, \( s \), called the wheel slip ratio. For example, the slip ratio of the front left wheel can be expressed as

\[ s = \frac{R_{fl} \cdot \omega_{fl} - V_{veh}}{V_{veh}} = \frac{R_{fl} \cdot \omega_{fl}}{V_{veh}} - 1 \]  \hspace{1cm} (15)

Figure 4  Schematic of tyre.
where $R_n$ is radius of the front left wheel and $V_{veh}$ is the vehicle speed. During braking condition $R_n \cdot \omega_n < V_{veh}$, as a result the slip becomes negative, and during acceleration the slip is positive. Influenced by the load, $F_z$, on the wheel, the tyres produce a circumferential force, $F_t$, which accelerates the vehicle. The values of $F_z$ and $F_t$ are related by the coefficient of friction as

$$
\mu = \frac{F_z}{F_t}
$$

A typical set of curves showing the coefficient of friction versus slip variation for different road surface is illustrated (Srinivasa et al., 1980). These curves show that the peak frictional coefficient occurs at about $s = 0.2$ regardless of the road surface, reducing steadily thereafter. When this frictional coefficient reaches the peak, the wheel either starts to lock on braking or spin on accelerating. In order to stop the vehicle efficiently with good handling, it is best to maintain $s$ around 0.2. This model is capable of handling both traction and braking. When a vehicle travels along a curve, the front and rear wheels move along arcs of different radii, as shown in Figure 5, because only the front wheels are steered. Considering the left wheels (similar for right wheels) for slip calculations,

$$\text{speed ratio} = \frac{\text{speed of inside left wheel}}{\text{speed of inside rear left wheel}} = \frac{R_{fl} \cdot \omega_{fl}}{R_{rl} \cdot \omega_{rl}}$$

where $R_{fl}$ and $\omega_{fl}$ respectively indicate the radius and angular speed of the rear left wheel. Using Equation 15, we can write

$$s = \frac{R_{fl} \cdot \omega_{fl}}{R_{rl} \cdot \omega_{rl}} - 1 = \frac{\sqrt{R_{fl}^2 + B^2}}{R_{rl}} \cdot \frac{\omega_{fl}}{\omega_{rl}} - 1$$

where $B$ is the wheelbase, as shown in Figure 5.

![Figure 5: Vehicle making a turn.](image-url)
Bakker and Pacekja provide a more complete model which includes the standard SAE manoeuvres (i.e. lane changes and driving in a curve). However, in our simulator, the simpler Dugoff model is implemented. The effects of the lateral load transfer during cornering, the lateral slip and yaw, and the braking distribution on the vehicle system dynamics are not taken into consideration in this simulation, but will be incorporated in the future with the Pacekja model.

2.5 Brake model

Figure 6 shows a typical brake system. In this implementation, there are two ON/OFF solenoids per brake caliper. The brake pressure is increased by opening the inlet valves and decreased by opening the outlet valves which sends the brake fluid back to the reservoir. These two valves also provide the ability to hold brake pressure. This simple model adequately simulates the main behaviour of a typical electro-hydraulic brake system as shown by Hrovat. The time delay of the pump is modelled as a first order system. The brake pressure is obtained from regressed data of the volume of the brake fluid passed to the caliper, including the activation delays. The caliper torque is a function of many variables, e.g. brake pressure, contact surface area of brake shoe, friction between shoe and rotor, and the temperature. The various leakages in the hydraulic path have been neglected.

![Diagram of brake system](image)

**Figure 6** Schematic of brake system.

2.6 Vehicle model

The vehicle model incorporates the drag forces due to the wind resistances and the forces against gravity when the vehicle moves up an incline. Applying Newton’s law of motion,
tractive force, \( F_i = \frac{T_{fl} + T_{fr}}{R_{tyre}} \) \hspace{1cm} (19)

where \( R_{tyre} \) is the rolling radius of the tyre and \( T_{fl}, T_{fr} \) are the front left, right tractive torques respectively. If \( M \) is the mass of the vehicle and \( F_{load} \) is the resistive load due to rolling and aerodynamic drag, then

vehicle speed, \( V_{veh} = \frac{1}{M} \int (F_i - F_{load}) dt \) \hspace{1cm} (20)

2.7 Driver model

This represents the user's input/output interface to the overall system. The inputs required by the simulator are the primary throttle positions, idle engine speed (rpm), road surface information like inclination and frictional coefficient, etc. The driver-controlled throttle could be varied during the run via a pedal, but the sequence of the variation in the road properties has to be initially preset. The outputs are the different vehicle parameters needed to verify the effectiveness of traction control, e.g. driven wheel speeds, vehicle speed, engine, shaft and brake torques, etc., which can be viewed graphically.

3 Implementation

The software, coded in C++, runs on an IBM PS/2 486 machine. For real-time simulation, the PCM, secondary throttle and ABS/TA module are interfaced to the PC via A/D and D/A boards. For TA simulation, the user inputs the type of road surface, slope of the road and the primary throttle angle, which can be changed continuously during the run. Figure 7 shows the overall system model of the simulator, and Figure 8 shows a more detailed diagram of the PC interfacing. The wheel speeds are sent to the ABS/TA load box in the form of pulses which vary in frequency as the corresponding wheel speed changes. This is done via the D/A board through a voltage/frequency converter. The load box conditions the signals being sent and received by the PC and also represents the loads of the solenoid valves as seen in the real vehicle; otherwise, the ABS/TA module will go into a fault mode of operation. Since the vehicle being simulated was a front wheel drive, a common signal is sent for both the rear wheels. The spark timer converts the pulse width signal of the spark advance or retard into binary for identification by the PC, and also provides a visual display of the spark angle. The injector cut monitor provides a visual display of the injectors being controlled by the PCM in the form of LEDs to represent each injector, where an LED ON indicates an injector cut-off. The engine parameters, e.g. engine speed, mass air flow, turbine speed, vehicle speed, are sent by the PC to the PCM. The PCM then determines the corresponding spark timing and amount of fuel required for the most efficient performance of the engine. The braking sequence used in the ABS is input to the PC via the hydraulic solenoid control signals, which are in the form of pulses. A function generator provides a constant frequency that simulates the hydraulic pump in the braking system. The ABS/TA module communicates with the PCM through a SAE J1850-compatible communication bus. When the ABS/TA module detects a wheel slip, it controls the brakes and informs the PCM through the SAE J1850 bus to adjust the engine torque accordingly; once a threshold is reached, the brakes activate to bring down slip.
quickly, the hard deadlines for the model execution are met through variable timed interrupts, with the brake model running the fastest.
Test Bench Hardware Interconnection

Figure 8  Test bench hardware interconnection.
The OO language C++ and Borland’s Application Framework Object Windows were used in developing the application in a Microsoft Windows environment. It has an object-oriented class library that encapsulates the behaviour that Windows applications commonly perform, reducing development time. The Windows in this application are derived from the basic Window frames provided in their class library (Alles et al., 1993). The resource workshop editor was used to create buttons, dialog boxes and pull-down menus. The buttons enable the user to select a particular operation, e.g. start simulation, plot, view files and documentation, etc. The dialog boxes allow convenient user input. The pull-down menus within each window facilitates the selection of different options in each category as depicted by its title. Figure 9 shows the snapshots of the screen at different points during the simulation. Figure 9a shows the initial simulation control panel. Button ‘Sim Models’ invokes the window shown in Figure 9b, which allows the user to pick objects from the pull-down menus under each category, such as a particular engine type, driveline tyres with different sizes and standard threads, etc., from the available objects, depending on the vehicle being simulated. Button ‘Build Sim’ then dynamically links these selected objects. Button ‘Activate Sim’ allows the user to either enter new initial conditions or retrieve previous settings through the dialog boxes, as shown in Figure 9c. Split and time-varying μ-surfaces (e.g. checkerboard, etc.) are also given as another option. Once the user is satisfied, he/she starts the simulation. On completion of the run, the user activates the button ‘Plot Data’, which brings up the screen shown in Figure 9d (on ice with wide-open-throttle), for plotting the selected parameters, selected from the pull-down menu ‘Y-axis’, either individually or overlaid with the same parameter from a previous file.
Figure 9b

Figure 9c

Figure 9d
4 Traction assist simulation

A typical application for this hardware-in-the-loop simulation was for traction control. To obtain stability, steerability and maximum traction the effective torque on the wheel should be maintained at about the peak friction level. This can be best achieved by:

- Application of brake torque to control spin.
- Effectively reducing engine torque.
- Suitable combination of the above.

The braking solution is most effective for reducing speed in a slip condition as it acts on the wheels directly, but repeated use of the brakes will cause excessive wear and enormous heat, which could deteriorate other brake components. This is similar to driving with the handbrake on. The engine torque can be reduced by the following methods or combinations of them:

- Retarding spark.
- Cutting off injectors, thereby reducing fuel.
- Throttle control reducing air flow into cylinders.

Spark retardation and fuel control provide a quick response for torque reduction but they create problems for emission control and can degrade the catalytic converter. Throttle control has no such disadvantages, but its response is relatively slow. A gear upshift strategy also helps the transmission and lowers the drive torque as required. Engine control can effectively reduce the torque driving the wheels but when the individual wheels spin at different speeds a brake intervention is needed for individual wheel spin control. In reality, the split $\mu$-surfaces are the more common case, e.g. a vehicle accelerating with the left wheel may be on asphalt and the right wheel may be on snow. Therefore, a suitable combination of engine torque reduction and brake control is the most effective strategy for traction control. In VEH4.6, the engine intervention strategies used are spark and series throttle, whereas in the VEH3.8 application, a spark, injector control and gear shift method is used. Cutting off too many injectors creates a sense of engine stall which is unfavourable. In some cases, installing a secondary throttle can create a packaging problem. On sensing a wheel slip, the PCM responds to a torque reduction request from the ABS/TA module over the SAE J1850 link by controlling the spark, fuel and throttle, in that order, while the ABS/TA module controls the brakes. The simulation was performed for a vehicle moving on a uniform $\mu$-surface of packed snow ($\mu = 0.25$) with the driver demanding a wide-open throttle (WOT) for a run of 10 sec. starting on first gear. A WOT was selected as it provides the largest torque which produces the chance for the largest slip for a TA event.

Generally, one of the worst conditions experienced is when starting from rest on a slippery surface. Maintaining stability and driveability while making a tight turn on a slippery surface is also of great concern; a knowledge of the steering angle is also necessary, but such a condition is not evaluated in this simulation. The goal is to bring down the speed of the driven wheels to that of the vehicle in the shortest possible time while trying to achieve the best acceleration as demanded by the driver, without affecting stability, controllability and driveability of the vehicle. Figures 10–12 represent simulation of the VEH4.6. There are many more parameters that are required by the design engineer.
which are available in the simulation, but only some are shown due to length limitations. All simulations shown are for a uniform $\mu$-surface. Figure 10a–d shows control of slip with engine management techniques consisting of spark and throttle only. As shown, on detection of a slip condition, the speed of the driven wheels is reduced to follow the vehicle speed. Once the speed of the driven wheels is within a certain wheel slip threshold, the traction control is released and the wheels start to spin up again. This explains the oscillations in Figure 10a–b. This threshold is needed to prevent any false TA activity. By using a combination of the above strategies to control slip, it is possible to obtain optimum performance, stability and manoeuvrability. Figure 11a–f shows the simulation for the traction control event with the combined effect of the engine and brake control strategies. It can be seen that the wheel spin is brought under control faster with the combination of both strategies. The actual traction control data, used for validation, were obtained from winter testing. The simulation is sufficiently representative of the behaviour of the actual vehicle. Although the vehicle was tested on packed snow, the frictional coefficient is never uniform, but with the simulator we specified distinct coefficients of friction for various surfaces. Therefore we cannot expect an identical match.

![Figure 10a Engine rpm, with engine control only.](image)

![Figure 10b Driven wheel speed (mph).](image)
Figure 10c  Non-driven wheel speed (mph).

Figure 10d  Series throttle (degs).
Figure 11a  Engine rpm with all controls.

Figure 11b  Driven wheel speed (mph).
Figure 11c  Non-driven wheel speed (mph).

Figure 11d  Speed ratio.
Figure 11e  Series throttle (degs).

Figure 11f  Brake IN/OUT valve activation.
Figure 12a  Engine rpm.

Figure 12b  Driven wheel speed (mph).
Figure 12c  Speed ratio.

Figure 12d  Spark (degs).

Figure 11a shows the variation of engine speed, which should follow that of the driven wheel speed. Figures 10d and 11e show how the secondary throttle closes in reducing the engine torque, thereby reducing the drive torque at the wheels which reduces the application of the brakes drastically. Figure 11f represents the operation of the brake pulses on the driven wheel, showing the opening/closing of these inlet/outlet valves which creates the brake pressure and the corresponding brake torque is generated. As shown, the brakes come on only during the initial onset of slip in order to control slip quickly, thereafter engine intervention is used to control slip. When starting the vehicle
on an inclination and even in stop-go situations which are very common in busy cities, this wheel slip threshold needs to be adjusted dynamically. Different control strategies could be saved in files and later compared or even rerun depending on the requirement. This repeatable simulation allows the engineers to adjust, develop and validate the hardware and software for TA before installation in the vehicle. The final tests are still made at winter test sites; however, the performance of the first winter test runs will be much closer to the desired performance if the HIFL simulator, as described above, is used ahead of time. Simulations for VEH3.8 are not shown as it behaves similar to VEH4.6.

5 Model validation

The model was validated by comparing the model behaviour with that of the real vehicle when both were driven under identical input conditions. The intended application of the simulation is in traction control development. The simulation was initially done for a $\mu = 1.0$, which is a dry concrete surface, as illustrated in Figure 12a–d. As in Figure 12c, a relatively good match of the speed ratio is important as it decides the operating point in the torque-converter characteristics; this in turn determines the torque generated through the driveline producing the corresponding wheel speeds. Figure 12d shows the change in spark angle for a WOT on a $\mu = 1.0$. The initial spark is set by the PCM depending on the rpm at that instant. Since the real vehicle had a different idle rpm from that of the simulation, this explains the initial deviation in spark. We also repeated the simulation on a uniform $\mu$ surface as shown in Figure 11a–f, with $\mu = 0.25$ for that particular snow surface being used. The tests were also run on wet asphalt ($\mu = 0.6$ approx.) and ice ($\mu = 0.8$ approx., not shown). Under these various $\mu$-surfaces, the simulation showed very good resemblance to the data collected from the vehicle. We used methods presented in existing literature on simulation model validation as described by Balci (1990). The subjective validation techniques used were:

Event validation: Employed identical input patterns to compare the model with vehicle behaviour, e.g.: Inputs – primary throttle, identical road conditions. Outputs – engine rpm, series throttle, speed ratio, driven and non-driven wheel speeds. Graphical comparisons of the different variables over time were used to investigate similarities in periodicity, skewness, number and location of inflection points, rise and linearity, trends, etc.

Face validation: Potential users of the simulation and people knowledgeable about the system compared the results with the anticipated behaviour of the system to verify whether the simulation produced reasonable accuracy. This was used as the preliminary approach to validation.

Field tests: The simulation was run for various input conditions. The results were shown to be of a reasonable accuracy for model usage, which was used to develop confidence in the simulation.

Sensitivity analysis: The inputs were systematically changed from WOT to partial throttle, and from high to low $\mu$ and the corresponding effects on the model were observed. The results observed were as anticipated.

Submodel testing: Each submodel was developed on Matrix X, verified and validated over its normal operating range.
A statistical comparison was used to see how the simulation data corresponded with the actual data collected from winter tests. Different parameters were compared for various surfaces, and on average the simulation data were within 15% of the actual data; however, the simulation data for a run on a $\mu = 1.0$ (no slip) were within 5%.

The major source of deviation was in the uncertainty of the road surface condition on which the tests were performed. There was no method of calculating the actual road $\mu$-surface. The secondary spins change as the road surface changes. Since the tests were performed with new tyres, the tread has a significant added tractive effort. This will decrease the oscillations that were observed. In Figure 10a–d, the data collection was done when the vehicle began from rest, whereas the simulation could only be done when the vehicle rolled at 2.5 mph. A WOT from rest creates some instability in the simulation due to the tyre characteristics that were not modelled in this simulator; this is also seen from the actual data shown in Figure 10c.

6 Future work

More detailed models will be incorporated in the simulator which will, with the suspension model, include tyre patch and tread into the tyre model, and compensation for transient behaviour will be added to make the simulator more accurate and more generic. Since this will require faster computations to achieve real-time capabilities, digital signal processors (DSP) and/or transputers to speed-up the computation will be good alternatives. The DYNMOD simulator which runs on a VAX 4090 is the currently used simulator by the PCM strategists in development. Our experience will help in the set-up of a HITL DYNMOD model with the PCM in the loop. This will enable all new software updates to be verified on the bench, resulting in fewer problems in the vehicle.

7 Conclusion

A sufficiently detailed hardware-in-the-loop simulation model has been developed. HITL simulation has been quite widely used in the aerospace industry and is beginning to be popular in the automotive industry with the advent of complex electronic controls. The need to develop and produce reliable vehicles reducing the time between design and production is becoming more of an important issue, further supporting the role of HITL simulation. The object-oriented design approach was used, as the software structure reflects the structure of the application problem, since everything can be represented as objects and messages. The inherent parallelism in the software system can be modelled naturally. The data abstraction and encapsulation reduces the interdependencies between the software components and consequently facilitates modification. The advantage of the object-oriented approach is not only its closeness to the real world situation but also the ability to run the code allocated to each object on different processors like transputers or digital signal processors to speed up overall program execution. This enables constant model development with increasing complexity during the life cycle of the system. The structure and models are generic to be utilized for a variety of control system developments. It will be a very useful tool for both hardware and software engineers. This will reduce the time taken for hardware testing now done in the vehicle. The software
strategy engineers will have all the advantages of a dynamic real time simulation over the static testing now performed. We have verified our simulation model by comparing simulation results with the actual vehicle performance data obtained from winter testing. All the results show that our model represents the behaviour of a vehicle within 5-15% of the actual data for different run conditions, and therefore can be used for vehicle traction control system development. Although this paper shows the design and results for a particular type of vehicle, this modelling approach can be adapted for the simulation of any other vehicle as well.

Acknowledgements

The authors are indebted to many people who aided in this work by sharing their technical expertise. In particular, we would like to thank Barry Powell, Davor Hrovat, Alex Petiniunas, Doug Marsden, Larry Bush, Mike Cullen, Jim Kindree, Timber Hissong and Behrouz Ashrafi, only to name a few for their invaluable help in the modelling effort and correspondence. Mat Gerhart for providing the data and other useful information on the TA system. Chris Gee for all his expertise in circuit design and other hardware implementations. Sam Cole and Mark Hoffman for their support of this project.

References


