SAE TECHNICAL
PAPER SERIES

950308

An Overview of the HVE Vehicle Model

Terry D. Day
Engineering Dynamics Corp.

Reprinted from: New Developments in Vehicle Dynamics, Simulation, and Suspension Systems (SP-1074)

The Engineering Society For Advancing Mobility Land Sea Air and Space®
INTERNATIONAL

International Congress and Exposition
Detroit, Michigan
February 27 - March 2, 1995

400 Commonwealth Drive, Warrendale, PA 15096-0001 U.S.A. Tel: (412)776-4841 Fax:(412)776-5760

ISSN 0148-7191

Printed in USA 90-1203D/PG

950308

An Overview of the HVE Vehicle Model

Terry D. Day
Engineering Dynamics Corp.

ABSTRACT

Developers of vehicle dynamics simulation software inherently use a mathematical/physical model to represent the vehicle. This paper describes a pre-programmed, object-oriented vehicle model for use in vehicle dynamics simulations. This vehicle model is included as part of an integrated simulation environment, called HVE (Human-Vehicle-Environment), described in previous research [1,2]*. The current paper first provides a general overview of the HVE user and development environments, and then provides detailed specifications for the HVE Vehicle Model. These specifications include definitions for model parameters (supported vehicle types; vehicle properties, such as dimensions, inertias, suspensions; tire properties, such as dimensions and inertias, mu vs slip, cornering and camber stiffnesses; driver control systems, such as engine, transmission/differential, brakes and steering; restraint systems, such as belts and airbags). The paper also provides detailed specification of the HVE time-dependent vehicle output group parameters (kinematics, kinetics, tire data, wheel data, inter-vehicle connections, drivetrain, driver controls, contact surface forces, and restraint system forces).

VEHICLE DYNAMICS SIMULATION models are useful in several types of research. Vehicle manufacturers use vehicle simulation to assist in the design and development of chassis and suspension systems. Human factors experts use simulation to discover how drivers approach the task of driving. University researchers use simulation, both as a classroom learning tool and as a research tool for the study of vehicle and highway safety. Accident reconstructionists use simulation to study accident causation and avoidability. Recently, simulation has begun to see use in the evaluation of NASCAR and INDY car crashes in an effort to improve safety for race car drivers [3].

* Numbers in brackets designate references found at the end of the paper.

Developers of vehicle dynamics simulation software inherently use a mathematical/physical model (i.e., an object described by properties such as dimensions, inertias and mechanical constants) to represent the vehicle. The required complexity of the vehicle model is dependent upon the complexity of the motion being simulated. For example, planar, two-dimensional (2-D) motion requires a relatively simple vehicle model requiring only a dozen parameters, while the simulation of three-dimensional (3-D) motion requires over 100 parameters. Historically, researchers have been reticent to develop 3-D simulators because of the massive amounts of vehicle data required to execute them.

This paper describes a 3-D vehicle model available for use by developers of vehicle dynamics simulation programs. The model, called the HVE Vehicle Model, is included as part of an integrated simulation environment, called HVE (Human-Vehicle-Environment), described in previous research [1,2]. The purpose of this vehicle model is to provide a standard, pre-programmed model available to researchers. It is hoped that by providing such a robust model in pre-programmed form, researchers will be inclined to produce more sophisticated simulators, thus, improving the state of the art in vehicle dynamics simulation.

The purpose of this paper is to provide details of the HVE Vehicle Model so that researchers may assess the applicability and suitability of the model to their vehicle dynamics simulations.

OVERVIEW OF HVE

HVE is a computer environment for setting up and executing human and vehicle dynamics simulations. It may be viewed as a computer abstraction of the nine-cell matrix for accident reconstruction originally proposed by the late Dr. William Haddon, first Director of the National Highway Traffic Safety Administration [4]. The nine-cell matrix describes the possible interactions between humans, vehicles and their environment during the pre-crash, crash and post-crash phases of an accident.

HVE is not itself an accident reconstruction program. Rather, HVE is an interface for *running* accident reconstruction and simulation programs, much like Microsoft Windows™ is an interface for running PC programs.

The HVE interface is an integrated set of editors. The *Human Editor*, *Vehicle Editor* and *Environment Editor* are used for creating 3-D physical and visual models of humans, vehicles and environments. Once created, the interactions between these models may be simulated using *any* HVE-compatible human or vehicle simulation model in the HVE *Event Editor*. The program results may be displayed both numerically and *visually* by HVE. Using the HVE *Playback Editor*, simulations from several events may be edited into a single coherent sequence involving multiple humans and vehicles. The output may be routed to a display, printer, plotter or VCR. See reference 1 for further details.

HVE Environment

A block diagram for the HVE simulation environment is shown in Figure 1. Note that, conceptually, the HVE interface *surrounds* the simulation model. The interface is comprised of five modes: Human Mode, Vehicle Mode, Environment Mode, Event Mode and Playback Mode.

The HVE Developer's Toolkit [2] is a library of functions and data structures that provide the developer of a human or vehicle dynamics model access to the HVE interface.

The remainder of this document describes the details of the HVE Vehicle Model which is created using the HVE Vehicle Editor. Although the information is provided in the form expected by the programmer/developer (refer to Appendix A for the actual data structures), it should also be useful to any technical person wishing to understand the basic parameters which define the vehicle model.

VEHICLE MODEL

The HVE Vehicle Model is viewed and edited in the Vehicle Editor, one of five editing modes which comprise HVE's graphical user interface (see figure 2). In general, the Vehicle Editor allows the user to produce, from HVE's vehicle database, one or more vehicles to be included in the *Active Vehicles List*. One or more of the vehicles in this list may then be selected for study in a vehicle dynamics (or human dynamics) simulator.

This section of the paper describes the following details of the HVE Vehicle Model:

- Vehicle Database (General Parameters)
- Vehicle Properties (Model Inputs)
- Event-related Parameters (In-use Inputs)
- Output Parameters (Model Outputs)

Vehicle Database

The HVE Vehicle Database is a user-extendable library of vehicles selectable according to the following keys:

- Type (Passenger Car, Pickup, Multi-purpose, Van, Truck, Trailer, Dolly, Fixed Barrier, Movable Barrier)
- Year
- Make
- Model
- Body Style

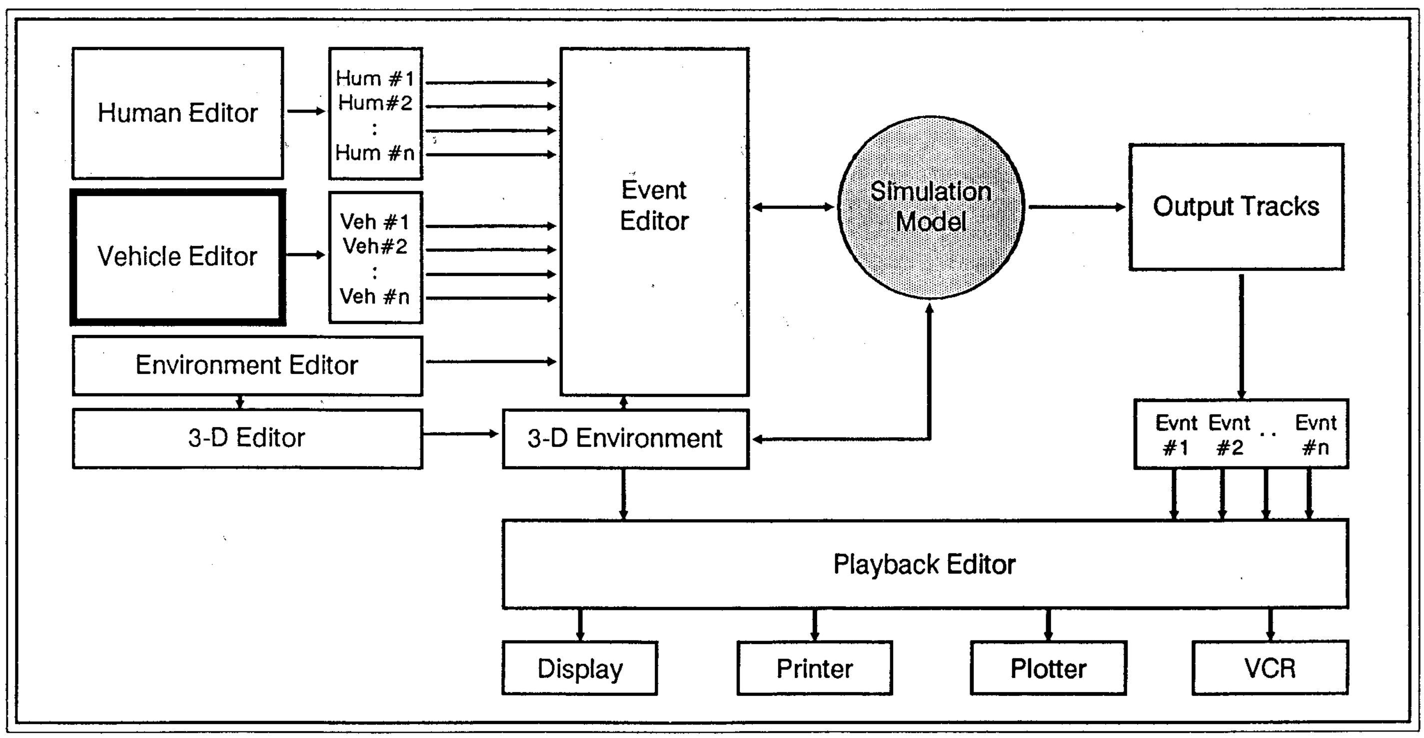

Figure 1 - Block Diagram for HVE application environment. The HVE Vehicle Model is created and edited in the Vehicle Editor (see heavy box). These vehicles may then be used in the Event Editor by any HVE-compatible simulation model.

The allowable vehicle types are fixed (see above). However, the *Year, Make, Model* and *Body Style* keys are user-definable, thus allowing the user to create and save new vehicles.

Once a vehicle is selected, the following general vehicle parameters are also defined and may be edited:

- Number of Axles (None, 1, 2 or 3)
- Driver Location (None, Left, Center or Right)
- Engine Location (None, Front, Mid or Rear)
- Drive Axle (None, 1, 2, ... - depending on the number of axles)
- Image Filename

The Vehicle Editor creates a *copy* of the selected vehicle for use in the current study. Therefore, changes to an individual vehicle do not affect the vehicle database. Any modified vehicle may be added to the vehicle database after modification.

The HVE vehicle database contains more than 200 vehicles of intermediate detail.

Vehicle Properties

The HVE Vehicle Editor displays the current vehicle in the *Vehicle Viewer* (see Figure 2). Input data categories

Table 1. Sprung Mass Parameters

Variable Name	Pgm Units	Comments
EXTERIOR SURFACE		
Dimensions	in	CG to Front, Right, Back and Left
A stiffness	lb/in	for selected surface
B Stiffness	lb/in^2	for selected surface
K_v Stiffness	lb/in^2	for selected surface
INTERTIAS		
Mass	$lb\text{-}sec^2/in$	sprung mass only
Roll, Pitch, Yaw Inertia	$lb\text{-}sec^2\text{-}in$	sprung mass only
CG LOCATION		
Δx, Δy, Δz	in	from standard CG
INTER-VEHICLE CONNECTIONS		
Front Type		None, Hitch, Kingpin, Pintle Eye
Front x,y,z Coords	in	
Rear Type		None, Ball, FifthWheel, Pintle Hook
Rear x,y,z Coords	in	
Rear Conect'n Radius	in	
Rear Conct'n Friction	dimensionless	
Rear Max Articul'n Ang	rad	
DRAG		
Air Drag Coefficient	$lb\text{-}sec^2/in^2$	incl. frontal area
Rolling Resistance Coef	lb-sec/in	
Rolling Resistance Const	lb	

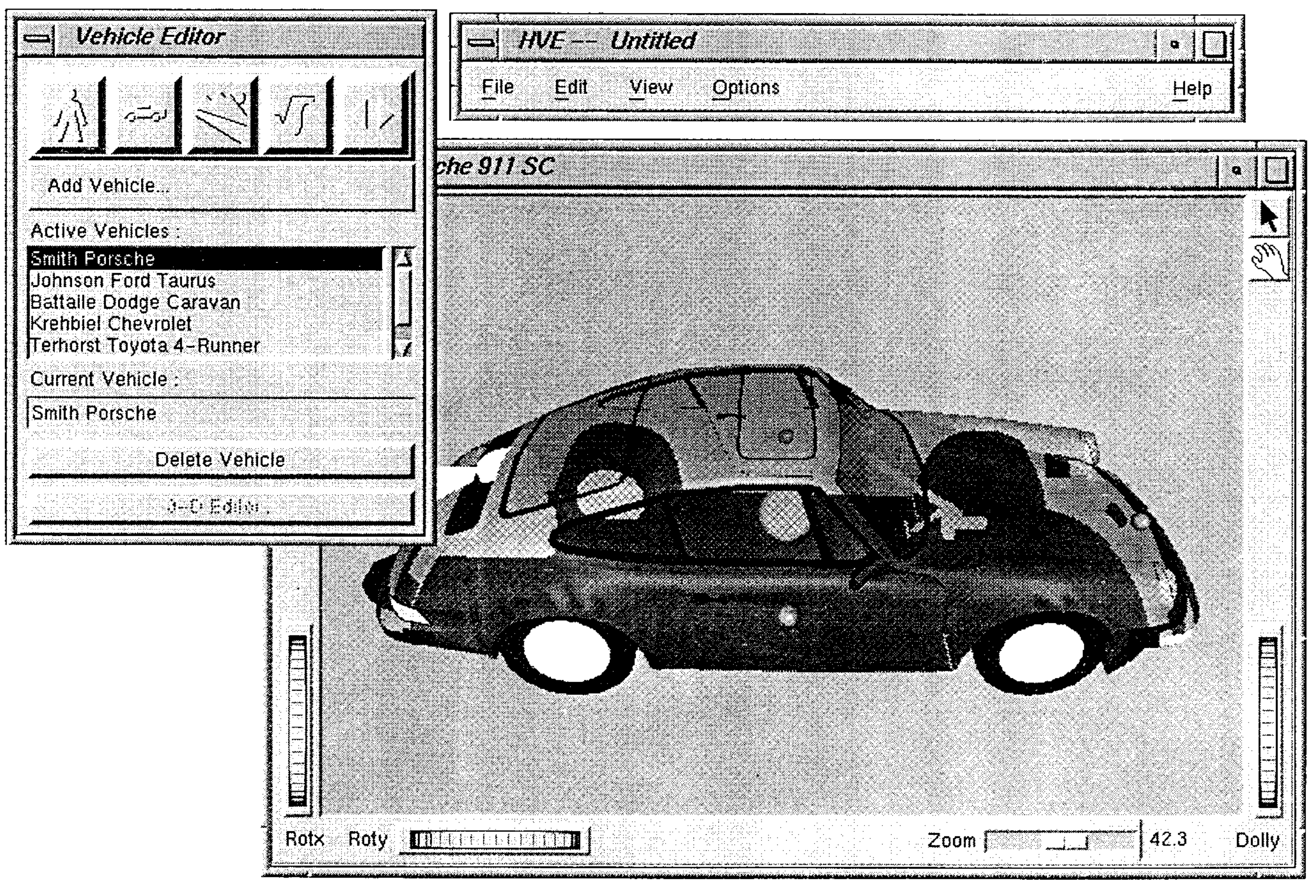

Figure 2 - HVE Vehicle Editor and Vehicle Viewer. The Vehicle Editor is used for selecting and editing HVE Vehicle Models.

for the current vehicle are selected by clicking the mouse on the desired item:

- Sprung Mass (at CG or exterior surfaces)
- Unsprung Mass (at wheel)
- Steering System
- Brake System
- Engine/Drivetrain
- Safety Systems

These input categories, and their associated parameters, are defined below.

Sprung Mass

The following data categories are available by clicking on the vehicle:

- Exterior Dimensions and Stiffness
- Inertial Parameters
- CG Location
- Inter-vehicle Connections
- Drag

The individual parameters are shown in Table 1.

Exterior Dimensions and Stiffness

The exterior dimensions assign the overall vehicle dimensions. These parameters are required for simulators which include a collision model.

All vehicle dimensions, as well as locations for vehicle-fixed objects, such as wheel locations and seat belt anchors, are based on a *standard* CG location.

Stiffness properties are also required for collision simulators, and allow the user to enter separate coefficients for the front, right, back and left sides. Most frontal stiffness coefficients were derived from barrier crash tests (see reference 5); side and rear coefficients were obtained from references 5 - 8.

Inertial Parameters

Mass is actually entered as weight and divided by the current value of the acceleration of gravity. Product of inertia, I_{xy}, is not included.

CG Location

The *inertial* CG location may be moved in the x,y,z directions. Other vehicle-fixed objects (e.g., wheels, seatbelt anchors and payload) do not move when the inertial CG is relocated.

Inter-vehicle Connections

These parameters are used to build multi-unit vehicles, such as tractor-trailers, from individual vehicle units of the proper type (i.e., Truck, Trailer, Dolly). The simulation code is responsible for ensuring that the vehicle units are compatible by comparing the rear connection type of the tow vehicle with the front connection type of the trailer. For example, if the tow vehicle has a fifth wheel, the trailer must have a kingpin.

Table 2. Unsprung Mass Parameters

Variable Name	Pgm Units	Comments
WHEEL LOCATION		
x,y,z location	in	relative to std. CG
SUSPENSION		
Type		Independent, Solid, Tandem Axles
Inter-tandem Load X-fer	dimensionless	tandems only
Ride Rate at Wheel	lb/in	
Auxilliary Roll Stiffness	in-lb/rad	
Roll Center Height	in	
Lateral Spring Spacing	in	
Damping Rate at Wheel	lb-sec/in	
Coulomb friction	lb	
Friction Null Band	in/sec	
Mass	$lb\text{-}sec^2/in$	
Roll/Yaw Inertia	$lb\text{-}sec^2\text{-}in$	
Stop Name		Jounce or Rebound
Max Deflection to Stop	in	
Linear Rate of Stop	lb/in	
Cubic Rate of Stop	lb/in^3	
Energy Ratio of Stop	%/100	
Caster	rad	
King Pin Inclination	rad	
Steering Offset	in	
Toe-in	in	
Suspension Deflection	in	for camber & half-track
Camber Table	rad	
Half-track Change Table	in	
Suspension Deflection	in	for anti-pitch
Anti-pitch Coefficient Table	lb/in-lb	
Roll Steer Constant	rad	rad/rad if solid axle
Roll Steer Linear Coef	rad/in	N/A if solid axle
Roll Steer Quadratic Coef	rad/in^2	N/A if solid axle
Roll Steer Cubic Coef	rad/in^3	N/A if solid axle
TIRES		
Dual Flag	boolean switch	TRUE or FALSE
Inter-dual Spacing	in	duals only
Unloaded Radius	in	
Initial Deflection Rate	lb/in	
Secondary Deflection Rate	lb/in	
Defl @ Secondary Rate	in	
Maximum Deflection	in	
Pneumatic Trail	in	
Aligning Torque Stiffness	in-lb/rad	
Mass	$lb\text{-}sec^2/in$	
Spin Inertia	$lb\text{-}sec^2\text{-}in$	
Peak Long. Friction	dimensionless	
Peak Lateral Friction	dimensionless	
Slide Friction	dimensionless	
Slip @ Peak Long. Friction	%/100	
Long Stiffness	lb/slip	
In-use Factor	dimensionless	
Cornering Stiffness	lb/rad	
In-use Factor	dimensionless	
Camber Stiffness	lb/rad	
In-use Factor	dimensionless	
Longitudinal Slip	%/100	
Lateral Slip	%/100	
BRAKES		
Time Lag	sec	
Rise Time	sec	
Push-out Pressure	lb/psi	
Brake Torque Ratio	in-lb/psi	
Proportioning On	boolean switch	TRUE or FALSE
Begin Proportioning Press	psi	
Proportioning Ratio	dimensionless	
Antilock On	boolean switch	TRUE or FALSE
Antilock Effectiveness	%/100	

Drag

The HVE environment model allows the user to specify wind velocity, direction, temperature and pressure. These environmental parameters are used to compute the air density and, thus, make possible the calculation of aerodynamic pressure acting on the vehicle. The aerodynamic drag coefficient includes frontal area.

Unsprung Mass

The following data categories are available by clicking on the individual wheels:

- Wheel Location
- Suspension Parameters
- Tire Parameters
- Wheel Brake Parameters

The individual parameters are shown in Table 2.

Wheel Location

The wheel location specifies the vehicle-fixed x,y,z coordinates of the wheel center relative to the standard CG. Right- and left-side wheels may have different values, allowing researchers to develop simulations which study vehicles damaged by impact.

Suspension Parameters

The HVE Vehicle Model permits four types of suspensions:

- Independent
- Solid Axle
- 4-Spring Tandem Axle
- Walking Beam Tandem Axle

Tandem axles are permitted when the vehicle has more than one axle (other logic is involved as well). When tandem axles are specified, an inter-axle load transfer is allowed. This parameter defines the redistribution of vertical load between tandem axles due to drive and brake torque.

Suspension properties are divided into the following categories:

- Springs and Shocks
- Inertia
- Jounce/Rebound
- Spindle Axis
- Camber
- Anti-pitch
- Roll Steer

Each category is discussed below (see also Table 2).

Springs and Shocks - These data provide all of the common parameters required for suspension modeling. Rates are effective at the wheel. Several dependencies exist in the suspension data. For example, roll center height and lateral spring spacing do not apply to independent suspensions.

Suspension Inertia - These values apply only to solid axle suspension types. User-entered weight is stored as mass. (If desired, suspension mass associated with independent suspensions may be included in wheel inertia.)

Jounce/Rebound - Individual values for suspension deflection limits apply to the jounce (compression) stop and the rebound (extension) stop.

Spindle Axis - The common suspension alignment parameters are available to the HVE Vehicle Model.

Camber - If the suspension type is independent, camber and half-track tables may be entered. These tables define camber and half-track change as a function of suspension deflection. If the suspension is a solid axle type, only a single value for camber angle per unit vehicle roll applies.

Anti-pitch - The suspension anti-pitch parameters available to the HVE Vehicle Model are defined in terms of vertical force per unit of wheel torque.

Roll Steer - If the suspension type is independent, roll steer coefficients for a 3^{rd} order polynomial curve fit may be entered which define roll steer as a function of suspension deflection. If the suspension is a solid axle type, only a single roll steer coefficient which defines wheel steer per unit vehicle roll applies.

Tire Parameters

HVE includes a user-extendable tire database which uses the following keys:

- Type (Passenger Car, Light Truck, Heavy Truck and Mobile Home)
- Manufacturer
- Model
- Size

The available tire types are pre-defined. Manufacturer, Model and Size are user-definable. The entered tire size string is parsed to confirm that a valid designation (according to the Tire and Rim Association [9]) has been entered. The size string is then used to assign the tire's default inside radius, outside radius and width.

The user may specify dual tires at a given wheel location by selecting the Dual Tires checkbox and entering the inter-dual spacing (inner and outer tires are assigned the same parameters).

The HVE Vehicle Model includes a *copy* of each tire. Therefore, changes to an individual tire do not affect the tire database.

Tire properties are divided into the following categories:

- Physical
- Friction Table
- Cornering Stiffness Table
- Camber Stiffness Table
- Slip-Rolloff Table

Tire data have been obtained from a variety of sources, including direct measurement and the Tire and Rim Association [9]. A majority of the remaining tire data comes from Calspan [10] and UMTRI [11,12] research. Each tire data category is discussed below (see also Table 2).

Physical Information - These parameters define non-speed- and load-dependent properties of the tire. The user-entered weight is stored as mass. If desired, the wheel weight may be increased for independent suspension types to account for unsprung mass.

Friction Table - These friction parameters define the shear force characteristics of the tire. A unique set of values may be entered for up to six test loads and speeds. The longitudinal data are used to produce a mu vs slip curve for the current load and speed. The longitudinal stiffness defines the slope of the mu-vs slip curve at zero slip. The in-use factor is a single value which may be used to vary the dependent (friction) values at all loads and speeds en masse. These parameters are displayed graphically for each load/speed combination. The Tire Friction Table dialog, the style of which is typical of many dialogs in the HVE Vehicle Editor, is shown in Figure 3.

Cornering Stiffness Table - The cornering stiffness, $C\alpha$, defines the lateral tire force produced per unit of tire slip. A unique cornering stiffness may be entered for up to six test loads and speeds. The in-use factor is a single value which may be used to vary the cornering stiffness at all loads and speeds en masse. $C\alpha$ vs Load is displayed graphically for each for each test speed.

Camber Stiffness Table - The camber stiffness, $C\gamma$, defines the lateral tire force per unit of tire camber. A unique camber stiffness may be entered for up to six test loads and speeds. The in-use factor is a single value which may be used to vary the camber stiffness at all loads and speeds en masse. An individual $C\gamma$ vs Load curve is displayed for each for each test speed.

Slip-Rolloff Table - Longitudinal rolloff defines the reduction in lateral tire force per unit of longitudinal slip; lateral rolloff defines the reduction in longitudinal force per unit of lateral slip. A unique set of longitudinal and lateral rolloff may be entered for up to six longitudinal and lateral slips. Individual graphs for longitudinal and lateral rolloff are displayed as a function of longitudinal slip for each lateral slip angle.

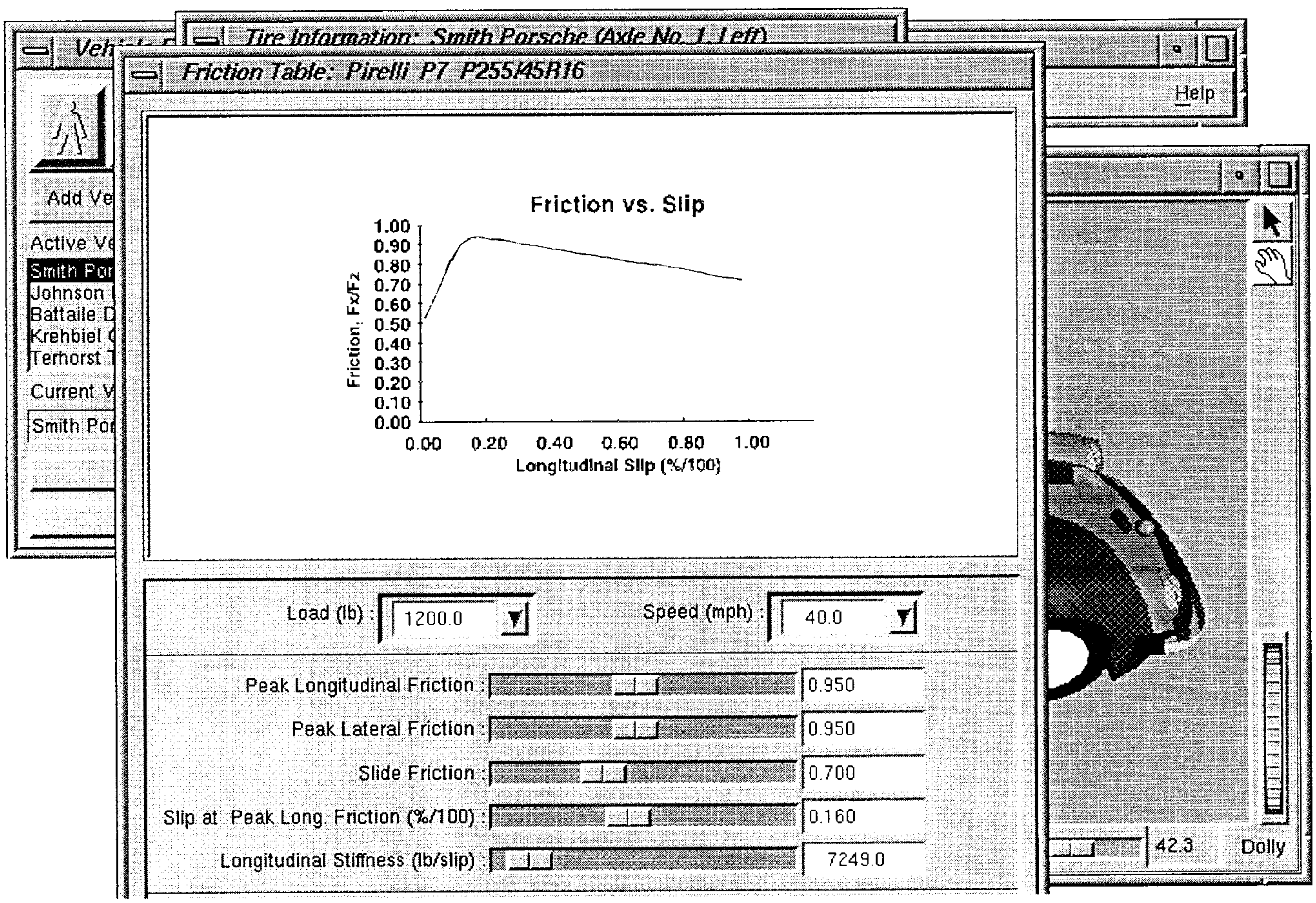

Figure 3- Typical input dialog (in this case, the Tire Friction Data dialog).

Table 3. Steering System Parameters

Variable Name	Pgm Units	Comments
AxleNum		Axle 1, 2 or 3
Steerable	boolean switch	TRUE or FALSE
Steering Gear Ratio	dimensionless	
Column Stiffness	in-lb/rad	
Linkage Stiffness	in-lb/rad	

Wheel Brake Assembly

These parameters are used for modeling brake torque, and are combined with the vehicle brake system parameters and driver control tables to determine the brake torques at individual wheels.

Time lag, rise time, push-out pressure and brake torque apply if the *At Pedal* option for brake table input is used (refer to section on Event editing). The vehicle simulation code is responsible for computing the current level of brake torque from these data.

If proportioning is selected, each individual wheel pressure may be modified by the proportioning ratio. Like the brake torque calculations, the simulation code is responsible for the calculations.

If antilock is selected, the antilock effectiveness coefficient may be used to determine the increase in longitudinal friction force above that associated with locked-wheel braking. Again, the simulation code is responsible for these calculations.

Steering System

Steering System parameters used in the HVE Vehicle Model are listed in Table 3. These parameters are available if the vehicle has axles and has been assigned a driver.

The steering gear ratio is the angle at the steering wheel divided by the resulting angle at the tire. It is used during Event editing when the driver enters the steer table using the *At Steering Wheel* option (see Event-related Parameters in the following section). Column stiffness defines torsional stiffness between the steering wheel and driver-side tire; linkage stiffness defines the torsional stiffness between the driver-side and passenger-side tires.

Even though an axle may not normally be steerable, the user may wish to make an axle steerable if collision damage has occurred. This allows steer angles to be entered during event editing.

Brake System

The Brake System properties used in the HVE vehicle are listed in Table 4. Like the steering system parameters, brake system properties are available only if the vehicle has axles and has been assigned a driver.

Table 4. Vehicle Brake System Parameters

Variable Name	Pgm Units	Comments
Pedal Ratio	psi/lb	

Table 5. Engine/Drivetrain Parameters

Variable Name	Pgm Units	Comments
ENGINE		
Throttle Position		Wide-Open, Closed
Engine Speed	rad/sec	
Engine Power	in-lb/sec	
TRANSMISSION		
Number of Trans gears		from 2 to 12
Gear Ratio for each gear		
DIFFERENTIAL		
Number of Diff gears from		1 to 3
Gear Ratio for each gear		

The pedal ratio defines the brake system pressure at the outlet of the master cylinder per unit of brake pedal application force. The pedal ratio is used during event editing when the driver creates the brake table using the *At Pedal* option as part of the brake torque calculations.

Drivetrain

Drivetrain parameters (see Table 5) may be made available to any vehicle simulator with an engine model, and are divided into the following categories:

- Engine
- Transmission
- Differential

Each category is described below.

Engine

The engine model uses the current throttle position (see Event-related Parameters), engine speed and final drive ratio to calculate drive torque at each drive wheel. These calculations are the responsibility of the simulation code. Separate power curves are defined for wide-open and closed throttle positions. Speed and power are entered; torque is calculated and displayed. (The vehicle model actually stores engine speed and torque.)

Transmission

Multiple transmissions (up to 12 speeds), each having different numbers of gears and gear ratios, may be defined for an HVE vehicle. Entered gear ratios for transmissions always include reverse and neutral (the ratio for neutral is always 0).

Differential

An HVE vehicle may also have up to three differentials. The differential has no neutral; otherwise, it is treated just like the transmission.

Safety Systems

Safety Systems are divided into the following categories:

- Contact Surfaces
- Belt Restraints
- Airbag Restraints

Table 6. Safety Systems Parameters

Variable Name	Pgm Units	Comments
BELT RESTRAINTS		
Location Name		9 total seat positions
Section Name		Lap/Shoulder
Device Installed	boolean switch	TRUE or FALSE
Vehicle x,y,z Anchor Points	in	
Linear Stiffness	lb/in	
Quadratic Stiffness	lb/in^2	
Cubic Stiffness	lb/in^3	
Damping Constant	lb-sec/in	
Maximum Force	lb	
Edge Constant	dimensionless	
Unloading Slope	lb/in	
AIRBAG SYSTEMS		
Location Name		9 total seat positions
Device Installed	boolean switch	TRUE or FALSE
Vehicle x,y,z Coordinates	in	
Initial Bag Radius	in	
Bag Length	in	
Max Deflection into Bag	in	
Initial Bag Pressure	lb/in^2	
Bag Membrane thickness	in	
Bag Elastic Modulus	lb/in^2	
Bag Rebound Modulus	lb/in^2	
Bag Bottom-out Modulus	lb/in^2	
Full Inflation volume	in^3	
Discharge Coefficient	dimensionless	
Vent Hole Area	in^2	
Vent Pressure	lb/in^2	
Gas Density	lb/in^3	
Gas Mass Flow Rate	lb/sec	
Gas Specific Heat	in-lb/lb	
Column angle	rad	
Column Max Collapse Dist	in	
Column Max Collapse Load	lb	
Convergence Criterion	lb	force tolerance
Backside Contact Surface		
CONTACT SURFACES		
Surface Name	char string	up to 30 chars
Location	boolean switch	interior or exterior
Coordinates	in	
Properties Database:		
Linear Stiffness	lb/in	
Quadratic Stiffness	lb/in^2	
Cubic Stiffness	lb/in^3	
Damping Constant	lb-sec/in	
Maximum Penetration	in	
Maximum Force	lb	
Edge Constant	dimensionless	
Unloading Slope	lb/in	

Restraint systems are provided in the HVE Vehicle Model for use in occupant simulations. The available restraint system properties are shown in Table 6.

Contact Surfaces

Contact surface properties are required when the vehicle is used in an occupant or pedestrian simulation.

Contact surfaces are 4-cornered planes defined by the user-entered coordinates for three corners (the simulation code is responsible for locating the fourth corner, thus defining the total contact area). The coordinates for three corners are entered clockwise; the right-hand rule is used to determine the front and back sides of the surface.

Table 7. Event-related Parameters

Variable Name	Pgm Units	Comments
PositionName		Initial, Begin Perception, Begin Braking, Impact, Separation, Point-on-curve, End-of-rotation, Final/Rest
X,Y,Z	in	earth-fixed
Roll, Pitch, Yaw	rad	
THROTTLE TABLE		
Table Type		Throttle Table Method
Time	sec	
% Wide-open Throttle	%/100	method 1
% Available Friction	%/100	method 2 (at each drive wheel)
Tractive Effort	lb	method 3 (at each drive wheel)
BRAKE TABLE		
Table Type		Brake Table Method
Time	sec	
% Pedal Force	lb	method 1
% Available Friction	%/100	method 2 (at each wheel)
Brake Force	lb	method 3 (at each wheel)
STEERING TABLE		
Table Type		Steer Table Method
Time	sec	
Steering Wheel Angle	rad	method 1
Wheel Plane angle	rad	method 2 (at each steer wheel)
GEAR TABLE		
Table Type		Trans or Diff
Time	sec	
Gear Number		
PAYLOAD		
Payload Exists	boolean switch	TRUE or FALSE
CG x,y,z Coordinates	in	relative to std. CG
Mass	lb-sec^2/in	
Roll, Pitch, Yaw Inertias	lb-sec^2-in	

By specifying the location as *interior* or *exterior*, the simulation code may reduce the number of surface interactions for which calculations must be performed (i.e., for occupant simulations, there is normally no need to calculate forces on exterior contact surfaces; for pedestrians, there is no need to calculate forces on interior contact surfaces).

Each surface is assigned physical properties shown in Table 6. HVE maintains a database of contact materials, such as foams, glazings and sheet metals. This database is not user-extendable, however, the assigned properties may be edited for each vehicle. Sources for the contact surface properties include references 13 and 14.

Belt Restraints

The HVE Vehicle Model provides a belt restraint system for each of nine occupant positions, and permits both lap and shoulder belt sections.

Most belt material properties were obtained from reference 15. Vehicle-fixed x,y,z anchor points are defined here; anchor points for the opposite end of the belt are attached to the occupant and are defined during event editing.

Airbag Restraints

The HVE vehicle airbag model is based on the model developed by Wayne State University [16], and uses the dimensional and thermodynamic parameters shown in Table 6. Bag length only applies if the selected position is not the driver position as defined earlier. *Backside Contact Surface* is the supporting mechanical surface on the opposite side of the airbag, and opposes the occupant force.

Event-related Parameters

Event-related parameters are *in-use factors* which affect the vehicle model during execution of the simulation. Because these parameters are not related to the vehicle, but, rather to the event, they are assigned to the vehicle during *event editing*. Event editing is the process of setting up the event for execution.

The event-related parameters in HVE include the following:

- Position/Velocity
- Driver Controls
- Damage
- Payload
- Contact
- Restraint Usage
- Collision Pulse

Of the above parameters, Position/Velocity, Driver Controls and Payload provide important data to the vehicle model when used for vehicle simulations. The parameters for these three data groups are shown in Table 7. The Contact, Restraint Usage and Collision Pulse parameters apply to human occupant and pedestrian simulation. See reference 17 for details regarding parameters relating to occupant and pedestrian simulation.

Position/Velocity

Vehicle simulations require initial positions and velocities for each degree of freedom. Linear and angular positions are supplied during event editing, either by direct manipulation of the vehicle in the environment, or by data entry into the position/velocity dialog. Velocities are assigned using the position/velocity dialog.

The simulation code is responsible for telling HVE which degrees of freedom to make available in the Position/Velocity dialog. For example, a yaw plane simulator would not request entries for roll, pitch and Z, whereas a 3-D simulator would request entries for all fields.

Additional positions may be assigned as visual *targets*. Targets are vehicles which are displayed at user-selected positions along the path, and allow the researcher to assess how closely the simulated path matches the desired (actual) path. The available target positions are *Begin Perception, Begin Braking, Impact, Separation, Point-on-curve, End-of-rotation* and *Final/Rest*.

Driver Controls

HVE vehicle simulations use open-loop driver tables to provide driver control parameters during execution. The vehicle model includes the following time-dependent driver controls:

- Throttle
- Brake
- Steer
- Gear Selection

Driver control parameters are listed in Table 7, and are described below.

Throttle Table

The Throttle Table provides three methods: The *Percent Wide-open-throttle* method provides the current throttle position. The simulation may use this value, along with the HVE Vehicle Model's drivetrain (see Table 5), to determine the current level of tractive effort at each drive tire. The *Tractive Effort* option allows the user to directly enter a desired tractive effort at each drive tire. The *Percent Available Friction* method allows the user to specify the fraction of available tire-ground friction which is being used to produce tractive effort.

Brake Table

The Brake Table provides three methods: The *Pedal Force* method provides the current brake pedal force. The simulation may use this value, along with the HVE Vehicle Model's brake system parameters (see Tables 2 and 4), to determine the current level of braking force at each tire. The *Braking Force* method allows the user to directly enter the desired brake force at each tire. The *Percent Available Friction* method allows the user to specify the fraction of available friction which is being used to produce brake force.

Steer Table

The Steer Table provides two methods: The *At Steering Wheel* method provides the current steering wheel angle and steering gear ratio to determine the steer angle at each steerable wheel. The *At Axle* method allows the user to directly enter a steer angle for each steerable wheel.

Gear Table

The Gear Selection Table allows the user to enter the current transmission gear selection. The gear table is used, along with the Throttle Table and other drivetrain data, to determine the tractive effort at each drive tire. The Gear Selection Table applies only if the current throttle table method is *Percent Wide-open-throttle*.

Payload

A payload may be added to the HVE Vehicle Model. Individual payloads may be assigned to each vehicle of a vehicle-trailer combination. The payload is located relative to the standard CG.

Table 8. HVE Vehicle Output Parameters

Variable Name	Pgm Units	Comments
SPRUNG MASS KINEMATICS		
X,Y,Z coordinates	in	
Roll, Pitch, Yaw angles	rad	
Path Radius	in	
Course Angle	rad	
Total Velocity	in/sec	
Sideslip angle	rad	
u, v, w linear velocity	in/sec	
p, q, r angular velocity	rad/sec	
Forward, Lateral velocity	in/sec	
Total acceleration	in/sec^2	
Forward, Lateral accel	in/sec^2	
Tang, Centrip accela	in/sec^2	
u-dot, v-dot, w-dot accel	in/sec^2	
p-dot, q-dot, r-dot accel	rad/sec^2	
SPRUNG MASS KINETICS		
$\Sigma F_{x,tire}$, $\Sigma F_{y,tire}$, $\Sigma F_{z,tire}$	lb	from tires
$\Sigma M_{x,tire}$, $\Sigma M_{y,tire}$, $\Sigma M_{z,tire}$	in-lb	from tires
$\Sigma F_{x,col}$, $\Sigma F_{y,col}$, $\Sigma F_{z,col}$	lb	from collision
$\Sigma M_{x,col}$, $\Sigma M_{y,col}$, $\Sigma M_{z,col}$	in-lb	from collision
$\Sigma F_{x,air}$, $\Sigma F_{y,air}$, $\Sigma F_{z,air}$	lb	from aerodynamics
$\Sigma M_{x,air}$, $\Sigma M_{y,air}$, $\Sigma M_{z,air}$	in-lb	from aerodynamics
$\Sigma F_{x,con}$, $\Sigma F_{y,con}$, $\Sigma F_{z,con}$	lb	from connection
$\Sigma M_{x,con}$, $\Sigma M_{y,con}$, $\Sigma M_{z,con}$	in-lb	from connection
TIRES		
x,y,z contact patch coords	in	veh-fixed
X,Y,Z contact patch coords	in	earth-fixed
long ($F_{x'}$) tire force	lb	rel to tire plane
lat ($F_{y'}$) tire force	lb	rel to tire plane
normal ($F_{z'}$) tire force	lb	rel to tire plane
overturning moment ($M_{x'}$)	in-lb	rel to tire plane
rolling resist moment (My')	in-lb	rel to tire plane
aligning torque ($M_{z'}$)	in-lb	rel to tire plane
loaded tire radius	in	
slip angle	rad	
skid flag	boolean switch	TRUE or FALSE
scuff flag	boolean switch	TRUE or FALSE
WHEELS		
x,y,z wheel center coords	in	relative to std CG
camber angle, gamma	rad	
spin angle	rad	
steer angle	rad	
wheel deflection dz	in	
dz/dt of wheel center	in/sec	
dgamma/dt	rad/sec	
spin velocity	rad/sec	
F_x, F_y, F_z at wheel center	lb	
F_z from suspension	lb	
F_z from damping	lb	
F_z from antipitch	lb	
drive torque	in-lb	
brake torque	in-lb	
brake pressure	psi	
INTER-VEHICLE CONNECTIONS		
roll, pitch, yaw art angle	rad	
roll, pitch, yaw art vel	rad/sec	
roll, pitch, yaw art accel	rad/sec^2	
F_x, F_y, F_z	lb	at connection
M_x, M_y, M_z	in-lb	at connection
DRIVETRAIN		
engine speed	rad/sec	
engine power	in-lb/sec	
engine torque	in-lb	
transmission gear ratio	dimensionless	
differential gear ratio	dimensionless	

Table 8. HVE Vehicle Output Parameters (cont)

Variable Name	Pgm Units	Comments
DRIVER CONTROLSr		
Throttle Position	%/100	% of WOT
Brake Pedal Force	lb	
Brake System Pressure	psi	@ master cylinder
Transmission Gear Number	dimensionless	
Differential Gear Number	dimensionless	
Steering Wheel Angle	rad	
CONTACTS		
Contact Name	string	
Ellipsoid Name	string	
x,y,z Contact Location	in	veh-fixed
Contact Deflection	in	
Contact Total Force	lb	
Contact Normal Force	lb	
Contact Friction Force	lb	
BELTS		
x,y,z Anchor Point	in	veh-fixed
Belt Angle About y, z Axes	rad	
Belt Strain	in	
Belt Tension	lb	
AIRBAGS		
Bag Position Name		
Bag Pressure	psi	
Bag Radius	in	
Contact Force	lb	

Output Parameters

The output parameters produced by the HVE Vehicle Model are called *vehicle output tracks*. Output tracks contain time-dependent results calculated by the simulation model.

The output tracks for the HVE Vehicle Model are divided into ten categories:

- Kinematics
- Kinetics
- Tire Data
- Wheel Data
- Connections
- Drivetrain
- Driver Inputs
- Occupant Contacts
- Belt Restraint Forces
- Airbag Restraint Forces

The available output track parameters for each category are shown in Table 8. From this list of available results, the simulation code tells HVE which individual variables it wishes to produce by setting a switch for each variable (this is explained in references 1, 2 and 18). In general, a 2-D simulator assigns yaw plane results (X,Y,Ψ), while a 3-D simulator may assign all the available outputs. Each output track group is described below.

Kinematics

The Kinematics output group contains position, velocity and acceleration results for each vehicle. The individual parameters are defined according to SAE J670e [19].

Kinetics

The Kinetics output group contains the summation of moments and forces acting on the vehicle. The source of these moments and forces may be from the tires, collision, aerodynamics and inter-vehicle connections (articulated vehicles).

Tires

The Tire output group contains the earth-fixed and vehicle-fixed coordinates of the tire contact patch and the tire moments and forces resolved according to the tire-fixed coordinate system [see SAE J670e [19]). Data are produced for single and dual tires. Skid and scuff flags, indicating tire saturation, are also included (the simulation's tire model must set these flags). The earth-fixed location of the tire contact patch is used to draw simulated tire marks.

Wheels

The Wheel output group contains vehicle-fixed wheel center coordinates and other wheel geometry parameters. It also contains the vehicle-fixed components of the tire forces and individual force contributions from suspension, damping and anti-pitch. Current levels of drive torque, brake torque and wheel cylinder pressure are also available.

Inter-vehicle Connections

The Inter-vehicle Connections output group applies to articulated vehicles, and contains the current articulation angles, angular velocities and accelerations for each articulated vehicle, relative to the towing vehicle. Inter-vehicle moments and forces at the connection are also included.

Drivetrain

The Drivetrain output group contains the current levels of engine output, and transmission and differential ratios. This group requires the *Percent Wide-open-throttle* driver input method.

Driver Controls

The Driver Controls output group contains the current levels of driver inputs (throttle, steering, braking and gear selection). The availability of these outputs is dependent upon the options used during Event editing (e.g., the current level of steering wheel input requires use of the *At Steering Wheel* steer table option).

Contacts

The Contacts output group contains the parameters associated with interactions between occupant ellipsoids and vehicle contact surfaces. Vehicle-fixed contact locations, force and deflection results are included.

Belts

The Belts output group contains the parameters associated with interactions between the occupant and vehicle belt restraint systems, and includes belt stretch and tension.

Airbags

The Airbags output group contains the parameters associated with interactions between the occupant and the vehicle airbag system, and includes current levels of airbag pressure and force

Static Reports

In addition to the time-dependent output track parameters described above, additional output information related to the HVE vehicle is monitored. These results are displayed in the form of static reports, and include:

- **Messages** - Textual information relevant to the simulation
- **Accident History** - Time, simulated position and velocity for each of the specified positions
- **Vehicle Data** - A report containing a list of the HVE Vehicle Model parameters which were actually *used* by the simulation (Note that, although the HVE Vehicle Model may contain several hundred parameters, the simulation might *use* only ten or 20.)
- **Damage Data** - A report containing a numeric description of the user-entered or simulated damage profile
- **Damage Profile** - A graphical report containing a view of the vehicle and its user-entered or simulated damage profile
- **Momentum Diagram** - A graphical report containing a vector diagram showing the pre- and post-impact momentum vectors in a classical momentum diagram

Whereas the output track groups produce results at each simulation output interval, the above static reports are produced only once; this occurs at the end of the run.

DATA SOURCES

Vehicle data sets have been compiled from a wide variety of sources, including direct measurement, the Detroit Public Library, and a substantial number of references found in the literature. A comprehensive databook, containing all the parameters and their sources, has been compiled for each individual vehicle in the HVE Vehicle Database [20].

FUTURE WORK

The HVE vehicle model includes *measured* suspension parameters for properties such as spring rates, roll rates and suspension geometry (caster, camber, kingpin axis and offset). Another approach under consideration is to include a *Suspension Builder* option which would allow the user to specify dimensional, inertial and compliance properties of individual suspension components using a modeler. The simulation code could then calculate the required properties directly from the suspension model.

The HVE user environment currently allows only open-loop driver tables for throttle, braking and steering input. A closed-loop, path follower input table is under consideration for future versions of HVE.

The current version of the HVE Vehicle Model includes a rather simplistic anti-lock model. More robust models exist (e.g., [21]), and it is anticipated that the HVE Vehicle Model's anti-lock model will be improved.

DISCUSSION

The HVE Vehicle Model may be used by relatively simple, 2-D simulators as well as sophisticated, 3-D simulators. Although the HVE Vehicle Model contains literally hundreds of parameters per vehicle, the programmer may select as many or as few of these parameters as are necessary to meet the needs of the simulation model.

The HVE Vehicle Model is included as part of the HVE interface specification [1]. As an evolving standard, the HVE Vehicle Model and its associated vehicle database could be incorporated into any vehicle simulator via a relatively simple data translator.

Because it is an evolving standard, simulation developers are likely to develop simulations which require a few parameters not included in the HVE Vehicle Model. For parameters not included in the vehicle model, the developer may hard-code the values (this is only reasonable for parameters which seldom change) or use a separate file to be called by the simulation code. However, the developers of HVE are encouraging these simulation developers to participate and make suggestions for new vehicle parameters consistent with the needs of vehicle dynamics researchers. Researchers are also encouraged to extend the use of the vehicle database by creating new vehicle datasets.

A typical vehicle simulation program using the HVE Vehicle Model is built using the HVE Developer's Toolkit [2,18] and thus takes advantage of the HVE 3-dimensional interface for setting up, executing and viewing simulations.

The HVE Developer's Toolkit is written in C and assumes the simulation will be programmed in C or C^{++}. FORTRAN programs have not yet been included. It is, however, possible that FORTRAN programs could be incorporated using C or C^{++} "wrappers" around the FORTRAN program. This requires further research.

The HVE interface has been developed for use on Silicon Graphics (SGI) workstations. It is written in C^{++}, and leverages off the SGI Open GL and video libraries. Open Inventor [22] is used to visualize all HVE objects (humans, vehicle and environments). HVE requires a level of graphics processing power not yet available on personal computers (see reference 1 for performance comparisons). The HVE executable program is approximately 10 megabytes, requires a minimum of 32 megabytes of random access memory, and at least 1 gigabyte of hard disk space. An individual vehicle binary file is 20,743 bytes (without 3-D image geometry data).

SUMMARY

This paper has presented a detailed overview of the features of the HVE Vehicle Model. Individual input and output parameters were presented and discussed.

The main purpose of the HVE Vehicle Model is to provide a standard model available for use by the vehicle simulation community. The model may be used directly from within the HVE simulation environment or included as a vehicle standard for use in other simulation environments.

As a result of the availability of the HVE Vehicle Model, it is hoped that researchers will consider enhancing their current simulations as well as developing new and more advanced vehicle simulation applications.

TRADEMARKS

REFERENCES

1. Day, T.D., "A Computer Graphics Interface Specification for Studying Humans, Vehicles and Their Environment," SAE Paper No. 930903, Society of Automotive Engineers, Warrendale, PA, 1993.

2. Day, T.D., "An Overview of the HVE Developer's Toolkit," SAE Paper No. 940923, Society of Automotive Engineers, Warrendale, PA, 1994.

3. Kenefect, M., "Crash Safety in Racing Cars," SAE Paper No. 930999, Society of Automotive Engineers, Warrendale, PA, 1993.

4. Lee, S.N., Fell, J.C., "An Historical Review of the National Highway Safety Administration's Field Accident Investigation Studies," NHTSA, Washington, DC, 1988.

5. *Crush Stiffness Coefficients for Vehicle Model Years 1960 - 1994*, Neptune Engineering, Clovis, CA, 1994.

6. Prasad, A.K., "CRASH3 Damage Algorithm Reformulation for Front and Rear Collisions," SAE Paper No. 900098, Society of Automotive Engineers, Warrendale, PA, 1990.

7. Strother, C.E., Woolley, R.L., James, M.B., "A Comparison Between NHTSA Crash Test Data and CRASH3 Frontal Stiffness Coefficients," SAE Paper No. 900101, Society of Automotive Engineers, Warrendale, PA, 1990.

8. Woolley, R.L., Strother, C.E., James, M.B., "Rear Stiffness Coefficients Derived From Barrier Crash Data," SAE Paper No. 910120, Society of Automotive Engineers, Warrendale, PA, 1991.

9. *1993 Yearbook*, The Tire and Rim Association, Inc., Copley, OH, 1993.

10. Tapia, G.A., "Extended Tire Testing," Calspan Report No. 6871-V-1, Calspan Corporation, Buffalo, NY, 1983.

11. Fancher, P.S., Bernard, J.E., "Vehicle-In-Use Limit Performance and Tire Testing," HSRI Report No. UM-HSRI-PF-75-1-1, University of Michigan, Ann Arbor, MI, 1975.

12. HSRI Flatbed Tire Test Results, University of Michigan, 1981.

13. Segal, D.J., Kamholz, L.R., Griffith, D.G., "Vehicle Component Characterization, Vol. II: DataAppendices," DOT-HS-807-055, MGA Research Corp., Akron, NY, 1987.

14. Sanders, W.T., et al, "Performance Comparison of Air Damped Padding with Other Padding Materials," SAE Paper No. 920351, Society of Automotive Engineers, Warrendale, PA, 1992.

15. *Seat Belt Webbing Tests in Accordance with FMVSS 209*, Test Report No. DTB04R91-0393, Dayton T. Brown, Engineering and Test Division, LI, NY, 1991.

16. Bennett, R.O., Bowman, D.H., Robbins, D.H., "Supplement to the HSRI Three-Dimensional Crash Victim Simulation Report," HSRI Report No. UM-HSRI-77-20, University of Michigan, Ann Arbor, MI, 1977.

17. Day, T.D., "An Overview of the HVE Human Model," SAE Paper No 950659, Society of Automotive Engineers, Warrendale, PA, (to be presented at 1995 SAE International Congress, February, 1995).

18. *HVE Developer's Toolkit*, Engineering Dynamics Corporation, Beaverton, OR, 1994 (currently under development).

19. *Vehicle Dynamics Terminology*, SAE Technical Report No. J670e, Society of Automotive Engineers, Warrendale, PA, 1976.

20. *HVE Vehicle Database*, Engineering Dynamics Corporation, Beaverton, OR, 1994 (currently under development).

21. MacAdam, C.C., Fancher, P.S., Hu, G.T., Gillespie. T.D., "A Computerized Model for Simulating the Braking and Steering Dynamics of Trucks, Tractor-semi-trailers, Doubles and Triples - Phase 4 User's Manual," Report No. UM-HSRI-80-58, HSRI, University of Michigan, June, 1980.

22. *The Inventor Mentor*, Wernecke, J., Addison-Wesley Publishing Co., Menlo Park, CA, December, 1993.

Appendix A - HVE Vehicle Model Data Structures

Review of this structure provides an overview of the HVE Vehicle Model. Simulation programmers familiar with the C programming language may use this information to design vehicle simulations; it is also useful for providing a basic understanding of the model's parameters.

```
* vehicle.h
  Vehicle Data structure (12-1-94)
*/
    struct VehicleData {
      long  Id;
      char  Name[MAXNAMELENGTH];
      SHORT  Type;
      char  Year[MAXNAMELENGTH];
      char  Make[MAXNAMELENGTH];
      char  Model[MAXNAMELENGTH];
      char  Style[MAXNAMELENGTH];
      char  ImageFilename[MAXNAMELENGTH];
      SHORT  NumAxles;
      SHORT  DriverSide;
      SHORT  EngineLocation;
      SHORT  DriveAxle;

      struct VehicleColor {
        FLOAT r;
        FLOAT g;
        FLOAT b;
      } Color;

      FLOAT ChangeCG[3];

      struct FrontDimension {
        FLOAT  Xf;
        FLOAT  FrontOverhang;
        FLOAT  OverallLength;
      } CGtoFront;

      struct RearDimension {
        FLOAT  Xr;
        FLOAT  RearOverhang;
        FLOAT  OverallLength;
      } CGtoRear;

      struct RightDimension {
        FLOAT  Yr;
        FLOAT  OverallWidth;
      } CGtoRight;

      struct LeftDimension {
        FLOAT  Yl;
        FLOAT  OverallWidth;
      } CGtoLeft;

      struct VehicleStiffness {
        FLOAT  Astf;
        FLOAT  Bstf;
        FLOAT  Kstf;
      } Stiffness[MAXSIDES];

      struct VehicleInertia {
        FLOAT  TotalWeight;
        FLOAT  TotalMass;
        FLOAT  SprungInertia[3];
      } Inertia;

      struct VehicleContactSurface {
        SHORT  NumSurfaces;
        SHORT  CurrentSurface;
        struct Surface {
          char  Name[MAXNAMELENGTH];
          SHORT  Location;  /* interior or exterior */
          FLOAT  Coord[MAXCORNERS][3];
          struct ContactMaterialProperty {
            char  MaterialName[MAXNAMELENGTH];
            FLOAT  LinStf;
            FLOAT  QuadStf;
            FLOAT  CubicStf;
            FLOAT  DampConst;
            FLOAT  PenetrnMax;
            FLOAT  ForceMax;
            FLOAT  EdgeConst;
            FLOAT  UnloadStf;
          } Property;
        } Surface[MAXCONTACTS];
      } Contact;

      struct VehicleBelt {
        SHORT CurrentLocation;
        struct BeltLocation {
          SHORT CurrentSection;
          struct BeltSection {
            BOOLEAN DeviceInstalled;
            struct BeltProperty {
              FLOAT Coord[3];  /* veh anchor pts */
              FLOAT  LinStf;
              FLOAT  QuadStf;
              FLOAT  CubicStf;
              FLOAT  DampConst;
              FLOAT  ForceMax;
              FLOAT  UnloadStf;
            } Property[2];     /* rt and lt sides */
          } Section[2];        /* Torso or Lap    */
        } Location[MAXSEATLOCATIONS];  /* 9 pos locs */
      } Belt;

      struct VehicleAirbag {
        SHORT  CurrentLocation;
        struct AirbagProperties {
          BOOLEAN  DeviceInstalled;
          FLOAT  Coord[3];
          FLOAT  Radius;
          FLOAT  Length;
          FLOAT  Pressure;
          FLOAT  Thickness;
          FLOAT  Volume;
          FLOAT  VentCoef;
          FLOAT  VentArea;
          FLOAT  VentPress;
          FLOAT  DeflMax;
          FLOAT  ConvergCriterion;
          FLOAT  Elastic;
          FLOAT  ElasticReb;
          FLOAT  ElasticBotm;
```

Appendix A (cont).

```
      FLOAT  GasRho;
      FLOAT  GasFlowrate;
      FLOAT  GasCp;
      FLOAT  ColmDist;
      FLOAT  ColmLoad;
      FLOAT  ColmAngle;
      char  CurrentBackplane[MAXNAMELENGTH];
    } Property[MAXSEATLOCATIONS];  /* 9 pos locs */
} Airbag;

struct VehicleConnection {
  struct FrontConnection {
    SHORT  Type;
    FLOAT  Coord[3];
  } Front;
  struct RearConnection {
    SHORT  Type;
    FLOAT  Coord[3];
    FLOAT  Radius;
    FLOAT  Friction;
    FLOAT  ArticMax;
  } Rear;
} Connection;

struct VehicleDrag {
  FLOAT  Cd;
  FLOAT  LinearResist;
  FLOAT  Const;
} Drag;

struct VehicleDrivetrain {
  struct EngineData {
    SHORT CurrentType;
    struct ThrottleStatus {
      SHORT  TableLen;
      FLOAT  Table[MAXENGINETABLE][3];                /*speed,power,torq*/
    } Status[2];        /* WOT, Closed    */
  } Engine;

  struct TransmissionData {
    SHORT  CurrentTransType;
    FLOAT  Ratio[MAXTRANSGEARS][MAXTRANSRATIOS];
  } TransData;

  struct DifferentalData {
    SHORT  CurrentDiffType;
    FLOAT  Ratio[MAXDIFFGEARS][MAXDIFFRATIOS];
  } DiffData;
} Drivetrain;

 struct VehicleWheel {
  struct WheelLocation {
    FLOAT  Coord[3];
  } Location;

  struct VehicleSuspension {
    SHORT  CurrentSuspType;    /* ind, w-beam or 4-sprg  */
    FLOAT  InterTandemLoadXfer; /* per tandem axle set   */
    struct VehicleSpringShock {
      FLOAT  LinearRate;
      FLOAT  RollStf;              /* per axle         */
      FLOAT  RollCtrHt;           /* per SOLID axle*/
      FLOAT  LatSpringSpace;  /* per SOLID axle*/
      FLOAT  DampRate;
      FLOAT  Friction;
      FLOAT  Hysteresis;
    } Spring;

    struct VehicleSuspensionInertia {
      FLOAT  SolidAxleWeight;      /* per SOLID axle*/
      FLOAT  SolidAxleMass;        /* per SOLID axle*/
      FLOAT  SolidAxleInertia;     /*per SOLID axle*/
    }  Inertia;

    struct VehicleDeflection {
      SHORT  CurrentStop;          /* Upper or Lower */
      struct StopData {
        FLOAT  MaxDeflection;
        FLOAT  StopLinearRate;
        FLOAT  StopCubicRate;
        FLOAT  StopEnergyRatio;
      } Data[2];  /* Jounce, Rebound */
    } Deflect;

    struct VehicleSpindle {
      FLOAT  Caster;
      FLOAT  KingpinIncl;
      FLOAT  Offset;
      FLOAT  ToeIn;
    } Spindle;

    struct VehicleCamber {
      FLOAT  Const;        /* Solid Axle */
      SHORT  TableLen;
      FLOAT  Data[MAXCAMBERTABLE][3];
              /* defl, camb, 1/2track */
    } Camber;

    struct VehicleAntiPitch { /*defl,AntiPitch*/
      SHORT  TableLen;
      FLOAT  Data[MAXCAMBERTABLE][2];
    } AntiPitch;      /* ind or solid axle */

    struct VehicleRollSteer {
      FLOAT  Coef;          /* solid axle */
      FLOAT  Const;         /* ind axle   */
      FLOAT  Linear;
      FLOAT  Quad;
      FLOAT  Cubic;
    } RollSteer;
  } Suspension;

  struct VehicleTire {
    char  Name[MAXNAMELENGTH];
    char  Type[MAXNAMELENGTH];
    char  Mfr[MAXNAMELENGTH];
    char  Model[MAXNAMELENGTH];
    char  Size[MAXNAMELENGTH];
    BOOLEAN IsDual;
    FLOAT  DualSpace;
    struct PhysicalData {
      FLOAT  UnloadedRadius;
      FLOAT  InitialRideRate;
      FLOAT  SecondRideRate;
      FLOAT  SecondDefl;
      FLOAT  MaxDefl;
      FLOAT  PneumaticTrail;
      FLOAT  AlignTorqCoef;
      FLOAT  SpinInertia;
      FLOAT  Weight;
      FLOAT  Mass;
    } Physical;

    struct FrictionTable {
      SHORT  NumTableLoads;
      FLOAT  Load[MAXTIRETABLE];
      SHORT  NumTableSpeeds;
      FLOAT  Speed[MAXTIRETABLE];
      FLOAT  InUse;

      struct MuData {     /* load and speed */
        FLOAT  MuXPeak;
        FLOAT  MuYPeak;
        FLOAT  MuSlide;
        FLOAT  PeakPcnt;
        FLOAT  LongStiff;
      } Mu[MAXTIRETABLE][MAXTIRETABLE];
    } Friction;

    struct CalfaTable {
      SHORT  NumTableLoads;
      FLOAT  Load[MAXTIRETABLE];
      SHORT  NumTableSpeeds;
      FLOAT  Speed[MAXTIRETABLE];
      FLOAT  InUse;
              /* load      speed */
      FLOAT  Data[MAXTIRETABLE][MAXTIRETABLE];
    } Calfa;

    struct CgammaTable {
      SHORT  NumTableLoads;
      FLOAT  Load[MAXTIRETABLE];
      SHORT  NumTableSpeeds;
      FLOAT  Speed[MAXTIRETABLE];
      FLOAT  InUse;
              /* load      speed */
      FLOAT  Data[MAXTIRETABLE][MAXTIRETABLE];
    } Cgamma;

    struct RollOffTable {
      SHORT  NumTableSlips;
      FLOAT  Slip[MAXTIRETABLE];
      SHORT  NumTableAngles;
      FLOAT  Angle[MAXTIRETABLE];
      FLOAT  InUse;

      struct RollOffData {
        FLOAT  Long;
        FLOAT  Lat;
            /* slip      angle */
      } Data[MAXTIRETABLE][MAXTIRETABLE];
    } RollOff;
  } Tire;

  struct VehicleWheelBrake {
    FLOAT  LagTime;
    FLOAT  RiseTime;
    FLOAT  PushoutPress;
    FLOAT  TorqueRatio;
    BOOLEAN IsProportion;
    FLOAT  ProportionPress;
    FLOAT  ProportionRatio;
    BOOLEAN IsAntilock;
    FLOAT   AntilockEffectiveness;
  } Brake;
} Wheel[MAXHVEAXLES][2]; /* right and left sides */

struct VehicleBrakeSystem {
  FLOAT  PedalRatio;
} Brakes;

struct VehicleSteerSystem {
  SHORT CurrentAxle;
  struct SteerSystemData {
    BOOLEAN  IsSteerable;
    FLOAT    Ratio;
    FLOAT    ColumnStiffness;
    FLOAT    LinkageStiffness;
  } Data[MAXHVEAXLES];
} Steering;
};

/* End of vehicle.h */
```