



PB99-162794

VDM RoAD User Reference Manual

Version 1.0

UMTRI-98-20-2

April 1999

REPRODUCED BY: **NTIS**
U.S. Department of Commerce
National Technical Information Service
Springfield, Virginia 22161

PROTECTED UNDER INTERNATIONAL COPYRIGHT
ALL RIGHTS RESERVED.
NATIONAL TECHNICAL INFORMATION SERVICE
U.S. DEPARTMENT OF COMMERCE

Technical Report Documentation Page

1. Report No. FHWA-RD-98-169		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle VDM ROAD USER REFERENCE MANUAL				5. Report Date	
				6. Performing Organization Code	
				8. Performing Organization Report No. UMTRI-98-20-2	
7. Author(s) Michael W. Sayers				10. Work Unit No. (TRAIS)	
9. Performing Organization Name and Address The University of Michigan Transportation Research Institute 2901 Baxter Road Ann Arbor, Michigan 48109				11. Contract or Grant No. DTFH 61-93-R-00142	
				13. Type of Report and Period Covered Dec. 1993 - Nov. 1997	
12. Sponsoring Agency Name and Address Office of Engineering Research and Development Federal Highway Administration 6300 Georgetown Pike McLean, Virginia 22101-2296				14. Sponsoring Agency Code	
				15. Supplementary Notes Leonard C. Meczkowski, COTR (HSR-20)	
16. Abstract <p>The main objective of this project was to select among existing vehicle dynamics simulation programs and link them through a common database to commercial roadway CAD programs. The existing TruckSim and AutoSim technologies were selected and extended for this project to create a software package called VDM RoAD (Vehicle Dynamic Models Roadway Analysis and Design). VDM RoAD simulates the vehicle dynamics behavior of cars, trucks, and two-unit combination vehicles (tractor-semitrailers) on 3D road designs. The software runs under Windows 95 and Windows NT and shows results graphically with plots and animations. The software can be used by engineers who are not experts in simulation or even vehicle dynamics.</p> <p>The new software was developed with the intention of adding it to an interactive highway safety design model (IHSDM) that systematically considers safety of roadway and roadside design elements in creating cost-effective highway design alternatives. Although VDM RoAD is designed to work with other parts of the IHSDM, further integration is not recommended because most factors of performance that are of interest to roadway designers do not require vehicle dynamics simulation. This document describes the software architecture and provides instructions to potential users.</p>					
17. Key Words Computer simulation, vehicle dynamics, rollover, roadway design, geometry, IHSDM			18. Distribution Statement No restrictions. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 348	22. Price

METRIC (SI*) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol When You Know Multiply By To Find Symbol

LENGTH

in	inches	2.54	millimetres	mm
ft	feet	0.3048	metres	m
yd	yards	0.914	metres	m
mi	miles	1.61	kilometres	km

AREA

in ²	square inches	645.2	millimetres squared	mm ²
ft ²	square feet	0.0929	metres squared	m ²
yd ²	square yards	0.836	metres squared	m ²
mi ²	square miles	2.59	kilometres squared	km ²
ac	acres	0.395	hectares	ha

MASS (weight)

oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2000 lb)	0.907	megagrams	Mg

VOLUME

fl oz	fluid ounces	29.57	millilitres	mL
gal	gallons	3.785	litres	L
ft ³	cubic feet	0.0328	metres cubed	m ³
yd ³	cubic yards	0.0765	metres cubed	m ³

NOTE: Volumes greater than 1000 L shall be shown in m³.

TEMPERATURE (exact)

°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C
----	------------------------	----------------------------	---------------------	----

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol When You Know Multiply By To Find Symbol

LENGTH

mm	millimetres	0.039	inches	in
m	metres	3.28	feet	ft
m	metres	1.09	yards	yd
km	kilometres	0.621	miles	mi

AREA

mm ²	millimetres squared	0.0016	square inches	in ²
m ²	metres squared	10.764	square feet	ft ²
km ²	kilometres squared	0.39	square miles	mi ²
ha	hectares (10 000 m ²)	2.53	acres	ac

MASS (weight)

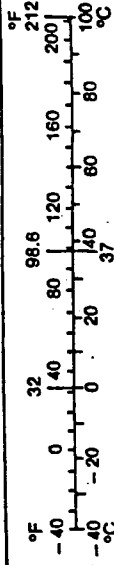
g	grams	0.0353	ounces	oz
kg	kilograms	2.205	pounds	lb
Mg	megagrams (1 000 kg)	1.103	short tons	T

VOLUME

mL	millilitres	0.034	fluid ounces	fl oz
L	litres	0.264	gallons	gal
m ³	metres cubed	35.315	cubic feet	ft ³
m ³	metres cubed	1.308	cubic yards	yd ³

TEMPERATURE (exact)

°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F
----	---------------------	-------------------	------------------------	----



These factors conform to the requirement of FHWA Order 5190.1A.

* SI is the symbol for the International System of Measurements

Table of Contents

1. Introduction	1
About This Manual.....	1
What VDM RoAD Is.....	1
What VDM RoAD Does	2
What VDM RoAD Does Not Do	3
How VDM RoAD Works.....	3
Installation of VDM RoAD.....	5
Backup Copy	5
2. Software Organization.....	6
Glossary Of Specialized Terms.....	6
The VDM RoAD Data Base.....	9
Data Screens.....	10
On-Line Help.....	12
A Map of the VDM RoAD Libraries and Screens	12
3. How to Use VDM RoAD	15
About the Runs Screen.....	15
How to Get to the Runs Screen	16
How to Make a New Run.....	17
How to Replace a Run.....	19
How to Make Long Runs (> 200 sec)	20
How to View an Animation.....	20
How to View a Single Predefined Plot.....	21
How to View Several Predefined Plots	21
How to Overlay Plots for Several Runs	22
How to Define Plots Interactively.....	23
How to View All Model Parameters and Inputs	24
How to Print a Data Set.....	24
How to Add a New Road to VDM RoAD.....	24
How to Make Wire-Frame Shapes for a Road	26
How to Go Directly to Any VDM RoAD Library	29
How to Lock Your Data	29
How to Delete Data Sets	31
How to Set Up Five or More Plots.....	32
How to Overlay Plots for Four or More Runs.....	33
How to Make Many Plots in Batch Mode.....	33
How to Set Up Multiple Runs	34
How to Make New Plot Descriptions.....	35
How to Change Plot Formatting.....	37
How to Assign the Default Text Editor.....	38
How to Make a New Vehicle Input.....	39

How to Modify an Existing Vehicle Input	40
How to Improve the Appearance of the VDM RoAD Screens	41
How to Install VDM RoAD in a New Directory.....	42
How to Import Data from Another Copy of VDM RoAD.....	45
How to Convert Units for Tabular Data.....	46
4. The Simulation Solver Programs	49
Overview of Program Operation.....	49
File Types.....	50
5. The Animator	53
Updates.....	53
Overview of Operation.....	53
Reference Frames.....	54
Files	55
Units	56
File Menu	56
Edit Menu.....	58
Animation Menu.....	60
Coordinates Menu	61
Options Menu.....	63
Help Menu.....	65
Testing New Shapes and Animator Data Sets.....	65
6. The Plotter.....	67
Updates.....	67
Overview of Operation.....	67
Batch and Interactive Operation.....	69
Zooming	69
File Menu	71
Edit Menu.....	74
Format Menu	75
Data Menu	82
View Menu.....	90
Windows Menu	91
Help Menu.....	93
Tool Bar.....	93
7. Design of VDM RoAD Data Screens	94
The Ribbon Bar	94
Data Links	98
Tabular Data.....	100
2D Tabular Data	102
File Menu	104
Edit Menu.....	105

Text Menu	107
Page Menu.....	108
Tools Menu.....	109

8. Alphabetical Library Screen Reference 111

Conventions in This Chapter.....	111
Aerodynamics.....	113
Animator Camera Setup.....	114
Animator Groups.....	117
Animator Reference Frames.....	120
Animator Shapes	123
Animator Wheels.....	125
Axles.....	127
Brake System.....	129
Brakes: Mechanical Properties.....	131
Calculator	133
Camber Angle.....	139
Cars.....	140
Computation Parameters	143
Generic 2D Table	146
Generic Group Data.....	147
Generic Table	149
Hitch	150
Input: Braking.....	152
Input: Closed-Loop Speed Control.....	154
Input: Highway Data (IHSDM).....	156
Input: Lateral Offset.....	158
Input: Wind.....	160
Library Editor.....	161
Load Dimensions.....	163
Plot Format.....	165
Plot Setup: Batch.....	168
Plot Setup: Single.....	170
Plot Transforms	173
Runs.....	175
Runs: Batch	181
Shock Absorbers.....	184
Springs: Car.....	186
Springs: Truck	188
Steering System: Truck	190
Steering System: Car	193
Steering System: Kinematics.....	194
Steering System: Rear/Front Wheel Ratio	196
Suspensions: Car	198
Suspensions: Solid Axle.....	201

Tires: Aligning Moment.....	202
Tires: Lateral Force	205
Tires: Longitudinal Force.....	207
Tires: Table-Lookup Model	209
Toe Angle.....	211
VDM RoAD Startup.....	213
Vehicle Unit: 1-Axle Trailer (Empty).....	217
Vehicle Unit: 2-Axle Tractor	219
Vehicle Unit: 2-Axle Trailer (Empty).....	222
Vehicle Unit: 2-Axle Truck or Bus (Empty).....	224
Vehicle Unit: 3-Axle Tractor	226
Vehicle: 2-Axle Truck or Bus (Loaded).....	228
Vehicle: Tractor-Semitrailer Combination.....	230
 Appendix A — ERD File Format	 233
The Header	233
The Data Section	237
 Appendix B — Plotter Files and Keywords	 239
The RunWinEP Program.....	239
PLT Batch Control Files.....	239
Plot Template Files.....	240
Plot Format Files	241
Preference File Format	244
 Appendix C — Animator Files and Keywords	 245
 Appendix D — Model Files and Keywords.....	 250
File Types.....	250
Viewing Keywords for the VDM RoAD Models	253
Keywords for the Truck Models	254
Keywords for the Car Model.....	277
 Appendix E — Truck Multibody Models	 291
Conventions.....	291
Overview of the Basic Factors Affecting Vehicle Behavior.....	291
Rigid Body Kinematics	294
Mass and Rotational Inertia.....	295
Equilibrium.....	296
Suspensions	296
Hitch Moments.....	299
Steering System.....	299
References	303

Appendix F — The Car Model	304
Multibody Ingredients	304
Nomenclature	304
Overview of Factors Affecting Vehicle Behavior.....	306
Rigid Body Kinematics	306
Masses and Inertias	308
Suspension Force Effects	308
References	310
Appendix G — Tire Models	311
Tire/Wheel Kinematics.....	312
Tire Forces and Moments.....	316
Sequence Of Calculations	318
Notes on Sign Conventions.....	319
References	319
Appendix H — Road Geometry.....	320
Right-Handed Axis Systems	320
The S-L-Z Coordinate System	321
Horizontal Geometry.....	322
Vertical Geometry	326
Appendix I — The Steer Controller.....	330
Optimal Control Theory	330
Application to Vehicle Control	332
References	337
Appendix J — Model Output Variables.....	338
Types of Output Variables	338
Creating Lists of Outputs	340
Car Model.....	341
Truck Models	343



1. Introduction

This chapter introduces the Vehicle Dynamic Models Roadway Analysis and Design (VDM RoAD) software. It provides an overview of what the software can do for you and how it works.

About This Manual

This manual is written to help you understand and use VDM RoAD. It covers several themes:

- The first two chapters provide an overview of VDM RoAD and its use.
- Chapter 3 provides a summary of how to perform common operations in VDM RoAD.
- Chapters 4 through 8 provide reference material for VDM RoAD features. Chapter 8, the largest in the manual, describes the VDM RoAD data screens in alphabetical order.
- The appendices provide technical details such as file formats, model parameter definitions, output variable names, and assorted modeling details.

Although we would not discourage you from reading this manual from beginning to end, we recognize that your time is valuable and that you might start by reading only what you think is absolutely necessary. **Please read Chapters 1 through 3.** These chapters provide an overview of how to use VDM RoAD. Other chapters can be read as needed or for reference.

What VDM RoAD Is

VDM RoAD is an easy-to-use integrated set of computer tools for simulating and analyzing the braking and handling behavior of vehicles on candidate road designs. VDM RoAD performs virtual tests, replacing the test vehicle with a mathematical computer model. It solves the equations of motion numerically to predict 3D motions of a vehicle in response to braking and steering inputs. The software runs on Intel PCs equipped with Windows 95 or Windows NT. It is self-contained, requiring no additional programs or tools to function.

VDM RoAD can be used to evaluate highway designs as they would be experienced by different vehicles traveling over a variety of speeds. VDM RoAD includes 12 example vehicles from the AASHTO Green Book (*A Policy of Geometric Design of Highways and Streets*, 1994), plus a car (Ford Taurus) and heavy truck whose properties were recently

measured to support research at National Highway Traffic Safety Administration (NHTSA) involving the validation of computer models of vehicle dynamics.

Simulation results can be viewed as wire-frame animations or as plots of output variables. These plots can be generated automatically for any combination of the many variables calculated during the simulation.

VDM RoAD was developed at The University of Michigan Transportation Research Institute (UMTRI) under a contract with the Federal Highway Administration (FHWA).

What VDM RoAD Does

VDM RoAD simulates a vehicle running down a road. The road geometry is described with an IHSDM interchange file as specified by the FHWA Geometrics Laboratory.

VDM RoAD uses detailed nonlinear tire models, nonlinear spring models, and includes the major kinematic and compliance effects in the suspensions and steering systems in cars and heavy trucks. The kinematical and dynamical equations are valid for full nonlinear 3D motions of rigid bodies.

VDM RoAD uses closed-loop controller algorithms to steer the vehicle along a specified path and to drive at a prescribed speed. Both speed and lateral position can be specified as constants or as variable functions of roadway station number. Braking inputs are specified as functions of time.

The simulations run much faster than you might expect for such detailed models. On Pentium-Pro computers, the programs run in real time or faster. In other words, a run simulating a 2 minute test will finish in less than 2 minutes.

The basic operation of VDM RoAD involves the following sequence:

1. Pick a road design of interest.
2. Pick a vehicle to run on the road.
3. Specify the target speed (target speed can vary as a function of station number).
4. Specify the target lateral position (target position can vary as a function of station number).
5. Run the simulation.
6. View the simulation results:
 - Animate the simulation output to get an overall view of the vehicle behavior.
 - Make plots to analyze behavior in more detail, looking at lateral tracking, lateral acceleration, friction use, etc.

With practice, you will be able to evaluate how roadway geometry limits the speed options available for different kinds of vehicles.

VDM RoAD uses mathematical models similar to those in the commercial programs TruckSim and CarSim. The models are based on research conducted over the past few decades at the University of Michigan. The TruckSim and CarSim software packages were further developed and commercialized by Mechanical Simulation Corporation (MSC), a private company in Ann Arbor, MI. MSC licenses, maintains, and supports TruckSim and CarSim.

The equations used in VDM RoAD are identical to those in the corresponding models in TruckSim and CarSim. In addition, the VDM RoAD versions include the capability of reading the IHSDM files. The VDM RoAD versions also have subtle modifications in the input requirements to make them more convenient for interacting with a road coordinate system based on station number and lateral position (S,L), in addition to the conventional X,Y coordinate required for vehicle models.

Note: The S-L coordinate system is described in Appendix H.

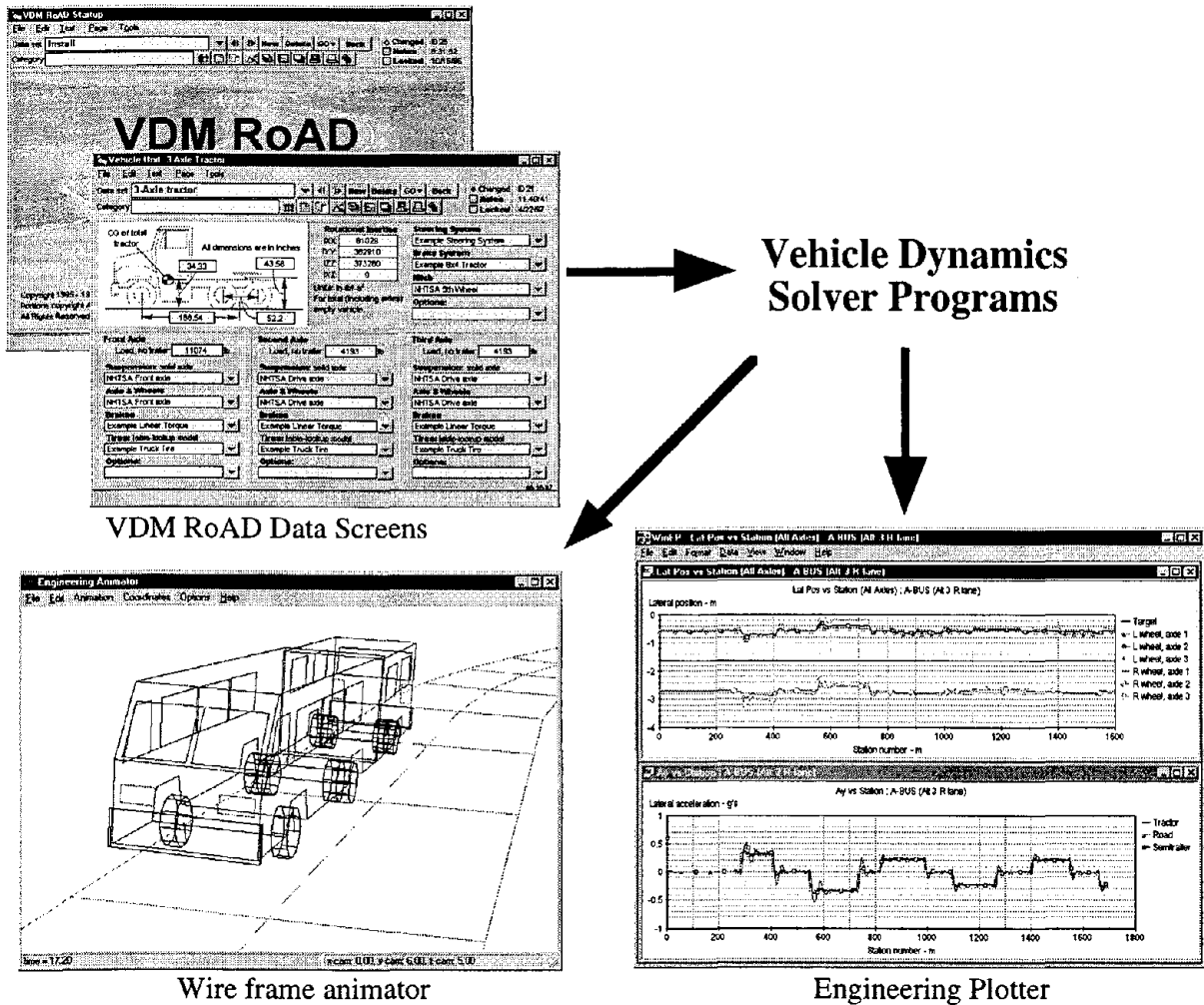
What VDM RoAD Does Not Do

Although the dynamics of the vehicle models in VDM RoAD are identical to those in TruckSim and CarSim, options that would not be of interest to a road designer have been removed. Output variables involving suspension and tire behavior are not written in the output files. Inputs that would be used to simulate proving ground tests (e.g., steering wheel angle as a function of time) are not supported. In order to automatically set initial conditions for the moving vehicle to match the target speed and road geometry, the option to set arbitrary initial conditions is not available.

How VDM RoAD Works

VDM RoAD includes a data base with vehicle data, simulation data, and control input data. The information in this data base is presented through a collection of data screens. Data screens are used for these purposes:

- to define the vehicle and maneuver,
- to control the simulation, and
- to present the output in animated or plotted format.



VDM RoAD Data Screens

Vehicle Dynamics Solver Programs

Wire frame animator

Engineering Plotter

Four parts of VDM RoAD.

VDM RoAD combines information from the data screens with vehicle dynamics programs to simulate the vehicle behavior. VDM RoAD also links the simulation results with animation and plotting programs.

VDM RoAD is made up of four tightly integrated types of software.

1. Data screens serve as your primary interface to VDM RoAD. They contain vehicle model parameters, control inputs, run settings, and links to IHSDM files with road geometric data. The data screens are part of a data base with many libraries of related data sets.
2. Simulation programs numerically solve equations of motion (i.e., mathematical vehicle models) to calculate output variables. The process of performing these calculations is called making a “simulation run.” These programs are applied automatically from the data screens when you click a “run button.”

3. An animator shows the computed vehicle motions using wire-frame shapes for the vehicle and road. You can view the simulated motions, zoom in and out with a virtual camera, and interactively move around the simulated vehicle to change your point of view.
4. The Windows Engineering Plotter (WinEP) creates plots of vehicle variables as functions of time or as cross plots of output variables. Use this tool to view any of the hundreds of variables computed by the simulation models. Plot any combination of variables, and overlay plots from different runs.

Installation of VDM RoAD

The VDM RoAD software is provided on a CD. It contains a folder INSTALL, with a file SETUP.EXE. Run SETUP to install the software. Select the full installation. When asked about installing program manager groups, click the YES button.

If you install VDM RoAD on drive C with the default name (C:\VDM_ROAD), that's it—you're done. However, if you install the software elsewhere you must correct some pathnames before making new simulation runs.

1. Start the software, taking you to the Startup screen.
2. Click the GO button (near the top of the window) to display a pull-down menu. In that menu, choose the item Input\Highway\Highway.tbk. See Figure 3.3 on page 25.
3. The yellow field shows a pathname such as:
C:\VDM_ROAD\ROADS\ALT3.IHM.
Click the button **Find IHSDM File...** to bring up the Windows file browser to locate the file on your hard disk. If you know the location, you can type it directly into the yellow field if you wish.
4. Use the arrow buttons at the top of the screen to browse through these data screens and correct the pathnames in all data sets in the library Highway.tbk.

Backup Copy

The CD also contains a backup copy of the software, in a folder named VDM_ROAD. It can be used to replace individual files if they get damaged.

Files copied directly from the CD drive are often designated by Windows as "Read Only." If you replace any files, check their properties and change them to have "Read and Write" capability.

2. Software Organization

This chapter provides an overview of how the various parts in VDM RoAD work together. Details about each part are provided in later chapters.

Glossary Of Specialized Terms

A number of terms are assumed to have certain specific meanings within the context of this manual. Some of the most common ones are listed here with short definitions. The concepts are developed more fully later in this chapter.

Animator — a program in VDM RoAD that is used for animating wire-frame representations of vehicles.

CarSim™ — a commercial software package similar in design to VDM RoAD that is used by vehicle engineers to study the dynamics of automobiles. The mathematical car model in VDM RoAD is essentially identical to the one in CarSim. The CarSim version has additional output variables of interest to mechanical engineers, and lacks the integration with the IHSDM data files.

Data screen — a view of a data set contained in the VDM RoAD data base. A data screen is where you enter the parameters that define your target simulation. It is a data set as viewed through a graphical user interface.

Data set — a collection of parameter values and related information, organized for display on a VDM RoAD screen. Data sets are contained in library files and edited in data screens.

Echo file — a text file, produced by a solver program, that lists every input parameter. The echo file documents the conditions that were simulated in a run. All echo files made by VDM RoAD solver programs are written in the PARSEFILE format. Echo files can also be read by the solver programs as inputs for future runs.

ERD file — a file stored in a standard format that supports automated plotting and post-processing. Output files produced by VDM RoAD solver programs are written in ERD format. The plotter and animator programs read ERD files. ERD stands for Engineering Research Division, a group at UMTRI.

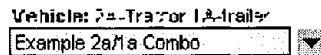
Go — in the context of VDM RoAD, going to a different data set, screen, or library means having VDM RoAD change the screen view to display a different data set. The new data set could be from the same library or a different one.

GRID — a utility program (*Grid.exe*) provided with VDM RoAD to create wire-frame representations of roads for viewing with the animator tool.

IHSDM — interactive highway safety design model. This is a multi-year project under development by FHWA to apply new computer technologies toward the process of highway design. VDM RoAD is potentially one part of the IHSDM. IHSDM is also used to indicate a text file containing road geometric data.

Library — a file containing one or more data sets of the same type, plus a standard graphic interface for viewing those data sets. All VDM RoAD library files have the extension TBK.

Link — a connection from one data set to another, indicated with a blue field and an adjacent triangle button with a triangle. For example, here is a link to a vehicle data set.



The adjacent button has a pull-down menu that can be used to rapidly change the link to a different data set. The menu can also be used to “follow the link” and go to the data set.

MSC — Mechanical Simulation Corporation. The company that licenses, maintains, and supports CarSim™, TruckSim™, the animator, and WinEP.

Numerical integration — a computation method used to solve the differential equations that define a mathematical model of vehicle dynamics. Output variables produced by the models in VDM RoAD are based on internal model variables that are calculated over time using numerical integration.

Parameter — a value that you can change from a VDM RoAD data screen. Parameters are properties of the VDM RoAD models that are constant during a computer run. Variables, on the other hand, are computed by the simulation programs and cannot be viewed until after a run is made.

PARSFILE (parameter file) — a keyword-based text file used to communicate between various VDM RoAD programs.

QLPad — a public-domain text editor included with VDM RoAD for editing and viewing text files.

Ribbon bar — the collection of buttons and user controls displayed at the top of nearly every VDM RoAD data screen.

Run — Shorthand for “run a simulated test” (e.g., “make a run”). Also used to refer to the “outcome of a simulated test run” (e.g., “look at a run”).


Screen — shorthand for “screen layout associated with a library.” This manual has many references to the act of going from one screen to another. Changing the screen simply changes the view to show data in a different library.

SGUI — Simulation Graphical User Interface. The architecture of VDM RoAD (data screens, plotter, simulation programs, animator) has been used for other software packages. The generic architecture, created at UMTRI and further developed by MSC, is called the SGUI.

Solver program — a program that numerically solves the equations of motion of a vehicle model to simulate a test. The VDM RoAD solver programs are customized with

the equations of motion for specific vehicle models. They handle all required calculations and input/output.

TBK files — binary files that contain the VDM RoAD libraries. In addition, some of the code needed to make the VDM RoAD buttons work is stored in TBK files. All necessary TBK files are provided in the VDM RoAD installation, and they are managed automatically by VDM RoAD. TBK files are native to the ToolBook software.

Triangle button — a button with a triangle icon () , usually located adjacent to a field. Press the button to display a pull-down menu.

ToolBook™ — a “Visual Authoring System” from Asymetrix, Inc. The graphical interface and the data base portion of VDM RoAD are managed by code that runs within ToolBook. Also, the integration between the data base and VDM RoAD programs is provided by ToolBook. The running of ToolBook code and management of ToolBook libraries is done by the ToolBook runtime package, which is included with VDM RoAD.

TruckSim™ — a commercial software package similar in design to VDM RoAD that is used by vehicle engineers to study the dynamics of heavy trucks. The mathematical truck models in VDM RoAD are essentially identical to those in TruckSim. The TruckSim models have additional output variables of interest to mechanical engineers, and lack the integration with the IHSDM data files.

UMTRI — The University of Michigan Transportation Research Institute. VDM RoAD was created at UMTRI.

Unit — in the context of a vehicle, a unit is a tractor, trailer, truck, or bus. A vehicle unit is a frame with the parts that are permanently attached (body, axles, hitch, etc.). Trucks and buses are called single-unit vehicles, and vehicles with two or more units are called combination vehicles.

Variable — an output quantity computed by the VDM RoAD vehicle models. In contrast to a parameter, a variable can change with time in a way that must be simulated using a detailed vehicle dynamics model. Output variables can be plotted with WinEP, the VDM RoAD plotter. Motion variables can be read by the animator to create animation of the simulated vehicle motions.

VDM RoAD — an integrated software package that includes solver programs, a data base, a plotter, an animator, and a graphical user interface that includes the ToolBook runtime package.

VDM RoAD data base — the collection of all data libraries within VDM RoAD.

WinEP — the Window version of an Engineering Plotter developed at UMTRI for making X-Y plots from ERD files.

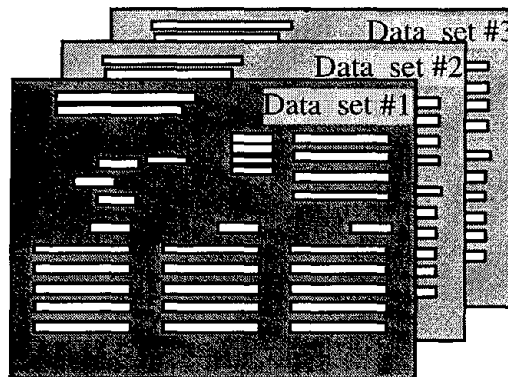
The VDM RoAD Data Base

Most of your interactions with VDM RoAD will involve preparing data sets for simulation runs, or viewing outputs of simulation runs. In either case, you will be dealing extensively with the VDM RoAD data base.

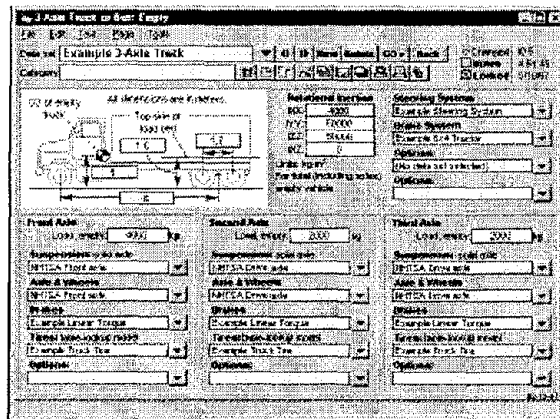
Throughout this manual, the data base is considered at four levels. Starting with what you see on the screen, the levels are:

1. **Data screens.** Any screen display in VDM RoAD that has editable fields or other forms of user settings is called a data screen.
2. **Data sets.** A data set is the information you provide and can edit in a data screen, minus the screen itself. The data screen is a view of a data set as seen through the SGUI (the VDM RoAD graphical interface).
3. **Libraries.** A library is a collection of one or more data sets, plus the information needed to provide a view in the user interface. The next figure shows the contents of a library: multiple data sets, plus a graphical view of one data set at a time.
4. **The VDM RoAD data base.** VDM RoAD includes about 50 libraries.

Individual data sets in the library



Data Screen:
A view of one data set though the graphical user interface



Contents of a VDM RoAD Library


VDM RoAD automatically handles the transfer of information between the libraries and solver programs. Thus, you do not have to know where the libraries are or what they contain. However, it is important to understand the distinction between data screens, data sets, and libraries. Specifically, you should understand that:

- Each VDM RoAD screen shows one data set from within one library.
- Changes made on a data screen affect the one data set, but all other data sets in the library are unaffected.
- **Data sets** are used in a simulation run only if they are linked (possibly in a chain of links) to the data set represented by the **Runs** screen when the run button was clicked.
- **Links** between data sets can be made and broken as you see fit, to rapidly swap vehicles, components, inputs, etc.
- When a data set is changed, simulation results will be affected if the simulation inputs include data sets that are linked to it directly or through a series of links.

Data Screens

Common elements in a data screen will be introduced for the example screen shown in Figure 2.1.

Each data screen has three kinds of elements that you use:

- **Yellow fields.** These contain data that you can edit directly. For example, to change wheelbase, find the yellow field ③ with the current wheelbase value, click on the field, and change the value using the mouse and keyboard.
- **Buttons.** All of the library screens include buttons at the top to quickly navigate through the data sets in the library, and to go to other libraries and programs in VDM RoAD (see the buttons near ②). In addition, some of the libraries have special buttons for functions such as running a simulation, running an animation, etc.
- **Pull-down menus.** Some of the buttons are marked with a triangle , indicating that pressing them with the left mouse button brings up a pull-down menu.

The yellow field in the upper-left corner of the screen is the title of the data set: **AASHTO 3-Axle Tractor** ①. The title is simply text that you choose to identify this data set in the library. It has no significance to the simulation.

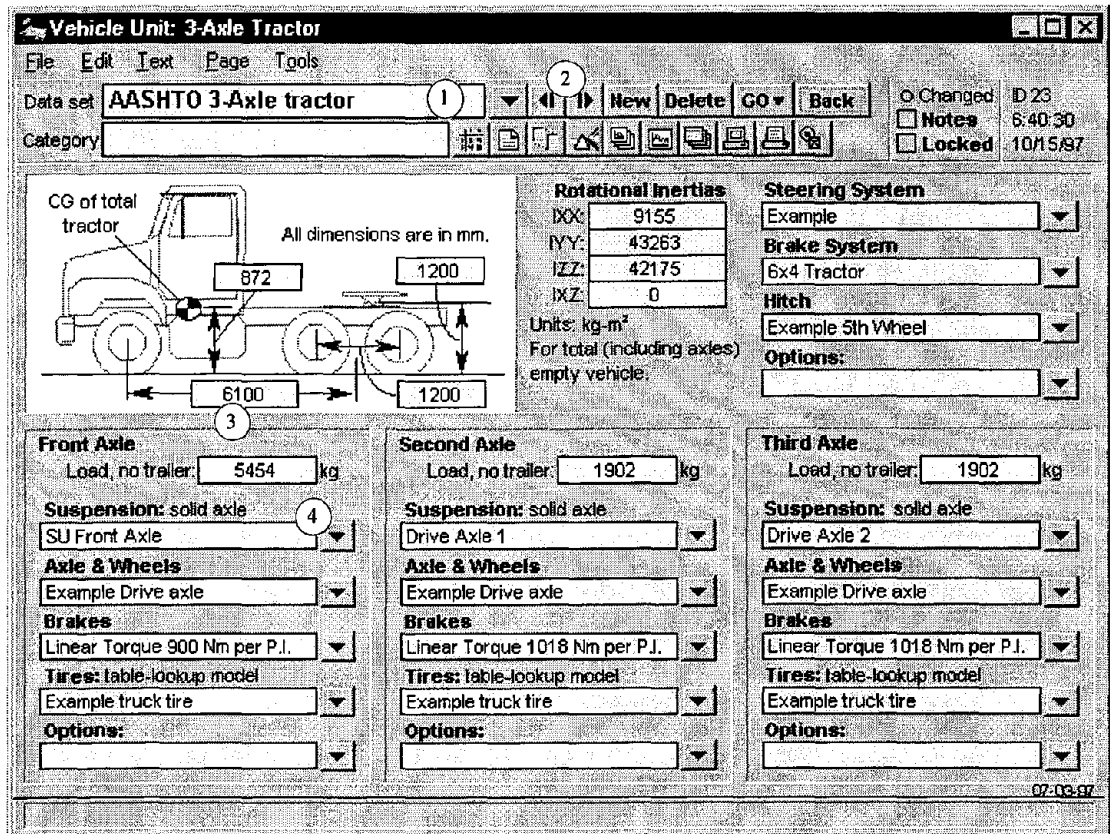
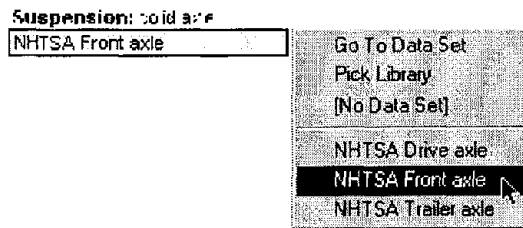


Figure 2.1. Example data screen.

Blue fields with adjacent triangle buttons are common in VDM RoAD. Each represents a link to another data set. The name of the data set is shown in the blue field, and the triangle button is used to display a pull-down menu. For example, click the triangle button ④ to view the following menu.



The menu can be used to select a different data set in the linked library (choose the data set of interest from the list displayed at the bottom of the menu). It can also be used to go to the linked data set (choose the menu option to **Go To Data Set**).

After using **Go To Data Set** to go to a different library, you can return to the previous screen using the **Back** button located in the upper-right corner of each data screen.



The standard elements in a data screen, such as the ribbon bar at the top, and the blue links, are described in more detail in Chapter 7.

On-Line Help

Every VDM RoAD screen includes a status bar at the bottom of the screen. Move the cursor over an object to read a one-line description of its function in the status bar.

A Map of the VDM RoAD Libraries and Screens

Nearly all control of VDM RoAD is made through the VDM RoAD data screens. Each screen design is associated with a separate library file. VDM RoAD includes the following types of libraries:

- Vehicle parameters
- Inputs for simulation runs
- Run setups including simulation and post-processing control
- Plot setups
- Animation setups
- Batch controls

The top level of VDM RoAD is shown graphically in Figure 2.2.

You normally start VDM RoAD at the **Startup** screen. Clicking the **Start** button on that screen takes you to the **Runs** screen. The **Runs** screen is the window to a library of many run descriptions, each of which has many links to vehicles, inputs, and various settings.

The **Runs** screen also has three buttons for:

- Running new simulations
- Using the plotter to view simulation results
- Using the animator to view simulated motions

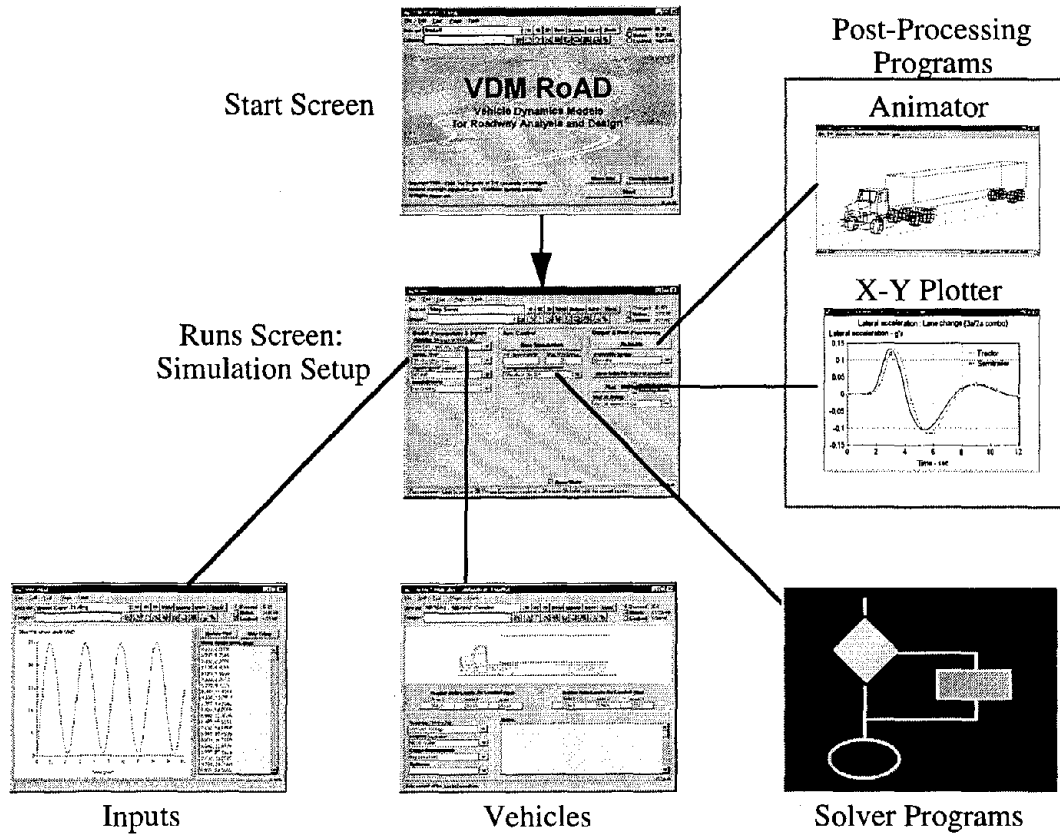


Figure 2.2. Role of Runs screen in VDM RoAD.

Figure 2.3 shows a more detailed map of the various libraries (screens). For example, from the **Runs** screen, you can go “down” to the various inputs (path, brake, steer, and speed), the vehicle descriptions, the plot and animation setup, and the parameters that control the numerical solution methods used to make a simulation.

Notes: In order to create the detailed map at a reasonable size, the names were shortened from the full screen titles.

The map shows some of the libraries for heavy trucks. Libraries for cars are not shown, but follow similar relationships

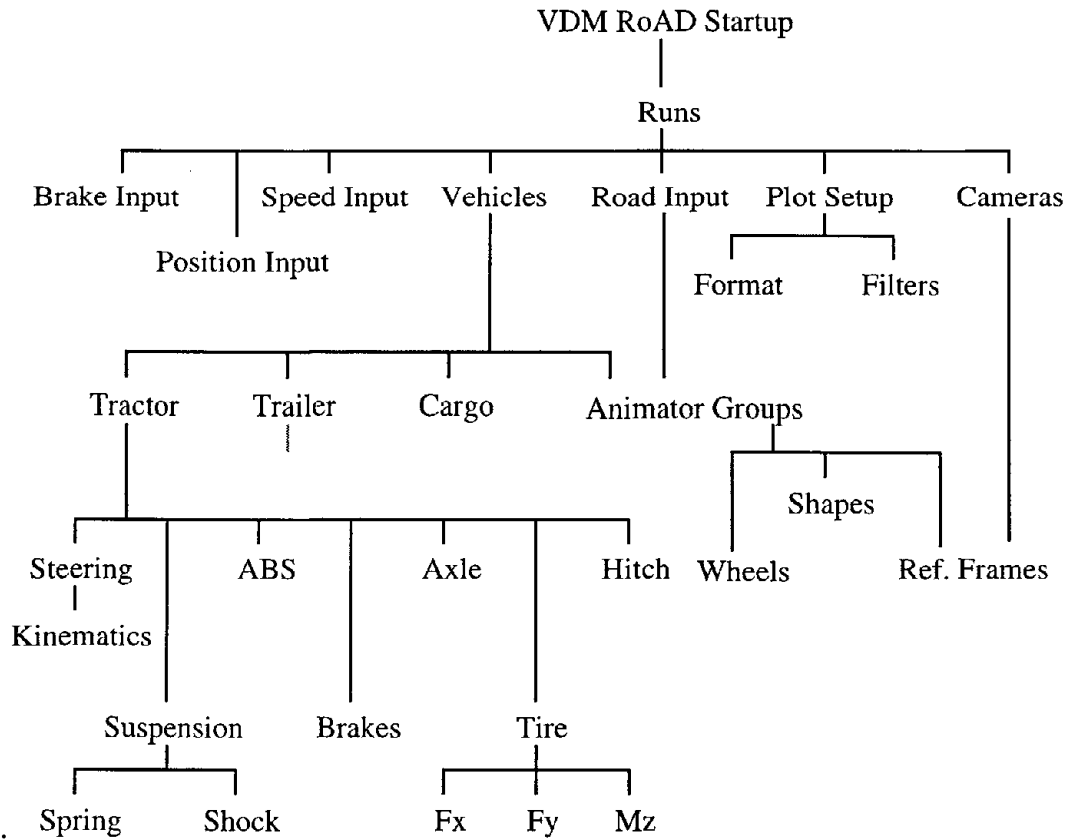
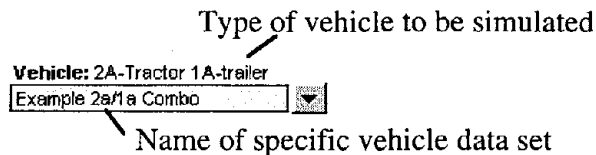


Figure 2.3. Map of the VDM RoAD libraries.

In the map, data sets from different libraries are connected with links. For example, here is a link between the **Runs** library and a vehicle library:



In general, libraries shown near the top of the map are more systems-oriented, and libraries shown near the bottom are more component-oriented.

You can traverse the map, starting at the top, by following the links down. For example, starting at a **Runs** screen, you can follow a link “down” to a combination tractor-semitrailer vehicle. From there, follow a link down to a tractor library. From the tractor library, you can go down to a steering system library, a suspension library, etc.

Travel back up by clicking the **Back** button on each data screen.



3. How to Use VDM RoAD

This chapter contains “how to” lists for accomplishing common tasks in VDM RoAD.

About the Runs Screen

The **Runs** screen is central to the operation of VDM RoAD. The diagram in Figure 2.2 on page 13 shows that it provides access to vehicle data, input data, post-processing programs, and solver programs.

Figure 3.1 shows the **Runs** screen for a typical data set. The circled numbers will be used in the next few sections to describe common actions.

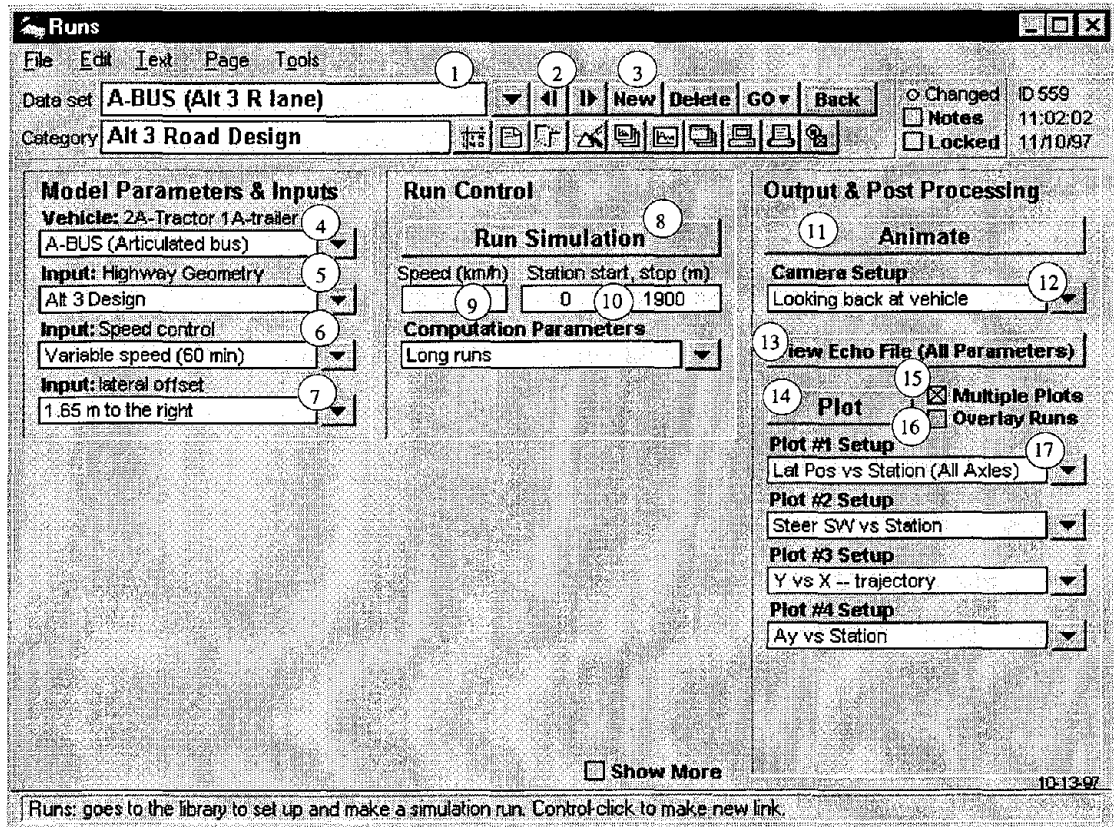


Figure 3.1. The Runs screen.

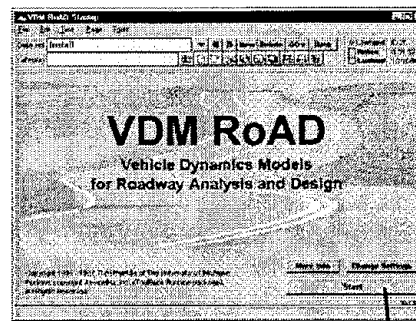
How to Get to the Runs Screen

Most operations begin with the **Runs** screen.

When Starting

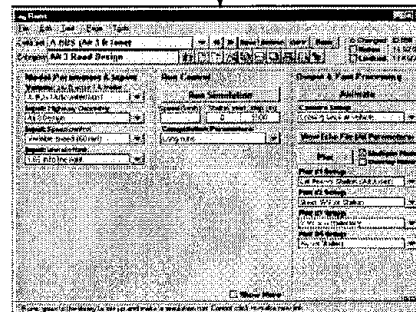
1. Start VDM RoAD to get to the **Startup** screen.
 - Use the Windows Start menu, or
 - open the file Startup.tbk

Start Screen



Click Here


Runs Screen:
Simulation Setup

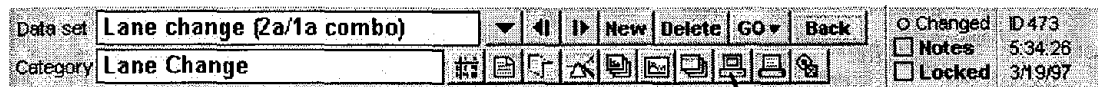


2. Click the **Start** button in the lower-right corner of the screen.

After Starting

All VDM RoAD screens include a standard ribbon bar with navigation buttons. One of these will take you to the **Runs** screen.

Click the **Runs** button in the ribbon bar  or use the **Runs** command from the **Tools** menu.



Click Here

From the Plotter or Animator

The VDM RoAD plotter and animator are independent programs. When you launch one of them by clicking a button on the **Runs** screen, the **Runs** screen is still there. However, it might be covered by the windows of the other programs, especially if you are not using a large computer monitor.

The recommended way to return to the **Runs** screen from the plotter or animator is to exit the program.

1. Click the X box in the upper-right corner of the WinEP or Animator screen:



How to Make a New Run

Use this method to make a new run in VDM RoAD. Existing runs are left intact and their output can still be viewed with the plotter and animator.

Start from the **Runs** screen (see Figure 3.1 on page 15).

1. Find an existing data set that is close to what you want using the navigation buttons (2). In particular, look at section of the screen labeled **Model Parameters and Inputs**.
 - What kind of vehicle is selected in the vehicle link (4)? VDM RoAD includes 12 examples sized to match examples in the 1994 AASHTO Green Book, plus a few others.
 - How is the speed controlled? Is a constant speed specified (9), or is there a link to a variable-speed input (6)?
 - Is there a link to a braking input?
2. Click the **New** button (3) in the ribbon bar.
3. Inspect and edit the simulation inputs.
 - Select the vehicle of interest from the linked vehicle library (4).
 - Select a road input from the linked library (5).
 - Select the vehicle speed in one of two ways:
 - a. enter a speed into the yellow field (9), or
 - b. select a speed profile from the linked library (6).

Note: If you enter a speed in the yellow field (9), the speed controller will attempt to maintain that value as long as the brakes are not applied. If you want to specify a target speed that varies during the run, look for an example run in which one of the input links (e.g., (6)) is connected to a speed input.

- Select a lateral position input (as a function of station number) (7).
- Enter the starting and stopping station numbers (10).

Note: The direction of travel is determined by the starting and stopping station numbers.

4. Click the **Run Simulation** button (8).

A console application window (DOS-style) appears while the simulation runs. A bar-graph at the bottom of the window indicates the progress. When the simulation finishes, the DOS window disappears and you are returned to the **Runs** screen.

```

D:\CODE\TRUCKSIM.402\C.IN\C2A1A_3D\C2A1A_3D\C2A1A_3D.EXE
2-axle tractor with 1-axle trailer
TruckSim braking and handling model (Version 4.0.2).
IN-LB Units
Generated by AutoSim June 27, 1997

Simfile found: batch operation.
Input file with parameter values:      C:\TRUCKSIM.IN\RUNS\492.LPI
Echo file with initial conditions:     C:\TRUCKSIM.IN\RUNS\492.LPO
Echo file with final conditions:       C:\TRUCKSIM.IN\RUNS\492.LPF
Output file with time-history data:    C:\TRUCKSIM.IN\RUNS\492.ERD
Output file with log of nested inputs: C:\TRUCKSIM.IN\RUNS\492.LOG
.....

Progress (percent complete):
0                               50                               100
=====

```

Console display when solver program is running.

Notes: If the vehicle comes to a complete stop, or rolls over, the normal settings cause the program to quit after updating all output files.

The Progress bar is not accurate because it is based on elapsed time, rather than station number. It is taken from the TruckSim and

CarSim versions, which generally stop a simulation when a specified time has been reached. In VDM RoAD, the simulation stops when the specified station number has been reached. Because the speed is permitted to be highly variable, it is not easy to determine the amount of time left.

There are a few occasions when the VDM RoAD solver program will not run, or will quit prematurely. Among them are:

1. The output file cannot be written if it is in use by the plotter or the animator (this can happen if you are replacing a run, as described in the next section).
2. In rare circumstances, a bad input parameter will cause the program to quit before it can create any output files. However, in most cases, the program will generate the output files even if it quits suddenly.

How to Replace a Run

Use this method to replace an existing run. Start from the **Runs** screen (see Figure 3.1 on page 15).

1. Find the existing data set in the **Runs** library that you want to replace. Use the navigation buttons ② next to the Data Set (title) box ①.
2. Inspect and edit the simulation inputs.
 - Select the vehicle of interest from the linked vehicle library ④.
 - Select a road input from the linked library ⑤.
 - Select the vehicle speed in one of two ways:
 - a. enter a speed into the yellow field ⑨, or
 - b. select a speed profile from the linked library ⑥.
 - Select a lateral position input (as a function of station number) ⑦.
 - Enter the starting and stopping station numbers ⑩.
3. Click the **Run Simulation** button ⑦. A run is made, as described in the previous section, except that in this case the previous run (selected in step 1) is over-written.

How to Make Long Runs (> 200 sec)

The solver programs in VDM RoAD stop when the indicated range of station has been reached, or when the simulation time hits 200 seconds. For runs longer than 200 seconds, it is necessary to override the default stop time.

Start from the **Runs** screen (see Figure 3.1 on page 15).

1. Set up the run, as described earlier, setting all links involving the vehicle, road, speed, etc.
2. Click the button **Show More** at the bottom of the **Runs** screen (② in Figure 3.2) to reveal additional information.
3. Enter the text `stopt 500` into the yellow Misc. field (①). This will change the stop time to 500 seconds. Use a number other than 500 if you wish.
4. Make the run as before, by clicking the button **Run Simulation**.

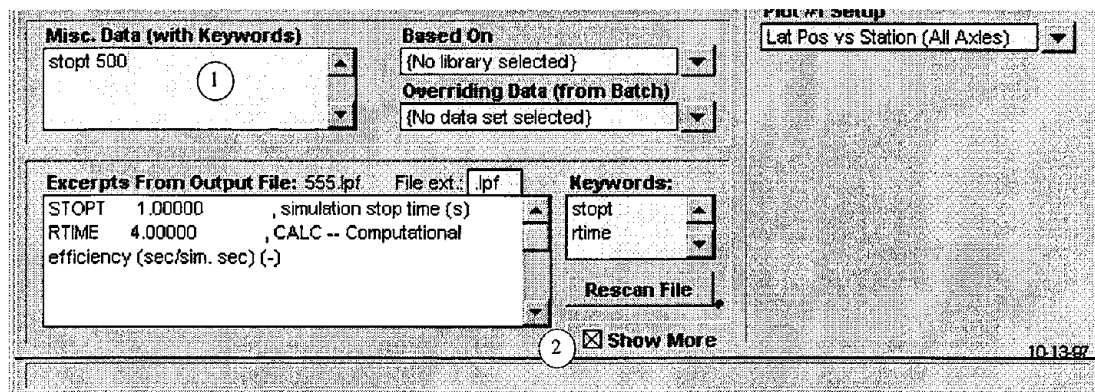


Figure 3.2. Bottom of Runs screen with additional options shown.

How to View an Animation

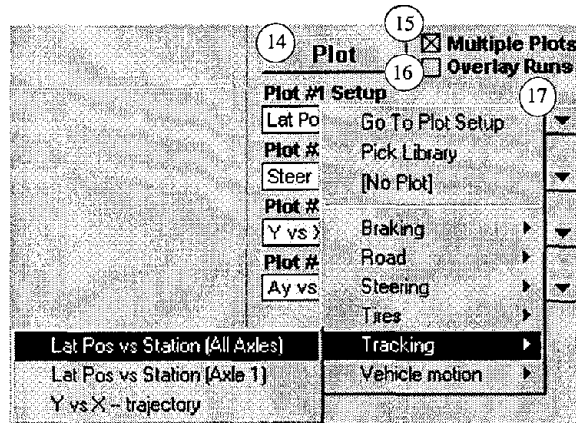
Start from the **Runs** screen (see Figure 3.1 on page 15). The simulation run must have already been made in order to view an animation.

1. Select a camera point of view using the camera link (⑫).
2. Click the **Animate** button (⑪).
3. If there is no motion, use the menu item **Start from Beginning (Ctrl-S)**, under the **Animation** menu in the Animator screen.
4. After viewing the animation, exit the animator by clicking the in the upper-right corner of the screen.

How to View a Single Predefined Plot

Start from the **Runs** screen (see Figure 3.1 on page 15). The simulation run must have already been made in order to view a plot.

1. To avoid making multiple plots, un-check the box **Multiple Plots** (15).
2. To avoid overlaying plots from different runs, un-check the box **Overlay Runs** (16).
3. Use the pull-down menu next to the blue field labeled **Plot Setup** (17) and select the plot of interest.

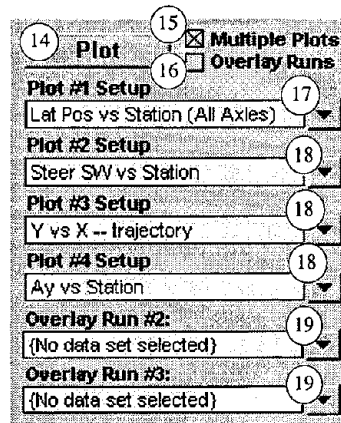


1. Click the **Plot** button (14).
2. After viewing the plot, exit the plotter by clicking the in the upper-right corner of the screen.

How to View Several Predefined Plots

Start from the **Runs** screen (see Figure 3.1 on page 15). The simulation run must have already been made in order to view a plot.

1. Check the box **Multiple Plots** (15). This reveals three more links to plotter data sets (18):



2. Select up to four plot setups, using the different links (17) and (18).
3. Click the **Plot** button (14).

Note: Multiple plots can be viewed in WinEP in several modes. The WinEP **Windows** menu can be used to locate any existing plot window. All of the plots can be viewed simultaneously by using the vertical or horizontal tiling options from the **Windows** menu.

4. After viewing the plots, exit the plotter by clicking the in the upper-right corner of the screen or by using the **File** menu or by typing Ctrl-Q.

How to Overlay Plots for Several Runs

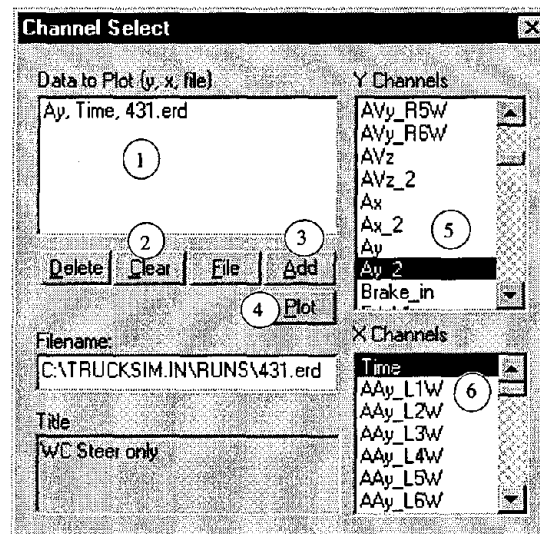
Start from the **Runs** screen (see Figure 3.1 on page 15). Several simulation runs must have already been made in order to view plots involving them.

1. Check the box **Overlay Runs** (16). This reveals two links labeled **Overlay Run #2** and **Overlay Run #3** (19).
2. Select another run of interest from the pull-down menu next to **Overlay Run #2** (19).
3. If you want to overlay three runs, link to another run of interest using the link labeled **Overlay Run #3**. On the other hand, if you only want to compare two runs, choose {No Data Set} for **Overlay Run #3**.
4. Select up to four predefined plots using links (17) and (18).
5. Click the **Plot** button (14).
6. After viewing the plots, exit the plotter by clicking the in the upper-right corner of the screen or by using the **File** menu or by typing Ctrl-Q.

How to Define Plots Interactively

Start from the **Runs** screen (see Figure 3.1 on page 15). The simulation run must have already been made in order to view a plot.

1. Start the plotter program by clicking the **Plot** button (14).
2. From within the plotter, select the menu item **Define New Plot...** from the **Data** menu (or type the keyboard command Ctrl-N). This brings up a window that lists all of the variables in the simulation output file.



3. Select X and Y variable names from the X and Y lists (5) and (6).

Note: You can select the names by clicking on them with the mouse. You can also click in a list to make it active and then type the first letter of the name or use the ↑ and ↓ keys to change the selection.

4. Add the selected variables to the list of data to plot (1) by using the **Add** button (3).

Note: You can also add a pair of variables by double-clicking on the name in either the X (5) or Y (6) lists.

4. Make the plot by clicking the **Plot** button (4).
5. To make another plot, repeat steps 2 through 4. Clear the list of the data to plot with the **Clear** button (2).

See Chapter 6 for more information on using the plotter interactively.

How to View All Model Parameters and Inputs

Start from the **Runs** screen (see Figure 3.1 on page 15). The simulation run must have already been made in order to view a list of all parameters.

1. Click the button **View Echo File (All Parameters)** (12).
2. This loads a text file into the default text editor. The text file is created whenever a simulation program is run, and lists all model parameter values, definitions, and units.

Notes: VDM RoAD is shipped with a built-in text editor. If you want to make a different text editor your default, see the section “How to Assign the Default Text Editor” on page 38.

3. Return to the **Runs** screen in VDM RoAD with the text editor still active by pressing **Alt-Tab**. Or, exit the text editor to return to the **Runs** screen.

How to Print a Data Set

You can print any screen in VDM RoAD. Copies of the data screens are sometimes helpful for showing model parameters as they appear when using the software. However, a more efficient method is to view all of the parameters in a text editor, as described in the previous section, and then print them from the text editor.

Start from the **Runs** screen (see Figure 3.1 on page 15). The simulation run must have already been made in order to view a list of all parameters.

1. Click the button **View Echo File (All Parameters)** (13).
2. Select **Print** from the **File** menu of the text editor.
3. Exit the text editor.

How to Add a New Road to VDM RoAD

Road descriptions originate (from the perspective of VDM RoAD) from the design data stored in an IHSDM interchange file, as defined by the A/E Group at the FHWA Geometrics Laboratory.

The road geometric data are used two ways in VDM RoAD:

1. to simulate the 3D surface over which a vehicle travels, and
2. to generate a wire-frame view of the road geometry for viewing outputs.

The simulation programs in VDM RoAD read the IHSDM files directly, and use the information to generate the S-L-Z coordinate system and also a 3D ground surface in the X-Y-Z coordinate system.

The animator does not read IHSDM files. It is generic, animating whatever wire-frame shapes are provided to it as inputs. For example, the animator uses the same generic equations to display vehicle shapes and road shapes.

To run the simulated vehicles over a new road as defined by an IHSDM file, the IHSDM file must be identified within VDM RoAD. The process is as follows:

1. From within VDM RoAD, click the **GO** button at the top of any data screen to display a pull-down menu. Select the library input\highway\highway.tbk. The screen display for the IHSDM library will be displayed, similar to the one shown in Figure 3.3.

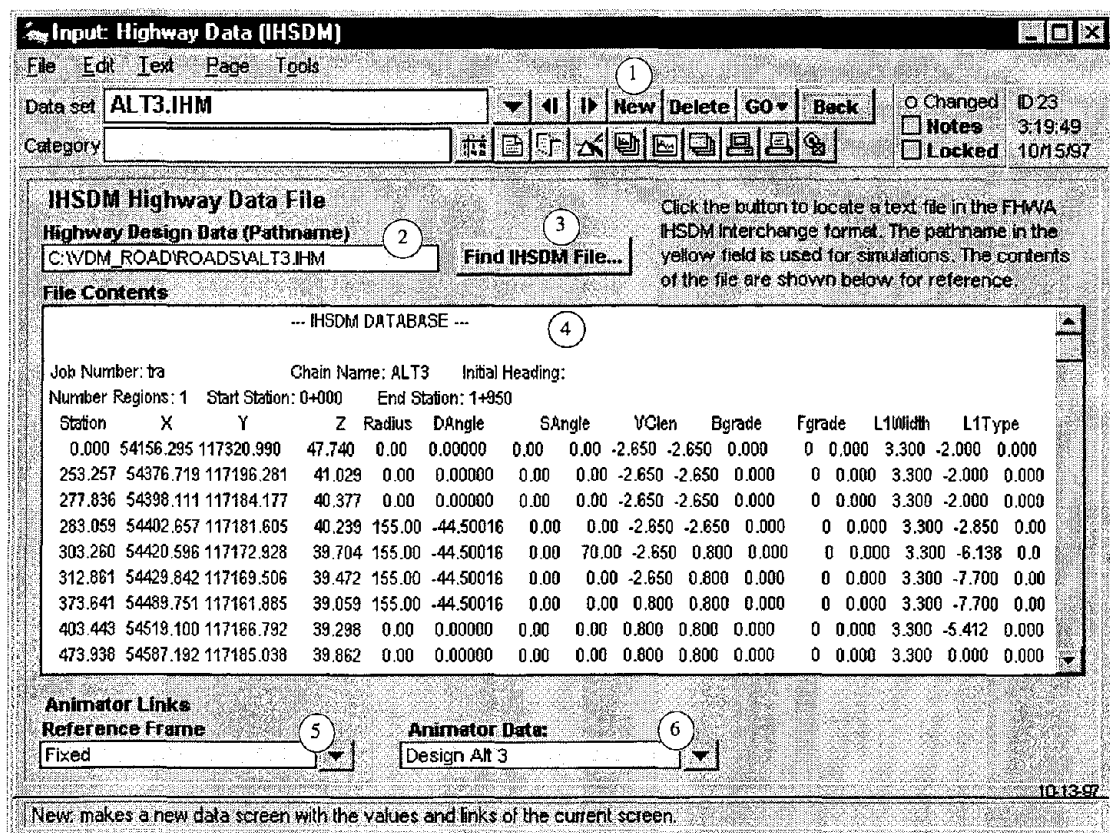


Figure 3.3. The IHSDM data screen.

2. Click the **New** button (1) to create a new data set.
3. Click the button **Find IHSDM File...** (3). The Windows file browser will appear. Use it to locate the text file with road data in the IHSDM interchange format. The full pathname will then be displayed in the yellow field (2). The contents of the file will be displayed in the large white field (4).
4. The Reference Frame link in the lower-left part of the screen (5) should be to the frame called **Fixed**.

5. The Animator Data link ⑥ should be made to a wire-frame shape or group of wire-frame shapes based on the IHSDM file. The method for creating the wire frame is described in the next section.

How to Make Wire-Frame Shapes for a Road

In order to view a vehicle traveling over a road based on an IHSDM data file, wire-frame shapes have to be created for the road. The process of creating a wire-frame shape for a road section is one of the few that are not done from within the VDM RoAD data base. However, it takes only a few minutes and is required only when a new road design is added to the data base.

Making the wire-frame shape

A utility program `Grid.exe` is used. Example input files are provided, along with a copy of `Grid.exe`, in the `Vdm_road\Roads` directory. The process is as follows:

1. Prepare an input file that specifies the grid spacing, the starting station, the ending station, and the lane width. An example input file is shown in
2. Listing 3.1. This file is named `In.par` and is included in the `Vdm_road\Roads` directory.
3. Move the `Grid.exe` file, the input file (e.g.,
4. Listing 3.1), and the IHSDM file into the same folder (e.g., `Roads`).
5. Run the Grid program. When prompted, enter the name of the IHSDM file (e.g., `Alt2.ihm`), the name of the input file (e.g., `In.par`), and the name of an output file (e.g., for lane1, use the name `lane1.txt`).
6. Open the output file (e.g., `lane1.txt`) with a text editor such as `QLPad` (`QLPad.exe` is in the `Vdm_road\Programs` directory).
7. Select the part of the file between the keywords `GRID_TABLE` and `ENDTABLE`, and copy the selection to the clipboard (see the bold text in Listing 3.2). These are the coordinates for the points in a wire-frame shape.
8. In VDM RoAD, use the **GO** menu to go to the library: `animate\shapes\shapes.tbk`.
9. Click the **New** button.
10. Clear all of the fields in the screen, and paste the coordinates from the clipboard into the field named **Coordinates**. See Figure 3.4.
11. Set the color, if desired.

Listing 3.1. Example input file for the GRID program.

```

PARSFILE
* This is an input to the GRID program, to make animation shape files
  from IHM files.

GRID_WIDTH      3.30000    , width of grid on outer-most lanes (m)
GRID_STEP      10.0000    , station step for road lane grid data (m)
GRID_START     1000      , starting station for road lane grid data (m)
GRID_END       1900      , ending station for road lane grid data (m)
GRID_LANE       2        , lane number (from Left)

END

```

Listing 3.2. Example output file from the GRID program.

```

PARSFILE
* Echo file created by:
* Intermediate tool for writing animator shapes (October 1997).
* Copyright 1989-1997 The Regents of The University of Michigan. All
  rights reserved.

TITLE
* Run was made 16:35 on Oct 15, 1997

* PARAMETER VALUES

GRID_WIDTH      3.30000    , width of grid on outer-most lanes (m)
GRID_STEP      10.0000    , station step for road lane grid data (m)
GRID_START     1000.00    , starting station for road lane grid data
                        (m)
GRID_END       1900.00    , ending station for road lane grid data (m)
GRID_DIVS       0        , number of lateral divisions per lane
GRID_LANE       0        , lane number (from Left)
* Animator Lane Grid Data: X, Y, Z
GRID_TABLE
  55031.336 117034.582  37.518
  55032.678 117031.567  37.650
  55041.813 117035.634  37.535
  ...
  55769.927 116786.505  43.801
  55764.751 116795.062  44.144
  55769.927 116786.505  43.801
ENDTABLE

END

```

After the wire-frame shape is created, it can be linked to the road data using the link in the IHSDM screen (see link ⑥ in Figure 3.3). If several shapes are to be grouped together, create a new data set in the animator groups library (animate\groups\groups.tbk) and link to all of the wire-frame shapes. If several shapes are

grouped together, then link to the group in the IHSDM screen rather a data set for a single shape.

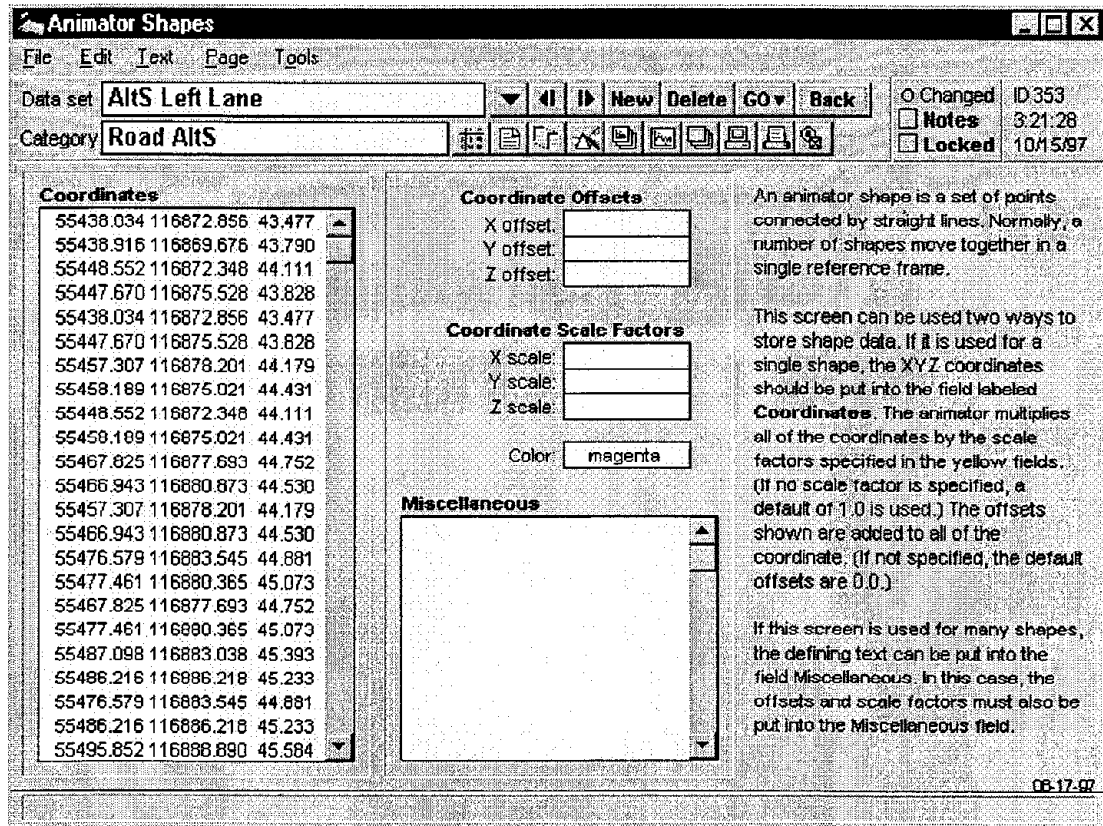


Figure 3.4. The animator shapes screen (with road grid data).

To better understand how road geometry is represented in the VDM RoAD data base, it is helpful to study the example already installed.

Wire-Frame Representation of the Example Road

An example of how the shape files are used can be seen with the examples based on the Alt 3 file, a two-lane road covering a little under 1900 meters. That road is represented with a group (in the animator groups library) of four shapes. The shapes are wire-frame representations of part of one lane. The four shapes are:

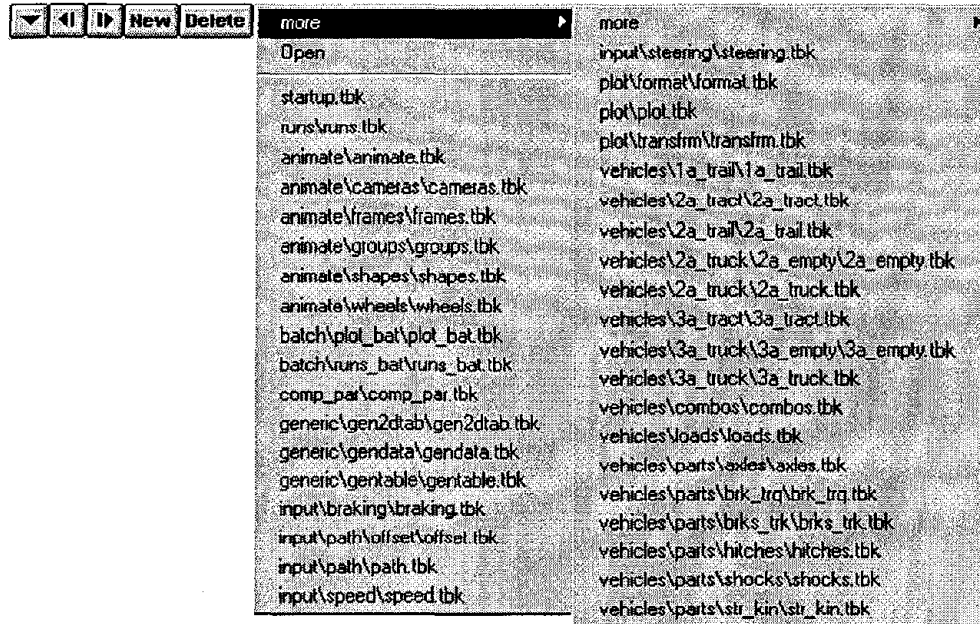
1. The right lane, from S=0 to S=1000 (green).
2. The right lane, from S=100 to S=1900 (green).
3. The left lane, from S=0 to S=1000 (magenta).
4. The left lane, from S=1000 to S=1900 (magenta).

In all four cases, the lane is 3.3 m wide, and grid lines are drawn at 10-m intervals.

How to Go Directly to Any VDM RoAD Library

You can jump from any VDM RoAD library to any other library.

1. Press the GO button in the ribbon bar to display a menu of other libraries in VDM RoAD. Select a library from this menu to leave the current library and go to the selected one.



The menu typically contains more items than will fit on a monitor with VGA resolution (640 x 480). In this case, the first item in the menu is **More**. Highlight it to view more menu items.

Notes: After using the GO menu, you can return to the previous library using the **Back** button from the ribbon bar.

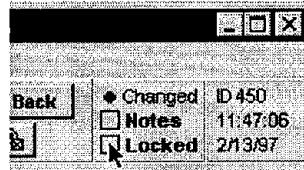
When you go to a library using the GO menu, VDM RoAD shows you the most recently viewed data set. This data set is not necessarily one that is used in the most recently made run.

How to Lock Your Data

VDM RoAD is designed to make it easy to answer “What if?” questions. Changing vehicle parts, inputs, run conditions, etc., is simple and rapid. At some point, you might overwrite some data you meant to keep, and you will no doubt conclude that changing parameters is too easy! However, VDM RoAD does have a means to protect your data from inadvertent changes.

Protecting a Single Data Set

1. Check the **Locked** box in the upper-right corner of a data screen.



Click here

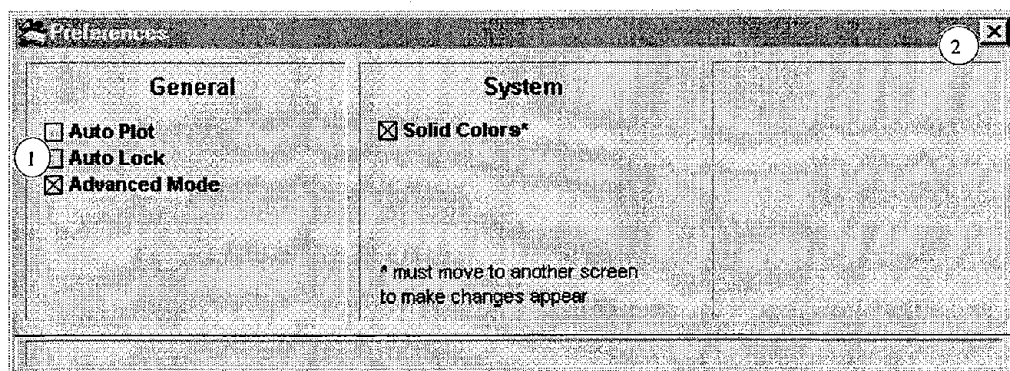
Notes: When the box is checked, try to modify a blue link or a yellow field. You will find that all pull-down menus next to blue links allow only the navigation option to go to the data set. You will also find that clicking on a yellow field brings up a message telling you that the data set is locked.

At any time, you can unlock the data set by clicking the box again to un-check it.

The locked box protects only the data shown on the screen. Data sets in linked libraries are not affected. For example, you might lock a data set in the **Runs** library. However you could still go to the vehicle data linked to the run and change the vehicle data set if it is not locked.

Automatically Locking All Data Sets As You Create Them

1. From any VDM RoAD data screen, click the preferences button in the ribbon bar (🔧) or select the **Preferences** command from the **Tools** menu. This brings up a dialog box with a few check boxes.



2. Check the box **Auto Lock** ①.
3. Close the dialog by clicking the X ② in the upper-right corner.

Notes: When the **Auto Lock** feature is enabled, every data set that you view will be locked automatically when you leave it.

It does not change the status of any existing data sets. They are locked after you visit them.

This feature can be disabled at any time by returning to the preferences screen and un-checking the box.

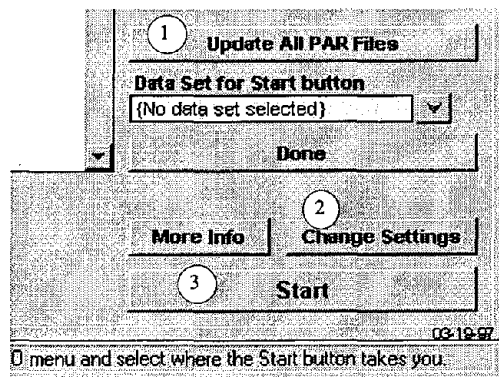
Automatically Locking All Data Sets in VDM RoAD

You can extend the above method to cause VDM RoAD to lock all data sets.

1. Follow steps 1 through 3 above.

Note: This will cause every data set to be locked once it is visited. Next, we will have VDM RoAD automatically visit every data set.

2. Use the **GO** button in the ribbon bar to go to the VDM RoAD startup screen (Startup.tbk).
3. Click the button **Change Settings** (2). This will hide the VDM RoAD logo and reveal more buttons, including those shown here.



4. Click the button **Update All PAR Files** (1). Besides updating all PAR files, the **Locked** buttons on every screen will be checked.

How to Delete Data Sets

VDM RoAD has several methods for deleting data sets.

To Delete a Single Data Set

1. Click the **Delete** button in the ribbon bar. VDM RoAD will display a warning, asking you to confirm that you want to delete the data.


Notes: You can by-pass the warning message by holding the Control key down when you click the **Delete** button.

The **Delete** button cannot be used if the **Locked** button is checked. First, the data set must be unlocked.

To Delete All Data Sets Except One



1. Click the **Delete** button in the ribbon bar while holding down the Shift key. VDM RoAD will display a warning, asking you to confirm that you want to delete the data.

To Delete an Arbitrary Group of Data Sets

1. Click the Library Tool button in the ribbon bar  or select the **Library Editor** menu item from the **Tools** menu. This brings up a floating palette with controls for selecting data sets by their titles, and then deleting the selected sets all at once.

How to Set Up Five or More Plots

The **Runs** screen has links to set up four plots. If you want to quickly generate five or more plots, you can use the **Plot Setup: Batch** screen, shown in Figure 3.5. Start from any VDM RoAD data screen.

1. Click the Batch Plot button in the ribbon bar  ¹⁶ or use the **Batch Plotting** command from the **Tools** menu. This takes you to the **Plot Setup: Batch** library.
2. Click the **New** button ¹⁵ to create a new data set for your new combination of plots and runs.
3. Using the **Add** ³ and **Remove** ⁴ buttons for the **Data Files**, put the name of the run of interest into the field with **Selected Data Files** ⁶.
4. Using the **Add** ⁹ and **Remove** ¹⁰ buttons for the **Plots**, put the name of all plots of interest into the field named **Selected Plots** ¹³.
5. Click the **Make Plots** button ¹⁴. This starts the program WinEP and instructs it to make one plot for each data set listed in the field **Selected Plots** ¹¹.
6. After viewing the plots, exit the plotter by clicking the  in the upper-right corner of the screen.

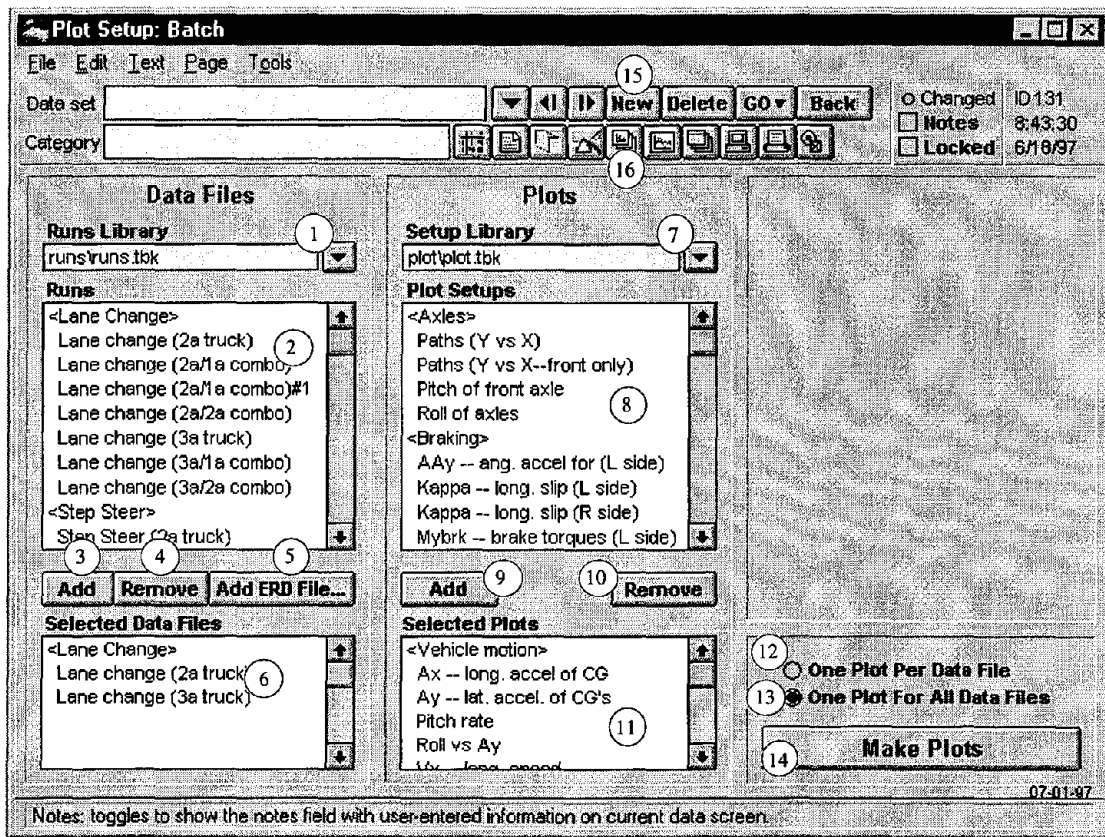


Figure 3.5. Setup for batch plotting.

How to Overlay Plots for Four or More Runs

The **Runs** screen has links to overlay data from three runs. If you want to overlay data from more runs, you can use the **Plot Setup: Batch** screen, shown in Figure 3.5.

The method is nearly identical to the one just described for making five or more plots. Two additional steps are:

- In step 3, put more than one run into the field with **Selected Data Files** (6).
- Be sure the **One Plot for All Data Files** button (13) is selected.

How to Make Many Plots in Batch Mode

You can generate a standard set of plots for many runs and send them directly to a printer. Again, you will use the **Plot Setup: Batch** screen, shown in Figure 3.5.

The method is nearly identical to the one just described for making five or more plots. Two additional steps are:

- In step 3, put more than one run into the field with **Selected Data Files** ⑥.
- Be sure the **One Plot per Data File** button ⑫ is selected.

How to Set Up Multiple Runs

It is possible to set up more than one run ahead of time. It is also possible to re-do a set of runs with different conditions. Start from the **Runs** library.

1. Set up the runs of interest by creating a distinct data set for each combination of vehicle, input, speed, stop time, etc., that is of interest. Do not click the **Run Simulation** button.
2. Go to the **Runs: Batch** library by clicking the button in the ribbon bar (📁) ⑨ or use the **Runs** menu item from the **Tools** menu. This brings up the data screen shown in Figure 3.6.
3. Click the **New** button ⑩ to create a new data set for your combination of runs.
4. Using the **Add** ③ and **Remove** ④ buttons, put the name of the runs of interest into the field named **Data Sets to Run** ⑤.
5. Optionally enter parameters (with keywords) in the yellow fields ⑥ to override the parameters that are specified in the selected Run data sets. For example to make all runs with the a simulation speed of 100 km/h, enter the line: `speed 100`

Note: See Appendix D for a complete list of keywords recognized by the simulation programs in VDM RoAD.

6. Optionally use Links 1, 2, and 3 ⑦ to link to data sets that will override those specified in the selected runs. For example, to make all the runs with no braking input, link to the braking input library and select the data set **No Braking**.

Notes: It is possible to link to a vehicle description, thereby overriding the vehicles that were selected in the runs data sets. However, if this is done, all runs must involve the same kind of vehicle. For example, if a run is set up for a three-axle truck, then when the run is made in batch mode the solver program for three-axle trucks will be used. If a different kind of vehicle is linked to the batch run screen, the results will not be what you expect.

If the vehicle is not specified on the batch run screen, then different types of vehicles can be simulated in one batch.

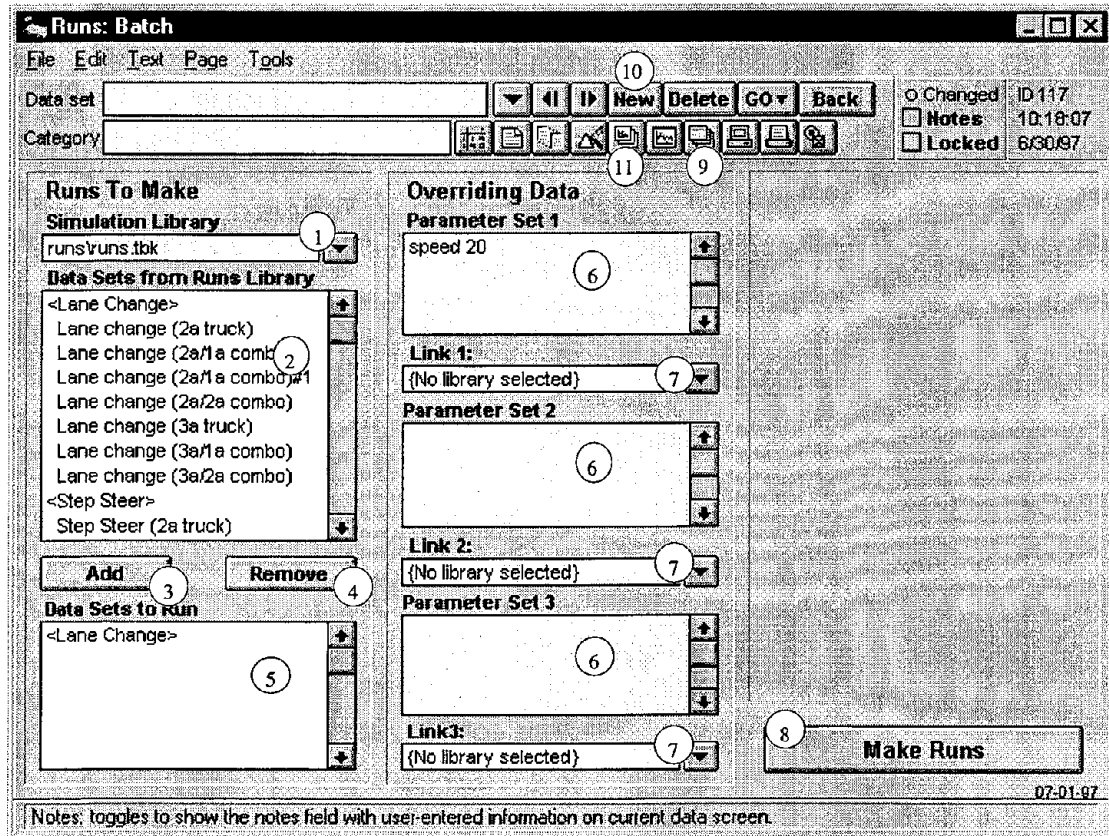


Figure 3.6. Batch run screen.

7. Click the **Make Runs** button (8) to make the batch of runs.

Notes: You can view the results from the individual runs data sets, or use the batch plot library (11).

If you specified any parameters or links in steps 5 and 6, then the individual run data sets are modified such that the link with the label **Overriding Data (from Batch)** is set to the batch data set. That way, you can get back to this data set (batch run) to see what parameters or links were applied when the run was made.

The echo files generated for each run show the model parameters that were used in the run, regardless of whether they came from the normal run data set or from the batch override.

How to Make New Plot Descriptions

VDM RoAD comes with about 30 plot descriptions. It's easy to make more. Start with any data screen in VDM RoAD.

1. Click on the plot setup icon in the ribbon bar (13) or use the **EP Setup** command from the **Tools** menu. This takes you to the plot setup library shown in Figure 3.7.

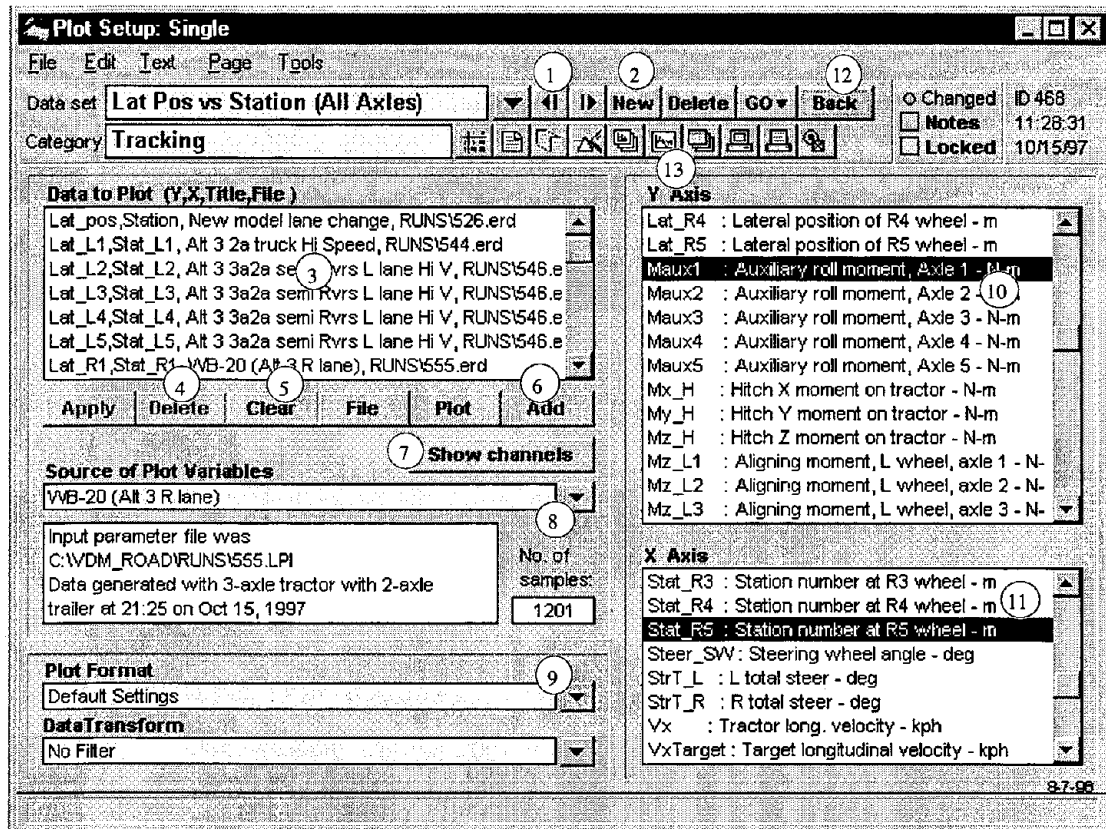


Figure 3.7. The Plot Setup screen.

2. Use the navigation buttons (1) in the ribbon bar to find a plot setup similar to what you want.
3. Click the **New** button (2) to make a new data set.
4. Select a run from the library labeled **Source of Plot Variables** (8). When you select a run, the output ERD file associated with the run is scanned and all of the variables are listed in fields (10) and (11).

Notes: The run must have been made in order for the ERD file to exist.

If you are sure that the run whose name is showing in the link (8) does exist, you can force VDM RoAD to scan the file by clicking the **Show Channels** button (7). The reason that the fields are initially blank is that the file may have been deleted after the data set was last viewed.

If the ERD file has more than about 150 variables, only the short names are shown in fields ⑩ and ⑪. If it has fewer variables, more information is shown.

5. Click the **Clear** button ⑤ to clear the field called **Data to Plot** ③.
6. Select X and Y channels of interest from fields ⑩ and ⑪. Add them to the list of data to plot ③ using the **Add** button ⑥ or by double-clicking in either field ⑩ or ⑪.
7. Continue to build the list of data to plot ③. Use the **Add** ⑥ and **Delete** ④ buttons as needed to build the list and correct mistakes.

Notes: When you click the **Plot** button from the **Run** screen, the list of variables ③ is sent to WinEP. The associated file names are not used. Therefore, it doesn't matter which run you use to build the file description, so long as the ERD file contains the variables that you want plotted.

You can add variables that are not produced by all solver programs. For example, tire forces for axles four and five are not available for simulations involving vehicles with less than four axles. When the plot data set is selected for an output file that does not have those variables, the plotter will ignore them and show those that are in the file. Thus, the same data set can be used to view the outputs for many different solver programs.

8. To make another plot setup, repeat steps 2 through 7.
9. When you are through making new plot descriptions, leave the library by clicking the **Back** button ⑫.

How to Change Plot Formatting

Plot appearance is controlled by a format data set. You use the formatting to assign the colors of lines, the size of text, whether a grid is shown, etc.

Each plot description is linked to a format data set (see ⑨ in Figure 3.7 on page 36). As shipped, every plot description is linked to a format data set with the title **Default Settings**. If you change the default settings, all future plots will have a new appearance.

1. Go to the **Plot Setup** library by clicking the plot setup icon in the ribbon bar (📄) ⑬ or use the **EP Setup** command from the **Tools** menu.
2. Go to the **Plot Format** library by following the link ⑨.

Note: Another way to get to the library is through the **GO** menu. Look for the menu item **Plot\Format\Format.tbk**.

3. Refer to the description of the **Plot Format** screen in Chapter 8 for information about how you can control the plot appearance from the these settings.
4. Modify the data set whose title is `Default Settings`.

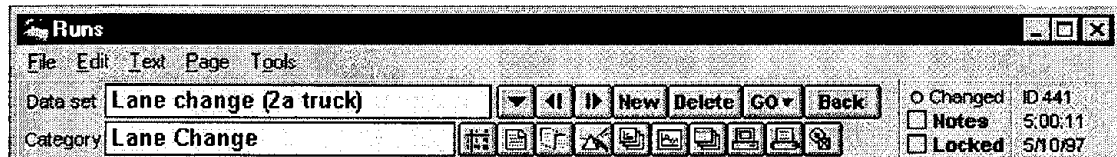
How to Assign the Default Text Editor

The **View Echo File (All Parameters)** button on the **Runs** screen automatically loads a text file into an editor. You can choose the editor that is used.

As shipped, the default text editor is program called `QLPa.d`. You can replace the default with a different Windows editor if you prefer.

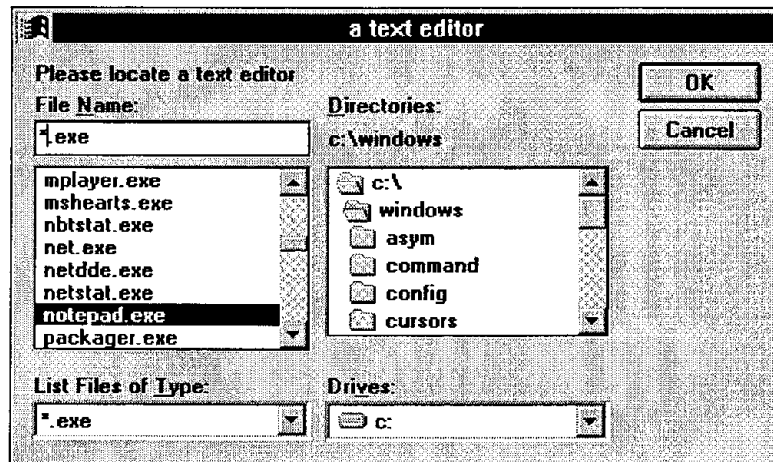
To assign a new editor, start with any data screen in VDM RoAD.

1. Control-click on the text icon in the ribbon bar.



Control-Click here

This should bring up the following dialog box.



2. Link to a text editor using the above dialog box.
3. Click the OK button after you have selected an EXE file.

How to Make a New Vehicle Input

Use the following method to create a new vehicle input that can be used for future runs. Using this process leaves existing vehicle descriptions intact.

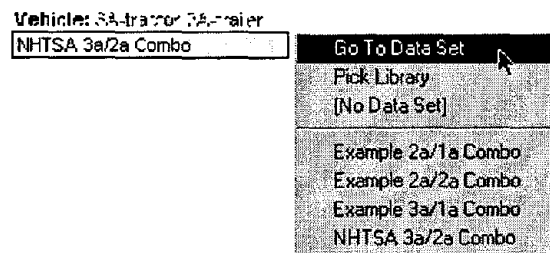
The parameter values for a vehicle model are spread over many data sets. Typically, you will want to change a few and leave most the way they are.

Note: In the following instructions, the concept of going “up” and “down” is based on the map shown in Figure 2.3 on page 14.

Method

Start from the **Runs** screen (see Figure 3.1 on page 15).

1. Find an existing data set that involves the type of vehicle you are interested in. To navigate through the **Runs** library, use the buttons (2) next to the Data Set (title) box (1).
2. Go down to the linked vehicle data set.



Notes: The top-level vehicle screen defines a loaded vehicle. If you only want to change the axle loads, you do not have to go any further. However, the following steps are needed to go down and change the vehicle properties.

For straight trucks, there is a single link to the unloaded vehicle. For combination vehicles, there is one link for the tractor and another for the trailer.

3. Repeat step 2 as needed to go down to more detailed levels. To start, go down to the linked data set for a vehicle unit (unloaded truck, tractor, or trailer). If you are interested in making a new component or subsystem data set (tire, suspension, etc.), use the appropriate link to go down to it. As indicated in the map shown in Figure 2.3 on page 14, it is possible to go down through several levels of detail.

For the purpose of making changes, you are at the “bottom” when

- the current screen has no blue links, or
- the current screen has blue links, but the data sets available from the menus associated with the links are suitable for your purposes. In other words, you do not have to modify or create any data sets that are further “down” in the data base.


4. Click the **New** button (in the ribbon bar) to copy the current data set.
5. Give the new data set an appropriate title in the yellow “Data Set” field ①.
6. Modify the data set:
 - Modify the values in the yellow fields as needed.
 - Change the data sets in any blue links, using the adjacent pull-down menu.

Note: When you clicked the **New** button, you created a new data set that is now available for use in VDM RoAD. It is in the same library as the original, and appears in pull-down menus that show the library contents. However, it will not be used in any simulations until you have linked to it from a higher-level data set.

7. Go back up one level, using the **Back** button in the ribbon bar.
8. Click the **New** button (in the ribbon bar) to copy the current data set.
9. Give the new data set an appropriate title (see the yellow field “Data Set” ①).
10. Go to the blue link that connects to the library in which you just added a data set, and select the new data set. For example, if you modified a wheelbase in a tractor data set, then go to the blue link for tractors and select the new tractor data set you just created.
11. Repeat steps 7 through 10 until you are back at the **Runs** screen. At this point, you can make a new run or modify the current run. Either way, you will select the new vehicle data set from the linked library of vehicles.

How to Modify an Existing Vehicle Input

Use the following method to modify an existing vehicle description. Start from the **Runs** screen (see Figure 3.1 on page 15).

1. Find an existing data set that involves the vehicle whose description you will modify.
2. Go down to the linked vehicle data set. Change any yellow fields or blue links as needed.
3. Repeat step 2 to go down more links and change other components as needed.
4. When all changes have been made, return to the **Runs** screen by clicking the **Runs** button in the ribbon bar () or using the **Tools** menu.

Warning: Almost every data set in the VDM RoAD data base is referenced by other data sets (via the blue links). When you change values in a data set, you have in effect also changed all data sets that reference it. For example, if you change a steering system parameter, every vehicle data set that is linked to the steering system data set will use the new parameter values in future simulation runs.

Hopefully, this is exactly what you intend. If not, then consider using the **New** button to create a copy of the data set, then change the copy and link to it in those places where you want to use the new data.

How to Improve the Appearance of the VDM RoAD Screens

Windows has many settings for screen size, number of colors, and default font size.

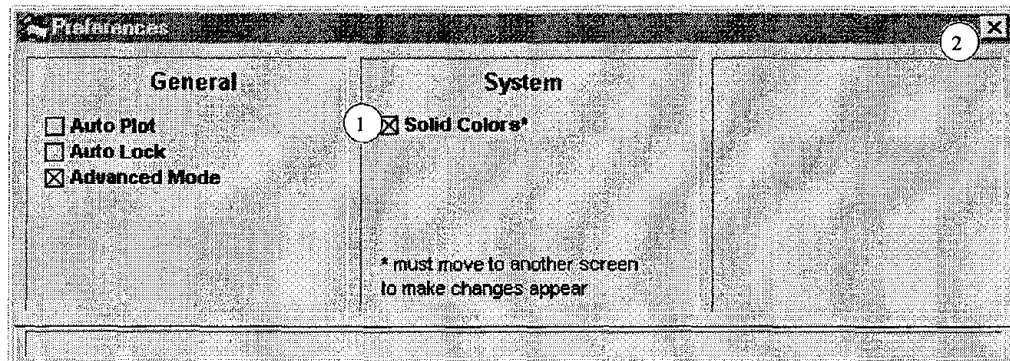
Font Size

The VDM RoAD data screens look best when the system display properties are set for **Small Fonts**. However, the screens should be readable for any valid setting.

Colors

The VDM RoAD screens look best when 256 or more colors are supported. If your machine is set to support 16 or fewer colors, VDM RoAD has a setting to produce yellow and blue fields using dot patterns rather than solid colors. A global setting is used to control whether VDM RoAD uses solid colors or patterns.

1. From any VDM RoAD data screen, click the preferences button in the ribbon bar (🔧) or select the **Preferences** command from the **Tools** menu. This brings up a dialog box with a few check boxes.



2. If your machine is set up to support 256 or more colors, Check the box **Solid Colors** (1). Otherwise, un-check the box.
3. Close the dialog by clicking the X (2) in the upper-right corner.

Size of VDM RoAD Data Screens

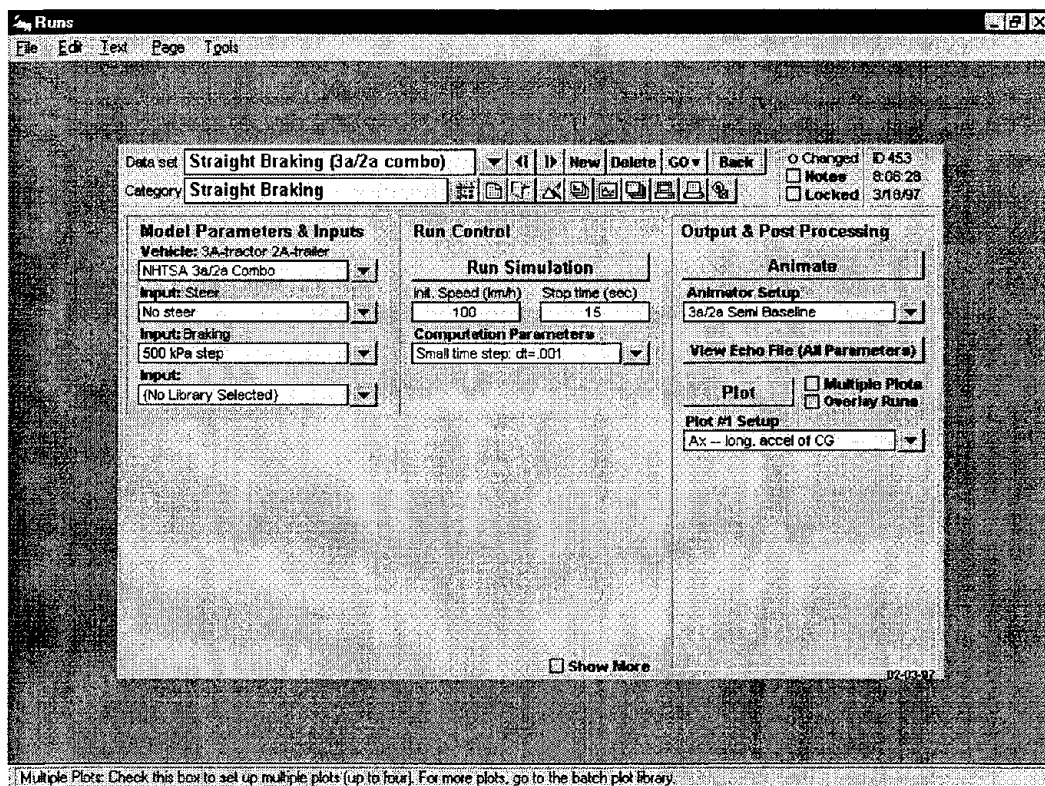
The VDM RoAD data screens are designed to fit on a VGA display (640 x 480).

1. The upper-right corner of the window has three Windows-standard buttons: [restore down] [maximize] [close]. The middle button “maximizes” the window to fill the monitor screen. If your monitor

has a larger display area than VGA, the result is that the menu bar is put at the top of the total screen, the main part is centered, and the space around the main part is filled with gray.

2. To return to the normal VDM RoAD display, press the F11 key or use the **Size to Page** command from the **Page** menu.

Note: If you have modified the window size by dragging the window borders, the F11 key (or **Page** menu) is the only way to restore the size.



Appearance of a data screen that has been maximized.

How to Install VDM RoAD in a New Directory

You can move VDM RoAD to a new place in your file system, so long as you do not modify the structure within the VDM RoAD directory. For example, you can move the root VDM RoAD folder from drive C to drive D. If you move the VDM RoAD files after the initial installation, you must make three additional changes:

1. Inform Windows of the new location, in order to start VDM RoAD using the runtime ToolBook software.

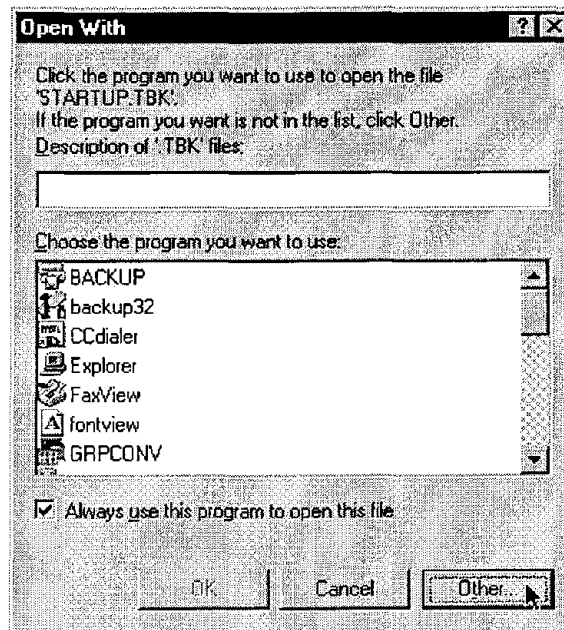
2. Cause VDM RoAD to regenerate text files used to communicate with the plotter, animator, and solver programs.
3. Manually edit pathnames to IHSDM files in the Input\Highway library.

Runtime ToolBook

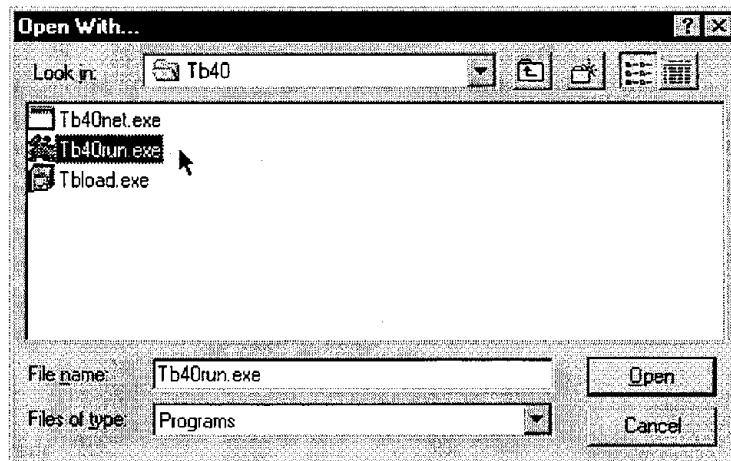
The VDM RoAD data base is managed by a program called `Tb40run.exe`, located in the `Tb40` folder in VDM RoAD. As part of the VDM RoAD installation, the `Tb40run` program is registered as being associated with the file extension `TBK`. If you move the folder containing `Tb40run`, then Windows must be informed of the new locations.

Your objective is to associate the file type `TBK` with the file `Tb40.exe`. If you know how to do this, go ahead and do it. If not, the following steps describe a method that will work.

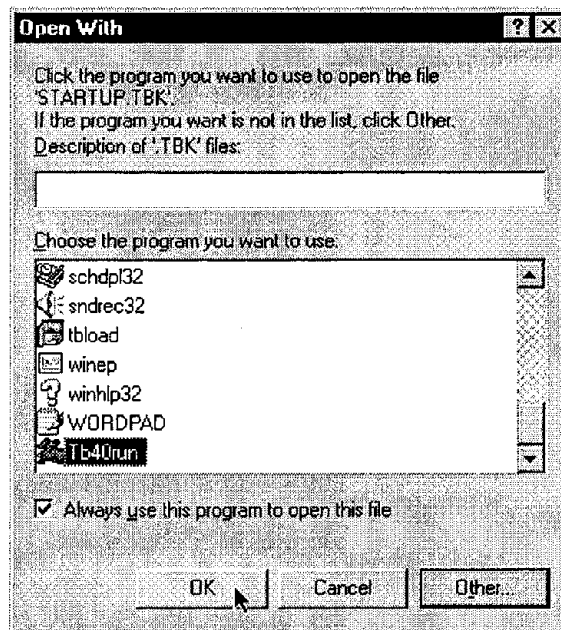
1. From the desktop or Windows Explorer, find the file `Startup.tbk` in the VDM RoAD folder. Double-click the file to try to open it. If VDM RoAD launches, congratulations! No work is needed. However, more likely, Windows will not know how to open the file, and will bring up the following dialog:



2. Click the button `Other`. This brings up the next dialog, to locate a file.



3. Locate the file `TB40run.exe`, in the folder `Tb40` in VDM RoAD. After selecting the file, click the `Open` button to complete the selection and close the dialog box. The previous dialog should now show the program.



4. Make sure the box “Always use this program to open this file” is checked.
5. Click the `OK` button to close the dialog.

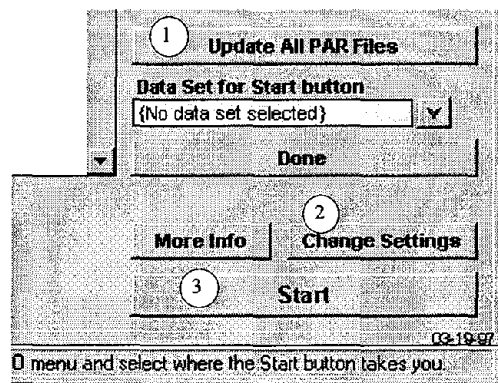
You have now associated the file type `TBK` with the program `Tb40run` in its new location.

Update Pathnames in PAR Files

The solver programs require absolute pathnames, and VDM RoAD creates them “behind the scenes” using PAR text files. When VDM RoAD is installed, it “looks” at its location and generates many files using that location to create pathnames. For example, if it is in a folder `c:\my_vdm`, then all pathnames are automatically set to begin with “`c:\my_vdm.`” If you move the VDM RoAD directory, you must instruct VDM RoAD to regenerate all of the absolute pathnames in the PAR files.

After you have moved the root VDM RoAD folder to a new location, do the following:

1. Use the **GO** button in the ribbon bar to go to the VDM RoAD startup screen (`Startup.tbk`).



Lower-right corner of Startup screen.

2. Click the button **Change Settings** ②. This will hide the VDM RoAD logo and reveal more buttons, including those shown here.
3. Click the button **Update All PAR Files** ①.

Note: The process of updating all PAR files to generate new absolute pathnames usually takes a few minutes. The time required depends on the speed of your computer and the number of data sets you have in the VDM RoAD data base.

How to Import Data from Another Copy of VDM RoAD

There are several occasions where you may want to import data from another copy of VDM RoAD. If you receive an update for VDM RoAD, this is how you save data entered in the older version. Another reason might be that VDM RoAD is installed on several computers, and you want to import data from one installation into the other.

To import data, both installations of VDM RoAD must be available from the same computer.

To Import Data from One Library

1. Go to the library in VDM RoAD from which you want to receive the imported data. Any data screen in that library will do.
2. Select the menu item **Import Data Library** from the **File** menu. VDM RoAD will bring up a file browser dialog box.
3. Select a library TBK file of the same type as the current one. For example, if the current library contains tire data, you should open another tire library file. Often, the two files have the same name but are in different folders.
4. Click OK. VDM RoAD will copy each data set of the old library and paste the contents into the new one. If a data set already exists with the same name, the copy is given a unique name by appending a number.

Note: This method is reliable for libraries with only yellow data fields. For libraries with blue links, the names of the linked data sets are copied. However, if the data sets do not exist in the new VDM RoAD installation, then the links are not valid and must be reset by hand.

To Import All Data

1. Go to Startup screen.
2. Select the menu item **Import Data Library** from the **File** menu. VDM RoAD will bring up a file browser dialog box.
3. Select the Startup library file `Startup.tbk` for the other copy of VDM RoAD.
4. Click OK. VDM RoAD will recursively copy each data set of the old library and paste the contents into the new one.

Note: This method works well when the directory structures of the two versions of VDM RoAD are identical. If the structures differ in any way, then some of the links will not be valid and must be set by hand.

How to Convert Units for Tabular Data

You may have occasion to import measured tabular data into VDM RoAD. If the units used for the data do not match the requirements for the VDM RoAD data screen, you can use a built-in calculator tool to convert the units. The process is as follows.

1. Copy the tabular data to the clipboard.
2. From any VDM RoAD data screen, click the calculator tool button in the ribbon bar or select the **Calculator** command from the **Tools** menu.



Click here

- Paste the tabular data into the tabular data field ① of the calculator data screen.

- Type definitions for the transformed X and Y values into the definitions field ③. The definitions should have the form: $A*X$, $B*Y$ where A and B are numerical scale factors. For example, suppose the X values are in inches and Y values are in pounds. To convert to a table where the X values are in millimeters and the Y values are in Newtons, use the definition: $25.4*X$, $4.4482216152605*Y$.
- Click the **Transform Series** button ④ to perform the conversion.
- Click the **Plot** button ② to confirm that the conversion was performed as you would expect.
- Optional. If the data should not be separated with commas, click the button **Remove ',?'**.
- Select the entire contents of the tabular data field. The menu command **Select All**, under the **Edit** menu (Ctrl-A) is helpful for doing this.

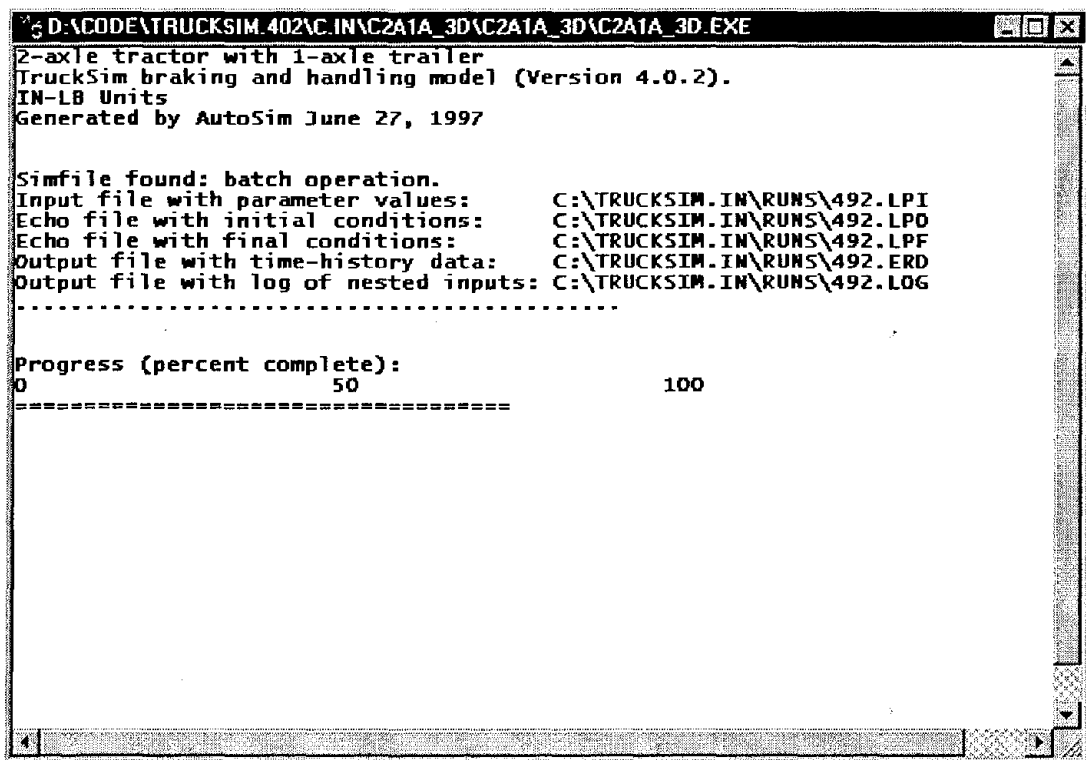
9. Select the **Copy** command from the **Edit** menu (Ctrl-C).
10. Go to the data screen where the tabular data will be stored.
11. If a data set does not exist for the transformed data, create one using the **New** button and give it a name.
12. Paste the clipboard contents into the field used to store the numbers.
See Chapter 8 for details on using the calculator data screen.

4. The Simulation Solver Programs

VDM RoAD contains programs that solve the equations of motion for vehicle models, predicting motions, forces, and other variables. A solver program is run automatically whenever you click the **Run Simulation** button from the **Runs** screen. VDM RoAD allows you to set up inputs to the programs and view their outputs without ever dealing with the details of how these programs work. However, it can sometimes be useful to understand how they read and write files.

Overview of Program Operation

The solver programs are so-called console applications. Each behaves as a terminal console, with no user interface other than the display of text and the acceptance of keyboard entries. They have the same appearance as a plain DOS text program, although they are technically 32-bit Windows programs.



```
D:\CODE\TRUCKSIM.402\C.IN\C2A1A_3D\C2A1A_3D\C2A1A_3D.EXE
2-axle tractor with 1-axle trailer
TruckSim braking and handling model (Version 4.0.2).
IN-LB Units
Generated by AutoSim June 27, 1997

Simfile found: batch operation.
Input file with parameter values: C:\TRUCKSIM.IN\RUNS\492.LPI
Echo file with initial conditions: C:\TRUCKSIM.IN\RUNS\492.LPO
Echo file with final conditions: C:\TRUCKSIM.IN\RUNS\492.LPF
Output file with time-history data: C:\TRUCKSIM.IN\RUNS\492.ERD
Output file with log of nested inputs: C:\TRUCKSIM.IN\RUNS\492.LOG
.....

Progress (percent complete):
0 50 100
=====
```

Screen display when solver program is running.

When the program starts, it creates a window in which text is displayed. The name of the program file is typically shown as the window title.

You can run the solver programs interactively or in batch mode. The mode is determined by the existence of a batch control file with the name `Simfile`. `Simfile` is created automatically by VDM RoAD when you click a Run button.

If the file `Simfile` does not exist (in the folder in which the program is located) then the program runs in interactive mode. It prompts you for an input file, an output ERD file, and two output echo files. After the last file is written, the program quits.

If the file `Simfile` does exist, then all filenames are read from it. It quits when the `simfile` is fully processed and the last output file is written by the solver program.

Properties of the window created by the solver program can be adjusted by selecting the solver program (extension EXE, found in the folder `VDM RoAD\Programs`), right-clicking to display a pop-up menu, and selecting **Properties** from the menu.

As shipped, the solver programs close their windows when they finish. When the window is closed, the most recently accessed program comes to the top. When run normally from within VDM RoAD, the most recent program is ToolBook, which was showing the **Runs** screen from which the run was initiated. Therefore, the normal behavior is that the Solver program brings up a window, runs for the time needed to perform the simulation calculations, then quits and closes the window, leaving you back at the **Runs** screen.

Notes: The progress bar-graph that is shown during a run is generally not a good indicator of the progress of a run. The reason is that it is based on time, not distance. The default stop times in VDM RoAD are 200 seconds, because the simulation stop condition is usually due to having reached a target station number. Therefore, the chart will be updated slowly unless the run would actually extend to 200 sec.

You can change the stop time from 200 sec to some other value, as described on page 20 in chapter 3.

File Types

When run from the graphic interface in VDM RoAD (i.e., by clicking the **Run Simulation** button from a **Runs** screen), all files associated with a run have the same base name and differ only in their extensions. The base name is the current ID number displayed in the upper-right corner of the **Runs** screen.

Table 4.1 shows the files and their extensions.

Details of the file types are provided in Appendix D, along with some examples. Short summaries are provided below.

Simfile

As noted earlier, `Simfile` is a batch control file. Whenever you click the **Run** button, a new `Simfile` is created in the same folder as the simulation program (by convention, this is the folder `Programs`). If a file named `Simfile` already exists, it is overwritten. Because `Simfile` is automatically re-generated before each run, it can be deleted at any time without loss of information.

Table 4.1. Standard files created when a simulation is run.

Name	Location	Creator	Description
<code>Simfile</code>	<code>Programs</code>	SGUI	Batch control file.
<code><id>.LPI</code>	<code>Runs</code>	SGUI	List of parameters, read as input by program.
<code><id>.LPO</code>	<code>Runs</code>	Program	List of parameters, written as output by program. Also contains initial conditions. Can be used to repeat a run.
<code><id>.LPF</code>	<code>Runs</code>	Program	List of parameters and final conditions, written as output by program. Can be used to continue a run.
<code><id>.ERD</code>	<code>Runs</code>	Program	Header for ERD file, needed by plotter and animator programs.
<code><id>.BIN</code>	<code>Runs</code>	Program	Numerical values of output variables, stored in binary form, and used in association with ERD files.
<code><id>.LOG</code>	<code>Runs</code>	Program	Log of all input files, including nested PAR files.

The `Programs` folder contains the `simfile` used for the most recent run. Therefore, if you run a solver program independently of VDM ROAD, it will repeat the last run made. However, if the last run made was with a different program, the run may not be what you expect. For example, if the last run was made with a two-axle truck model (`f2a.exe`), and you run a tractor-semitrailer model (`f3a2a.exe`), only the parameters common to the two models are used.

To run a program in the interactive mode, you must delete the `simfile`.

LPI Files

The LPI file lists the input parameters for the simulation program.

When you click the **Run Simulation** button, a new LPI file is created in the folder containing the `Runs` library (e.g., `Runs`). The LPI file contains information from the **Runs** screen, such as the title of the run and pathnames for the data sets linked to the run (vehicle, steer input, etc.).

LPO and LPF Echo Files

When a simulation program runs, it creates summary files that list each and every parameter value. One of these files is created before the run (LPO), and the other is

created at the end of the run (LPF). Both contain all parameter values. In addition, the LPO file contains the initial conditions for the state variables in the simulation. The information in the LPO file is sufficient to exactly repeat a run.

The difference between the LPI and LPO files is that the LPO file contains every parameter value in one file, whereas the LPI file contains pathnames of other files that contain the data. Also, the LPI file may not reference every parameter used by the simulation program. For example, the initial conditions listed in the LPO file are normally not specified on input (they are assigned default values by the simulation program).

The LPF file is nearly identical, except that instead of initial values, it contains final values of the state variables. By modifying the start and stop times, an existing run can be continued.

ERD and BIN Files

The main purpose of each simulation program is to calculate time histories of variables of interest. Those time histories are stored in a binary data file with the extension BIN. The BIN files contain numerical data organized by channel number and sample number, similar to test data recorded on a multi-track recorder. A companion file, with extension ERD, describes the layout of the BIN file and also contains labeling information for each variable. It also contains the information that would normally be put into a log sheet summarizing the data, including text needed for preparing graphical plots of the data.

By itself, a BIN file is useless. It has no structure, and cannot be easily understood without the layout information contained in the ERD header file.

By convention, ERD and BIN file pairs are simply called ERD files. The name ERD is used because the format was created by the Engineering Research Division at UMTRI.

Data processing programs for ERD files obtain most of the information needed from the file itself. For example, the high level of automation in the animator and plotter exists because both were designed to work with ERD files.

The format of an ERD file is described in Appendix A. Appendix J contains lists of all of the variables in the ERD and BIN files for the VDM RoAD models.

Text Output Files

VDM RoAD can be made to produce text-only ERD files. In this case, the header information in the ERD file is followed immediately by the numbers, written in text. There is no BIN file.

VDM RoAD can also be made to produce simple text output files, for export to spreadsheet programs and other analysis software.

See the section **Computation Parameters** in Chapter 8 for details on how to generate text output files.

5. The Animator

VDM RoAD includes a program for animating wire-frame figures to visualize vehicle motions. The animation is accomplished by drawing images similar to what would be seen with a video camera, and updating the images many times per second to show motion.

Updates

The animator program is used in a number of software packages. The software was developed jointly by MSC and UMTRI, and is now maintained by MSC. The latest version is available from the Internet at:

<http://www.trucksim.com/animator.html>

If you download a newer version, simply replace the file `Animator.exe` in the `Animate` folder in VDM RoAD.

Overview of Operation

To help understand the animator, imagine that you are aiming a video camera at a vehicle as it moves down the road. The camera records an image of the vehicle and ground as seen from your point of view 30 times per second. By playing back those images at the same rate (30 times per second), you see the vehicle move relative to its surroundings.

If the camera has a zoom lens, you can zoom in to fill the screen with the vehicle. Or, you can zoom out to shrink its apparent size.

What you see depends in part on where you are located. You could be standing on the ground, in which case your position would not change. Or, you might be in another vehicle that moves with the target. In the animator, your location is defined as the “camera point.”

Your view is also determined by how you aim the camera. You might have the camera sitting on a tripod, aimed at a fixed point in space. The vehicle would only be visible while it is in camera view. More likely, you would move and aim the camera as needed to keep the vehicle in view. In the animator, the camera is aimed directly at a “look point.”

Figure 5.1 shows the basic geometry and the relationships between the camera point, a look point, the 3D system being animated, and the 2D image that is recorded.

The animator allows you, the user, to build and modify descriptions of the system to be animated. In order to use the program effectively, it is helpful to understand the concept of a moving reference frame.

Reference Frames

The animator creates images based on a set of visible objects that includes a grid, wheels, and arbitrary wire-frame shapes defined by a sequence of connected lines. Some of the wire-frame objects are organized into groups that move together (for example, the body of a truck).

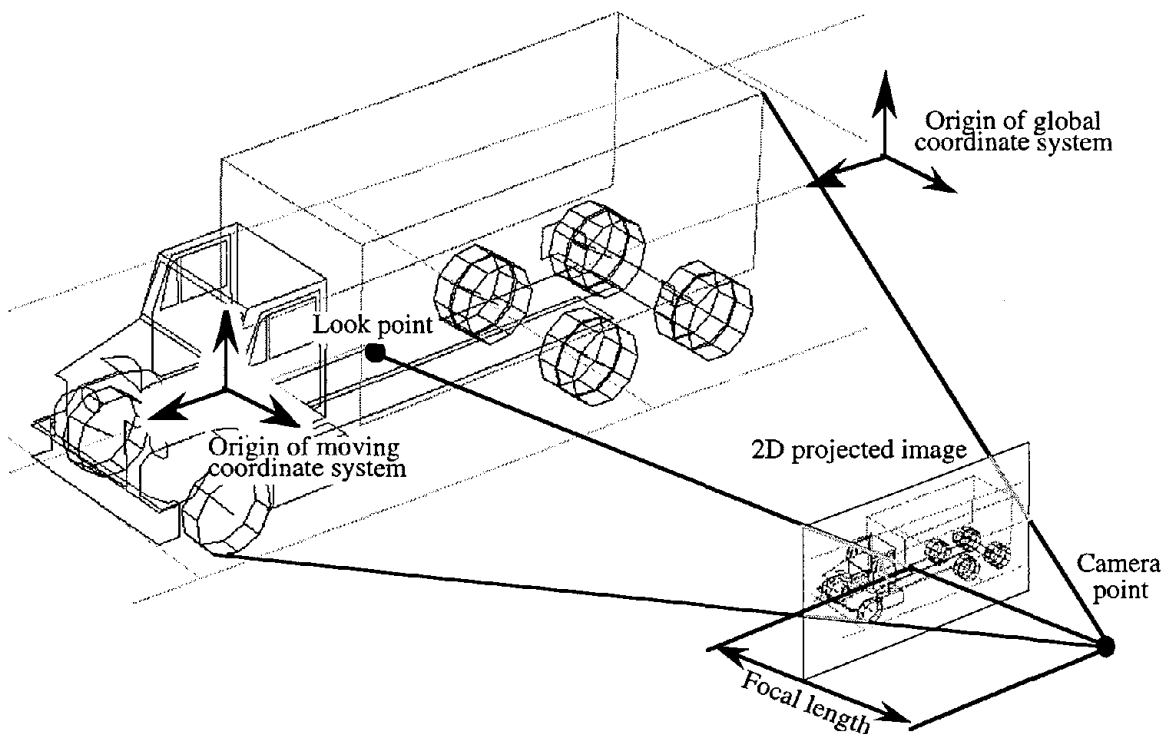


Figure 5.1. Geometry of the camera point and the look point.

A group of points and objects that maintain a fixed relationship (i.e., that constitute a rigid body) is called a reference frame. Although the reference frame might move and rotate, the spatial relationships between objects in the reference frame do not change relative to each other. In the animator, all motions are associated with reference frames. Each reference frame has a rectangular coordinate system that is used to describe 3D locations of points within that reference frame. For example, Figure 5.1 shows the origins and axes for two reference frames: a global non-moving frame, and a frame that moves with a truck body.

In general, it is not possible to tell how many reference frames exist by looking at a single image. However, by observing an animation, it is sometimes possible to visually see the effects of all reference frames. For example, the system shown in Figure 5.1 includes eight reference frames:

- A fixed global reference frame is used to locate a fixed grid that shows the ground plane. The three axes of a coordinate system fixed in this reference frame are shown in the figure.

- A moving frame is associated with the vehicle body. The body is simply a series of lines drawn in this reference frame. The three axes of a coordinate system fixed in this reference frame are shown in the figure. Given that the coordinate system is fixed in a moving reference frame, then it follows that the coordinate system moves.
- Each of the six wheels is associated with a separate reference frame. In the case of the rear axles, both tires on the dual wheel assemblies are fixed together, and therefore are contained in the same frame. Although the left and right wheel assemblies for an axle move together with the axle, they are allowed to spin independently. Because they have different spins, separate reference frames are required for the left and right sides.

Wire-frame shapes are all defined using local X-Y-Z coordinates. The animator transforms the local X-Y-Z values to global X-Y-Z values in order to draw the shape. To perform the transformation, the animator must know the location and orientation of the reference frame relative to the global reference frame. This is defined by six variables: three coordinates (global X-Y-Z) and three rotation angles that are called Euler angles.

The six variables needed to locate and orient the reference frame for the vehicle bodies and wheels are all computed as part of the simulation. It is also possible for you, the user, to define new reference frames using combinations of the variables computed in the simulations. Details for specifying reference frames are provided in Chapter 8 in the section **Animator Reference Frames**. That section also provides more information about Euler angles.

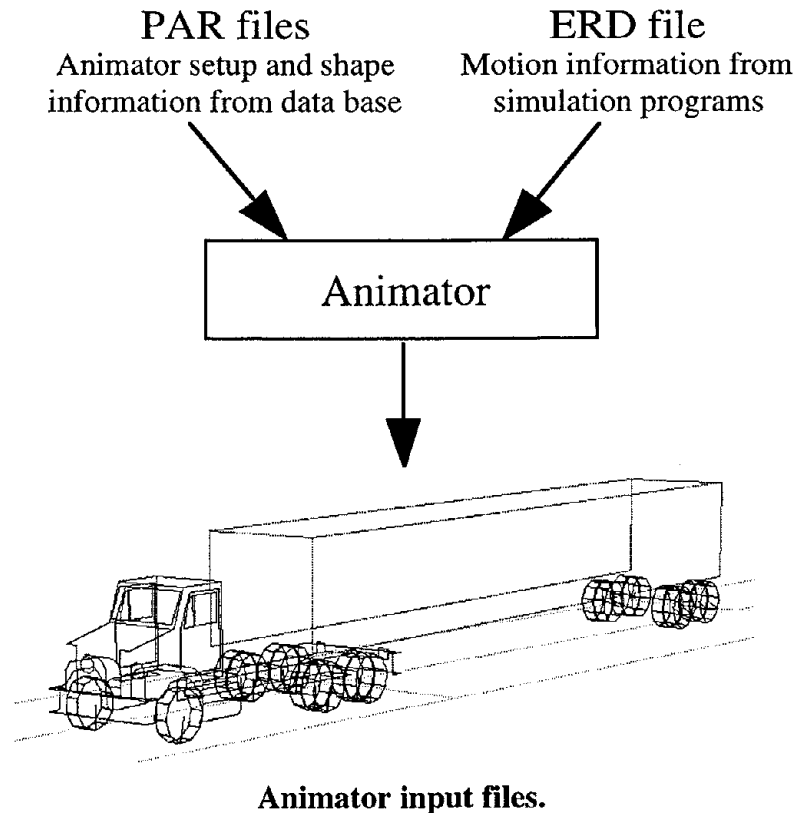
Files

When the animator is started from the VDM RoAD **Runs** screen, the necessary files are automatically opened. Therefore, it is not necessary to know the names of the input files. However, an understanding of how the animator reads the necessary information from files might be helpful in understanding its operation.

The animator program reads two kinds of input files (see the next figure). As noted above, all motion is due to movements of reference frames, as defined by up to six variables. Those six variables are read from an ERD file, created by the simulation programs whenever a run is made. Appendix A provides details of the ERD file format, and Appendix J lists the variables contained in a typical VDM RoAD ERD file.

Other information, such as program settings, definitions of parts, shape information, etc. is described with keyword-based text files, typically with the extension PAR. These files, called PARFILES, have the same general design as the input files read by the simulation programs. A single top-level PARFILE contains the names of other PARFILES with camera information, vehicle information, reference frames, etc.

Appendix C describes the keywords used in the animator PARFILES. The keywords are also mentioned in the documentation for the six animator library screens contained in Chapter 8.



Units

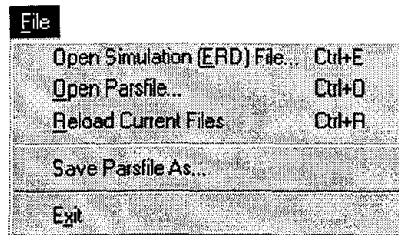
The animator requires all angles to be in degrees, and all coordinates to be in the same units of length (meters). The animator allows user-defined scale factors for all coordinates and variables. This means that a mixture of units can be accommodated. For example, if some coordinates have units of meters and others millimeters, the values in millimeters can be associated with a scale factor of 0.001 to convert them to meters.

See the sections **Animator Reference Frames**, **Animator Shapes**, and **Animator Wheels** in Chapter 8 for details on how to set scale factors.

File Menu

The file menu is used to read files, write files, and exit the program. Printing is not functional in the current version. To print a graphic, you can (1) copy it to the clipboard

and paste it into another application (e.g., WinPad), and (2) print from the other application.



Open Simulation (ERD) File...

Select this command to open an ERD file with the Windows file browse dialog. Recall that the six variables needed to define each reference frame are obtained from the ERD file produced by a simulation program. If you read a new ERD file, the animator will show motions from that simulation, using the same camera point of view, shape descriptions, etc. The keyboard command is Ctrl-E.

Open Parsfile...

Select this command to open a PARSEFILE (extension = PAR) with the Windows file browse dialog. If you read a new PARSEFILE, the animator will show the same motions as before, using the new camera point of view, shape descriptions, etc., as defined in the new PARSEFILE. The keyboard command is Ctrl-O.

Reload Current Files

Use this command to restore the original settings by reloading the current ERD file and PARSEFILES. The keyboard command is Ctrl-R.

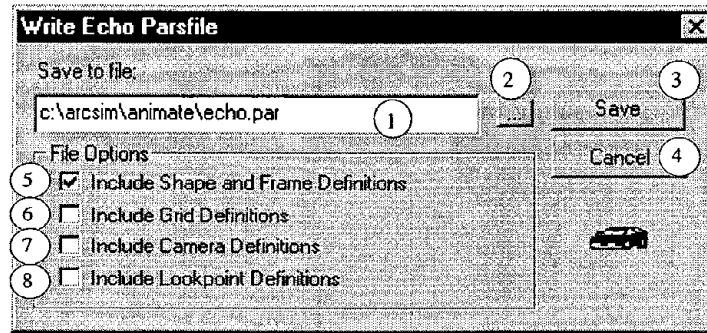
Besides restoring settings, this menu command can speed up the process of creating new shapes or other animator inputs, as described later in this chapter in the section **Testing New Shapes and Animator Data Sets**.

Save Parsfile As

Select this command to save most of the animator settings. The animator data from the VDM RoAD SGUI are spread over the five animator libraries described in Chapter 8. Multiple shapes and reference frames are commonly used, which means that the animator program might read 20 or more PARSEFILES when it runs. This menu command creates a single PARSEFILE with all of the information consolidated.

The PARSEFILE can be used later as an input to the animator, to exactly recreate the current settings. The file can also be viewed in a text editor to debug inputs that cause problems, or to obtain camera and look-point coordinates that were set interactively using the **Coordinates** menu.

When you select the command, it displays the following dialog box. The numbered items in the dialog box are described below.

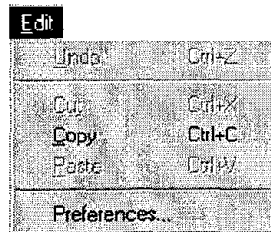


- ① Pathname for file to be created. The default is `echo.par`, in the same folder as the animator program. You can type a new pathname in this field or use the browse button ②.
- ② Browse button. Click to bring up the Windows file browse dialog to search your directory system for a suitable folder. If you select a folder using the browse dialog, the resulting pathname is automatically entered into the file field ①.
- ③ Save button. Click to create the Parsfile.
- ④ Cancel button. Click to exit the dialog without creating a file.
- ⑤ Check box for shape and frame definitions. If checked when you click the Save button, all vehicle shape and frame data will be included in the file.
- ⑥ Check box for grid definitions. If checked when you click the Save button, the current grid information will be included in the file.
- ⑦ Check box for Camera definitions. If checked when you click the Save button, the current camera reference frame and point coordinates will be included in the file.
- ⑧ Check box for Look-point definitions. If checked when you click the Save button, the current look-point reference frame and coordinates will be included in the file.

Exit — exits the animator program. To use the keyboard, type the Windows-standard sequence: Alt-F, X.

Edit Menu

The **Edit** menu exists to support the clipboard and to set preferences. When a dialog box is in front, the **Edit** menu cannot be used with the mouse. However, the keyboard commands are functional and can be used to cut, copy, and paste within text fields.



Undo

The keyboard command (Ctrl-Z) undoes the most recent entry in a dialog box.

Cut

The keyboard command (Ctrl-X) cuts currently selected text to the clipboard. This item is only in effect when editing text in dialog boxes. Modification of the graphic display is not supported.

Copy

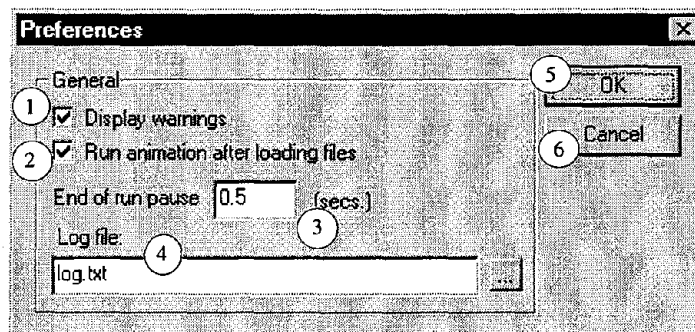
Select this command to copy the screen display to the clipboard as a bitmap. If a dialog box is displayed, the keyboard command (Ctrl-C) copies the currently selected text to the clipboard.

Paste

The keyboard command (Ctrl-V) pastes text from the clipboard into the current cursor location. If text is selected, it is replaced by the contents of the clipboard. This item is only in effect when editing text in dialog boxes.


Preferences...

Select this command to set a few options for the animator that are saved when you quit the program. The preferences are stored in a file `Prefs.txt`, described in Appendix C.



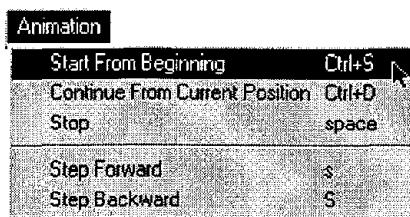
The numbered items in the dialog box are described below.

- ① Check box to display warnings. When checked, the animator will print messages to indicate errors in the input files.

- ② Check box to run after loading files. When checked, the animator starts running as soon as the files have been read. If the box is not checked, the animation must be started by typing Ctrl-S or using the **Animation** menu.
- ③ End of run pause. The animator delays this amount of time at the end of a run before looping to start over. A pause of 0.5 sec or more gives a visual indication that the run has ended.
- ④ Log file. Enter a name for an optional file written by the animator as it processes the inputs (use the adjacent button  to select a destination folder). The log file is sometimes helpful for debugging, but it is not needed for normal operation.
- ⑤ OK button. Click to apply the new preferences.
- ⑥ Cancel button. Click this to close the dialog box without changing anything (keep the previous preferences).

Animation Menu

The animation menu exists mainly to remind you of keyboard commands that control animation. Although the menu items are functional, you will usually find it quicker to press the indicated keys.



Start From Beginning

Select this command to restart the animation from the beginning of the simulation run. The keyboard command is Ctrl-S.

Continue From Current Position

Select this command to continue the animation if it has been stopped. The keyboard command is Ctrl-D.

Stop

Select this command to stop (pause). This is like being able to freeze time. While stopped, you can still use the **Coordinates** menu to move the camera around, zoom in and out, and change the look point. The keyboard command is the space key.

Step Forward

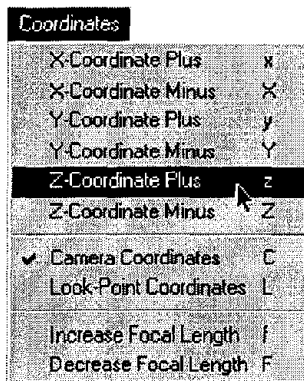
Select this command to step forward one increment in time, like a frame advance on a VCR. The keyboard command is the 's' key.

Step Backward

Select this command to step backward one increment in time. The keyboard command is Shift-S.

Coordinates Menu

This menu exists mainly to remind you of keyboard commands that control the camera viewpoint. Although the menu items are functional, you will usually find it quicker to press the indicated keys.



The first six items are used to change the position of either the camera point or the look point, depending on which is checked further down in the menu. The currently affected point is checked in the menu, and is written in the animator status bar (at the bottom of the animator window).

The camera and look points are shown in Figure 5.1 on page 54. The camera point defines where the camera is located, and the look point defines where the camera is looking. For example, consider changing the Z coordinate (height) of either point. Increasing the Z coordinate of the camera has the effect of raising the camera, as if you are viewing the vehicle from a helicopter that rises. On the other hand, increasing the Z coordinate of the look point has the effect looking up. When you look up, everything in view shifts down.

X-Coordinate Plus

Select this command to increase the X coordinate of the camera or look point. The keyboard command is 'x' key.

X-Coordinate Minus

Select this command to decrease the X coordinate of the camera or look point. The keyboard command is Shift-X.

Note: The keyboard commands are assigned so plain key commands increase things, and using the shift key decreases things. For example, 's' steps forward, Shift-S (S) steps backward; 'x' increases an X coordinate, Shift-X (X) decreases it.

Y-Coordinate Plus

Select this command to increase the Y coordinate of the camera or look point. The keyboard command is the 'y' key.

Y-Coordinate Minus

Select this command to decrease the Y coordinate of the camera or look point. The keyboard command is Shift-Y.

Z-Coordinate Plus

Select this command to increase the Z coordinate of the camera or look point. The keyboard command is the 'z' key.

Z-Coordinate Minus

Select this command to decrease the Z coordinate of the camera or look point. The keyboard command is Shift-Z.

Camera Coordinates

Select this command to cause the camera position to be controlled by the first six menu items. The keyboard command is the C key.

Look-Point Coordinates

Select this command to cause the loop point position to be controlled by the first six menu items. The keyboard command is the L key.

Increase Focal Length

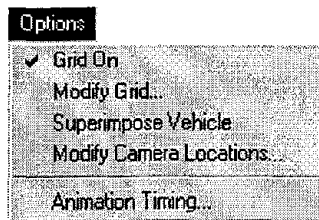
Select this command to increase the camera focal length. This is the same as zooming in. The keyboard command is the 'f' key.

Decrease Focal Length

Select this command to increase the camera focal length. This is the same as zooming out. The keyboard command is Shift-F.

Options Menu

This menu has commands for setting several options in the animator operation.



Grid On

Select this command to toggle the grid display.

Modify Grid

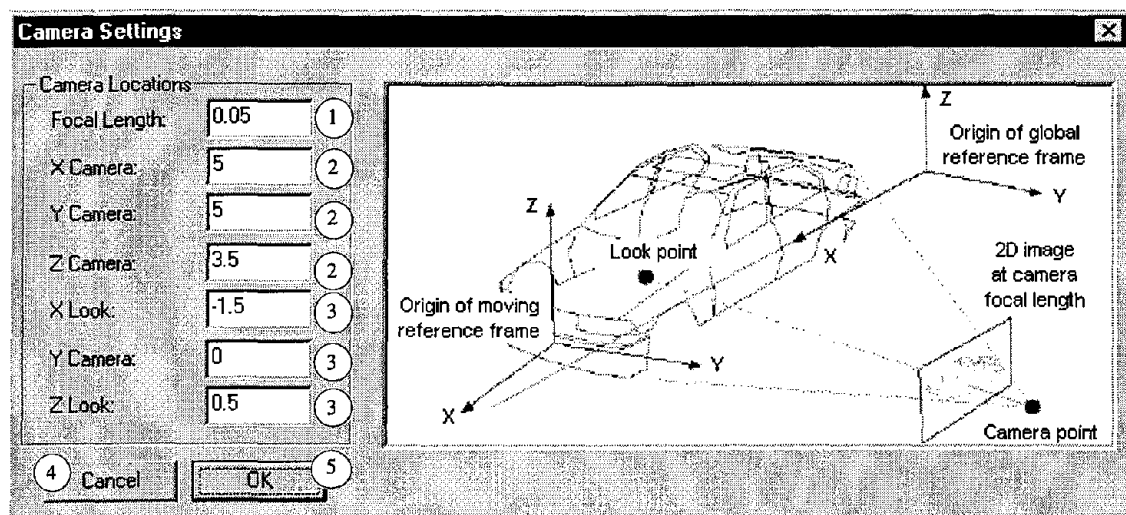
Select this command to display a dialog box for setting properties of the grid.

Superimpose Vehicle

Select this command to toggle an option to draw new images without erasing old ones. This mode is useful with some configurations to show trajectories of vehicle motion, or to show amplitudes of oscillations.

Modify Camera Locations

Select this command to display the following dialog. The numbered items are described below.



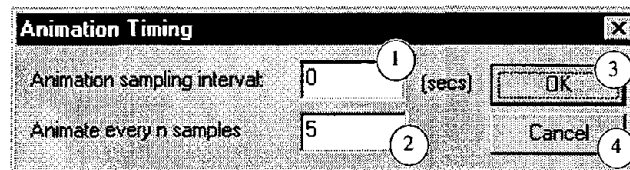
Notes: These settings are similar to those contained in the **Animator Camera Setup** library, described in Chapter 8.

Units of length should be compatible with coordinates used for shapes and reference as described in the section **Units** on page 56.

- ① Focal length. This is the distance from the camera point to the 2D plane used to project the 3D objects. See Figure 5.1 for the geometry.
- ② X, Y, and Z coordinates of camera point, in its reference frame.
- ③ X, Y, and Z coordinates of look point, in its reference frame.
- ④ Cancel button. Click to exit the dialog box without changing the camera settings.
- ⑤ OK button. Click to exit the dialog box and apply the new settings.

Timing

Select this command to display the following dialog and control the rate at which the animator can play back a simulation run. The numbered items are described below.



- ① Sampling interval. The animator normally obtains the time interval between stored data points from the ERD file header. However, it is possible for an ERD file to lack this information. In that case, the animator can be provided an interval in this field.

Note: The programs included in VDM RoAD all provide the sample interval. This setting is only needed when viewing ERD files made from other programs.

- ② Animate every n samples. This number is normally 1: the animator draws a picture for every set of variables in the file. If the sample interval in the file is very small, or you are running on a slower computer, you can put a number larger than 1 in this field to speed up the animator. The animator will not run faster than real time. If necessary, it waits between frames to maintain the proper relation between animation time and real time.
- ③ OK button. Click to exit the dialog box and apply the new settings.
- ④ Cancel button. Click to exit the dialog box without changing the camera settings.

Help Menu

This menu is standard in Windows for providing information about a program.



About Engineering Animator

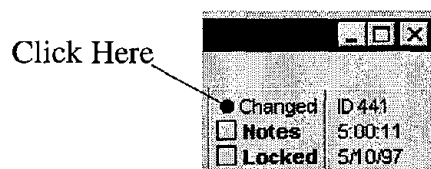
Select this option to display the current animator version and creation date.

Testing New Shapes and Animator Data Sets

Because the animation setup information is contained in the VDM RoAD data base, but displayed by the animator program, it is helpful to know how to use both the data screens and the animator to test and debug new animator settings.

The basic method for editing shapes and testing animator data sets is as follows:

1. Create animator settings using the five animator libraries described in Chapter 8.
2. Link to the new settings in the **Runs** screen (via the **Animator Setup** link) and click the **Animate** button.
3. View the animation. When you are ready to modify the settings, stop the animation. (press the space key or use the **Stop** command from the **Animation** menu).
4. Without quitting the animator, switch back to the VDM RoAD **Runs** screen. (Use Alt-Tab in Windows to switch between running programs.)
5. Within the ToolBook part of VDM RoAD, navigate to the screen with the animator data to be edited (e.g., an animator shape data set).
6. Edit the data in the VDM RoAD data set. As soon as you modify any data, you will see a red circle in the upper-right corner of the window.



7. Click on the red circle in the upper-right corner. This causes VDM RoAD to update the PARSEFILE using the new data from the screen.

Note: Normally, communication between the data base and the animator is transparent. All necessary files are properly updated when you

click the **Animate** button in the **Runs** screen. Here, you are bypassing the normal operation, and must force the SGUI to update the PARSEFILE before it would normally do so.

8. Switch back to the animator (use Alt-Tab).
9. Select the menu item **Reload Current Files** under the **File** menu (or type Ctrl-R) to see the effect of the new data.
10. Optionally switch back to the SGUI screen and repeat steps 6 through 9 as needed.

6. The Plotter

You view plots of the VDM RoAD simulation results with a program called Windows Engineering Plotter (WinEP). WinEP is a versatile tool you can use to plot any two variables against each other and overlay data from the same file or different files.

Although WinEP is designed for close integration with the other parts of VDM RoAD, it also functions as a stand-alone program. The University of Michigan and Mechanical Simulation Corporation place no restrictions on the copying and use of WinEP, so feel free to use it for applications outside of VDM RoAD.

Updates

The plotting program is used in a number of software packages. The software was developed by UMTRI but is now maintained by MSC. The latest version is available from the Internet at:

<http://www.trucksim.com/winep.html>

If you download a newer version, simply replace the file `winep.exe` in the Plot folder in VDM RoAD.

Overview of Operation

WinEP has a workspace defined by a main window with a menu bar. The main window contains plot windows, each showing plots made with a single set of X-Y axes. Figure 6.1 shows the main WinEP window with three example plot windows.

Each plot window contains a graphical representation of one or more X-Y data sets. An X-Y data set is a series of X and Y values obtained from a data file. The X-Y data sets can come from the same file or from different files. The X values in each data set do not have to be the same, and the data sets do not have to contain the same number of points.

WinEP reads the data from ERD and text files and has completely automated scaling, formatting, and labeling capabilities. However, labeling options are limited for non-ERD text files.

Note: If a file is not in ERD format, it can be converted by adding several lines of text to the beginning of the file. A description of the ERD file format is provided in Appendix A.

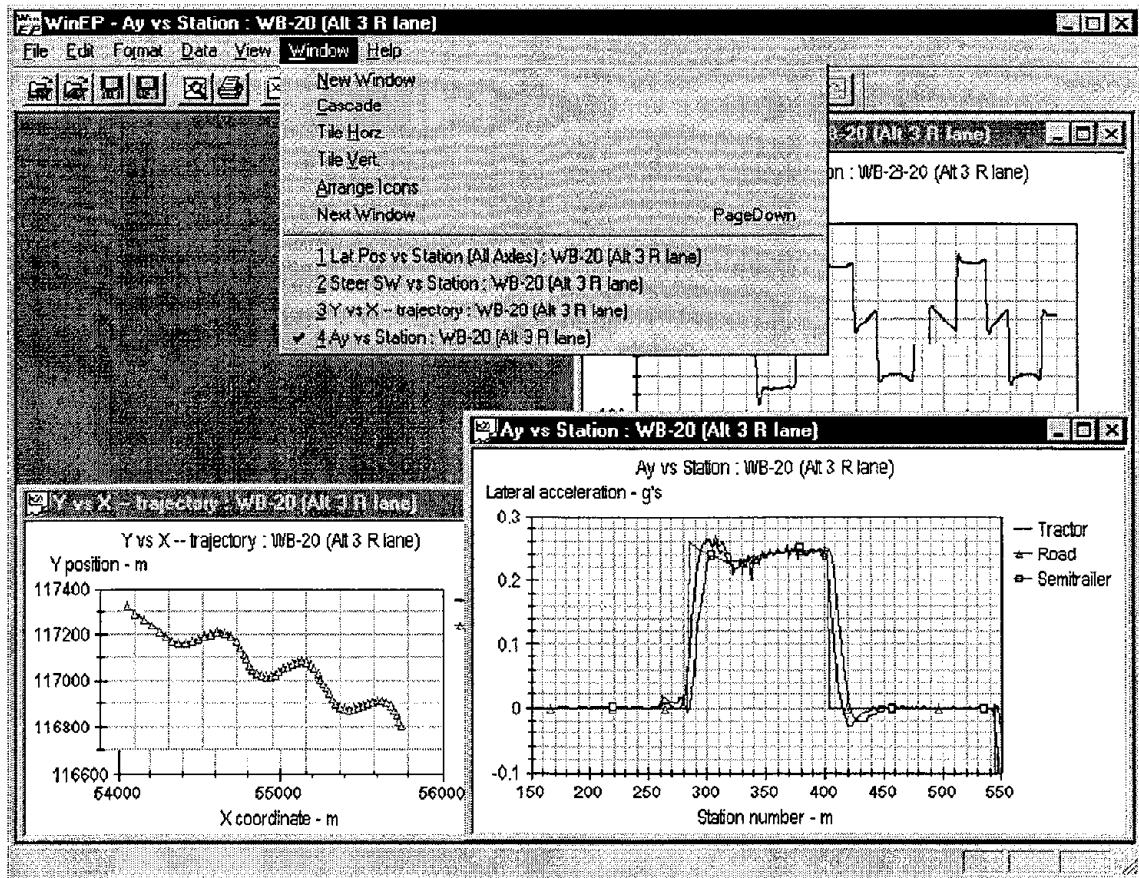


Figure 6.1. The WinEP workspace.

Once a plot window is created, the X-Y data sets can be transformed by subtracting offsets or by applying high-pass or low-pass filters. However, the data sets cannot be deleted or replaced. If you create a plot with different data sets, the new plot will have a new window.

In addition to the X-Y data, each plot window has a set of formatting attributes, such as the colors of the lines, the size of the labeling text, the type of grid, etc. You can show the data points with connecting lines, symbols, or both. You can use linear or log scaling and you can choose from several axis types. The formatting can be set independently for each plot, and changed interactively at any time.

You can copy the WinEP plots to the clipboard and send them to a printer.

There are no built-in limits to the number of active plot windows that can be maintained by WinEP. The only limits are imposed by computer memory.

Batch and Interactive Operation

WinEP can be started in one of two modes. When you click a **Plot** button from one of the VDM RoAD data screens, VDM RoAD starts WinEP in batch mode. It sends WinEP a file with information that you specified in the VDM RoAD data screen. Based on the contents of the file from VDM RoAD, WinEP initially creates one or more plots. Once the initial batch of plots is made, WinEP reverts to interactive mode. In the interactive mode you can view the plots, print them, and change their formatting. You can also create new plots.

You can also run WinEP as a normal Windows application. Start in this mode by double-clicking on the file `winep.exe`. When started in this way, the WinEP workspace is empty and you must use a menu command to define the data for the first plot. A better way to start is by double-clicking on a file with the extension ERD. This starts the WinEP program and automatically loads the selected ERD file. You must then pick X and Y variables from the file to create the first plot.

Note: The file `winep.exe` must be associated with the file type ERD if you want to launch it by double-clicking on an ERD file. The association is made automatically as part of the VDM RoAD installation. You would need to re-establish the association only if you move the VDM RoAD directory, or if you copy the WinEP program to another computer that does not have VDM RoAD on it.

Appendix B describes the batch control file for WinEP, created automatically by VDM RoAD, and the keywords that are recognized by WinEP.

The rest of this chapter covers the interactive use of WinEP.

Zooming

WinEP usually creates each plot window with the axes scaled to show all data points. To zoom in, use the mouse to create a zoom rectangle. Press the left button to anchor one corner at the current cursor location, then drag with the button still pressed to define the rectangle. When the rectangle covers the desired range, release the button. WinEP will re-draw the plot using the region of the rectangle as the new limits for the axes.

Figure 6.2 shows the appearance of a plot with a zoom rectangle drawn, just before the mouse button is released. Figure 6.3 shows the same plot window immediately after the button is released.

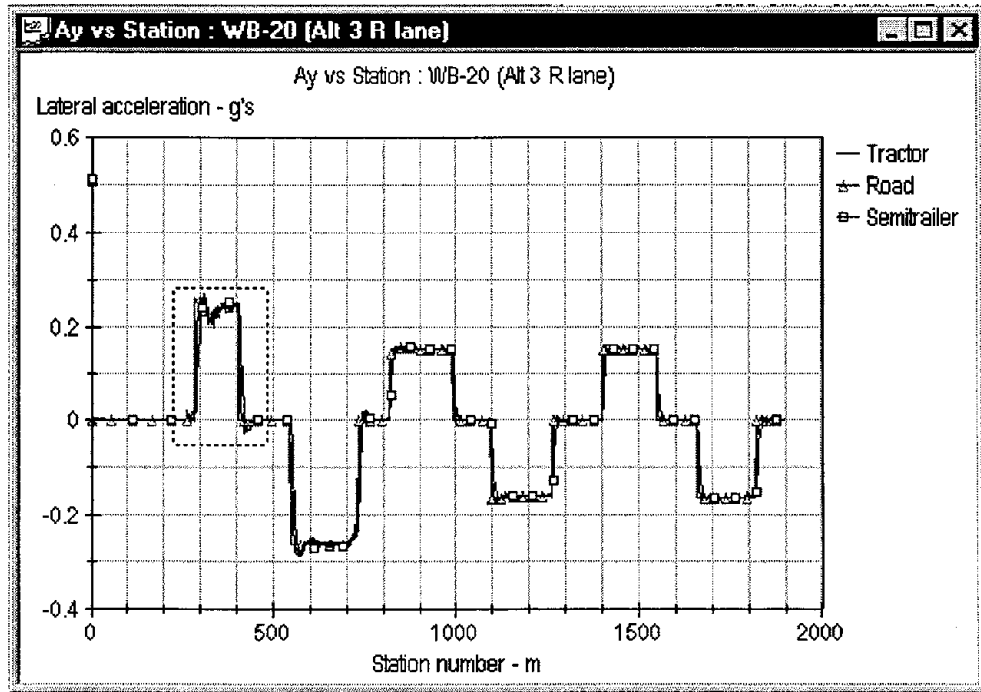


Figure 6.2. Plot before zooming.

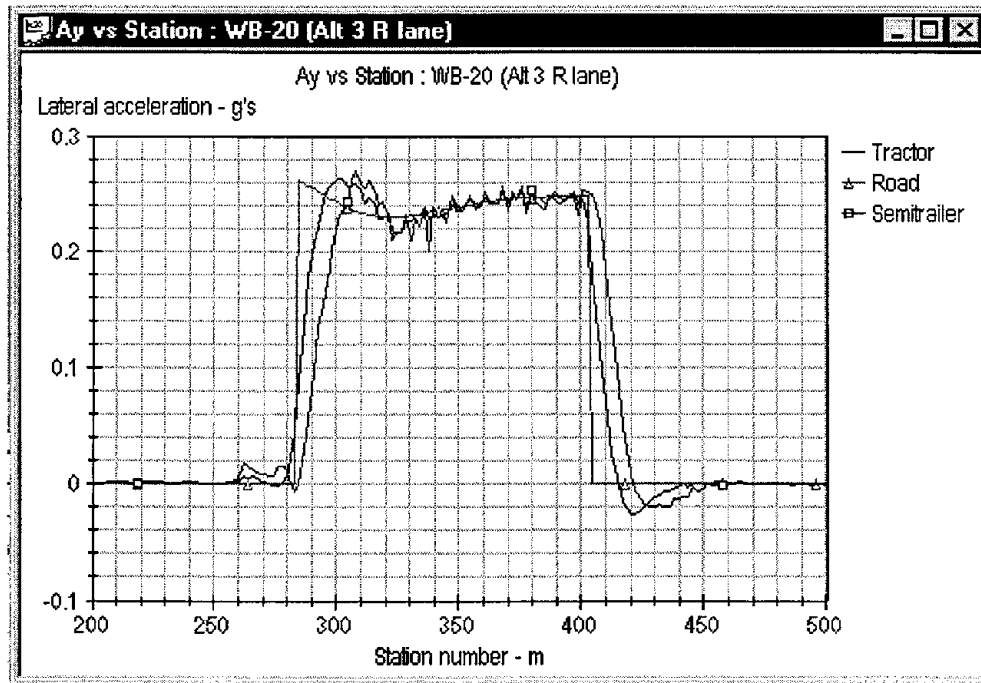


Figure 6.3. Plot after zooming.

Note: WinEP rounds off the maximum and minimum values for the axes. The new plot is drawn to include the full range of the rectangle you drew with the mouse, plus some additional range to round off the axes. For example, the maximum X value from the zoom rectangle shown in Figure 6.2 is about 480. The maximum X value in the zoomed plot of Figure 6.3 is rounded up to 500.

If you click in a plot window without drawing a rectangle, WinEP re-draws the screen with the same scale as before.

File Menu

The file menu is used to read files, write files, print windows, and exit the program.

File	
Load ERD File...	Ctrl+E
Load Batch File...	Ctrl+B
Load Plot Format...	Ctrl+D
Close Top Window	Ctrl+W
Save Plot Format...	Ctrl+S
Save Plot Data...	
Save Channel List...	
Print...	Ctrl+P
Print Preview	
Print Setup...	
Exit	Ctrl+Q

Load ERD File...

Select this command to open an ERD file with the Windows file browse dialog. After the file is opened, a dialog box appears that you use for selecting variables from the file to plot. The keyboard command is Ctrl-E.

Load Batch Plot File...

Select this command to process a batch plot file with the Windows file browse dialog. When created automatically by VDM RoAD, batch plot files have the extension PLT and are located in the directory containing the library where the Plot button was clicked (either `Runs` or `Batch\Plot_bat`). When loaded, the batch file will cause WinEP to generate one or more plots. The keyboard command is Ctrl-B.

Load Plot Format...

Formatting information for a plot window can be stored in a file. These files can be created from within WinEP using the **Save Plot Format** command (from this menu) and also from within the VDM RoAD data base (see the section **Plot Format** in Chapter 8).

Select the **Load Plot Format** command to load one of these format files. This command brings up the Windows file browse dialog to open a file with the extension PAR. When loaded, WinEP will redraw the current plot with whatever new formatting information was loaded. The keyboard command is Ctrl-O.

Note: Format files created within the VDM RoAD graphical user interface are in the VDM RoAD folder Plot\Format.

Close Top Window

This closes the top (active) plot window. The keyboard command is Ctrl-W.

Save Plot Format

Use this command to save the current plot formatting information. The default extension is PAR. The formatting can be applied in the future using the **Load Plot Format** command. When loaded, the formatting file will cause WinEP to redraw the current plot with the new format. The keyboard command is Ctrl-S.

Save Plot Data

Use this command to save the current X-Y data set in a text file. It is a handy way to extract data from an ERD file, two channels at a time.

Save Channel List

This command initiates a two-step process for creating a text file listing all of the channels in an ERD file. The steps are:

1. WinEP brings up the Windows file browser dialog for choosing an ERD file. Select the ERD file that will be scanned to obtain a list of the channels (variables) that can be plotted.
2. WinEP brings up a dialog for saving a text file with the list of channels. A list from an example save file is shown below.

The number starting in the first column is the channel number, the text immediately after the hyphen is the name of the variable as it appear in lists within WinEP (see the menu command **Define New Plot** on page 82), and the text immediately after the colon is a short description of this variable with its units.

Two complete channel list files are printed in Appendix J.

```

File: C:\VDM_ROAD\RUNS\541.ERD
File generated by WinEP: 11/21/97 10:17:32

Channel Names
1. AVz      : Vehicle yaw rate - deg/s
2. Ax      : Vehicle longitudinal accel. - g's
3. Ay      : Vehicle lateral accel. - g's
4. Ay_road : Input (road) lateral accel. - g's
5. Az      : Vehicle vertical accel. - g's
6. BetaAero: Aerodynamic slip angle - deg
7. dZ/dL_LF: LF road cross slope at wheel - -
8. dZ/dL_LR: LR road cross slope at wheel - -
9. dZ/dL_RF: RF road cross slope at wheel - -
10. dZ/dL_RR: RR road cross slope at wheel - -
11. dZ/dS_LF: LF road long. slope at wheel - -
12. dZ/dS_LR: LR road long. slope at wheel - -
13. dZ/dS_RF: RF road long. slope at wheel - -
14. dZ/dS_RR: RR road long. slope at wheel - -
15. Fxaero  : Longitudinal aerodynamic force - N
16. Fx_LF   : LF tire longitudinal force - N
17. Fx_LR   : LR tire longitudinal force - N
18. Fx_RF   : RF tire longitudinal force - N
19. Fx_RR   : RR tire longitudinal force - N
20. Fyaero  : Lateral aerodynamic force - N
21. Fy_LF   : LF tire lateral force - N
22. Fy_LR   : LR tire lateral force - N
...

```

Print

Use this command to make a hard copy of the active plot using the current Windows printer. The keyboard command is Ctrl-P.

Print Preview

This command opens a window showing how the active plot would be printed using the current print settings.

Print Setup

This command brings up the Windows dialog box for changing printer information.

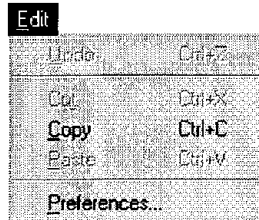
Exit

Use this command to exit WinEP. To use the keyboard, type the keyboard command Ctrl-Q, or the Windows-standard sequence: Alt-F, X.

Note: Alternatively, you can click the in the upper-right hand corner of this screen to return to the **Runs** screen.

Edit Menu

The **Edit** menu has commands to support the clipboard and to set preferences. When a dialog box is in front, the **Edit** menu cannot be used with the mouse. However, the keyboard commands are functional and can be used to cut, copy, and paste within text fields



Undo

The keyboard command (Ctrl-Z) undoes the most recent entry in a dialog box.

Cut

The keyboard command (Ctrl-X) cuts currently selected text to the clipboard. This item is only in effect when editing text in dialog boxes. Modification of the graphic display is not supported.

Copy

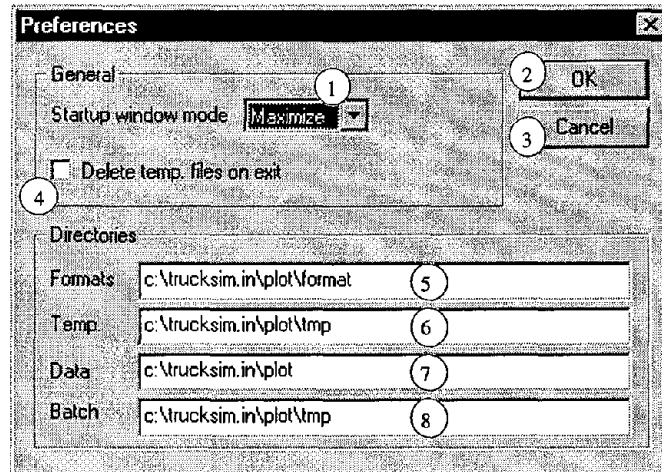
This command copies the plot from the active plot window to the clipboard. The graphic is copied as a bitmap. If a dialog box is active, this command copies currently selected text to the clipboard. The keyboard command is Ctrl-C.

Paste

The keyboard command (Ctrl-V) pastes text from the clipboard into the current cursor location. If text is selected, it is replaced by the contents of the clipboard. This item is only in effect when editing text in dialog boxes.

Preferences

This command brings up the **Preferences** dialog box. When you quit WinEP, the settings from this dialog box are automatically saved to a file named `epprefs.txt`.



The numbered items in the dialog box are described below.

- ① Startup mode. This controls the initial appearance of the plot windows when started in batch mode. For the Cascade setting, the individual plot windows are smaller than the WinEP workspace. For the Maximize setting, the active plot window is maximized to fill the WinEP workspace. For the Tile settings, multiple plots are tiled to fill the WinEP workspace.
- ② OK button. Click this to close the dialog box and apply the new preferences.
- ③ Cancel button. Click this to close the dialog box without changing anything (keep the previous preferences).
- ④ WinEP creates temporary files for certain operations. Check this box to cause WinEP to delete those files when it quits.
- ⑤ Folder where last format file was read. The next time the **Load Plot Format** command (from the **File** menu) is used, WinEP will open the file browser starting with this folder.
- ⑥ Folder where WinEP creates temporary files.
- ⑦ Folder where the last ERD file (with numerical data to plot) was read. The next time the **Load ERD File** command (from the **File** menu) is used, WinEP will open the file browser starting with this folder.
- ⑧ Folder where the last batch file was read. The next time the **Load Batch File** command (from the **File** menu) is used, WinEP will open the file browser starting with this folder.

Format Menu

This menu has commands for controlling the appearance of the plots made.

Format	
Log X Axis	Ctrl+H
Log Y Axis	Ctrl+G
Redraw Current Scale	
Redraw Original Scale	Ctrl+R
Customize Plot Format...	Ctrl+F
Labels...	Ctrl+L

Log X Axis

Use this command to toggle between linear and log scaling for the X axis of the active plot. If the log axis is selected, the menu item has a check mark. The keyboard command is Ctrl-H.

Note: WinEP will ignore the X axis log option and revert to the linear option if there are one or more zero X values in the data set being plotted, even if it is out of range.

Log Y Axis

Use this command to toggle between linear and log scaling for the Y axis of the active plot. If the log axis is selected, the menu item has a check mark. The keyboard command is Ctrl-G.

Note: WinEP will ignore the Y axis log option and revert to the linear option if there are one or more zero Y values in the data set being plotted, even if it is out of range.

Re-Draw Current Scale

Use this command to re-draw the active plot with the current formatting. It is sometimes needed if the window display gets corrupted by another window being in front. You can accomplish the same thing by clicking in the plot area with the mouse.

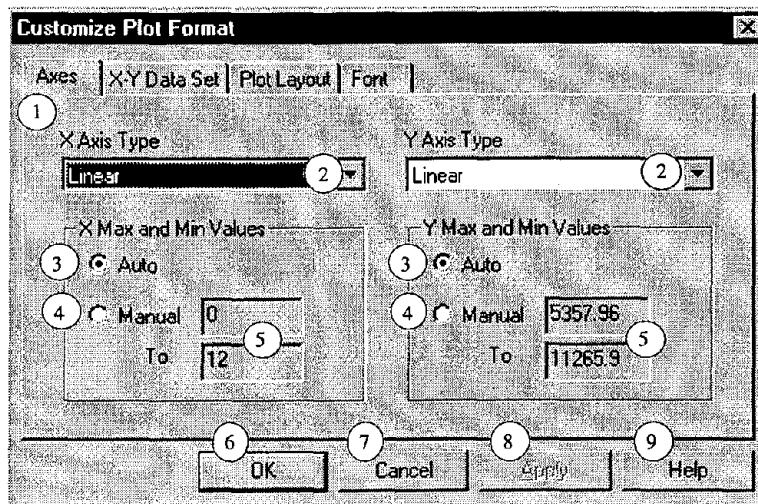
Re-Draw Original Scale

Use this command to re-draw the active plot without any zooming. The current scaling option is used, as specified with the **Customize Plot Format** option under this menu.

Customize Plot Format

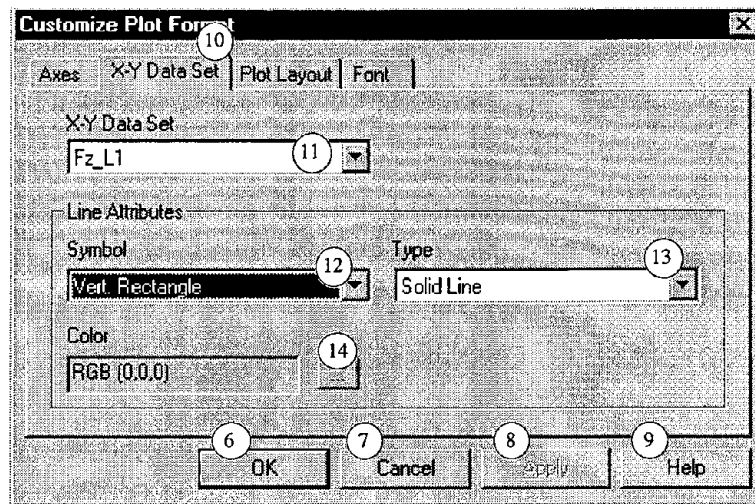
Use this command to display a preferences dialog box for setting most formatting options. The dialog is used for four sets of data, selected with tabs.

The first tab, Axes (1), controls the scaling of the X and Y axes.







You can use this dialog box to type in the desired ranges of X and Y values to be covered in the next plot.

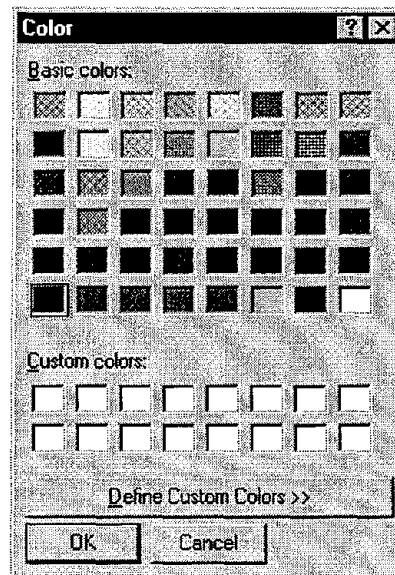
- ② X, Y axis type. These controls define whether the axis will be drawn with a linear or log scale. They duplicate the Log X and Log Y menu commands.
- ③ Autoscaling (X and Y axes). Check to cause WinEP to scan the data and size the axis to include the full range of data. This setting is recommended for routine use.
- ④ Manual scaling (X and Y axes). Check to scale the axis according to the range specified in the adjacent fields ⑤. This setting is provided for situations in which you wish to make many plots with the same axes, regardless of the range covered by the data.
- ⑤ Range of data covered by axis. Specify minimum and maximum values for scaling the axis. These are used only when the manual scaling button ④ is selected.
- ⑥ OK button. Click to close the dialog box and apply the settings as displayed.
- ⑦ Cancel button. Click to close the dialog box and revert to the previous settings.
- ⑧ Apply button. Click to apply the current settings without closing the dialog box. This button is not functional in the current release.
- ⑨ Help button. Click to display on-line help. This button is not functional in the current release.



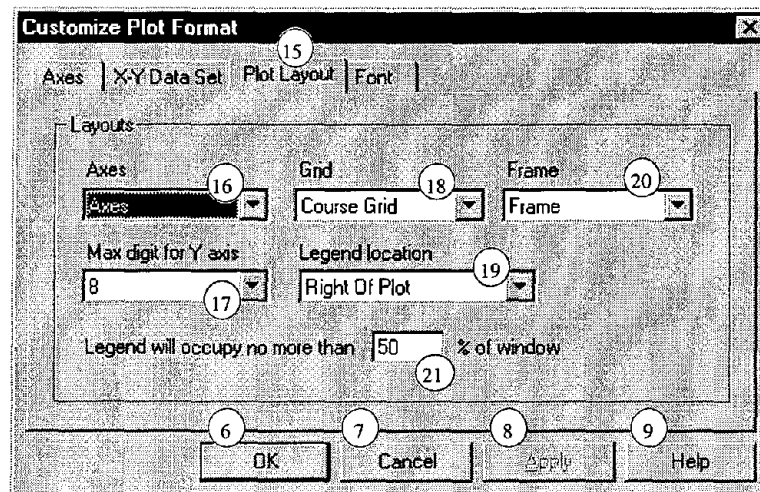
- ⑩ X-Y Data Set tab. Click to show the settings that control the display of a data set.

Note: An X-Y Data set is a set of X and Y values from a specific file. Each data set is represented graphically in a plot. When multiple data sets are shown in the same plot, they can be distinguished by assigning them different display attributes.

- ⑪ X-Y Data set. Press the button  to display a pull-down menu with the list of Y variables for each data set. Select a data set to display and possibly edit its attributes.
- ⑫ Symbol. The field shows the symbol currently associated with the data set ⑪. Press the button  to display a pull-down menu of possible symbols, and choose one to change the current symbol.
- ⑬ Line type. The field shows the type of line currently associated with the data set ⑪. Press the button  to display a pull-down menu of possible types, and choose one to change the way connecting lines are drawn between the data points.
- ⑭ Color. The field shows the color currently associated with the data set ⑪, in terms of RGB values (24-bit color, composed of components of Red, Green, and Blue). Press the button  to display a palette of colors.



After selecting a color, click to OK button to return to the format dialog box. The RGB values for the selected color will be shown.



- ⑮ Plot Layout tab. Click to show the settings that control the display of the axes, frame, and grid.
- ⑯ Axes. Press the button to display a menu with the choices: Axes or No Axes. The No Axes option doesn't work unless you also select No Grid ⑰ and No Frame ⑱.
- ⑰ Max digits for Y axis. This is the number of digits set aside for the tick labels of the Y axis. For example, if set to 3, WinEP draws the Y axis of every plot with enough space for 3-digit labels for the Y axis. If set to 0, WinEP determines the spacing automatically.

A value of 0 is recommended unless you want all plots to be drawn in exactly the same place relative to the left edge of the window.

- ⑱ Grid. Use this to choose between the options: No Grid, Course Grid (grid lines at major tick marks), or Fine Grid (grid lines at minor tick marks).
- ⑲ Legend location. Use this to choose among five possible locations for the legend used to identify the X-Y data sets in overlay plots. The options are Right of Plot, On Plot Upper-Left, On Plot Upper-Right, On Plot Lower-Left, and On Plot Lower-Right. The legend and other labels in the plot are shown below in Figure 6.4.

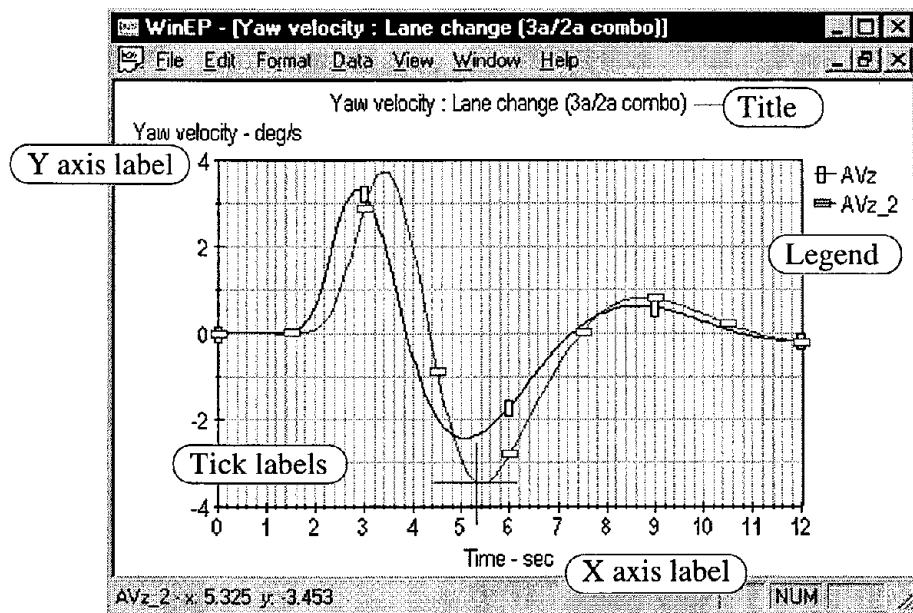
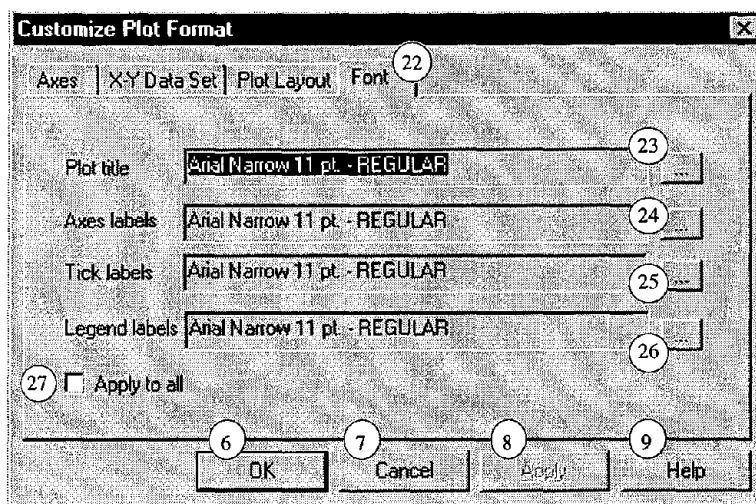



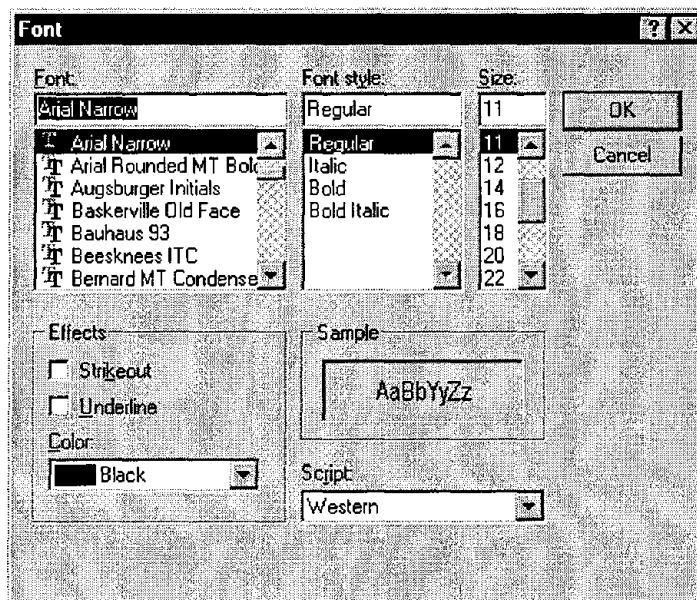
Figure 6.4. Names of plot labels.

- ⑳ Frame. Use this to determine whether the plot area has an enclosing rectangular frame. The options are No Frame, Frame, and Frame with ticks. The No Frame option doesn't work unless you also select No Grid ⑱.
- ㉑ Size limit for legend. When WinEP draws the legend for overlay plots to the right of the plot area, it uses whatever space is needed. If the labels are lengthy, or if the window is small, this can limit the horizontal space available for the plot. Use this field to set a maximum amount of the window space that can be taken for the legend.


This setting does not affect the plots unless: (1) there are at least two X-Y data sets being overlaid, (2) the specified position ⑲ is "Right of Plot," and (3) the length of the longest label exceeds the specified percentage.



- ② Font tab. Click to show the settings that control the text used to label the axes and data in the plot. The four types of labels, controlled with items ③ - ⑥, are shown in Figure 6.4.
- ③ Plot title font attributes. Click the selector button  to bring up a Windows font selection dialog box. The properties selected in the box will be applied to the plot title. After choosing the font properties, click the OK button to make the change, or click the Cancel button to keep the original properties.

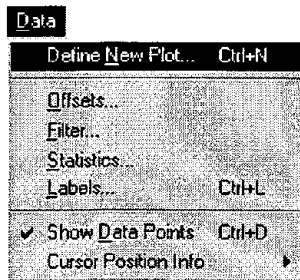


- ④ Axes labels. This is similar to the plot title button ③, except that the settings made using the Windows font dialog box are applied to the text used to label the X and Y axes.

- ②⑤ Axis numbers. This is similar to the plot title button ②③, except that the settings made using the Windows font dialog box are applied to the text used to number the X and Y axes.
- ②⑥ Legend labels. This is similar to the plot title button ②③, except that the settings made using the Windows font dialog box are applied to the text used to label the individual data sets in the legend.
- ②⑦ Apply to all check box. When this is checked, changing font properties using any of the buttons  (②③ ②④ ②⑤ ②⑥) will change all of the text in the plot.

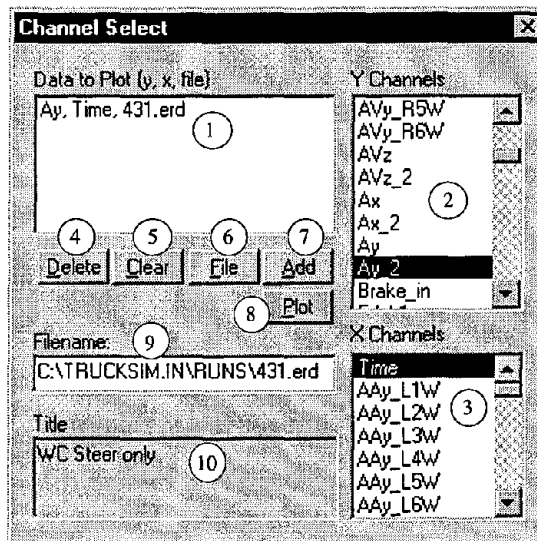
Data Menu

The **Data** menu has commands that involve the numbers to be plotted and the associated labels from the data files. The commands involve the selection of X-Y data sets, possible transformation of the numbers, labels for the plot, and the viewing of the numerical values using a cursor.



Define New Plot...

Select this command to bring up the **Channel Select** dialog box. The keyboard command is Ctrl-N. Use this dialog box to create a list of X-Y data sets to plot. Each data set consists of three parts: (1) a Y channel, (2) an X channel, and (3) a file name. The numbered items in the dialog box are described below.



- ① List of X-Y data sets to plot. This list shows the data sets that will be plotted. You cannot edit it directly, but can modify it using the buttons below.
- ② List of variables in the current ERD available for defining the Y coordinate in a plot. To select a variable for plotting on the Y axis, click on its name. Double-clicking in this list has the same effect as clicking the Add button ⑦.
- ③ List of variables in the current ERD available for defining the X coordinate in a plot. To select a variable for plotting on the X axis, click on its name. Double-clicking in this list has the same effect as clicking the Add button ⑦.
- ④ Delete button. Click this to delete a data set from the list of data to plot ①. This button has no effect unless a line is selected in the list ①.
- ⑤ Clear button. Click this to delete all data sets from the list of data to plot ①.
- ⑥ File button. Click this to bring up the Windows file browser dialog box to choose an ERD file or text file with plot data. This button has the same effect as selecting the command **Open ERD File** from the **File** menu.

Note: While the dialog box is displayed, the WinEP menus cannot be used. Therefore, the File button is the only way to open a new file.
- ⑦ Add button. Click this to add a new X-Y data set to the list of data to plot ①. The new data set has the currently selected Y channel ②, the currently selected X channel ③, and the currently open file (shown in ⑨).
- ⑧ Plot button. Click this to close the dialog box and make a new plot using the current list of X-Y data sets ①.
- ⑨ File name. This field shows the current data file. The name can be copied to the clipboard, but there is not a reason to edit it directly. To change the file, you must click the File button ⑥.

- ⑩ Title. This field shows the title from the data file, if (1) the file is an ERD file, and (2) the optional TITLE keyword is used in the ERD header. All ERD files created by VDM RoAD models show the title of the corresponding **Run** data set.

Offsets...

Select this command to bring up the **Data Offsets** dialog box. Use this dialog box to specify offsets for up to 20 data sets laterally (X) or vertically (Y). The values you enter in these fields will be subtracted from the values obtained from the data file. For any fields left blank, no offset is applied.

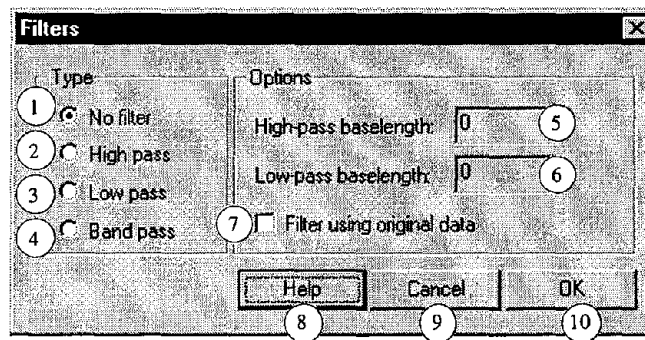
No.	X Offset	Y Offset	Org. Data	No.	X Offset	Y Offset	Org. Data
1	0	0	<input type="checkbox"/>	11	0	0	<input type="checkbox"/>
2	0	0	<input type="checkbox"/>	12	0	0	<input type="checkbox"/>
3	0	0	<input type="checkbox"/>	13	0	0	<input type="checkbox"/>
4	0	0	<input type="checkbox"/>	14	0	0	<input type="checkbox"/>
5	0	0	<input type="checkbox"/>	15	0	0	<input type="checkbox"/>
6	0	0	<input type="checkbox"/>	16	0	0	<input type="checkbox"/>
7	0	0	<input type="checkbox"/>	17	0	0	<input type="checkbox"/>
8	0	0	<input type="checkbox"/>	18	0	0	<input type="checkbox"/>
9	0	0	<input type="checkbox"/>	19	0	0	<input type="checkbox"/>
10	0	0	<input type="checkbox"/>	20	0	0	<input type="checkbox"/>

For each data set, there is a check box. Check the box to apply the offset to the original data (as stored in the file). If not checked, the offsets are applied to the data as shown in the last plot, which could potentially include previously applied offsets.

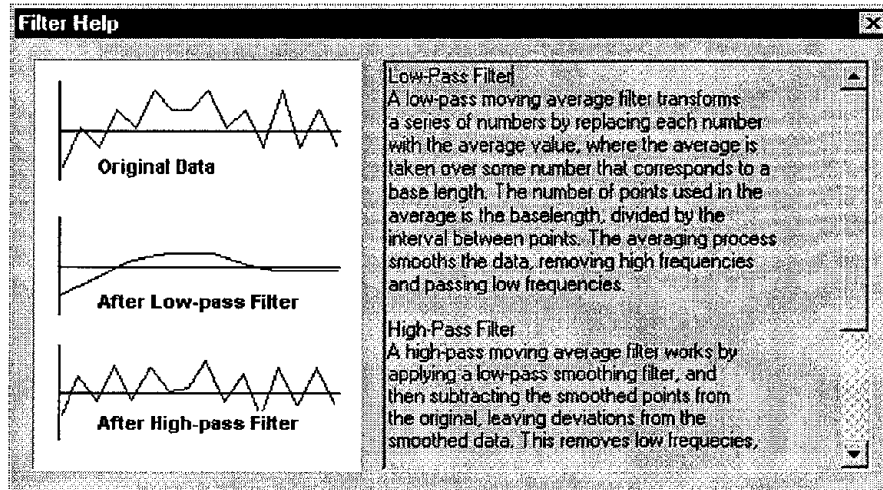
When you are finished specifying the offsets, click the **OK** button. These offsets will be applied to your active plot.

Filters...

Select this command to bring up the **Filters** dialog box. Use this dialog box to filter the Y values in the data sets of the active plot using a moving average. The numbered items in the dialog box are described below.



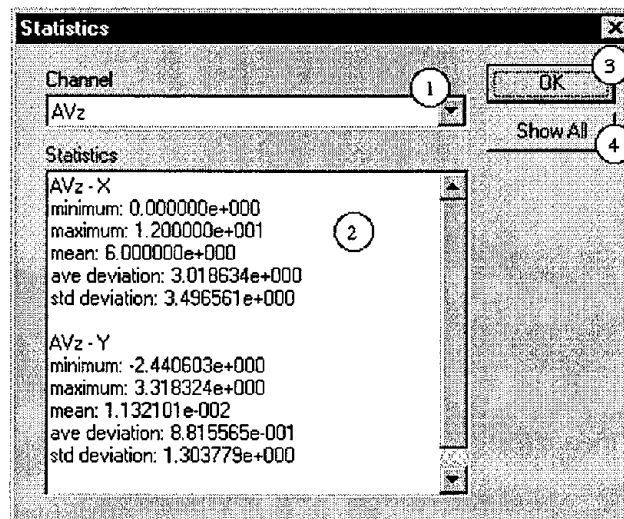
- ① No filter button. When this button is selected, the data are not modified.
- ② High pass button. When this button is selected, the Y variables are processed with a high-pass filter. The data are smoothed using the specified baselength ⑤, and the smoothed values are subtracted from the original. See the Help button ⑧ for details on how the filter works.
- ③ Low pass button. When this button is selected, the Y variables are processed with a low-pass filter. The data are smoothed using the specified baselength ⑥. See the Help button ⑧ for details on how the filter works.
- ④ Band pass button. When this button is selected, the Y variables are processed with a band-pass filter. The data are processed by a high-pass filter using the high-pass baselength ⑤, and then by a low-pass filter using the low-pass baselength ⑥. See the Help button ⑧ for details on how the filter works.
- ⑤ High-pass baselength. This is a reference used to define the scope of a moving average. It has the same units as the variable plotted on the X axis. For example, if the X variable is time with units of seconds, the baselength is the number of seconds covered by the moving average. This baselength is also used for band-pass filtering.
- ⑥ Low-pass baselength. This is a reference used to define the scope of a moving average. It has the same units as the variable plotted on the X axis. For example, if the X variable is time with units of seconds, the baselength is the number of seconds covered by the moving average. This baselength is also used for band-pass filtering.
- ⑦ Check box: Filter using original data. When this box is checked, the specified filters are applied to the original data — the values read from the file. When not checked, the filters are applied to the data as modified by previous filtering. If this box is not checked, the data are processed every time the filter dialog box is displayed and exited with the OK button.
- ⑧ Help button. Click this to display the following information.



- ⑨ Cancel button. Click this button to exit the dialog box without changing the data in the active plot.
- ⑩ OK button. Click this button to exit the dialog box and apply the specified filter to all Y variables in the active plot.

Statistics...

Select this command to display a dialog box with simple statistics for the plotted variables. The numbered items in the dialog box are described below.

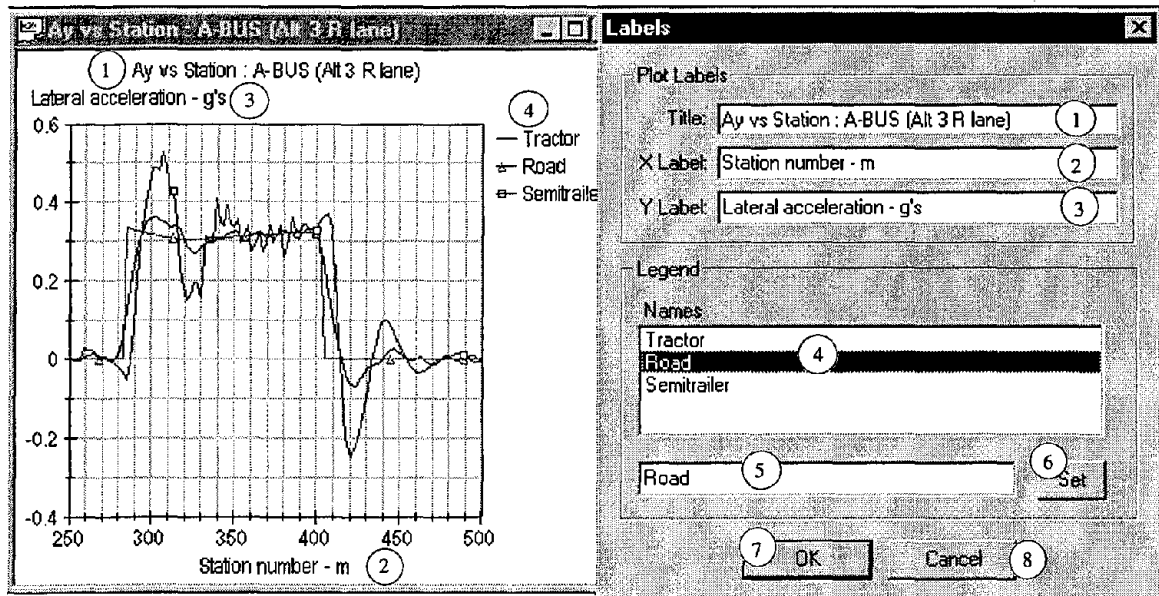


- ① X-Y data set name. Use the pull-down menu to pick the data set in the active plot whose statistics will be displayed.

- ② Statistics for specified data set. The field is sized to display the statistics for two variables making up a single X-Y data set. If you choose to show the statistics for all data sets in the plot, then the scroll bar can be used to view the statistics for the data sets after the first.
- ③ OK button. Click this button to exist the dialog box.
- ④ Show All button. Click this button to display the statistics of all data sets in the active plot.

Labels...

Use this command to display the **Labels** dialog box. Use this dialog box to edit the plot title, Y axis, X axis, and legend labels for the active plot. The next figure shows the Labels dialog box along with the active plot. The circled numbers appear both in the plot and the dialog box, to indicate the correspondence between the field in the dialog box and the plot.



- ① Title. The title is shown in the top of the plot and also in the title bar of the plot window. It can be edited in the dialog box.
- ② X Label. This is the label written underneath the X axis. It can be edited in the dialog box.
- ③ Y Label. This is the label written above the Y axis. It can be edited in the dialog box.
- ④ Names list, used to identify the X-Y data sets in the legend. To change any of these labels:

1. Select it from the Names list ④.

2. Edit the name as it appears in the edit field ⑤.
 3. Click the Set button ⑥ to apply the change.
- ⑤ Edit field used to change a name in the legend.
 - ⑥ Set button. Click to set the name selected in the Names list ④ to match the text in the edit field ⑤.
 - ⑦ OK button. Click to close the dialog and apply the new settings to the active plot.
 - ⑧ Cancel button. Click to close the dialog without modifying the active plot.

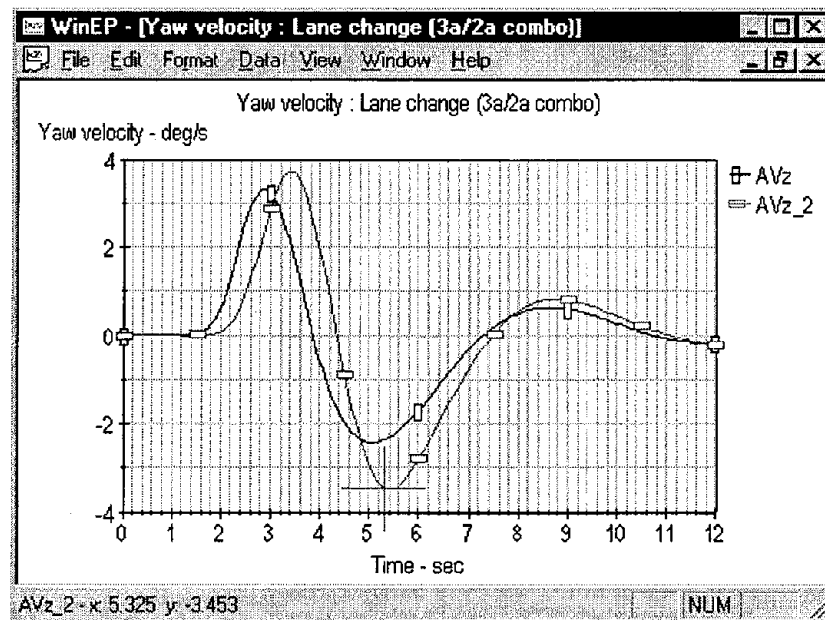
Show Data Points

This command toggles the mode of showing a cursor on the plot and the associated coordinates in the status bar. The keyboard command is Ctrl-D. A shortcut button is also available on the tool bar:



When you select this option, a cursor will appear at the first data point contained in the data file. For time history plots, the first point is at the far left side of the plot. However, for cross-plots, the first point may be located somewhere else.

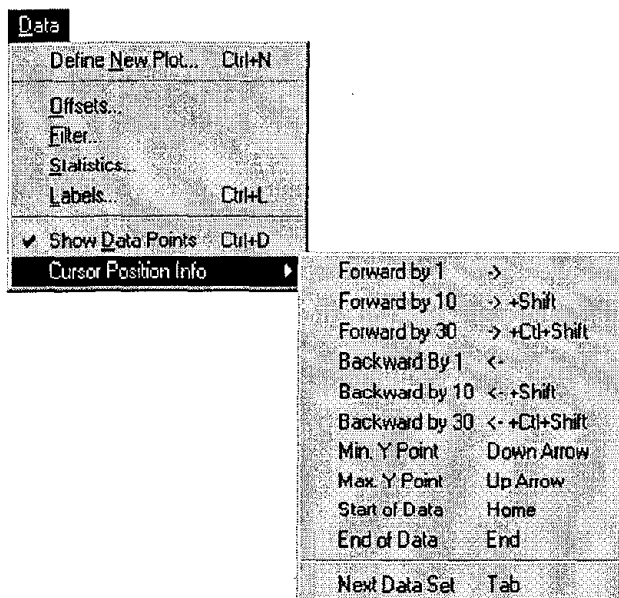
The X and Y values associated with the cursor are displayed in the status bar. For example, in the figure below, the cursor is at the point: time = 5.325 sec, Yaw velocity = -3.453 deg/s.



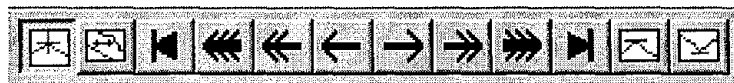
When there are multiple plot windows, the cursor and X-Y values are always associated with the active window.

Cursor Position Info

This menu item displays a submenu that exists mainly to remind you of keyboard commands that move the cursor. Although the menu items are functional, you will usually find it quicker to press the indicated keys.



All of the functions on the submenu are also available as buttons on the tool bar. The buttons are dimmed unless the option to Show Data Points is enabled. This is indicated visually in the toolbar by the button being “pushed in” as shown below.



Forward by 1 — use the right arrow key (→) to move one point forward, based on the order the X values are stored in the file. When the cursor is on the last point in the file, this command has no effect.



Forward by 10 — use the right arrow key (→) plus the Shift key to move 10 points forward, based on the order the X values are stored in the file. When the cursor is on the last point in the file, this command has no effect.



Forward by 30 — use the right arrow key (→) plus the Shift key plus the Control key to move 30 points forward, based on the order the X values are stored in the file. When the cursor is on the last point in the file, this command has no effect.



Backward by 1 — use the left arrow key (←) to move one point backward, based on the order the X values are stored in the file. When the cursor is on the first point in the file, this command has no effect.



Backward by 10 — use the left arrow key (←) to move 10 points backward, based on the order the X values are stored in the file. When the cursor is on the first point in the file, this command has no effect.



Backward by 30 — use the left arrow key (←) to move 30 points backward, based on the order the X values are stored in the file. When the cursor is on the first point in the file, this command has no effect.



Min Y Point — use the down arrow key (↓) to locate the point with the minimum Y value for the points currently displayed.



Max Y Point — use the up arrow key (↑) to locate the point with the maximum Y value for the points currently displayed.



Start of Data — use the Home key to find the first point. This is the same as pressing the left arrow repeatedly.



End of Data — use the End key to find the last point. This is the same as pressing the right arrow repeatedly.



Next Data Set — use the Tab key to move the cursor between currently displayed data sets in overlay plots. If the active plot has only a single data set, then this command has no effect.

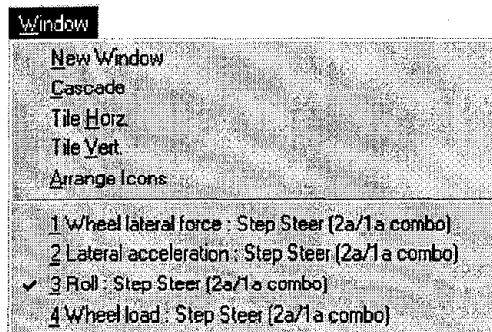
View Menu

Use this menu to toggle the display of the status bar, the tool bar, and a special debugging window that maintains and displays errors and warning messages.



Windows Menu

Use this menu to control the appearance of the plot windows in WinEP.

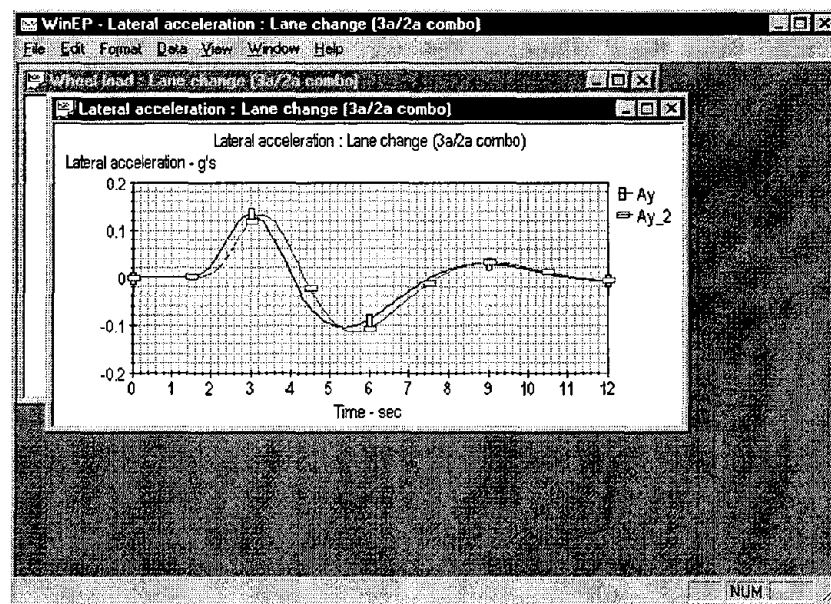


New Window

Use this command to copy the active plot window, along with the data and all format settings.

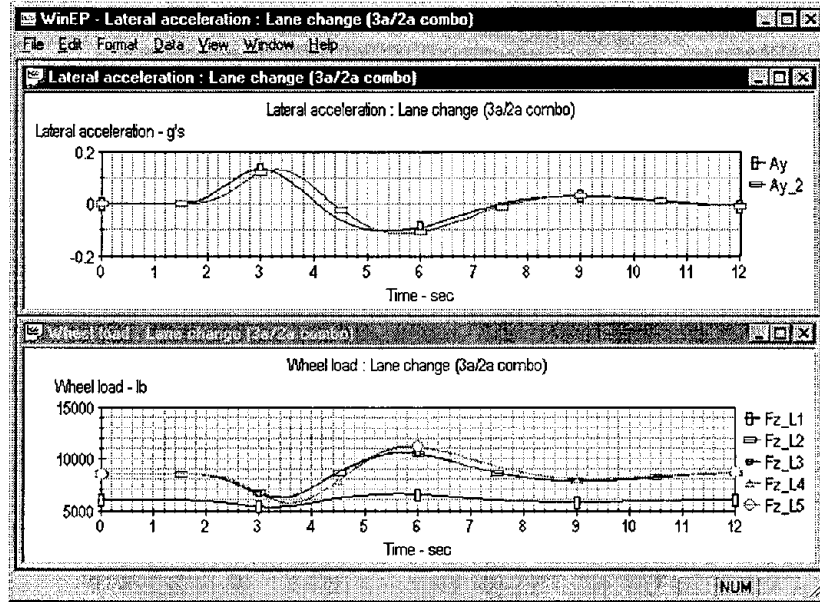
Cascade

Use this command to arrange all plot windows in a cascade, as shown below for two.



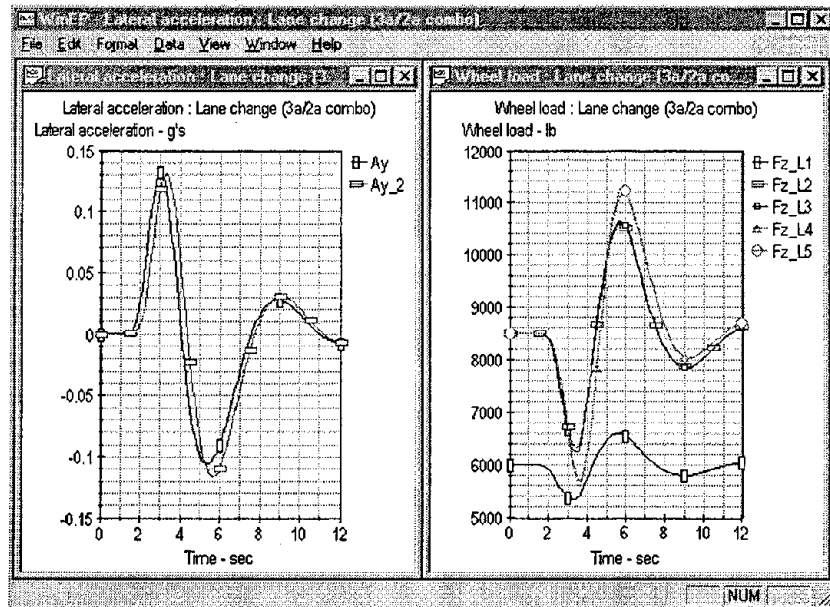
Tile Horz.

Use this command to tile the plot windows with horizontal divisions, as shown below for two. If there are four or more windows, they are arranged in columns and rows.



Tile Vert.

Use this command to tile the plot windows with vertical divisions, as shown below for two. If there are four or more windows, they are arranged in columns and rows.



Arrange Icons

Use this command to tidy any “iconized windows” in the WinEP workspace.

Help Menu

This menu has a command to open the **About WinEP** dialog box. Use it to obtain the version number and to obtain the current web site for updates.



Tool Bar

WinEP includes a toolbar with buttons for applying some of the most common commands. Place the cursor over a button to read its function in a pop-up box and also in the status bar.

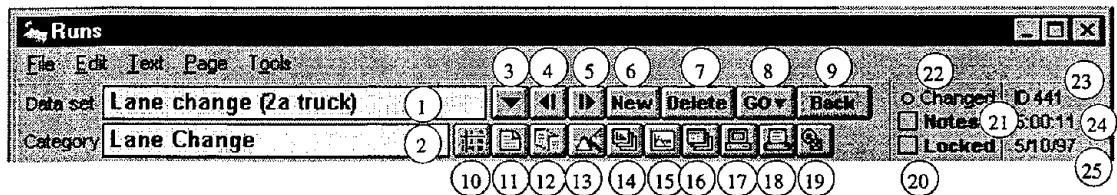


7. Design of VDM RoAD Data Screens

This section describes several standard elements in the VDM RoAD design that are shared in many of the data screens. All data screens have the same set of controls at the top in a ribbon bar. All have the same menu bar. Also, data screens with tabular data have a common layout.

The Ribbon Bar

Almost every screen in VDM RoAD has a standard ribbon bar at the top. It contains the name of the data sets, information about the last time the data set was modified, and a number of navigational controls.



Fields

- ① Data Set field. This field shows text entered by you, the user, to name the current data set. There are a few restrictions on titles: (1) each data set in the library must have a unique title; (2) the title cannot include a comma (the software will automatically replace it with a semicolon); and (3) the length of the title is limited to 32 characters.

Note: On the **Runs** screen this is where you give a unique name to the run of a specific vehicle, maneuver, and simulation model. In a vehicle screen, it typically serves to identify the vehicle and its properties, and so on for other screens.

- ② Category field. This field shows text you use to create a submenu for listing the contents of the library (see Figure 7.1 for an example). The Category field is provided as a convenience for you to divide large numbers of data sets into logical groups in pull-down menus. If the category field is blank, then the title is shown in the top-level menu.

Data sets in different categories are still subject to the restriction that they cannot have the same names.

Buttons

- ③ Library triangle button. Press this button to display a pull-down menu listing all of the data sets in the library, grouped by categories.

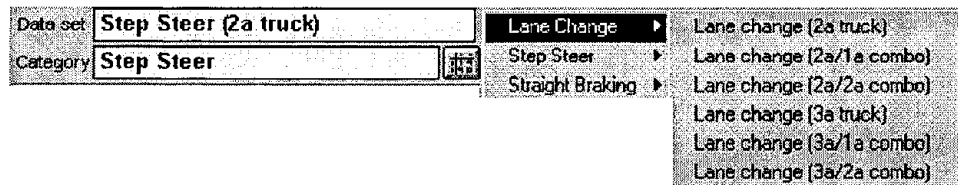


Figure 7.1. Pull-down menu by the Data Set field.

- ④ Library Left button. The data sets in the library are ordered alphabetically first by category and then by title. Click this button to go to the preceding data set. If the current data set is the first one, this button takes you to the end (the ordering is circular).
- ⑤ Library Right button. The data sets in the library are ordered alphabetically first by category and then by title. Click this button to go to the next data set. If the current data set is the last one, this button takes you to the beginning (the ordering is circular).
- ⑥ New button. Click this button to make a full copy of the data set. The copy is identical to the original, except that the name ① will be different (usually it will have a number appended to it). When the copy is made, the Data Set field will be highlighted to encourage you to type a distinctive title for the new data set.
- ⑦ Delete button. Click this button to delete the data set.

When you click the **Delete** button, two things happen: (1) the current data set as displayed on the screen is deleted, and (2) auxiliary files associated with the data set are also deleted. Thus, VDM RoAD automatically performs housecleaning.

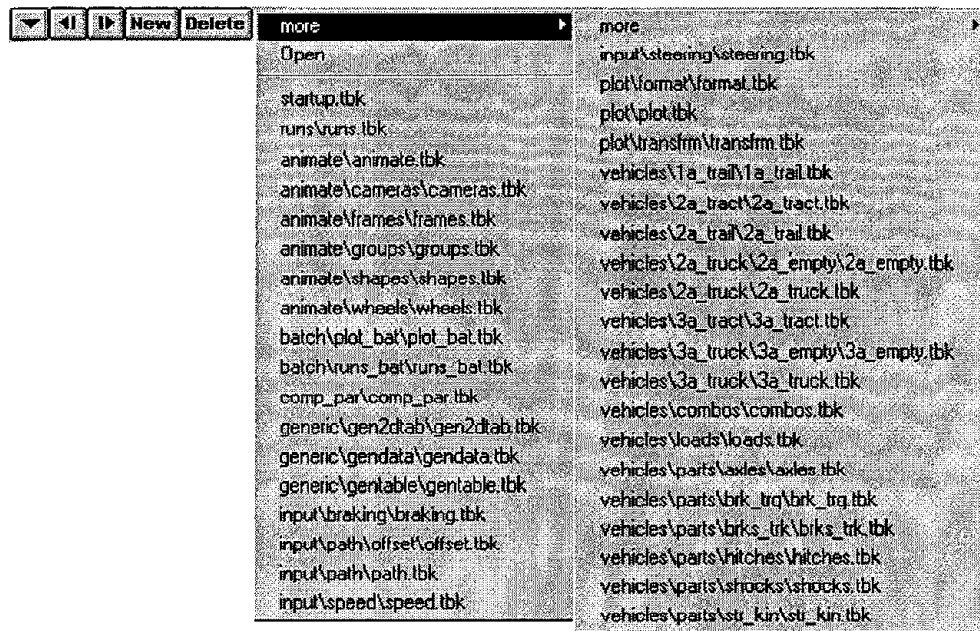
If the library has only one data set, the **Delete** button cannot be used — there must always be a minimum of one data set in a library. Also, the **Delete** button will not work on the current data set if the **Locked** box ②② is checked.

There are two keyboard modifiers for the **Delete** button:

- Press the **Ctrl** key when clicking the button to avoid the confirmation message.
- Press the **Shift** key when clicking the button to delete all data sets except the current one. There will be a confirmation message, giving you a chance to change your mind after clicking the button. Be aware that if you go ahead, other data sets can be deleted even if their **Locked** boxes are checked. Use this feature with care!

Note: When you position the cursor over the **Delete** button, the Status Bar at the bottom of the window describes the above two keyboard modifications.

- ⑧ GO button. Press this button to display a menu of all libraries in VDM RoAD. Select a library from this menu to leave the current library and go to the selected one.



The menu usually contains more items than will fit on a monitor with VGA resolution (640 x 480). If this happens, the first item in the menu is **More**. Highlight it to view more menu items.

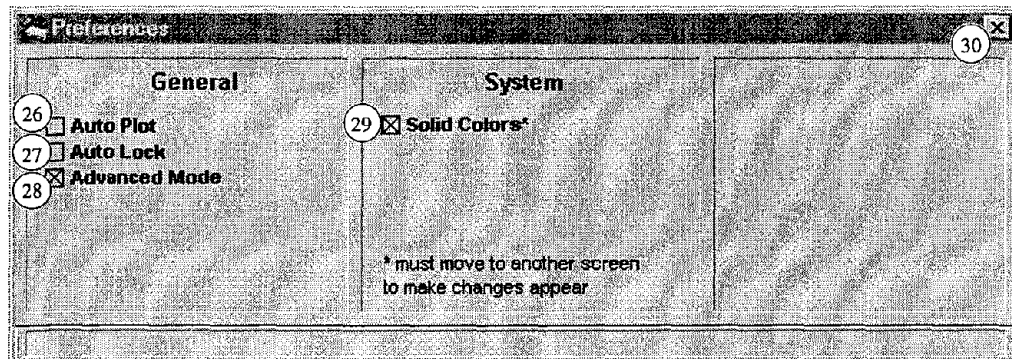
- 9 **Back** button. Click this button to go back to the previous library. This is the library that you came from by using a link, the GO menu, or the History dialog box.
- 10 **Number Tools** button. Click this button to go to the calculator screen. This is equivalent to the **Calculator** command from the **Tools** menu. The calculator screen is used to create and edit tabular data. Control-click to get a file browser to locate the library and establish a link to it. The default is `Sgui_lib\Calc.tbk`.
- 11 **Text Editor** button. Click this button to go to a utility library used to create and edit tabular data. This is equivalent to the **Text Editor** command from the **Tools** menu. This can also be set to go to a text editor, such as the Windows **Notepad** program. Control-click to get a file browser to locate an executable program file and establish a link to it. The default link is `Programs\QLPad.exe`.
- 12 **Library Editor** button. Click this button to bring up a floating window that can be used to edit libraries. This is equivalent to the **Library Editor** command from the **Tools** menu. The current library is automatically loaded into the editor window.
- 13 **Spectrum Analyzer** button. Click this button to go to the **Spectrum Setup** library with settings for a spectrum analyzer. This is equivalent to the **Spectrum** command from the **Tools** menu. Control-click to get a file browser to make a different library the default.

Note: The spectrum analyzer and the settings screen are not present in the standard VDM RoAD package.

- 14 **Batch Plot** button. Click this button to go to the **Plot Setup: Batch** library. This is equivalent to the **Batch Plotting** command from the **Tools** menu. Control-click to get a

file browser to make a different library the default. The default link is `Batch\Plot_bat\Plot_bat.tbk`.

- (15) **Plot Setup** button. Click this button to go to the **Plot Setup: Single** library with plot settings that define plot variables, formats, etc. This is equivalent to the **EP Setup** command from the **Tools** menu. Control-click to get a file browser to make a different library the default. The default link is `Plot\Plot.tbk`.
- (16) **Batch Runs** button. Click this button to go to the **Runs: Batch** library. Control-click to get a file browser to make a different library the default. The default link is `Batch\Runs_bat\Runs_bat.tbk`.
- (17) **Runs** button. Click this button to go to a **Runs** library with settings to make a single simulation run. This is equivalent to the **Runs** command from the **Tools** menu. Control-click to get a file browser to make a different library the default. The default link is `Runs\Runs.tbk`.
- (18) **Print** button. Click this button to print a copy of the current screen display.
- (19) **Preferences** button. Click this button to bring up a floating window used to set global preferences for VDM RoAD. This is equivalent to the **Preferences** command from the **Tools** menu.



- (26) **Auto Plot**. When checked, the plotter is launched whenever you change a plot setup link from the **Runs** screen. If you mainly make one plot at a time, this saves time because you don't have to click the **Plot** button after choosing the plot setup. If you mainly make several plots at a time, you may find this distracting.
- (27) **Auto Lock**. When checked, VDM RoAD automatically locks every data screen when you leave it. Otherwise, data sets are locked only when you click the locked boxes (22) manually.
- (28) **Advanced Mode**. When checked, you can change links to connect with different libraries. When not checked, you must live with the links the way they are.
- (29) **Solid Colors**. Check this box if your video display supports 256 or more colors. Un-check it if your display is limited to 16 or fewer colors.

- ③① **Close.** Click here to exit the **Preferences** window.

Check Boxes

- ②① **Locked** box. Click this box to check or un-check it. When checked, the data set is locked and you cannot change text in any of the edit fields or any links. You also cannot delete the data set using the **Delete** button.
- ②① **Notes** box. Click this box to check or un-check it. When checked, the **Notes** field is displayed. Otherwise, it is hidden. The contents of a **Notes** field are strictly for your use in documenting data. They are not sent to the simulation program.

This button hides the **Notes** field, but does nothing to the contents. You do not lose your notes when you hide them.

When you create a copy of a data set using the **New** button ⑥, the contents of the **Notes** field are also copied, even if the box is not checked.

Data Set Information

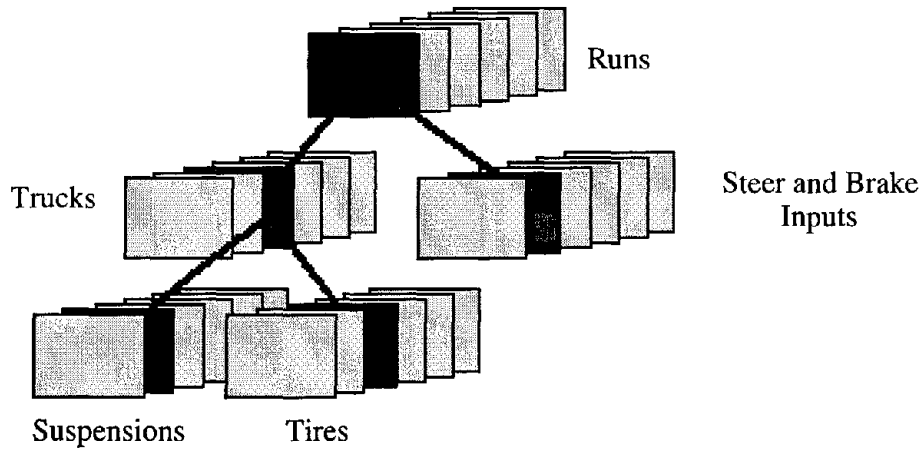
- ②② **Changed** circle. The small circle is normally transparent. However, a red dot appears when a link is changed, or when a field is changed and you leave the field by clicking somewhere else, using the **Tab** key, etc. The red dot indicates that the **PAR** file associated with the current data set will be updated when you leave the screen.

Click on the red circle to force VDM RoAD to overwrite the **PAR** file immediately, such as when testing new animator settings as described in section **Testing New Shapes and Animator Data Sets** in Chapter 5 on page 65.

- ②③ **ID** Number. Each data set in the library has a unique ID number. The number is used to automatically name the **PAR** files associated with the data set. For example, if the ID is 131, a file 131.par is created automatically by VDM RoAD and located in the same folder as the **TBK** library file. Click on the red dot to cause VDM RoAD to update the auxiliary file immediately. When the file is created, the red dot disappears and the circle is transparent again.
- ②④ **Time** of last change. This is set whenever the **PAR** file is updated.
- ②⑤ **Date** of last change. This is set whenever the **PAR** file is updated.

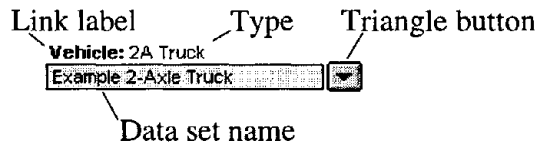
Data Links

Each VDM RoAD screen contains a data set that is part of a relational data base. Data sets are linked together approximately in a top-down hierarchical fashion.



Parts of a Data Link

The object used to link data sets together is called a data link. As shown below, a data link has several components:

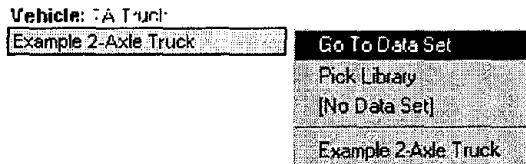


- The name of the linked data set is shown in the blue field, e.g., Example 2-Axle Truck.

Notes: {No library selected} appears in the blue field if the link is not pointing to anything

{No data set selected} appears if the link is pointing to a library, but has not yet been linked to a specific data set in that file.

- Most data links have labels, in bold typeface, that describe the type of data to which the link points, e.g., **Vehicle**. In some cases, additional information is shown in plain typeface, e.g., 2A truck.
- The triangle button is used to display a pull-down menu.



- There are two keyboard modifiers for the triangle button:
 1. Press the **Ctrl** key when clicking the button to get the full pathname of the linked library, e.g., c:\VDM RoAD\Vehicles\2a_truck.tbk.

2. Press the `Shift` key when clicking the button as a shortcut for selecting the menu item **Go To Data Set**. This will take you to the linked data set (it is the same as selecting the first menu item). This is usually quicker than using the menu.

The Data Link Menu

The items perform the following functions:

- **Go To Data Set**. Goes to the currently linked data set.
- **Pick Library**. Brings up the file browser dialog to select a new TBK library file for the link. The browser may request a library of a certain type, e.g., **Vehicle**. If the TBK file that is selected does not contain the right kind of data, a warning is displayed. However, you can override the warning and make the link anyway. **This is not recommended for beginning users.**
- **[No Data Set]**. This breaks the current link to a data set. The link to the library remains, but, without a data set, data from the library are not used in simulations.
- *Data Set Names*. All of the menu items after the dividing line are the names of the data sets in the linked library. Change the link to a different data set by selecting its name.

If the current screen has the **Locked** box checked, only the first item on the data link menu is active. Other items that would involve a change in the data set are not allowed.

Tabular Data

Some of VDM RoAD data sets involve tables of numbers. The values are plotted, to show the functional relation graphically, and to help identify errors. Figure 7.2 shows an example data set.

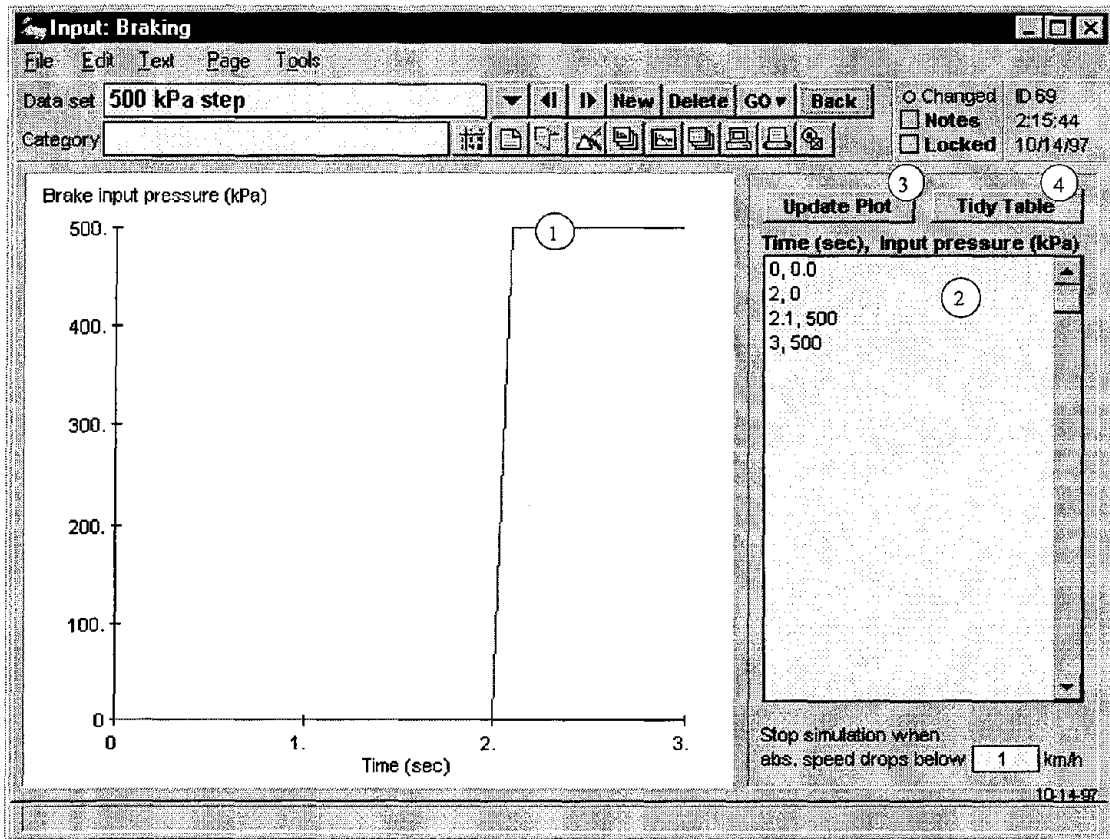


Figure 7.2. Example screen for tabular data.

Note: For these data sets, the **Notes** field is often located on top of the table of numbers ②. Therefore, to view the numbers in the table, the **Notes** button must not be checked.

There are four standard items on each screen with tabular data.

- ① **Plot.** The plot is based on the tabular data ②, using the first number on each line for an X value and the second number for a Y value. The axes are scaled automatically to include the full range of numbers in the table.

When the mouse pointer is positioned over the plot, the corresponding X and Y values are shown in the status bar. Clicking the left mouse button causes the currently displayed X and Y values to be appended to the table. This feature can be used to rapidly build a new data set. The new data points are not plotted until the **Update Plot** button ③ is clicked.

- ② **Table field.** Each line in this field should have an X value followed by a Y value. The numbers must be separated by a comma. Additional information can be added after a second comma. These numbers are passed to the simulation programs exactly as they appear in the field.

The table field requires at least two lines of data in order to create a plot ①. If you attempt to make a plot or leave the screen with only one pair of X-Y points, an error message is generated.

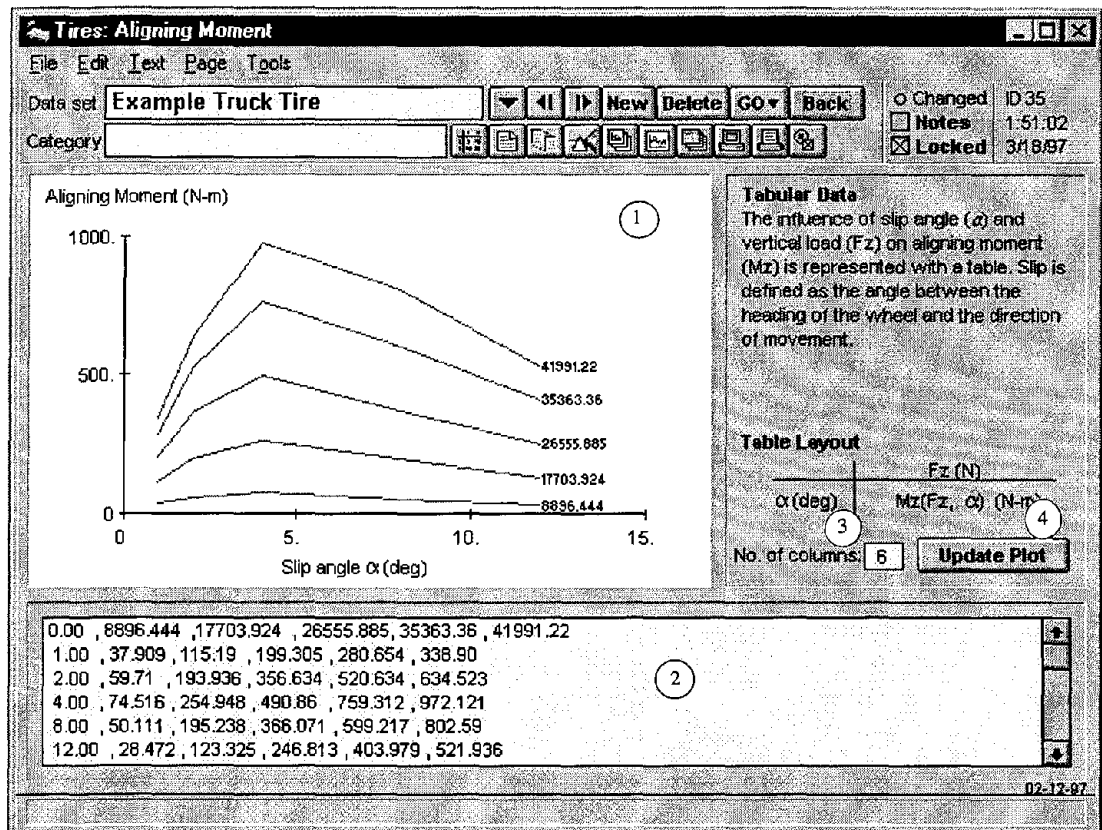
The plot ① shown on this type of screen covers only the range of data. In the simulation, values might extend beyond the range shown. For example, if the simulation runs longer than 3 seconds, the independent variable (time) would have a value outside the range covered by the table.

The simulation programs in VDM RoAD have two methods for extrapolating outside the range of a table.

1. For variables that are inputs to the vehicle (controls, such as brake pressure, or disturbances, such as wind), *flat-line extrapolation* is used. If the independent variable is less than the first value in the table, the first value of the dependent value is used. If the independent variable is greater than the last (highest) value in the table, then the last value of the dependent variable is used. For example, the data shown in Figure 7.2 would cause a brake pressure of 0.0 to be used for values of time less than zero. For values of time greater than 3, the brake pressure would be 500.
 2. For variables that describe vehicle properties (for example, spring force vs. deflection), *constant-grade extrapolation* is used. Depending on whether the independent value less than or greater than the range of the table, the first or last two points are used to extrapolate by assuming the same gradient between the dependent and independent variables.
- ③ **Update Plot** button. Click this button to create a new plot of the data in the field ②.
- ④ **Tidy Table** button. Click this button to “clean up” the appearance of the numbers in the table field. A space is inserted after the comma, and additional information after a second comma is removed. This operation is useful when tabular data are pasted into the edit field from some other source, such as an echo file created by a VDM RoAD solver program.

2D Tabular Data

Some of the data sets in VDM RoAD involve two-dimensional tables of numbers, commonly called “carpet plots.” The values are plotted to show the functional relation graphically and to help identify errors. The figure below shows an example data set. The format of the table is usually shown in the area to the right of the plot, and the data for the table are typed in the field underneath the plot ②. The plot shown is based on these numbers.



There are four standard items on each screen with carpet plot data.

- ① Plot. The plot is based on the tabular data in ②, using the format shown in the area entitled **Table Layout**. The axes are scaled automatically to include the full range of numbers in the table.
- ② Table field. The top row contains the values for one of the independent variables. The left column contains the values for the other independent variable. As indicated in the area titled **Table Layout**, the values in the first row are vertical load (Fz) and those in the first column are slip angle α . The other values are the dependent variable: in this case, aligning moment.

The first number is common to both the first row and the first column, and is ignored. However, a number must be present to hold the place and maintain the correct size of the table.

All carpet plot tables must have at least two columns and two rows. An error message is generated if there are not enough data.

- ③ No. of Columns. This is the total number of columns in the Table field. In the example it is 6, (1 column of slip values plus 5 columns for different loads).

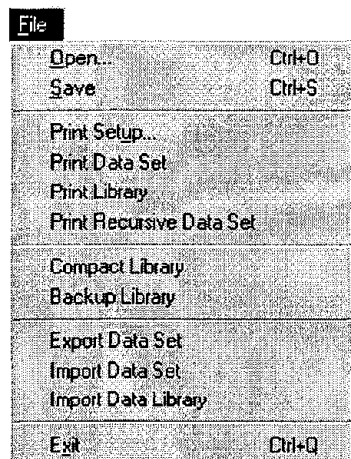
If this value is set too small, data in higher columns are ignored. For example, if it were set to 5, the last column would not be used by the simulation program.

If this value is set too large, an error will occur when the program tries to read additional numbers.

- ④ **Update Plot** button. Click this button to create a new plot of the data in the field ②.

File Menu

The File menu contains commands pertaining to library TBK files. In VDM RoAD, this menu is not used as much as in most Windows programs, because the opening and saving of files is handled automatically.



Open

Select this menu item to open an arbitrary TBK library file.

Save

Select this menu item to immediately save the current TBK file to disk. The files are automatically saved whenever you move from one library to another, so this command is used mainly when the machine stability is in doubt and a program crash is possible.

Print Setup

Select this menu item to select a Windows printer and possibly configure it.

Print Data Set

Select this menu item to print the current screen.

Print Library

Select this menu item to print all data sets in the current library, using the data screen layout.

Print Recursive Data Set

Select this menu item to print the current screen, plus all data sets linked to the current one, plus all data sets linked to those, and so on.

Compact Library

This command causes ToolBook to compact the file, eliminating space that was used by data sets that have been deleted.

Over time, the TBK library files can grow more than expected, because when you delete data sets the file space is not recovered. Use this menu item if you notice that some of the TBK files have gotten very large. Depending on the amount of graphic content, the normal size of a TBK file can be as small as 70K, or as large as 700K.

Backup Library

Select this menu item to make a backup copy of the current TBK library file. The copy is given the extension BAK.

Export Data Set

Select this menu item to create a text file that can be read by VDM RoAD to completely recreate a data set.

Import Data Set

Select this menu item to import a text file previously created with the **Export Data Set** command.

Import Data Library

Select this menu item to import all data sets in another TBK file into the current one. This command is useful for transferring information between separate VDM RoAD installations. It is also useful for importing data from an old version of VDM RoAD into an updated version.

If you select this item from the **Startup** screen, and import data from another Startup screen, then VDM RoAD will attempt to import **all** data from the other copy.

Exit

Select this menu item or use the keyboard command Ctrl-Q to exit from VDM RoAD.

Edit Menu

The **Edit** menu supports the clipboard and the Find command.

Edit	
Undo	Ctrl+Z
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
Clear	Del
Select All	Ctrl+A
Find..	

Undo

Select this menu item or type the keyboard command (Ctrl-Z) to undo the most recent entry or change in a text field.

Cut

Select this menu item or type the keyboard command (Ctrl-X) to cut the currently selected text to the clipboard.

Copy

Select this menu item or type the keyboard command (Ctrl-C) to copy the currently selected text to the clipboard.

Paste

Select this menu item or type the keyboard command (Ctrl-V) to paste the clipboard to the current cursor location. If text is currently selected, it is replaced by the contents of the clipboard.

Clear

Select this menu item to clear the currently selected text.

Select All

Select this menu item to select all text in the field where the cursor is currently located. The keyboard command is Ctrl-A.

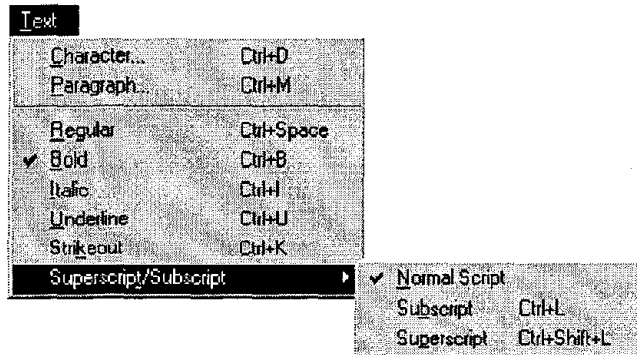
Find

This command can be used by advanced users to locate information in VDM RoAD data fields.

WARNING: It is possible to trick the underlying ToolBook runtime system, leading to errors that will corrupt your files. Use with caution.

Text Menu

The **Text** menu contains commands involving the display of text in edit fields and notes. Although text formatting has no effect on how the simulations are run, you can control the appearance of the text in notes and data fields if you wish.



Note: The items in the Text menu apply only to the currently selected text in a single field.

Character

Select this command to bring up a floating window to change the character properties of the selected text (Font, Font Style, Size, etc.).

Paragraph

Select this command to bring up a floating window to change the spacing (e.g., Alignment, Indentation, Spacing , etc.) of the selected text.

Regular

Select this menu item to remove any special character properties of the selected text, such as bold, underline, etc.

Bold

Select this menu item to toggle the selected text between **bold** and not bold.

Italic

Select this menu item to toggle the selected text between *italic* and not italic.

Underline

Select this menu item to toggle the selected text between underline and not underline.

Strikeout

Select this menu item to toggle the selected text between ~~strikeout~~ and not strikeout.

Superscript

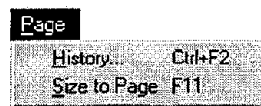
Select this menu item to toggle the selected text between superscript and not superscript.

Subscript

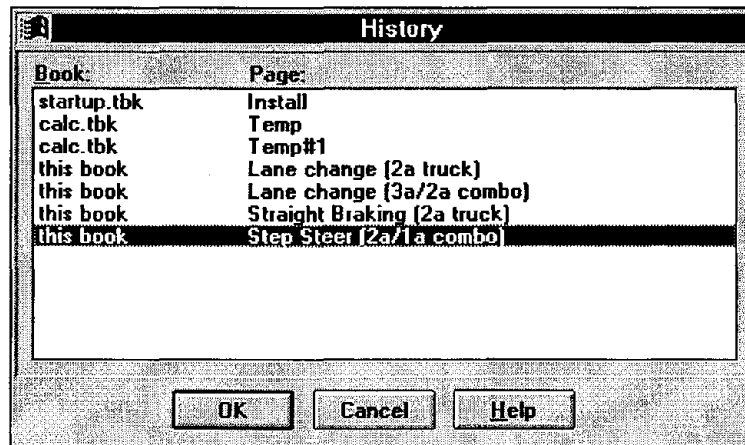
Select this menu item to toggle the selected text between subscript and not subscript.

Page Menu

This menu offers access to previously viewed screens (pages), and also control of how the current data screen is viewed.

**History**

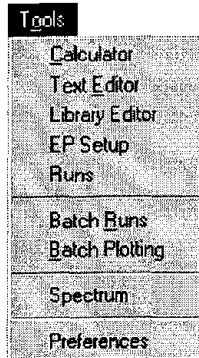
Select this menu item to bring up the following floating window to see a list of the libraries that you have visited during this session. From the window you can select a library and go to it by double-clicking on it or clicking the OK button.

**Size to Page**

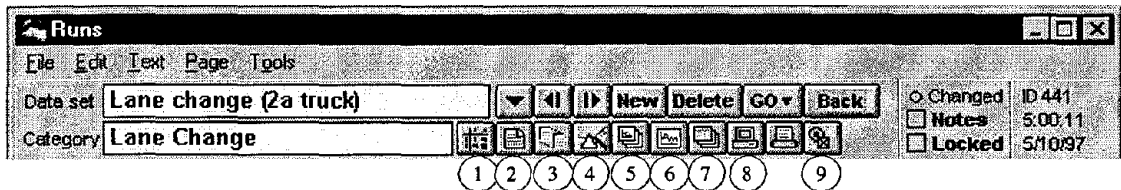
Select this menu item to set the VDM RoAD window to its default size. If you happen to click on the Windows zoom icon, Windows will fill your screen with the current window. However, the window doesn't show any more information. Use this menu item to restore its appearance.

Tools Menu

Use the Tools menu as an alternative to the buttons in the ribbon bar.



The ribbon bar, shown below, was described in more detail starting on page 94.



Calculator

Select this menu item to go to the calculator screen (same as ①).

Text Editor

Select this menu item to go to a utility library used to create and edit text files (same as ②).

Library Editor

Select this menu item to bring up a floating window used to edit existing libraries (same as ③).

EP Setup

Select this menu item to go to the **Plot Setup: Single** library with plot settings that define plot variables, formats, etc. (same as ⑥).

Runs

Select this menu item to go to a **Runs** library with settings to make a single simulation run (same as ⑧).

Batch Runs

Select this menu item to go to the **Runs: Batch** library (same as ⑦).

Batch Plotting

Select this menu item to go to the **Plot Setup: Batch** library (same as ⑤).

Spectrum

This item is reserved for the **Spectrum Setup** library (same as ④).

Preferences

Select this menu item to bring up a floating window used to set global preferences for VDM RoAD (same as ⑨).

8. Alphabetical Library Screen Reference

This chapter describes all of the libraries in VDM RoAD. They are listed in alphabetical order, based on the screen (window) title.

Conventions in This Chapter

Each section shows an example data screen. Descriptions are provided for data fields, links, buttons, and other objects in the user interface.

Standard Objects

Standard objects that were described in Chapter 7 are not covered again in this chapter. For example, the buttons at the top of the screen (**New**, **Delete**, etc.) are the same in every library.

Keywords

Parameter values that are displayed in yellow fields will usually appear in the echo files produced by the simulation programs. These echo files identify the parameters with keywords and list them in alphabetical order (see Appendix D for an example). Along with the definitions for the parameters, keywords are defined so you will know where to look for them in the echo files. They are shown in parentheses in the *Courier* font.

Many of the parameters in the VDM RoAD models are applied in several places on the vehicle. For example, every axle has a mass, and the same data screen is used to describe an axle mass, no matter where it is located in the vehicle. Every axle mass is associated with the keyword `MUS`. The location of the axle is specified with additional keywords `IUNIT` and `IAXLE` that are written automatically by VDM RoAD to tell the simulation which axle mass is being specified. For components that differ on the left and right side, a keyword `ISIDE` is also used.

Parameters that are repeated are called indexed parameters, with the indexing being defined with the keywords `IUNIT`, `IAXLE`, etc. The indexing keywords can be of interest to advanced users who override vehicle properties using the Miscellaneous fields scattered throughout the VDM RoAD data screens. Therefore, they are specified along with the parameter keyword. For example, the entry for axle mass reads: Axle Mass (keyword = `MUS (IUNIT, IAXLE)`).

Indexed parameters that are only applied in one place are shown with numerical indices. For example, the wheelbase of unit 1 is shown as `LWB(1)`.

Location of Library

Each section ends with the location of the library from two points of view. First is a list of the screens needed to reach the library using the user interface, as shown in Figure 2.2, Map of the VDM RoAD libraries, on page 14. For example, to get to the Animator Camera screen, start with the **VDM RoAD Startup** screen, then press the **Start** button to go to the **Runs** screen, then follow a blue link to go to **Animator Camera** screen. This sequence is written at the end of the reference section as:

VDM RoAD Startup
Runs
Animator Camera

In addition, a relative pathname is given for the library TBK file (e.g., `Animate\Camera\Camera.tbk`). Pathnames are relative to the root directory of VDM RoAD. For example, the absolute pathname would be something like `C:\VDM_RoAD\Animate\Camera\Camera.tbk`. The relative pathnames are useful to know even if you are not looking for a file, because they appear in the pull-down menu of the GO button (in the ribbon bar).

Aerodynamics

This screen contains all parameters that affect the aerodynamic forces and moments acting on a moving automotive vehicle. This library is not used for the heavy truck models.

Discussion

Aerodynamic effects are represented in the vehicle model by a force and a moment acting on the sprung mass. The components of the force and moment in the body-fixed sprung mass frame are calculated according to the formulas shown on the screen, which involve Q , the dynamic air pressure ($Q = \frac{\rho V^2}{2}$) along with the angle of the air stream relative to the vehicle—the aerodynamic slip angle β .

The coefficients are based on the SAE convention in which aerodynamics forces are applied to a point in the sprung mass at the midpoint of the track width and wheelbase, nominally coincident with ground level.

A more complete description of aerodynamic effects and some charts with typical coefficient values appear in Pages 79-105 of *Fundamentals of Vehicle Dynamics* (Gillespie, T.D., Society of Automotive Engineers, Inc., Warrendale, PA, 1992). Note that the calculations for the aerodynamics force take into account the speed and direction of the wind as specified on the **Input: Wind** screen (V is the magnitude of the speed of the vehicle relative to the air striking it).

Aerodynamics

File Edit Text Page Tools

Data set: **Taurus aero data (full)** | New Delete GO Back | Changed ID 34
 Notes 12/13/26
 Locked 12/17/96

Category: _____

Aerodynamic Equations

$$F_x = -Q \cdot A \cdot (C_{D0} + K_D \cdot \beta^2)$$

$$F_y = -Q \cdot A \cdot K_Y \cdot \beta$$

$$F_z = Q \cdot A \cdot (C_{L0} + K_L \cdot \beta^2)$$

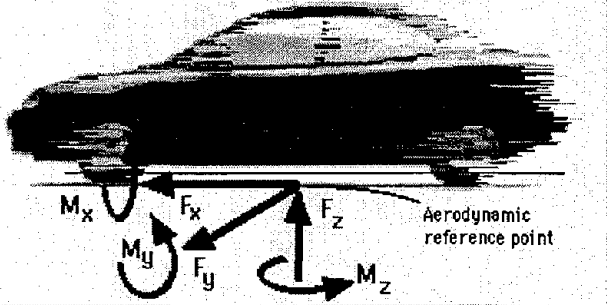
$$M_x = -Q \cdot A \cdot L_{wb} \cdot K_R \cdot \beta$$

$$M_y = -Q \cdot A \cdot L_{wb} \cdot (C_{M0} + K_M \cdot \beta^2)$$

$$M_z = -Q \cdot A \cdot L_{wb} \cdot K_N \cdot \beta$$

in which

$$Q = (\rho \cdot V^3) / 2$$

$$\beta = \text{Aerodynamic slip angle}$$


Drag coefficient (C _{D0}):	1	0.372	-
Drag coefficient (K _D):		0.480e-3	2 1/deg ²
Side force coefficient (K _Y):	3	0.476	1/deg
Lift coefficient (C _{L0}):		0.2152	4 -
Lift coefficient (K _L):	5	.9330e-3	1/deg ²
Roll coefficient (K _R):		0.290e-2	6 1/deg
Pitch coefficient (C _{M0}):	7	-0.55	-
Pitch coefficient (K _M):		0.2180e-	8 1/deg ²
Yaw coefficient (K _N):	9	0.40e-2	1/deg

Frontal area (A):	10	2.23	m ²
Air density: (ρ)		1.22	11 kg/m ³

The aerodynamic reference point is in the ground plane, at the mid-point of the vehicle track and wheelbase.

5-7-96

User Settings

- ① Drag coefficient: aerodynamic drag proportional to Q when β is zero (keyword = CDAERO).
- ② Drag coefficient: portion of drag that is quadratic with respect to β (keyword = KDAERO).
- ③ Side force coefficient proportional to Q and β (keyword = KYAERO).
- ④ Lift coefficient: aerodynamic lift proportional to Q when β is zero (keyword = CLAERO).
- ⑤ Lift coefficient: portion of lift that is quadratic with respect to β (keyword = KLAERO).
- ⑥ Roll moment coefficient proportional to Q and β (keyword = KRAERO).
- ⑦ Pitch moment coefficient: aerodynamic pitch moment proportional to Q when β is zero (keyword = CMAERO).
- ⑧ Pitch moment coefficient: portion of pitch moment that is quadratic with respect to β (keyword = KMAERO).
- ⑨ Yaw moment coefficient proportional to Q and β (keyword = KNAERO).
- ⑩ Frontal area of vehicle (keyword = ACSAERO).

- ⑪ Air density (keyword = DAIR).

Location in VDM RoAD

VDM RoAD Startup

Runs

Cars

Aerodynamics

File Location

Vehicles\Aero\Aero.tbk

Animator Camera Setup

The Animator Camera screen defines the location of the virtual camera in the animator, the direction in which the virtual camera is looking, the amount of zoom, and the properties of an optional grid.

Discussion

Imagine you have a video camera and are viewing a vehicle as it moves down the road. Your location (actually, the location of the camera you are holding) determines your point of view. From that point, you can aim the camera anywhere. However, you will probably pan the camera to keep the vehicle in view. If the camera has a zoom lens, you might zoom in or out to control the size of the vehicle in the viewfinder.

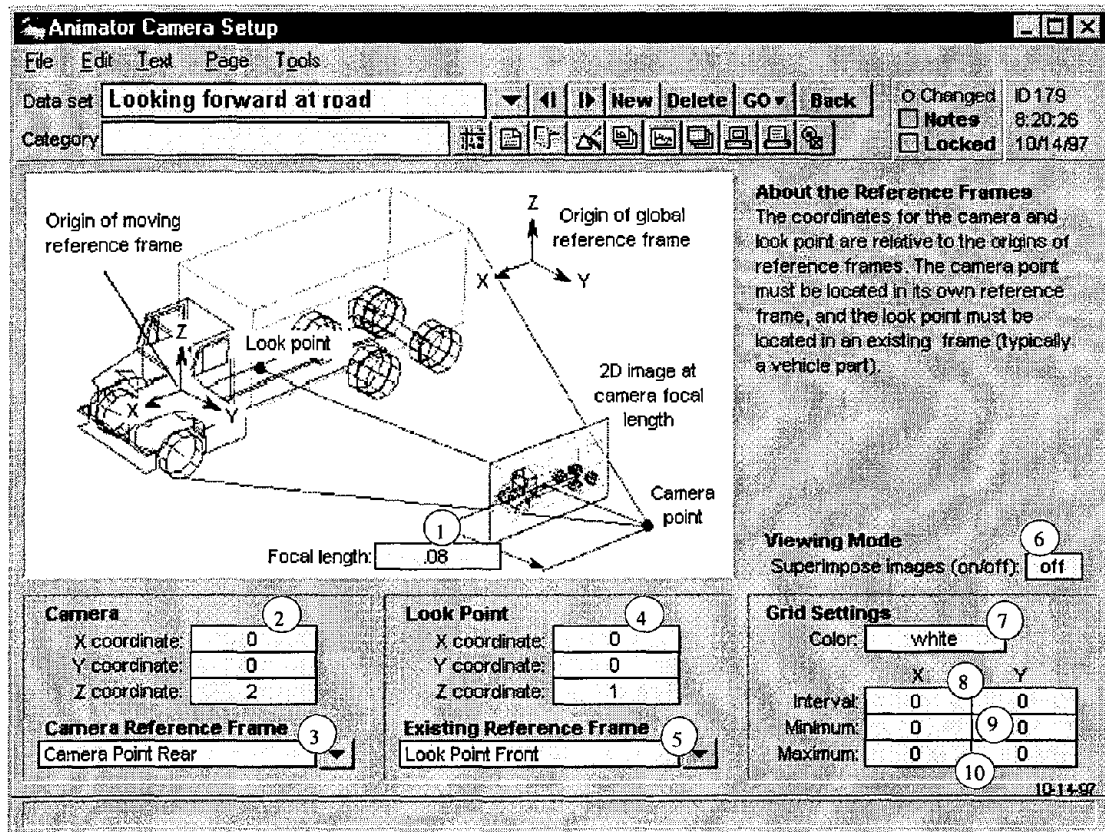
In VDM RoAD, there is no physical vehicle to look at. However, to go along with the simulated vehicles, the animator program described in Chapter 5 simulates the motions that you would view with a video camera. Use the **Animator Camera Setup** screen to establish various camera positions and aiming strategies.

The parameters in this library are associated with two geometric points illustrated on the screen: (1) the camera point, (2) and the look point. At each output time interval, the animator generates a 2D image based on the relationships between the simulated vehicle and the camera and look points (see the illustration on the data screen). The locations of vehicle parts are determined by the simulation program. The camera and look points are determined by the information provided in this data set.

Point locations are defined in 3D space with sets of X, Y, and Z coordinates. In order to provide almost any imaginable view of the simulated vehicle motion, the animator program allows both the camera and look points to be associated with any user-defined reference frame. The reference frame can be fixed or it can be a vehicle part (e.g., a sprung mass. It is even possible to define a new moving reference frame using variables that are available in the simulation output files.

A grid can be drawn by the animator, oriented in the global X and Y directions. For animations involving a vehicle traversing a road, the grid should be turned off (specify

the color white). For braking simulations, the grid provides a useful background reference. The minimum and maximum X and Y global coordinates for the grid can be specified. Different grid spacing can be used in the X and Y directions. The grid can be turned off manually within the Animator, as described in Chapter 5.



Notes: Chapter 5 explains reference frames. Also, the section **Animator Reference Frames** in this chapter describes how you create them.

All of the numbers shown on this screen can be modified interactively when the animator is running. See Chapter 5 for details.

User Settings

- ① Focal length of the simulated camera (keyword = `set_focal_length`). A large value produces a telephoto lens effect and a small value produces a wide-angle lens effect. Focal length has units of meters.
- ② X, Y, and Z coordinates of the camera point (keywords = `set_camera_x`, `set_camera_y`, `set_camera_z`). These coordinates have units of meters. The coordinates apply to the reference frame specified in ③.

- ③ Link to the reference frame in which the camera is located. The reference frame can be fixed or moving. The camera must have its own named reference frame (in the example above, it is Camera tracking X-Y).

Notes: The animator program requires each reference frame to have a unique name. The reference frame used to define the camera coordinates cannot be used anywhere else. If you want to locate the camera in a vehicle reference frame (e.g., the sprung mass), then create a copy of the reference frame of interest, give the copy a new name, and link to the copy.

Although vehicle reference frames can be used for the camera, it is more common to define a custom reference frame.

- ④ X, Y, and Z coordinates of the Look Point (keywords = `set_lookpoint_x`, `set_lookpoint_y`, `set_lookpoint_z`). These coordinates have units of meters. The coordinates apply to the reference frame specified in ⑤.
- ⑤ Link to the reference frame in which the look point is located. The reference frame can be fixed or moving. This reference frame must also be used somewhere else. (It is typically the same as the camera point reference frame ③.)
- ⑥ Viewing mode (keyword = `set_superimpose`). This value specifies whether the animator is to erase each frame after it is drawn or not (superimpose images). Set on to superimpose images, set off for normal viewing. The default is off.
- ⑦ Grid color (keyword = `set_color`). This is text that specifies the color of the grid. Valid values are black, white, red, green, blue, yellow, magenta, light gray, and dark gray. If no color is specified, the grid is shown in black. To avoid showing the grid, set its color to white.
- ⑧ Grid intervals for X and Y (keywords = `set_interval_x`, `set_interval_y`). The units for these values are meters.
- ⑨ Minimum X and Y values covered by the grid (keywords = `set_min_x`, `set_min_y`). The units for these values are meters. If not specified, the default values are zero.

The animator will automatically determine the range for the grid if the minimum and maximum values are set equal or left blank.

- ⑩ Maximum X and Y values covered by the grid (keywords = `set_max_x`, `set_max_y`). The units for these values are meters. If not specified, the default values are zero.

The animator will automatically determine the range for the grid if the minimum and maximum values are set equal or left blank.

Note: The X and Y directions are handled independently. Even if the range is set manually in one direction, the range for the other direction can be determined automatically.

Location in VDM RoAD

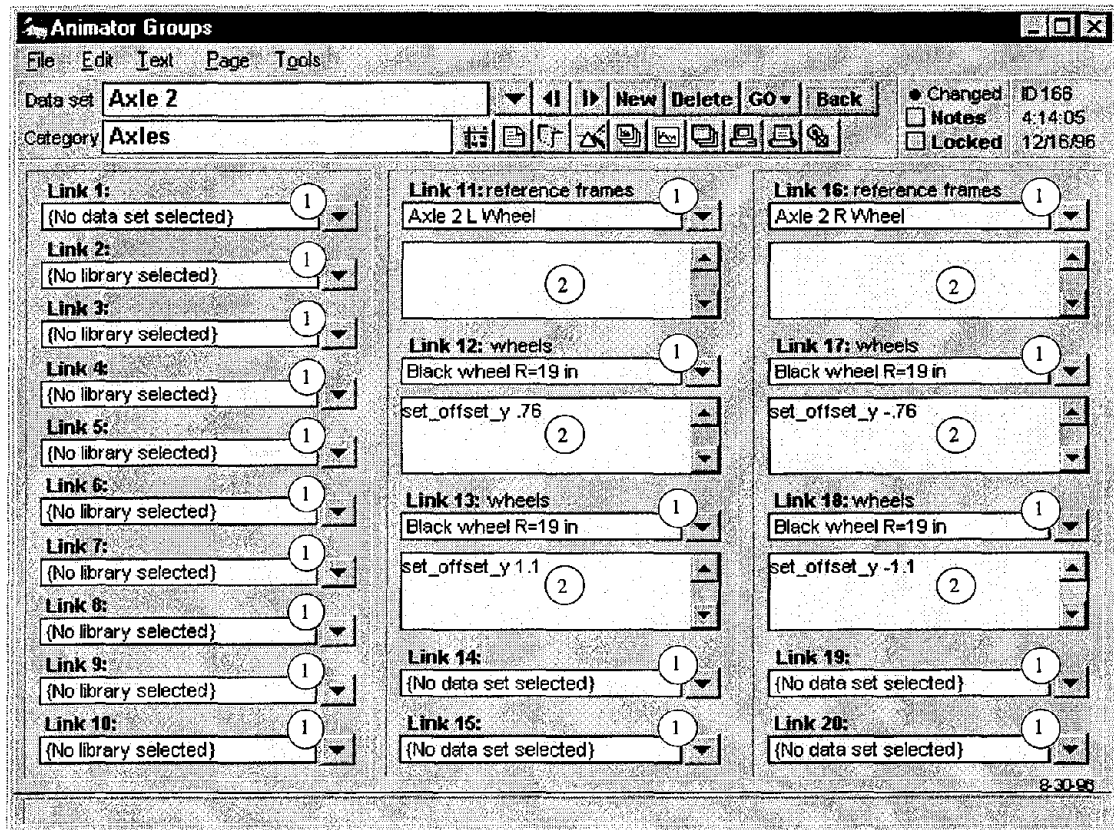
VDM RoAD Startup
Runs
Animator Camera

File Location

Animate\Camera\Camera.tbk

Animator Groups

The **Animator Groups** screen is used to group shapes, wheels, reference frames, and other groups. The library can be used for several purposes, because of the versatility of the animator.



Discussion

The Animator draws three kinds of objects: a single grid, multiple shapes, and multiple wheels. In order to build a detailed visual representation of a road or vehicle, groups of shapes and wheels must be collected and associated with moving reference frames.

This screen is used to group other animator data sets together. Typically, the data sets being grouped are reference frames, body shapes, wheels, and other groups that make up the image of a vehicle or road.

As explained in Chapter 5, reference frames are used to define motions of vehicle parts. As the inputs to the animator are processed, each shape or wheel is associated with a reference frame.

There is always a single active reference frame. Each shape and wheel that is processed is defined within this frame.

It is important to understand that all objects in a reference frame must be processed together. For example, suppose there is a reference frame with the title `truck body`. The animator will not allow some shapes to be processed for `truck body`, then process shapes for another reference frame, and then switch back to `truck body`. Once the animator starts processing data for a particular reference frame, it is no longer possible to add shapes or wheels to reference frames that were previously processed.

Although shapes and wheels are always associated with the active reference frame, the camera and look-point coordinates are not. However, because the camera and look-point data are contained in the library **Animator Camera Settings**, you should not have occasion to deal with them in an **Animation Groups** data set.

Note: The reason that shapes and wheels are handled differently than camera and look-point data is that shapes and wheels can be duplicated. For example, the same wheel information is usually linked to every axle reference frame. In contrast, there is only a single camera point and a single look point.

An animator group can be set up several ways.

- It can group wheels or shapes together and associate them with a reference frame. For example, a `truck body` group might associate several shapes together with a `truck body` reference frame.
- It can group several shapes or wheels together without associating them with a reference frame. For example, you might want to create a complex object that could be associated with several different reference frames under different conditions.
- It can group together other groups. For example, a whole truck group could be assembled from a `truck body` group, an `axle 1` group, and an `axle 2` group.

- It can group together objects with more than one reference frame. For example, an axle group might associate two wheel objects with a reference frame for a dual assembly on the left-hand side of the axle, plus two wheel objects with a reference frame for the dual assembly on the right-hand side of the axle.

When combining reference frames and other data sets, be aware that the data are sent to the animator in the same order they are numbered on the screen: top to bottom, then left to right. If one of the links is to a reference frame, it should be the first one (Link 1). It applies until another reference frame is encountered.

Note: The animator group is one of the more difficult to understand screens in VDM RoAD. It may help to browse through the groups that are installed with VDM RoAD, to see by example how complex systems are assembled using this screen.

User Settings

- ① Miscellaneous links. Links are used to include other groups, wheels, shapes, and reference frames. A group, wheel, or shape appearing in one of these links is attached to the reference frame link most immediately above it. If none of the links are made to a reference frame, then all wheels and shapes are attached to the most recently introduced reference frame, in the data set above this one (i.e., the data set that has a link down to this one).
- ② Miscellaneous fields. Enter keywords and the value you want to assign them. The format is that each line has a keyword and value, separated with white space (at least a single space). These keyword values apply to the Link that they are directly underneath (Links 11, 12, 13, 16, 17, or 18). These fields can be used to add to shape or wheel data. For example, to mirror an object from the left to the right side, enter the keyword `set_scale_y` followed by a value `-1`.

Location in VDM RoAD

Note: This library can be applied in different ways, which means there may be links to it from libraries other than those shown below.

Vehicle Definition

VDM RoAD Startup

Runs

Vehicle: {car, unloaded single-unit truck, or combination}

Animator Groups

Road Definition

VDM RoAD Startup

Runs

Input: Highway Data (IHSDM)

Animator Groups

File Location

Animate\Groups\Groups.tbk

Animator Reference Frames

This screen is used to define a reference frame (moving coordinate system) for use by the animator. The concept of a reference frame is important if you wish to effectively use the animator. Chapter 5 introduces the concept. This section explains how you define a reference frame within VDM RoAD.

Animator Reference Frames

File Edit Text Page Tools

Data set: **Truck body** | Changed ID:174
 Notes 9:05:21
 Locked 2/18/97

Category: **Bodies**

Origin of global reference frame

Origin of moving reference frame

A reference frame is a moving set of points that maintain their spatial relationship relative to each other at all times. Points in the frame are specified by coordinates that are local to the moving frame.

In the animator, the reference frame is defined by the position and orientation of a right-handed rectangular coordinate system. The position is defined by up to 3 global coordinates (X, Y, Z) and 3 Euler rotation angles (Yaw, Pitch, Roll).

Values for the coordinates and angles are read from an ERD file. The names are taken from the yellow fields on the left. Each coordinate and angle can be given a scale factor and offset. The value used by the animator is: $Vo + \text{value} * SF$ where **value** is read from the ERD file, and **Vo** and **SF** are specified in the yellow fields to the left.

If a name is not specified, a value of 0 is used for **value**. A value of 0 is also used if the specified name is not found in the ERD file.

The rotation sequence should be either **yaw_pitch_roll** or **yaw_roll_pitch**.

8:31:00

Global Coordinates of Origin			
	Name	Offset	Scale Factor
X coordinate:	Xo	1	3
Y coordinate:	Yo	2	
Z coordinate:	Zo		

Euler Angles for Frame			
	Name	Offset	Scale Factor
X rotation:	Roll		
Y rotation:	Pitch	4	6
Z rotation:	Yaw		

Rotation sequence: 7 yaw_pitch_roll

Discussion

The animator in VDM RoAD shows moving and stationary wire-frame figures. The lines used to draw each wire-frame object may move relative to other objects, but the spatial relationships between the lines in a single object are fixed.

When the relationships between a set of points do not ever change (i.e., they form a rigid structure), they are said to exist in the same reference frame. Therefore, the coordinates

provided for shapes are constants when based on a coordinate system fixed in the appropriate reference frame.

In order to draw the wire-frame shapes, the animator must convert relative coordinates in a moving coordinate system to absolute coordinates in the global coordinate system. The conversion is defined by the global position of the origin of the moving coordinate system (its global X, Y, and Z coordinates), along with the orientations of its three axes.

The global coordinate system has the Z axis pointing up (opposite the direction of gravity). The relationship between the vehicle axis systems to the global one depends on the road geometry. Usually, the X, Y, and Z coordinates are specified in the road input file according to station number, and the location of the vehicle relative to the global origin is defined by the X, Y, and Z coordinates of the initial station number and lateral position of the vehicle. If the road heading angle is zero, then the X axis of the vehicle is parallel with the global X axis.

The orientation of a reference frame is defined by three consecutive rotations that are called Euler angles. For vehicles, the angles are commonly called yaw (rotation about the Z axis), pitch (rotation about the Y axis), and roll (rotation about the X axis).

Starting such that the X, Y, and Z axes of the moving frame are parallel with those of the global frame, the moving frame can be oriented by consecutive rotations about its axes. For example, first rotate the frame about its Z axis by a yaw angle. After this is done, the new Z direction is the same as the old, but the X and Y axes are pointed in new directions, called X' and Y'. Next, rotate about the new Y axis, Y', by a pitch angle. After the pitch rotation, the Y axis is still in the Y' direction, but the X and Z axes are pointed in new directions, called X'' and Z''. Finally, rotate a third time, about the new X axis, X'', by the roll angle.

Although it might not be obvious at first if you are not experienced with 3D kinematics, any conceivable orientation of a reference frame can be described with three Euler angles. However, it is necessary to specify the order of the rotations.

Overall, a reference frame is defined by six variables: three coordinates (X, Y, and Z), and three Euler angles. The animator reads the required six variables from the output files generated by the simulation programs. After reading the six variables, each coordinate and Euler angle is calculated with a relationship of the form:

$$\text{coordinate} = C_0 + C * SF_c \quad (1)$$

$$\text{angle} = A_0 + A * SF_a \quad (2)$$

where C and A are the coordinate and angle variables obtained from the ERD file, C_0 and A_0 are the constant offsets, and Sf_a and SF_c are scale factors (gains).

User Settings

- ① Names of coordinate variables for X, Y, and Z (keywords = `set_x_name`, `set_y_name`, `set_z_name`). These are short names in the ERD files, associated with the variable C in eq. 1. If no name is listed, then a constant value of 0.0 is used to

compute the coordinate. If the name is not found in the ERD file, then a value of 0.0 is used.

- ② Offsets for coordinate variables for X, Y, and Z (keywords = `set_offset_var_x`, `set_offset_var_y`, `set_offset_var_z`). These values are numbers, used to replace the symbol C_0 in eq. 1. If no number is entered, a value of 0.0 is used.
- ③ Scale factors for coordinate variables for X, Y, and Z (keywords = `set_scale_var_x`, `set_scale_var_y`, `set_scale_var_z`). These values are numbers, used to replace the symbol SF_C in eq. 1. If no number is entered, a value of 1.0 is used.
- ④ Names of Euler angle variables for roll, pitch, and yaw (keywords = `set_roll_name`, `set_pitch_name`, `set_yaw_name`). These are short names in the ERD files, associated with the variable A in eq. 2. If no name is listed, then a constant value of 0.0 is used. If the name is not found in the ERD file, then a value of 0.0 is used.
- ⑤ Offsets for Euler angle variables for roll, pitch, and yaw (keywords = `set_offset_var_roll`, `set_offset_var_pitch`, `set_offset_var_yaw`). These values are numbers, used to replace the symbol A_0 in eq. 2. If no number is entered, a value of 0.0 is used.
- ⑥ Scale factors for Euler angle variables for roll, pitch, and yaw (keywords = `set_scale_var_roll`, `set_scale_var_pitch`, `set_scale_var_yaw`). These values are numbers, used to replace the symbol SF_a in eq. 2. If no number is entered, a value of 1.0 is used.
- ⑦ Rotation sequence for Euler angles (keyword = `set_euler_angles`). In general, there are 12 possible sequences of body-fixed orientation angle sets. (The constraint is that there cannot be two consecutive rotations about the same axis. Thus, there are three choices for the first rotation, two for the second, and two for the third.) However, only two sequences are common for vehicle dynamics: yaw-pitch-roll (used for large body motions), and yaw-roll-pitch, used for the spinning wheels.

For the animator in VDM RoAD, the value must either be the text `yaw_pitch_roll` or the text `yaw_roll_pitch`.

Location in VDM RoAD

Note: This library can be applied in different ways, which means there may be links to it from libraries other than those shown below.

Camera Setup

VDM RoAD Startup

Runs

Camera Setup

Animator Reference Frames

Vehicle Definition

VDM RoAD Startup

Runs

Vehicle: {car, unloaded single-unit truck, or combination}

Animator Groups

Animator Reference Frames

Road Definition

VDM RoAD Startup

Runs

Input: Highway Data (IHSDM)

Animator Groups

File Location

Animate\Frames\Frames.tbk

Animator Shapes

This screen is used to define one or more wire-frame shapes that are drawn by the animator.

Discussion

A shape is a set of points connected by straight lines. Each point is defined by a set of three coordinates (X-Y-Z). The animator starts with the first point, and draws connecting lines to each following point.

All coordinates are assumed to be in a local coordinate system, associated with the active reference frame. (See the discussion on how shapes and reference frames are associated in the section **Animator Groups**, in this chapter.)

The animator also supports offsets and scale factors. All of the coordinates are calculated using the equation:

$$\text{coordinate} = C_0 + C * SF \quad (1)$$

where C is the original coordinate, C₀ is the offset, and SF is a scale factor (gain).

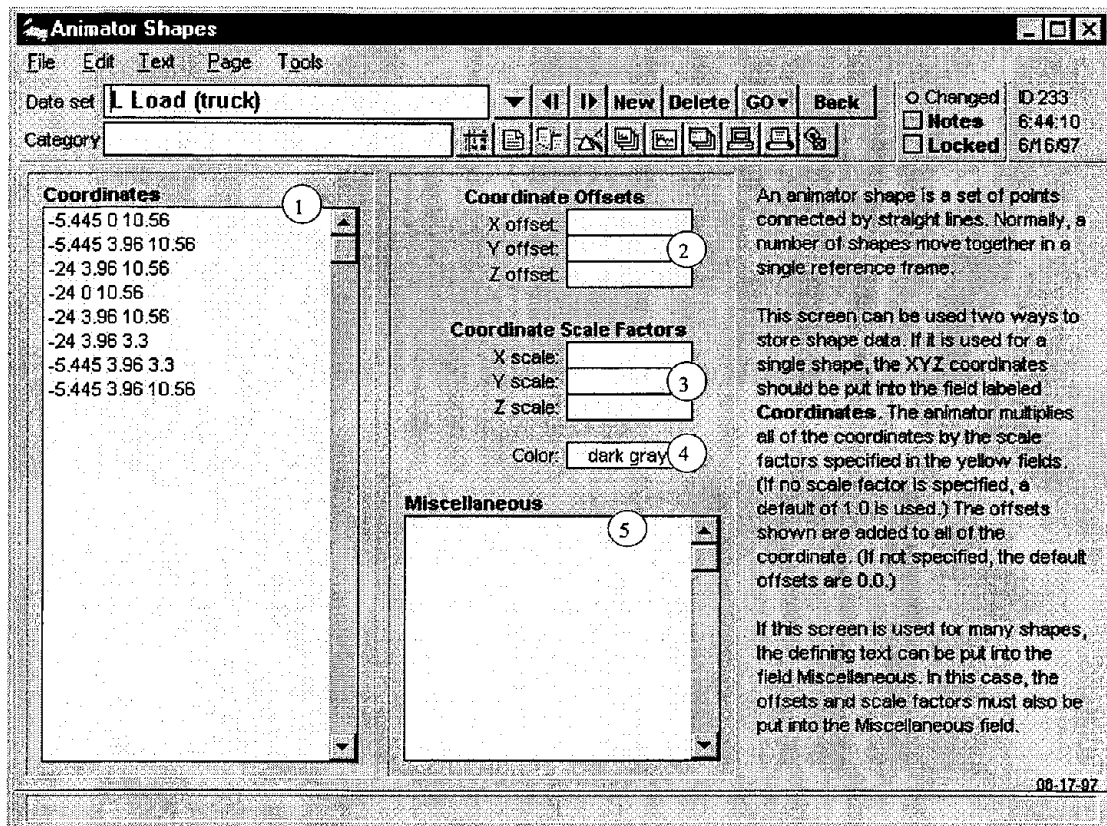
The offsets and scale factors allow the shapes to be relocated and resized without requiring all of the coordinates to be changed by hand. Applying an offset has the effect of relocating the shape within the reference frame. For example, offsets are used to place wheels at the ends of axles, rather than at their centers. Scale factors can be used to change the size of a shape. For example, to make a trailer body twice as long, enter an X scale factor of 2.0.

Scale factors can also be used to mirror a shape. If the scale factor is negative, then all of the corresponding coordinates are given the opposite sign. For example, a left fender can be converted to a right fender by setting the Y scale factor to -1 .

The shape library can handle two kinds of shape data:

1. A single shape is described, using the appropriately labeled fields.
2. Multiple shapes can be described, using the Miscellaneous field.

The second approach is useful if there are many shapes used to describe a complex body, and they are unlikely to be used for any other purpose. By putting the many shapes into one data set, fewer files are processed by the animator, and there is little likelihood of accidentally modifying a group.



User Settings

- ① **Coordinates** (keywords = `set_coordinates`, `end_coordinates` to indicate the start and end of the list). Each line should have the three coordinates of a single point. The units are meters. The syntax is that each of the values is separated by at least one space. Blank lines are not allowed.

- ② **Coordinate Offsets** (keywords = `set_offset_x`, `set_offset_y`, `set_offset_z`). All coordinates in the shape are adjusted by these offsets according to eq. 1. The offsets must have the same units as the coordinates. If not specified, the animator uses default values of 0.0.
- ③ **Coordinate Scale Factors** (keywords = `set_scale_x`, `set_scale_y`, `set_scale_z`). All coordinates in the shape are adjusted by these scale factors according to eq. 1. The scale factors should be dimensionless. If not specified, the animator uses default values of 1.0.
- ④ **Color** (keyword = `set_color`). This sets the color of the lines drawn to connect the points. Allowable values are the words: `black`, `white`, `red`, `green`, `blue`, `yellow`, `magenta`, `light gray`, and `dark gray`. The default is `black`.
- ⑤ **Miscellaneous field**. This field is for advanced users. It can be used to enter multiple shapes. Enter keywords and the value you want assigned to them. The format is a single set per line, with white space (at least a single space) separating them. The field should look like a portion of a PARSEFILE that can be processed by the animator. See Appendix C for the animator keywords and some example files.

Location in VDM RoAD

Vehicle Definition

VDM RoAD Startup

Runs

Vehicle: {car, unloaded single-unit truck, or combination}

Animator Groups

Animator Shapes

Road Definition

VDM RoAD Startup

Runs

Input: Highway Data (IHSDM)

Animator Groups

Animator Shapes

File Location

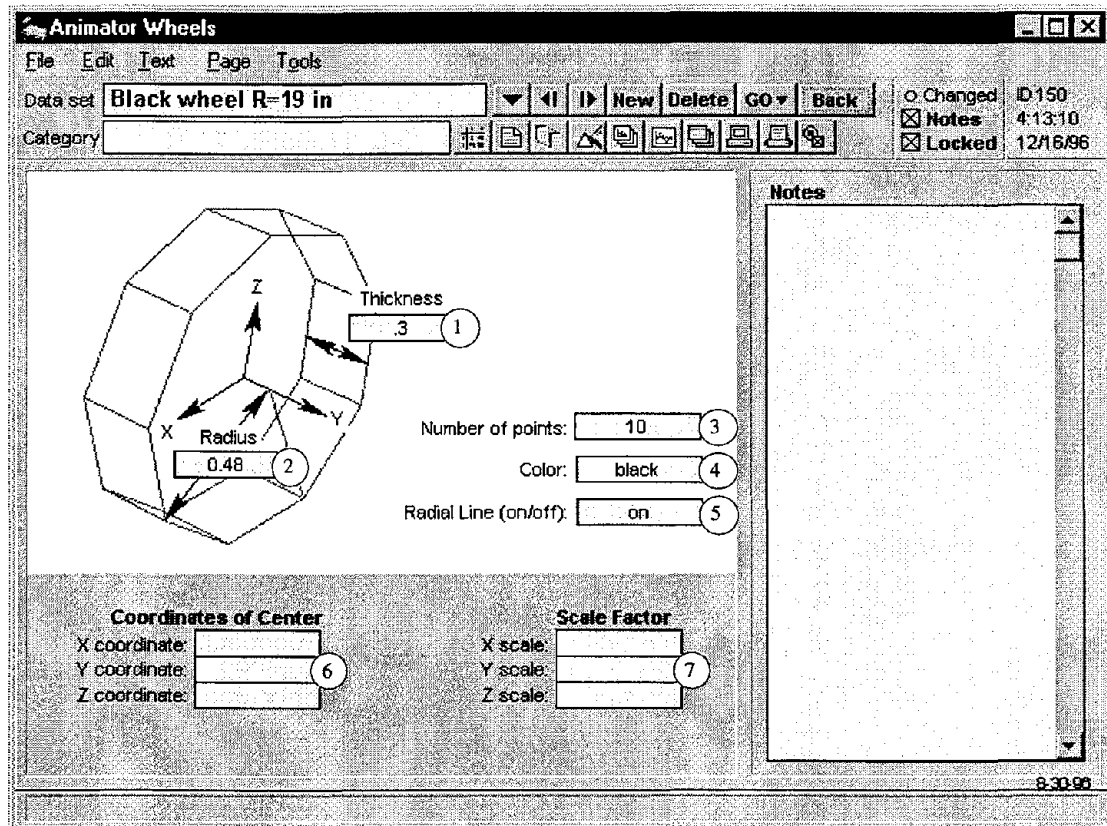
`Animate\Shapes\Shapes.tbk`

Animator Wheels

The current version of the animator supports a single 3D graphic primitive—a cylinder. It is typically used to draw vehicle wheels. This screen is used to define a wheel.

Discussion

The animator considers a wheel to be two polygons with a specified radius, separated by a specified thickness. In addition, the corresponding nodes of the polygons are connected by lines. The wheel is oriented such that its central axis is parallel to the Y axis of the corresponding reference frame.



User Settings

- ① Thickness (keyword = `set_thickness`). This should have units of meters.
- ② Radius (keyword = `set_radius`). This should have units of meters.
- ③ Number of points used for polygon approximation of circle (keyword = `set_num_points`).
- ④ Color of lines drawn by animator to show the wheel (keyword = `set_color`). The allowable values are black, white, red, green, blue, yellow, magenta, light gray, and dark gray. The default is black.
- ⑤ Radial line option (keyword = `set_radial_line`). This determines whether a line is drawn from the center of one of the polygons to the first node. The single line is useful

for determining when wheels lock up during braking simulations. The allowable values for this field are `on` and `off`.

- ⑥ **Coordinates of center** (keywords = `set_offset_x`, `set_offset_y`, `set_offset_z`). These are the X, Y, and Z coordinates of the wheel center in the reference frame with which the wheel is associated. The default values of 0.0 are used if no values are provided.
- ⑦ **Scale factors** in three directions (keywords = `set_scale_x`, `set_scale_y`, `set_scale_z`). These are dimensionless scale factors in the X, Y, and Z directions. They are multiplied by the coordinates of the wheel, centered at 0,0,0, before the offsets ⑥ are added. The Y scale factor adjusts the thickness, and the other two can be used to resize the wheel. If the X and Z scale factors are not equal, the wheel will be elliptical, rather than circular. The default values of 1.0 are used if no values are provided.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: {car, unloaded single-unit truck, or combination}

Animator Groups

Animator Wheels

File Location

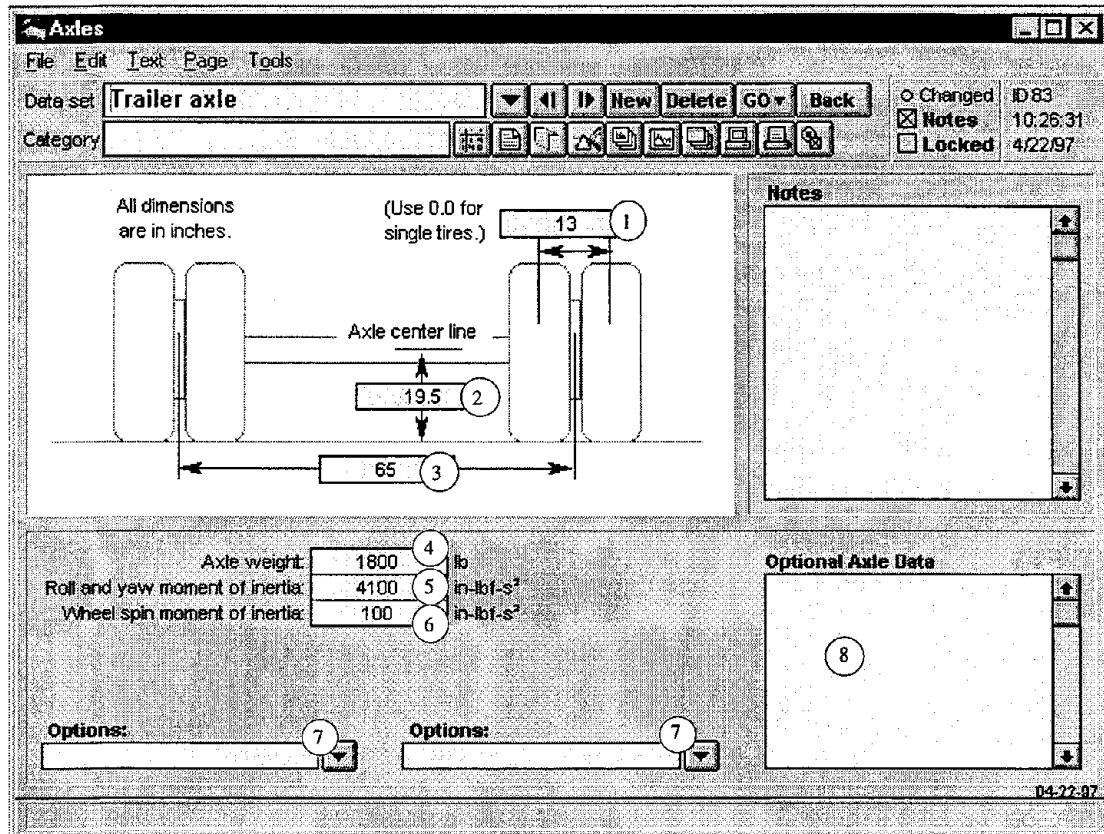
`Animate\Wheels\Wheels.tbk`

Axles

Use this screen to specify the relevant modeling parameters of the solid axles used on trucks and trailers. This library is not used for the car model in VDM RoAD.

Discussion

You must describe the inertial and dimensional properties of **every** axle. The “axle” is considered to be the entire unsprung mass (axle, brakes, wheels and tires), even though additional properties for these components are specified on other screens.



User Settings

- ① Dual tire spacing (keyword = `LDUAL (IUNIT, IAXLE)`). Lateral distance between the centerlines of the two tires of a dual wheel set. If the axle has single tires, use a spacing of zero.
- ② Height of the axle spin axis (keyword = `HCGA (IUNIT, IAXLE)`). This value is used to establish the relationship between wheel spin and forward speed, and it is also used for the height of the mass center. The two numbers are so close that only a single value is used, to avoid potential errors.
- ③ Track width (keyword = `LTK (IUNIT, IAXLE)`). Lateral distance between the centers of dual wheel sets. If the axle has single tires, this is the distance between the centers of the two tires.
- ④ Axle Mass (keyword = `MUS (IUNIT, IAXLE)`). Mass of the axle with attached hardware such as brakes, wheels, and tires. This is the effective mass of all components that move as an "unsprung" unit. Therefore, half of the mass of connecting hardware (e.g., springs and control arms) is usually included.
- ⑤ Roll and Yaw Moments of Inertia (keyword = `IA (IUNIT, IAXLE)`). Because of symmetry of the axle, the moments of inertia about these two axes are so similar that only one value is used.

- ⑥ Wheel Spin Moment of Inertia (keyword = IW (IUNIT, IAXLE)). Combined moment of inertia of all rotating parts on one wheel, including tires, the wheel, brake drum, and axle shaft.
- ⑦ Optional links. These links support customized axle models but are not used in the standard VDM RoAD.
- ⑧ Optional miscellaneous field to support customized axle models. Not used in the standard VDM RoAD.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Vehicle Unit: (empty truck, tractor, or trailer)

Suspension: Solid Axle

File Location

Vehicles\Parts\Axles\Axles.tbk

Brake System

Use this screen to define the pneumatic delays and ABS parameters for the truck brake system.

This library is not used for the car model in VDM RoAD.

Discussion

You use this screen to enter the properties of the brake application system for any vehicle or unit of a vehicle combination (tractors and semitrailers). The important properties are the time delay for brake application pressure to reach a brake, and the properties of the Anti-Lock Brake System (ABS). This screen does not include the brake torque properties (i.e., the torque generated as a function of application pressure), which are input on the **Brakes** screen.

The ABS system is designed to prevent the wheels from completely locking during hard braking. When a wheel starts to lock up, its longitudinal slip decreases from 0 to -1. When the slip decreases past some level, a valve releases the air pressure to the brake, allowing the wheel to spin back up. Then, when the slip has approached zero, the valve allows the air pressure to actuate the brake again. At very low speed, just before the vehicle stops, the ABS is disabled and wheel lockup is allowed. Actual ABS control logic is usually proprietary and is complicated because it must deal with many unusual circumstances. The VDM RoAD truck models include a simple logic that results in realistic cycling of the brake system.

The cycling frequency during a limit stop in which the ABS controller is active is typically between 2 cycles/sec and 10 cycles/sec, depending on the design and the time lags in the mechanical brake due to the air flow.

Brake System

File Edit Text Page Tools

Data set: 6x4 Tractor

Category:

o Changed ID 45
 Notes 4:29:42
 Locked 11/19/96

Time Delays for Air Brake Systems (by Axle)

Axle number:	1	2	3
Time delay:	.05	.1	.5

sec

Air-actuated brake experience a delay from the time that an input is generated by the driver, due to the time needed for air to travel through tubes from the air supply to the brakes. The delay is related to the speed of sound and the length of tubing connecting the supply to the chambers of each axle.

Generic ABS System Parameters

Axle number:	1	2	3
Slip OFF:	0.51	.51	.51
Slip ON:	.31	.31	.31
Speed Limit:	4	4	4

mph

The generic ABS model controls each wheel independently. The same three parameter values are used for the brakes on each axle. When the dimensionless slip ratio exceeds the value called Slip OFF (0=free rolling, 1=lock up), the brake is turned off. When the wheel spins up and the slip drops to the value called Slip ON, the brake is turned back on. When the speed drops below the specified limit, the ABS logic is disabled and the brake is on.

Options:

Options:

Notes

Optional Brake System Data

11-20-96

User Settings

- ① Time Delays (keyword = TBRKLAG (IUNIT, IAXLE)). These are the times required for the air pressure signal from the Brake Input to reach the brake actuators on each axle.
- ② Antilock brake system (ABS) parameter: longitudinal slip for turning OFF the brake (keyword = ABS_SLIPOFF (IUNIT, IAXLE)). Defines a wheel slip condition at which brakes are turned off (0=free rolling, 1=lock up). To disable the ABS control, set this parameter to a value larger than 1. A typical value would be about 0.2.
- ③ Antilock brake system (ABS) parameter: longitudinal slip for turning ON the brake (keyword = ABS_SLIPON (IUNIT, IAXLE)). This is the wheel slip condition at which the brakes are again turned on (0=free rolling, 1=lock up). It should be lower than the value for ABS_SLIPOFF. A typical value would be about 0.1.
- ④ Speed limit (keyword = ABS_VMIN (IUNIT, IAXLE)). This is the vehicle speed below which the ABS system is turned off and wheel lock is allowed. A typical value would be around 5 to 10 km/h.

- ⑤ Optional links. These links support customized braking models but are not used in the standard VDM RoAD models.
- ⑥ Optional miscellaneous field to support customized braking models. Not used in the standard VDM RoAD models.

Location in VDM RoAD

Truck Models

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Vehicle Unit: (empty truck, tractor, or trailer)

Brake System

Car Model

VDM RoAD Startup

Runs

Cars

Brake System

File Location

Vehicles\Parts\Brks_trk\Brks_trk.tbk

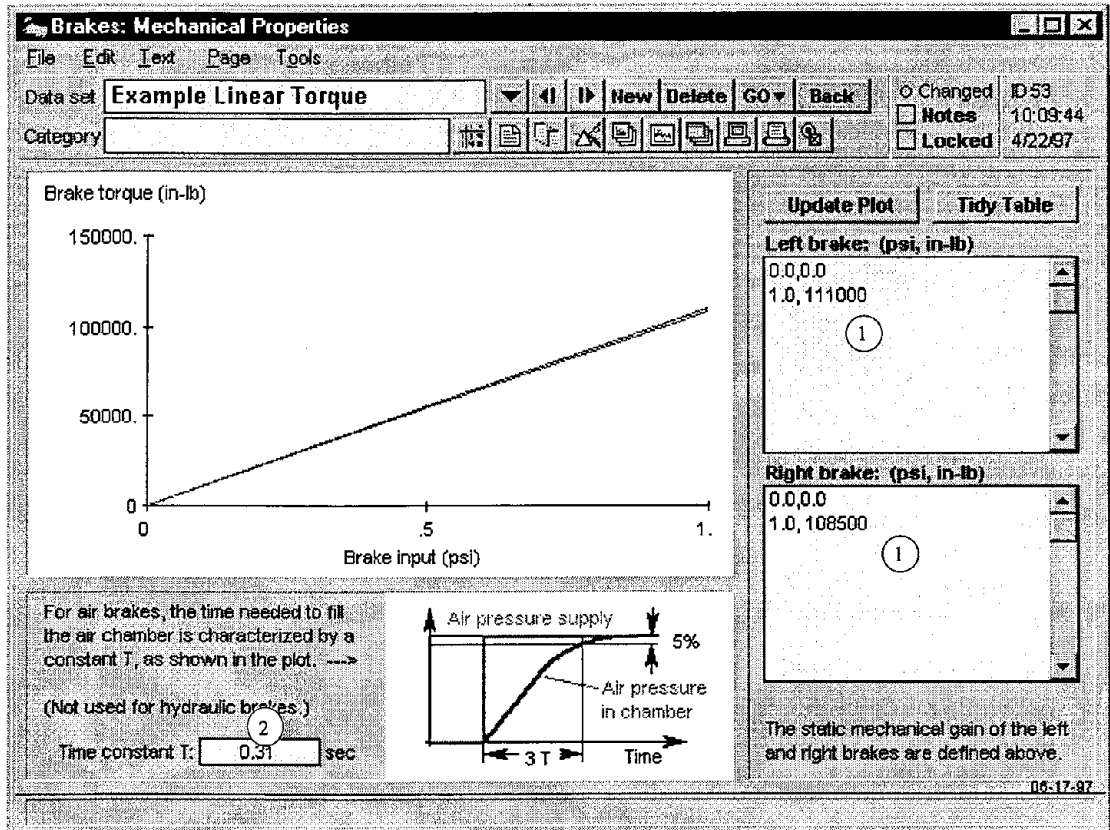
Brakes: Mechanical Properties

Use this screen to specify the torque properties of the brakes as a function of brake input (application) pressure and the fill characteristics of the air chambers.

Discussion

You define brake torque properties by a table look-up of torque, on the vertical axis, as a function of brake input, on the horizontal axis. The brake input corresponds to the pressure values shown on the **Input: Braking** screen. For truck simulations, the input pressure is delayed in time in accordance with the time delay (transport delay) given on the **Brake System** screen.

A single library contains brake mechanical data for all VDM RoAD models (cars and trucks). However, the time constant parameter is not used for car simulations.



User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Table field for brake torque properties (keyword = MYBK_TABLE (IUNIT, IAXLE, ISIDE) ; for ISIDE, 1=left side, 2=right side). The brake torque properties are defined by air chamber pressure values and the associated brake torque.

The simulation programs use constant-gradient extrapolation with this table. For values of pressure that are less than the range covered in the table, the first point is used to extrapolate the first gradient backward. The gradient is determined from the first two values of pressure and torque. For values of pressure larger than the range covered, the last point is used to extrapolate forward using the gradient between the last two points. The table needs at least two lines (for the simplest case of a constant ratio between torque and pressure).

- ② Time constant of the first-order lag model for filling the brake air chambers (keyword = TBRKTC (IUNIT, IAXLE)). The input to the chamber is the pressure from the pressure-time table defined in the **Input: Braking** library, modulated by the ABS controllers defined in the **Brake System** library.

This parameter is ignored when the data set is applied to the car model, because the car model has a hydraulic brake system in which delays in the fluid system are neglected.

Location in VDM RoAD

Truck Models

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Vehicle Unit: (empty truck, tractor, or trailer)

Brakes: Mechanical Properties

Car Model

VDM RoAD Startup

Runs

Cars

Brakes: Mechanical Properties

File Location

Vehicles\Parts\Brk_trq\Brk_trq.tbk

Calculator

Use this screen to create and edit tabular numerical data. It is available from a button on the ribbon bar and from the **Tools** menu.

Discussion

This data screen is a tool supplied in VDM RoAD because there are many times when preparing VDM RoAD inputs that simple calculations are needed. This screen can be used to convert units for existing data, or to create tables from scratch. For example (explained in detail later), the screen display shows how a series of X-Y coordinates are created for a sinusoidal path.

This screen does not directly feed data to a simulation model. To use the data, you must copy the numbers to the clipboard and then paste them into an appropriate data screen.

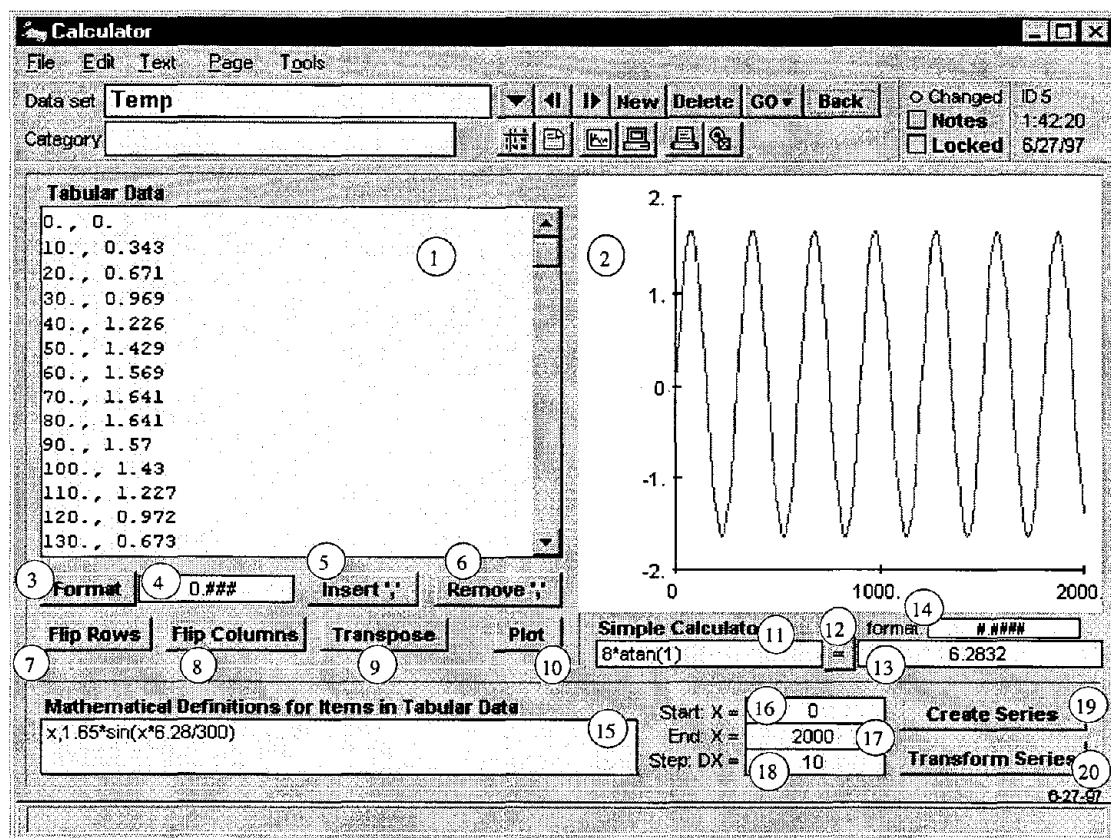


Figure 8.1. The calculator screen.

The screen design is fairly complex, because the calculator tool has several capabilities:

- It plots tabular data for one or more pairs of X-Y data points.
- It performs simple calculations by evaluating a mathematical expression and printing the numerical result.
- It evaluates mathematical expressions involving values in a 2D table, updating the entire table with a single button click. For example, you can convert the units for different columns in a table using different scale factors for each column.
- It creates tabular data from scratch, based on start and end conditions for a series.

After the user settings are described, a few examples are provided to show how they are used.

User Settings and Controls

- ① **Tabular Data** field. This is where the tabular data values are created and edited. The data are assumed to fit in a tabular organization, with each line in the field representing a row. Each row should have the same number of items (columns), separated by commas and/or spaces. All values in the table must be numbers.

The values in this field are shown graphically in the adjacent plot (2).

The buttons underneath (3) - (9) are used to change the precision of the numbers, determine whether commas separate the numbers, and to modify the column-row structure of the table.

The fields and buttons in the bottom area of the screen (15) - (20) are used to perform calculations to create the tabular data from scratch, or to transform existing values.

The only part of the screen that is not related to the tabular data field is the simple calculator, with items (11) - (14).

- (2) **Plot.** This graphic is created by plotting values of columns 2 and higher on the Y axis, against the values of column 1 on the X axis. Thus, the number of plots is N-1 where N is the number of columns. The plot is not made automatically—you must click the **Plot** button (10) to create it or update it after modifying the tabular data (1).
- (3) **Format** button. Click to format the numbers in the tabular data field (1) according to the format string (4).
- (4) **Format field.** The text in this field specifies whether numbers are written with fixed decimal places or in scientific notation. It also specifies how many digits are written to the right of the decimal point. The format string has the form: X.X where the X's represent a series of place holders:
 - 0 — place holder for a digit.
 - # — Same as 0 except the digit is not printed unless it is needed.
 - E+ or e+ — Scientific notation with sign always used for the exponent. The number of digits in the exponent is determined by the number of place holders.

If the format field is blank, then numbers are written to their full precision.

Following are some example format strings and the effect they have on three example mathematical expressions.

String	45/atan(1)	atan(1)/45	5/4
none	57.2957795130823	1.74532925199433e-2	1.25
#####	57.2958	0.0175	1.25
0.0000	57.2958	0.0175	1.2500
#####e+###	5.729578e+1	1.745329e-2	1.25 e+0
0.000000E+000	5.729578E+001	1.745329E-002	1.250000E+000

The format field is applied when you click the **Format** button (3), the **Create Series** button (19), or the **Transform Series** button (20).

- (5) **Insert ','** button. Click to ensure that all numbers in a row are separated with commas. If commas already exist, this has no effect except possibly to tidy the appearance of the table (1).

- ⑥ **Remove ‘,’** button. Click to ensure that all numbers in a row are separated with white space only. If commas exist, they are removed. If numbers are already separated only by white space, this has no effect.
- ⑦ **Flip Rows** button. Click to reverse the order of the lines in the tabular data field ①.
- ⑧ **Flip Columns** button. Click to reverse the order of the numbers in each line in the tabular data field ①.
- ⑨ **Transpose** button. Click to transpose the columns and rows in the tabular data field ①. For example:

1,2,3	Becomes	1,5
5,6,7		2,6
		3,7

- ⑩ **Plot** button. Click this to update the graphic ② based on the current contents of the tabular data field ①. The graphic is created by plotting values of columns 2 and higher on the Y axis, against the values of column 1 on the X axis. Thus, the number of plots is N-1 where N is the number of columns.
- ⑪ **Calculator input field**. Enter a mathematical expression, then click the “=” button ⑫ to see the result in the calculator output field ⑬. In addition to normal arithmetic operations, the calculator has a number of built-in functions that are described in a later subsection.
- ⑫ **Calculate** button. Click to evaluate the expression in the input field ⑪ and print the result in the output field ⑬.
- ⑬ **Calculator output field**. This displays the results of the calculation performed when you click the “=” button ⑫. The format can be specified using the format string ⑭.
- ⑭ **Calculator output format**. This field contains a format string used to control the round-off in the calculator output ⑬. It is applied when you click the “=” button ⑫. The syntax is the same as the other format field ④, described previously.
- ⑮ **Mathematical definitions**. This field is used to create or transform numbers in the tabular data field ①. The field is used for two distinctly different purposes:
 1. It provides mathematical definitions of the numbers that will be calculated and stored in the tabular data field when you click the **Create Series** button ⑲. In this case, all definitions must involve only numbers and a single variable X. X is calculated automatically and given a value for each new row based on the three fields ⑯, ⑰, and ⑱.
 2. It provides transforms of existing numbers that will be calculated when you click the **Transform Series** button ⑳. In this case, the definitions must involve numbers and the variables X, Y, Y2, Y3, ..., where X is the first number in each line of the tabular data field, Y is the second number, Y2 is the third, and so on.

In the first case, the field is cleared and all new numbers are put into it, based on the series information (⑯, ⑰, and ⑱) and the mathematical definitions. In the second case, the tabular data field is not cleared—the existing numbers are replaced.

In both cases, the tabular data field will have the same number of items as the mathematical definitions field. Initially, items (numerical results of the calculations) are separated by commas. (Use the **Remove ‘,’** button (6) to remove the commas if needed.)

In both cases, the variable X can be included in the expressions. The meaning is not always the same. In the first case, X is an arbitrary independent variable whose values are defined by the range and interval specified in the fields (16), (17), and (18). In the second case, X is the first number in the tabular data field (1).

The variables Y, Y2, etc., are recognized only in the second case. They should not be used when generating new numbers with the **Create Series** button (19).

The variables and function names used in this field are not sensitive to case: x and X refer to the same variable.

Several examples are provided in the next subsection to indicate how you might use this field.

- (16) **Start value.** This is the value assigned to X for the first row of numbers created in the tabular data field (1) when you click the **Create Series** button (19).
- (17) **End value.** This is the value assigned to X for the last row of numbers created in the tabular data field (1) when you click the **Create Series** button (19).
- (18) **Step.** This is the interval used to calculate X for every row between the first and last in the tabular data field (1) when you click the **Create Series** button (19).
- (19) **Create Series button.** Click to replace the contents of the tabular data field (1) with rows of numbers. Each row will have one number for each item in the mathematical definitions field (15). The number of rows is equal to $(\text{End} - \text{Start} + 1) / \text{Step}$, where the values of Start, End, and Step are obtained from the fields (16), (17), and (18).
- (20) **Transform Series button.** Click to transform the contents of the tabular data field (1). When the transformation is complete, each row in the tabular data field will have one number for each item in the mathematical definitions field (15). The number of rows in the tabular data field is not changed. The series Stop, End, and Step fields ((16), (17), and (18)) are not used in these calculations.

Examples

Generate a Sine Wave

To generate a set of X-Y values that define a sine wave with a period of 300 and an amplitude of 1.65, do the following (see Figure 8.1):

1. Set Start X (16) to 0.
2. Set End X (17) to 2000.
3. Set Step DX (18) to 10 (this will create a table with 201 points).
4. Define the two variables in the mathematical definitions field (15):
 $x, 1.65 * \sin(x * 6.2832 / 300)$.

5. Set the format string (4) to: 0 . ### .
6. Click the **Create Series** button (19).
7. Check your work by clicking the **Plot** button (10).

Rescale the Sine Wave

Suppose the sine wave has an amplitude of 1.65 m, and you want to increase it by 20%.

1. Define the transformation in the mathematical definitions field (15): $X, Y*1.2$.
2. Click the **Transform Series** button (20).
3. Check your work by clicking the **Plot** button (10).

Available Functions

In addition to standard arithmetic operators, the compute button can process the following functions:

abs(<number>) Absolute value of number.

acos(<number>) Arccosine (number in radians).

asin(<number>) Arcsine (number in radians).

atan(<number>) Arctangent (number in radians).

atan2(<number1>,<number2>) Arctangent of <number1> divided by <number2> (numbers in radians).

average(<list of numbers>) Returns the sum divided by the number of items in the list

ceiling(<number>) Rounds up to nearest integer.

cos(<number>) Cosine (number in radians).

cosh(<number>) Hyperbolic Cosine (number in radians)

exp(<number>) Exponent (2.7182818) raised to power of number.

floor(<number>) Rounds down to nearest integer.

hypotenuse(<length>,<length>) Length of hypotenuse.

ln(<number>) Natural Log (base e)

log(<number>,<base>) Log of number in base.

max(<list of numbers>) Highest value in list.

min(<list of numbers>) Lowest value in list.

round(<number>) Rounds to nearest integer.

sin(<angle>) Sine of an angle (in radians).

sinh(<angle>) Hyperbolic sine of an angle (in radians).

sqrt(<number>) Square root of a positive number.


sum(<list of numbers>)

tan(<angle>) Tangent of an angle (in radians).

tanh(<angle>) Hyperbolic tangent of an angle (in radians).

truncate(<number>) Truncates to integer.

Location in VDM RoAD

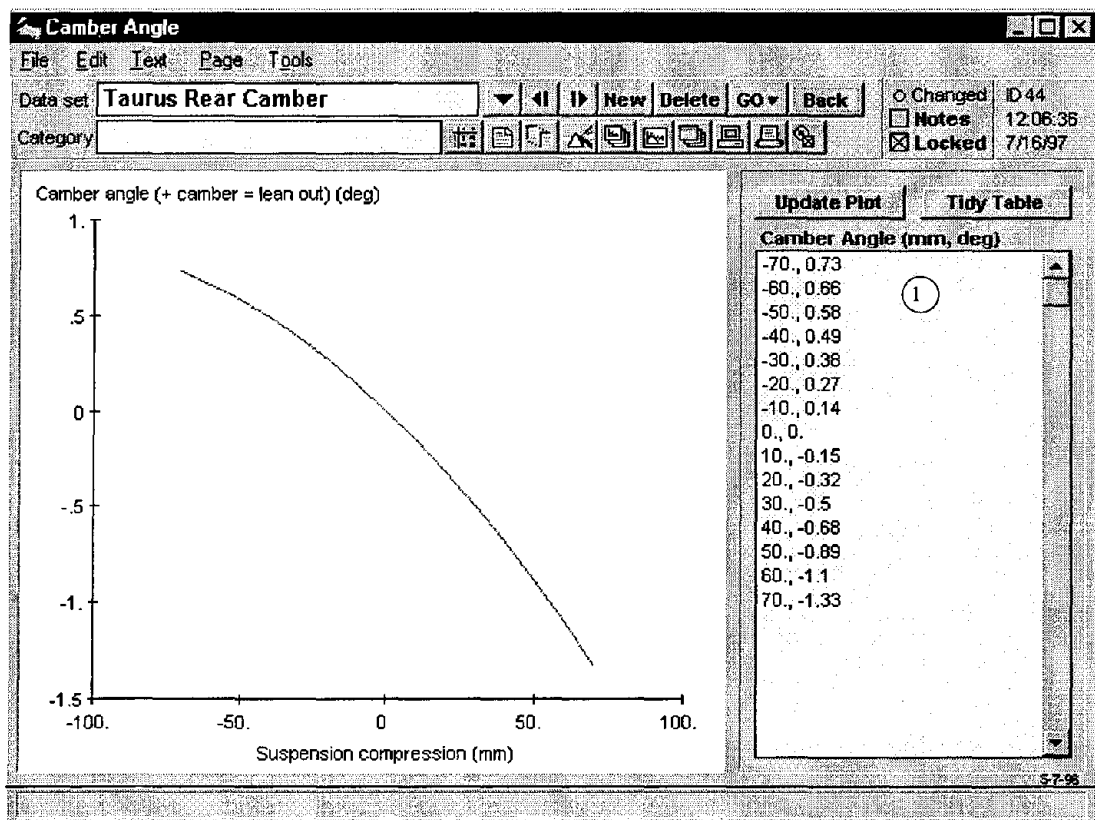
Accessed from the **Tools** menu or the ribbon bar with the button: 

File Location

Sgui_lib\calc.tbk

Camber Angle

This screen has the table look-up that defines the wheel camber angle for an automotive independent suspension as a function of suspension jounce (compression). This library is not used for the truck models in VDM RoAD.



Discussion

The suspension kinematics in the car model in VDM RoAD include the camber angle change as a function of suspension travel. In general, camber decreases as suspension compression increases.

User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Table of values of wheel camber angle as a function of suspension compression (keyword = CAMBER_TABLE). Suspension compression is in mm and the camber angle is in degrees.

The sign convention for suspension travel (labeled *Suspension compression* on the screen) is that a positive value puts the suspension in jounce. Positive camber occurs when the wheels lean out (are further apart at the top than at the bottom).

This table uses constant-gradient extrapolation. For values of compression that are less than the range covered in the table, the first point is used to extrapolate the first gradient backward. The gradient is determined from the first two values of compression and camber. For values of compression larger than the range covered, the last point is used to extrapolate forward using the gradient between the last two points. The table needs at least two lines (for the simplest case of a constant ratio between camber and compression).

Location in VDM RoAD

```
VDM RoAD Startup
  Runs
    Cars
      Suspensions
        Camber Angle
```

File Location

```
vehicles\parts\camber\camber.tbk
```

Cars

Use this screen to define the basic properties of a passenger car and links to its suspension system, axles, wheels, tires, and brakes. This library is not used for the truck models in VDM RoAD.

Discussion

This is the main data screen for building a car or light vehicle. It has the weight and dimensions, plus links to other libraries with component data. It also has a link to data for the animator.

Cars
File Edit Text Page Tools

Data set: **P (AASHTO Pass. Car)** | New | Delete | GO | Back

Category: [] | [] | [] | [] | [] | [] | [] | [] | [] | []

○ Changed ID 54
□ Notes 1:44:20
□ Locked 10/15/97

Diagram: mass center of entire vehicle, dimensions are in millimeters. 1320 (1), 3300 (2)

Masses and Inertias Front and Rear

	Front (3)	Rear	Unit
Vehicle mass supported by tires:	1460	2490	kg
Unsprung mass (both sides):	220	220	kg
Spin moment of inertia for one wheel:	2	2	kg-m ²

Inertia Properties: Entire Vehicle

Roll Inertia (Ixx):	6	1669	kg-m ²
Pitch Inertia (Iyy):	7	6172	kg-m ²
Yaw Inertia (Izz):	8	6172	kg-m ²
Product (Ixz):	9	0.0	kg-m ²

(Optional) Radii of Gyration

Estimate Ixx (10)	Rx:	11	.65	m
Estimate Iyy (10)	Ry:	12	1.25	m
Estimate Izz (10)	Rz:	13	1.25	m

Estimate inertia with eq. $I = M \cdot R^2$

Steering System (17): Taurus Steering

Aerodynamics (18): Taurus Aero Data (drag only)

Animator Group (19): P Car (HMMVV)

Notes

Front Suspension (14): HMMVV Front Suspension

Rear Suspension (14): HMMVV Rear Suspension

Front Tire: table-lookup model (15): Taurus Tires

Rear Tire: table-lookup model (15): Taurus Tires

Front Brakes (16): Taurus front

Rear Brakes (16): Taurus rear

07-13-97

User Settings

- ① Height of mass center of entire vehicle above the ground (keyword = HCG).
- ② Wheelbase (keyword = LWB). Distance from the center of the front axle to the center of the rear axle.
- ③ Vehicle mass supported by wheels (keyword = MF for front two wheels and MR for rear two wheels). These are the masses supported by ground below both front wheels and both rear wheels, as would be measured when the front and rear of the car are alternatively driven over a scale.
- ④ Front and rear unsprung masses (keyword = MUS (IAXLE)). These are the sums of the front two unsprung masses and the sum of the rear two unsprung masses.

- ⑤ Spin moment of inertia of one wheel (keyword = IW (IAXLE)).
- ⑥ Roll Moment of Inertia of entire vehicle (keyword = IXX). The moment is taken about the mass center, normal to an X axis (longitudinal) that is parallel with the ground when the vehicle is at rest on a flat level surface.
- ⑦ Pitch Moment of Inertia of entire vehicle (keyword = IYY). The moment is taken about the mass center, normal to a Y axis (lateral) that is parallel with the ground when the vehicle is at rest on a flat level surface.
- ⑧ Yaw Moment of Inertia of entire vehicle (keyword = IZZ). The moment is taken about the mass center, normal to a Z axis (vertical) that is parallel with the gravity vector when the vehicle is at rest on a flat level surface.
- ⑨ X-Z Product of Inertia of entire vehicle (keyword = IXZ). The product is taken about the mass center, and is defined as the negative of the volume integral over X*Z dV ($I_{xz} = -\text{Int}(\text{RHO} * X * Z * DV)$). The product is positive when the principle X axis tilts down (looking forward).
- ⑩ Calculator button. Click this to calculate IXX, IYY, and IZZ from the radii of gyration RX, RY, and RZ. Sometimes measured values for IXX, IYY, and IZZ are not available and RX, RY, and RZ can be estimated to compute the inertia parameters based on the total mass ($I = M * R^2$).
- ⑪ Roll Radius of Gyration (not a parameter). Use if IXX is not available and RX can be estimated. If used, a guess for Rx might be a third of the vehicle width.
- ⑫ Pitch Radius of Gyration (not a parameter). Use if IYY is not available and RY can be estimated. If used, a guess for Ry might be half the wheelbase.
- ⑬ Yaw Radius of Gyration (not a parameter). Use if IZZ is not available and RZ can be estimated. If used, a guess for Rz might be half the wheelbase.
- ⑭ **Suspension** links. Link to the screen that sets the suspension properties.
- ⑮ **Tire** links. Link to the screen that sets the tire properties.
- ⑯ **Brakes** links. Link to the screen that sets the brake properties.
- ⑰ **Steering System** link. Link to the screen that sets parameters for the steering system.
- ⑱ **Aerodynamics** link. Link to the screen that sets the aerodynamics properties.
- ⑲ **Animator Group** link. Link to an animator group with a reference frame and shape data needed to animate the vehicle. The animation data are not used during the simulation, but are required for viewing the simulation results afterwards using the animator program in VDM RoAD.

Location in VDM RoAD

VDM RoAD Startup

Runs

Cars

File Location

Vehicles\Cars\Cars.tbk

Computation Parameters

Use this screen to set parameters for the numerical integrator and to control the frequency and format of output.

Discussion

The simulation programs operate by numerically integrating a set of nonlinear differential equations over time. Over small time intervals, they approximate the integration using a numerical integration algorithm.

This data screen contains parameters that have little to do with the vehicle model or its inputs. Rather, they are related to the numerical solution method and to the form of the output file used to store the computed results.

Computation Parameters

File Edit Text Page Tools

Data set: **Stiff integrator: dt = .002** | New | Delete | GO | Back

Category: | Changed | D:21
 Notes | 11:13:34
 Locked | 6/5/97

Main Integration Parameters

Integration time step: 0.002 sec (1)
Integration method: 1 (2)

The integration method should be 0, 1, 2, or 3:
0 - Adams predictor-corrector (VS,VO*)
1 - Gear (VS,VO,S, occasional perturbation computations)
2 - Gear (VS,VO,S, frequent perturbation computations)
3 - RK2 (Runge-Kutta, second-order, no error checking)
*VS=variable step, VO=variable-order, S=OK with stiff eqs.

Variable-Step Parameters*

Integrator error tolerance: 0.01 (3)
Initial integration Step-size: 0.00001 sec (4)
Minimum Allowable Step Size: 0.0000001 sec (5)
*Not used for RK2 (method=3)

Options 0, 1, and 2 are further defined in "Gear, CW. *Numerical Initial Value Problems in Ordinary Differential Equations, 1971*."
Option 3 usually runs the fastest for vehicle models. However, you must occasionally verify that the results are valid.

Output Parameters

Print Interval: 20 (6)
Output file format: (7)

Print Interval

The sample interval in the output plot files is the product of the integration time step and the print interval. Use 1 for methods 0,1,2, and use larger values of method 3.

Output File Format

Three output file formats are available:
BINARY = ERD file with BINARY data file. This is the most efficient and should be used for most runs.
TEXT = spreadsheet text file, with a 1-line header and numbers separated by commas. The animator does not work with TEXT files. This should be used for simple export to analysis software (spreadsheets, math packages, etc.)
ERDTEXT = ERD text file, which is readable by external programs and also contains the header information needed by the animator and plotter.

06-10-97

User Settings

- ① Integration time step (keyword = `STEP`). The VDM RoAD solver programs generate output file by solving equations of motion in a mathematical model of the vehicle. They repeat the calculations at small intervals of simulated time, where the interval is called a time step. The time needed to run a simulation is inversely related to the time step. If you increase the time step by a factor of two, the program runs about twice as fast because it only makes half as many calculations. However, the calculation method is valid only for time steps that are “sufficiently small.” For typical truck models, “sufficiently small” means about 0.002 seconds. If the springs are unusually stiff, a time step of 0.001 may be required.

Note: If you are making many runs with a few vehicle descriptions, you may want to determine how small the time step needs to be. Make several runs changing only the time step. For example, try using values of 0.004, 0.002, 0.001, and 0.0005 second. If the runs are valid, plots of the same variables should overlay perfectly. If the results at a large time step differ from those made with a small time step, the typical assumption is that the results made with the large time step are in error.

- ② Integration method (keyword = `INTOPT`). The VDM RoAD models have several integration methods built in. Method 3 (second-order fixed step Runge-Kutta) usually gives the best results (i.e., run faster) for truck models with closed-loop controls or any form of braking. However, if you are making runs with step inputs or other conditions where there is some dynamic behavior followed by many seconds of nearly static behavior, then Method 2 (Gear) may give equivalent accuracy with a bigger time step and less time.

The variable step methods often allow larger time steps to be used, but they fail if the time step is set too large. Time steps larger than 0.01 sec are likely to cause problems, except for long runs with little change in the inputs. The limits depend both on the vehicle parameters and the types of inputs, so trial and error are needed to determine the largest step size that will work, even when using the variable step methods.

- ③ Integrator error tolerance (keyword = `EPSINT`). Not used if ② is equal to 3.
- ④ Initial integrator step size (keyword = `STEP0`). Not used if ② is equal to 3.
- ⑤ Minimum allowable integration step size (keyword = `STEPMIN`). Not used if ② is equal to 3.
- ⑥ Number of time steps between output printing (keyword = `IPRINT`). The time step ① determines how often calculations are made. The time interval for the output files is the product of the step and this interval (`IPRINT`). When the RK2 integration method (number 2) is used, a print interval between 10 and 30 is recommended. When a variable step method is used, you can sometimes set the step ① to be the intended output interval and set `IPRINT = 1`.

Notes: This parameter determines the resolution for plots and also the speed for the animator. To make the animated motions runs faster, use a larger value of IPRINT. To run in slow motion (say, to accurately see wheel rotations) use a small value.

This parameter has only a minor effect on the speed of a simulation run. However, it has a direct effect on the size of the output files and the time needed to load them into the plotter or animator. Big files take longer!

- ⑦ Output file format (keyword = FORMAT). It is a keyword (BINARY or TEXT or ERDTEXT):

BINARY — the solver program creates a binary output file (extension = BIN) and an ERD header file. This option is the most efficient, resulting in less time needed to run the simulation and view output, together with less disk space being needed to store information.

TEXT — the solver program creates a simple text output file that can be imported into other software packages such as spreadsheets (Excel, Lotus, etc.) and mathematical analysis programs (MATLAB, etc.). The first line contains short names for the output variables, separated by commas. The following lines contain numbers separated by commas. Each row has all the values for a single point in time. These files cannot be viewed with the animator. WinEP can read text files, although the plots contain less information because the file has no title, no units, etc.

ERDTEXT — the solver program creates a text ERD file. It has all the labeling information required by the animator and plotter, followed by the printed values of all output variables. These files can be imported into other programs (with some editing of the header information) and they can also be viewed within VDM RoAD using the standard VDM RoAD controls.

Note: Text files are not recommended for routine use of VDM RoAD. The solver programs take longer to write them, and the plotter and animator programs take significantly longer to read them. Also, they occupy about three times as much disk space as the binary files.

Location in VDM RoAD

VDM RoAD Startup
 Runs
 Computation Parameters

File Location

Comp_par\Comp_par.tbk

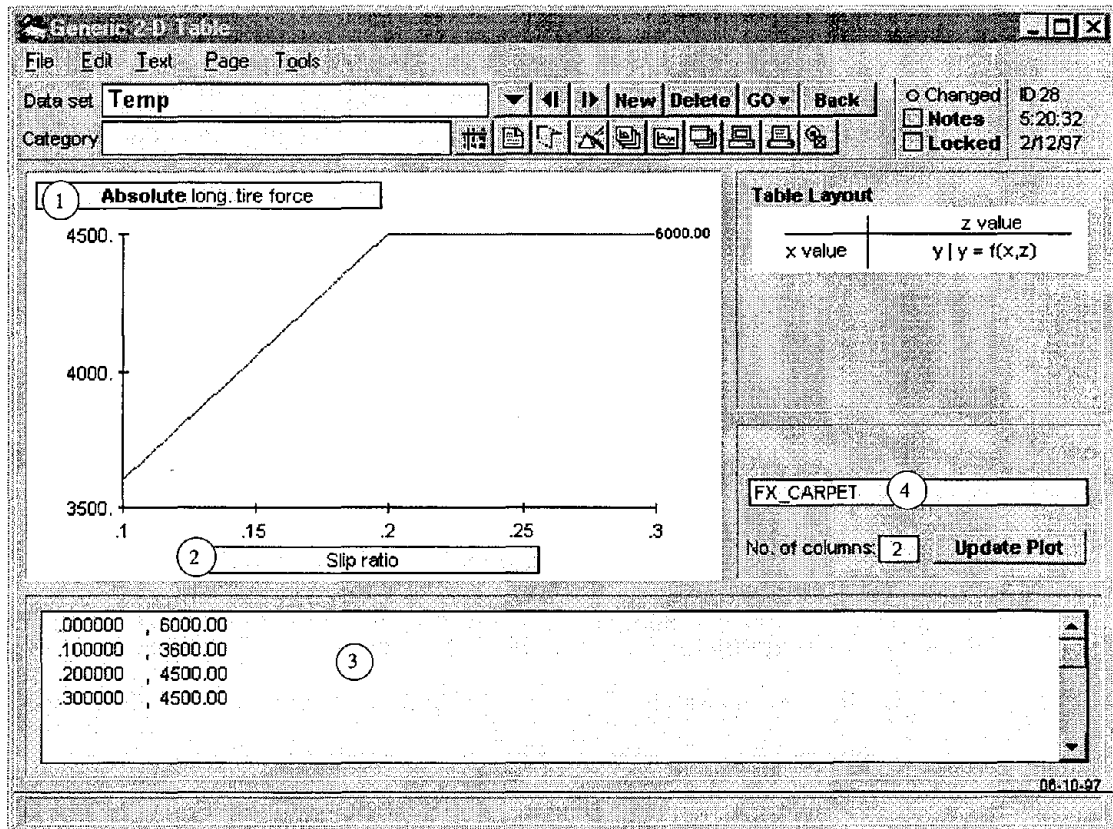
Generic 2D Table

Use this screen to store and display tabular data involving two independent variables for custom VDM RoAD models. This library is not needed for the standard VDM RoAD models.

Discussion

All tables in the standard VDM RoAD models are associated with libraries in the VDM RoAD data base. However, VDM RoAD can be extended to include new models. If the specialized models involve tabular data with two independent variables, then this library can be used to store the data.

Unlike most of the libraries in VDM RoAD, each data set in this library can represent a different kind of data. This is possible because the keyword used by the simulation solver programs to identify tabular data is a part of the data set.



User Settings

Note: User settings that are common for all 2D tabular data screens are described in Chapter 7, in the section **2D Tabular Data**.

- ① Label for Y axis. This label is not used by the simulation programs. It is a user-comment field, provided as a means for documenting the information on the screen.
- ② Label for X axis. This label is not used by the simulation programs. It is a user-comment field, provided as a means for documenting the information on the screen.
- ③ Tabular data. The first row has values of the first independent variable, the first column has values of the other independent variable, and all other numbers are values of the dependent variable.

For values of X outside the range of the data, the form of extrapolation depends on the use made of the data. For table functions that involve vehicle properties, the data are extrapolated using the closest gradient of Y relative to X. On the other hand, for table functions that define inputs (controls and disturbances) to the vehicle, flat-line extrapolation is used. The value returned is either the first or last value of the dependent variable.

- ④ Keyword. This keyword is required for a simulation program to make use of the data in the table ③. Unlike most VDM RoAD libraries, the keyword is not hidden. If the keyword in this field is not recognized by the simulation program, then the data will not be used.

Location in VDM RoAD

This library does not have a default position in VDM RoAD. You can link to it from any blue field in the VDM RoAD libraries.

File Location

Generic\Gen2dtab\Gen2dtab.tbk

Generic Group Data

Use this screen to create groups of data.

Discussion

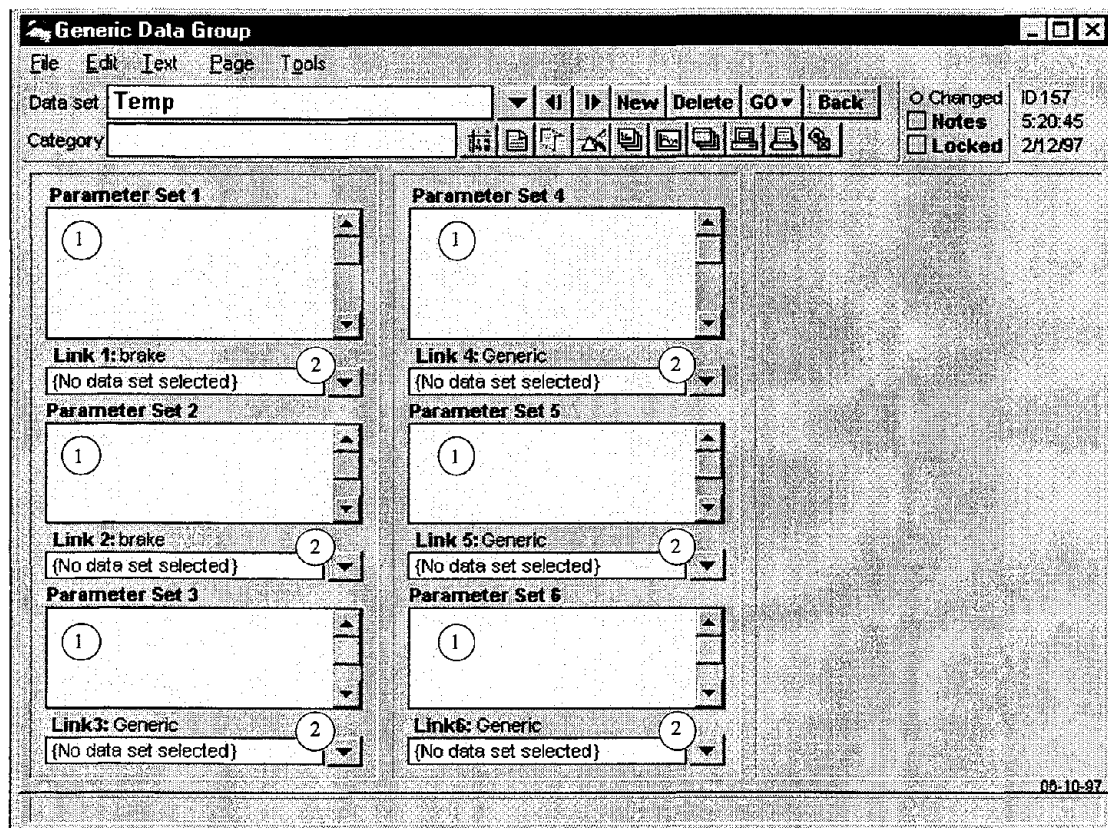
There are at least three applications for this library.

1. Create sets of related inputs, such as combinations of braking, steering, and speed.
2. Create sets of vehicle parameters that you want to apply as a group. For example, you could make a group that applies alternate tire data for every axle in a vehicle.
3. Specify parameter values that do not fit in existing data screens. For the standard VDM RoAD package, there are only a few such parameters: friction (MU) and a few simulation control values (ROLL_STOP, V_STOP). However, if you add new vehicle models, then you might want to use this library to set values for additional parameters that do not exist in the standard models.

When combining parameters and other data sets, be aware that the data are sent to the solver programs in the same order as they are numbered on the screen: top to bottom, then left to right.

Many of the truck components are used more than once (tires, axles, springs, etc.), and a keyword is used to determine where the description of the component should be applied. For example, the keyword IAXLE is used to associate axle-related data sets to axles. Put the line "IAXLE 3" in a yellow field, and then any links that follow will be associated with axle 3. The screen layout includes six pairs of yellow fields and blue links. Therefore, it is possible to specify data for six different places.

The best way to determine how keywords are used to locate data sets is by viewing an echo file produced by the simulation program you will use (see Appendix D for an example echo file).



User Settings

- ① Miscellaneous parameter set fields. Enter keywords and the value you want assigned to them. The format is that each line has a keyword and value, separated with white space (at least a single space). These keyword values for vehicle units remain the same (IUNIT, IAXLE, ISIDE) until the keyword is used again.

- ② Links are used to include other VDM RoAD data sets.

Note: You can link to other generic data groups, if the number of fields and links on the screen is not sufficient.

Location in VDM RoAD

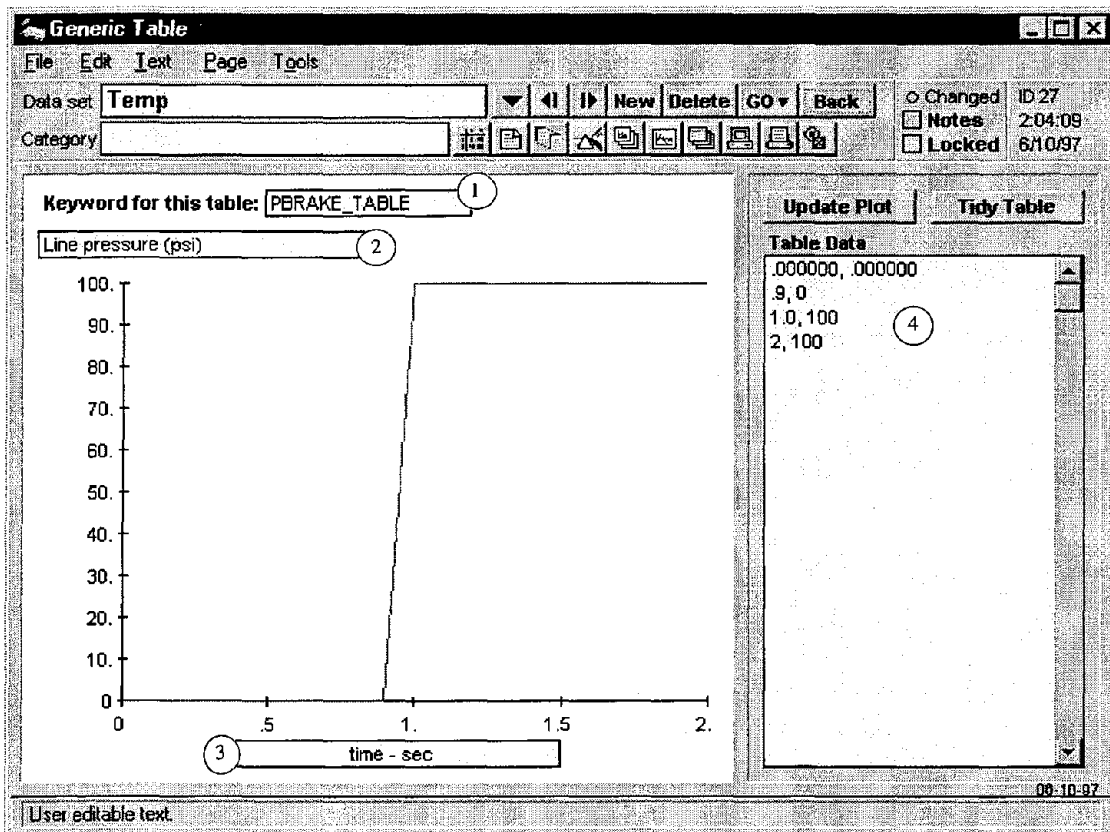
This library does not have a default position in VDM RoAD. You can link to it from any blue field in the VDM RoAD libraries.

File Location

Generic\Gendata\Gendata.tbk

Generic Table

Use this screen to store and display tabular data involving a single independent variable for custom VDM RoAD models. This library is not needed for the standard VDM RoAD models.



Discussion

All tables in the standard VDM RoAD models are associated with libraries in the VDM RoAD data base. However, VDM RoAD can be extended to include new models. If the new models involve tabular data with one independent variable, then this library can be used to store the data.

Unlike most of the libraries in VDM RoAD, each data set in this library can represent a different kind of data. This is possible because the keyword used by the simulation solver programs to identify tabular data is a part of the data set.

User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Keyword. This keyword is required for a simulation program to make use of the data in the table ④. If the keyword in this field is not recognized by the simulation program, then the data will not be used.
- ② Label for Y axis. This label is not used by the simulation programs. It is a user-comment field, provided as a means for documenting the information on the screen.
- ③ Label for X axis. This label is not used by the simulation programs. It is a user-comment field, provided as a means for documenting the information on the screen.
- ④ Tabular data. The first row has values of the first independent variable, the first column has values of the other independent variable, and all other numbers are values of the dependent variable.

Location in VDM RoAD

This library does not have a default position in VDM RoAD. You can link to it from any blue field in the VDM RoAD libraries.

File Location

Generic\Gentable\Gentable.tbk

Hitch

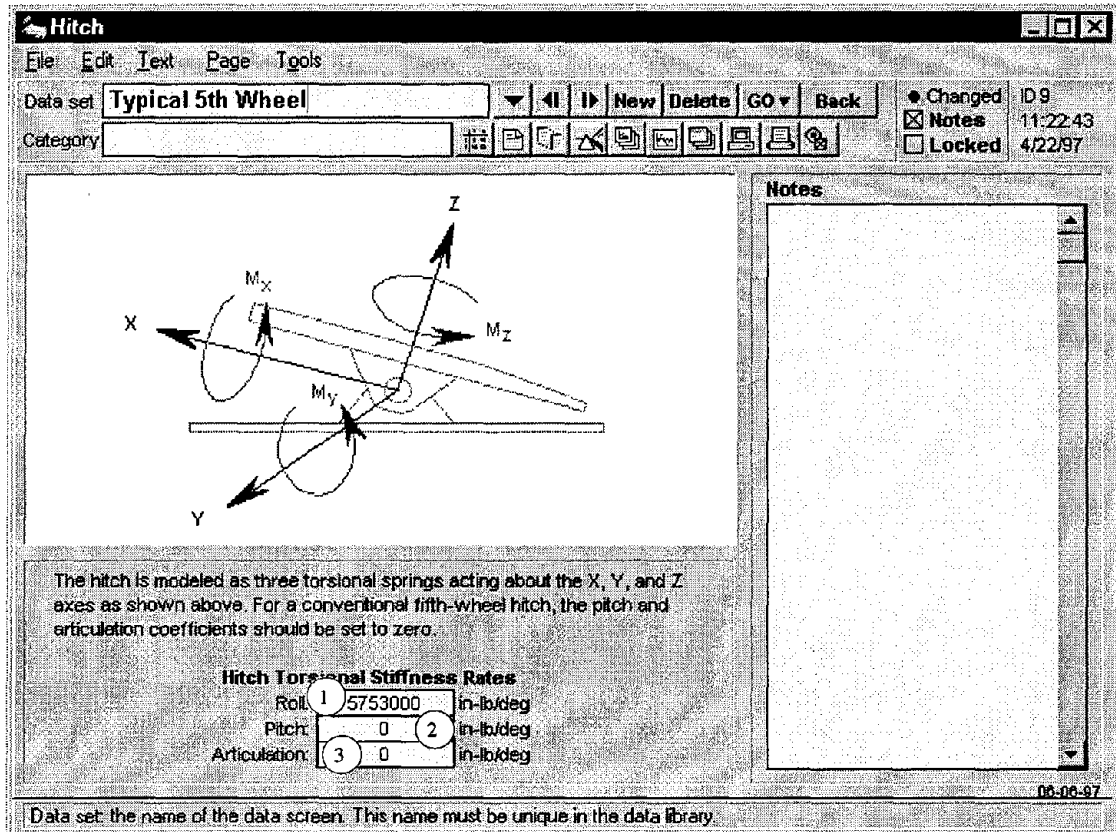
Use this screen to define the torsional stiffness properties of the fifth wheel connection between a tractor and semi-trailer. This library is not used for the car model in VDM RoAD, nor is it used for the single-unit truck models.

Discussion

The torsional stiffness properties of the fifth wheel connecting a tractor and semitrailer are important to cornering performance. The input on this screen allows specification of that stiffness along with those for the orthogonal directions. The hitch vertical location is

defined on the vehicle screen for the tractor. Its longitudinal position is determined (calculated) to produce the axle load distribution specified for the loaded vehicle combination.

To simulate a ball-joint hitch (for example, a sport-utility vehicle towing a boat), set all of the torsional spring rates to zero.



The hitch is modeled as three torsional springs acting about the X, Y, and Z axes as shown above. For a conventional fifth-wheel hitch, the pitch and articulation coefficients should be set to zero.

Hitch Torsional Stiffness Rates		
Roll: 1	5753000	in-lb/deg
Pitch: 2	0	in-lb/deg
Articulation: 3	0	in-lb/deg

Data set: the name of the data screen. This name must be unique in the data library.

User Settings

- ① Roll torsional stiffness (keyword = $KHX(1)$). Torsional stiffness of the fifth wheel measured about the tractor roll axis. For a conventional fifth wheel this value should be at least several hundred thousand N-m/deg. For a simple ball joint hitch this value should be set to zero.
- ② Pitch torsional stiffness (keyword = $KHY(1)$). Torsional stiffness of the fifth wheel measured about the tractor pitch axis. This value should be set to zero for either a conventional fifth wheel or a simple ball joint.
- ③ Articulation torsional stiffness (keyword = $KHZ(1)$). Torsional stiffness of the fifth wheel measured about the articulation axis. This value should be set to zero for either a conventional fifth wheel or a simple ball joint.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: Tractor-Semitrailer Combination

Vehicle Unit: (Tractor)

Hitch

File Location

Vehicles\Parts\Hitches\Hitches.tbk

Input: Braking

Use this screen to specify the braking application input to the brake system in terms of control pressure as a function of time.

Discussion

You can control braking by applying effort at the brake pedal. The brake pedal output is a control, or application, pressure that is applied to the brake system. It is described via a table look-up function of brake input pressure as a function of time. The plot's vertical axis must be the same as the horizontal axis on the **Brakes: Mechanical Properties** screen, and, therefore, the dimensional units must be the same on both screens.

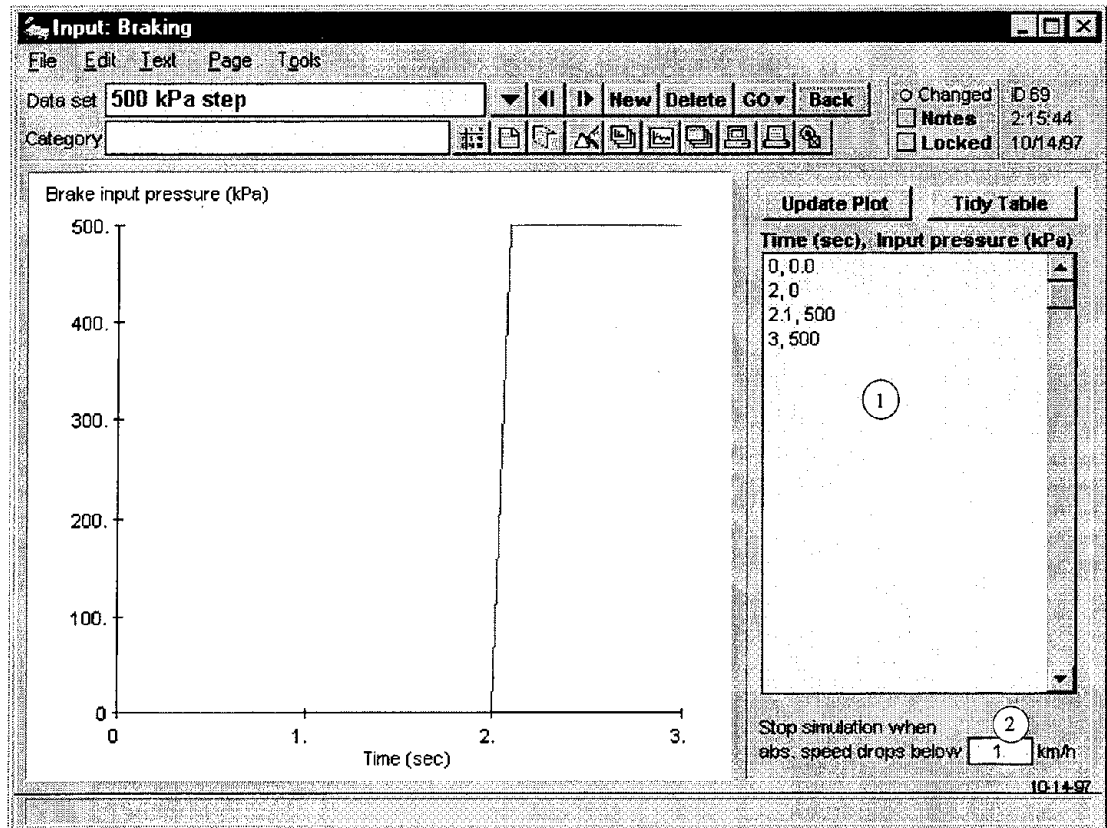
Note: The labels on the screens suggest the input should be pressure, with units of kPa. However, as long as the same units are used for input and mechanical properties, other units can be used if you prefer (e.g., percent, psi, pedal force).

Brake input as a function of station number does not have the same practical use as other variables. Therefore, braking input is specified as a function of time, the same as in the related software packages CarSim and TruckSim.

Note: Brake input is the only time-based control in VDM RoAD.

This screen also has a parameter that controls whether a simulation stops when vehicle comes to rest.

A single library contains braking input data for all VDM RoAD models (cars and trucks).



User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Table field for brake input pressure (keyword = `PBRAKE_TABLE`). Each line should have a value of time (seconds) followed by a corresponding value of braking input. The normal units for brake input are kPa. However, the important thing is that the units on this screen match those used for the brake system mechanical properties.

The simulation programs use flat-line extrapolation with this table. For values of time that are less than the range covered in the table, the first value of pressure is used. For values of time larger than the range covered, the last value of pressure is used. The table needs at least two lines of data or else an error message is generated.

- ② Stop speed (keyword = `V_STOP`). A simulation continues until one of several conditions occurs:
- The simulation time reaches the stop time (keyword = `STOPT`).
 - The vehicle S coordinate (at the front axle) reaches the specified station number (keyword = `STOPS`).

- The vehicle roll angle exceeds a specified limit that implies that a rollover accident was inevitable (keyword = ROLL_STOP).
- The absolute vehicle speed dropped below a specified threshold (keyword = V_STOP).

It is the last case that is associated with braking simulations. If you want to use VDM RoAD to determine stopping distance or other measures of braking performance, it is convenient to have the simulation stop when the absolute speed is close to zero. A typical value for this purpose would be $V_STOP = 1$ km/h. To keep the simulation running, specify a negative number, because the absolute speed cannot go lower than zero.

Location in VDM RoAD

VDM RoAD Startup
Runs
Input: Braking

File Location

Input\Braking\Braking.tbk

Input: Closed-Loop Speed Control

This screen is used to define target speed as a function of station number for closed-loop speed control.

Discussion

Although the VDM RoAD vehicle simulation models do not have details of a driveline, the effect of a driver attempting to maintain a target speed is simulated through the use of a closed-loop speed controller that applies torque to the drive wheels as needed to follow a specified speed. The drive torque causes the wheel spin to increase, resulting in tire tractive force that acts to change the vehicle speed. For modest drive torque, the tire forces should be approximately the same as would be obtained with a full drive line model.

The controller algorithm uses proportional-integral (PI) control. It is characterized by two feedback coefficients: one being proportional to the current speed error, and the other being proportional to the integral of the speed error.

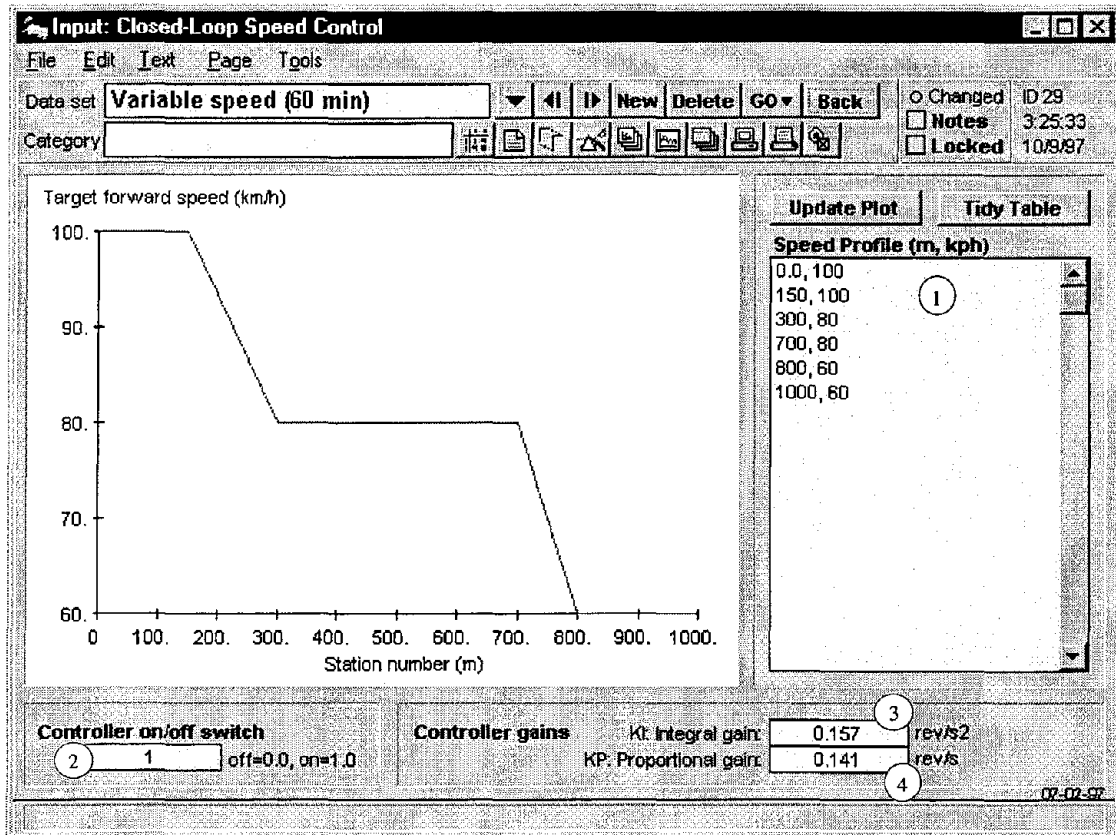
The controller functions only if there is no brake input. As soon as the brake input is non-zero, the speed controller is turned off for the remainder of the simulation run.

The controller does not employ the braking system to slow the vehicle, even if you define a drastic reduction in speed. The retarding effect is through the driveline. For severe deceleration, instabilities can occur in the simulation that are not realistic, because the

distribution (by axle) is not the same as during braking. Wheels could lock up (or even spin backwards) on the affected axle, while wheels on other axles continue free rolling.

To maintain realistic changes in speed, the acceleration and deceleration levels should be limited to several hundredths of a G. It is easy to check this by making a plot of the longitudinal acceleration (A_x) after a simulation has been run.

A single library contains target speed data for all VDM RoAD models (cars and trucks).



User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Two-column table of target speed as a function of station number (keyword = SPEED_TABLE). Station number is in meters, and speed has units of km/h. The simulation programs use flat-line extrapolation with this table. For values of time that are less than the range covered in the table, the first value of speed is used. For values of time larger than the range covered, the last value of speed is used. The table needs at least two lines of data or else an error message is generated.
- ② **Controller on/off switch** (keyword = SPEED_ON_OFF). This value determines whether the controller is on or off. 0=off, 1=on. If the switch is off, the vehicle initial speed is set

to match the target speed, but no drive torque will be applied during the run. The option for turning off the controller is provided for advanced users.

- ③ Speed control integral gain (keyword = `SPEED_KI`). This value is the integral control coefficient. It is scaled to correspond to a response frequency, with units of cycles/sec. Typical values are about 0.1. Setting the value higher causes the drive torque to be modulated more aggressively to maintain a smaller position error. If the number is too high, the changes in drive torque will become unrealistically high. If the number is set low, the position error is larger, and the actual speed may not match the target speed.
- ④ Speed control proportional gain (keyword = `SPEED_KP`). This value is the proportional control coefficient. It is scaled to correspond to a response frequency, with units of cycles/sec. Typical values are about 0.1. Setting the value higher causes the drive torque to be modulated more rapidly. Although this gain controls the speed of the controller response, it does not force the speed error to zero. To make the vehicle speed match the target speed, integral control is needed ③.

Location in VDM RoAD

VDM RoAD Startup

Runs

Input: Closed-Loop Speed Control

File Location

Input\Speed\Speed.tbk

Input: Highway Data (IHSDM)

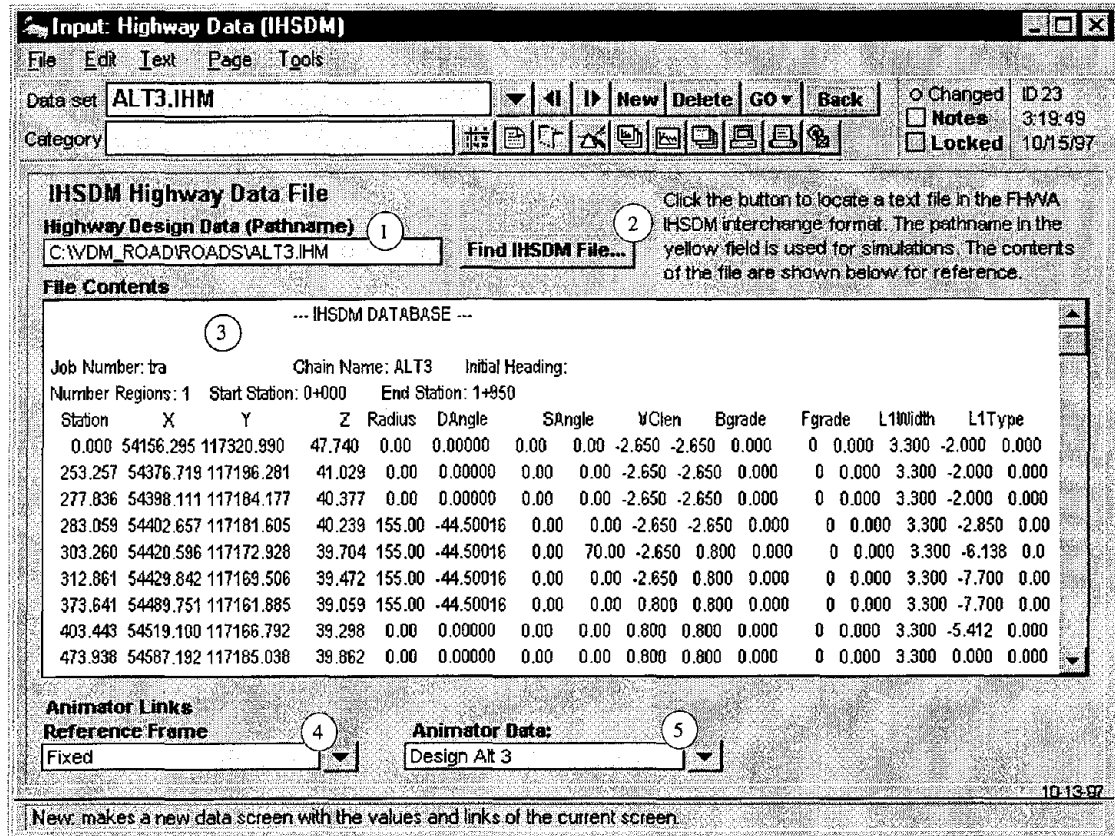
Use this screen to define road geometry, both for the vehicle simulation models and for the wire-frame animation.

Discussion

The driver model in the vehicle follows a road design, as specified in an IHSDM interchange file as defined by the A/E Group in the FHWA Turner-Fairbank Geometrics Group. The IHSDM file contains the data. This screen just has the pathname for the file, in order to communicate the pathname to the vehicle simulation programs.

In order to view the road in animations, the road must also be described with animator shape data. To help keep the animator data associated with the simulation data, this screen has links to the animator data.

A single library contains road data for all VDM RoAD models (cars and trucks).



User Settings

- ① **Highway Design Data (Pathname).** This is a pathname for an IHSDM interchange file (keyword = IHSDM). The pathname is used by the solver program to read the highway design data and generate the ground 3D surface under the vehicle during a simulation run.
- ② **Find IHSDM File** button. Click this button to bring up the Windows file browser, and locate an IHSDM file. When a file is selected, its pathname is copied into the yellow field ①, and the contents of the file are displayed in the white field ③.
- ③ **File Contents** field. This field shows the contents of an IHSDM file, if it is selected using the button ②. The contents of this field are not sent to the simulation program—they are shown to provide a check that the file whose pathname is specified has the correct data.
- ④ **Reference Frame** link. This is normally linked to a reference frame called **Fixed**—a non-moving frame in which the road wire-frame shapes are defined.
- ⑤ **Animator Data** link. Link this to an animator shape data set or to an animator group with several shape data sets. The shape data sets should be set up to show the road geometry as defined in the IHSDM file. See the section **How to Make Wire-Frame Shapes for a Road** in Chapter 3 for instruction in creating a data set for a wire-frame representation of a road.

Location in VDM RoAD

VDM RoAD Startup

Runs

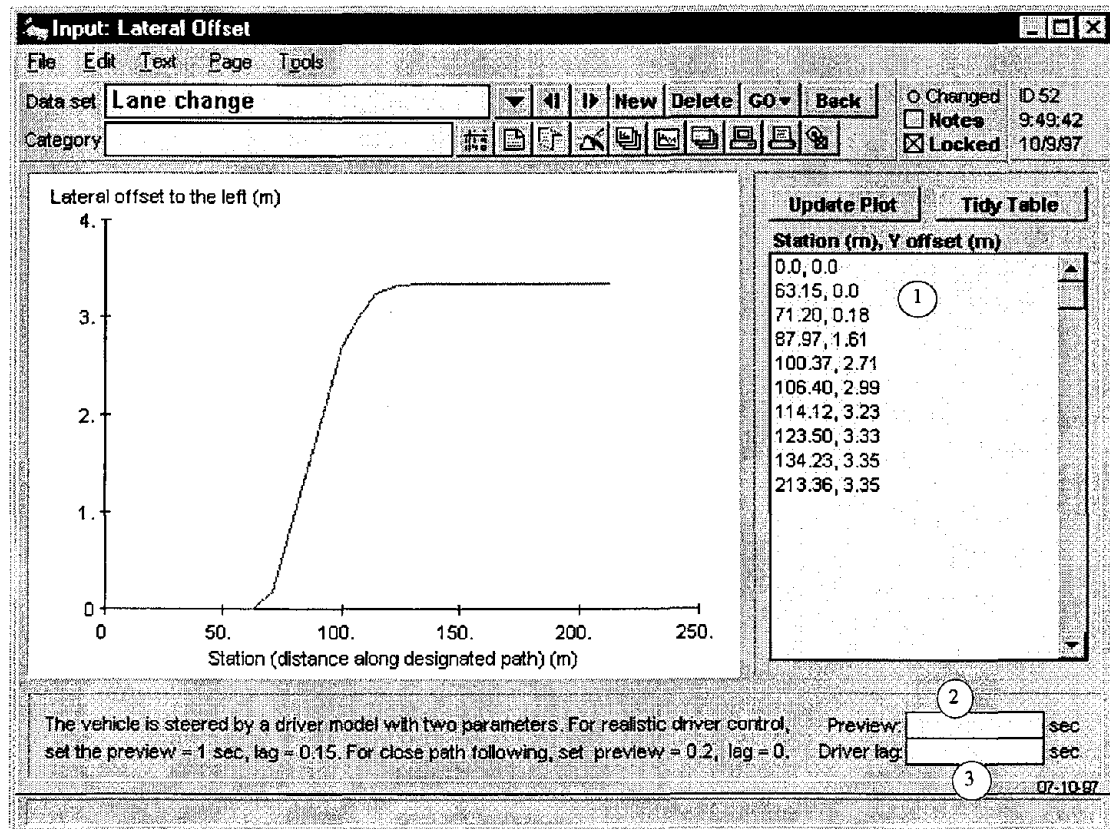
Input: Highway Data (IHSDM)

File Location

Input\Highway\Highway.tbk

Input: Lateral Offset

Use this screen to define a lateral offset from road centerline defined in the IHSDM file specified using an **Input: Highway Data (IHSDM)** data set.



Discussion

The VDM RoAD vehicle models include a closed-loop steering controller that generates a steering wheel angle in an attempt to follow a target path. The target path is defined with a lateral offset relative to the centerline defined in an IHSDM file. The path uses the

S-L-Z coordinate system, described in Appendix H. On this screen, you specify the target path in terms of S and L coordinates.

For example, to run the vehicle in a path 1.65 meters to the left of the design centerline, specify a lateral offset (L) of 1.65 for all station numbers (S). To run 1.65 meters to the right of the centerline, specify an offset (L) of -1.65 meters.

The directions “left” and “right” assume that you are looking down the road in the direction of increasing station number. If the stopping station number specified in the **Runs** screen is higher than the starting station number, then the vehicle would be moving in the direction you are facing. If the stopping station number is less than the starting station number, then the vehicle would be moving in the opposite direction.

Along with the target path, this screen has two parameters that determine the properties of the driver controller. Appendix I describes the algorithm used by the driver controller.

A single library contains lateral offset data for all VDM RoAD models (cars and trucks).

User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Table field of values for S and L values of the target path (keyword = LTARG_TABLE). S is station number, and L is the lateral distance to the left of that path. The simulation programs use flat-line extrapolation with this table. For values of S that are less than the range covered in the table, the first value of L is used. For values of S larger than the range covered, the last value of L is used. The table needs at least two lines of data or else an error message is generated.
- ② Preview time (keyword = TPREV). This is the look-ahead time used by the driver controller algorithm to steer the vehicle. A shorter time causes the controller to steer more rapidly in response to deviations of the vehicle from the target path. A longer time causes the vehicle to look ahead more, steering more slowly in response to changes in the target path.

A realistic value is about 1 second. For constant lateral offsets, a shorter preview time such as 0.2 seconds can be used to generate more aggressive steering to keep the vehicle on the target path. For variable paths (say, generated with a driver model in IHSDM), the short response time might lead to steering that is too aggressive and cause a loss of control.

If the table of S-L values ① has discontinuous changes, a preview time of 1 sec is recommended to avoid over-steering by the controller.

- ③ Driver lag (keyword = TDLAG). Steering wheel angles generated by the driver controller are delayed by this amount of time, to simulate the neuro-muscular delay in humans. A realistic value is about 0.15 sec. Larger values can be used to simulate impaired drivers.

As the lag increases, the driver-vehicle systems tends to over-correct to the point of instability.

If you are attempting to follow a path closely, without trying to simulate human response dynamics, a value of 0.0 is recommended. This results in the best steering behavior.

Location in VDM RoAD

VDM RoAD Startup

Runs

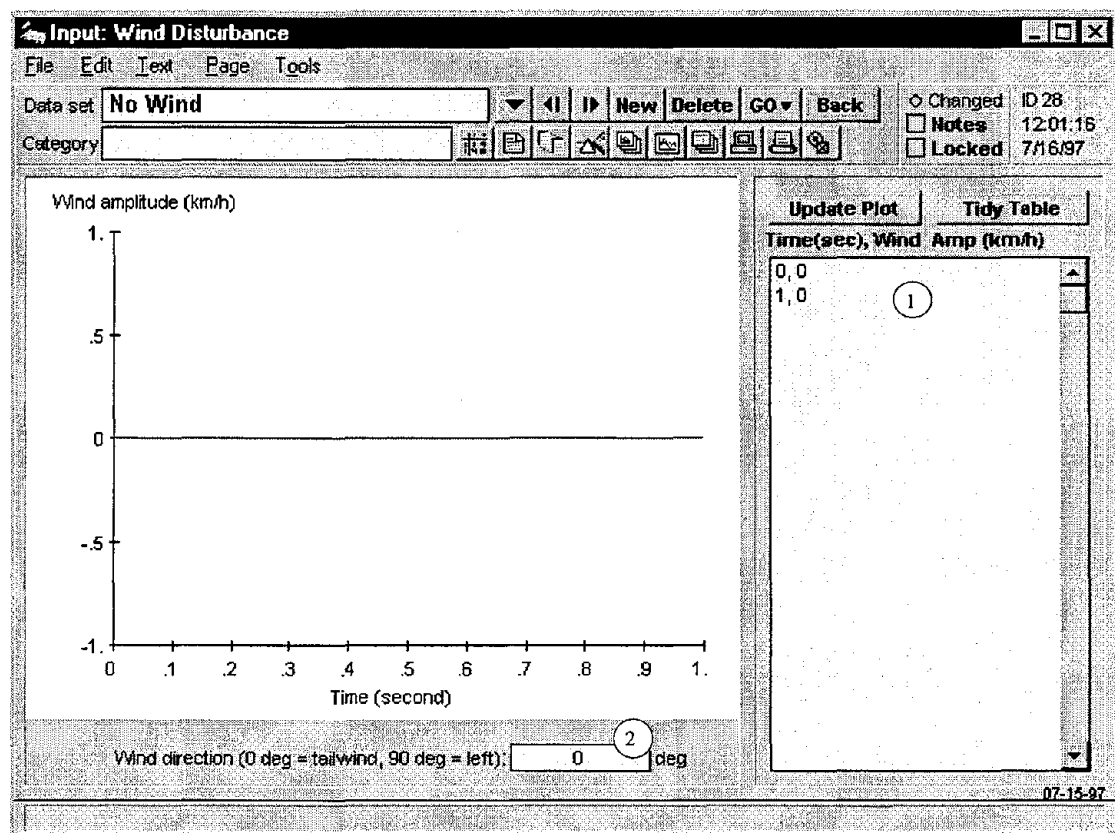
Input: Lateral Offset

File Location

Input\Offset\Offset.tbk

Input: Wind

Use this screen to define the wind amplitude (used to calculate aerodynamics forces) as a function of time. This library is not used for the truck models in VDM RoAD.



Discussion

The car model in VDM RoAD includes the effects of aerodynamic forces and moments. This screen is used to add the effect of a user-defined wind. The wind is characterized by an amplitude, which varies with time, and a direction, which is constant.

User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Two-column table of wind speed as a function of time (keyword = VW_TABLE). Time is in seconds, and wind speed has units of km/h. The simulation programs use flat-line extrapolation with this table. For values of time that are less than the range covered in the table, the first value of speed is used. For values of time larger than the range covered, the last value of speed is used. The table needs at least two lines of data or else an error message is generated.
- ② Heading angle of wind direction (keyword = PSIWND). If the road heading (from the IHSDM file) matches the wind direction heading, the vehicle has a tailwind when going in the direction of increasing station number. It would have a headwind going in the direction of decreasing station number.

Location in VDM RoAD

VDM RoAD Startup

Runs

Input: Wind

File Location

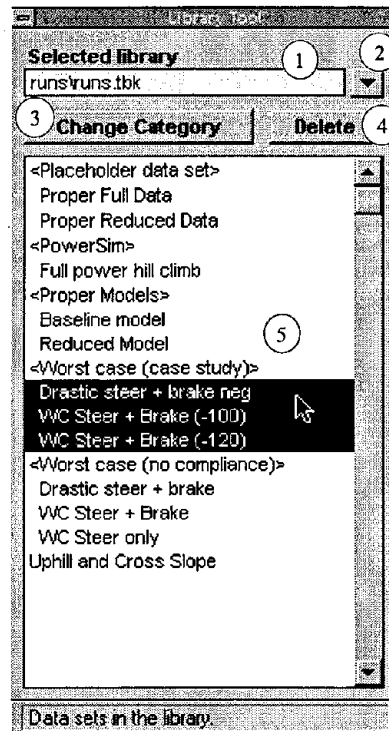
Input\Wind\Wind.tbk

Library Editor

The Library editor is a tool built into VDM RoAD to help you organize libraries by performing batch delete and renaming operations.

Discussion

The Library editor is available from any VDM RoAD library through the ribbon bar and the **Tools** menu. When you bring it up, it automatically links to the current library. Use it to delete a group of data sets from the library, or to organize data sets into categories.



User Settings and Controls

- ① Link to a library. You can edit one library at a time, and this blue field shows the one that is active. When the editor is opened, the current library is loaded into this link.

Note: Items ① and ② are similar to the ubiquitous blue link. However, instead of this being a link to a data set, it is a link to the entire library. Rather than showing the name of a data set, it shows a pathname to a library file. The pathname is relative to the root VDM RoAD folder. For example, if the root VDM RoAD folder is `c:\VDM RoAD`, then the relative pathname `runs\runs.tbk` refers to the file: `c:\VDM RoAD\Runs\Runs.tbk`.

- ② Triangle button. Press this button to bring up a command for changing the library link shown in ①.
- ③ **Change Category** button. Click this to change the category for all data sets currently selected in ⑤. It calls up a dialog box for you to enter a new category name and then modifies the category field for the selected data sets. The result is the same as if you went to each data set and changed the category field.


Note: Categories are solely for grouping the data sets in the library and have no impact on the data set name or parameter values. This change has no effect on parameter values. It only affects how the data sets are grouped in pull-down menus.

- ④ **Delete** button. This button will remove all data sets currently selected in ⑤. This is the same as going to each data set and clicking the **Delete** button for that screen. A warning

message appears, giving you a chance to back out. Be aware that there is no undo option: Once the data sets are deleted, they are gone!

- ⑤ List of data sets from the linked library. The buttons ③ and ④ affect the data sets that are highlighted. Select data sets by clicking on them. Use shift-click to select a continuous range, and use control-click to select lines that are not consecutive.

Location in VDM RoAD

Accessed from the **Tools** menu or the ribbon bar with the button: 

File Location

This editor is not contained in a conventional VDM RoAD library file. It is stored with SGUI code in `Sgui_lib\Bootfile.tbk`.

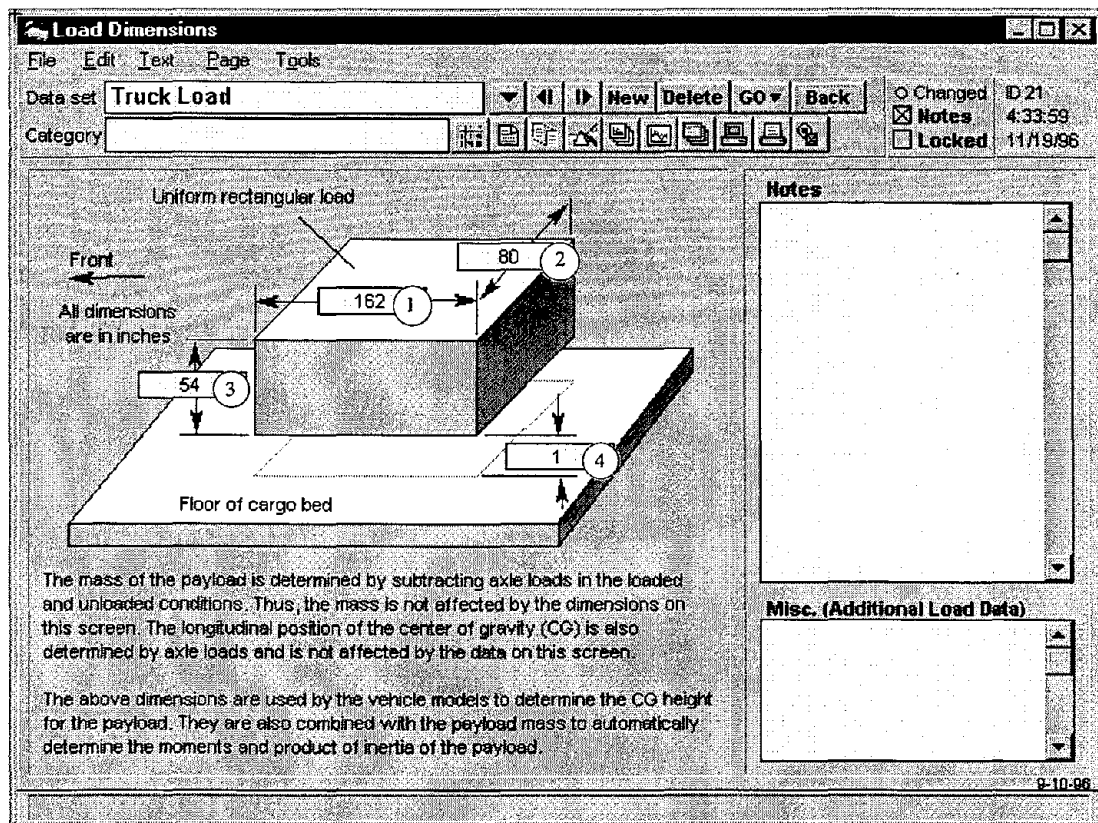
Load Dimensions

Use this screen to define the load (cargo) dimensions on a truck or trailer. This library is not used for the car model in VDM RoAD.

Discussion

Use this screen to define the load (cargo) dimensions on a truck or trailer, so that the base vehicle can be specified and then conveniently loaded to different conditions. The dimensions specified are needed to calculate the height of the center of gravity (CG) for the vehicle and the rotational inertia. The load is assumed to be rectangular in shape and uniform in density.

The mass value for the load is determined elsewhere—based on the difference between weights of the loaded and unloaded vehicle(s). The longitudinal position of the load CG is also determined elsewhere, based on the axle load distribution for the loaded vehicle.



User Settings

- ① Width (lateral dimension) of the load (keyword = LYRL (1)).
- ② Length (longitudinal dimension) of the load (keyword = LXRL (1)).
- ③ Height (vertical dimension) of the load (keyword = LZRL (1)). This value is important because it determines the height of the mass center of the load.
- ④ Elevation of the bottom of the load above the floor of the cargo bed (keyword = HLLB (1)). The floor of the cargo bed is specified in the Empty Truck or Empty Trailer screen. This value is also important because it contributes to the height of the mass center of the load.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Load Dimensions

File Location

Vehicles\Loads\Loads.tbk

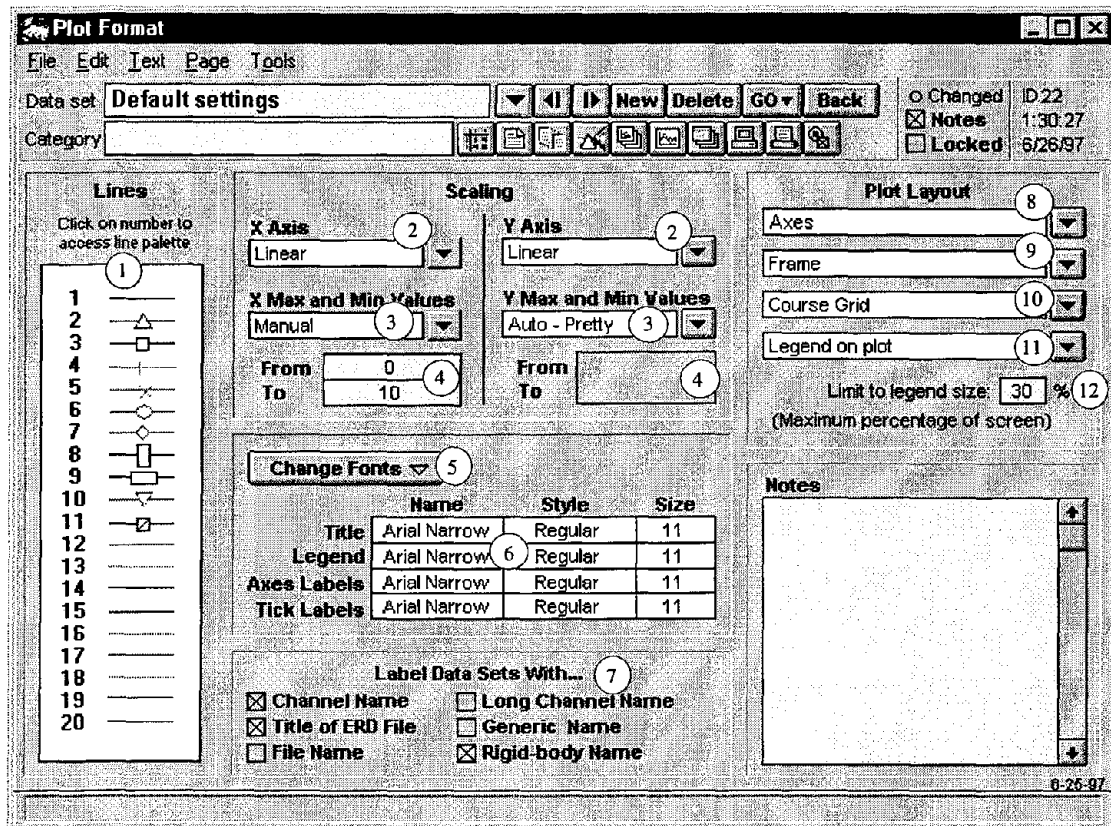
Plot Format

The format screen determines the style and layout properties of a plot. The formats cover line style, scaling, front properties, size, and axis type.

Discussion

The WinEP program supports a number of options for controlling the format of the generated plots. As was described in Chapter 6, dialog boxes are used for setting all options interactively, and those options can be stored in text files for future use.

This data screen is also used to specify plotting formats. Every user setting on this screen can also be set interactively from within WinEP. However, when set here, the formatting information is stored in the VDM RoAD data base and can be applied easily to any plots initiated from within VDM RoAD (clicking the **Plot** button from the **Runs** screen or the **Plot: Batch Setup** screen).



User Settings

- ① Lines and symbol specification (keywords = `symbols`, `linestyl`, `colors`). Clicking on a number or line will bring up the **Lines** palette, shown in Figure 8.2. The palette

applies to one line number, which will be highlighted. For example, line 2 is selected in Figure 8.2.

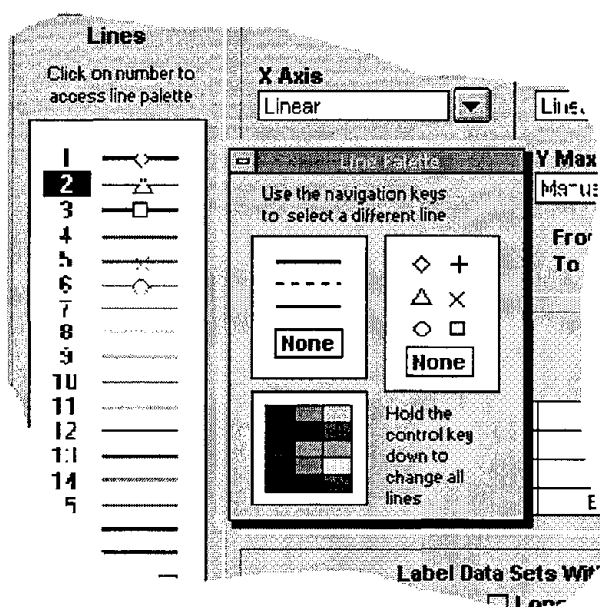


Figure 8.2. The Lines palette.

- Choose the color, line style (dotted, thin, thick), and the symbol for the selected line. To change a different line, click on a different line number, or use the up and down arrow keys. To change all of the lines at once, hold down the control key when selecting the style, color, or symbol.
- ② Axis type (keywords = `xlinear`, `ylinear`). Each white field has an adjacent button with a pull-down menu with choices for **Linear** and **Log**.
 - ③ Mode of scaling (keywords = `xmaxmin`, `ymaxmin`). Each white field has an adjacent button with a pull-down menu with choices for **Auto-pretty** and **Manual**. In the auto-scaling mode, the data are scanned to determine the maximum and minimum values for the axis. **Manual** uses the values in the **From** and **To** fields ④.
 - ④ Axis limits (only used in **Manual** scaling mode).
 - ⑤ **Change Fonts** button. Press the button to display a pull-down menu for selecting font properties for the title, legend, axes labels, and axis numbers. These parts of the plot are identified in Figure 8.3. A standard Windows dialog box for specifying font properties then appears and is used to specify the font properties.

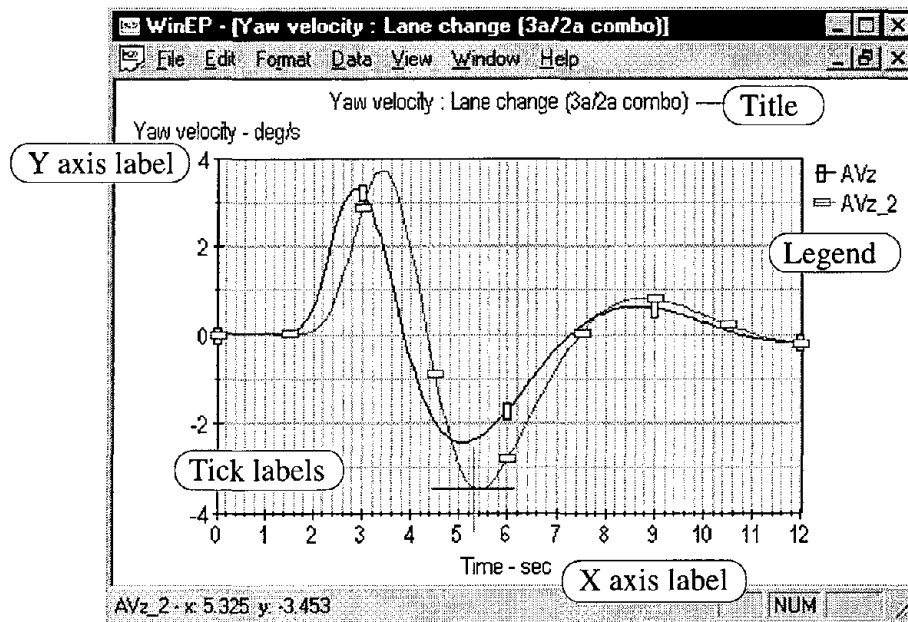


Figure 8.3. Labels in a plot.

- ⑥ Current font settings. The table shows the current font settings for the four types of labels appearing in the plot. You cannot change the values shown directly—you must go through the **Change Fonts** button ⑤.
- ⑦ Check boxes for labeling options. When more than one data set is plotted, the individual data sets are identified with a legend. These check boxes control which labels are used to identify the plots. Along with title and file name are variable or channel specific information including name, long name, generic name, and rigid-body name.
- ⑧ Axis selection. The button has a pull-down menu with two options: no axes or axes. The current selection is shown in the white field.
- ⑨ Frame selection. The button has a pull-down menu with three options: no surrounding frame, a rectangular frame, or a frame with tick marks. The current selection is shown in the white field.
- ⑩ Grid selection. The button has a pull-down menu with three options: no grid, fine grid, or course grid. The current selection is shown in the white field.
- ⑪ Legend location. The button has a pull-down menu with two possible locations for the legend that identifies data sets in overlay plots. The choices are to the right of the plot, or four locations within the plot area. The current selection is shown in the white field.
- ⑫ Legend size limit. When two or more data sets are overlaid, and the legend is located to the right of the plot area, WinEP sizes the plot area to leave just enough room to print the legend. In the event that at least one of the labels in the legend is long, or that the window is not very wide, then little space is available for the plot. This field is used to set a maximum amount of the total window width that will be used for the legend.

This setting has no effect unless three conditions are met:

1. At least two data sets are overlaid in the plot.
2. The legend is placed to the right of the plot (as specified in (11)).
3. The length of the longest label in the legend exceeds the specified percentage of the window width.

If the above three conditions are met, then the plot is sized giving the legend the specified amount. For example, if the limit is set to 30%, then the plot area is sized to leave exactly 30% of the width for the legend. The labels that are too long are truncated to fit.

Location in VDM RoAD

VDM RoAD Startup
Runs
Plot Setup: Single
Plot Format

File Location

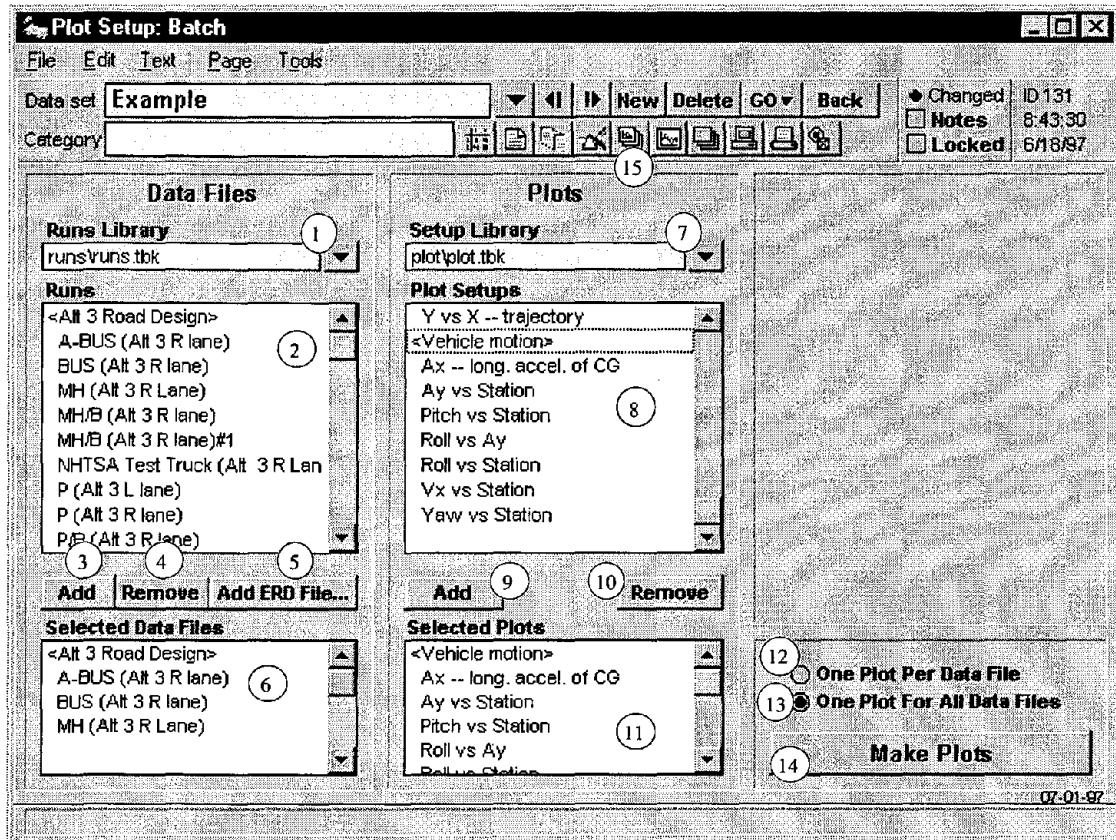
Plot\Format\Format.tbk

Plot Setup: Batch

The batch plot screen is used to combine a list of runs with a list of plot conditions to automatically generate a set of plots.

Description

The standard Runs screen allows multiple plots to be specified, and multiple runs to be overlaid. However, it is limited to four plot setups and three runs. If you want to trigger more plots with one button click, or overlay data from more than three runs, you can do it from this library.




User Settings and Controls

- ① **Runs Library** link. The pull-down menu has two options: (1) to go to the currently selected library, or (2) to pick a new library. The linked library must be a runs or simulation library that has corresponding ERD files. When you pick a new library, the field below is updated to show the names of all of its data sets.
- ② The **Runs** list. List of all of the data sets in the **Runs** library. Any line in brackets (<>) indicates a category heading. Double-click on a line to add to the **Selected Data Files** list ⑥.
- ③ The **Add** button. Click to add highlighted lines from the **Runs** list ② to the **Selected Data Files** list ⑥. Shift-click to add all.
- ④ The **Remove** button. Click to remove highlighted data sets from the **Selected Data Files** list ⑥. Shift-click to remove all.
- ⑤ The **Add ERD File** button. Calls up a dialog box for you to select ERD files that are added to the **Selected Data Files** list ⑥. The dialog box will continue to request selections until you click the **Cancel** button.
- ⑥ The **Selected Data Files** list. List of the selected data sets from the **Runs** list. These are the data sets that will be plotted. Any line in brackets (<>) indicates a category heading. Double-click on a line to remove from the list.

- ⑦ **Plot Setup** link. The pull-down menu has two options: (1) to go to the currently selected library, or (2) to pick a new library. The linked library must contain plot setup information. When you pick a new library, the field below is updated to show the names of all of its data sets.
- ⑧ The **Plot Setups** list. List of all of the plot setups in the EP setup library. Any line in brackets (<>) indicates a category heading. Double-click on a line to add to the **Selected Plots** list ⑪.
- ⑨ The **Add** button. Adds highlighted lines from the **Plot Setups** list ⑧ to the **Selected Plots** list ⑪. Shift-click to add all.
- ⑩ The **Remove** button. Removes highlighted data sets from the **Selected Plots** list ⑪. Shift-click to remove all.
- ⑪ The **Selected Plots** list. List of the selected plot setups in the EP setup library. These are the plot setups that will be used for the plots. Any line in brackets (<>) indicates a category heading. Double-click on a line to remove from the list.
- ⑫ The **One Plot Per Data File** button. Click to cause the next set of plots to be made with a separate plot for each combination of a run and plot setup. The total number of plots will be the number of data sets multiplied by the number of selected plots.
- ⑬ The **One Plot For All Data Files** button. Click to cause the next set of plots to be made by overlaying all runs for each plot setup. The total number of plots will be the number of selected plots.
- ⑭ The **Make Plots** button. Click to start the program WinEP and instruct it to make a series of plots based upon the selected runs, plot setups, printing, and overlay options selected.

Location in VDM RoAD

Accessed with the **Tools** menu or the ribbon bar button:  ⑮

File Location

Batch\Plot_bat\Plot_bat.tbk

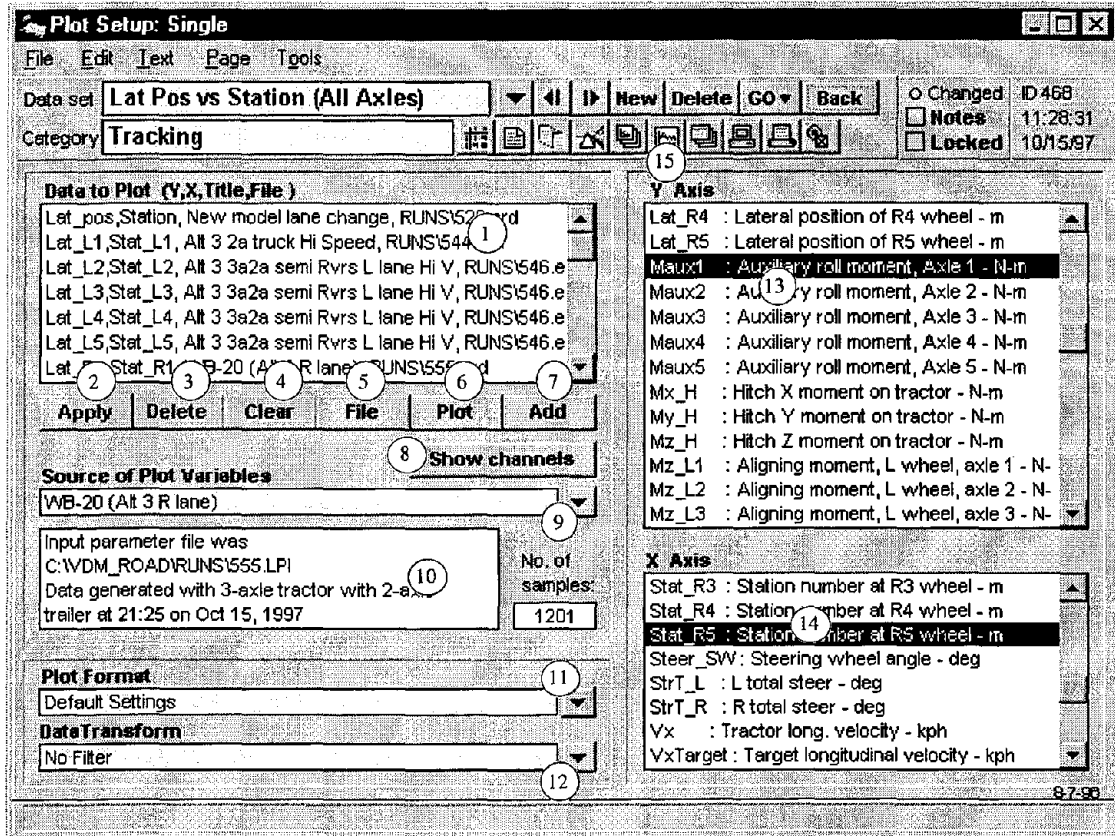
Plot Setup: Single

The data sets in this library each define a graphical plot with specifications of the data (channels and files) to plot, formatting preferences, and filtering (transform) options.

Discussion

This data set is a template, used to specify what a plot should contain and how it should look. The template is applied from the **Runs** screen and the **Plot Setup: Batch** screens. Plots can also be made directly from this library, but this is mainly for quick testing of new data sets.

VDM RoAD comes with more than 20 installed plot templates. You can modify any of these, plus you can define new ones. Once a template is defined, it can be selected from the **Runs** screen and applied routinely to all runs to generate plots.



User Settings and Controls

- ① The **Data to Plot** list. This is the list of variables and file names that will be plotted when the Plot button is clicked. Each line contains the Y axis variable, the X axis variable, the title of the data file, and the file name. The lines in this field can be selected, but not edited directly. They are added and deleted using the buttons underneath. All lines in this field are used in the plot whether they are selected or not.

When the setup is applied from another library such as the batch plotter, the runs screen, etc., the file information is ignored and only the X-Y channel pairs are used.


- ② **Apply** button. If you have already selected several sets of X-Y channels to plot from one run (file) you can click this button to put exactly the same X-Y channels into the **Data to Plot** list ① for the current run (file).
- ③ **Delete** button. Click to remove highlighted lines in the **Data to Plot** list ①.
- ④ **Clear** button. Click to remove all lines in the **Data to Plot** list ①.

- ⑤ **File** button. Click to bring up a dialog box to locate an ERD file. The names of all variables are then loaded and displayed in the **X Axis** and **Y Axis** fields (⑬ and ⑭).
- ⑥ **Plot** button. Starts the program WinEP and generates a plot defined by the channels and files specified in the **Data to Plot** list ①, the format ⑪, and the data transform ⑫.
- ⑦ **Add** button. Click to add the selected channels in the X and Y axis lists (⑬ and ⑭) from the ERD file ⑨ to the **Data to Plot** list ①.
- ⑧ **Show Channels** button. Puts the channel names from the current data file ⑨ into the **X Axis** and **Y Axis** fields (⑬ and ⑭).
- ⑨ Link to **Runs** library or ERD file. Shows the title of the currently selected run or ERD file. If the data set was selected from a data library, only the title is shown. If an ERD file was selected directly with the **File** button ⑤, both the title and file name are shown. You can also select a new ERD file with the adjacent pop-up menu.
- ⑩ Data file information. The field on the left shows the text following the keyword HISTORY in the header of the data file. The field on the right gives the number of samples in the data file.
- ⑪ **Format** link. The linked data set determines the look and scale of a plot. For more information about the Format library, see the section **Plot Format Screen** in this chapter.
- ⑫ **Transform** link. This is for applying a filter to the data for plotting. The available filters are moving average high, low, and band pass.
- ⑬ **Y Axis** list of variables. This shows all the variables in the current data file. Double click on a line to add it and the line highlighted in the **X Axis** list to the **Data to Plot** list ①.
- ⑭ **X Axis** list of variables. This shows all the variables in the current data file. Double click on a line to add it and the line highlighted in the **Y axis** list to the **Data to Plot** list ①.

Making a Plot


Most plots are made from other screens, using the **Plot Setup** data as templates. However, it is also possible to generate plots directly from this screen.

1. Click the **Clear** button ④ to clear the **Data to Plot** list ①.
2. Select a file with data to plot.
 - a. Use the pop-up menu button ⑨ to select a data file from a library by its title, or
 - b. Use the **File** button ⑤ to select a file by its name.
3. Select the variable to plot on the Y axis and the variable to plot on the X axis from the two scrollable lists (**Y axis** ⑬ and **X axis** ⑭) on the right side of the screen
4. Click the **Add** button ⑦ to add the selected X-Y variables from the current file to the **Data to Plot** list ①.
5. Repeat steps 3 and 4 as needed to select more variables. Repeat step 2 to overlay data from another file.

6. Click the **Plot** button  to generate the plot with WinEP.

Location in VDM RoAD

This library is accessible from several places in VDM RoAD.

1. Accessed from the ribbon bar with the button:  (15)
2. Accessed with the **Tools** menu.
3. Accessed from the Plot Setup: Batch screen.
4. VDM RoAD Startup
Runs
Plot Setup: Single

File Location

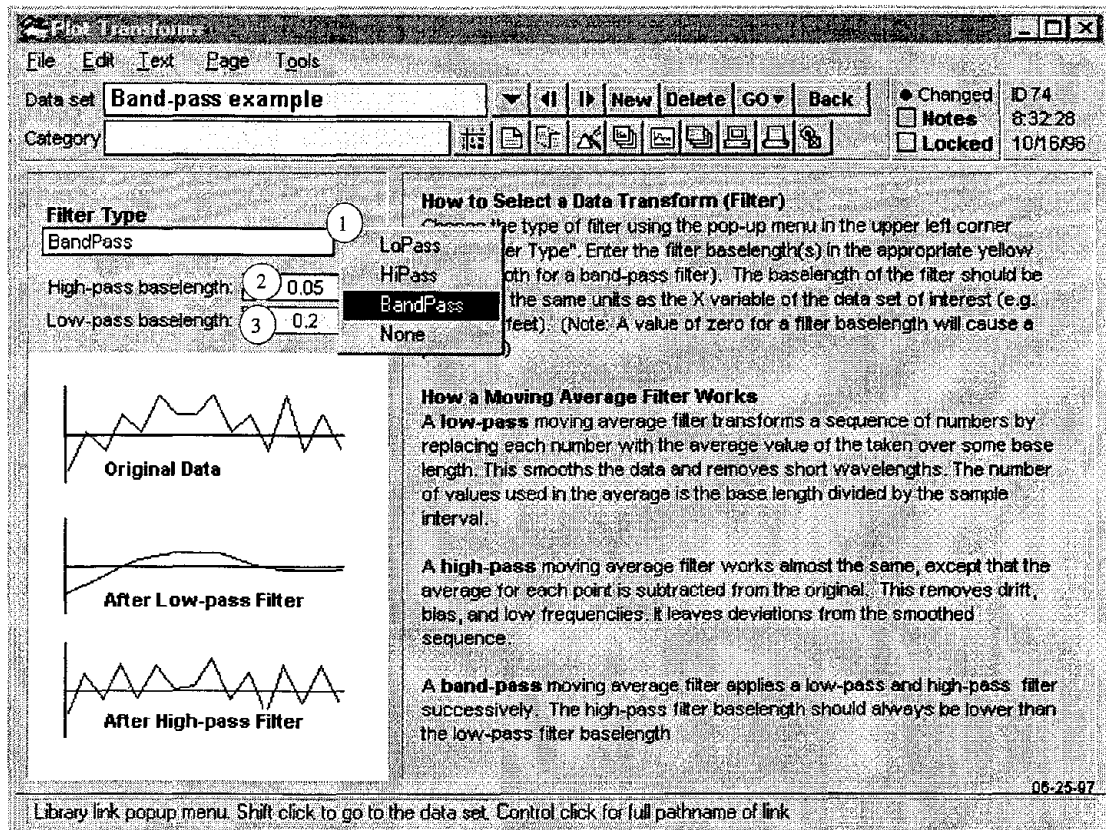
Plot\Plot.tbk

Plot Transforms

This screen is used for setting up plots with smoothing (low pass filter) and unsmoothing (high pass filter).

Discussion

Filtering is normally used to view measured data. It is not routinely applied to simulation results generated by VDM RoAD models. However, it is useful if you have occasion to view experimental data with WinEP.



User Settings

- ① Filter type. Press adjacent button to display pull-down menu with choice of LoPass, HiPass, BandPass, or no filter. See the above screen for more details.
- ② High-pass baselength. This is a baselength for a moving average when the filter type is either high-pass or band-pass. For no filter or a low-pass, this field is hidden.
- ③ Low-pass baselength. This is a baselength for a moving average when the filter type is either low-pass or band-pass. For no filter or a high-pass, this field is hidden.

Location in VDM RoAD

VDM RoAD Startup
 Runs
 Plot Setup: Single
 Plot Transform

File Location

Plot\Transfrm\Transfrm.tbk

Runs

This is the main screen in VDM RoAD. It has two functions:

1. to set up simulation runs and make them, and
2. to run post-processing programs to view results

Note: The **Runs** Screen has several possible appearances depending on whether various check boxes are checked. Figure 8.4 and Figure 8.5 show two possible appearances.

Discussion

The **Runs** screen controls all aspects of a VDM RoAD simulation. Notice that the screen image is divided into three regions

- **Model Parameters & Inputs** — this has links to inputs to the computer model. They include the vehicle and control inputs.
- **Run Control** — here are parameters that control the extent of the run, and a button to make the run.
- **Output & Post Processing** — this has controls for viewing outputs generated by the run.

The **Runs** screen is special within VDM RoAD because it has several buttons that cause other programs to run and do things (simulate a vehicle, animate results, show graphs of output variables). However, like all the other screens covered in this chapter, it also defines a data set within a library. In this case, the data set defines the conditions covered by a run.

Data in the left-most two regions have little to do with outputs of a run. They define the inputs and parameters that will be used in the computer model if and when a run is made. In order to have an effect, changes must be made **before** a run is made. At any time: (1) any of these inputs can be changed, and (2) a new run can be made. The simulation program will always get its inputs using the current data from this screen.

On the other hand, settings in the right-most region have no use until **after** a run is made. Linked data sets control the appearances of plots and animations. Changes in data in this region do not affect a simulation. They only affect how the simulation results are viewed.

Note: The vehicle and highway design data sets do affect the use of the animator. They have links to animation data sets that are used by the animator program to create the wire-frame shapes for the vehicle and road, respectively.

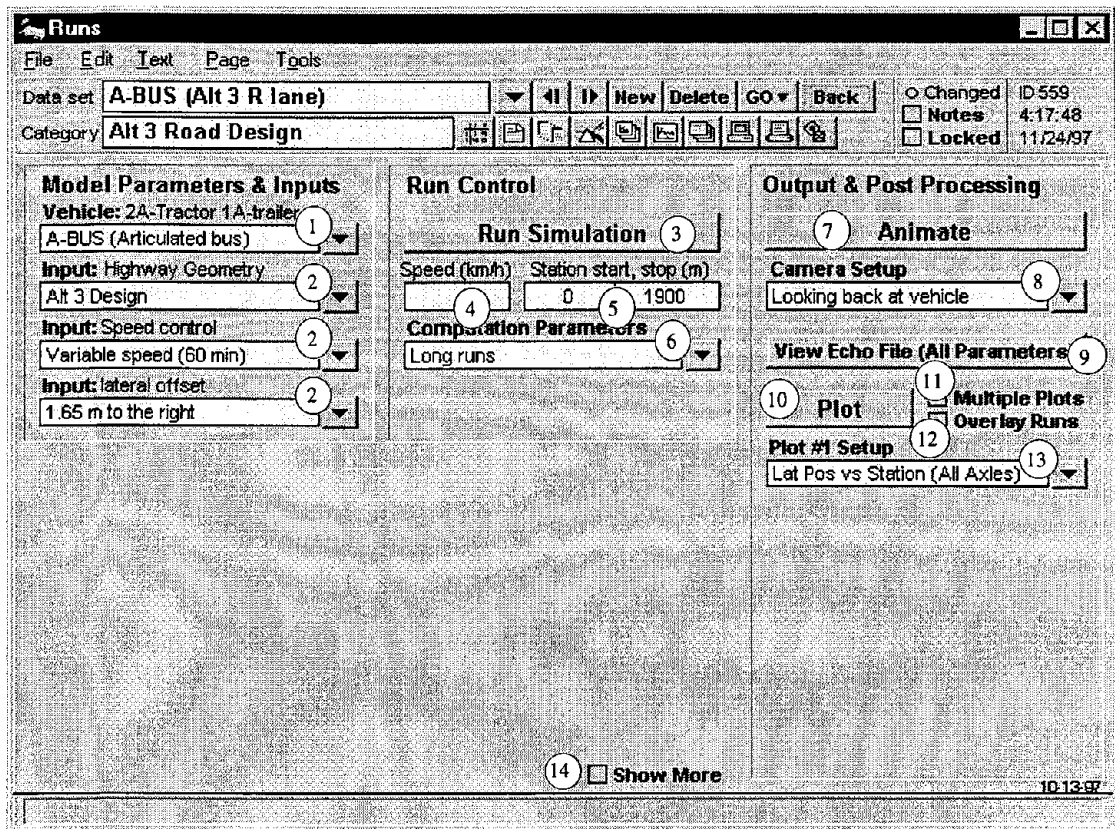


Figure 8.4. The Runs screen (minimal controls).

User Settings and Controls (Simple Display)

- ① Link to vehicle data set. As shipped, various data sets in the **Runs** library have links to all kinds of vehicles that are supported by VDM RoAD.

The type of vehicle is shown as a subtitle above the blue field. In Figure 8.4, the type is 2A-tractor 1A-trailer.

If you want to simulate a different type of vehicle, the recommended method is to:

1. find an existing **Runs** data set that is already set up to simulate the desired type of vehicle, then
 2. copy the data set by clicking the **New** button (to avoid overwriting previous simulation results), and then
 3. modify the copied data set as desired for the new run.
- ② Links to screens for inputs and disturbances. The main inputs used in VDM RoAD are the road design, speed control, and lateral path input.

For braking simulations, one of the input links is set to specify a braking input.

The speed control link is recommended only when you want to simulate the vehicle changing speed according to a prescribed table of speed vs. station number. For constant speed runs, it is simpler to enter the target speed in the yellow field ④.

- ③ **Run Simulation** button. Click to run the appropriate vehicle simulation program using the current model parameters and inputs.
- ④ Initial speed (keyword = `SPEED`). If a speed is entered into this field, the simulation models normally apply a closed-loop speed controller to maintain this speed. The controller is turned off if the brakes are applied. Speed has units of km/h.

Whenever a speed is entered into this field, any speed tables specified in the input links ② are not used. In order to specify a variable speed input, you must both

1. make sure that the field ④ is blank, and
 2. link to an input speed table with one of the input links ②.
- ⑤ Starting and stopping station numbers (keywords = `STARTS` and `STOPS`). These values have several purposes.
 - The starting station number `STARTS` determines where the vehicle is located on the road when the simulation starts. The vehicle is aligned such that its longitudinal axis is parallel with the road design centerline at its front axle. It is recommended that simulations be started on tangent sections, to reduce transients of the vehicle “settling” onto the 3D road surface in a banked turn.
 - The direction of travel is determined by the comparison of `STARTS` and `STOPS`. If `STARTS` is less than `STOPS`, the vehicle is simulated moving in the direction of increasing station. If `STARTS` is greater than `STOPS`, the vehicle goes in the other direction.

Note: The lane used by the vehicle depends on the lateral path input. Regardless of the direction of travel, the S-L coordinates (see Appendix H) define positive lateral distance as being to the left of the centerline, as viewed by an observer looking down in the road in the direction of increasing station number.

- The simulation is stopped when the stopping station number is reached.

The simulation normally runs until the final station number is reached. However, it will stop earlier under some conditions:

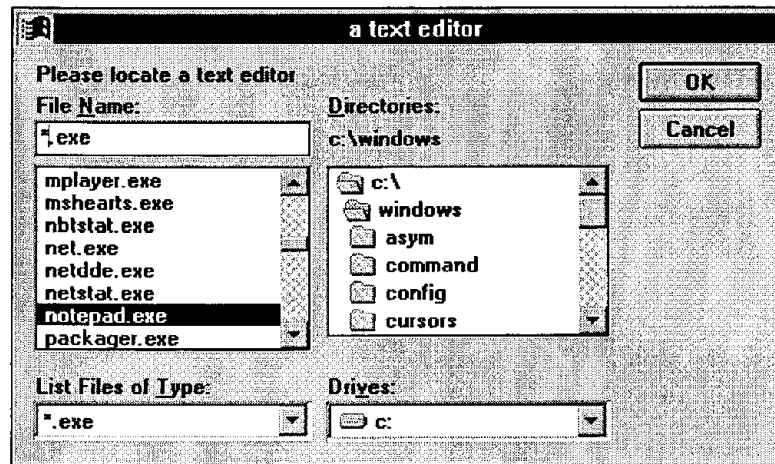
- The simulation time reaches the stop time (keyword = `STOPT`).
- The vehicle roll angle exceeds a specified limit that implies that a rollover accident was inevitable (keyword = `ROLL_STOP`).
- The absolute vehicle speed dropped below a specified threshold (keyword = `V_STOP`).

The low-speed threshold (V_STOP) is set on the **Input: Braking** screen. The other parameters can be optionally set using the **Misc. Data** field (6), described below and shown in Figure 8.5.

- ⑥ **Computation Parameters** link. The linked data set specifies the integration time step, the print interval, the type of integration, and related parameters.
- ⑦ **Animate** button. Click to run the animator program and view motions of the vehicle as predicted by the simulation. An error message will be printed if the output file does not exist, which usually indicates that the run has not yet been made.
- ⑧ **Camera Setup** link. The linked data controls the camera viewpoint when the animator displays outputs produced by the simulation program.
- ⑨ **View Echo File (All Parameters)** button. This button causes a text editor to run and display an echo file produced by the simulation program.

The echo file has the extension LPF, and is similar to the file listed in Appendix D. If the run has not yet been made, or if it aborted without writing the LPF file, then the text editor will show a blank window.

If you click this button and get an error message or a file browser dialog, then VDM RoAD could not find the text editor.



The default text editor is a program called QLPad.exe, located in the Programs directory. If you want to change the default text editor, you can Control-click the button to bring up the file dialog and choose a different text editor.

- ⑩ **Plot** button. Click to view a plot of output variables calculated during the simulation run. The plots are drawn by the WinEP program. An error message will be printed if the output file does not exist (for example, if the run has not yet been made).

Up to four separate plots can be generated (each with many variables, taken from up to three runs). In order to generate more than one plot, the **Multiple Plots** box (11) must be checked.

- ⑪ **Multiple Plots** check box. This check box has two effects:
1. When checked it displays three plot setup links in addition to the first one ⑬.
 2. When checked, it allows up to four plots to be created with a single click of the **Plot** button ⑩.

If the box is not checked, the additional plot links are not cleared, they are just hidden.

- ⑫ **Overlay Runs** check box. This check box has two effects:
1. When checked it displays two links for other runs or ERD files.
 2. When checked, it allows data from up to three files to be overlaid. (One file is the output associated with the current **Runs** data set. The other two are specified with the additional two links.)

If the box is not checked, the additional links are not cleared, they are just hidden.

- ⑬ **Plot Setup** link. The linked data controls what information will be extracted from the output file and how that information will be displayed.
- ⑭ **Show More** check box. This check box has a single effect: it shows more control objects, as shown in Figure 8.5.

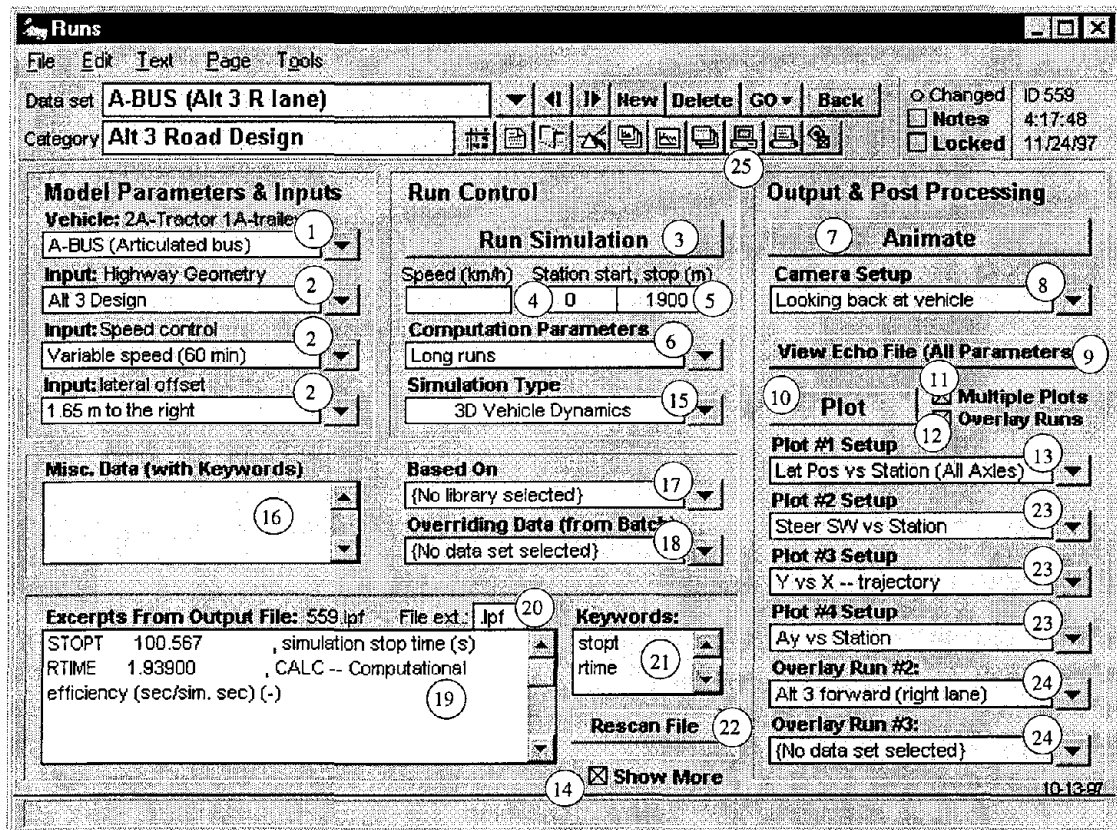


Figure 8.5. The Runs screen (all controls).

Note: Figure 8.5 shows the **Runs** screen with all three boxes checked. All possible data fields and links are displayed.

Additional User Settings and Controls

- ⑮ **Simulation Type** link. This shows the current type of simulation model. As shipped, VDM RoAD has a single type: 3D Vehicle Dynamics. However, custom versions may have modified or extended models.

The triangle button displays a pull-down menu with options for adding and deleting simulation types, and for locating the simulation solver program (extension EXE). Unless you add new solver programs, you should never use this menu. In fact, the menu is disabled unless the check box for **Advanced Users** is checked in the **Preferences** screen.

- ⑯ **Misc. Data** field. This is a field where any parameters can be set. The format is a parameter name and then at least a single blank space and then the parameter value. This field can contain any text that would be recognized by the simulation program. See Appendix D for a list of all the keywords and parameters that can be specified.

One parameter that is commonly set here is the tire/ground friction coefficient, MU. Another is the stopping time, STOPT.

- ⑰ **Based On** link. This link is sometimes used to base a new run on the data from an old run. If a new run is made, all model parameters and inputs from the old run will be read by the solver program before any of the inputs from this data screen are read. The final conditions from the other run are also read, which means that the new run will continue where the old one left off. Any inputs specified on this page will overwrite the data from the previous run.


This link is provided for special applications. (Most users never need it.) A few cases where you might consider using it are:

- to continue a run, perhaps after changing one or more model parameters;
- to make a run using a vehicle description whose parameter values have been lost (the **based on** method reads from an LPF echo file, rather than the normal PAR files associated with the SGUI screens); or
- for certain debugging operations, such as confirming that the parameters specified in the screen are being processed by the solver program.

- ⑱ **Overriding Data** link. This link is used by the system when batch runs are made using the **Runs: Batch** library. If a run is made under batch control, this link can be used to go to the batch data set.
- ⑲ **Excerpts from Output File**. This field shows selected lines from one of the output files produced by the solver programs. It is updated whenever a run is made or when the **Rescan** button ⑳ is clicked.

- ⑳ File extension. This determines which echo file is scanned when a run is made or the **Rescan File** button is clicked. The normal values are LPF (scan the file made with final conditions), or LPO (scan the file made with initial conditions).
- ㉑ **Keywords** field. Each line in this field is a keyword that will be used to scan an output echo file whenever a run is made or when the **Rescan** button is clicked.
- ㉒ **Rescan** button. Click this button to cause VDM RoAD to scan a file for lines beginning with the keywords listed in the **Keywords** field ㉑. Those lines are copied into the adjacent white field ㉓. The file that is scanned has a root name matching the ID of the current data set and the specified extension ㉔.
- ㉓ Additional **Plot Setup** links. These links are to plot data, similar to that in ㉒. A separate plot will be generated for each active link.
- ㉔ **Overlay Run** links. These links are used to generate overlay plots, in which variables from the current simulation run are compared with the same variables from other files. To overlay variables from two or three simulation runs, link to this library (**Runs**) and then pick another run to compare with. To overlay variables with ERD files not from this library (for example, measured test data), use the menu command to open an ERD file.

Location in VDM RoAD

Accessed from the **Tools** menu and the ribbon bar button:  ㉕

Also accessed from the VDM RoAD Startup screen:

VDM RoAD Startup
Runs

File Location

Runs\Runs . tdk

Runs: Batch

The batch runs screen is used to set up several runs ahead of time and make them all at once. It also can be used to override parameters.

Discussion

This screen is used to set up a batch of runs. It is handy for redoing a group of runs, say, because an error was made in a vehicle parameter value. It can also be used to change a parameter such as test speed, and make a set of runs again using the new parameter.

The basic method is:

1. Create a list of runs to be made using existing data sets from the **Runs** library.
2. Define parameters and link to other data sets to override the conditions specified in the **Runs** data sets.

3. Click the **Make Runs** button.

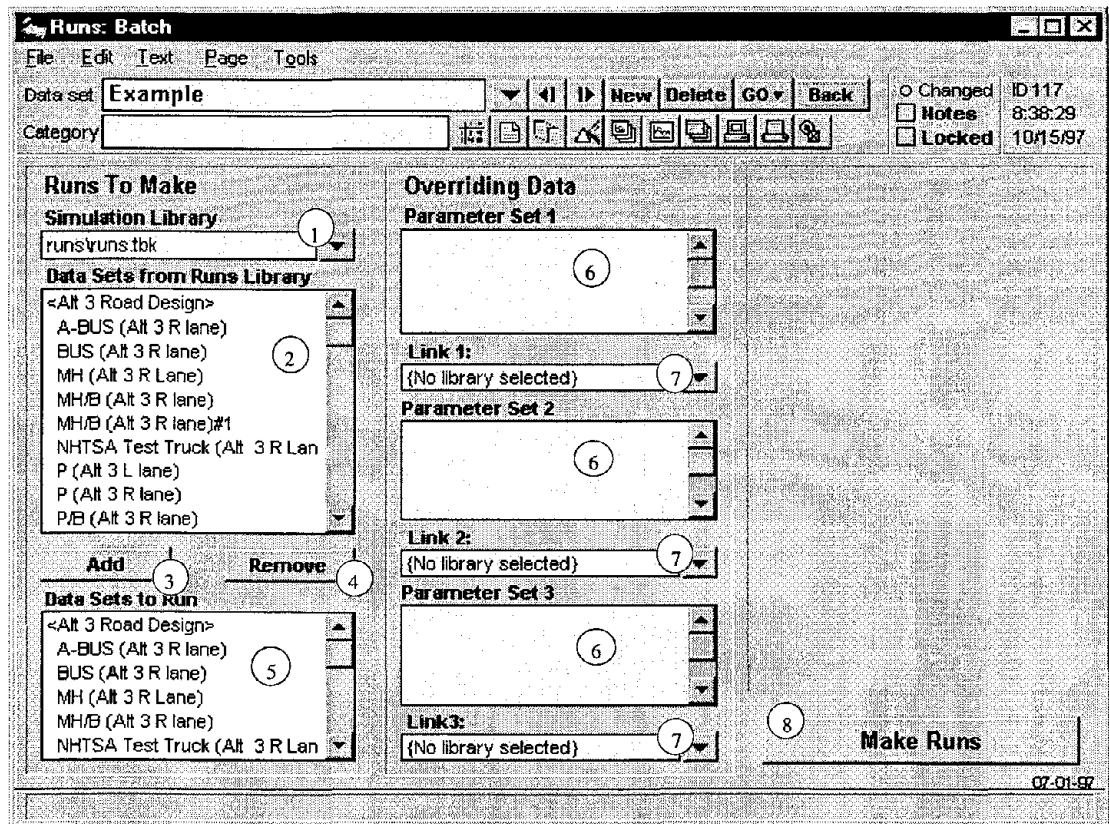
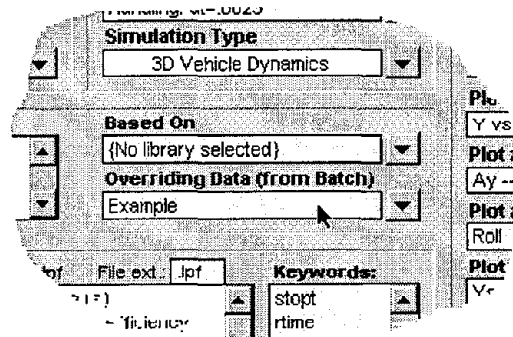


Figure 8.6. The Batch Runs screen.

If no overriding data values or links are provided in step 2, the effect is exactly the same as going to each of the data sets from the **Runs** library and making the run again. However, if any parameters are specified or links are made, each data set in the **Runs** library is modified to use the overriding data. A link called **Overriding Data (from Batch)** is made to this data set. For example, in Figure 8.6, the name of the data set is **Example**. If any runs were made, each affected run would have the following link set automatically:



This link is made so you can get from the runs data set back to the batch data set and see what parameters and links were used when the run was made. (For the example screen shown in Figure 8.6, there were no overriding parameters.)

User Settings and Controls

- ① **Runs link.** In order for the batch run option to work, a link is needed to an existing **Runs** library (e.g., `Runs\Runs.tbk`). When this data set is opened, the linked library is automatically scanned by VDM RoAD, and the names of all data sets are listed in the field below ②. The adjacent triangle button has a pull-down menu that can be used to link to a different **Runs** library.

- ② **The Data Sets From Runs Library list.** This is a list of all of the data sets in the **Runs** library whose local pathname is shown above ①. Any line in brackets (<>) indicates a category heading.

Double-click on a line to add that run to the list named **Data Sets to Run** ⑤. This is the same as selecting the line and then clicking the **Add** button ③. Double-click on a line in brackets (<>) to add all runs in the category to the list ⑤.

- ③ **The Add button.** Click to add the currently highlighted lines from the **Data Sets From Runs Library** list ② to the **Data Sets to Run** list ⑤. Shift-click to add all runs.
- ④ **The Remove button.** Click to remove highlighted data sets from the **Data Sets to Run** list ⑤. Shift-click to remove all.

Note: The **Remove** button only affects the list of **Data Sets to Run** ⑤.
No data sets are deleted from the data base.

- ⑤ **The Data Sets to Run list.** This is a list of the selected data sets from the **Runs** library. These are the data sets that will be run using the overriding data to the right (⑥, ⑦). Any line in brackets (<>) indicates a category heading.

Double-click on a line to remove it from the list. This is the same as selecting the line and then clicking the **Remove** button ④. Double clicking on a line in brackets will remove the entire category.


- ⑥ **Parameter Sets (1, 2, and 3).** These lists hold overriding data. If used, each line typically has a keyword followed by a value. For example, to set the speed, enter the line `speed 60`. See Appendix D for the keywords recognized by VDM RoAD models.

Any parameters specified in these fields will be used instead of the ones referenced in the **Runs** data set, or in data sets linked to the **Runs** data set.

This field should be left blank if you do not want to modify the input parameters or model descriptions for the runs to be made.

- ⑦ **Links (1, 2, and 3)**. Use these links to apply data from any other VDM RoAD library. If the data in the library can be applied to different parts of the vehicle (for example, a tire description can be applied to any of the axles on the vehicle), a reference to a part of the vehicle must be specified in the preceding **Parameter Set** ⑥.
- ⑧ The **Make Runs** button. Click to run all of the data sets listed in the **Data Sets to Run** list ⑤. Hold down the control key to interrupt the runs in progress.

Location in VDM RoAD

Accessed from the **Tools** menu and the ribbon bar button: 

File Location

Batch\Runs_bat\Runs_bat.tbk

Shock Absorbers

This screen defines the damping force from a shock absorber.

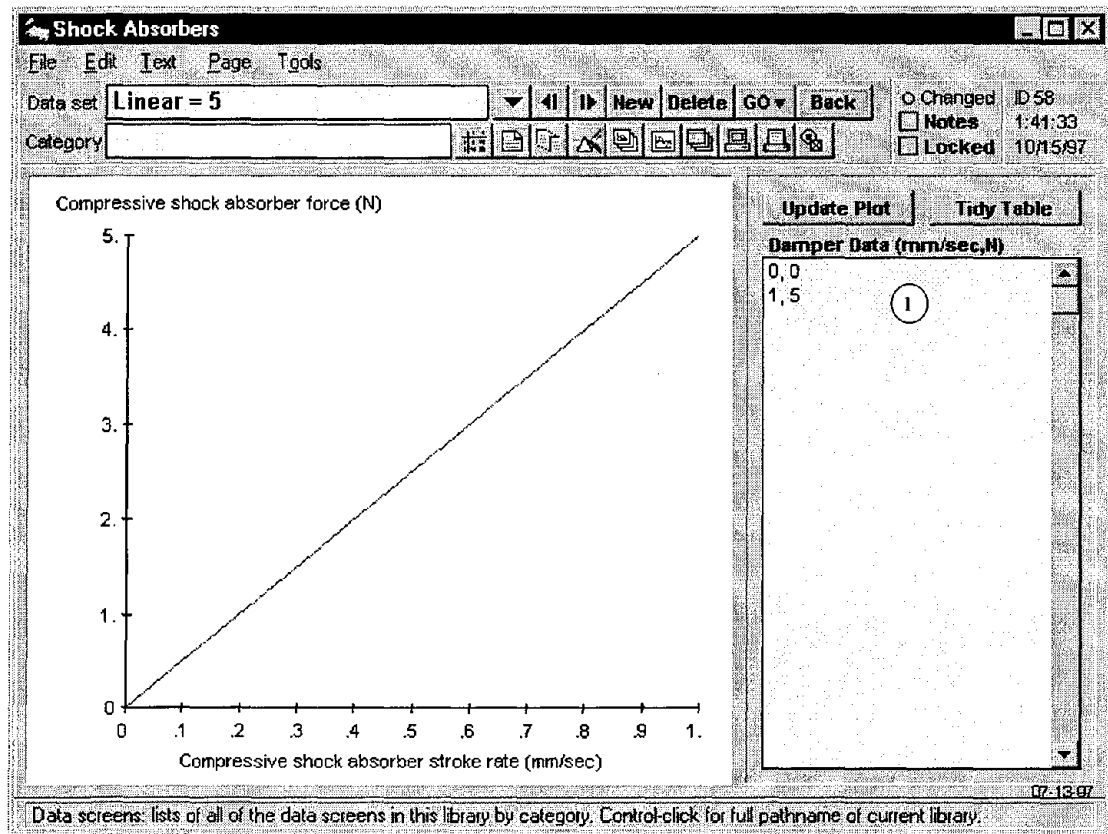
Discussion

Shock absorbers are dampers that produce a force resisting motion in the suspension. The force is dependent on velocity, but may vary nonlinearly with velocity because of the orifices and blow-off valves used in shock absorbers. In VDM RoAD the force-velocity relationship is represented with a table, which can be used to represent linear or nonlinear relationships.

In the case of the truck models in VDM RoAD, the shock absorbers are assumed to be oriented vertically in the positions shown in the suspension screen for solid axles (see page 201). If the shock absorbers are inclined, the values in this table must be adjusted to produce the vertical force as a function of vertical motion at the specified location.

The car model is set up under the assumption that the motion input for the shock absorber is not the same as the wheel vertical movement. A parameter is provided for the mechanical advantage of the suspension kinematics (see the car suspension screen, described on page 198). The shock absorber data can be for the shock as measured in isolation. If you have data that have been modified to account for the kinematics of the suspension, then specify a mechanical advantage of 1.0.

A single library contains shock absorber data for all VDM RoAD models (cars and trucks).



User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Two-column table of shock absorber compression rate (speed) and damping force (keyword = `FD_TABLE`). Values are positive for compression (jounce) and negative for extension (rebound).

The simulation programs use constant-gradient extrapolation with this table. For input values that are less than the range covered in the table, the first point is used to extrapolate the first gradient backward. The gradient is determined from the first two values of mm/s and N. For values of mm/s larger than the range covered, the last point is used to extrapolate forward using the gradient between the last two points. The table needs at least two lines (for the simplest case of a constant damping rate).

Location in VDM RoAD

Truck Models

VDM RoAD Startup

Runs

Loaded vehicle descriptions (Truck or combination vehicle)

Unit vehicles (unloaded truck, tractor, or trailer)
 Suspensions
 Shock Absorbers

Car Model

VDM RoAD Startup
 Runs
 Cars
 Suspensions
 Shock Absorbers

File Location

Vehicles\Parts\Shocks\Shocks.tbk

Springs: Car

The car springs are represented with tables of force as a function of spring compression. The tables can represent linear or nonlinear springs.

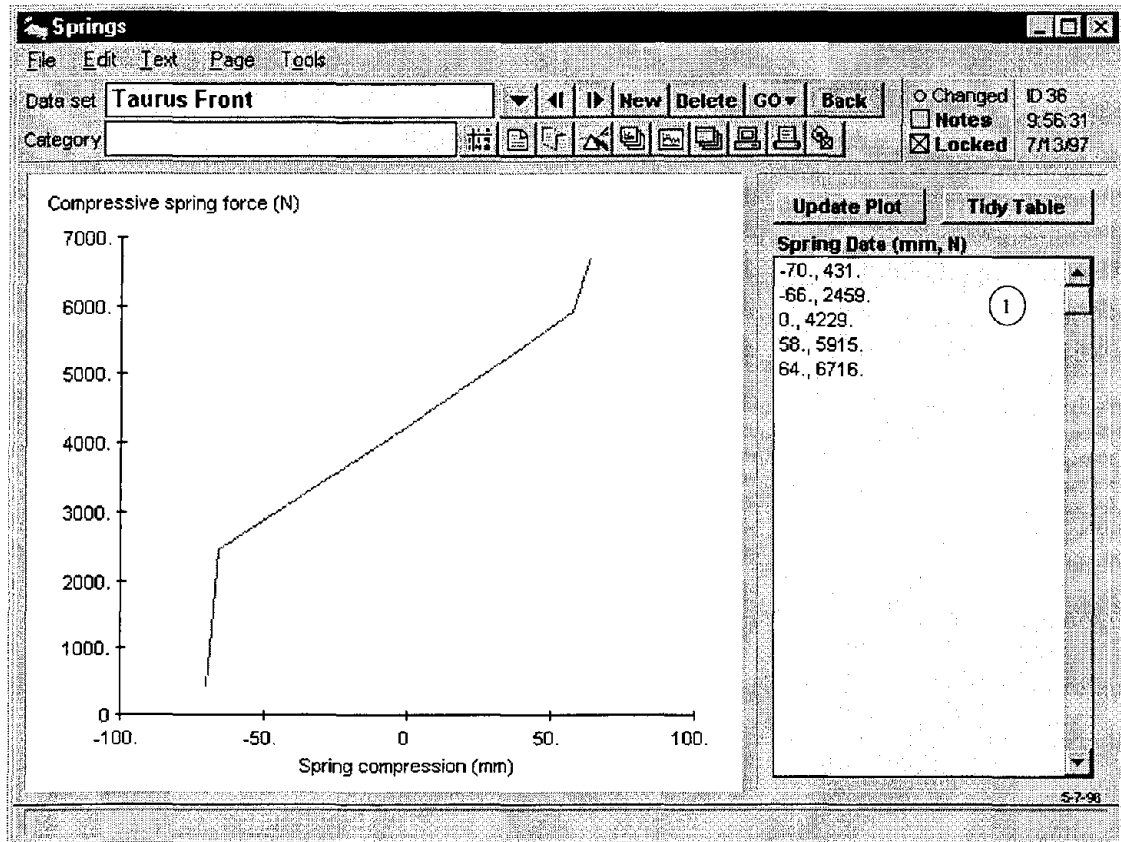
Discussion

In order to account for the mechanical advantage of the suspension kinematics, the car model includes a parameter on the suspension data screen (see page 198). If you have data for force and deflection at the wheel, then set the mechanical advantage parameter to 1.0.

Bump stops can be incorporated by an increase in spring rate at the deflection point at which the bump stops are encountered.

VDM RoAD includes two spring libraries: one for cars and one for heavy trucks. Both involve tables. However, the car spring model does not include friction effects, which are important for heavy trucks but not for cars. It is important that car spring data be used with the car model. The keywords used in the car and truck models are different, which means that if you somehow link to the wrong library, the values will not be used.

Note: VDM RoAD is set up to make it difficult to get the springs mixed up. All of the example data sets are linked to the right kind of libraries, and you have to work hard to change the library links.



User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Table field for envelope of spring force as a function of compression (keyword = FS_TABLE). This represents the force vs. deflection that would be measured for the spring by itself (not installed in the suspension).

The simulation programs use constant-gradient extrapolation with this table. For input values that are less than the range covered in the table, the first point is used to extrapolate the first gradient backward. The gradient is determined from the first two values of compression and force. For values of compression larger than the range covered, the last point is used to extrapolate forward using the gradient between the last two points. The table needs at least two lines (for the simplest case of a constant spring rate).

Location in VDM RoAD

VDM RoAD Startup
 Runs
 Cars
 Suspensions
 Springs

File Location

Vehicles\Parts\Sprg_car\Sprg_car.tbk

Springs: Truck

The Springs Screen is used to specify the force-displacement properties of the springs used on a truck suspension system, including their friction.

Discussion

Trucks commonly use leaf springs in their suspensions. Leaf springs may have a high level of interleaf friction, which makes the suspension spring force dependent on past motion. The simulation model uses the UMTRI leaf spring model, which allows the spring force to be dependent on where it is in its displacement and the direction it is traveling. The rate at which force builds up with deflection when the motion direction is changed is specified by the Beta value.

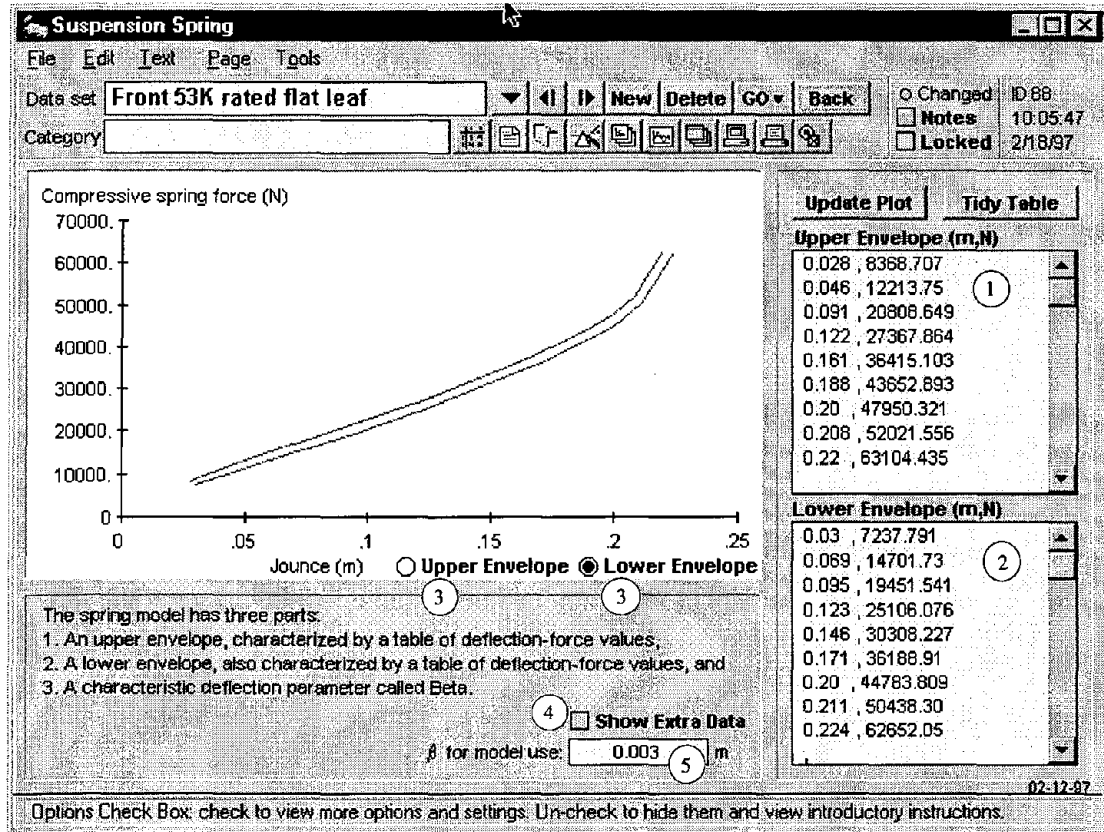
The spring model and its parameters are defined mathematically in Appendix E, along with the rest of the vehicle model.

In the event a suspension uses an alternate type of spring (e.g., air, torsion-bar) the same representation can be used. The availability of nonlinear rates and hysteretic friction simply allows you, the user, to model them more precisely.

Bump stops can be incorporated by an increase in spring rate at the deflection point at which the bump stops are encountered.

VDM RoAD includes two spring libraries: one for cars and one for heavy trucks. Both involve tables. However, the car spring model does not include friction effects, which are important for heavy trucks but not for cars. It is important that truck spring data be used with the truck models. The keywords used in the car and truck models are different, which means that if you somehow link to the wrong library, the values will not be used.

Note: VDM RoAD is set up to make it difficult to get the springs mixed up. All of the example data sets are linked to the right kind of libraries, and you have to work hard to change the library links.



User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

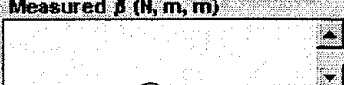
- ① Table field for envelope of spring force as a function of compression (keyword = SPRING_UPPER_ENVELOPE). This represents the force vs. deflection that would be measured while the spring is moving in compression (jounce).

The simulation programs use constant-gradient extrapolation with this table. For input values that are less than the range covered in the table, the first point is used to extrapolate the first gradient backward. The gradient is determined from the first two values of compression and force. For values of compression larger than the range covered, the last point is used to extrapolate forward using the gradient between the last two points. The table needs at least two lines (for the simplest case of a constant spring rate).

- ② Table field for envelope of spring force as a function of rebound (keyword = SPRING_LOWER_ENVELOPE). This represents the force vs. deflection that would be measured while the spring is moving in extension (rebound).

The simulation programs use constant-gradient extrapolation with this table, as described above for the upper envelope data.

- ③ **Upper Envelope** and **Lower Envelope** radio buttons. Click in the plot area (with the **Locked** box of the ribbon bar not checked) and the deflection-force coordinates of the point where you click are appended to one of the table fields: ① or ②. These two buttons determine which field is active. The numbers are appended to the field whose associated radio button is selected.
- ④ **Show Extra Data** check box. Check the box to see additional information that can be kept with the spring data. None of the additional information is used by the VDM RoAD simulation models.

Force offset:	0	N	Measured β (N, m, m)  <input checked="" type="checkbox"/> Show Extra Data β for model use: 0.003 m
Rated load:	53378	N	
Spring type:	Flat Leaf		
Suspension type:	single, steer		
Suspension class:	single		
Origin:			

- ⑤ Beta parameter (keyword = SPRING_BETA). This is approximately 1/3 the distance needed to travel through a spring hysteresis (friction) loop. This parameter is described mathematically in Appendix E.

Location in VDM RoAD

VDM RoAD Startup

Runs

Loaded vehicle descriptions (Truck or combination vehicle)

Unit vehicles (unloaded truck, tractor, or trailer)

Suspensions

Springs

File Location

Vehicles\Parts\Susp_trk\Sprg_trk\Sprg_trk.tbk

Steering System: Truck

This screen is used to define the essential geometry and compliance coefficients of the steering system for heavy trucks. The data on this screen are not used by the car model.

Discussion

The geometry and compliance of the steering system strongly influence the vehicle's directional response. The geometry at the road wheels (caster angle, kingpin inclination angle, and kingpin offset) influences the torque about the steer axis generated by the

forces and moments imposed at the tire contact patches. The combined torque from both wheels reacts against the compliance of the steering linkages, causing the road wheels to deviate from the command steer angle input at the steering wheel.

When the suspension deflects in jounce (compression) and rebound (extension), errors in the geometry can produce steer inputs known as bump steer. With brake application the axle wrap-up in the pitch direction can introduce further steering deviations, as well as change in the caster angle. The parameters needed to model these effects are provided on this screen.

VDM RoAD includes two steering system libraries: one for cars and one for heavy trucks. It is important that truck data be used with the truck models. The keywords used in the car and truck models are different, which means that if you somehow link to the wrong library, the values will not be used.

Note: VDM RoAD is set up to make it difficult to get the libraries mixed up. All of the example data sets are linked to the right kind of libraries, and you have to work hard to change the library links.

Steering System

File Edit Text Page Tools

Data set: **Example Steering System** | New | Delete | GO | Back

Category: | Changed | ID12 | 3:25:46 | Notes | 3/18/97 | Locked

	Left Wheel	Right Wheel	
King-pin offset at center:	0.1	0.1	m
King-pin inclination angle:	7.2	7.2	deg
Caster angle:	5.2	5.2	deg

System Parameters

Axle wrap compliance:	0.000186	deg/(N-m)
Steer vs. axle wrap:	0.14	deg/deg
Ave steer (L, R) vs. jounce:	0.00627	deg/m

Optional Steering Data

9

The kinematics links define the road-wheel steer angle vs. driver steering wheel angle with no applied forces

Steering Kinematics, Left Side
Default LF Steer 10

Steering Kinematics, Right Side
Default RF Steer 10

Options (L): 11

Options (R): 11

02-12-97

This data set is locked. Unlock data set to view the context sensitive help in this status bar.

User Settings

- ① Steering torsional compliance (keyword = CSMZS) . Compliance of the linkages between the steering wheel and the left road wheel. The compliance is expressed in terms of the steer that results due to torque applied to the wheel.
- ② Tie rod torsional compliance (keyword = CSMZR) . Compliance of the tie-rod linkage connecting the left and right road wheels. The compliance is expressed in terms of the steer that results due to torque applied to the wheel.
- ③ Kingpin offset (keyword = LKPO (1 or 2, 1 = Left wheel, 2 = Right wheel)). Lateral distance between the kingpin axis and the center of the wheel on the wheel spin axis.
- ④ Kingpin inclination (keyword = AKPI (1 or 2, 1 = Left wheel, 2 = Right wheel)). Kingpin axis inclination in the lateral plane of the vehicle.
- ⑤ Caster angle (keyword = ACAST (1 or 2, 1 = Left wheel, 2 = Right wheel)). Kingpin axis inclination in the longitudinal plane of the vehicle.
- ⑥ Axle wrap-up compliance (keyword = CWRAP) . Compliance of the axle on its suspension in the pitch direction, which allows it to roll forward when acted upon by brake torque.
- ⑦ Wrap-up steer coefficient (keyword = RWRAPSTR). Coefficient for road-wheel steer per unit of axle wrap-up.
- ⑧ Bump steer coefficient (keyword = RBMPSTR). Coefficient for average left and right steer per unit of axle compression (jounce) displacement.
- ⑨ Optional miscellaneous field to support customized steering models. Not used in the standard VDM RoAD.
- ⑩ **Steering Kinematics** links. The linked data sets describe the steering kinematics for the left and right road wheels.
- ⑪ Optional links. These links support customized steering models but are not used in the standard VDM RoAD.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Vehicle Unit: (empty truck or tractor)

Steering System

File Location

Vehicles\Parts\Strs_trk\Strs_trk.tbk

Steering System: Car

This screen is used to define the essential geometry and compliance coefficients of the steering system for a car.

Steering System

File Edit Text Page Tools

Data set: **Taurus Steering** New Delete GO Back

Category: []

o Changed ID:40
 Notes 4:46:42
 Locked 12/17/96

Steering Kinematics

Steering wheel input → $\frac{1}{R_{sw}}$ → Left-front shaping function → Left-front steer angle
 → Right-front shaping function → Right-front steer angle
 → Rear/front steering ratio → Rear wheel steer angle

Steering Kinematics

Steering gear ratio (Rsw): 15.97 ¹ deg/deg

Left-Front Steer Kinematic ³: Taurus Left

Right-Front Steer Kinematic ³: Taurus Right

Rear/Front Steering ratio ⁴: Front Wheel Steer

Steering Compliance

Steering system compliance: 0.001 ² deg/(N-m)

Steering system compliance is the total compliance (front) minus the suspension compliance. Compliance is the ratio of steer due to Mz (aligning moment) per unit of Mz. Measure with steering wheel fixed and apply Mz to both front wheels.

Notes: Taurus data from VRTC. 10/95

5-7-98

Discussion

The steering of the wheels is determined by properties of the steering system and also the suspension. This library has parameters that are associated only with the steering system. As shown by the flow chart on the screen, these parameters affect how the steering wheel angle from the driver causes the wheels to steer.

Other parameters, which describe the interactions between suspension movements and steer angle, are contained in the suspension screen (see page 198). Compliance effects are associated with the suspension, except for a single overall compliance parameter that is associated only with the steering system.

VDM RoAD includes two steering system libraries: one for cars and one for heavy trucks. It is important that car data be used with the car model. The keywords used in the car and truck models are different, which means that if you somehow link to the wrong library, the values will not be used.

Note: VDM RoAD is set up to make it difficult to get the libraries mixed up. All of the example data sets are linked to the right kind of libraries, and you have to work hard to change the library links.

User Settings

- ① **Steering gear ratio** (keyword = RSW). This is the ratio between the steering wheel angle and a nominal steer of the front wheels. As shown in the flow chart on the data screen, the steer of the road wheels is the product of the steering wheel angle, the inverse of the gear ratio, and a shaping function.

The definition of a “nominal steer” of the road wheel is left to the user. The quotient of the kinematical shaping function divided by the steering gear ratio is an important property of the vehicle. However, the “nominal steer” does not directly enter into the vehicle response.

- ② **Steering system compliance** (keyword = CSMZF). This is a compliance in the steer system that is not accounted for by the individual suspension steering compliance in the suspension screen (see page 198). It would be measured by a test in which the same aligning torque is applied to both front wheels with the steering wheel locked. The total compliance [deg/(N-m)] includes the compliance of the suspension. With the suspension compliance alone, a certain amount of steer would be expected. Subtract this effect to determine the residual steer, and divide it by the sum of the aligning torque applied to both front wheels.
- ③ **Steering Kinematics** links. The linked data sets describe the steering kinematics for the left and right road wheels.
- ④ **Rear/Front Wheel Ratio** link. The linked data set describes the kinematics for four-wheel steering.

Location in VDM RoAD

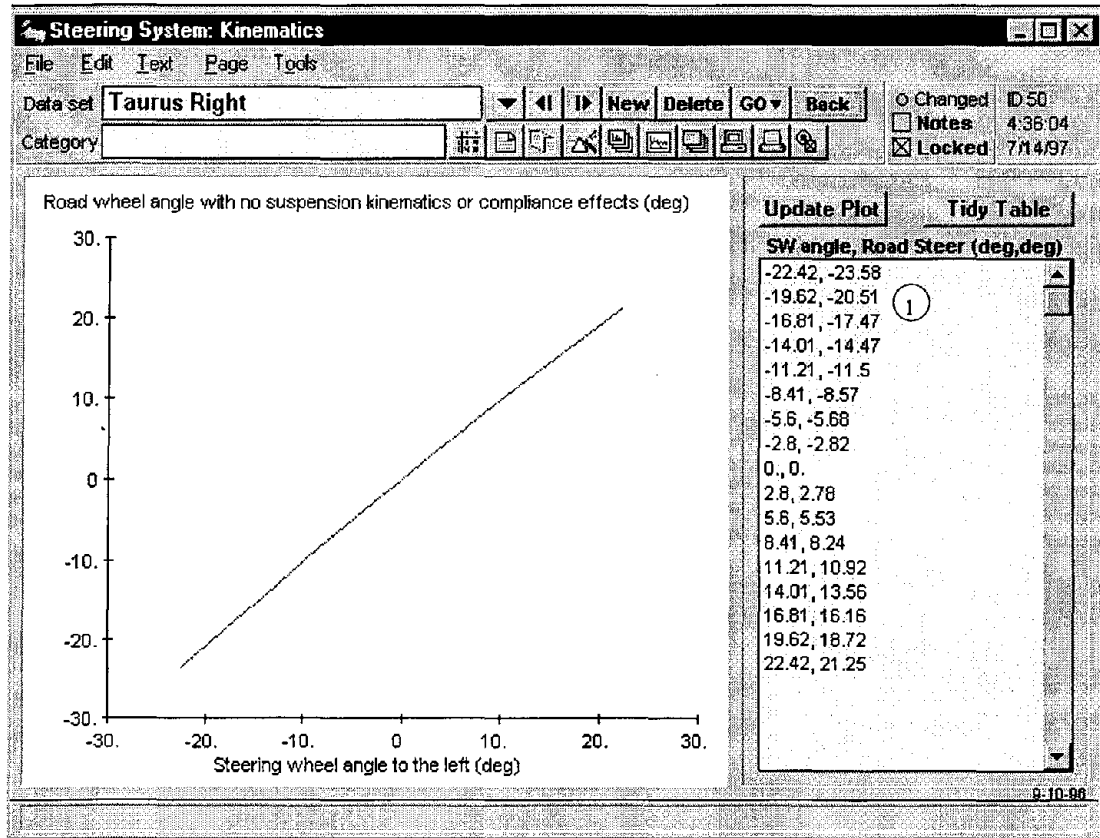
```
VDM RoAD Startup
  Runs
    Cars
      Steering System
```

File Location

```
Vehicles\Parts\Strs_car\Strs_car.tbk
```

Steering System: Kinematics

Use this screen to define the relationship between the steering wheel angle and the steer angles of the left and right road wheels.



Discussion

This screen is used to define the left and right road wheel steer angles as a function of steering wheel angle with table look-ups. Separate tables are used for the left and right wheels. The relationship between steering wheel angle and road wheel angles should reflect the overall ratio of the steering hardware and Ackerman geometry.

A single library contains steering kinematical data for all VDM RoAD models (cars and trucks). However, the data for trucks usually include the effect of gear ration, whereas the data for cars are generally normalized to exclude the scaling due to the gear ratio.

User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Two-column table of steering wheel angle and road wheel angle (keyword = STEERRW_TABLE (ISIDE); for ISIDE, 1=left side, 2=right side). Positive steer is to the left.

For the truck models, the table includes the effect of gearing down. For the car model, a separate parameter is available for the gear ratio. For the car model, the table involves the “nominal” road wheel angle [nominal angle = (steering wheel angle)/(steering gear ratio)].

The simulation programs use constant-gradient extrapolation with this table. For input values that are less than the range covered in the table, the first point is used to extrapolate the first gradient backward. The gradient is determined from the first two values of steering wheel and road wheel angle. For steering wheel angles larger than the range covered, the last point is used to extrapolate forward using the gradient between the last two points. The table needs at least two lines (for the simplest case of a constant gear ratio).

Location in VDM RoAD

Truck Models

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Vehicle Unit: (empty truck or tractor)

Steering System

Steering System: Kinematics

Car Model

VDM RoAD Startup

Runs

Cars

Steering System

Steering System: Kinematics

File Location

Vehicles\Parts\Str_kin\Str_kin.tbk

Steering System: Rear/Front Wheel Ratio

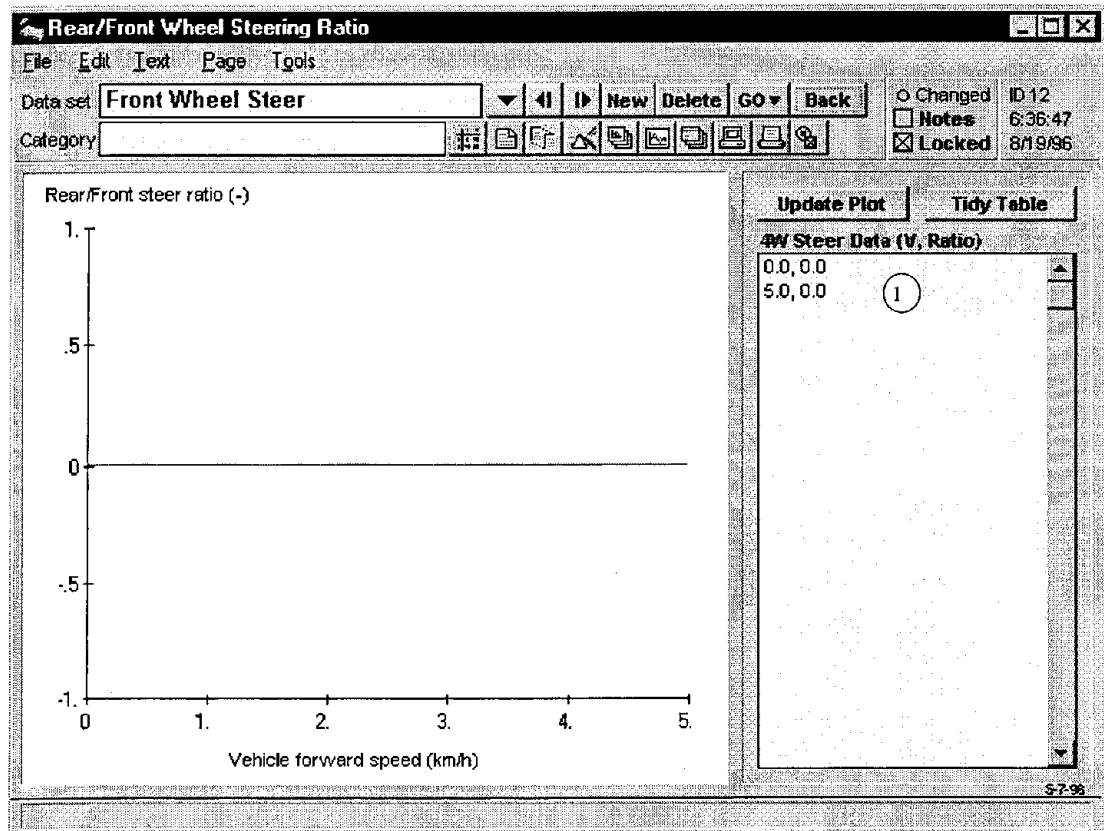
The car model in VDM RoAD has a provision for simulating a common four-wheel steering design. The steer of the rear wheels is specified as a ratio of the nominal front-wheel steer, with the ratio being speed dependent.

This library is not used for the heavy truck models.

Discussion

In order to simulate cars with four-wheel steering, a simple model is used in which the steering wheel angle is divided by a steering gear ratio (see the **Steering System: Car** screen on page 193), and then divided by another ratio that depends on speed. The result is used as the nominal steering angle for the rear wheels. Those angles are modified according to kinematical relations involving rear suspension deflection, along with rear suspension compliance.

To simulate cars without rear-wheel steer, set the gain to zero for all speeds.



User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Two-column table of the ratio of between the nominal rear steer angle and the nominal steer angle, as a function of vehicle forward speed (keyword = STEERRF_TABLE).

The simulation programs use constant-gradient extrapolation with this table. For input values that are less than the range covered in the table, the first point is used to extrapolate the first gradient backward. The gradient is determined from the first two values of speed and ratio. For speeds larger than the range covered, the last point is used to extrapolate forward using the gradient between the last two points. The table needs at least two lines (for the simplest case of a constant ratio).

Location in VDM RoAD

VDM RoAD Startup
 Runs
 Cars
 Steering System

Steering System: Rear/Front Wheel Ratio

File Location

Vehicles\Parts\Str_4w\Str_4w.tbk

Suspensions: Car

Use this screen to define suspension properties for the car models in VDM RoAD. This library is not used for the heavy truck models.

Suspensions

File Edit Text Page Tools

Data set: **Taurus Front Suspension** | New | Delete | GO | Back

Category: | Changed ID 50 | Notes 7:54:56 | Locked 7/15/97

End view of Wheel Movement

1540 (1) | Wheel center

Roll center

130 (2) | 320 (3)

All dimensions are in millimeters

Side View of Wheel Movement

Ratio: longitudinal movement per unit of vertical movement

.077 (4)

Ratio is positive if wheelbase increases when suspension is compressed

Springs and Dampers

Suspension compression at the wheel

Compression of component (spring or damper)

Spring: Taurus Front (7)

Damper: Taurus Shock Absorber (8)

Ratios of component compression per unit of suspension compression:

Spring: 1.0 (14) | Damper: 1.0 (15) m/m

Auxiliary Roll Stiffness

Stiffness*: 364 (16) N-m/deg

*Suspension roll stiffness minus roll stiffness due to ride stiffness and track width

Suspension Steer and Toe Compliance

Toe vs. Fx: 9 4283e-3 deg/N

Steer vs. Fy: -0.270e-10 (10) deg/N

Steer vs. Mz: 11 .360e-2 deg/N/m

To measure compliance, apply Fx, Fy, and Mz to the wheel from the ground. Apply Mz to both wheels in opposite directions.

Steer wheel plane

Fx Fy Mz

Toe vs. Jounce (5)

Taurus Front Toe

Camber vs. Jounce (6)

Taurus Front Camber

Solid Axle Properties (if applicable)

Roll-steer coefficient: 12 deg/deg

Yaw moment of inertia: 13 kg-m²

Roll-steer: deg of axle steer per deg of vehicle roll, relative to axle

5-7-98

Discussion

This screen shows parameters and links that characterize the most important properties of an automotive suspension, with respect to predicting the braking and handling behavior.

The kinematics of the suspension linkages are described in terms of how a wheel moves laterally and longitudinally as the suspension deflects vertically. The lateral movement affects the transfer of tire lateral force to the body and the resulting body roll. These effects are accounted for with the concept of a roll center. In the model, the roll center

height is used to define the amount a wheel moves laterally as the suspension is compressed. The mathematical model does not include an actual roll center point.

The longitudinal movement has a similar effect involving pitch due to longitudinal tire force.

Changes in steer due to compliance and suspension kinematics can drastically affect the response of a car with respect to steering inputs. The compliance is represented with linear coefficients. Kinematical effects are represented with nonlinear tables.

User Settings

- ① Track width at the simulation load condition (keyword = LTK (IAXLE)).
- ② Height of axle roll center above ground at the simulation load condition (keyword = HRC (IAXLE)). This is typically a value between 0 and 2100 mm for independent suspensions and from 300 to 500 mm for solid axle suspensions.
- ③ Height of wheel center at design load (keyword = HWC (IAXLE)).
- ④ Wheelbase change per unit jounce (keyword = RAP (IAXLE)). This should be a positive number for both front and rear suspensions, with a value between 0.05 and 0.25 mm/mm.

Note: The sign convention is designed such that a positive value means the wheel moves away from the center of the vehicle as it moves up. In the front, this is called anti-dive. In the rear, anti-squat.

- ⑤ **Toe Angle** link. Link to library of data sets with toe angle as a function of suspension compression.
- ⑥ **Camber Angle** link. Link to library of data sets with camber angle as a function of suspension compression.
- ⑦ **Spring** link to library with car spring data.
- ⑧ **Shock absorber** link to library with shock absorber data.
- ⑨ Coefficient for change in toe per change of tire longitudinal force (keyword = CTFX (IAXLE)). A forward force (driver) can bend a suspension forward, steering the wheel inward (positive toe due to suspension compliance). Therefore this parameter is likely to have a small but positive value.
- ⑩ Coefficient for change of steer angle per change of tire lateral force (keyword = CSFY (IAXLE)). For wheels that can be steered, the steer axis is usually inclined to intersect the ground in front of the center of tire contact. Thus, a positive lateral force (to the left), acting behind the steer axis, usually causes some steer to the right (negative). Therefore, this coefficient is likely to have a small negative value for a steered wheel. For rear wheels, it should have a value close to zero.
- ⑪ Coefficient for change of steer angle per change of tire aligning torque (keyword = CSMZ (IAXLE)). The suspension elements usually deflect when a steering torque is applied to

the wheel. Because the steer and moment have the same sign convention, the compliance coefficient is nearly always positive.

Steering as a result of aligning torque is due both to compliance in the suspension and to compliance in the steering column. If equal and opposite steering torques are applied to both the left and right road wheels, the resulting steer deflection is mainly due to the suspension alone.

The suspension compliance is subtracted from the total steering compliance that would be measured for one wheel, to determine the compliance for the steering column (specified in the **Steering System: Cars** screen, on page 193).

- ⑫ Roll-steer coefficient. This coefficient is used for car suspensions with solid axles. It is not used for the car model in VDM RoAD.
- ⑬ Yaw moment of inertia for axle. This coefficient is used for car suspensions with solid axles. It is not used for the car model in VDM RoAD.
- ⑭ Ratio of spring jounce to jounce at wheel (keyword = RSPRING (IAXLE)). This determines the mechanical advantage of the suspension with respect to the spring. The value is typically between 0.5 (for some SLA suspensions) and 1.0 (for some MacPherson strut suspensions).
- ⑮ Ratio of damper jounce to jounce at wheel (keyword = RDAMP (IAXLE)). This determines the mechanical advantage of the suspension with respect to the shock absorber. The value is typically between 0.5 (for some SLA suspensions) and 1.0 (for some MacPherson strut suspensions).
- ⑯ Auxiliary roll stiffness, including stabilizer bar (keyword = KAUX (IAXLE)). This coefficient is provided to account for the difference between the overall roll stiffness and the stiffness provided by the springs alone. It is usually positive.

The overall roll stiffness of a suspension is the roll moment (N-m) needed to roll the vehicle body one degree, with the other suspension disconnected, and the tires replaced with rigid elements. The springs provide a certain amount of roll stiffness, based on the spring rate, the mechanical advantage of the suspension, and the track width. The total roll stiffness may differ from the rate computed for the springs alone. Additional stiffness is provided by anti-sway bars and by over-constrained suspension linkages.

On the other hand, additional compliance can be introduced through software attachment points, leading to a negative value for auxiliary stiffness.

Location in VDM RoAD

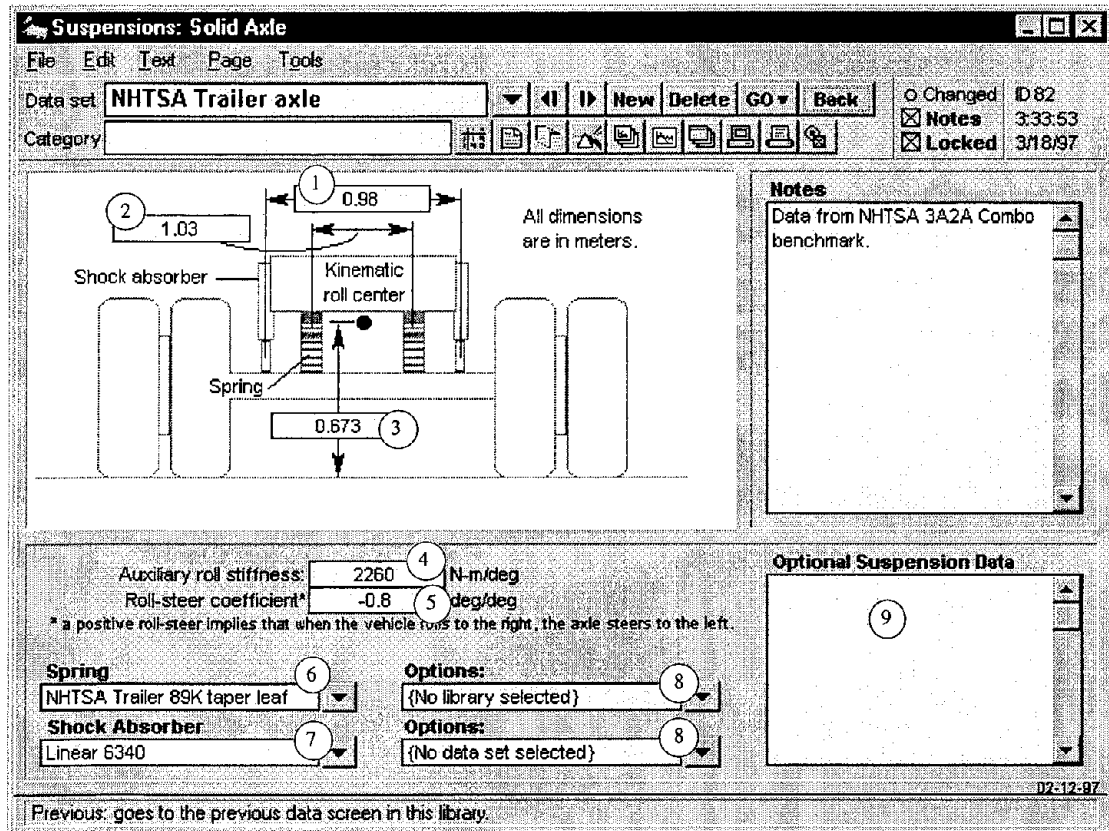
VDM RoAD Startup
 Runs
 Cars
 Suspensions

File Location

Vehicles\Parts\Susp_car\Suspen.tbk

Suspensions: Solid Axle

Use this screen to specify the relevant dimensional properties of the solid axles used on trucks and trailers. This library is not used for the car model.



Discussion

The lateral separation between the springs is important because it causes the springs to resist roll. Likewise, lateral separation of the shock absorbers produces roll damping. The roll center height is the elevation at which lateral forces generated on the axle are transmitted to the sprung mass. The roll center is assumed to fall on the center of the vehicle, and any deviations when the vehicle rolls are neglected in the simulation. Roll center heights are determined by the kinematics of the suspension linkages.

Additional (auxiliary) roll stiffness can be produced in the suspension by mechanisms such as stabilizer bars or twist in the leaf springs. Specification of an auxiliary roll stiffness allows you to replicate the influences of these other mechanisms.

The roll steer coefficient quantifies the degree to which the axle may steer as the vehicle takes on a roll angle.

User Settings

- ① Shock absorber spacing (keyword = LTD (IUNIT, IAXLE)). Lateral distance between the centerlines of left and right shocks.
- ② Spring spacing (keyword = LTS (IUNIT, IAXLE)). Lateral distance between the centerlines of left and right springs.
- ③ Roll center height (keyword = HRC (IUNIT, IAXLE)). Height of the roll center above the ground at the simulation load condition.
- ④ Auxiliary roll stiffness (keyword = KAUX (IUNIT, IAXLE)). Suspension roll stiffness not accounted for by lateral spacing of the springs.
- ⑤ Roll steer coefficient (keyword = RSTR (IUNIT, IAXLE)). Coefficient describing degrees of axle steer per degree of frame roll relative to the axle. A positive value means that when the vehicle rolls to the right, the axle steers to the left.
- ⑥ **Spring** link. This links to spring data for this axle.
- ⑦ **Shock Absorber** link. This links to shock absorber table for this axle.
- ⑧ Optional links. These links are for use with custom models but are not used with the standard VDM RoAD package.
- ⑨ Miscellaneous field for additional data for custom models. Not used for standard VDM RoAD models.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Vehicle Unit: (empty truck, tractor, or trailer)

Suspensions: Solid Axle

File Location

Vehicles\Parts\Susp_trk\Susp_trk.tbk

Tires: Aligning Moment

Use this screen to define tire aligning moment as a 2D table-lookup function of slip angle and vertical load.

Discussion

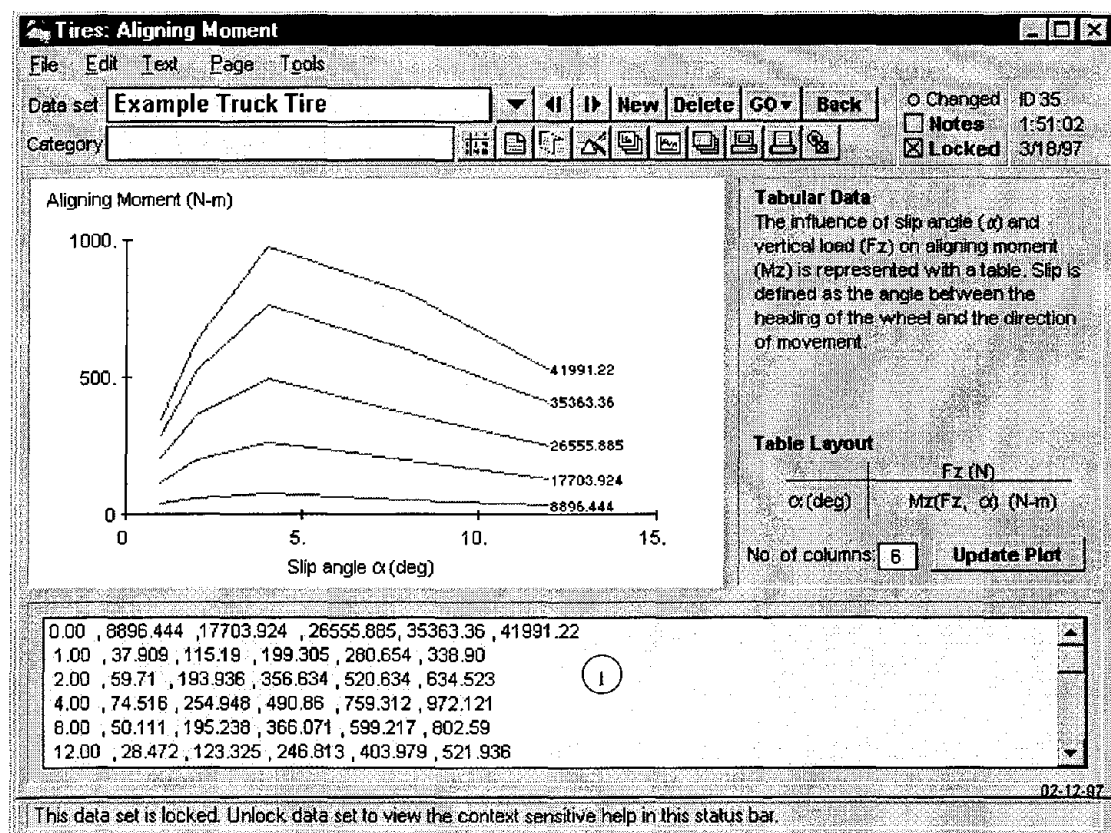
When a tire assumes a slip angle and develops a lateral force, the resultant force is generated toward the rear of the contact patch. It is standard practice to measure tire forces and moments at the center of the contact patch. When the lateral force is resolved

to this point, a moment, known as the aligning moment, results. The aligning moment is nominally related to the lateral slip angle and the vertical load.

This screen allows you to enter a 2D table defining tire aligning moment as a function of lateral slip angle and vertical load.

The tire model used in VDM RoAD assumes that the tire behavior is symmetric about the origin with respect to lateral slip. Thus, only one quadrant of data is needed, for positive values of slip.

A single library contains aligning moment data for all VDM RoAD models (cars and trucks).



User Settings

Note: User settings that are common for all 2D tabular data screens are described in Chapter 7, in the section **2D Tabular Data**.

- ① Tabular data for tire aligning moment (keyword = MZ_CARPET (IUNIT, IAXLE)). The first value, in column 1 and row 1, is a place holder. It must be present, but the value is not used. The first row has the vertical loads that apply for each column (except for the first entry). The first column has the slip angles that apply for each row (except for the

first entry). All remaining numbers are values of aligning moment as a function of vertical force and slip angle.

The tire model is symmetric for slip, using the same data for positive and negative slip angles. For small amounts of slip angle, aligning moment is usually positive. For large amounts, it remains positive for some tires, but can go negative for others.

The VDM RoAD tire models assume that a row of zero values exists, such that for zero slip there is zero moment, regardless of the load. Also, the models assume that a column of zeros exists, such that for zero load there is no moment, regardless of slip. You should not enter a row or column of zeros, as this can cause simulation errors.

For values of slip or load that are higher than the range covered by the table, the simulation programs extrapolate using the gradient of M_z between the last two points in the table.

If the table does not cover large slip angles, simulation can predict moments that are much too large, due to the extrapolation. Due to compliance in the steering system, the model would then predict large steering angles that would not be feasible. Therefore, if the data do not cover large slip angles, you should add a row with slip 1 degree larger than the previous row, and repeat all the moment values. This means that for slip angles outside the range of the table, the extrapolated moment will be reasonable.

Location in VDM RoAD

Truck Models

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Vehicle Unit: (empty truck, tractor, or trailer)

Tires: Table-Lookup Model

Tires: Aligning Moment

Car Model

VDM RoAD Startup

Runs

Cars

Tires: Table-Lookup Model

Tires: Aligning Moment

File Location

Vehicles\Tires\Tires_mz\Tires_mz.tbk

Tires: Lateral Force

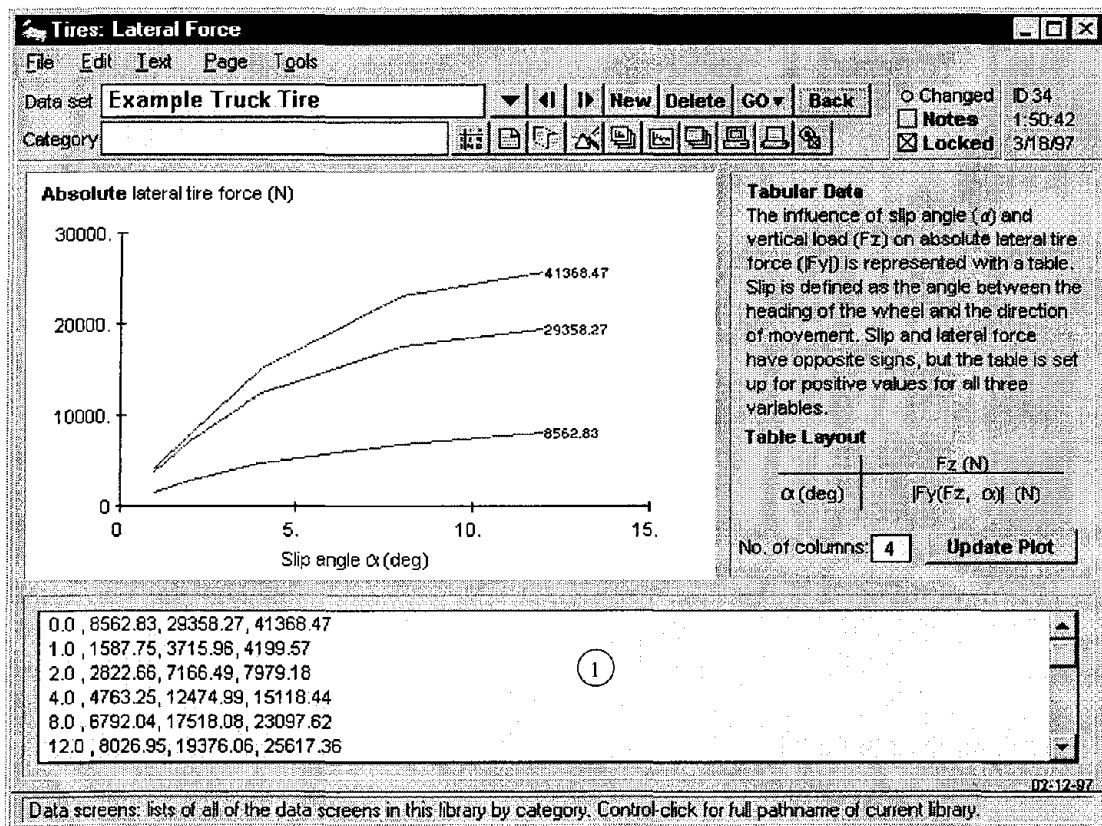
Use this screen to define the absolute lateral tire force as a function of lateral slip angle and tire vertical load.

Discussion

When a tire generates lateral force by slipping sideways with respect to the direction in which it is pointed, the angle between the wheel heading and the tire velocity vector is known as slip angle. The lateral force is strongly dependent on the slip angle and the vertical load. This screen allows you to enter a 2D table defining lateral force as a function of slip angle and vertical load.

The tire model used in VDM RoAD assumes that the tire behavior is symmetric about the origin with respect to lateral slip. Thus, only one quadrant of data is needed, for positive values of slip. With typical vehicle dynamics sign conventions, the lateral force has the opposite sign of the slip angle: positive slip induces negative side force. To avoid possible errors with minus signs, and to create more intuitive graphs, the data screen requires all values to be positive. This is why the graph is labeled “**Absolute** lateral force.”

A single library contains lateral tire force data for all VDM RoAD models (cars and trucks).



User Settings

Note: User settings that are common for all 2D tabular data screens are described in Chapter 7, in the section **2D Tabular Data**.

- ① Tabular data for tire side force (keyword = `FY_CARPET (IUNIT, IAXLE)`). The first value, in column 1 and row 1, is a place holder. It must be present, but the value is not used. The first row has the vertical loads that apply for each column (except for the first entry). The first column has the slip angles that apply for each row (except for the first entry). All remaining numbers are values of absolute lateral force as a function of vertical force and slip angle.

The tire model is symmetric, using the same data for positive and negative slip angles. Although positive slip causes negative lateral force, the sign conversion is handled internally to avoid the need for entering large amounts of negative data. Therefore, only positive values should be entered into the table.

For values of slip or load that are higher than the range covered by the table, the simulation programs extrapolate using the gradient of F_y between the last two points in the table.

The VDM RoAD tire models assume that a row of zero values exists, such that for zero slip there is zero lateral force, regardless of the load. Also, the models assume that a column of zeros exists, such that for zero load there is no lateral force, regardless of slip. You should not enter a row or column of zeros, as this can cause simulation errors.

Location in VDM RoAD

Truck Models

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Vehicle Unit: (empty truck, tractor, or trailer)

Tires: Table-Lookup Model

Tires: Lateral Force

Car Model

VDM RoAD Startup

Runs

Cars

Tires: Table-Lookup Model

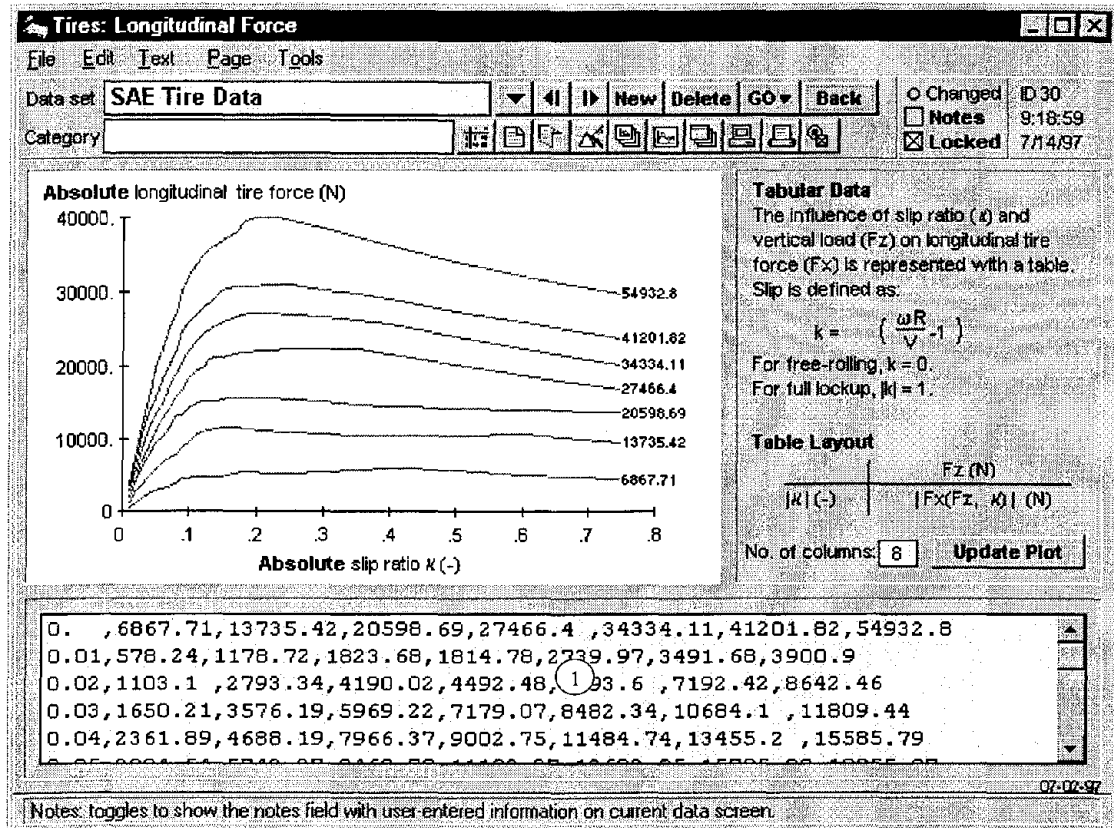
Tires: Lateral Force

File Location

Vehicles\Tires\Tires_fy\Tires_fy.tbk

Tires: Longitudinal Force

Use this screen to define longitudinal tire force as a function of slip ratio and tire vertical force.



Discussion

A tire generates longitudinal force when the rim speed (at the static tire radius) is different from the vehicle forward speed. The difference in speed is known as longitudinal slip when expressed in non-dimensional form. On dry pavements at normal travel speed, the wheel must slip at around 10% to 20% of its forward velocity to produce a maximum longitudinal force. Above this slip level the force diminishes. This screen allows entry of a 2D table defining longitudinal force as a function of longitudinal slip and vertical load.

The tire model used in VDM RoAD assumes that the tire behavior is symmetric about the origin with respect to longitudinal slip. Thus, only one quadrant of data is needed, for positive values of slip. Longitudinal slip is defined as a negative quantity during braking. Longitudinal force is also defined to be negative during braking. To avoid possible errors with minus signs, and to create more intuitive graphs, the data screen requires both slip

and longitudinal force to be positive. This is why the graph is labeled “**Absolute longitudinal tire force**” and “**Absolute slip ratio**.”

A single library contains longitudinal tire force data for all VDM RoAD models (cars and trucks).

User Settings

Note: User settings that are common for all 2D tabular data screens are described in Chapter 7, in the section **2D Tabular Data**.

- ① Tabular data for tire longitudinal force (keyword = `FX_CARPET (IUNIT, IAXLE)`). The first value, in column 1 and row 1, is a place holder. It must be present, but the value is not used. The first row has the vertical loads that apply for each column (except for the first entry). The first column has the slip ratios that apply for each row (except for the first entry). All remaining numbers are values of longitudinal force as a function of vertical force and slip ratio.

Only positive values should be entered into the table. The tire model is symmetric, using the same data for drive and braking force.

For values of slip or load that are higher than the range covered by the table, the simulation programs extrapolate using the gradient of F_x between the last two points in the table.

The VDM RoAD tire models assume that a row of zero values exists, such that for zero slip there is zero longitudinal force, regardless of the load. Also, the models assume that a column of zeros exists, such that for zero load there is no longitudinal force, regardless of slip. You should not enter a row or column of zeros, as this can cause simulation errors.

Location in VDM RoAD

Truck Models

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Vehicle Unit: (empty truck, tractor, or trailer)

Tires: Table-Lookup Model

Tires: Longitudinal Force

Car Model

VDM RoAD Startup

Runs

Cars

Tires: Table-Lookup Model

Tires: Longitudinal Force

At low speeds the determination of tire forces can become erratic because of numerical problems in calculating the slip conditions. In order to avoid these problems, you can specify a cut-off speed below which such mechanisms as relaxation length are modified.

Most of the tire behavior is determined by table lookup. This screen has links for the tabular data to predict longitudinal force, lateral force, and aligning moment.

When the tire is subjected to combined cornering and braking, the shear force (and aligning moment) is limited by friction. The Pajecka "Magic Formula" is used to combine the cornering and braking effects. Also, Pajecka's method of similarity is used to account for road surface friction that is different from the frictional conditions in the laboratory test equipment. Details of the tire model are presented in Appendix G.

All properties are specified for one tire. If a wheel has dual tires, the simulation program computes the tire forces separately for each.

A single library contains tire data for all VDM RoAD models (cars and trucks).

User Settings

- ① **Longitudinal Force** link. The linked data set contains longitudinal tire force data in tabular form.
- ② **Lateral Force** link. The linked data set contains lateral tire force data in tabular form.
- ③ **Aligning Moment** link. The linked data set contains tire aligning moment data in tabular form.
- ④ Tire vertical stiffness (keyword = `KT (IUNIT, IAXLE)`).
 Note: For the car model, the IUNIT index is not used in the indexed parameters.
- ⑤ Tire relaxation length (keyword = `LRELAX (IUNIT, IAXLE)`). This is about one-third the distance that the tire must roll before side force due to slip angle builds up to its full value. It is basically a spatial version of a time constant.
- ⑥ Cutoff speed for tire relaxation equations (keyword = `V_CUTOFF (IUNIT, IAXLE)`). The concept of relaxation length is valid for a range of speeds, but leads to strange behavior as speeds approach zero because the time constant associated with the dynamic lag goes to infinity. The time constant associated with relaxation is frozen when the speed drops below this value.
- ⑦ Optional field for use with custom tire models. It is not used for the standard VDM RoAD models.
- ⑧ Optional links for use with custom tire models. They are not used for the standard VDM RoAD models.
- ⑨ Camber thrust coefficient (keyword = `KGAMMA (IAXLE)`). This coefficient is used only when the tire data are applied to the car model. It defines a side force due to inclination angle. The effect is small, but can affect the accuracy of the simulations in some cases.

Location in VDM RoAD

Truck Models

VDM RoAD Startup

Runs

Vehicle: (loaded truck or combination vehicle)

Vehicle Unit: (empty truck, tractor, or trailer)

Tires: Table-Lookup Model

Car Model

VDM RoAD Startup

Runs

Cars

Tires: Table-Lookup Model

File Location

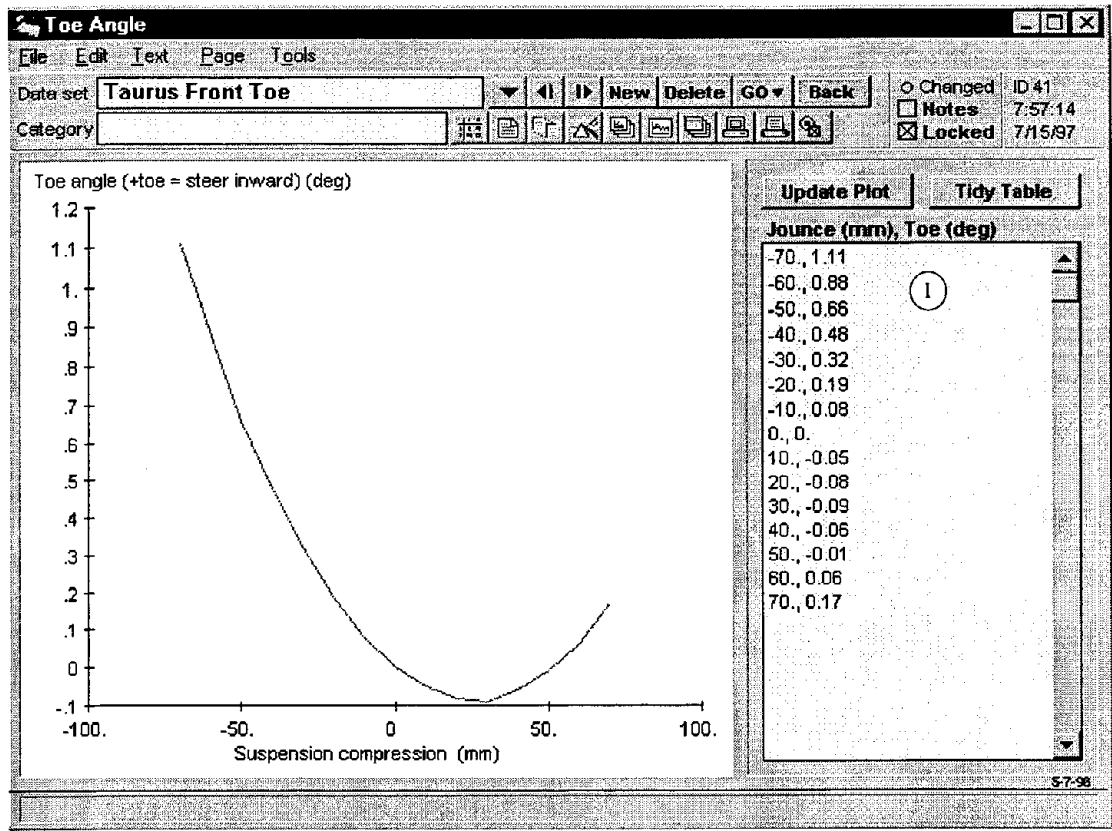
Vehicles\Tires\Tires.tbk

Toe Angle

This screen has the table lookup, which defines the wheel toe angle for an automotive independent suspension as a function of suspension jounce (compression). This library is not used for the truck models in VDM RoAD.

Discussion

The VDM RoAD car model includes the toe angle change as a function of suspension travel. The toe, which contributes to the wheel steer angle, is due to the 3D kinematics of the suspension and steering system linkages. Engineers usually try to design the suspension to minimize the change in toe. It is usually more significant for the front wheels than for the rear.



User Settings

Note: User settings that are common for all tabular data screens are described in Chapter 7, in the section **Tabular Data**.

- ① Table of values of wheel camber angle as a function of suspension compression (keyword = TOE_TABLE). Suspension compression is in mm and the toe angle is in degrees.

The sign convention for suspension travel (labeled *Suspension compression* on the screen) is that a positive value puts the suspension in jounce. The sign convention in VDM RoAD is that toe-in is described as positive and toe-out is described as negative.

This table uses constant-gradient extrapolation. For values of compression that are less than the range covered in the table, the first point is used to extrapolate the first gradient backward. The gradient is determined from the first two values of compression and toe. For values of compression larger than the range covered, the last point is used to extrapolate forward using the gradient between the last two points. The table needs at least two lines (for the simplest case of a constant ratio between toe and compression).

Location in VDM RoAD

VDM RoAD Startup

Runs

Cars

Suspensions
Toe Angle

File Location

Vehicles\Parts\Toe\Toe.tbk

VDM RoAD Startup

This screen appears immediately after VDM RoAD has been started. Normally, to start a simulation you just click the Start button and go directly to the **Runs** screen.

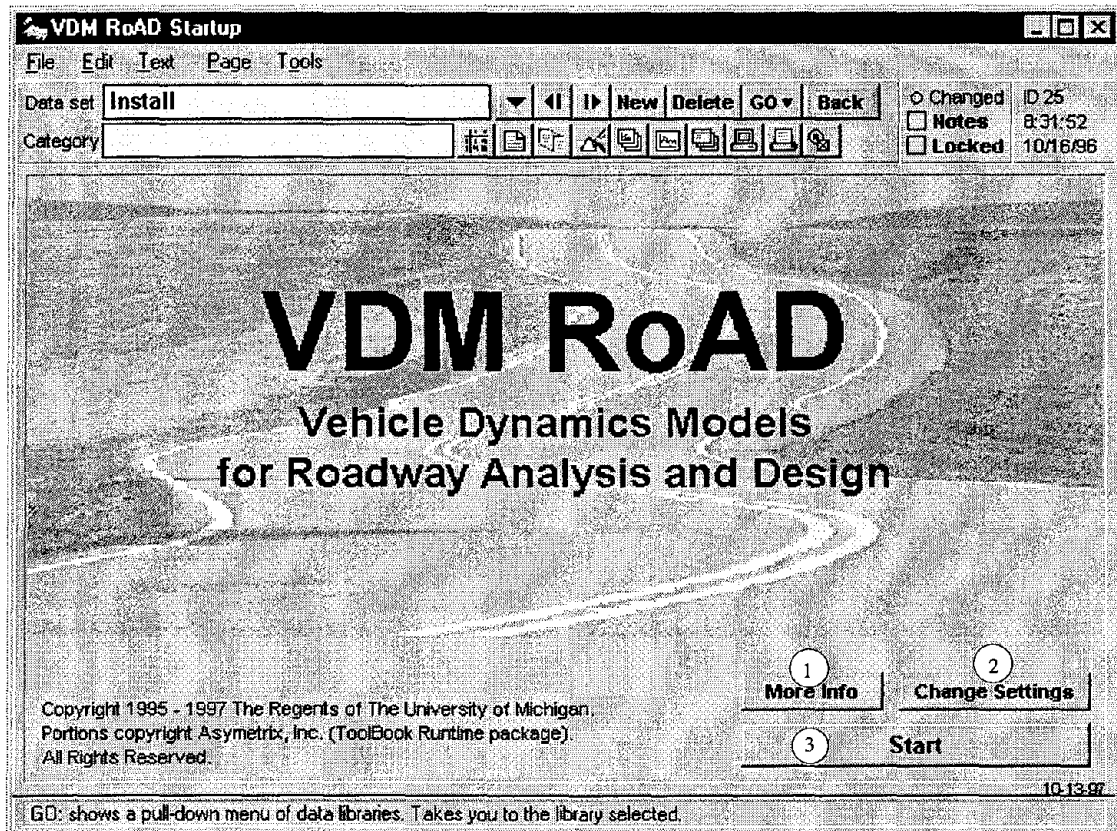


Figure 8.7. Startup screen (simple view).

Note: The VDM RoAD Startup Screen has two possible appearances. Figure 8.7 shows the simplest display. Figure 8.8 shows the other appearance.

Discussion

This screen provides a starting point for the software. It also includes settings that help configure VDM RoAD for your particular installation.

There are at least three reasons why you might someday want to modify the VDM RoAD settings or make a new Startup data set with different settings:

1. If you move the VDM RoAD software to a new hard disk volume or folder, all of the text files used to communicate between VDM RoAD data screens and the VDM RoAD solver programs must be updated. Otherwise the solver programs will give error messages about files not being found.
2. If you make many data sets in a library (e.g., the **Runs** library), you may want to create a new folder and create a second library. If you add any folders to VDM RoAD, you should install them in the **Startup** library.
3. You may want to change the GO menu, to show fewer libraries.

On the other hand, you may never need to change the settings contained in this library.

User Settings (Simple Display)

- ① **More Info** button. Click this button to go to screens with information about VDM RoAD that involve the version, copyright information, etc.
- ② **Change Settings** button. Click this button to show additional settings that can be modified. When clicked, the VDM RoAD pictures are replaced with the settings shown in Figure 8.8.
- ③ **Start** button. Click this button to go to the **Runs** library. This is the most common way to start using the software. The default is that clicking the button will take you to the most recently used data set in the **Runs** library. However, you can modify the function of this button as described below.

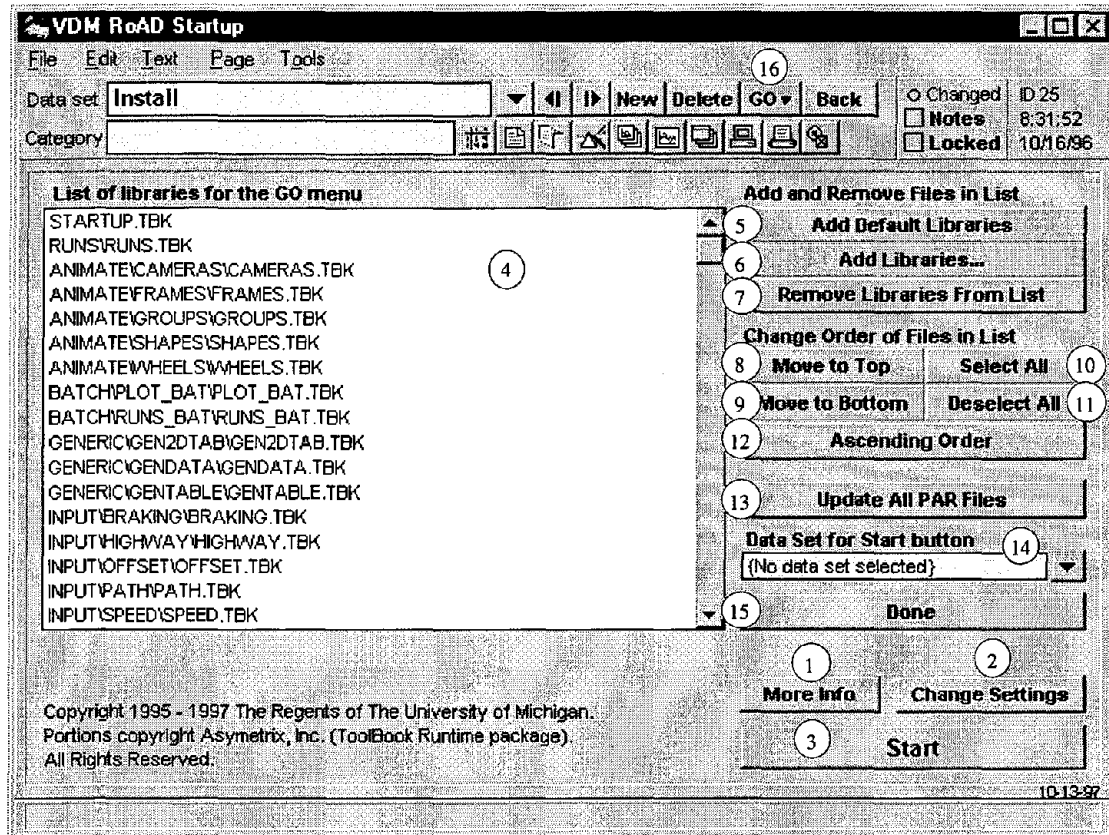


Figure 8.8. Startup screen after the Change Settings button has been clicked.

Additional User Settings and Controls

- ④ List of libraries for the GO menu. This list is also used for other functions within VDM RoAD. Basically, this list tells VDM RoAD what library files exist and where they are located.

The files are represented as pathnames relative to the VDM RoAD root directory. For example, if VDM RoAD is in `c:\VDM_RoAD`, then the partial pathname `RUNS\RUNS.TBK` means the full pathname for the file is `c:\VDM_RoAD\Runs\Runs.tbk`.

This list is used by VDM RoAD for two purposes:

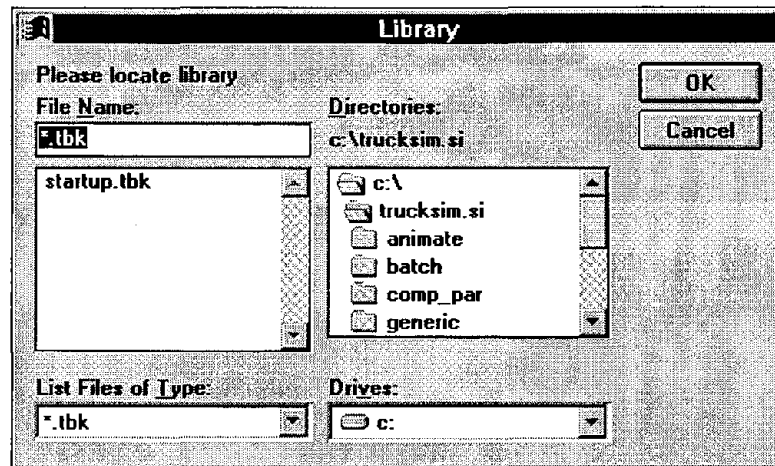
1. It defines which files are affected when text data files are updated if you click the **Update All PAR Files** button (13).
2. It determines which files are listed in the pull-down menu that appears when you press the **GO** button in the ribbon bar (16).

In both cases, files are included only if they are listed in the field **and** if they actually exist.

This list cannot be edited directly. Lines in it can be selected by clicking on them, and the contents are manipulated by using the adjacent buttons.

- ⑤ **Add Default Libraries** button. Click to add all of the library files included in the VDM RoAD installation. This button is provided as a way to undo damage and return to the status “as shipped.” If a default library file is already in the field ④, it is not added again.
- ⑥ **Add Libraries** button. Use this to add new libraries files to VDM RoAD. For example, if you have so many data sets in a library that it is unwieldy, you can duplicate the TBK file and its folder, then delete all but one data set from the copy. Give it a new name and add it to the system.

Click the button to bring up the Windows Find File browser dialog. TBK files selected with the browser are added to the list ④.



Note: You can add more than one file at a time. After you click OK, the dialog re-appears. When you are through adding files, click the **Cancel** button.

- ⑦ **Remove Libraries From List** button. Click to remove any selected lines from the list ④. The actual files are not affected by this—only the list is altered.
- ⑧ **Move to Top** button. Click to move any selected lines in the list to the top. For example, to make the **Runs** library the first item on the GO menu, select the line RUNS\RUNS.TBK in the list ④ and then click this button.
- ⑨ **Move to Bottom** button. Click to move any selected lines in the list to the bottom.
- ⑩ **Select All** button. Click to select all lines in the field ④. The contents of the field usually extend beyond the visible area, and can only be viewed by using the scroll bar. Clicking this button causes all lines to be selected, even if they are not visible.
- ⑪ **Deselect All** button. Click to deselect all lines in the field ④.
- ⑫ **Ascending Order** button. Click to alphabetically sort all lines in the field ④.

- ⑬ **Update All PAR Files** button. Click to process all of the VDM RoAD library files in the field ④, updating all text files. You should do this if you move VDM RoAD or rename any of the TBK files or folders.

Each data set in the VDM RoAD library has an associated text file with extension PAR that is used to communicate with the vehicle dynamics solver programs. These files have absolute pathnames in them. If any changes have been made to the file system (say, you moved VDM RoAD from drive C to drive D), all the old pathnames won't work.

The operation initiated when you click this button will probably take a couple of minutes to complete. The amount of time depends on the number of data sets in your VDM RoAD installation and the speed of your computer.

- ⑭ **Data Set for Start Button** link. This link defines what happens when you click the **Start** button ③. You can link to a library with no particular data set, or you can link to a particular data set within a library. If you do not choose a data set, then clicking the **Start** button takes you to the last data set visited in the library. This is the way VDM RoAD is set when initially installed. If you would like the **Start** button to always take you to a specific data set, link to it here.

You might change this link if you move the **Runs** library, or if you create a new **Runs** library in a different folder.

What if you have several **Runs** libraries? In that case, you can make several data sets in the **VDM RoAD Startup** library. Click the **New** button to make a new data set. Then, in the new data set, link to a different **Runs** library.

- ⑮ **Done** button. Click to hide all the user settings and return to the simpler view with the VDM RoAD logo, as shown in Figure 8.7.

Location in VDM RoAD

VDM RoAD Startup

File Location

Startup

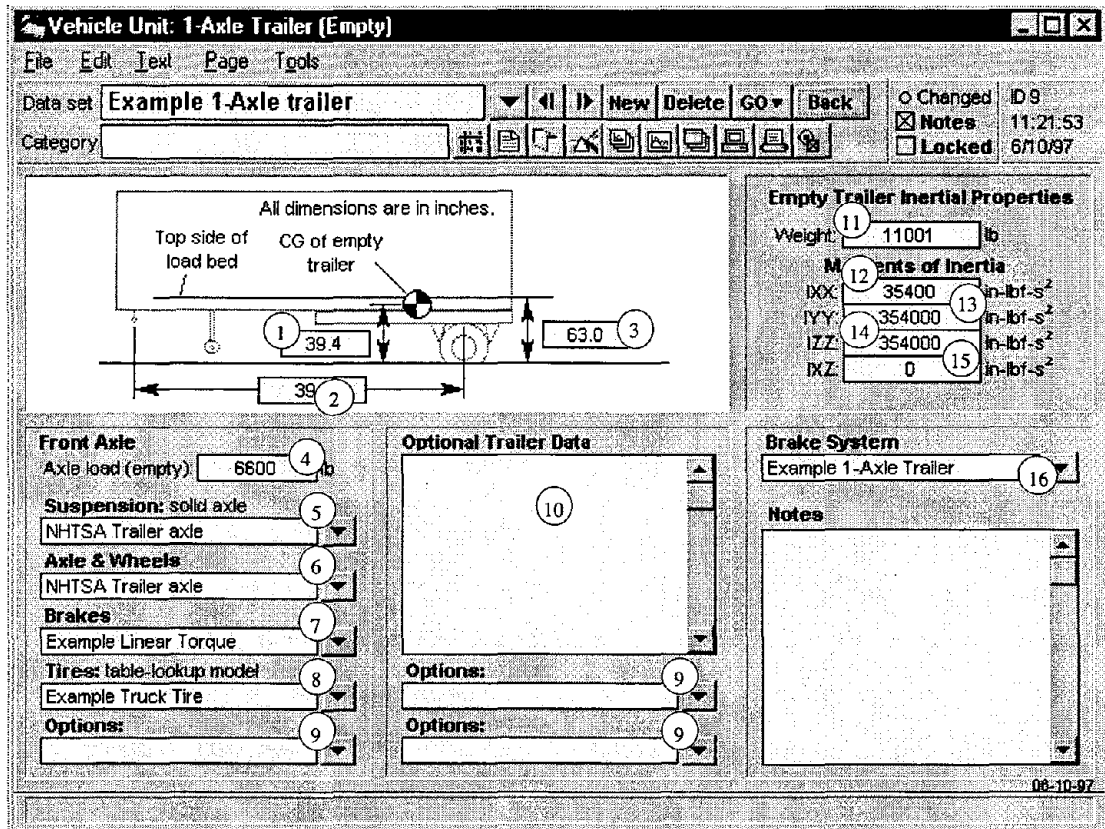
Vehicle Unit: 1-Axle Trailer (Empty)

Use this screen to define the basic properties of an empty single-axle trailer and links to its suspension system, axles, wheels, tires, and brakes.

Discussion

VDM RoAD allows you to connect a single-axle semitrailer to a tractor. This screen provides the means to define the relevant semitrailer characteristics as well as links to the other components and subsystems that are specified in other libraries.

Although the picture on the screen shows a van body, the parameter set can describe a flat bed or any configuration involving a single-axle trailer.



User Settings

- ① Height of the mass center above the ground (keyword = HCGTU (2)). This refers to the mass center of the total unladen trailer (sprung plus unsprung masses).
- ② Wheelbase (keyword = LWB (2)). Distance from the fifth wheel pin to the center of the axle.
- ③ Load bed height above the ground (keyword = HTLB (2)). This is used by the VDM RoAD simulation programs to determine the height of the mass center of the load.
- ④ Axle load (keyword = MUL (IUNIT, IAXLE)). Weight that would be measured if the axle were placed on a weigh scale for the empty trailer.
- ⑤ **Suspension** link. The linked data set defines the suspension dimensions and components.
- ⑥ **Axles & wheels** link. The linked data set defines axle mass properties and dimensions of the axle and wheels.
- ⑦ **Brakes** link. The linked data set defines the mechanical properties of the brakes.

- ⑧ **Tires** link. The linked data set defines the tire properties.
- ⑨ Optional data links. These links are provided for custom models. They are not used for standard VDM RoAD models.
- ⑩ Optional trailer data. This field is provided for custom models. It is not used for standard VDM RoAD models.
- ⑪ Empty trailer mass (keyword = MTRAIL(2)). This includes sprung plus unsprung masses.
- ⑫ Roll moment of inertia of the total unladen semitrailer (keyword = IXXTU(2)). The moment is taken about the mass center, normal to an X axis (longitudinal) that is parallel to the ground when the vehicle is at rest on a flat level surface.
- ⑬ Pitch moment of inertia of the total unladen semitrailer (keyword = IYYTU(2)). The moment is taken about the mass center, normal to a Y axis (lateral) that is parallel to the ground when the vehicle is at rest on a flat level surface.
- ⑭ Yaw moment of inertia of the total unladen semitrailer (keyword = IZZTU(2)). The moment is taken about the mass center, normal to a Z axis (vertical) that is parallel to the gravity vector when the vehicle is at rest on a flat level surface.
- ⑮ Roll-yaw product of inertia of the total unladen semitrailer (keyword = IXZTU(2)). The product is taken about the mass center, and is defined as the negative of the volume integral over $X*Z$ dV ($I_{xz} = -\text{Int}(\text{RHO}*X*Z*\text{DV})$). The product is positive when the principal X axis tilts down (looking forward).
- ⑯ **Brake System** link. The linked data set specifies brake transport delays and ABS parameters.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: Tractor-Semitrailer Combination

Vehicle Unit: 1-Axle Trailer (Empty)

File Location

Vehicles\1a_trail\1a_trail.tbk

Vehicle Unit: 2-Axle Tractor

Use this screen to define the basic properties of a two-axle tractor with no trailer attached and links to its suspension system, steering system, axles, wheels, tires, and brakes.

Vehicle Unit: 2-Axle Tractor

File Edit Text Page Tools

Data set: **Example 2-Axle Tractor** [New] [Delete] [GO] [Back] Changed ID:20

Category: [] Notes 11:43:05

Locked 4/22/97

CG of total tractor

All dimensions are in inches.

39.4 (1)

197 (2)

63.0 (3)

Rotational Inertias

IXX: 37000 (4)

IYY: 180000 (5)

IZZ: 190000 (6)

IXZ: 0 (7)

Units: in-lbf-s²

For total (including axles) empty vehicle.

Steering System (8)

Example Steering System

Brake System (9)

Example 4x2 Tractor

Hitch (10)

NHTSA 5th Wheel

Options: (11)

Front Axle (12)

Load, no trailer: 11000

Suspension: solid axle (13)

NHTSA Front axle

Axle & Wheels (14)

NHTSA Front axle

Brakes (15)

Example Linear Torque

Tires: table-lookup model (16)

Example Truck Tire

Options: (17)

Rear Axle (12)

Load, no trailer: 6600

Suspension: solid axle (13)

NHTSA Drive axle

Axle & Wheels (14)

NHTSA Drive axle

Brakes (15)

Example Linear Torque

Tires: table-lookup model (16)

Example Truck Tire

Options: (17)

Notes

06-10-97

This data set is locked. Unlock data set to view the context sensitive help in this status bar.

Discussion

VDM RoAD allows you to define a two-axle tractor to be used as the tow unit for a semitrailer. This screen provides the means to define the relevant tractor characteristics as well as links to the other components and subsystems that are specified in other libraries.

Although the picture on the screen shows a conventional body, the parameter set can describe a cab-over-engine or any configuration involving a two-axle vehicle.

This data set is normally used to make up a combination vehicle. In this case, the data set that links to it is a tractor-semitrailer combination data set. However, you can also link to it from a two-axle vehicle data set, to simulate the tractor in its bobtail configuration.

User Settings

- ① Height of the mass center above the ground (keyword = HCGTU (1)). This refers to the mass center of the total bobtail tractor (sprung plus unsprung masses).
- ② Wheelbase (keyword = LWB (1)). Distance from the center of the front axle to the center of the rear axle.
- ③ Hitch height above the ground (keyword = HH (1)). The point of interest is the pitch pivot of the hitch.

- ④ Roll moment of inertia of the total unladen tractor (keyword = `IXXTU(1)`). The moment is taken about the mass center, normal to an X axis (longitudinal) that is parallel to the ground when the vehicle is at rest on a flat level surface.
- ⑤ Pitch moment of inertia of the total unladen tractor (keyword = `IYYTU(1)`). The moment is taken about the mass center, normal to a Y axis (lateral) that is parallel to the ground when the vehicle is at rest on a flat level surface.
- ⑥ Yaw moment of inertia of the total unladen tractor (keyword = `IZZTU(1)`). The moment is taken about the mass center, normal to a Z axis (vertical) that is parallel to the gravity vector when the vehicle is at rest on a flat level surface.
- ⑦ Roll-yaw product of inertia of the total unladen tractor (keyword = `IXZTU(1)`). The product is taken about the mass center, and is defined as the negative of the volume integral over $X*Z$ dV ($I_{xz} = -\text{Int}(\text{RHO}*X*Z*DV)$). The product is positive when the principal X axis tilts down (looking forward).
- ⑧ **Steering System** link. The linked data set contains the steering system properties.
- ⑨ **Brake System** link. The linked data set contains brake transport delays and ABS parameters.
- ⑩ **Hitch** link. The linked data set contains hitch stiffness coefficients.
- ⑪ **Optional data** link. This link is provided for custom models. It is not used for standard VDM RoAD models.
- ⑫ **Axle load** (keyword = `MUL (IUNIT, IAXLE)`). Weight that would be measured if the axle were placed on a weigh scale for the tractor without an attached trailer.
- ⑬ **Suspension** link. The linked data set defines the suspension dimensions and components for the indicated axle.
- ⑭ **Axles & wheels** link. The linked data set defines axle mass properties and dimensions of the axle and wheels.
- ⑮ **Brakes** link. The linked data set defines the mechanical properties of the brakes on the indicated axle.
- ⑯ **Tires** link. The linked data set defines the tire properties for the indicated axle.
- ⑰ **Optional data** links. These links are provided for custom models. They are not used for standard VDM RoAD models.

Location in VDM RoAD

As Part of Combination Vehicle

VDM RoAD Startup

Runs

Vehicle: Tractor-Semitrailer Combination

Vehicle Unit: 2-Axle Tractor

As Bobtail Tractor

VDM RoAD Startup

Runs

Vehicle: 2-Axle Truck or Bus (Loaded)

Vehicle Unit: 2-Axle Tractor

File Location

Vehicles\2a_tract\2a_tract.tbk

Vehicle Unit: 2-Axle Trailer (Empty)

Use this screen to define the basic properties of an empty two-axle and links to its suspension system, axles, wheels, tires, and brakes.

Discussion

VDM RoAD allows you to connect a two-axle semitrailer (tandem rear axle set) to a tractor. This screen provides the means to define the relevant semitrailer characteristics as well as links to the components and subsystems that are specified in other libraries.

Although the picture on the screen shows a van body, the parameter set can describe a flat bed or any configuration involving a two-axle trailer.

User Settings

- ① Load bed height above the ground (keyword = HTLB (2)). This is used by the VDM RoAD simulation programs to determine the height of the mass center of the load.
- ② Wheelbase (keyword = LWB (2)). Distance from the fifth wheel pin to the centerline of the axles.
- ③ Height of the mass center above the ground (keyword = HCGTU (2)). This refers to the mass center of the total unladen trailer (sprung plus unsprung masses).
- ④ Axle spacing (keyword = LTAND (2)). Longitudinal distance between the centerlines of the axles.
- ⑤ Empty trailer mass (keyword = MTRAIL (2)). This includes sprung plus unsprung masses.
- ⑥ Roll moment of inertia of the total unladen semitrailer (keyword = IXXTU (2)). The moment is taken about the mass center, normal to an X axis (longitudinal) that is parallel to the ground when the vehicle is at rest on a flat level surface.
- ⑦ Pitch moment of inertia of the total unladen semitrailer (keyword = IYYTU (2)). The moment is taken about the mass center, normal to a Y axis (lateral) that is parallel to the ground when the vehicle is at rest on a flat level surface.
- ⑧ Yaw moment of inertia of the total unladen semitrailer (keyword = IZZTU (2)). The moment is taken about the mass center, normal to a Z axis (vertical) that is parallel to the gravity vector when the vehicle is at rest on a flat level surface.
- ⑨ Roll-yaw product of inertia of the total unladen semitrailer (keyword = IXZTU (2)). The product is taken about the mass center, and is defined as the negative of the volume integral over $X*Z$ dV ($I_{xz} = -\text{Int}(\text{RHO}*X*Z*DV)$). The product is positive when the principal X axis tilts down (looking forward).
- ⑩ Axle load (keyword = MUL (IUNIT, IAXLE)). Weight that would be measured if the indicated axle were placed on a weigh scale for the empty trailer.
- ⑪ **Suspension** link. The linked data set defines the suspension dimensions and components.
- ⑫ **Axles & wheels** link. The linked data set defines axle mass properties and dimensions of the axle and wheels.
- ⑬ **Brakes** link. The linked data set defines the mechanical properties of the brakes on the indicated axle.

- ⑭ **Tires** link. The linked data set defines the tire properties for the indicated axle.
- ⑮ Optional data link. These links are provided for custom models. They are not used for standard VDM RoAD models.
- ⑯ **Brake System** link. The linked data set specifies brake transport delays and ABS parameters.
- ⑰ Optional trailer data. This field is provided for custom models. It is not used for standard VDM RoAD models.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: Tractor-Semitrailer Combination

Vehicle Unit: 2-Axle Trailer (Empty)

File Location

Vehicles\2a_trail\2a_trail.tbk

Vehicle Unit: 2-Axle Truck or Bus (Empty)

Use this screen to define the basic properties of an empty two-axle truck and links to its suspension system, steering system, axles, wheels, tires, and brakes.

Discussion

VDM RoAD allows you to specify a two-axle truck. Use this screen to define the relevant truck characteristics as well as links to the other components and subsystems that are specified in other libraries.

Although the picture on the screen shows a conventional truck, the parameter set can describe a bus, a bobtail tractor, or other vehicles with two axles.

Vehicle Unit: 2-Axle Truck or Bus (Empty)

File Edit Text Page Tools

Data set: **Example 2-Axle Truck** | New | Delete | GO | Back | Changed ID 21 | Notes 11:16:09 | Locked 4/23/97

Category: []

CG of empty truck

All dimensions are in inches.

Top side of load bed

63.0

39.4

236

Rotational Inertia

IXX: 35400 (4)

IYY: 442500 (5)

IZZ: 442500 (6)

IXZ: 0 (7)

Units: in-lb-s²

For total (including axles) empty vehicle.

Steering System (8)

Example Steering System

Brake System (9)

Example 6x4 Tractor

Options: (10)

{No data set selected}

Options: (10)

Front Axle (11)

Load, empty: 8800 (10)

Suspension: solid axle (12)

NHTSA Front axle (12)

Axle & Wheels (13)

NHTSA Front axle (13)

Brakes (14)

Example Linear Torque (14)

Tires: table-lookup model (15)

Example Truck Tire (15)

Options: (10)

Second Axle (11)

Load, empty: 4400 (10)

Suspension: solid axle (12)

NHTSA Drive axle (12)

Axle & Wheels (13)

NHTSA Drive axle (13)

Brakes (14)

Example Linear Torque (14)

Tires: table-lookup model (15)

Example Truck Tire (15)

Options: (10)

Notes

06-10-97

User Settings

- ① Load Bed Height (keyword = HTLB (1)). Vertical distance from the ground to the load bed on the empty truck.
- ② Height of the mass center above the ground (keyword = HCGTU (1)). This refers to the mass center of the total unladen vehicle (sprung plus unsprung masses).
- ③ Wheelbase (keyword = LWB (1)). Distance from the center of the front axle to the center of the rear axle.
- ④ Roll moment of inertia of the total unladen vehicle (keyword = IXXTU (1)). The moment is taken about the mass center, normal to an X axis (longitudinal) that is parallel to the ground when the vehicle is at rest on a flat level surface.
- ⑤ Pitch moment of inertia of the total unladen vehicle (keyword = IYYTU (1)). The moment is taken about the mass center, normal to a Y axis (lateral) that is parallel to the ground when the vehicle is at rest on a flat level surface.
- ⑥ Yaw moment of inertia of the total unladen vehicle (keyword = IZZTU (1)). The moment is taken about the mass center, normal to a Z axis (vertical) that is parallel to the gravity vector when the vehicle is at rest on a flat level surface.

- ⑦ Roll-yaw product of inertia of the total unladen vehicle (keyword = IXZTU (1)). The product is taken about the mass center, and is defined as the negative of the volume integral over X*Z dV ($I_{xz} = -\text{Int}(\text{RHO} * X * Z * DV)$). The product is positive when the principal X axis tilts down (looking forward).
- ⑧ **Steering System** link. Link to the screen that sets steering system properties.
- ⑨ **Brake System** link. The linked data set specifies brake transport delays and ABS parameters.
- ⑩ Optional data links. These links are provided for custom models. They are not used for standard VDM RoAD models.
- ⑪ Axle load (keyword = MUL (IUNIT, IAXLE)). Weight that would be measured if the indicated axle were placed on a weigh scale for the empty vehicle.
- ⑫ **Suspension** link. The linked data set defines the suspension dimensions and components.
- ⑬ **Axles & Wheels** link. The linked data set defines axle mass properties and dimensions of the axle and wheels.
- ⑭ **Brakes** link. The linked data set defines the mechanical properties of the brakes on the indicated axle.
- ⑮ **Tires** link. The linked data set defines the tire properties for the indicated axle.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: 2-Axle Truck or Bus (Loaded)

Vehicle Unit: 2-Axle Truck or Bus (Empty)

File Location

Vehicles\2a_truck\2a_empty\2a_empty.trk

Vehicle Unit: 3-Axle Tractor

Use this screen to define the basic properties of a three-axle tractor (with no trailer) and links to its suspension system, steering system, axles, wheels, tires, and brakes.

Discussion

VDM RoAD allows you to define a three-axle tractor (front steer axle and a tandem rear axle set) to be used as the tow unit for a semitrailer. Use this screen to define the relevant tractor characteristics as well as links to the other components and subsystems that are specified in other libraries.

Although the picture on the screen shows a conventional body, the parameter set can describe a cab-over-engine or any configuration involving a three-axle vehicle.

Vehicle Unit: 3-Axle Tractor

File Edit Text Page Tools

Data set: **3.Axle tractor** [New] [Delete] [GO] [Back] Changed ID:21

Category: [] Notes 11:40:41

[] Locked 4/22/97

CG of total tractor

All dimensions are in inches.

1: 34.33, 2: 188.54, 3: 43.58, 4: 52.2

Rotational Inertias

IXX: 81029 (5)

IYY: 382910 (6)

IZZ: 373280 (7)

IXZ: 0 (8)

Units: in-lbf-s²

For total (including axles) empty vehicle.

Steering System (9)

Example Steering System

Brake System (10)

Example 6x4 Tractor

Hitch (11)

NHTSA 5th Wheel

Options: (12)

Front Axle (13)

Load, no trailer: 11074 lb

Suspension: solid axle (14)

NHTSA Front axle

Axle & Wheels (15)

NHTSA Front axle

Brakes (16)

Example Linear Torque

Tires: table-lookup model (17)

Example Truck Tire

Options: (18)

Second Axle (13)

Load, no trailer: 4193 lb

Suspension: solid axle (14)

NHTSA Drive axle

Axle & Wheels (15)

NHTSA Drive axle

Brakes (16)

Example Linear Torque

Tires: table-lookup model (17)

Example Truck Tire

Options: (18)

Third Axle (13)

Load, no trailer: 4193 lb

Suspension: solid axle (14)

NHTSA Drive axle

Axle & Wheels (15)

NHTSA Drive axle

Brakes (16)

Example Linear Torque

Tires: table-lookup model (17)

Example Truck Tire

Options: (18)

08-10-97

User Settings

- ① Height of the mass center above the ground (keyword = HCGTU (1)). This refers to the mass center of the total bobtail tractor (sprung plus unsprung masses).
- ② Wheelbase (keyword = LWB (1)). Distance from the center of the front axle to the center of the rear axle.
- ③ Hitch height above the ground (keyword = HH (1)). The point of interest is the hitch pitch pivot.
- ④ Axle spacing (keyword = LTAND (1)). Longitudinal distance between the centers of the rear axles.
- ⑤ Roll moment of inertia of the total unladen tractor (keyword = IXXTU (1)). The moment is taken about the mass center, normal to an X axis (longitudinal) that is parallel to the ground when the vehicle is at rest on a flat level surface.
- ⑥ Pitch moment of inertia of the total unladen tractor (keyword = IYYTU (1)). The moment is taken about the mass center, normal to a Y axis (lateral) that is parallel to the ground when the vehicle is at rest on a flat level surface.

- ⑦ Yaw moment of inertia of the total unladen tractor (keyword = IZZTU(1)). The moment is taken about the mass center, normal to a Z axis (vertical) that is parallel to the gravity vector when the vehicle is at rest on a flat level surface.
- ⑧ Roll-yaw product of inertia of the total unladen tractor (keyword = IXZTU(1)). The product is taken about the mass center, and is defined as the negative of the volume integral over $X*Z$ dV ($I_{xz} = -\text{Int}(\text{RHO}*X*Z*DV)$). The product is positive when the principal X axis tilts down (looking forward).
- ⑨ **Steering System** link. The linked data set contains the steering system properties
- ⑩ **Brake System** link. The linked data set contains brake transport delays and ABS parameters.
- ⑪ **Hitch** link. The linked data set contains hitch stiffness coefficients.
- ⑫ Optional data link. This link is provided for custom models. It is not used for standard VDM RoAD models.
- ⑬ Axle load (keyword = MUL (IUNIT, IAXLE)). Weight that would be measured if the indicated axle were placed on a weigh scale for the tractor without an attached trailer.
- ⑭ **Suspension** link. The linked data set defines the suspension dimensions and components.
- ⑮ **Axles & wheels** link. The linked data set defines axle mass properties and dimensions of the axle and wheels.
- ⑯ **Brakes** link. The linked data set defines the mechanical properties of the brakes on the indicated axle.
- ⑰ **Tires** link. The linked data set defines the tire properties for the indicated axle.
- ⑱ Optional data link. These links are provided for custom models. They are not used for standard VDM RoAD models.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: Tractor-Semitrailer Combination

Vehicle Unit: 3-Axle Tractor

File Location

Vehicles\3a_tract\3a_tract.trk

Vehicle: 2-Axle Truck or Bus (Loaded)

Use this screen to define the basic properties of a loaded two-axle vehicle and link to the data set for the empty vehicle.

Discussion

This library defines a two-axle loaded vehicle. Although the picture is of a truck, the vehicle could also be a bus or bobtail tractor.

The screen has links for an empty vehicle, load dimensions, and a reference point. It also has fields for the axle loads. Added together, the axle loads provide the total vehicle weight. The VDM RoAD solver programs subtract the empty vehicle weight from the total weight to determine the cargo weight. The cargo weight is then used, along with vehicle dimensions, to modify the mass center location for the entire vehicle and also its moments of inertia.

If you want to simulate the same vehicle under different loading conditions, you need only to modify the axle loads on this screen.

The screen is designed to link to an unloaded two-axle truck/bus data set. However, you can also link to a two-axle tractor data set to simulate a bobtail tractor.

Vehicle: 2-Axle Truck or Bus (Loaded)

File Edit Text Page Tools

Data set: Example 2-Axle Truck

Category:

Weight Distribution: Axle Loads (lb)

Axle 1	Axle 2
11000	17600

Empty 2-Axle Vehicle: Example 2-Axle Truck

Reference Point: Reference origin

Cargo Dimensions: Truck Load

Options: (two dropdowns)

Misc. (Custom) Vehicle Data

Notes

08-10-97

User Settings

- ① Axle load (keyword = MUL (IUNIT, IAXLE)). Weight that would be measured if the indicated axle were placed on a weigh scale for the loaded vehicle.

- ② Empty two-axle vehicle. Link to the screen where the unloaded vehicle is defined. This is normally a two-axle vehicle unit. However, you can also link to a two-axle tractor library to simulate a bobtail tractor.

Notes: In order to link to a two-axle tractor, you have to change the library. To do this:

1. Press the button next to the link to display the menu.
2. Select the menu item **Pick Library**. If the menu item is dimmed, check the preferences button in the ribbon bar, and make sure the **Advanced** mode is enabled.
3. Locate and select the library file `Vehicles\2a_tract\2a_tract.tbk`. VDM RoAD will warn you that the library is not of the expected type (a two-axle truck). Click the button **Use Anyway** to make the link.

If you link to a two-axle tractor, you should set the axle loads on this screen to match the axle loads for the tractor.

- ③ **Reference Point** link. The linked data set contains the coordinates of an arbitrary reference point, fixed in the sprung mass, at which motion variables will be calculated.
- ④ **Cargo Dimensions** link. The linked data set contains the cargo (load) dimensions.
- ⑤ Optional data links. These links are provided for custom models. They are not used for standard VDM RoAD models.
- ⑥ Optional data. This field is provided for custom models. It is not used for standard VDM RoAD models.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: 2-Axle Truck or Bus (Loaded)

File Location

`Vehicles\2a_truck\2a_truck.trk`

Vehicle: Tractor-Semitrailer Combination

Use this screen to define the basic properties of a loaded tractor-semitrailer combination and link to data sets for the empty tractor and trailer.

Discussion

This library screen has links for a tractor, a semitrailer, cargo dimensions, and reference points. It also has fields for the axle loads. Added together, the axle loads provide the

total vehicle weight. The VDM RoAD solver programs subtract the empty vehicle weight from the total weight to determine the cargo weight. The cargo weight is then used, along with vehicle dimensions, to modify the mass center location for the trailer and the hitch location.

Although the picture is for a five-axle vehicle, the data set can describe vehicles with any combination of two- and three-axle tractors with one- or two-axle trailers.

If you want to simulate the same vehicle under different loading conditions, you need only to modify the axle loads on this screen.

Weight Distribution: Axle Loads (lb)				
Tractor axle 1	Tractor axle 2	Tractor axle 3	Trailer axle 1	Trailer axle 2
12024	17042	17042	17042	17042

User Settings

- ① Axle load (keyword = MUL (IUNIT, IAXLE)). Weight that would be measured if the indicated axle were placed on a weigh scale.

WARNING The fields for tractor axle 3 and trailer axle 2 must be left blank if those axles do not exist in the vehicle units selected in links ② and ⑤. For example, if you link to a two-axle tractor and put a value in the field for axle 3, the simulation results will be in error.

- ② Tractor link. The linked data set describes the properties of the empty tractor.

- ③ **Reference Point** link for the tractor. The linked data set contains the coordinates of an arbitrary reference point, fixed in the tractor sprung mass, at which motion variables will be calculated.
- ④ Optional data links. These links are provided for custom models. They are not used for standard VDM RoAD models.
- ⑤ **Trailer** link. The linked data set describes the properties of the empty trailer.
- ⑥ **Reference Point** link for the trailer. The linked data set contains the coordinates of an arbitrary reference point, fixed in the trailer sprung mass, at which motion variables will be calculated.
- ⑦ **Cargo Dimensions** link. The linked data set contains the cargo (load) dimensions.
- ⑧ Optional data. This field is provided for custom models. It is not used for standard VDM RoAD models.

Location in VDM RoAD

VDM RoAD Startup

Runs

Vehicle: Tractor-Semitrailer Combination

File Location

Vehicles\Combos\Combos.trk

Appendix A — ERD File Format

The ERD file format was developed within the Engineering Research Division (ERD) of the University of Michigan Transportation Research Institute (UMTRI) to facilitate automated plotting of simulation data, experimentally measured data, and data from various analysis programs.

A freely available plotter called EP (Engineering Plotter) has been developed for viewing data in ERD files. Versions of EP exist for Mac, Windows, and some UNIX platforms. The Windows version is called WinEP.

The animator program in VDM RoAD is also designed to work with ERD files.

An ERD file contains two independent sections, the header and data. The header contains only text, and the data section contains only numbers. The numbers can be written in either text or binary form. The text form is convenient for viewing and editing data with a word processor, whereas the binary form provides more efficient access for automatic processing. If the data section is in text format, then both the header and the data are kept in a single file. However, if the data are in binary format, two files are used. The header is in an ordinary text file with the extension ERD, and the data are in a file with the name of the header file and the extension BIN. For example, if the header file is named `Out.erd` the name `Out.bin` must be used for the data file. Both files must lie in the same folder.

The Header

The ERD file header consists of a series of conventional text lines that are human readable. These lines contain the information used by post-processing tools to read the numerical data.

Required Lines

As a minimum, the header contains three lines of text. The first line identifies the file as following the ERD format. The second line describes the way that the numerical data are stored in the data section of the file. The third required line is an END statement that indicates the end of the header portion. Any number of optional lines can be included between line #2 and the END line. Table A.1 summarizes the lines in an ERD file, and describes the parameters used in line #2 to describe the numerical data.

The second line of the file shown in Listing A.1 shows that the file contains data for two channels, with 529 samples/channel, stored as one binary record consisting of 4232 bytes, that the data storage format is type 1 (4-byte binary), that the interval between samples is 1.00, and that the status of the auxiliary numbers is -1.

Table A.1. Summary of records in an ERD file header.

Line No.	Description
1	ERDFILEV2 . 00 — identifies file as having ERD format
2	<p>NCHAN, NSAMP, NRECS, NBYTES, KEYNUM, STEP, KEYOPT — use commas to separate numbers</p> <p>NCHAN [integer] = Number of data channels</p> <p>NSAMP [integer] = Number of samples for each channel. The total number of sampled values in the data portion of the file is $NCHAN \times NSAMP$. (If unknown, use -1.)</p> <p>NRECS [integer] = Number of records of data. (record \approx line) Ignored for text data (KEYNUM = 5.) (If unknown, use -1.)</p> <p>NBYTES [integer]</p> <p><i>binary data:</i> Number of bytes per record. If KEYNUM=0,1, or 5, this should be chosen such that each record begins with channel 1: that is, $NBYTES = K \times NCHAN \times B$, where K is an integer and B is the number of bytes/number (B=2 for integer, B=4 for floating-point). If KEYNUM=10,11, or 15, this should be $NSAMP \times B$.</p> <p><i>text data:</i> Number of samples per record. Thus each record contains $NBYTES \times NCHAN$ numbers (for KEYNUM=5).</p> <p>KEYNUM[integer] Indicates how the data are stored.</p> <p>0, 10 = 2-byte integer (binary),</p> <p>1, 11 = 4-byte floating point (binary),</p> <p>5, 15 = Formatted floating-point (text). The format must be specified using the FORMAT keyword.</p> <p>For KEYNUM=0,1, and 5, the data are stored with all channels for the first sample together, then all channels for the second sample, etc.</p> <p>For KEYNUM=10,11, and 15, the data are stored with all samples for the first channel together, then all samples for the second channel, etc.</p> <p>STEP [real] = sample interval (e.g., time step)</p> <p>KEYOPT [integer] = auxiliary number used by some programs</p>
	<p>Optional records. Each record begins with an 8-character keyword, followed by information associated with that keyword. Table A.2 lists keywords that have been used to date.</p>
<i>last line</i>	END — indicates the end of the header

Listing A.1. Short Header for an ERD File with Binary Data.

```

ERDFILEV2.00
2, 529, 1, 4232, 1, 1.00000, -1,
TITLE 1993 RPUG Study, Dipstick, Section 1, Measurement 1
SHORTNAMLelev. RElev.
UNITSNAMft ft
XLABEL Distance
XUNITS ft
END

```

Listing A.2 shows a longer header for a file with its data in text form. Note that the data begin immediately after the END line of the header.

Listing A.2. Typical Header for an ERD File with Text Data.

```

ERDFILEV2.00
2, 529, 1, 4232, 1, 1.00000, -1,
TITLE 1993 RPUG Study, Dipstick, Section 1, Measurement 1
SHORTNAMLelev. RElev.
LONGNAMELeft Elevation Right Elevation
UNITSNAMft ft
GENNAME Profile Elevation Profile Elevation
XLABEL Distance
XUNITS ft
FORMAT (2G14.6)
PROFINSTDipstick
HISTORY Converted to ERD format at 23:46, Oct. 23, 1994
END
0.000000 0.000000
0.416667E-03 -0.141667E-02
0.416667E-03 0.583333E-03
0.666667E-03 0.916667E-03
0.133333E-02 0.133333E-02
0.750000E-03 -0.166667E-02
-0.300000E-02 -0.458333E-02
-0.558333E-02 -0.500000E-02
-0.625000E-02 -0.658333E-02
-0.775000E-02 -0.825000E-02

```

Optional Lines with Keywords

Optional lines in the header begin with an eight-character keyword that defines a particular type of data contained in the remainder of the line. Keywords are associated with one of five general data types: integers, floating point (real) numbers, 8-character names, 32-character names, and 80-character names. The number of data items is either one per file (e.g., TITLE of data set in the file), one per channel (e.g., a short name for each channel), an arbitrary number N (e.g., static axle loads for N axles), or repeatable (e.g., comments stored using the HISTORY keyword).

Table A.2 lists common keywords recognized by most post-processing tools. The use of some of these keywords is demonstrated in Listing A.1 and Listing A.2.

Table A.2. Keywords in ERD file.

keyword	Description	No. of Values	Variable Type
VERSION	Line 1 in header file.	1	char*32
LINE 2	Line 2 in header file. See Table A.2 for details.	7	int, real
&n	Continuation keyword, indicates that the previous line ended in column <i>n</i> and is continued in this line in column 9. Used to break long lines into multiple short lines.		
<i>(The following 6 lines are used by EP and are recommended for inclusion in all ERD files)</i>			
GENNAME	Generic names for variables, used for labeling Y axis when several variables are plotted on the same axis (e.g., force).	NCHAN	char*32
LONGNAME	Long names for channels.	NCHAN	char*32
SHORTNAM	Short names for channels.	NCHAN	char*8
TITLE	Title used for file.	1	char*80
UNITSNAM	Units of channels.	NCHAN	char*8
XUNITS	Units of independent variable (e.g., sec).	1	char*8
<i>(The following line is required for EP to create Channel 0, e.g., time)</i>			
XLABEL	Name of ind. variable in ERD file (e.g., time).	1	char*32
FORMAT	FORMAT statement for text data. Ex: (4F10.4)	1	char*32
GAIN	Gains for channels. (Default = 1.) Usually required for integer*2 data.	NCHAN	real
OFFSET	Offsets for channels. (default = 0.) Usually required for integer*2 data.	NCHAN	real
PROFINST	Instrument or model associated with data.	1	char*32
RIGIBODY	Body or part associated with each channel.	NCHAN	char*32
SPEEDMPH	Speed associated with data, in mi/h.	1	real
TESTID	Number used to identify a test.	1	real
XSTART	Starting value of ind. variable. At each sample <i>i</i> , the X value is: $X = (i-1) * STEP + XSTART$	1	real

Usually, names associated with a keyword are shorter than the space allowed. When several names are on the same line, the names are padded with blanks as needed so that following names begin at the correct column positions. For example, the header shown in Listing A.1 includes names of the units for each channel, as identified with the keyword UNITSNAM. The name of units for the first channel, ft, has only two characters. Thus, it

is followed by six spaces so that the name for the second channel, ft, begins in the correct column position.

The Data Section

The data section of the ERD file contains nothing but numbers, organized into columns and rows. The form in which the numbers are stored depends on the value of the KEYNUM parameter from line 2 of the header (see Table A.1). The total number of values that will appear in the data section is NCHAN x NSAMP. All of the numbers in the data portion are stored in the same format, and there can be no missing values.

Text Data

The text format can be used for transporting data in ERD files between different computers, and sometimes even for reading the same file with different programs on the same computer. It is also convenient when numbers are typed in manually, or when numbers are to be edited using a text editor. There are penalties for using text representations of numbers, however. First and foremost, the computer must work hard to translate the text numbers into binary form. It takes about 10 times longer to read a text file than a binary equivalent. A second penalty is that text files take up much more disk storage than binary files.

When data are stored in text form, the numbers are kept in the same file as the header, with the numbers beginning immediately after the header. The ERD file in Listing A.2 shows an example of numerical data in text form.

Another option is available when the numbers are always separated by delimiters such as spaces or commas. This occurs when the numbers are obtained by a commercial analysis program or when they are "captured" from another computer. The file of numbers can be made into an ERD file by inserting a three-line header at the beginning of the file.

If the header of the ERD file does not contain a line with the FORMAT keyword, it is the same as if the FORMAT is a blank. When this occurs for a text file, the file is assumed to contain numbers in free form. The only restriction on free format numbers is that adjacent numbers must be separated. For example, the following line is valid for representing five numbers 1.2, 3, 4, -.0201, 14.3: 1.2000 3 4 -2.01E-01 14.3

The following line is not, because the third and fourth numbers touch.

```
1.2000 3.0000 4.0000-2.01E-01 14.3000
```

Numbers may be separated by one or more spaces, the tab character, or a comma.

Binary Data

Reading and writing binary data are very efficient, because the computer does not need to perform any conversions or transformations as the data values are moved between the file and the computer memory. When a binary format is used, the data portion of an ERD file is a direct copy of a portion of the computer memory, corresponding to a two-

dimensional array having dimensions sized to the number of channels and the number of samples. As indicated in Table A.1, two forms of binary data are presently supported: 2-byte integer and 4-byte floating point. 2-byte integer data are typically obtained by data-acquisition systems. Each integer value is a sampled reading obtained from a digitizer during a test. For most engineering applications, data are stored (in the computer memory) in 4-byte floating point format, also known as single-precision floating point. The 4-byte floating point format is commonly used for data generated by computer. The maximum efficiency for data processing is usually obtained when the 4-byte floating point format is used.

The ERD file format is used on a variety of computer systems and for a variety of mass storage media. On some systems, binary data are stored in discrete records. A computer program reading such a file needs to know how many bytes each record contains, and how many records are in the file. Thus, the header contains these two parameters.

Disk files on workstations and desktop computers are not structured: a binary file is simply a continuous stream of bytes that continues to the end of the file. Thus, technically correct parameter values for the header could be one record, containing all of the bytes for the file. Also, there is a certain amount of overhead associated with reading a record. The time needed to read the data for a file is minimized if a single read operation is performed for the entire file.

On the other hand, if the file is large, the memory needed to read the entire file in one chunk may not be available with some programs. A second problem can occur if the true number of bytes in the file is less than the number as inferred by the parameters *nbytes* and *nrecs* (i.e., the total size of the file should be *NBYTES* x *NRECS*). The last "record" is not read, resulting in a loss of data. If the records are large, this loss could be significant. These problems are reduced if a value of *NBYTES* is specified such that it divides the data into *NRECS* records of smaller chunks of data.

Appendix B — Plotter Files and Keywords

When WinEP is run from VDM RoAD, it receives the information that describes a plot or set of plots from a text file. The passing of information between a VDM RoAD data screen and WinEP is automatic, and the details of the files used to transfer information do not have to be understood for normal use of VDM RoAD.

The RunWinEP Program

The current version of WinEP (4.1) has a minor conflict with the ToolBook program used to manage the VDM RoAD data base. In some Windows installations, WinEP goes into the background when launched directly from ToolBook. The problem is thought to be due to a bug in the way ToolBook passes control to a Windows program created with the Microsoft Studio compiler using the Microsoft Foundation Class (MFC).

As a work around, a separate program called RunWinEP.exe is used to launch WinEP. As far as ToolBook is concerned, the plotter is RunWinEP. RunWinEP performs just one function: it launches WinEP and then quits. (The RunWinEP program was created using different compiler settings and does not have conflicts with ToolBook.)

The RunWinEP program is hard-coded to run the file WinEP.exe from the same folder that it is in. Therefore, do not change the file name of the copy of WinEP that is used within VDM RoAD.

PLT Batch Control Files

Most plots made in VDM RoAD are generated when you click the **Plot** button on the **Runs** screen or the **Plot Setup: Batch** screen. When this occurs, the VDM RoAD library creates a text file with the extension PLT and sends that file to WinEP. The PLT file is similar to the PAR files used in VDM RoAD to transfer information from the data base to the simulation solver programs. It is a plain ASCII text file that contains keywords with parameters that tell WinEP how to make a plot.

Listing B.1 shows an example PLT file and Table B.1 summarizes the format.

Listing B.1. Example PLT batch plot file.

```

FILELIST
FILENAME C:\VDM ROAD.IN\RUNS\491.ERD
FILENAME C:\VDM ROAD.IN\RUNS\430.erd
PARSFILE C:\VDM ROAD.IN\PLOT\422.par
RUN Screen

FILENAME C:\VDM ROAD.IN\RUNS\491.ERD
FILENAME C:\VDM ROAD.IN\RUNS\430.erd
PARSFILE C:\VDM ROAD.IN\PLOT\428.par
RUN Screen

FILENAME C:\VDM ROAD.IN\RUNS\491.ERD
FILENAME C:\VDM ROAD.IN\RUNS\430.erd
PARSFILE C:\VDM ROAD.IN\PLOT\433.par
RUN Screen

FILENAME C:\VDM ROAD.IN\RUNS\491.ERD
FILENAME C:\VDM ROAD.IN\RUNS\430.erd
PARSFILE C:\VDM ROAD.IN\PLOT\439.par
RUN Screen
END

```

The lines in the PLT file are grouped into plot sets, separated by lines beginning with the RUN keyword. Lines within a plot set can be arranged in any order.

Table B.1. PLT batch plot file format.

Line	Contents	Instruction to Program
1	FILELIST	Identify this file batch plot file.
.	FILENAME <i>filename</i>	Open the file named <i>filename</i>
.	PARSFILE <i>filename</i>	Open the file named <i>filename</i> and read a single plot template
last line of plot set	RUN <i>dest</i>	Use the current information to create a plot and send it to <i>dest</i>
last line read	END	Stop reading data.
.	miscellaneous	ignore all lines after END.

Plot Template Files

A plot template is a list of variables to extract from the data files, along with formatting information. Templates are created automatically by the VDM RoAD library **Plot Setup: Single**. They are stored in the same PARSEFILE format used in VDM RoAD to transfer

information from the data base to the simulation solver programs, and have the extension PAR. Listing B.2 shows an example PARFILE with information to define a plot.

Listing B.2. Example plot template file.

```
PARSFILE
DATASET C:\VDM ROAD.IN\RUNS\455.erd Time      ,Ay
DATASET C:\VDM ROAD.IN\RUNS\455.erd Time      ,Ay_2

PARSFILE C:\VDM ROAD.IN\PLOT\FORMAT\1.par
PARSFILE C:\VDM ROAD.IN\PLOT\transfrm\63.par
PTITLE Ay -- lat. accel. of CG's
END
```

WinEP processes a template in two possible ways, depending on the circumstances in which the file is encountered. If WinEP is launched with the PAR file as the input, it processes the file as a batch control file. Each ERD file named in a DATASET is opened. This happens in VDM ROAD only when you click the **Plot** button in the **Plot Setup: Single** library.

Note: Plotting from the **Plot Setup: Single** library is a feature that exists in VDM ROAD for purposes of backward compatibility. This method of making a plot is not recommended, and may be removed from future versions.

On the other hand, if WinEP encounters the same file while processing a PLT file (first line = FILELIST), then the files named in the PAR file are ignored. Only the variable names and formatting information are used. The file acts as a template that is applied to the files named in the PLT file.

Plot Format Files

The initial appearance of a plot is read from a format file. Format files are plain ASCII text files that contain keywords with parameters that tell WinEP how to format a plot. All format files follow the PARFILE format and have the extension PAR.

To apply the format settings from a file, load the format file using the command **Load Plot Format** from the **File** menu.

To create a format settings file, use the command **Save Plot Format** from the **File** menu. The keywords used in a format file are described in Table B.2.

Table B.2. Format keywords and descriptions.

Keyword	Description
PARSFILE	Specifies the start of the format file.
END	Specifies the end of the format file.
XLINEAR <i>type</i>	X axis type. Valid values are <code>log</code> and <code>linear</code> .
YLINEAR <i>type</i>	Y axis type. Valid values are <code>log</code> and <code>linear</code> .
XMAXMIN <i>type</i>	Either a keyword specifying that the X axis is auto-scaled, or two numbers: the minimum and maximum values for the axis.
YMAXMIN <i>type</i>	Either a keyword specifying that the Y axis is auto-scaled, or two numbers: the minimum and maximum values for the axis.
SYMBOLS <i>s1,s2, ...,s20</i>	Symbols used for each X-Y data set. There must be a total of 20 symbols indicated. Symbol key: 0 = No symbol 1 = Square 2 = Triangle 3 = Diamond 4 = Cross X 5 = Plus 6 = Circle
LINESTYL <i>s1,s2,s3, ...,s20</i>	Line style for each X-Y data set. There must be a total of 20 styles indicated. Line style key: 0 = No Line 1 = Solid Line 2 = Dotted Line 3 = Heavy Solid Line 4 = Dash Line 5 = Dash Dot Line 6 = Dash Dot Dot Line
TITLEFONTSIZE <i>ps</i>	Size of the title font: <i>ps</i> = font point size.
TITLEFONTNAME <i>name</i>	Name of the title font (Ariel, etc.).
TITLEFONTSTYLE <i>style</i>	Style of the title font (Regular, Italic, Bold, Bold Italic)
LEGENDEFONTSIZE <i>ps</i>	Size of the legend font: <i>ps</i> = font point size.
LEGENDFONTNAME <i>name</i>	Name of the legend font (Ariel, etc.).
LEGENDFONTSTYLE <i>style</i>	Style of the legend font (Regular, Italic, Bold, Bold Italic)
LABELFONTSIZE <i>ps</i>	Size of the axis label font: <i>ps</i> = font point size.
LABELFONTNAME <i>name</i>	Name of the axis label font (Ariel, etc.).
LABELFONTSTYLE <i>style</i>	Style of the axis label font (Regular, Italic, etc.)
TICLABELFONTSIZE <i>ps</i>	Size of the axis numbers: <i>ps</i> = font point size.
TICLABELFONTNAME <i>name</i>	Name of the axis number font (Ariel, etc.).

TICLABELFONTSTYLE <i>style</i>	Style of the axis number font (Regular, Italic, etc.)
LEGENDLABEL <i>b1, b2, b3, b4, b5, b6</i>	Names to use in creating the legend. Values are 0 (do not use) or 1 (use). The sequence is: b1= shortname, b2 = longname, b3 = genname, b4 = rigidbody name, b5 = file title, b6 = file name.
LEGENDLOCATION <i>name</i>	Location in plot window for legend. Options are RightOfPlot, OnPlotUpperLeft, OnPlotUpperRight, OnPlotLowerLeft, and OnPlotLowerRight
GRID <i>type</i>	Type of grid. Options are NoGrid, CourseGrid, and FineGrid.
COLORS <i>c1,c2,c3,...,c20</i>	Color for each channel. There must be a total of 20 colors indicated. Color key: 0 = Black 1 = Maroon 2 = Red 3 = Green 4 = Bright Green 5 = Dark Blue 6 = Light Gray 7 = Light Blue 8 = Blue 9 = Muddy Yellow 10 = Yellow 11 = Gray 12 = Purple 13 = Teal 14 = Bright Purple

Preference File Format

The preference file `Epprefs.txt` is used to set and save information that is retained between WinEP sessions. This information includes the window positions and paths to directories. The information can be viewed and edited using the command **Preferences** under the **Edit** menu. Listing B.3 shows an example preferences file.

Listing B.3. Example Preferences file.

```
ep-prefs
temp-folder-path c:\VDM RoAD.in\plot\tmp
delete-temp-files off
last-datafile-path c:\VDM RoAD.in\plot
last-batchfile-path c:\VDM RoAD.in\plot\tmp
last-formatfile-path c:\VDM RoAD.in\plot\format
startup-window-mode maximize
plot-window-position 7,13,774,582
end
```

Appendix C — Animator Files and Keywords

Most information about objects and how they are viewed is read from an input PARSEFILE. It contains several kinds of information: (1) numerical constants such as coordinates of points, (2) names of variables in the ERD file that must be read by the animator, (3) names associated with objects (e.g., `black`), and (4) pathnames for linked PARSEFILES with more data. In all cases, white space immediately following the keyword is skipped, and the remainder of the line is interpreted as the value to be assigned to an internal variable. Spaces are significant after the first non-blank character following the keyword.

The PARSEFILE is similar to those used by other AutoSim-based programs. It should begin with a line containing the single keyword, `parsefile`. It should end with a line containing the single word `end`. (However, the `end` keyword is optional. If not present, the animator reads to the end of the file.) Blank lines and lines with unknown keywords are skipped when the file is read. Multiple lines with the same keyword are permitted.

Links to other PARSEFILES are provided with the `parsefile` keyword:

```
parsefile pathname
```

Through the use of the `parsefile` links, data for each object can be contained in a separate parsefile. For example, each part data set can be placed in a separate file. The data set for a reference-frame can be in a file that contains a `parsefile` link to each part associated with the frame. The entire animation data can be just a set of `parsefile` links to the non-moving parts, followed by the data sets for the moving reference frames.

At the top level, the animation shape data should be organized by reference frame. When a shape is introduced, it must be associated with a specific reference frame. After a new reference frame is introduced, all following inputs are said to fall within the *scope* of that reference frame. For example, a shape is in the scope of a reference frame if its coordinates are defined by the animator as being fixed in that reference frame.

Overall, the animation input data should be organized as follows:

```
parsefile
<info for fixed frame>
add_reference_frame {name1}
<info for first moving reference frame>
add_reference_frame {name2}
<info for second moving reference frame>
add_reference_frame {name3}
<info for third moving reference frame>
end
```

Reference Frame

Listing C.1 shows an example reference_frame description.

Listing C.1. Data file for a vehicle body.

```
parsfile
; start moving reference frame and give name
add_reference_frame vehicle sprung mass

set_euler_angles yaw_pitch_roll

; Set 6 names of variables in ERD file with XYZ, Roll, Pitch, Yaw
set_x_name Xo
set_y_name Yo
set_z_name Zo
set_pitch_name Pitch
set_roll_name Roll
set_yaw_name Yaw

; links to parts that make up vehicle
parsfile c:\animate\hood.par
parsfile c:\animate\trunk.par
parsfile c:\animate\roof.par
parsfile c:\animate\fenderLF.par
...
end
```

The keyword `add_reference_frame` has the dual effect of starting the scope of a new reference frame and ending the scope of the previous one. Until the first occurrence, inputs associated with a reference frame will have unpredictable results. It also assigns a name to the frame. In the example, the name is `vehicle sprung mass`.

Each reference frame must have a unique name.

The keyword `set_euler_angles` is used to specify the type of transformation used. There are two options: `yaw_pitch_roll` (used for vehicle sprung mass reference frames) and `yaw_roll_pitch` (used for rolling wheel reference frames).

The next six keywords in Listing C.1 are used to specify ERD file short names, and therefore their values are strings of text. White space between the keyword and the short name is ignored by the animator, and then the next eight characters are taken literally, including spaces. However, if the line ends before eight characters are read, the animator pads the short name with spaces until it is eight characters long. The six variables from the ERD file define the position and orientation of the reference frame. All six are optional. If not specified, the animator uses a value of zero for that variable.

Within the scope of each moving frame, you should specify one or more objects. The keywords `add_wheel` and `add_part` have the effect of starting the scope of a new object. They also have the effect of ending the scope of the previous object. However, they do not affect the scope of the current moving reference frame. In the example listing,

descriptions of parts and wheels are contained in other PARSEFILES, identified with pathnames following the `parsfile` keyword.

Parts

There are four properties that can be set for a part: a name, a list of XYZ coordinates, line thickness, and color. Listing C.2 shows an example PARSEFILE containing a complete part description.

Listing C.2. Part data description.

```
parsfile
add_part hood
set_Color blue

set_coordinates
-0.6273 -0.7344 0.8721
-0.5355 -0.4131 0.8874
-0.4896 -0.0000 0.9027
-0.5355 0.4131 0.8874
-0.6273 0.7344 0.8721
0.0000 0.7191 0.7803
0.5202 0.6579 0.6732
0.6426 0.2601 0.6885
0.6579 -0.0000 0.6885
0.6579 -0.2601 0.6885
0.4590 -0.6885 0.7038
0.0000 -0.7191 0.7803
-0.6273 -0.7344 0.8721
end_coordinates

set_scale_x 1
set_scale_y 1
set_scale_z 1

set_offset_x 0
set_offset_y 0
set_offset_z 0

end
```

It is sometimes convenient to use the same part in more than one reference frame. Therefore, the name is not necessarily unique. The part name is used only to make the PARSEFILE more readable, and possibly to provide more detailed error messages in future versions.

The color and line thickness default to values of black and 1, respectively. Therefore, the commands `set-line-thickness` and `set_color` are optional. There is no default set of coordinates, so the list must be provided. The list of coordinates begins with a line containing the keyword `set_coordinates`. Each following line should contain an X, Y, and Z coordinate, separated by white space, until the list ends with a line containing

the keyword `end_coordinates`. Within the block defined by the keywords `set_coordinates` and `end_coordinates`, there must not be blank lines, comments, or other keywords. The coordinates are assumed to apply to the current reference frame.

The listed coordinates for the part are transformed by the equations:

$$x_{\text{new}} = x_0 + s_x X$$

$$y_{\text{new}} = y_0 + s_y Y$$

$$z_{\text{new}} = z_0 + s_z Z$$

where x_0 , y_0 , and z_0 are offsets and s_x , s_y , and s_z are scale factors specified with the keywords `set_offset_x`, `set_offset_y`, `set_offset_z`, `set_scale_x`, `set_scale_y`, and `set_scale_z`. The keywords for scale factors and offsets are optional. For each part, the scale factors default to values of 1 and the offsets default to 0. By setting a scale factor to -1, a user can easily mirror a part. The offsets allow a user to change the location of a part within the reference frame without calculating new values for the coordinates.

Wheels

For a wheel, there are five properties that can be set: name, radius, width, line-thickness, and color. All properties except the name have a default value, and therefore, the commands shown below `add_wheel` are optional. It is likely that the same wheel definition will be used several times, and therefore, the name is not necessarily unique.

Listing C.3. Wheel data set.

```
parsfile
add_wheel Black wheel with 0.3 m radius
set_color black
set_num_points 10
set_radius 0.3
set_thickness 0.2
end
```

The six keywords for setting offsets and scale factors can also be specified for a wheel. A common application for vehicle animation is to use the `set_offset_y` keyword to locate wheels laterally on an axle.

Grid

The animator will draw a grid fixed in the global reference frame. If not specified, then there is no grid. It is added with the command `add_grid`, as shown in Listing C.4.

Listing C.4. Grid data set.

```
parsfile
add_grid
set_min_x 0
set_max_x 1000

set_min_y 0
set_max_y 500

set_interval_x 10
set_interval_y 5
end
```

Camera View

The viewing of a scene is controlled by the locations of two points: the camera and look point. Listing C.5 shows how the camera view is set.

Listing C.5. Camera data set.

```
parsfile

; set the focal length for the camera. It is roughly comparable to
; the lens of a 35-mm camera. For length units of meters, a typical
; value is 0.05 (50 mm). For units of feet, a typical value is 0.2.

set_focal_length 0.05

; Identify the camera reference frame and coordinates.

set_camera_reference_frame global

set_camera_point_x 100
set_camera_point_y 10
set_camera_point_z 1

set_focal_length 0.05

; Set the look point reference frame and coordinates.
set_camera_reference_frame sprung mass

set_look_point_x -2
set_look_point_y 0
set_look_point_z 1

end
```

Appendix D — Model Files and Keywords

This appendix describes the input and output files associated with the simulation programs in VDM RoAD. It lists two example echo files that contain the keywords recognized by the VDM RoAD solver programs.

File Types

Every time a new run is made from the **Runs** screen, seven files are created.

Table D.1. Standard files created when a simulation is run.

Name	Location	Creator	Description
Simfile	Programs	SGUI	Batch control file.
<id>.LPI	Runs	SGUI	List of parameters, read as input by program.
<id>.LPO	Runs	Program	List of parameters, written as output by program. Also contains initial conditions. Can be used to repeat a run.
<id>.LPF	Runs	Program	List of parameters and final conditions, written as output by program. Can be used to continue a run.
<id>.ERD	Runs	Program	Header for ERD file, needed by plotter and animator programs.
<id>.BIN	Runs	Program	Numerical values of output variables, stored in binary form.
<id>.LOG	Runs	Program	List of all PARSEFILES that were processed to make a run.

One file always has the same name, *Simfile*, and the other five all have the same base name—the current ID number displayed in the upper-corner of the **Runs** screen.

Simfile

Simfile is a batch control file. When a simulation program starts, it looks for the existence of *Simfile*. If the file is not found, the program will prompt you for an input file name. If the file is found, the program obtains all necessary input and output file names from *Simfile*. An example *Simfile* is shown below.

Listing D.6. Example Simfile.

```

SIMFILE
INPUT C:\VDM_ROAD\RUNS\575.LPI
ECHO C:\VDM_ROAD\RUNS\575.LPO
FINAL C:\VDM_ROAD\RUNS\575.LPF
ERDFILE C:\VDM_ROAD\RUNS\575.ERD
LOGFILE C:\VDM_ROAD\RUNS\575.LOG
END

```

Although it is a convention in VDM RoAD to use the same root name for all files, the individual simulation programs are capable of creating any names allowable under the file operating system.

Notice that the names in `Simfile` are full pathnames. The individual files can be located anywhere on the computer.

Whenever you click the **Run** button, a new `Simfile` is created in the same folder as the simulation program (by convention, this is the folder `Programs`). If a file with that name already exists, it is overwritten.

LPI Files

The LPI file lists the input parameters for the simulation program.

When you click the **Run** button, a new LPI file is created in the folder containing the `Runs` library (i.e., the `Runs.tbk` file). The LPI file contains information from the **Runs** screen. As shown in the following example, the file is relatively short.

Listing D.7. Example LPI file.

```

PARSFILE
parsfile C:\VDM_ROAD\VEHICLES\COMBOS\45.par
parsfile C:\VDM_ROAD\INPUT\HIGHWAY\23.par
parsfile C:\VDM_ROAD\INPUT\SPEED\28.par
parsfile C:\VDM_ROAD\INPUT\OFFSET\49.par
parsfile C:\VDM_ROAD\COMP_PAR\24.par

starts 0
stops 100
stopt 150

testid 575
title MH/B (Alt 3 R lane)
END

```

Each link seen in the **Runs** screen is represented with a full pathname to an existing data file. In the above example, the data from the selected vehicle is contained in the file `8.par`, in the directory `C:\VDM_ROAD\VEHICLES\COMBOS`.

The keyword `PARSFILE` is similar to the `INCLUDE` directive in `C` or `FORTRAN`: it instructs the program to open the specified file and to continue reading from the new file.

When the new file has been scanned to its end, the program continues reading from the original file.

PARSFILES are often nested five or six layers deep.

In order to view all of the parameters used in a simulation run, you could trace the input files the same way as the program: start with the Simfile to find the name of the input (LPI) file. Then, open the input file and view any parameter values in it. Open any files identified with the keyword PARSFILE, and continue the inspection.

LPO and LPF Echo Files

In VDM RoAD, the input parameter values are typically spread over many files. When a simulation program runs, it creates summary files that list each and every parameter value that is being used in the current simulation.

One of these files is created before the run (LPO), and the other is created at the end of the run. Both contain all parameter values used in the simulation. In addition, the LPO file contains the initial conditions of the state variables in the simulation model. The information in the LPO file is sufficient to exactly repeat a run, using only the LPO file as an input.

The other file is nearly identical, except that instead of the initial values, it contains the final values of the state variables. By modifying the start and stop time parameters, an existing run can be continued if the LPF file is used as the input file for the simulation solver program.

An example LPF file is listed later, to show all of the keywords recognized by a VDM RoAD model.

ERD and BIN Files

The main purpose of each simulation program is to predict time histories of variables of interest. These time histories are stored in a binary data file with the extension BIN. A companion file, with extension ERD, describes the layout of the BIN file and also contains labeling information for each variable.

The format of an ERD file is described in Appendix A. Appendix J contains a list of all of the variables contained in the ERD and BIN files for one of the VDM RoAD models.

LOG Files

The input parameters for the simulation models are spread over many screens in the VDM RoAD data base. Each data set is also contained in a PARSFILE. A LOG file is created to provide a record of every PARSFILE that was read during a simulation run. It is mainly used for debugging—most users will never have occasion to look at the LOG files.

When the simulation program executes, it reads the PARSFILE specified in the SIMFILE. If that PARSFILE includes the keyword PARSFILE, the named file is opened and the program continues reading from the newly opened file. For example, the top-

level file shown in Listing D.7 contains PARSEFILE keywords that identify other files with more specific types of data. Vehicle parameters are obtained by reading the file C:\VDM_ROAD\VEHICLES\COMBOS\45.PAR; highway design data are obtained by reading the file parsefile C:\VDM_ROAD\INPUT\HIGHWAY\23.PAR; the target vehicle speed as a function of station number is obtained by reading the file C:\VDM_ROAD\INPUT\SPEED\28.PAR; and so on. When the vehicle file is read, the program finds more references to other PARSEFILES, with data for suspensions, tires, and various vehicle properties.

The LOG file is a list of every PARSEFILE that was referenced, starting with the file named in the SIMFILE. In VDM RoAD, the top-level file has the extension LPI and all others have the extension PAR. Listing D.8 shows the top portion of a LOG file.

Listing D.8. Example LOG File.

```

Include PARSEFILE "C:\VDM_ROAD\RUNS\575.LPI"
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\COMBOS\45.par"
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\2A_TRACT\26.par"
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\PARTS\STRS_TRK\12.par"
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\PARTS\STR_KIN\53.par"
Heavy Tractor Ackerman; Left
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\PARTS\STR_KIN\54.par"
Heavy Tractor Ackerman; Right
Example
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\PARTS\BRKS_TRK\75.par"
4x2 Tractor
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\PARTS\HITCHES\13.par"
Ball Joint
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\PARTS\SUSP_TRK\107.par"
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\PARTS\SPRG_TRK\121.par"
HMMWV Rear
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\PARTS\SHOCKS\57.par"
Linear = 1
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\PARTS\AXLES\93.par"
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\PARTS\BRK_TRQ\56.par"
Linear Torque 900 Nm per P.I.
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\TIRES\TIRE_TAB\66.par"
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\TIRES\TIRE_TAB\TIRES_MZ\37.par"
Example Truck Tire
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\TIRES\TIRE_TAB\TIRES_FY\37.par"
Example Truck Tire
Include PARSEFILE "C:\VDM_ROAD\VEHICLES\TIRES\TIRE_TAB\TIRES_FX\30.par"
SAE Tire Data
Example truck tire
...

```

Viewing Keywords for the VDM RoAD Models

The files read by the simulation programs in VDM RoAD are scanned for input data. Parameters are identified by keywords.

Viewing a Complete List of Parameters

The following list of output variables was created from within the software for the 3a2a tractor-semitrailer vehicle. To create a similar list for other vehicle models, do the following:

1. Go to the **Runs** library, and find a run involving the vehicle of interest.
2. If the run has not already been made, click the **Run** button.
3. Click the **View All Parameters** button to view the file in a text editor.

Alternatively, look at the ID number in the upper-right corner of the **Runs** screen (e.g., 430). Then, use a file browser such as Windows Explorer to find the files with the ID numbers as a name with extensions LPO and LPF (e.g., 430.LPO, 430.LPF). Either one contains every keyword recognized by the simulation program.

Indexed Keywords

Many of the parameters in the VDM RoAD models are applied in several places on the vehicle. For example, every axle has a mass. The same data screen is used to describe an axle mass, no matter where it is located in the vehicle. The location of the axle is specified with additional keywords IUNIT and IAXLE. For components that differ on the left and right side, a keyword ISIDE is also used.

Parameter values such as an axle mass are written in the PARSEFILE with numerical indices, e.g., MUS(2,1) indicates the unsprung mass of unit 2 (trailer) and axle 1. Tabular data are written with keywords in front of the table to indicate which unit, axle, etc., are associated with the table.

Keywords for the Truck Models

Listing D.9 shows the complete set of model parameters for a heavy truck model in VDM RoAD.

Note: The example truck echo file is for a five-axle tractor semitrailer vehicle. Therefore it has parameters for two vehicle units and five axles. Keywords and parameters listed in this file do not apply for vehicle models if they refer to parts that don't exist. For example, trailer parameters are not relevant for a simulation program involving a single-unit truck or bus.

Listing D.9. LPF file for 3a2a vehicle.

PARSFILE

* 3-axle tractor with 2-axle trailer

* VDM road braking and handling model.

* MM Units

* Generated by AutoSim October 15, 1997

* Copyright 1997. The Regents of the University of Michigan. All Rights Reserved.

TITLE WB-20 (Alt 3 R lane)

* Input File: C:\VDM_ROAD\RUNS\555.LPI

* Run was made 21:32 on Oct 15, 1997

FORMAT

INTOPT 3, Type of integration (0=Adams, 1=Gear+PEDERV, 2=Gear diff., 3=RK2) (-)

IPRINT 40, number of time steps between output printing (counts)

EPSINT .100000E-01, Dimensionless integrator error tolerance (-)

STARTT .000000, simulation start time (s)

STEP .250000E-02, simulation time step (s)

STEPO .100000E-04, Initial integration step (s)

STEPMIN .100000E-06, Minimum allowable integration step (s)

STOPT 117.597, simulation stop time (s)

* PARAMETER VALUES

ABS_SLIPOFF(1,1) .300000, ABS parameter: slip ratio to turn brake off (-)

ABS_SLIPOFF(1,2) .300000, ABS parameter: slip ratio to turn brake off (-)

ABS_SLIPOFF(1,3) .300000, ABS parameter: slip ratio to turn brake off (-)

ABS_SLIPOFF(2,1) .300000, ABS parameter: slip ratio to turn brake off (-)

ABS_SLIPOFF(2,2) .300000, ABS parameter: slip ratio to turn brake off (-)

ABS_SLIPON(1,1) .200000, ABS parameter: slip ratio to turn brake back on (-)

ABS_SLIPON(1,2) .200000, ABS parameter: slip ratio to turn brake back on (-)

ABS_SLIPON(1,3) .200000, ABS parameter: slip ratio to turn brake back on (-)

ABS_SLIPON(2,1) .200000, ABS parameter: slip ratio to turn brake back on (-)

ABS_SLIPON(2,2) .200000, ABS parameter: slip ratio to turn brake back on (-)

ABS_VMIN(1,1) 6.00000, ABS parameter: minimum speed for ABS control (kph)

ABS_VMIN(1,2) 6.00000, ABS parameter: minimum speed for ABS control (kph)

ABS_VMIN(1,3) 6.00000, ABS parameter: minimum speed for ABS control (kph)

ABS_VMIN(2,1) 6.00000, ABS parameter: minimum speed for ABS control (kph)

ABS_VMIN(2,2) 6.00000, ABS parameter: minimum speed for ABS control (kph)

ACAST(1) 5.20000, Caster angle for LF wheel (deg)

ACAST(2) 5.20000, Caster angle for RF wheel (deg)

AKPI(1) 7.20000, Kingpin inclination angle for LF wheel (deg)

AKPI(2) 7.20000, Kingpin inclination angle for RF wheel (deg)

CSMZR .420000E-03, Torsional steering system complinace due to tie rods (deg/N/m)

CSMZS	.222000E-02	, Steering shaft compliance (deg/N/m)
CWRAP	.186000E-03	, Wrap-up compliance of front axle (deg/N/m)
HCGA(1,1)	480.000	, Height of Axle 1 CG above ground (mm)
HCGA(1,2)	480.000	, Height of Axle 2 CG above ground (mm)
HCGA(1,3)	480.000	, Height of Axle 3 CG above ground (mm)
HCGA(2,1)	546.000	, Height of Axle 4 CG above ground (mm)
HCGA(2,2)	546.000	, Height of Axle 5 CG above ground (mm)
HCGPL	2180.00	, CALC -- CG height of box load (mm)
HCGSML(1)	1079.46	, CALC -- CG height of sprung mass of laden Tractor (mm)
HCGSML(2)	2038.85	, CALC -- CG height of sprung mass of laden Semitrailer (mm)
HCGTL(1)	872.000	, CALC -- CG height of total laden Tractor (mm)
HCGTL(2)	1956.27	, CALC -- CG height of total laden Semitrailer (mm)
HCGTU(1)	872.000	, Height of total unladen tractor CG above ground (mm)
HCGTU(2)	1300.00	, Height of total unladen semitrailer CG above ground (mm)
HH(1)	1200.00	, Height of hitch 1 above ground (mm)
HLLB(1)	30.0000	, Height of box load bottom above cargo floor (mm)
HRC(1,1)	427.000	, Height of Axle 1 roll center above ground (mm)
HRC(1,2)	695.000	, Height of Axle 2 roll center above ground (mm)
HRC(1,3)	695.000	, Height of Axle 3 roll center above ground (mm)
HRC(2,1)	673.000	, Height of Axle 4 roll center above ground (mm)
HRC(2,2)	673.000	, Height of Axle 5 roll center above ground (mm)
HTLB(2)	1150.00	, Height of semitrailer cargo floor above ground (mm)
IA(1,1)	540.000	, Axle 1 yaw and roll moment of inertia (kg-m ²)
IA(1,2)	540.000	, Axle 2 yaw and roll moment of inertia (kg-m ²)
IA(1,3)	540.000	, Axle 3 yaw and roll moment of inertia (kg-m ²)
IA(2,1)	600.000	, Axle 4 yaw and roll moment of inertia (kg-m ²)
IA(2,2)	600.000	, Axle 5 yaw and roll moment of inertia (kg-m ²)
IW(1,1)	60.0000	, Wheel spin inertia, Axle 1 (kg-m ²)
IW(1,2)	60.0000	, Wheel spin inertia, Axle 2 (kg-m ²)
IW(1,3)	60.0000	, Wheel spin inertia, Axle 3 (kg-m ²)
IW(2,1)	50.7000	, Wheel spin inertia, Axle 4 (kg-m ²)
IW(2,2)	50.7000	, Wheel spin inertia, Axle 5 (kg-m ²)
IXXPL	13481.3	, CALC -- Moment of inertia for box load (kg-m ²)
IXXSM(1)	6782.10	, CALC -- Moment of inertia for sprung mass of laden Tractor (kg-m ²)
IXXSM(2)	24926.8	, CALC -- Moment of inertia for sprung mass of laden Semitrailer (kg-m ²)
IXXTL(1)	9155.00	, CALC -- Moment of inertia for total laden Tractor (kg-m ²)
IXXTL(2)	29284.7	, CALC -- Moment of inertia for total laden Semitrailer (kg-m ²)
IXXTU(1)	9155.00	, Moment of inertia for total unladen tractor (kg-m ²)
IXXTU(2)	11822.0	, Moment of inertia for total unladen semitrailer (kg-m ²)
IXZSM(1)	2996.73	, CALC -- Product of inertia for sprung mass of laden Tractor (kg-m ²)
IXZSM(2)	181.121	, CALC -- Product of inertia for sprung mass of laden Semitrailer (kg-m ²)
IXZTL(1)	.000000	, CALC -- Product of inertia for total laden Tractor (kg-m ²)
IXZTL(2)	-13583.6	, CALC -- Product of inertia for total laden Semitrailer (kg-m ²)
IXZTU(1)	.000000	, Product of inertia for total unladen tractor (kg-m ²)
IXZTU(2)	.000000	, Product of inertia for total unladen semitrailer (kg-m ²)
IYYPL	175257.	, CALC -- Moment of inertia for box load (kg-m ²)

IYYSM(1) 3319.94 , CALC -- Moment of inertia for sprung mass of laden Tractor (kg-m2)
 IYYSM(2) 396188. , CALC -- Moment of inertia for sprung mass of laden Semitrailer (kg-m2)
 IYYTL(1) 43263.0 , CALC -- Moment of inertia for total laden Tractor (kg-m2)
 IYYTL(2) 459883. , CALC -- Moment of inertia for total laden Semitrailer (kg-m2)
 IYYTU(1) 43263.0 , Moment of inertia for total unladen tractor (kg-m2)
 IYYTU(2) 234300. , Moment of inertia for total unladen semitrailer (kg-m2)
 IZZPL 175257. , CALC -- Moment of inertia for box load (kg-m2)
 IZZSM(1) 1364.84 , CALC -- Moment of inertia for sprung mass of laden Tractor (kg-m2)
 IZZSM(2) 406565. , CALC -- Moment of inertia for sprung mass of laden Semitrailer (kg-m2)
 IZZTL(1) 42175.0 , CALC -- Moment of inertia for total laden Tractor (kg-m2)
 IZZTL(2) 468301. , CALC -- Moment of inertia for total laden Semitrailer (kg-m2)
 IZZTU(1) 42175.0 , Moment of inertia for total unladen tractor (kg-m2)
 IZZTU(2) 246700. , Moment of inertia for total unladen semitrailer (kg-m2)
 KAUX(1,1) 1045.00 , Axle 1 auxiliary roll stiffness (N-m/deg)
 KAUX(1,2) 1356.00 , Axle 2 auxiliary roll stiffness (N-m/deg)
 KAUX(1,3) 1356.00 , Axle 3 auxiliary roll stiffness (N-m/deg)
 KAUX(2,1) 2260.00 , Axle 4 auxiliary roll stiffness (N-m/deg)
 KAUX(2,2) 2260.00 , Axle 5 auxiliary roll stiffness (N-m/deg)
 KHX(1) 650000. , Hitch roll torsional stiffness (N-m/deg)
 KHY(1) .000000 , Hitch pitch torsional stiffness (N-m/deg)
 KHZ(1) .000000 , Hitch articulation torsional stiffness (N-m/deg)
 KT(1,1) 875.000 , Axle 1 tire spring rate (N/mm)
 KT(1,2) 875.000 , Axle 2 tire spring rate (N/mm)
 KT(1,3) 875.000 , Axle 3 tire spring rate (N/mm)
 KT(2,1) 875.000 , Axle 4 tire spring rate (N/mm)
 KT(2,2) 875.000 , Axle 5 tire spring rate (N/mm)
 LCPF 20.0000 , Distance front camera point is in front of front axle (m)
 LCPR 20.0000 , Distance rear camera point is behind rear axle (m)
 LDUAL(1,1) 330.000 , Axle 1 dual tire spacing (use 0 for singles) (mm)
 LDUAL(1,2) 330.000 , Axle 2 dual tire spacing (use 0 for singles) (mm)
 LDUAL(1,3) 330.000 , Axle 3 dual tire spacing (use 0 for singles) (mm)
 LDUAL(2,1) 330.000 , Axle 4 dual tire spacing (use 0 for singles) (mm)
 LDUAL(2,2) 330.000 , Axle 5 dual tire spacing (use 0 for singles) (mm)
 LKPO(1) 100.000 , Kingpin offset (at wheel center) for LF wheel (mm)
 LKPO(2) 100.000 , Kingpin offset (at wheel center) for RF wheel (mm)
 LRELAX(1,1) 600.000 , Tire relaxation length, axle 1 (mm)
 LRELAX(1,2) 600.000 , Tire relaxation length, axle 2 (mm)
 LRELAX(1,3) 600.000 , Tire relaxation length, axle 3 (mm)
 LRELAX(2,1) 600.000 , Tire relaxation length, axle 4 (mm)
 LRELAX(2,2) 600.000 , Tire relaxation length, axle 5 (mm)
 LTAND(1) 1200.00 , Longitudinal tandem axle spacing for Tractor (mm)
 LTAND(2) 1200.00 , Longitudinal tandem axle spacing for Semitrailer (mm)
 LTD(1,1) 800.000 , Distance between dampers on Axle 1 (mm)
 LTD(1,2) 950.000 , Distance between dampers on Axle 2 (mm)
 LTD(1,3) 950.000 , Distance between dampers on Axle 3 (mm)
 LTD(2,1) 980.000 , Distance between dampers on Axle 4 (mm)

LTD(2,2)	980.000	, Distance between dampers on Axle 5 (mm)
LTK(1,1)	1810.00	, Axle 1 ltk (to center of dual tires) (mm)
LTK(1,2)	1810.00	, Axle 2 ltk (to center of dual tires) (mm)
LTK(1,3)	1810.00	, Axle 3 ltk (to center of dual tires) (mm)
LTK(2,1)	1943.00	, Axle 4 ltk (to center of dual tires) (mm)
LTK(2,2)	1943.00	, Axle 5 ltk (to center of dual tires) (mm)
LTS(1,1)	823.000	, Distance between springs on Axle 1 (mm)
LTS(1,2)	1030.00	, Distance between springs on Axle 2 (mm)
LTS(1,3)	1030.00	, Distance between springs on Axle 3 (mm)
LTS(2,1)	1030.00	, Distance between springs on Axle 4 (mm)
LTS(2,2)	1030.00	, Distance between springs on Axle 5 (mm)
LWB(1)	6100.00	, Wheelbase of tractor (mm)
LWB(2)	14300.0	, Wheelbase of semitrailer (mm)
LXCGPL	7389.72	, CALC -- X dist (to rear) of box load CG (mm)
LXCGSML(1)	1680.67	, CALC -- X dist (positive to rear of tractor front axle) of sprung mass of laden Tractor CG (mm)
LXCGSML(2)	7793.10	, CALC -- X dist (positive to rear of hitch 1) of sprung mass of laden Semitrailer CG (mm)
LXCGTL(1)	2506.42	, CALC -- X dist (positive to rear of tractor front axle) of total laden Tractor CG (mm)
LXCGTL(2)	8153.05	, CALC -- X dist (positive to rear of hitch 1) of total laden Semitrailer CG (mm)
LXCGTUL(1)	2506.42	, CALC -- X dist (positive to rear of tractor front axle) of total unladen Tractor CG (mm)
LXCGTUL(2)	10392.1	, CALC -- X dist (positive to rear of hitch 1) of total unladen Semitrailer CG (mm)
LXHITCH(1)	6100.00	, CALC -- X dist (positive to rear of tractor front axle) of hitch 1 (mm)
LXRL(1)	10000.0	, Length of box load (mm)
LYRL(1)	2000.00	, Width of box load (mm)
LZRL(1)	2000.00	, Height of box load (mm)
M(1,1)	5454.00	, Laden load for Axle 1 (kg)
M(1,2)	7730.00	, Laden load for Axle 2 (kg)
M(1,3)	7730.00	, Laden load for Axle 3 (kg)
M(2,1)	7730.00	, Laden load for Axle 4 (kg)
M(2,2)	7730.00	, Laden load for Axle 5 (kg)
MPLD	20222.0	, CALC -- Mass of box load (kg)
MSTL(1)	6054.00	, CALC -- Sprung laden mass of Tractor (kg)
MSTL(2)	25616.0	, CALC -- Sprung laden mass of Semitrailer (kg)
MTL(1)	9258.00	, CALC -- Total laden mass of Tractor (kg)
MTL(2)	27116.0	, CALC -- Total laden mass of Semitrailer (kg)
MTRAIL(2)	6894.00	, Mass of total unladen semitrailer (kg)
MTUL(1)	9258.00	, CALC -- Total unladen mass of Tractor (kg)
MTUL(2)	6894.00	, CALC -- Total unladen mass of Semitrailer (kg)
MU	.800000	, Tire/ground friction coefficient (-)
MUL(1,1)	5454.00	, Unladen load for Axle 1 (vehicle unit by itself) (kg)
MUL(1,2)	1902.00	, Unladen load for Axle 2 (vehicle unit by itself) (kg)
MUL(1,3)	1902.00	, Unladen load for Axle 3 (vehicle unit by itself) (kg)
MUL(2,1)	2505.00	, Unladen load for Axle 4 (vehicle unit by itself) (kg)
MUL(2,2)	2505.00	, Unladen load for Axle 5 (vehicle unit by itself) (kg)
MUS(1,1)	1068.00	, Mass of Axle 1 (kg)
MUS(1,2)	1068.00	, Mass of Axle 2 (kg)

MUS(1,3) 1068.00 , Mass of Axle 3 (kg)
 MUS(2,1) 750.000 , Mass of Axle 4 (kg)
 MUS(2,2) 750.000 , Mass of Axle 5 (kg)
 RBMPSTR .627000E-02 , Bump-steer ratio: steer/jounce (deg/mm)
 ROLL_STOP 45.0000 , Roll angle for stopping the simulation (deg)
 RSTR(1,1) -.225000 , Axle 1 roll-steer coefficient (axle steer / axle roll) (-)
 RSTR(1,2) .143000 , Axle 2 roll-steer coefficient (axle steer / axle roll) (-)
 RSTR(1,3) .143000 , Axle 3 roll-steer coefficient (axle steer / axle roll) (-)
 RSTR(2,1) -.800000 , Axle 4 roll-steer coefficient (axle steer / axle roll) (-)
 RSTR(2,2) -.800000 , Axle 5 roll-steer coefficient (axle steer / axle roll) (-)
 RTIME 3.47797 , CALC -- Computational efficiency (sec/sim. sec) (-)
 RWRAPSTR .140000 , Ratio: steer due to axle wrap (-)
 SPEED_INT 1.00000 , Forward speed/spin rate initialization switch: if 0.0 --> do not initialize,
 initialized otherwise (-)
 SPEED_KI .157000 , Integral control gain for speed controller (rev/s²)
 SPEED_KP .141000 , Proportional control gain for speed controller (rev/s)
 SPEED_ON_OFF 1.00000 , Speed control switch (0.0 -> off, 1.0 -> on) (-)
 SPRING_INIT 1.00000 , UMTRI spring init: 0.0 -> equilibrium, otherwise use initial condition (-)
 STARTS .000000 , Starting station number (beginning of simulation) (m)
 STATION 1900.01 , Station number -- distance along input path (m)
 STOPS 1900.00 , Stopping station number (stop simulation when this is reached) (m)
 TBRKLAG(1,1) .500000 , Transport delay for air to brakes on axle 1 (s)
 TBRKLAG(1,2) .500000 , Transport delay for air to brakes on axle 2 (s)
 TBRKLAG(1,3) .500000 , Transport delay for air to brakes on axle 3 (s)
 TBRKLAG(2,1) .500000 , Transport delay for air to brakes on axle 4 (s)
 TBRKLAG(2,2) .500000 , Transport delay for air to brakes on axle 5 (s)
 TBRKTC(1,1) .310000 , Time constant for air chambers for brakes on axle 1 (s)
 TBRKTC(1,2) .310000 , Time constant for air chambers for brakes on axle 2 (s)
 TBRKTC(1,3) .310000 , Time constant for air chambers for brakes on axle 3 (s)
 TBRKTC(2,1) .310000 , Time constant for air chambers for brakes on axle 4 (s)
 TBRKTC(2,2) .310000 , Time constant for air chambers for brakes on axle 5 (s)
 TDLAG .000000 , Lag time used by driver model (s)
 TESTID 555.000 , Run ID number (echoed in ERD file header) (-)
 TPREV .200000 , Preview time used by driver model (s)
 VLOW_ALPHA(1,1) 16.0934 , Low-speed threshold for modified tire relaxation equations, axle 1 (kph)
 VLOW_ALPHA(1,2) 16.0934 , Low-speed threshold for modified tire relaxation equations, axle 2 (kph)
 VLOW_ALPHA(1,3) 16.0934 , Low-speed threshold for modified tire relaxation equations, axle 3 (kph)
 VLOW_ALPHA(2,1) 16.0934 , Low-speed threshold for modified tire relaxation equations, axle 4 (kph)
 VLOW_ALPHA(2,2) 16.0934 , Low-speed threshold for modified tire relaxation equations, axle 5 (kph)
 VLOW_KAPPA(1,1) 8.04672 , Low-speed threshold for modified longitudinal slip equations, axle 1 (kph)
 VLOW_KAPPA(1,2) 8.04672 , Low-speed threshold for modified longitudinal slip equations, axle 2 (kph)
 VLOW_KAPPA(1,3) 8.04672 , Low-speed threshold for modified longitudinal slip equations, axle 3 (kph)
 VLOW_KAPPA(2,1) 8.04672 , Low-speed threshold for modified longitudinal slip equations, axle 4 (kph)
 VLOW_KAPPA(2,2) 8.04672 , Low-speed threshold for modified longitudinal slip equations, axle 5 (kph)
 VLOW_SPINA(1,1) 8.04672 , Low-speed threshold for modified wheel spin equations, Axle 1 (kph)
 VLOW_SPINA(1,2) 8.04672 , Low-speed threshold for modified wheel spin equations, Axle 2 (kph)

VLOW_SPINA(1,3) 8.04672 , Low-speed threshold for modified wheel spin equations, Axle 3 (kph)
 VLOW_SPINA(2,1) 8.04672 , Low-speed threshold for modified wheel spin equations, Axle 4 (kph)
 VLOW_SPINA(2,2) 8.04672 , Low-speed threshold for modified wheel spin equations, Axle 5 (kph)
 V_STOP -1.00000 , Low-speed condition for stopping the simulation (kph)

* Cross Slope Data: Station, DZDL, LAT, DZDL, ... , LAT, DZDL

* where LAT = lateral position (left, m), DZDL = cross-slope

* for lane (rise going left = positive, %).

CROSS_SLOPE_TABLE 4

.000000	, 2.00000	, .000000	, -2.00000
253.257	, 2.00000	, .000000	, -2.00000
277.836	, -2.00000	, .000000	, -2.00000
283.059	, -2.85000	, .000000	, -2.85000
303.260	, -6.13800	, .000000	, -6.13800
312.861	, -7.70000	, .000000	, -7.70000
373.641	, -7.70000	, .000000	, -7.70000
403.443	, -5.41200	, .000000	, -5.41200
473.938	, .000000	, .000000	, .000000
545.427	, 5.48800	, .000000	, 5.48800
575.537	, 7.80000	, .000000	, 7.80000
645.750	, 7.80000	, .000000	, 7.80000
699.452	, 7.80000	, .000000	, 7.80000
729.561	, 4.56300	, .000000	, 4.56300
772.013	, .000000	, .000000	, .000000
815.710	, -4.69700	, .000000	, -4.69700
846.434	, -8.00000	, .000000	, -8.00000
958.065	, -8.00000	, .000000	, -8.00000
988.789	, -5.06300	, .000000	, -5.06300
1041.75	, .000000	, .000000	, .000000
1094.71	, 5.06300	, .000000	, 5.06300
1125.43	, 8.00000	, .000000	, 8.00000
1153.65	, 8.00000	, .000000	, 8.00000
1230.52	, 8.00000	, .000000	, 8.00000
1261.24	, 5.53200	, .000000	, 5.53200
1330.09	, .000000	, .000000	, .000000
1398.95	, -5.53200	, .000000	, -5.53200
1429.67	, -8.00000	, .000000	, -8.00000
1516.21	, -8.00000	, .000000	, -8.00000
1546.94	, -5.15200	, .000000	, -5.15200
1602.53	, .000000	, .000000	, .000000
1652.04	, 4.58900	, .000000	, 4.58900
1658.12	, 5.15200	, .000000	, 5.15200
1688.84	, 8.00000	, .000000	, 8.00000
1789.57	, 8.00000	, .000000	, 8.00000
1820.29	, 3.00000	, .000000	, 3.00000
1826.43	, 2.00000	, .000000	, 2.00000

```
1851.01 , 2.00000 , .000000 , -2.00000
ENDTABLE
```

```
IUNIT 1
IAXLE 1
FD_TABLE Damper force vs. stroke rate
.000000 , .000000 , point in table: (mm/s, N)
25400.0 , 71.0000 , point in table: (mm/s, N)
ENDTABLE
```

```
IUNIT 1
IAXLE 2
FD_TABLE Damper force vs. stroke rate
.000000 , .000000 , point in table: (mm/s, N)
25400.0 , 160.140 , point in table: (mm/s, N)
ENDTABLE
```

```
IUNIT 1
IAXLE 3
FD_TABLE Damper force vs. stroke rate
.000000 , .000000 , point in table: (mm/s, N)
25400.0 , 160.140 , point in table: (mm/s, N)
ENDTABLE
```

```
IUNIT 2
IAXLE 1
FD_TABLE Damper force vs. stroke rate
.000000 , .000000 , point in table: (mm/s, N)
1000.00 , .000000 , point in table: (mm/s, N)
ENDTABLE
```

```
IUNIT 2
IAXLE 2
FD_TABLE Damper force vs. stroke rate
.000000 , .000000 , point in table: (mm/s, N)
1000.00 , .000000 , point in table: (mm/s, N)
ENDTABLE
```

```
* Horizontal Geometry Data: Station, Radius, Spiral Par., X, Y, Heading
HGDATA
```

```
.000000 , .000000 , .000000 , 54156.3 , 117321. , -29.5000
283.059 , -155.000 , .000000 , 54402.7 , 117182. , -29.5000
403.443 , .000000 , .000000 , 54519.1 , 117167. , 15.0003
545.427 , 150.000 , .000000 , 54656.2 , 117204. , 15.0003
729.561 , .000000 , .000000 , 54818.4 , 117144. , -55.3339
815.710 , -125.000 , .000000 , 54867.4 , 117073. , -55.3339
```

```

988.789 , .000000 , .000000 , 55021.1 , 117030. , 23.9997
1094.71 , 125.000 , .000000 , 55117.9 , 117073. , 23.9997
1261.24 , .000000 , .000000 , 55267.6 , 117035. , -52.3333
1398.95 , -125.000 , .000000 , 55351.8 , 116926. , -52.3333
1546.94 , .000000 , .000000 , 55484.1 , 116882. , 15.5002
1658.12 , 125.000 , .000000 , 55591.3 , 116912. , 15.5002
1820.29 , .000000 , .000000 , 55731.5 , 116856. , -58.8332

```

ENDTABLE

LTARG_TABLE Lateral offset target (+ = left, - = right) as function of station.

```
.000000 , -1.65000 , point in table: (m, m)
```

```
1.00000 , -1.65000 , point in table: (m, m)
```

ENDTABLE

IUNIT 1

IAXLE 1

ISIDE 1

MYBK_TABLE Brake torque at wheel vs. line pressure

```
.000000 , .000000 , point in table: (kPa, N-m)
```

```
168.700 , 300000. , point in table: (kPa, N-m)
```

ENDTABLE

IUNIT 1

IAXLE 1

ISIDE 2

MYBK_TABLE Brake torque at wheel vs. line pressure

```
.000000 , .000000 , point in table: (kPa, N-m)
```

```
165.000 , 300000. , point in table: (kPa, N-m)
```

ENDTABLE

IUNIT 1

IAXLE 2

ISIDE 1

MYBK_TABLE Brake torque at wheel vs. line pressure

```
.000000 , .000000 , point in table: (kPa, N-m)
```

```
168.700 , 300000. , point in table: (kPa, N-m)
```

ENDTABLE

IUNIT 1

IAXLE 2

ISIDE 2

MYBK_TABLE Brake torque at wheel vs. line pressure

```
.000000 , .000000 , point in table: (kPa, N-m)
```

```
165.000 , 300000. , point in table: (kPa, N-m)
```

ENDTABLE

```
IUNIT 1
IAXLE 3
ISIDE 1
MYBK_TABLE Brake torque at wheel vs. line pressure
.000000 , .000000 , point in table: (kPa, N-m)
168.700 , 300000. , point in table: (kPa, N-m)
ENDTABLE
```

```
IUNIT 1
IAXLE 3
ISIDE 2
MYBK_TABLE Brake torque at wheel vs. line pressure
.000000 , .000000 , point in table: (kPa, N-m)
165.000 , 300000. , point in table: (kPa, N-m)
ENDTABLE
```

```
IUNIT 1
IAXLE 1
ISIDE 1
MYBK_TABLE Brake torque at wheel vs. line pressure
.000000 , .000000 , point in table: (kPa, N-m)
168.700 , 300000. , point in table: (kPa, N-m)
ENDTABLE
```

```
IUNIT 1
IAXLE 1
ISIDE 2
MYBK_TABLE Brake torque at wheel vs. line pressure
.000000 , .000000 , point in table: (kPa, N-m)
165.000 , 300000. , point in table: (kPa, N-m)
ENDTABLE
```

```
IUNIT 1
IAXLE 2
ISIDE 1
MYBK_TABLE Brake torque at wheel vs. line pressure
.000000 , .000000 , point in table: (kPa, N-m)
168.700 , 300000. , point in table: (kPa, N-m)
ENDTABLE
```

```
IUNIT 1
IAXLE 2
ISIDE 2
MYBK_TABLE Brake torque at wheel vs. line pressure
.000000 , .000000 , point in table: (kPa, N-m)
165.000 , 300000. , point in table: (kPa, N-m)
```

ENDTABLE

PBRAKE_TABLE Brake input pressure vs. time

.000000 , .000000 , point in table: (s, kPa)

2.00000 , .000000 , point in table: (s, kPa)

ENDTABLE

SPEED_TABLE Vehicle forward speed vs. station number

.000000 , 100.000 , point in table: (m, kph)

150.000 , 100.000 , point in table: (m, kph)

300.000 , 70.0000 , point in table: (m, kph)

700.000 , 70.0000 , point in table: (m, kph)

800.000 , 50.0000 , point in table: (m, kph)

1000.00 , 50.0000 , point in table: (m, kph)

ENDTABLE

* Data for the UMTRI spring model

IUNIT 1

IAXLE 1

SPRING_BETA 2.00000 , "beta" parameter--characteristic deflection (mm)

SPRING_UPPER_ENVELOPE

43.9400 , 3392.70 , point in table: (mm, N)

56.3900 , 7916.32 , point in table: (mm, N)

100.330 , 18773.0 , point in table: (mm, N)

155.450 , 31439.1 , point in table: (mm, N)

181.360 , 37998.3 , point in table: (mm, N)

199.900 , 45236.1 , point in table: (mm, N)

209.550 , 50438.3 , point in table: (mm, N)

223.010 , 65366.2 , point in table: (mm, N)

ENDTABLE

SPRING_LOWER_ENVELOPE

50.0400 , 4749.81 , point in table: (mm, N)

74.4200 , 11082.9 , point in table: (mm, N)

109.220 , 18999.2 , point in table: (mm, N)

163.320 , 31665.3 , point in table: (mm, N)

178.560 , 35736.6 , point in table: (mm, N)

203.960 , 44557.6 , point in table: (mm, N)

210.570 , 48402.6 , point in table: (mm, N)

227.080 , 65366.2 , point in table: (mm, N)

ENDTABLE

* Data for the UMTRI spring model

IUNIT 1

IAXLE 2

SPRING_BETA 1.70000 , "beta" parameter--characteristic deflection (mm)

SPRING_UPPER_ENVELOPE

29.9700 , 5426.56 , point in table: (mm, N)
 48.2600 , 14473.8 , point in table: (mm, N)
 61.9800 , 22920.5 , point in table: (mm, N)
 75.6900 , 31362.8 , point in table: (mm, N)
 80.0100 , 34681.1 , point in table: (mm, N)
 91.9500 , 47647.0 , point in table: (mm, N)
 105.920 , 63023.7 , point in table: (mm, N)
 120.140 , 82928.4 , point in table: (mm, N)

ENDTABLE

SPRING_LOWER_ENVELOPE

31.2400 , 3318.21 , point in table: (mm, N)
 42.8500 , 9647.70 , point in table: (mm, N)
 64.2600 , 16889.1 , point in table: (mm, N)
 75.4400 , 22920.5 , point in table: (mm, N)
 86.8700 , 32568.3 , point in table: (mm, N)
 99.0600 , 44631.2 , point in table: (mm, N)
 107.440 , 54278.9 , point in table: (mm, N)
 121.410 , 70265.1 , point in table: (mm, N)

ENDTABLE

* Data for the UMTRI spring model

IUNIT 1

IAXLE 3

SPRING_BETA 1.70000 , "beta" parameter--characteristic deflection (mm)

SPRING_UPPER_ENVELOPE

29.9700 , 5426.56 , point in table: (mm, N)
 48.2600 , 14473.8 , point in table: (mm, N)
 61.9800 , 22920.5 , point in table: (mm, N)
 75.6900 , 31362.8 , point in table: (mm, N)
 80.0100 , 34681.1 , point in table: (mm, N)
 91.9500 , 47647.0 , point in table: (mm, N)
 105.920 , 63023.7 , point in table: (mm, N)
 120.140 , 82928.4 , point in table: (mm, N)

ENDTABLE

SPRING_LOWER_ENVELOPE

31.2400 , 3318.21 , point in table: (mm, N)
 42.8500 , 9647.70 , point in table: (mm, N)
 64.2600 , 16889.1 , point in table: (mm, N)
 75.4400 , 22920.5 , point in table: (mm, N)
 86.8700 , 32568.3 , point in table: (mm, N)
 99.0600 , 44631.2 , point in table: (mm, N)
 107.440 , 54278.9 , point in table: (mm, N)

121.410 , 70265.1 , point in table: (mm, N)
 ENDTABLE

* Data for the UMTRI spring model

IUNIT 2

IAXLE 1

SPRING_BETA 4.45000 , "beta" parameter--characteristic deflection (mm)

SPRING_UPPER_ENVELOPE

13.9700 , 3918.69 , point in table: (mm, N)

30.7300 , 4523.62 , point in table: (mm, N)

33.5300 , 7236.90 , point in table: (mm, N)

43.6900 , 23521.0 , point in table: (mm, N)

55.8800 , 47647.0 , point in table: (mm, N)

63.7500 , 61818.3 , point in table: (mm, N)

72.9000 , 80517.7 , point in table: (mm, N)

ENDTABLE

SPRING_LOWER_ENVELOPE

12.9500 , 3318.21 , point in table: (mm, N)

31.7500 , 4221.15 , point in table: (mm, N)

35.3100 , 7236.90 , point in table: (mm, N)

43.1800 , 16889.1 , point in table: (mm, N)

57.6600 , 41917.9 , point in table: (mm, N)

64.2600 , 57294.7 , point in table: (mm, N)

72.9000 , 74486.2 , point in table: (mm, N)

ENDTABLE

* Data for the UMTRI spring model

IUNIT 2

IAXLE 2

SPRING_BETA 4.45000 , "beta" parameter--characteristic deflection (mm)

SPRING_UPPER_ENVELOPE

13.9700 , 3918.69 , point in table: (mm, N)

30.7300 , 4523.62 , point in table: (mm, N)

33.5300 , 7236.90 , point in table: (mm, N)

43.6900 , 23521.0 , point in table: (mm, N)

55.8800 , 47647.0 , point in table: (mm, N)

63.7500 , 61818.3 , point in table: (mm, N)

72.9000 , 80517.7 , point in table: (mm, N)

ENDTABLE

SPRING_LOWER_ENVELOPE

12.9500 , 3318.21 , point in table: (mm, N)

31.7500 , 4221.15 , point in table: (mm, N)

35.3100 , 7236.90 , point in table: (mm, N)

43.1800 , 16889.1 , point in table: (mm, N)

57.6600 , 41917.9 , point in table: (mm, N)
 64.2600 , 57294.7 , point in table: (mm, N)
 72.9000 , 74486.2 , point in table: (mm, N)

ENDTABLE

* Steer at wheel vs. steering wheel angle (iside: 1=l, 2=r)

ISIDE 1 , Table ID number

STEERRW_TABLE Steer at wheel vs. steering wheel angle (ISIDE: 1=L, 2=R)

-511.320 , -29.6300 , point in table: (deg, deg)
 -464.220 , -28.0700 , point in table: (deg, deg)
 -396.020 , -25.3000 , point in table: (deg, deg)
 -327.820 , -22.2000 , point in table: (deg, deg)
 -280.720 , -19.3900 , point in table: (deg, deg)
 -243.410 , -17.2000 , point in table: (deg, deg)
 -164.290 , -12.0100 , point in table: (deg, deg)
 -79.1300 , -5.68000 , point in table: (deg, deg)
 .000000 , .000000 , point in table: (deg, deg)
 85.5600 , 6.60000 , point in table: (deg, deg)
 173.330 , 13.9600 , point in table: (deg, deg)
 241.530 , 19.3900 , point in table: (deg, deg)
 292.020 , 22.4200 , point in table: (deg, deg)
 344.020 , 26.4800 , point in table: (deg, deg)
 390.740 , 29.7600 , point in table: (deg, deg)
 438.600 , 32.7600 , point in table: (deg, deg)
 479.630 , 35.5300 , point in table: (deg, deg)

ENDTABLE

ISIDE 2 , Table ID number

STEERRW_TABLE Steer at wheel vs. steering wheel angle (ISIDE: 1=L, 2=R)

-514.330 , -36.4700 , point in table: (deg, deg)
 -459.700 , -33.9300 , point in table: (deg, deg)
 -396.020 , -30.3600 , point in table: (deg, deg)
 -366.630 , -28.2100 , point in table: (deg, deg)
 -301.820 , -23.7200 , point in table: (deg, deg)
 -224.950 , -17.6200 , point in table: (deg, deg)
 -182.370 , -14.7200 , point in table: (deg, deg)
 -79.1300 , -6.27000 , point in table: (deg, deg)
 .000000 , .000000 , point in table: (deg, deg)
 85.5300 , 6.23000 , point in table: (deg, deg)
 165.790 , 12.3900 , point in table: (deg, deg)
 248.310 , 17.6400 , point in table: (deg, deg)
 286.370 , 19.4300 , point in table: (deg, deg)
 344.020 , 22.4300 , point in table: (deg, deg)
 383.210 , 24.5300 , point in table: (deg, deg)

440.480 , 27.1500 , point in table: (deg, deg)
 478.910 , 28.5700 , point in table: (deg, deg)

ENDTABLE

* Tire Data:

IUNIT 1

IAXLE 1

* Longitudinal force (N) as a function of slip rate (-) and vertical load (N).

* top row = load, left column = slip, other cells = longitudinal tire force

FX_CARPET 8, columns in table

.000000	, 6867.71	, 13735.4	, 20598.7	, 27466.4	, 34334.1	, 41201.8	, 54932.8
.100000E-01	, 578.240	, 1178.72	, 1823.68	, 1814.78	, 2739.97	, 3491.68	, 3900.90
.200000E-01	, 1103.10	, 2793.34	, 4190.02	, 4492.48	, 5893.60	, 7192.42	, 8642.46
.300000E-01	, 1650.21	, 3576.19	, 5969.22	, 7179.07	, 8482.34	, 10684.1	, 11809.4
.400000E-01	, 2361.89	, 4688.19	, 7966.37	, 9002.75	, 11484.7	, 13455.2	, 15585.8
.500000E-01	, 2904.54	, 5742.37	, 9469.79	, 11182.3	, 13682.0	, 15705.9	, 19855.9
.600000E-01	, 3158.08	, 6756.51	, 10221.5	, 12877.0	, 14780.7	, 18010.0	, 22409.0
.700000E-01	, 3229.25	, 7365.89	, 11622.6	, 14384.8	, 17316.1	, 20278.4	, 25304.7
.800000E-01	, 3780.80	, 7801.79	, 13437.4	, 15848.2	, 18948.5	, 22409.0	, 27328.5
.900000E-01	, 4505.82	, 8767.01	, 13370.7	, 16733.4	, 20545.3	, 24717.5	, 30201.9
.100000	, 4425.76	, 9549.86	, 14233.6	, 18316.9	, 22151.0	, 26025.3	, 32176.8
.110000	, 4719.33	, 10408.3	, 14700.6	, 19282.1	, 23276.4	, 26937.1	, 33671.4
.120000	, 4808.29	, 11013.3	, 15132.1	, 19793.6	, 24366.1	, 27853.4	, 34738.9
.130000	, 4737.12	, 11249.0	, 15274.4	, 20959.0	, 24993.3	, 28654.0	, 35735.2
.140000	, 4723.78	, 11360.2	, 15256.6	, 21074.6	, 25504.8	, 29432.4	, 36375.7
.150000	, 4843.87	, 11613.7	, 15501.3	, 21301.5	, 25914.0	, 30290.9	, 36882.8
.160000	, 5030.69	, 11636.0	, 15790.4	, 21546.1	, 26238.8	, 30508.8	, 37296.5
.170000	, 5275.33	, 11600.4	, 15741.5	, 21701.8	, 26656.9	, 30624.5	, 38008.2
.180000	, 5519.97	, 11533.7	, 15741.5	, 21817.4	, 26830.3	, 30704.5	, 39240.3
.190000	, 5373.18	, 11409.1	, 15754.8	, 21928.6	, 27066.1	, 30735.7	, 39809.6
.200000	, 5377.63	, 11297.9	, 15728.1	, 22039.8	, 27150.6	, 30820.2	, 39894.1
.210000	, 5306.46	, 11204.5	, 15697.0	, 22142.1	, 27190.6	, 30931.4	, 39996.4
.220000	, 5279.78	, 11120.0	, 15670.3	, 22186.6	, 27155.0	, 30949.2	, 40018.7
.230000	, 5270.88	, 11035.5	, 15608.0	, 22244.5	, 27079.4	, 30940.3	, 39925.3
.240000	, 5235.30	, 10973.2	, 15536.9	, 22306.7	, 26994.9	, 30922.5	, 39809.6
.250000	, 5226.40	, 10910.9	, 15465.7	, 22369.0	, 26928.2	, 30864.7	, 39640.6
.300000	, 5457.70	, 10657.4	, 15127.7	, 22489.1	, 26616.8	, 30344.3	, 38644.2
.350000	, 5640.06	, 10443.9	, 14847.4	, 22271.1	, 26203.2	, 29712.6	, 37425.5
.400000	, 5991.46	, 10390.5	, 14607.2	, 21603.9	, 25647.2	, 29049.9	, 36233.4
.500000	, 5804.64	, 10501.7	, 14322.6	, 20167.2	, 24019.2	, 27444.2	, 34009.4
.600000	, 5044.03	, 10604.0	, 14131.3	, 18859.5	, 22311.2	, 25896.3	, 32132.3
.750000	, 4434.66	, 9563.20	, 13704.3	, 16884.6	, 20314.0	, 23836.8	, 29810.5

ENDTABLE

* Lateral force (N) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = lateral tire force

FY_CARPET 4, columns in table

.000000	, 8562.83	, 29358.3	, 41368.5
1.00000	, 1587.75	, 3715.96	, 4199.57
2.00000	, 2822.66	, 7166.49	, 7979.18
4.00000	, 4763.25	, 12475.0	, 15118.4
8.00000	, 6792.04	, 17518.1	, 23097.6
12.0000	, 8026.95	, 19376.1	, 25617.4

ENDTABLE

* Aligning moment (N-m) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = aligning moment

MZ_CARPET 6, columns in table

.000000	, 8896.44	, 17703.9	, 26555.9	, 35363.4	, 41991.2
1.00000	, 37.9090	, 115.190	, 199.305	, 280.654	, 338.900
2.00000	, 59.7100	, 193.936	, 356.634	, 520.634	, 634.523
4.00000	, 74.5160	, 254.948	, 490.860	, 759.312	, 972.121
8.00000	, 50.1110	, 195.238	, 366.071	, 599.217	, 802.590
12.0000	, 28.4720	, 123.325	, 246.813	, 403.979	, 521.936

ENDTABLE

* Tire Data:

IUNIT 1

IAXLE 2

* Longitudinal force (N) as a function of slip rate (-) and vertical load (N).

* top row = load, left column = slip, other cells = longitudinal tire force

FX_CARPET 8, columns in table

.000000	, 6867.71	, 13735.4	, 20598.7	, 27466.4	, 34334.1	, 41201.8	, 54932.8
.100000E-01	, 578.240	, 1178.72	, 1823.68	, 1814.78	, 2739.97	, 3491.68	, 3900.90
.200000E-01	, 1103.10	, 2793.34	, 4190.02	, 4492.48	, 5893.60	, 7192.42	, 8642.46
.300000E-01	, 1650.21	, 3576.19	, 5969.22	, 7179.07	, 8482.34	, 10684.1	, 11809.4
.400000E-01	, 2361.89	, 4688.19	, 7966.37	, 9002.75	, 11484.7	, 13455.2	, 15585.8
.500000E-01	, 2904.54	, 5742.37	, 9469.79	, 11182.3	, 13682.0	, 15705.9	, 19855.9
.600000E-01	, 3158.08	, 6756.51	, 10221.5	, 12877.0	, 14780.7	, 18010.0	, 22409.0
.700000E-01	, 3229.25	, 7365.89	, 11622.6	, 14384.8	, 17316.1	, 20278.4	, 25304.7
.800000E-01	, 3780.80	, 7801.79	, 13437.4	, 15848.2	, 18948.5	, 22409.0	, 27328.5
.900000E-01	, 4505.82	, 8767.01	, 13370.7	, 16733.4	, 20545.3	, 24717.5	, 30201.9
.100000	, 4425.76	, 9549.86	, 14233.6	, 18316.9	, 22151.0	, 26025.3	, 32176.8
.110000	, 4719.33	, 10408.3	, 14700.6	, 19282.1	, 23276.4	, 26937.1	, 33671.4
.120000	, 4808.29	, 11013.3	, 15132.1	, 19793.6	, 24366.1	, 27853.4	, 34738.9
.130000	, 4737.12	, 11249.0	, 15274.4	, 20959.0	, 24993.3	, 28654.0	, 35735.2
.140000	, 4723.78	, 11360.2	, 15256.6	, 21074.6	, 25504.8	, 29432.4	, 36375.7

.150000	, 4843.87	, 11613.7	, 15501.3	, 21301.5	, 25914.0	, 30290.9	, 36882.8
.160000	, 5030.69	, 11636.0	, 15790.4	, 21546.1	, 26238.8	, 30508.8	, 37296.5
.170000	, 5275.33	, 11600.4	, 15741.5	, 21701.8	, 26656.9	, 30624.5	, 38008.2
.180000	, 5519.97	, 11533.7	, 15741.5	, 21817.4	, 26830.3	, 30704.5	, 39240.3
.190000	, 5373.18	, 11409.1	, 15754.8	, 21928.6	, 27066.1	, 30735.7	, 39809.6
.200000	, 5377.63	, 11297.9	, 15728.1	, 22039.8	, 27150.6	, 30820.2	, 39894.1
.210000	, 5306.46	, 11204.5	, 15697.0	, 22142.1	, 27190.6	, 30931.4	, 39996.4
.220000	, 5279.78	, 11120.0	, 15670.3	, 22186.6	, 27155.0	, 30949.2	, 40018.7
.230000	, 5270.88	, 11035.5	, 15608.0	, 22244.5	, 27079.4	, 30940.3	, 39925.3
.240000	, 5235.30	, 10973.2	, 15536.9	, 22306.7	, 26994.9	, 30922.5	, 39809.6
.250000	, 5226.40	, 10910.9	, 15465.7	, 22369.0	, 26928.2	, 30864.7	, 39640.6
.300000	, 5457.70	, 10657.4	, 15127.7	, 22489.1	, 26616.8	, 30344.3	, 38644.2
.350000	, 5640.06	, 10443.9	, 14847.4	, 22271.1	, 26203.2	, 29712.6	, 37425.5
.400000	, 5991.46	, 10390.5	, 14607.2	, 21603.9	, 25647.2	, 29049.9	, 36233.4
.500000	, 5804.64	, 10501.7	, 14322.6	, 20167.2	, 24019.2	, 27444.2	, 34009.4
.600000	, 5044.03	, 10604.0	, 14131.3	, 18859.5	, 22311.2	, 25896.3	, 32132.3
.750000	, 4434.66	, 9563.20	, 13704.3	, 16884.6	, 20314.0	, 23836.8	, 29810.5

ENDTABLE

* Lateral force (N) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = lateral tire force

FY_CARPET 4, columns in table

.000000	, 8562.83	, 29358.3	, 41368.5
1.00000	, 1587.75	, 3715.96	, 4199.57
2.00000	, 2822.66	, 7166.49	, 7979.18
4.00000	, 4763.25	, 12475.0	, 15118.4
8.00000	, 6792.04	, 17518.1	, 23097.6
12.0000	, 8026.95	, 19376.1	, 25617.4

ENDTABLE

* Aligning moment (N-m) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = aligning moment

MZ_CARPET 6, columns in table

.000000	, 8896.44	, 17703.9	, 26555.9	, 35363.4	, 41991.2
1.00000	, 37.9090	, 115.190	, 199.305	, 280.654	, 338.900
2.00000	, 59.7100	, 193.936	, 356.634	, 520.634	, 634.523
4.00000	, 74.5160	, 254.948	, 490.860	, 759.312	, 972.121
8.00000	, 50.1110	, 195.238	, 366.071	, 599.217	, 802.590
12.0000	, 28.4720	, 123.325	, 246.813	, 403.979	, 521.936

ENDTABLE

* Tire Data:

IUNIT 1

IAXLE 3

* Longitudinal force (N) as a function of slip rate (-) and vertical load (N).

* top row = load, left column = slip, other cells = longitudinal tire force

FX_CARPET 8, columns in table

.000000	, 6867.71	, 13735.4	, 20598.7	, 27466.4	, 34334.1	, 41201.8	, 54932.8
.100000E-01	, 578.240	, 1178.72	, 1823.68	, 1814.78	, 2739.97	, 3491.68	, 3900.90
.200000E-01	, 1103.10	, 2793.34	, 4190.02	, 4492.48	, 5893.60	, 7192.42	, 8642.46
.300000E-01	, 1650.21	, 3576.19	, 5969.22	, 7179.07	, 8482.34	, 10684.1	, 11809.4
.400000E-01	, 2361.89	, 4688.19	, 7966.37	, 9002.75	, 11484.7	, 13455.2	, 15585.8
.500000E-01	, 2904.54	, 5742.37	, 9469.79	, 11182.3	, 13682.0	, 15705.9	, 19855.9
.600000E-01	, 3158.08	, 6756.51	, 10221.5	, 12877.0	, 14780.7	, 18010.0	, 22409.0
.700000E-01	, 3229.25	, 7365.89	, 11622.6	, 14384.8	, 17316.1	, 20278.4	, 25304.7
.800000E-01	, 3780.80	, 7801.79	, 13437.4	, 15848.2	, 18948.5	, 22409.0	, 27328.5
.900000E-01	, 4505.82	, 8767.01	, 13370.7	, 16733.4	, 20545.3	, 24717.5	, 30201.9
.100000	, 4425.76	, 9549.86	, 14233.6	, 18316.9	, 22151.0	, 26025.3	, 32176.8
.110000	, 4719.33	, 10408.3	, 14700.6	, 19282.1	, 23276.4	, 26937.1	, 33671.4
.120000	, 4808.29	, 11013.3	, 15132.1	, 19793.6	, 24366.1	, 27853.4	, 34738.9
.130000	, 4737.12	, 11249.0	, 15274.4	, 20959.0	, 24993.3	, 28654.0	, 35735.2
.140000	, 4723.78	, 11360.2	, 15256.6	, 21074.6	, 25504.8	, 29432.4	, 36375.7
.150000	, 4843.87	, 11613.7	, 15501.3	, 21301.5	, 25914.0	, 30290.9	, 36882.8
.160000	, 5030.69	, 11636.0	, 15790.4	, 21546.1	, 26238.8	, 30508.8	, 37296.5
.170000	, 5275.33	, 11600.4	, 15741.5	, 21701.8	, 26656.9	, 30624.5	, 38008.2
.180000	, 5519.97	, 11533.7	, 15741.5	, 21817.4	, 26830.3	, 30704.5	, 39240.3
.190000	, 5373.18	, 11409.1	, 15754.8	, 21928.6	, 27066.1	, 30735.7	, 39809.6
.200000	, 5377.63	, 11297.9	, 15728.1	, 22039.8	, 27150.6	, 30820.2	, 39894.1
.210000	, 5306.46	, 11204.5	, 15697.0	, 22142.1	, 27190.6	, 30931.4	, 39996.4
.220000	, 5279.78	, 11120.0	, 15670.3	, 22186.6	, 27155.0	, 30949.2	, 40018.7
.230000	, 5270.88	, 11035.5	, 15608.0	, 22244.5	, 27079.4	, 30940.3	, 39925.3
.240000	, 5235.30	, 10973.2	, 15536.9	, 22306.7	, 26994.9	, 30922.5	, 39809.6
.250000	, 5226.40	, 10910.9	, 15465.7	, 22369.0	, 26928.2	, 30864.7	, 39640.6
.300000	, 5457.70	, 10657.4	, 15127.7	, 22489.1	, 26616.8	, 30344.3	, 38644.2
.350000	, 5640.06	, 10443.9	, 14847.4	, 22271.1	, 26203.2	, 29712.6	, 37425.5
.400000	, 5991.46	, 10390.5	, 14607.2	, 21603.9	, 25647.2	, 29049.9	, 36233.4
.500000	, 5804.64	, 10501.7	, 14322.6	, 20167.2	, 24019.2	, 27444.2	, 34009.4
.600000	, 5044.03	, 10604.0	, 14131.3	, 18859.5	, 22311.2	, 25896.3	, 32132.3
.750000	, 4434.66	, 9563.20	, 13704.3	, 16884.6	, 20314.0	, 23836.8	, 29810.5

ENDTABLE

* Lateral force (N) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = lateral tire force

FY_CARPET 4, columns in table

.000000	, 8562.83	, 29358.3	, 41368.5
1.00000	, 1587.75	, 3715.96	, 4199.57
2.00000	, 2822.66	, 7166.49	, 7979.18
4.00000	, 4763.25	, 12475.0	, 15118.4

8.00000 , 6792.04 , 17518.1 , 23097.6
 12.0000 , 8026.95 , 19376.1 , 25617.4

ENDTABLE

* Aligning moment (N-m) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = aligning moment

MZ_CARPET 6, columns in table

.000000 , 8896.44 , 17703.9 , 26555.9 , 35363.4 , 41991.2
 1.00000 , 37.9090 , 115.190 , 199.305 , 280.654 , 338.900
 2.00000 , 59.7100 , 193.936 , 356.634 , 520.634 , 634.523
 4.00000 , 74.5160 , 254.948 , 490.860 , 759.312 , 972.121
 8.00000 , 50.1110 , 195.238 , 366.071 , 599.217 , 802.590
 12.0000 , 28.4720 , 123.325 , 246.813 , 403.979 , 521.936

ENDTABLE

* Tire Data:

IUNIT 2

IAXLE 1

* Longitudinal force (N) as a function of slip rate (-) and vertical load (N).

* top row = load, left column = slip, other cells = longitudinal tire force

FX_CARPET 8, columns in table

.000000 , 6867.71 , 13735.4 , 20598.7 , 27466.4 , 34334.1 , 41201.8 , 54932.8
 .100000E-01, 578.240 , 1178.72 , 1823.68 , 1814.78 , 2739.97 , 3491.68 , 3900.90
 .200000E-01, 1103.10 , 2793.34 , 4190.02 , 4492.48 , 5893.60 , 7192.42 , 8642.46
 .300000E-01, 1650.21 , 3576.19 , 5969.22 , 7179.07 , 8482.34 , 10684.1 , 11809.4
 .400000E-01, 2361.89 , 4688.19 , 7966.37 , 9002.75 , 11484.7 , 13455.2 , 15585.8
 .500000E-01, 2904.54 , 5742.37 , 9469.79 , 11182.3 , 13682.0 , 15705.9 , 19855.9
 .600000E-01, 3158.08 , 6756.51 , 10221.5 , 12877.0 , 14780.7 , 18010.0 , 22409.0
 .700000E-01, 3229.25 , 7365.89 , 11622.6 , 14384.8 , 17316.1 , 20278.4 , 25304.7
 .800000E-01, 3780.80 , 7801.79 , 13437.4 , 15848.2 , 18948.5 , 22409.0 , 27328.5
 .900000E-01, 4505.82 , 8767.01 , 13370.7 , 16733.4 , 20545.3 , 24717.5 , 30201.9
 .100000 , 4425.76 , 9549.86 , 14233.6 , 18316.9 , 22151.0 , 26025.3 , 32176.8
 .110000 , 4719.33 , 10408.3 , 14700.6 , 19282.1 , 23276.4 , 26937.1 , 33671.4
 .120000 , 4808.29 , 11013.3 , 15132.1 , 19793.6 , 24366.1 , 27853.4 , 34738.9
 .130000 , 4737.12 , 11249.0 , 15274.4 , 20959.0 , 24993.3 , 28654.0 , 35735.2
 .140000 , 4723.78 , 11360.2 , 15256.6 , 21074.6 , 25504.8 , 29432.4 , 36375.7
 .150000 , 4843.87 , 11613.7 , 15501.3 , 21301.5 , 25914.0 , 30290.9 , 36882.8
 .160000 , 5030.69 , 11636.0 , 15790.4 , 21546.1 , 26238.8 , 30508.8 , 37296.5
 .170000 , 5275.33 , 11600.4 , 15741.5 , 21701.8 , 26656.9 , 30624.5 , 38008.2
 .180000 , 5519.97 , 11533.7 , 15741.5 , 21817.4 , 26830.3 , 30704.5 , 39240.3
 .190000 , 5373.18 , 11409.1 , 15754.8 , 21928.6 , 27066.1 , 30735.7 , 39809.6
 .200000 , 5377.63 , 11297.9 , 15728.1 , 22039.8 , 27150.6 , 30820.2 , 39894.1
 .210000 , 5306.46 , 11204.5 , 15697.0 , 22142.1 , 27190.6 , 30931.4 , 39996.4
 .220000 , 5279.78 , 11120.0 , 15670.3 , 22186.6 , 27155.0 , 30949.2 , 40018.7

.230000	, 5270.88	, 11035.5	, 15608.0	, 22244.5	, 27079.4	, 30940.3	, 39925.3
.240000	, 5235.30	, 10973.2	, 15536.9	, 22306.7	, 26994.9	, 30922.5	, 39809.6
.250000	, 5226.40	, 10910.9	, 15465.7	, 22369.0	, 26928.2	, 30864.7	, 39640.6
.300000	, 5457.70	, 10657.4	, 15127.7	, 22489.1	, 26616.8	, 30344.3	, 38644.2
.350000	, 5640.06	, 10443.9	, 14847.4	, 22271.1	, 26203.2	, 29712.6	, 37425.5
.400000	, 5991.46	, 10390.5	, 14607.2	, 21603.9	, 25647.2	, 29049.9	, 36233.4
.500000	, 5804.64	, 10501.7	, 14322.6	, 20167.2	, 24019.2	, 27444.2	, 34009.4
.600000	, 5044.03	, 10604.0	, 14131.3	, 18859.5	, 22311.2	, 25896.3	, 32132.3
.750000	, 4434.66	, 9563.20	, 13704.3	, 16884.6	, 20314.0	, 23836.8	, 29810.5

ENDTABLE

* Lateral force (N) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = lateral tire force

FY_CARPET 4, columns in table

.000000	, 8562.83	, 29358.3	, 41368.5
1.00000	, 1587.75	, 3715.96	, 4199.57
2.00000	, 2822.66	, 7166.49	, 7979.18
4.00000	, 4763.25	, 12475.0	, 15118.4
8.00000	, 6792.04	, 17518.1	, 23097.6
12.0000	, 8026.95	, 19376.1	, 25617.4

ENDTABLE

* Aligning moment (N-m) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = aligning moment

MZ_CARPET 6, columns in table

.000000	, 8896.44	, 17703.9	, 26555.9	, 35363.4	, 41991.2
1.00000	, 37.9090	, 115.190	, 199.305	, 280.654	, 338.900
2.00000	, 59.7100	, 193.936	, 356.634	, 520.634	, 634.523
4.00000	, 74.5160	, 254.948	, 490.860	, 759.312	, 972.121
8.00000	, 50.1110	, 195.238	, 366.071	, 599.217	, 802.590
12.0000	, 28.4720	, 123.325	, 246.813	, 403.979	, 521.936

ENDTABLE

* Tire Data:

IUNIT 2

IAXLE 2

* Longitudinal force (N) as a function of slip rate (-) and vertical load (N).

* top row = load, left column = slip, other cells = longitudinal tire force

FX_CARPET 8, columns in table

.000000	, 6867.71	, 13735.4	, 20598.7	, 27466.4	, 34334.1	, 41201.8	, 54932.8
.100000E-01	, 578.240	, 1178.72	, 1823.68	, 1814.78	, 2739.97	, 3491.68	, 3900.90
.200000E-01	, 1103.10	, 2793.34	, 4190.02	, 4492.48	, 5893.60	, 7192.42	, 8642.46
.300000E-01	, 1650.21	, 3576.19	, 5969.22	, 7179.07	, 8482.34	, 10684.1	, 11809.4

.400000E-01,	2361.89	, 4688.19	, 7966.37	, 9002.75	, 11484.7	, 13455.2	, 15585.8
.500000E-01,	2904.54	, 5742.37	, 9469.79	, 11182.3	, 13682.0	, 15705.9	, 19855.9
.600000E-01,	3158.08	, 6756.51	, 10221.5	, 12877.0	, 14780.7	, 18010.0	, 22409.0
.700000E-01,	3229.25	, 7365.89	, 11622.6	, 14384.8	, 17316.1	, 20278.4	, 25304.7
.800000E-01,	3780.80	, 7801.79	, 13437.4	, 15848.2	, 18948.5	, 22409.0	, 27328.5
.900000E-01,	4505.82	, 8767.01	, 13370.7	, 16733.4	, 20545.3	, 24717.5	, 30201.9
.100000	, 4425.76	, 9549.86	, 14233.6	, 18316.9	, 22151.0	, 26025.3	, 32176.8
.110000	, 4719.33	, 10408.3	, 14700.6	, 19282.1	, 23276.4	, 26937.1	, 33671.4
.120000	, 4808.29	, 11013.3	, 15132.1	, 19793.6	, 24366.1	, 27853.4	, 34738.9
.130000	, 4737.12	, 11249.0	, 15274.4	, 20959.0	, 24993.3	, 28654.0	, 35735.2
.140000	, 4723.78	, 11360.2	, 15256.6	, 21074.6	, 25504.8	, 29432.4	, 36375.7
.150000	, 4843.87	, 11613.7	, 15501.3	, 21301.5	, 25914.0	, 30290.9	, 36882.8
.160000	, 5030.69	, 11636.0	, 15790.4	, 21546.1	, 26238.8	, 30508.8	, 37296.5
.170000	, 5275.33	, 11600.4	, 15741.5	, 21701.8	, 26656.9	, 30624.5	, 38008.2
.180000	, 5519.97	, 11533.7	, 15741.5	, 21817.4	, 26830.3	, 30704.5	, 39240.3
.190000	, 5373.18	, 11409.1	, 15754.8	, 21928.6	, 27066.1	, 30735.7	, 39809.6
.200000	, 5377.63	, 11297.9	, 15728.1	, 22039.8	, 27150.6	, 30820.2	, 39894.1
.210000	, 5306.46	, 11204.5	, 15697.0	, 22142.1	, 27190.6	, 30931.4	, 39996.4
.220000	, 5279.78	, 11120.0	, 15670.3	, 22186.6	, 27155.0	, 30949.2	, 40018.7
.230000	, 5270.88	, 11035.5	, 15608.0	, 22244.5	, 27079.4	, 30940.3	, 39925.3
.240000	, 5235.30	, 10973.2	, 15536.9	, 22306.7	, 26994.9	, 30922.5	, 39809.6
.250000	, 5226.40	, 10910.9	, 15465.7	, 22369.0	, 26928.2	, 30864.7	, 39640.6
.300000	, 5457.70	, 10657.4	, 15127.7	, 22489.1	, 26616.8	, 30344.3	, 38644.2
.350000	, 5640.06	, 10443.9	, 14847.4	, 22271.1	, 26203.2	, 29712.6	, 37425.5
.400000	, 5991.46	, 10390.5	, 14607.2	, 21603.9	, 25647.2	, 29049.9	, 36233.4
.500000	, 5804.64	, 10501.7	, 14322.6	, 20167.2	, 24019.2	, 27444.2	, 34009.4
.600000	, 5044.03	, 10604.0	, 14131.3	, 18859.5	, 22311.2	, 25896.3	, 32132.3
.750000	, 4434.66	, 9563.20	, 13704.3	, 16884.6	, 20314.0	, 23836.8	, 29810.5

ENDTABLE

* Lateral force (N) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = lateral tire force

FY_CARPET 4, columns in table

.000000	, 8562.83	, 29358.3	, 41368.5
1.00000	, 1587.75	, 3715.96	, 4199.57
2.00000	, 2822.66	, 7166.49	, 7979.18
4.00000	, 4763.25	, 12475.0	, 15118.4
8.00000	, 6792.04	, 17518.1	, 23097.6
12.0000	, 8026.95	, 19376.1	, 25617.4

ENDTABLE

* Aligning moment (N-m) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = aligning moment

MZ_CARPET 6, columns in table


```
.000000 , 8896.44 , 17703.9 , 26555.9 , 35363.4 , 41991.2
1.00000 , 37.9090 , 115.190 , 199.305 , 280.654 , 338.900
2.00000 , 59.7100 , 193.936 , 356.634 , 520.634 , 634.523
4.00000 , 74.5160 , 254.948 , 490.860 , 759.312 , 972.121
8.00000 , 50.1110 , 195.238 , 366.071 , 599.217 , 802.590
12.0000 , 28.4720 , 123.325 , 246.813 , 403.979 , 521.936
```

ENDTABLE

* Vertical Geometry Data: Station, Z coord., Vert. Curv. Len, Back Grade, Fwd Grade

VGDATA

```
.000000 , 47.7400 , .000000 , -.265000E-01, -.265000E-01
303.260 , 39.7040 , 70.0000 , -.265000E-01, .800000E-02
373.641 , 39.0590 , .000000 , .800000E-02, .800000E-02
645.750 , 41.2360 , 50.0000 , .800000E-02, -.115000E-01
699.452 , 41.1060 , .000000 , -.115000E-01, -.115000E-01
1153.65 , 35.8830 , 90.0000 , -.115000E-01, .290000E-01
1261.24 , 37.1800 , .000000 , .290000E-01, .290000E-01
1652.04 , 48.5130 , 120.000 , .290000E-01, -.343300E-01
1789.57 , 47.5920 , .000000 , -.343300E-01, -.343300E-01
```

ENDTABLE

* FINAL CONDITIONS

```
XO      55771.35787091 , Tractor sprung mass origin absolute X position (m)
YO      116787.3403443 , Tractor sprung mass origin absolute Y position (m)
ZO      43.82715396957 , Tractor sprung mass origin absolute Z position (m)
YAW     -58.75835440121 , Tractor sprung mass absolute yaw (deg)
PITCH   2.038651334340 , Tractor sprung mass absolute pitch (deg)
ROLL    1.343996084032 , Tractor sprung mass absolute roll (deg)
PITCH_2 -.7449237820432E-01, Semitrailer rel. pitch (deg)
ROLL_H1 .1631424972993E-02, Semitrailer rel. roll (deg)
ART_H1  .2108807738760 , Semitrailer articulation (deg)
ROLL_A4 .1943093998272 , Axle 4 roll (deg)
JNC_A4  -.1700991645632E-03, Axle 4 jounce (m)
ROLL_A5 .1943153897836 , Axle 5 roll (deg)
JNC_A5  -.6560656140397E-03, Axle 5 jounce (m)
ROLL_A1 .1658086310419 , Axle 1 roll (deg)
JNC_A1  .4925101881018E-02, Axle 1 jounce (m)
ROLL_A2 .1470148012704 , Axle 2 roll (deg)
JNC_A2  -.2087591711715E-03, Axle 2 jounce (m)
ROLL_A3 .1485824964480 , Axle 3 roll (deg)
JNC_A3  -.1112852177152E-02, Axle 3 jounce (m)
FSL1    22522.96970261 , Spring force, L side, axle 1 (N)
FSR1    23072.88091430 , Spring force, R side, axle 1 (N)
FSL2    30726.81615165 , Spring force, L side, axle 2 (N)
FSR2    33339.80107058 , Spring force, R side, axle 2 (N)
```

FSL3	29763.80840360	, Spring force, L side, axle 3 (N)
FSR3	32294.03163929	, Spring force, R side, axle 3 (N)
FSL4	31596.44159827	, Spring force, L side, axle 4 (N)
FSR4	36111.98070630	, Spring force, R side, axle 4 (N)
FSL5	30301.83106755	, Spring force, L side, axle 5 (N)
FSR5	34785.26490144	, Spring force, R side, axle 5 (N)
ROTL1	629.9281101125	, L wheel, axle 1 rotation (rev)
ROTR1	629.6160596972	, R wheel, axle 1 rotation (rev)
ROTL2	626.7996147762	, L wheel, axle 2 rotation (rev)
ROTR2	626.9465060691	, R wheel, axle 2 rotation (rev)
ROTL3	626.2134580206	, L wheel, axle 3 rotation (rev)
ROTR3	626.3448830816	, R wheel, axle 3 rotation (rev)
ROTL4	552.9570843478	, L wheel, axle 4 rotation (rev)
ROTR4	552.6695698525	, R wheel, axle 4 rotation (rev)
ROTL5	552.9841153231	, L wheel, axle 5 rotation (rev)
ROTR5	552.6980860553	, R wheel, axle 5 rotation (rev)
TANAL1	-.2409841793166E-02	, Tan of lagged slip angle, L wheel, axle 1 (-)
TANAR1	-.2907904293846E-02	, Tan of lagged slip angle, R wheel, axle 1 (-)
TANAL2	-.1656749102767E-02	, Tan of lagged slip angle, L wheel, axle 2 (-)
TANAR2	-.1656726111016E-02	, Tan of lagged slip angle, R wheel, axle 2 (-)
TANAL3	-.1673154466890E-02	, Tan of lagged slip angle, L wheel, axle 3 (-)
TANAR3	-.1673142443965E-02	, Tan of lagged slip angle, R wheel, axle 3 (-)
TANAL4	-.2506843590939E-02	, Tan of lagged slip angle, L wheel, axle 4 (-)
TANAR4	-.2506839118405E-02	, Tan of lagged slip angle, R wheel, axle 4 (-)
TANAL5	-.2502048997907E-02	, Tan of lagged slip angle, L wheel, axle 5 (-)
TANAR5	-.2502044071983E-02	, Tan of lagged slip angle, R wheel, axle 5 (-)
PBRK1	.000000000000	, Air pressure in chamber for wheel L1 (kPa)
PBRK6	.000000000000	, Air pressure in chamber for wheel R1 (kPa)
PBRK2	.000000000000	, Air pressure in chamber for wheel L2 (kPa)
PBRK7	.000000000000	, Air pressure in chamber for wheel R2 (kPa)
PBRK3	.000000000000	, Air pressure in chamber for wheel L3 (kPa)
PBRK8	.000000000000	, Air pressure in chamber for wheel R3 (kPa)
PBRK4	.000000000000	, Air pressure in chamber for wheel L4 (kPa)
PBRK9	.000000000000	, Air pressure in chamber for wheel R4 (kPa)
PBRK5	.000000000000	, Air pressure in chamber for wheel L5 (kPa)
PBRK10	.000000000000	, Air pressure in chamber for wheel R5 (kPa)
IVERR	-.3471604229264	, Integral of velocity error for speed controller (m)
VXS	50.00051324192	, Tractor sprung mass CG absolute X trans. speed (body fixed) (kph)
VYS	-.6340096524478E-01	, Tractor sprung mass CG absolute Y trans. speed (body fixed) (kph)
VZS	.6844214623910E-01	, Tractor sprung mass CG absolute Z trans. speed (body fixed) (kph)
AVZ	.6872461491882E-02	, Tractor sprung mass absolute Z angular speed (body fixed) (deg/s)
AVY	-.6262367893600E-01	, Tractor sprung mass absolute Y angular speed (body fixed) (deg/s)
AVX	-.2901783420474E-01	, Tractor sprung mass absolute X angular speed (body fixed) (deg/s)
AVY_H2	.2265046976596E-01	, Semitrailer rel. pitch rate (deg/s)
AVX_H1	-.1729878665359E-01	, Semitrailer rel. roll rate (deg/s)
AVZ_H1	-.1046363375066E-02	, Semitrailer rel. yaw rate (deg/s)

AVX_A4 -.2720070437983E-01, Axle 4 roll rate (deg/s)
 JNCR_A4 -.3289225745986E-01, Axle 4 jounce rate (m/s)
 AVX_A5 -.2506585271723E-01, Axle 5 roll rate (deg/s)
 JNCR_A5 .2609294414540E-01, Axle 5 jounce rate (m/s)
 AVX_A1 -.3072170683400E-01, Axle 1 roll rate (deg/s)
 JNCR_A1 -.3293115672569E-02, Axle 1 jounce rate (m/s)
 AVX_A2 -.2386883661171E-01, Axle 2 roll rate (deg/s)
 JNCR_A2 .6522498938085E-03, Axle 2 jounce rate (m/s)
 AVX_A3 -.2327973636999E-01, Axle 3 roll rate (deg/s)
 JNCR_A3 .1727028213046E-02, Axle 3 jounce rate (m/s)
 AVy_L1 4.605146560981 , L wheel, axle 1 spin (rev/s)
 AVy_R1 4.605222750290 , R wheel, axle 1 spin (rev/s)
 AVy_L2 4.559302792791 , L wheel, axle 2 spin (rev/s)
 AVy_R2 4.562712908846 , R wheel, axle 2 spin (rev/s)
 AVy_L3 4.557845580047 , L wheel, axle 3 spin (rev/s)
 AVy_R3 4.561398125409 , R wheel, axle 3 spin (rev/s)
 AVy_L4 4.048416059060 , L wheel, axle 4 spin (rev/s)
 AVy_R4 4.048551022984 , R wheel, axle 4 spin (rev/s)
 AVy_L5 4.048415426507 , L wheel, axle 5 spin (rev/s)
 AVy_R5 4.048547817003 , R wheel, axle 5 spin (rev/s)

END

Keywords for the Car Model

Listing A.1 shows the complete set of model parameters for the car model in VDM RoAD.

Listing D.10. LPF file for a car.

PARSFILE

* VDM RoAD Car Model.
 * Generated by AutoSim October 14, 1997.
 * Portions Copyright 1996, 1997. Mechanical Simulation Corporation.
 * Portions Copyright 1993-1997. The Regents of The University of Michigan.
 * All rights reserved.
 *

TITLE P (Alt 3 R lane)

* Input File: C:\VDM_ROAD\RUNS\541.LPI
 * Run was made 20:56 on Oct 15, 1997

FORMAT

INTOPT 3, Type of integration (0=Adams, 1=Gear+PEDERV, 2=Gear diff., 3=RK2) (-)
 IPRINT 40, number of time steps between output printing (counts)

EPSINT .100000E-01 , Dimensionless integrator error tolerance (-)
 STARTT .000000 , simulation start time (s)
 STEP .250000E-02 , simulation time step (s)
 STEP0 .100000E-04 , Initial integration step (s)
 STEPMIN .100000E-06 , Minimum allowable integration step (s)
 STOPT 100.530 , simulation stop time (s)

* PARAMETER VALUES

ACSAERO 2.23000 , Area of vehicle cross-section (for aerodynamics) (m2)
 CDAERO .440000 , Aero drag coef: $F_x = -Q \cdot A \cdot (C_{daero} + K_{daero} \cdot \beta^{**2})$ (-)
 CLAERO .000000 , Aero lift coef: $f_z = -Q \cdot A \cdot (C_{laero} + K_{laero} \cdot \beta^{**2})$ (-)
 CMAERO .000000 , Aero pitch coef: $M_y = Q \cdot A \cdot L_{wb} \cdot (C_{maero} + K_{maero} \cdot \beta^{**2})$ (-)
 CSFY(1) .000000 , Compliance coef. for steer at front axle: $d(\text{steer})/d(F_y)$ (deg/N)
 CSFY(2) .000000 , Compliance coef. for steer at rear axle: $d(\text{steer})/d(F_y)$ (deg/N)
 CSMZ(1) .000000 , Compliance coef. for steer at front axle: $d(\text{steer})/d(M_z)$ (deg/N/m)
 CSMZ(2) .000000 , Compliance coef. for steer at rear axle: $d(\text{steer})/d(M_z)$ (deg/N/m)
 CSMZF .100000E-02 , Compliance coef. of steering system: $d(\text{steer})/d(M_{zl} + M_{zr})/2$ (deg/N/m)
 CTFX(1) .000000 , Compliance coef. for toe at front axle: $d(\text{toe})/d(F_x)$ (deg/N)
 CTFX(2) .000000 , Compliance coef. for toe at rear axle: $d(\text{toe})/d(F_x)$ (deg/N)
 DAIR .000000 , Air density (kg/m3)
 HCG 1320.00 , Height of entire vehicle C.G. above the ground (mm)
 HCGS 1427.81 , CALC - Height of sprung-mass CG above ground (mm)
 HRC(1) .000000 , Height of front axle roll center above the ground (mm)
 HRC(2) .000000 , Height of rear axle roll center above the ground (mm)
 HWC(1) 460.000 , Height of front wheel center at design load (mm)
 HWC(2) 460.000 , Height of rear wheel center at design load (mm)
 IW(1) 2.00000 , Spin moment of inertia of front wheel (kg-m2)
 IW(2) 2.00000 , Spin moment of inertia of rear wheel (kg-m2)
 IXX 1669.00 , Moment of inertia of entire vehicle (kg-m2)
 IXXS 938.418 , CALC - Sprung-mass IXX moment of inertia (kg-m2)
 IXZ .000000 , Product of inertia of entire vehicle (kg-m2)
 IXZS -183.217 , CALC - Sprung-mass IXZ product of inertia (kg-m2)
 IYY 6172.00 , Moment of inertia of entire vehicle (kg-m2)
 IYYS 4516.22 , CALC - Sprung-mass IYY moment of inertia (kg-m2)
 IZZ 6172.00 , Moment of inertia of entire vehicle (kg-m2)
 IZZS 4518.07 , CALC - Sprung-mass IZZ moment of inertia (kg-m2)
 KAUX(1) 30.0000 , Front auxiliary stiffness, including anti-sway bar (N-m/deg)
 KAUX(2) 30.0000 , Rear auxiliary stiffness, including anti-sway bar (N-m/deg)
 KDAERO .000000 , Aero drag coef: $F_x = -Q \cdot A \cdot (C_{daero} + K_{daero} \cdot \beta^{**2})$ (1/deg2)
 KGAMMA(1) .000000 , Front tire camber stiffness (N/deg)
 KGAMMA(2) .000000 , Rear tire camber stiffness (N/deg)
 KLAERO .000000 , Aero lift coef: $F_z = -Q \cdot A \cdot (C_{laero} + K_{laero} \cdot \beta^{**2})$ (1/deg2)
 KMAERO .000000 , Aero pitch coef: $M_y = Q \cdot A \cdot L_{wb} \cdot (C_{maero} + K_{maero} \cdot \beta^{**2})$ (1/deg2)
 KNAERO .000000 , Aero yaw coef: $M_z = -Q \cdot A \cdot L_{wb} \cdot K_{naero} \cdot \beta$ (1/deg)
 KRAERO .000000 , Aero roll coef: $M_x = -Q \cdot A \cdot L_{wb} \cdot K_{raero} \cdot \beta$ (1/deg)

KT(1)	220.000	, Front tire vertical stiffness (for 1 tire) (N/mm)
KT(2)	220.000	, Rear tire vertical stiffness (for 1 tire) (N/mm)
KYAERO	.000000	, Aero side force coef: $F_y = -Q \cdot A \cdot K_{yaero} \cdot \text{Beta}$ (1/deg)
LCGS	2134.19	, CALC - Distance from front axle to sprung-mass CG (mm)
LCGT	2080.25	, CALC - Distance from F axle to total vehicle CG (mm)
LCPF	20.0000	, Distance front camera point is in front of front axle (m)
LCPR	20.0000	, Distance rear camera point is behind rear axle (m)
LRELAX(1)	630.000	, Front tire relaxation length (mm)
LRELAX(2)	630.000	, Rear tire relaxation length (mm)
LTK(1)	1820.00	, Front axle track width (mm)
LTK(2)	1820.00	, Rear axle track width (mm)
LWB	3300.00	, Wheelbase (mm)
MF	1460.00	, Vehicle mass supported by front axle (2 wheels) (kg)
MR	2490.00	, Vehicle mass supported by rear axle (2 wheels) (kg)
MS	3510.00	, CALC - Mass of sprung mass (kg)
MT	3950.00	, CALC - Total vehicle mass (kg)
MU	.750000	, Tire-road coefficient of friction (-)
MUS(1)	220.000	, Front axle unsprung mass (2 wheels) (kg)
MUS(2)	220.000	, Rear axle unsprung mass (2 wheels) (kg)
PSIWND	.000000	, Heading of wind (0 deg=tailwind) (deg)
RAP(1)	.000000	, Ratio: wheelbase change per unit jounce at front axle (-)
RAP(2)	.000000	, Ratio: wheelbase change per unit jounce at rear axle (-)
RDAMP(1)	.550000	, Ratio: front jounce at wheel to damper stroke (-)
RDAMP(2)	.550000	, Ratio: rear jounce at wheel to damper stroke (-)
RDRIVE	.000000	, Ratio: rear drive torque to total: 0 = FWD, 0.5 = 4WD, 1 = RWD (-)
RMF	.369620	, CALC - Ratio: proportion of load on front axle (-)
RMR	.630380	, CALC - Ratio: proportion of load on rear axle (-)
ROLL_STOP	45.0000	, Roll limit for stopping the simulation (deg)
RSPRNG(1)	.550000	, Front ratio of jounce at wheel to spring jounce (-)
RSPRNG(2)	.550000	, Rear ratio of jounce at wheel to spring jounce (-)
RSW	15.9700	, Steering gear ratio (-)
RTIME	.646573	, CALC -- Computational efficiency (sec/sim. sec) (-)
SPEED_KI	.157000	, Integral control gain for speed controller (rev/s ²)
SPEED_KP	.141000	, Proportional control gain for speed controller (rev/s)
STARTS	.000000	, Starting station number (beginning of simulation) (m)
STATION	1900.04	, Station number -- distance along input path (m)
STOPS	1900.00	, Stopping station number (stop simulation when this is reached) (m)
TDLAG	.000000	, Lag time used by driver model (s)
TPREV	.200000	, Preview time used by driver model (s)
VLOW_ALPHA(1)	15.0000	, Front low-speed threshold for modified tire relaxation equations (kph)
VLOW_ALPHA(2)	15.0000	, Rear low-speed threshold for modified tire relaxation equations (kph)
VLOW_KAPPA(1)	8.00000	, Front low-speed threshold for modified longitudinal slip equations (kph)
VLOW_KAPPA(2)	8.00000	, Rear low-speed threshold for modified longitudinal slip equations (kph)
VLOW_SPINA(1)	10.0000	, Front low-speed threshold for modified wheel spin equations (kph)
VLOW_SPINA(2)	10.0000	, Rear low-speed threshold for modified wheel spin equations (kph)
V_STOP	-1.00000	, Low-speed limit for stopping the simulation (kph)

```
BRKIN_TABLE Brake input vs. time
.000000 , .000000 , point in table: (s, kPa)
2.000000 , .000000 , point in table: (s, kPa)
ENDTABLE
```

* Camber angle vs. susp jounce (iaxe: 1=f, 2=r)

```
IAXLE 1      , Table ID number
CAMBER_TABLE Camber angle vs. susp jounce (IAXLE: 1=F, 2=R)
-70.0000 , .430000 , point in table: (mm, deg)
-60.0000 , .440000 , point in table: (mm, deg)
-50.0000 , .420000 , point in table: (mm, deg)
-40.0000 , .380000 , point in table: (mm, deg)
-30.0000 , .310000 , point in table: (mm, deg)
-20.0000 , .230000 , point in table: (mm, deg)
-10.0000 , .130000 , point in table: (mm, deg)
.000000 , .000000 , point in table: (mm, deg)
10.0000 , -.150000 , point in table: (mm, deg)
20.0000 , -.320000 , point in table: (mm, deg)
30.0000 , -.510000 , point in table: (mm, deg)
40.0000 , -.720000 , point in table: (mm, deg)
50.0000 , -.950000 , point in table: (mm, deg)
60.0000 , -1.21000 , point in table: (mm, deg)
70.0000 , -1.49000 , point in table: (mm, deg)
ENDTABLE
```

```
IAXLE 2      , Table ID number
CAMBER_TABLE Camber angle vs. susp jounce (IAXLE: 1=F, 2=R)
-70.0000 , .430000 , point in table: (mm, deg)
-60.0000 , .440000 , point in table: (mm, deg)
-50.0000 , .420000 , point in table: (mm, deg)
-40.0000 , .380000 , point in table: (mm, deg)
-30.0000 , .310000 , point in table: (mm, deg)
-20.0000 , .230000 , point in table: (mm, deg)
-10.0000 , .130000 , point in table: (mm, deg)
.000000 , .000000 , point in table: (mm, deg)
10.0000 , -.150000 , point in table: (mm, deg)
20.0000 , -.320000 , point in table: (mm, deg)
30.0000 , -.510000 , point in table: (mm, deg)
40.0000 , -.720000 , point in table: (mm, deg)
50.0000 , -.950000 , point in table: (mm, deg)
60.0000 , -1.21000 , point in table: (mm, deg)
70.0000 , -1.49000 , point in table: (mm, deg)
ENDTABLE
```

* Cross Slope Data: Station, DZDL, LAT, DZDL, ... , LAT, DZDL

* where LAT = lateral position (left, m), DZDL = cross-slope

* for lane (rise going left = positive, %).

CROSS_SLOPE_TABLE 4

.000000	, 2.00000	, .000000	, -2.00000
253.257	, 2.00000	, .000000	, -2.00000
277.836	, -2.00000	, .000000	, -2.00000
283.059	, -2.85000	, .000000	, -2.85000
303.260	, -6.13800	, .000000	, -6.13800
312.861	, -7.70000	, .000000	, -7.70000
373.641	, -7.70000	, .000000	, -7.70000
403.443	, -5.41200	, .000000	, -5.41200
473.938	, .000000	, .000000	, .000000
545.427	, 5.48800	, .000000	, 5.48800
575.537	, 7.80000	, .000000	, 7.80000
645.750	, 7.80000	, .000000	, 7.80000
699.452	, 7.80000	, .000000	, 7.80000
729.561	, 4.56300	, .000000	, 4.56300
772.013	, .000000	, .000000	, .000000
815.710	, -4.69700	, .000000	, -4.69700
846.434	, -8.00000	, .000000	, -8.00000
958.065	, -8.00000	, .000000	, -8.00000
988.789	, -5.06300	, .000000	, -5.06300
1041.75	, .000000	, .000000	, .000000
1094.71	, 5.06300	, .000000	, 5.06300
1125.43	, 8.00000	, .000000	, 8.00000
1153.65	, 8.00000	, .000000	, 8.00000
1230.52	, 8.00000	, .000000	, 8.00000
1261.24	, 5.53200	, .000000	, 5.53200
1330.09	, .000000	, .000000	, .000000
1398.95	, -5.53200	, .000000	, -5.53200
1429.67	, -8.00000	, .000000	, -8.00000
1516.21	, -8.00000	, .000000	, -8.00000
1546.94	, -5.15200	, .000000	, -5.15200
1602.53	, .000000	, .000000	, .000000
1652.04	, 4.58900	, .000000	, 4.58900
1658.12	, 5.15200	, .000000	, 5.15200
1688.84	, 8.00000	, .000000	, 8.00000
1789.57	, 8.00000	, .000000	, 8.00000
1820.29	, 3.00000	, .000000	, 3.00000
1826.43	, 2.00000	, .000000	, 2.00000
1851.01	, 2.00000	, .000000	, -2.00000

ENDTABLE

* Susp comp damping force (1 side) vs. jnc rate (iaxle:1=f,2=r)

```

IAXLE 1      , Table ID number
FD_TABLE Susp comp damping force (1 side) vs. jnc rate (IAXLE:1=F,2=R)
.000000 , .000000 , point in table: (mm/s, N)
1.000000 , 5.000000 , point in table: (mm/s, N)
ENDTABLE

```

```

IAXLE 2      , Table ID number
FD_TABLE Susp comp damping force (1 side) vs. jnc rate (IAXLE:1=F,2=R)
.000000 , .000000 , point in table: (mm/s, N)
1.000000 , 5.000000 , point in table: (mm/s, N)
ENDTABLE

```

* Susp comp spring force (1 side) vs. jounce (iaxle: 1=f,2=r)

```

IAXLE 1      , Table ID number
FS_TABLE Susp comp spring force (1 side) vs. jounce (IAXLE: 1=F,2=R)
.000000 , .000000 , point in table: (mm, N)
1.000000 , 170.000 , point in table: (mm, N)
ENDTABLE

```

```

IAXLE 2      , Table ID number
FS_TABLE Susp comp spring force (1 side) vs. jounce (IAXLE: 1=F,2=R)
.000000 , .000000 , point in table: (mm, N)
1.000000 , 370.000 , point in table: (mm, N)
ENDTABLE

```

* Horizontal Geometry Data: Station, Radius, Spiral Par., X, Y, Heading

```

HGDATA
.000000 , .000000 , .000000 , 54156.3 , 117321. , -29.5000
283.059 , -155.000 , .000000 , 54402.7 , 117182. , -29.5000
403.443 , .000000 , .000000 , 54519.1 , 117167. , 15.0003
545.427 , 150.000 , .000000 , 54656.2 , 117204. , 15.0003
729.561 , .000000 , .000000 , 54818.4 , 117144. , -55.3339
815.710 , -125.000 , .000000 , 54867.4 , 117073. , -55.3339
988.789 , .000000 , .000000 , 55021.1 , 117030. , 23.9997
1094.71 , 125.000 , .000000 , 55117.9 , 117073. , 23.9997
1261.24 , .000000 , .000000 , 55267.6 , 117035. , -52.3333
1398.95 , -125.000 , .000000 , 55351.8 , 116926. , -52.3333
1546.94 , .000000 , .000000 , 55484.1 , 116882. , 15.5002
1658.12 , 125.000 , .000000 , 55591.3 , 116912. , 15.5002
1820.29 , .000000 , .000000 , 55731.5 , 116856. , -58.8332
ENDTABLE

```

```

LTARG_TABLE Lateral offset target (+ = left, - = right) as function of station.
.000000 , -1.65000 , point in table: (m, m)

```


1.00000 , -1.65000 , point in table: (m, m)
 ENDTABLE

IAXLE 1

ISIDE 1

MYBK_TABLE Brake torque at wheel vs. line pressure

.000000 , .000000 , point in table: (kPa, N-m)
 46.5000 , 376.569 , point in table: (kPa, N-m)
 60.3000 , 625.300 , point in table: (kPa, N-m)
 76.9000 , 749.665 , point in table: (kPa, N-m)
 83.9700 , 878.199 , point in table: (kPa, N-m)
 97.7000 , 1034.52 , point in table: (kPa, N-m)
 101.600 , 1074.82 , point in table: (kPa, N-m)
 104.900 , 1081.07 , point in table: (kPa, N-m)
 110.900 , 1116.51 , point in table: (kPa, N-m)
 137.900 , 1406.92 , point in table: (kPa, N-m)
 168.700 , 1672.33 , point in table: (kPa, N-m)

ENDTABLE

IAXLE 2

ISIDE 1

MYBK_TABLE Brake torque at wheel vs. line pressure

.000000 , .000000 , point in table: (kPa, N-m)
 47.0000 , 92.0000 , point in table: (kPa, N-m)
 60.0000 , 113.000 , point in table: (kPa, N-m)
 77.0000 , 121.000 , point in table: (kPa, N-m)
 84.0000 , 134.000 , point in table: (kPa, N-m)
 98.0000 , 144.000 , point in table: (kPa, N-m)
 102.000 , 149.000 , point in table: (kPa, N-m)
 105.000 , 149.000 , point in table: (kPa, N-m)
 111.000 , 156.000 , point in table: (kPa, N-m)
 138.000 , 182.000 , point in table: (kPa, N-m)
 169.000 , 192.000 , point in table: (kPa, N-m)

ENDTABLE

IAXLE 1

ISIDE 2

MYBK_TABLE Brake torque at wheel vs. line pressure

.000000 , .000000 , point in table: (kPa, N-m)
 46.5000 , 376.569 , point in table: (kPa, N-m)
 60.3000 , 625.300 , point in table: (kPa, N-m)
 76.9000 , 749.665 , point in table: (kPa, N-m)
 83.9700 , 878.199 , point in table: (kPa, N-m)
 97.7000 , 1034.52 , point in table: (kPa, N-m)
 101.600 , 1074.82 , point in table: (kPa, N-m)

104.900 , 1081.07 , point in table: (kPa, N-m)
 110.900 , 1116.51 , point in table: (kPa, N-m)
 137.900 , 1406.92 , point in table: (kPa, N-m)
 168.700 , 1672.33 , point in table: (kPa, N-m)
 ENDTABLE

IAXLE 2

ISIDE 2

MYBK_TABLE Brake torque at wheel vs. line pressure

.000000 , .000000 , point in table: (kPa, N-m)
 47.0000 , 92.0000 , point in table: (kPa, N-m)
 60.0000 , 113.000 , point in table: (kPa, N-m)
 77.0000 , 121.000 , point in table: (kPa, N-m)
 84.0000 , 134.000 , point in table: (kPa, N-m)
 98.0000 , 144.000 , point in table: (kPa, N-m)
 102.000 , 149.000 , point in table: (kPa, N-m)
 105.000 , 149.000 , point in table: (kPa, N-m)
 111.000 , 156.000 , point in table: (kPa, N-m)
 138.000 , 182.000 , point in table: (kPa, N-m)
 169.000 , 192.000 , point in table: (kPa, N-m)
 ENDTABLE

SPEED_TABLE Vehicle forward speed vs. station number

.000000 , 100.000 , point in table: (m, kph)
 150.000 , 100.000 , point in table: (m, kph)
 300.000 , 80.0000 , point in table: (m, kph)
 700.000 , 80.0000 , point in table: (m, kph)
 800.000 , 60.0000 , point in table: (m, kph)
 1000.00 , 60.0000 , point in table: (m, kph)
 ENDTABLE

STERRF_TABLE Rear/Front steer ratio vs. vehicle speed

.000000 , .000000 , point in table: (kph, -)
 5.00000 , .000000 , point in table: (kph, -)
 ENDTABLE

* Front wheel steer vs. average angle (iside: 1=lf, 2=rf)

ISIDE 1 , Table ID number

STERRW_TABLE Front wheel steer vs. average angle (ISIDE: 1=LF, 2=RF)

-22.6700 , -21.4600 , point in table: (deg, deg)
 -19.8300 , -18.9100 , point in table: (deg, deg)
 -17.0000 , -16.3200 , point in table: (deg, deg)
 -14.1700 , -13.7000 , point in table: (deg, deg)
 -11.3300 , -11.0300 , point in table: (deg, deg)

```

-8.50000 , -8.33000 , point in table: (deg, deg)
-5.67000 , -5.59000 , point in table: (deg, deg)
-2.83000 , -2.81000 , point in table: (deg, deg)
.000000 , .000000 , point in table: (deg, deg)
2.83000 , 2.85000 , point in table: (deg, deg)
5.67000 , 5.74000 , point in table: (deg, deg)
8.50000 , 8.67000 , point in table: (deg, deg)
11.3300 , 11.6300 , point in table: (deg, deg)
14.1700 , 14.6400 , point in table: (deg, deg)
17.0000 , 17.6800 , point in table: (deg, deg)
19.8300 , 20.7500 , point in table: (deg, deg)
22.6700 , 23.8700 , point in table: (deg, deg)

```

ENDTABLE

ISIDE 2 , Table ID number

STEERRW_TABLE Front wheel steer vs. average angle (ISIDE: 1=LF, 2=RF)

```

-22.4200 , -23.5800 , point in table: (deg, deg)
-19.6200 , -20.5100 , point in table: (deg, deg)
-16.8100 , -17.4700 , point in table: (deg, deg)
-14.0100 , -14.4700 , point in table: (deg, deg)
-11.2100 , -11.5000 , point in table: (deg, deg)
-8.41000 , -8.57000 , point in table: (deg, deg)
-5.60000 , -5.68000 , point in table: (deg, deg)
-2.80000 , -2.82000 , point in table: (deg, deg)
.000000 , .000000 , point in table: (deg, deg)
2.80000 , 2.78000 , point in table: (deg, deg)
5.60000 , 5.53000 , point in table: (deg, deg)
8.41000 , 8.24000 , point in table: (deg, deg)
11.2100 , 10.9200 , point in table: (deg, deg)
14.0100 , 13.5600 , point in table: (deg, deg)
16.8100 , 16.1600 , point in table: (deg, deg)
19.6200 , 18.7200 , point in table: (deg, deg)
22.4200 , 21.2500 , point in table: (deg, deg)

```

ENDTABLE

* Tire Data:

IAXLE 1

* Longitudinal force (N) as a function of slip rate (-) and vertical load (N).

* top row = load, left column = slip, other cells = longitudinal tire force

FX_CARPET 4, columns in table

```

.000000 , 2495.49 , 4145.04 , 5791.47
.650000E-02, 291.848 , 486.576 , 682.922
.100000E-01, 460.963 , 768.624 , 1078.63
.200000E-01, 951.606 , 1578.29 , 2197.57

```

.500000E-01	, 2022.79	, 3249.34	, 4346.21
.100000	, 2548.28	, 3983.74	, 5169.37
.150000	, 2607.24	, 4042.28	, 5199.63
.200000	, 2574.91	, 3979.20	, 5101.28
.250000	, 2522.43	, 3892.00	, 4981.48
.300000	, 2466.90	, 3803.03	, 4863.32
.350000	, 2413.15	, 3718.20	, 4752.32
.400000	, 2362.55	, 3638.97	, 4649.46
.450000	, 2315.34	, 3565.41	, 4554.39

ENDTABLE

* Lateral force (N) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = lateral tire force

FY_CARPET 4, columns in table

.000000	, 2495.49	, 4145.04	, 5791.47
1.00000	, 863.669	, 1169.21	, 1373.58
2.40000	, 1835.02	, 2579.65	, 3094.84
3.70000	, 2303.10	, 3348.08	, 4101.99
5.00000	, 2511.94	, 3736.16	, 4648.93
6.30000	, 2591.18	, 3910.73	, 4917.28
7.70000	, 2607.11	, 3973.89	, 5033.72
9.00000	, 2590.52	, 3971.88	, 5054.05
10.3000	, 2558.92	, 3938.84	, 5027.61
11.6000	, 2519.67	, 3888.78	, 4974.46
13.0000	, 2473.20	, 3824.56	, 4900.30
14.2000	, 2431.79	, 3764.96	, 4828.77

ENDTABLE

* Aligning moment (N-m) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = aligning moment

MZ_CARPET 4, columns in table

.000000	, 2495.49	, 4145.04	, 5791.47
1.00000	, 13.9231	, 32.0987	, 53.4606
2.40000	, 21.7546	, 54.6761	, 95.5255
3.70000	, 19.1791	, 52.8453	, 97.5123
5.00000	, 14.2480	, 42.3973	, 82.1569
6.30000	, 10.0405	, 31.6462	, 63.7421
7.70000	, 6.85269	, 22.5755	, 46.9057
9.00000	, 4.87560	, 16.5394	, 35.1008
10.3000	, 3.53788	, 12.2616	, 26.4387
11.6000	, 2.62040	, 9.22817	, 20.1389
13.0000	, 1.93818	, 6.91466	, 15.2399
14.2000	, 1.52132	, 5.47433	, 12.1455

ENDTABLE

* Tire Data:

IAXLE 2

* Longitudinal force (N) as a function of slip rate (-) and vertical load (N).

* top row = load, left column = slip, other cells = longitudinal tire force

FX_CARPET 4, columns in table

.000000	, 2495.49	, 4145.04	, 5791.47
.650000E-02	, 291.848	, 486.576	, 682.922
.100000E-01	, 460.963	, 768.624	, 1078.63
.200000E-01	, 951.606	, 1578.29	, 2197.57
.500000E-01	, 2022.79	, 3249.34	, 4346.21
.100000	, 2548.28	, 3983.74	, 5169.37
.150000	, 2607.24	, 4042.28	, 5199.63
.200000	, 2574.91	, 3979.20	, 5101.28
.250000	, 2522.43	, 3892.00	, 4981.48
.300000	, 2466.90	, 3803.03	, 4863.32
.350000	, 2413.15	, 3718.20	, 4752.32
.400000	, 2362.55	, 3638.97	, 4649.46
.450000	, 2315.34	, 3565.41	, 4554.39

ENDTABLE

* Lateral force (N) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = lateral tire force

FY_CARPET 4, columns in table

.000000	, 2495.49	, 4145.04	, 5791.47
1.00000	, 863.669	, 1169.21	, 1373.58
2.40000	, 1835.02	, 2579.65	, 3094.84
3.70000	, 2303.10	, 3348.08	, 4101.99
5.00000	, 2511.94	, 3736.16	, 4648.93
6.30000	, 2591.18	, 3910.73	, 4917.28
7.70000	, 2607.11	, 3973.89	, 5033.72
9.00000	, 2590.52	, 3971.88	, 5054.05
10.3000	, 2558.92	, 3938.84	, 5027.61
11.6000	, 2519.67	, 3888.78	, 4974.46
13.0000	, 2473.20	, 3824.56	, 4900.30
14.2000	, 2431.79	, 3764.96	, 4828.77

ENDTABLE

* Aligning moment (N-m) as a function of slip angle (deg) and vertical load (N).

* top row = load, left column = slip, other cells = aligning moment

MZ_CARPET 4, columns in table

.000000	, 2495.49	, 4145.04	, 5791.47
1.00000	, 13.9231	, 32.0987	, 53.4606

```

2.40000 , 21.7546 , 54.6761 , 95.5255
3.70000 , 19.1791 , 52.8453 , 97.5123
5.00000 , 14.2480 , 42.3973 , 82.1569
6.30000 , 10.0405 , 31.6462 , 63.7421
7.70000 , 6.85269 , 22.5755 , 46.9057
9.00000 , 4.87560 , 16.5394 , 35.1008
10.3000 , 3.53788 , 12.2616 , 26.4387
11.6000 , 2.62040 , 9.22817 , 20.1389
13.0000 , 1.93818 , 6.91466 , 15.2399
14.2000 , 1.52132 , 5.47433 , 12.1455

```

ENDTABLE

* Toe angle vs. susp. jounce (IAXLE: 1=f, 2=r)

IAXLE 1 , Table ID number

TOE_TABLE Toe angle vs. susp. jounce (IAXLE: 1=F, 2=R)

```

-70.0000 , 1.11000 , point in table: (mm, deg)
-60.0000 , .880000 , point in table: (mm, deg)
-50.0000 , .660000 , point in table: (mm, deg)
-40.0000 , .480000 , point in table: (mm, deg)
-30.0000 , .320000 , point in table: (mm, deg)
-20.0000 , .190000 , point in table: (mm, deg)
-10.0000 , .800000E-01 , point in table: (mm, deg)
.000000 , .000000 , point in table: (mm, deg)
10.0000 , -.500000E-01 , point in table: (mm, deg)
20.0000 , -.800000E-01 , point in table: (mm, deg)
30.0000 , -.900000E-01 , point in table: (mm, deg)
40.0000 , -.600000E-01 , point in table: (mm, deg)
50.0000 , -.100000E-01 , point in table: (mm, deg)
60.0000 , .600000E-01 , point in table: (mm, deg)
70.0000 , .170000 , point in table: (mm, deg)

```

ENDTABLE

IAXLE 2 , Table ID number

TOE_TABLE Toe angle vs. susp. jounce (IAXLE: 1=F, 2=R)

```

-70.0000 , 1.11000 , point in table: (mm, deg)
-60.0000 , .880000 , point in table: (mm, deg)
-50.0000 , .660000 , point in table: (mm, deg)
-40.0000 , .480000 , point in table: (mm, deg)
-30.0000 , .320000 , point in table: (mm, deg)
-20.0000 , .190000 , point in table: (mm, deg)
-10.0000 , .800000E-01 , point in table: (mm, deg)
.000000 , .000000 , point in table: (mm, deg)
10.0000 , -.500000E-01 , point in table: (mm, deg)
20.0000 , -.800000E-01 , point in table: (mm, deg)

```

30.0000 , -.900000E-01 , point in table: (mm, deg)
 40.0000 , -.600000E-01 , point in table: (mm, deg)
 50.0000 , -.100000E-01 , point in table: (mm, deg)
 60.0000 , .600000E-01 , point in table: (mm, deg)
 70.0000 , .170000 , point in table: (mm, deg)

ENDTABLE

* Vertical Geometry Data: Station, Z coord., Vert. Curv. Len, Back Grade, Fwd Grade

VGDATA

.000000 , 47.7400 , .000000 , -.265000E-01, -.265000E-01
 303.260 , 39.7040 , 70.0000 , -.265000E-01, .800000E-02
 373.641 , 39.0590 , .000000 , .800000E-02, .800000E-02
 645.750 , 41.2360 , 50.0000 , .800000E-02, -.115000E-01
 699.452 , 41.1060 , .000000 , -.115000E-01, -.115000E-01
 1153.65 , 35.8830 , 90.0000 , -.115000E-01, .290000E-01
 1261.24 , 37.1800 , .000000 , .290000E-01, .290000E-01
 1652.04 , 48.5130 , 120.000 , .290000E-01, -.343300E-01
 1789.57 , 47.5920 , .000000 , -.343300E-01, -.343300E-01

ENDTABLE

VW_TABLE Wind speed vs. time

.000000 , .000000 , point in table: (s, kph)

ENDTABLE

* FINAL CONDITIONS

XO 55771.37845678 , Abs. X trans. of S0 (m)
 YO 116787.2896883 , Abs. Y trans. of S0 (m)
 ZO 43.82782337535 , Abs. Z trans. of S0 (m)
 YAW -58.76415136824 , Abs. Z rot. of Spp (deg)
 PITCH 2.090825002853 , Y rot. of Sp rel. to Spp (deg)
 ROLL 1.474078310079 , X rot. of S rel. to Sp (deg)
 JNC_WLF -.1542292215819E-02, Trans. of USM_LF0 rel. to USM_LFJ (m)
 JNC_WLR -.6093035828748E-02, Trans. of USM_LR0 rel. to USM_LRJ (m)
 JNC_WRF .6778724864419E-02, Trans. of USM_RF0 rel. to USM_RFJ (m)
 JNC_WRR .7281421839093E-03, Trans. of USM_RR0 rel. to USM_RRJ (m)
 ROTLF 655.9429239575 , Rotation angle of LF wheel (rev)
 ROTLR 657.2105586240 , Rotation angle of LR wheel (rev)
 ROTRF 655.7519770268 , Rotation angle of RF wheel (rev)
 ROTRR 656.8842046059 , Rotation angle of RR wheel (rev)
 TANLF .1161598020432E-01, Slip angle tan of LF tire, with time lag (-)
 TANLR .6663108068166E-02, Slip angle tan of LR tire, with time lag (-)
 TANRF -.1392409136169E-01, Slip angle tan of RF tire, with time lag (-)
 TANRR -.9604208869061E-02, Slip angle tan of RR tire, with time lag (-)
 IVERR -.3304179233928 , Integral of velocity error for speed controller (m)
 VXS 60.05729445431 , Vehicle forward speed (kph)

VYS -.6673271692655E-01, Abs. Y trans. speed of SCMC (kph)
VZS .1430301553940 , Abs. Z trans. speed of SCMC (kph)
AVZ .2694060046020E-01, Abs. Z rot. speed of S (deg/s)
AVY -.4842430884888 , Abs. Y rot. speed of S (deg/s)
AVX .6835589781954E-01, Abs. X rot. speed of S (deg/s)
JNCR_WLF -.2140056640370E-01, Trans. speed of USM_LF0 rel. to S (m/s)
JNCR_WLR .6962472080017E-02, Trans. speed of USM_LR0 rel. to S (m/s)
JNCR_WRF -.1983432438957E-01, Trans. speed of USM_RF0 rel. to S (m/s)
JNCR_WRR .8228532635335E-02, Trans. speed of USM_RR0 rel. to S (m/s)
AVY_LF 5.650478645709 , Angular velocity of LF wheel (rev/s)
AVY_LR 5.368641662907 , Angular velocity of LR wheel (rev/s)
AVY_RF 5.638193145834 , Angular velocity of RF wheel (rev/s)
AVY_RR 5.343941891298 , Angular velocity of RR wheel (rev/s)

END

Appendix E — Truck Multibody Models

This appendix describes the truck and combination vehicle models used in VDM RoAD. It is intended to convey the modeling assumptions that were used to build a “fleet” of truck simulation programs for straight trucks and tractor-semitrailers with different axle configurations. The tire model is described separately in Appendix F.

An earlier version of this appendix was published as SAE paper 960173, by Michael W. Sayers and Stephen M. Riley.

Conventions

The global inertial frame is denoted by N (for Newtonian frame) and has unit vectors \mathbf{n}_x , \mathbf{n}_y , and \mathbf{n}_z . The convention of ISO 8855 is followed, in which \mathbf{n}_z is “up” (opposite to the direction of gravity) [2].

The terms *nominal* and *nominal configuration* refer to a state of the multibody system in which all position state variables are zero.

The abbreviation DOF refers to the term *degree of freedom* or *degrees of freedom*, depending on the context.

When a body is added to the multibody system, its allowable motion is defined relative to a body already in the model. (N is considered a body for this purpose.) The new body is said to be a *child* of the previously existing body, and the previously existing body is said to be the *parent* of the new body.

Overview of the Basic Factors Affecting Vehicle Behavior

The purpose of VDM RoAD models is to predict the dynamic 3D motions of heavy trucks in response to steering and braking inputs. Generally, directional control, roll stability, and stopping distance are of particular concern. Braking performance, and directional control and stability mainly involve dynamics in the yaw plane, while roll stability is principally related to the roll plane. Aerodynamic effects have a secondary influence with regard to both concerns and are ignored in VDM RoAD models.

Notes: Aerodynamic effects are left out of the models for two reasons: (1) adding them would do little to improve the accuracy of the models, and (2) the addition of aerodynamic terms raises the possibility of making errors due to mistakes in the inputs.

It is fairly easy to add the aerodynamic effects to the models, and they can be included in custom versions upon arrangement with MSC.

Yaw Plane

Figure E.1 shows a free-body diagram of a four-wheeled tractor as viewed from the top (the yaw plane). There are just three equations that govern its basic behavior in the yaw plane: the sum of the tire shear forces and the horizontal hitch force must equal the vehicle mass times its acceleration in both the vehicle X and Y directions, and the moment of those forces about the vehicle mass center must be equal to the product of the yaw acceleration and the vehicle yaw moment of inertia. Aligning moments at each wheel (M_{zi}) also contribute to the overall yaw moment. Thus, a main objective of VDM RoAD models is to accurately predict tire shear forces, tire aligning moments, and hitch forces.

Yaw behavior is also influenced by the rotary motion of the vehicle bodies in roll and pitch. Mechanical energy is transferred as the vehicle rolls and pitches, and these motions contribute to the vehicle transient response.

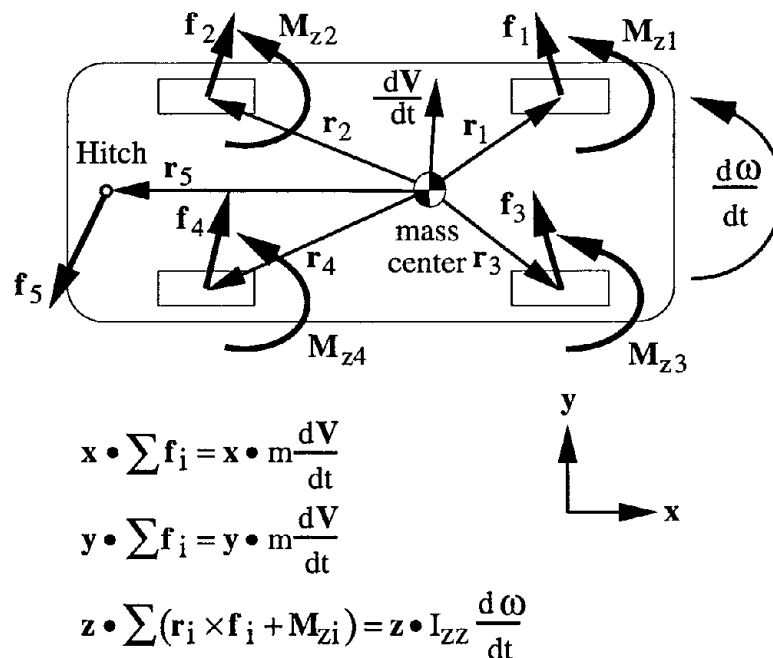


Figure E.1. Free-body diagram in the yaw plane.

Note: The equations appearing in figures E.1 and E.2 are written only to identify the basic factors affecting vehicle behavior. All equations for the rigid body motions in the VDM RoAD models were actually derived automatically by the AutoSim code generator. The

AutoSim equations are exact for the nonlinear 3D kinematical and dynamical models that will be described.

Roll Plane

Figure E.2 shows a free-body diagram in the roll plane for a truck body and one axle as viewed from the rear when making a left-hand turn. (A left turn implies a positive rotation of the vehicle about the Z axis.) The only external forces acting on the vehicle system are due to the tires and any hitches connecting it to other vehicle units. The vehicle is also subjected to roll moments (M_h) from the hitch(es). The vertical components of the tire forces, which do not appear explicitly in the yaw-plane equations, are basic factors in the roll-plane equations. The overall roll stability is characterized by the first equation in Figure E.2, involving the balance of moments about the vehicle mass center. When tires lift off of the ground, the forces are zero on one side of the axle. Lift-off usually occurs first for tires on the axle furthest to the rear of all vehicle units in the system, on the side that is on the inside of the turn (the left side in Figure E.2). When lift-off has occurred for more than a critical number of axles, roll stability cannot be maintained. This condition is considered to be the onset of rollover. Thus, the basic factors for predicting roll stability are the vertical tire forces, and their points of application relative to the vehicle mass center, which change as the vehicle rolls.

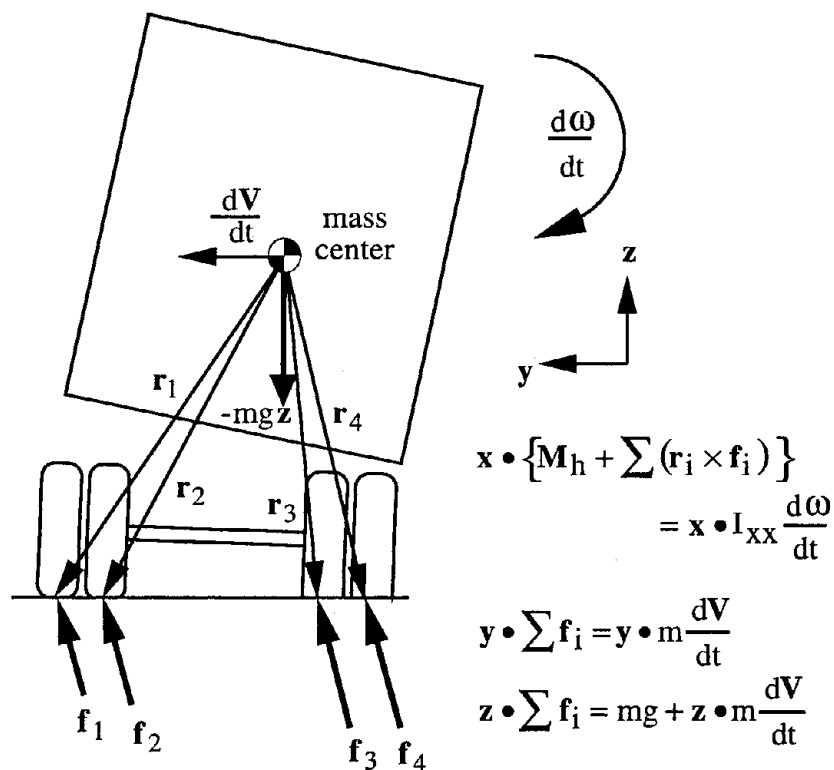


Figure E.2. Free-body diagram in the roll plane.

For simplicity in Figure E.2, the separate motion of the axle is not shown, and the moment balance in the X direction is written as if all of the roll inertia were lumped into

the sprung mass. The independent motions of the axles relative to the main bodies are a secondary factor that must be included in the model to obtain realistic transient effects.

Rigid Body Kinematics

The tractor-semitrailer model consists of rigid bodies for the sprung mass of the lead unit, the sprung mass of the trailing unit (if there is one), and the axles. The kinematics of these rigid bodies are described below.

Sprung Masses

The sprung mass of the lead unit, S_1 (with body fixed unit vectors s_{1X} , s_{1Y} , and s_{1Z}), is given six kinematical DOF, such that it can attain any position and/or any orientation in space.

The sprung mass of the trailing unit, S_2 (with body fixed unit vectors s_{2X} , s_{2Y} , and s_{2Z}), is a child of S_1 . The hitch connection between S_1 and S_2 is modeled kinematically as a ball joint. It is given three rotational DOF relative to S_1 , but no translational DOF. The motion of S_2 about the hitch is further retarded by a roll torsional spring, described later.

Axles

The movement of an axle relative to the sprung mass to which it is attached can be described by a sequence of two motions: (1) a rotation in roll, and (2) a vertical translation [5]. Two rigid bodies, A_r and A , are used to model the axle, each with one DOF. Body A_r is an intermediate frame that is given zero mass and rotational inertia. Body A is given the mass and rotational inertia properties of the axle. All external forces and moments that act on the axle (i.e., suspension and tire forces and moments) are applied to body A .

The bodies A_r and A , respectively, have body fixed unit vectors: and \mathbf{a}_x , \mathbf{a}_y , \mathbf{a}_z , and \mathbf{a}_x , \mathbf{a}_y , \mathbf{a}_z . Figure E.3 shows the directions of the unit vectors in the nominal configuration for a positive roll-steer coefficient. Body A_r is a child of S that rotates in roll about an axis aligned with \mathbf{a}_x .

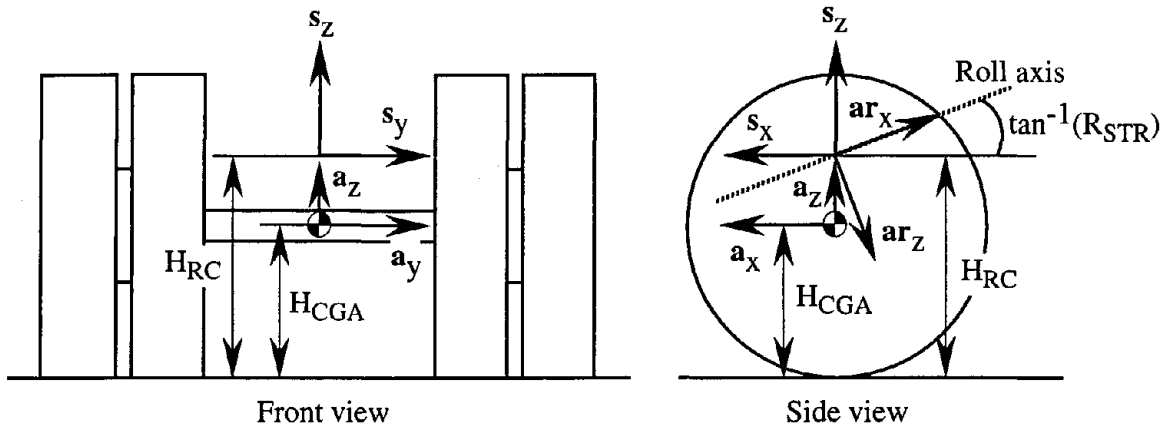


Figure E.3. Axle axes and directions.

The roll axis (ar_x) is inclined from $-s_x$ by the arc-tangent of R_{STR} , the roll-steer coefficient. The axis direction is set such that a positive roll of the sprung mass (leaning to the right, relative to the axle) also corresponds to a positive rotation of the axle about ar_x , relative to the sprung mass. Positive roll angle causes the axle to steer to the left for a positive value of R_{STR} . This sign convention for roll-steer is consistent with the ISO axis system where s_z is up and steer to the left is positive. However, for the SAE coordinate system, in which the s_z axis is pointed down, the sign of the roll-steer coefficient would be reversed because the left-hand steer would be negative [1].

Body A translates vertically in the direction a_z and it is a child of A_r . Nominally, body A is aligned with body S. (Thus, A is tilted with respect to A_r .) The origin of the coordinate system of body A is located at the axle mass center, which is assumed to be in line with the spin axis of the wheels on the axle.

The axle roll and mass centers are assumed to lie at the center of the axle laterally and longitudinally. The heights of the roll and mass centers are specified by the user with the parameters H_{RC} and H_{CGA} .

Mass and Rotational Inertia

The lateral locations of all mass centers (sprung masses and axles) are given Y coordinates of zero, placing them in the center pitch plane of the vehicle. The locations of the axle mass centers are also at the centers of the axles. The heights of the mass centers are provided by the user for the axles individually, and for the entire vehicle units. The heights of the mass centers for the sprung masses are calculated by VDM RoAD from other user input parameters.

The masses of the sprung masses are described with equations involving the axle loads. Equations are also used to locate the mass centers longitudinally. This is done to eliminate the need for the user to calculate mass properties that are based on axle loads—axle loads are instead used directly as inputs.

It is usually easier to measure the inertial properties for an entire unladen unit than for the laden (or unladen) sprung mass alone. VDM RoAD calculates the mass and rotational inertia properties of the sprung mass from measurements made for the entire unladen unit, the difference between the laden and unladen axle loads, and parameters that the user must set to estimate the size of the load. The load is assumed to be rectangular and of uniform density. This approach is taken to permit the user to enter data for the complete unladen units. These numbers never change for a particular vehicle, regardless of how it is loaded. Thus, to change the load, the user need only modify the load parameters.

The roll, pitch, and yaw moments of inertia are about the mass center with respect to the body X, Y, and Z axes. The products of inertia are not zero, but for most loading conditions, they contribute little to the system dynamics. For unusual loading conditions, however, (e.g., when the load mass center is very high, or very far forward or rearward), the X-Z product of inertia can affect the transient response (yaw rate in particular). A parameter for the X-Z product of inertia is included to account for such situations.

For the axles, the moments of inertia about the mass center in yaw and roll are assumed to be equal and must be provided by the user. The moment of inertia in pitch and the products of inertia are relatively small. They are neglected by setting them to zero in the VDM RoAD models.

Equilibrium

The heights of reference points such as wheel centers and the top side of the load bed are defined for the case of the vehicle resting in equilibrium on a flat level surface. In a real vehicle, these heights depend on the load: the heights decrease slightly as the vehicle settles down when the payload is increased.

In general, the sensitivity of these heights to load has a negligible effect on the vehicle behavior. The common practice is to use the same height values for all load conditions. However, if you are concerned with this effect, then the height parameters can be adjusted for the various loads.

Suspensions

The significant forces and moments acting between the suspensions and the sprung masses are due to suspension springs, hydraulic dampers, and auxiliary roll stiffness.

Vertical Spring Forces

The vertical stiffness of a truck suspension is modeled with two springs. The force of each spring acts through a point that is fixed in the axle body, and is reacted by the sprung mass. The two spring points for an axle are defined by the distance separating

them (the parameter L_{SPR}). They are on the axle Y axis, with local Y coordinates of $\pm \frac{L_{SPR}}{2}$. The direction of the force is coincident with a_z .

The force-deflection properties of the spring are intended to capture the observed hysteretic frictional behavior of heavy truck suspensions, as shown in Figure E.4. Each spring is characterized by three elements: (1) a tabular function that defines an upper force limit for deflection Δ_s , $F_U(\Delta_s)$, (2) a similar tabular function that defines a lower force limit, $F_L(\Delta_s)$, and (3) a characteristic deflection parameter β . The describing equations for the spring are built on a simpler model originally published by Fancher et al. [4] A state variable (F_s) is added to the model for each spring. The value of F_s is the instantaneous force of the spring.

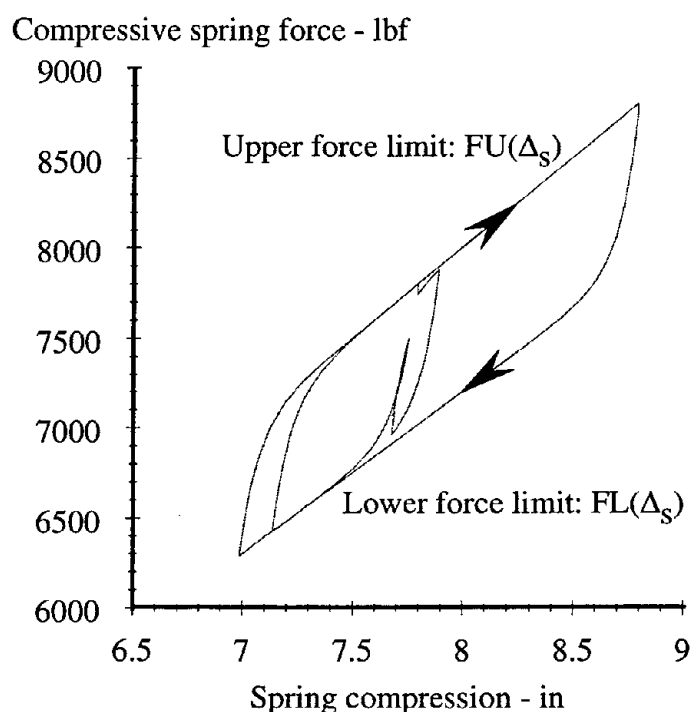


Figure E.4. Force-deflection behavior of spring.

At the beginning of the simulation run, F_s is set to an initial value, F_{s0} . The value of F_{s0} for a simulation run is determined by the parameter $INIT_SPG$. If $INIT_SPG$ is non-zero, F_{s0} is computed to keep the vehicle in static equilibrium (for a flat, level surface) and is given by

$$F_{s0} = (M - MUS)g/2 \quad (1)$$

where M is the laden axle load and MUS is the axle mass. The tables F_U and F_L are searched for an initial spring deflection, Δ_{s0} , that will place F_{s0} midway between F_U and F_L . This definition of the static deflection means that the jounce variables computed during a simulation have the same physical interpretation as the spring compression

provided in the input tables. Therefore, force-deflection curves (e.g., Figure E.4) based on simulation outputs should match the original data.

If INIT_SPG is zero, F_{s0} is set to the value of the state F_s read from the LPI file created by VDM RoAD. This allows the simulation to start from the final conditions of a run that was made previously.

Once the simulation starts, F_s is calculated by integrating the following differential equation:

$$\dot{F}_s = \dot{\Delta}_s \left\{ \text{sign}(\dot{\Delta}_s) (F_{\text{lim}} - F) / \beta + \frac{[F_s - \text{FL}(\Delta_s)]}{[\text{FU}(\Delta_s) - \text{FL}(\Delta_s)]} \left[\frac{\partial \text{FU}(\Delta_s)}{\partial \Delta_s} - \frac{\partial \text{FL}(\Delta_s)}{\partial \Delta_s} \right] + \frac{\partial \text{FL}(\Delta_s)}{\partial \Delta_s} \right\} \quad (2)$$

where

$$F_{\text{lim}} = \begin{cases} \text{FU}(\Delta_s), & \text{if } \dot{\Delta}_s > 0 \\ \text{FL}(\Delta_s), & \text{if } \dot{\Delta}_s < 0 \end{cases} \quad (3)$$

This equation quite accurately captures the observed hysteretic behavior of the Coulomb friction in heavy truck leaf springs. This same model can also be used to accurately model heavy truck air springs. Typically, however, the difference between the upper and lower envelopes, FU and FL, is much smaller for air springs.

Hydraulic Dampers

In addition to suspension damping due to the Coulomb friction in the vertical springs, a truck suspension may be equipped with a pair of hydraulic dampers. The dampers in the model are added in the same way as the springs. Each acts in the \mathbf{a}_z direction through a point fixed in the axle body. The points are located with a single parameter that specifies the distance between them. (This distance is not necessarily the same as that which is used for the springs.) The dampers are modeled with a nonlinear table of values of force versus stoke rate.

Auxiliary Roll Stiffness

Roll of the axle is resisted in the model by the vertical suspension springs and also by a torsional spring whose restoring moment is proportional to the roll of body A_r about \mathbf{ar}_x . The moment is applied to the axle (body A) in the \mathbf{s}_x direction and reacted by the sprung mass body S. The torsional spring accounts for additional stiffness due to linkages, and also from an anti-sway bar (if there is one).

The torsional spring rate is determined from test data by measuring the total roll stiffness from the axle and then subtracting the stiffness due to the springs at their spacing. It is possible for the auxiliary roll stiffness to be negative, if the frame has considerable compliance in twist.

Hitch Moments

The hitch is modeled as a ball joint. The trailing unit sprung mass (body S_2) is a “child” of the leading unit sprung mass (body S_1) in a multibody tree topology. The origin of the frame of reference of S_2 is at the hitch point. The origin is fixed in S_1 , but S_2 is given three rotational DOF relative to S_1 . The sequence of rotations is intended to match the kinematics of a conventional fifth wheel. The first rotation, pitch, is about s_{1y} . The third rotation, yaw (articulation angle), is about s_{2z} , and corresponds to the axis of the kingpin (assuming the kingpin is perfectly rigid). The second rotation, roll, is about the X-axis of an intermediate axis system. The rotations are shown in Figure E.5.

Proportional restoring moments act between S_1 and S_2 . The pitch moment is about the Y axis, the roll moment is about the X axis, and the yaw moment is about the Z axis. For a conventional fifth wheel, the pitch and yaw stiffnesses are often set to zero, and the roll stiffness is usually given a value of 1 to 3 million in-lbf/deg. The pitch and yaw stiffnesses are included in the model to represent the behavior of a hitch other than a conventional fifth wheel.

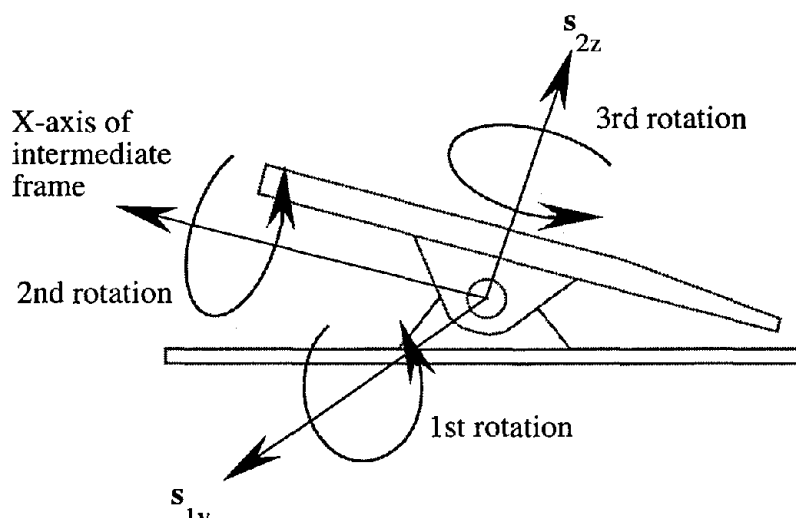


Figure E.5. Hitch model.

Steering System

Figure E.6 shows a top view of a truck steering system. The steering wheel angle is denoted by δ_{sw} . The total steer at the road of the left and right wheels on the front axle (δ_{TL} and δ_{TR} , respectively) are the sums of three kinematical terms: (1) roll-steer (δ_{roll}), (2) bump-steer (δ_b), and (3) Ackerman-steer (δ_{aL} and δ_{aR} , for the left and right wheels, respectively), and two compliance terms: (1) wrap-steer (δ_w) and (2) steer due to the combined effect of tie rod and steering column compliance (δ_{cl} and δ_{cr} , for the left and

right wheels, respectively). The modeling of these effects is described below. For all axles other than the front axle, the only steering effect taken into account is roll-steer.

Roll-steer

As mentioned in the section above on axle kinematics, the roll axis of each axle, \mathbf{ar}_x , is defined such that, as the axle rolls, it steers in accordance with its roll-steer coefficient, R_{STR} . The steering due to roll-steer enters the equations of motion through the tire equations, described in Appendix F. The motion of the axle about its roll axis causes the directions of vectors \mathbf{a}_y and \mathbf{a}_x to contain the effects of roll-steer, and thus, so does the wheel spin axis, \mathbf{w}_y . The wheel spin axis is used to define the tire X and Y axes, \mathbf{t}_x and \mathbf{t}_y , which in turn determine the tangent of the instantaneous value of lateral slip as described in Appendix F.

The roll-steer at the road is slightly asymmetric from left to right. The exact expressions are very complex, and are not given here. For the purpose of outputting total steer at the road, roll-steer is assumed to be the same for the left and right wheels, and is represented with the very good approximation of

$$\delta_{roll} = R_{str} r q(A) \tag{4}$$

where $rq(A)$ is the rotational coordinate of the axle.

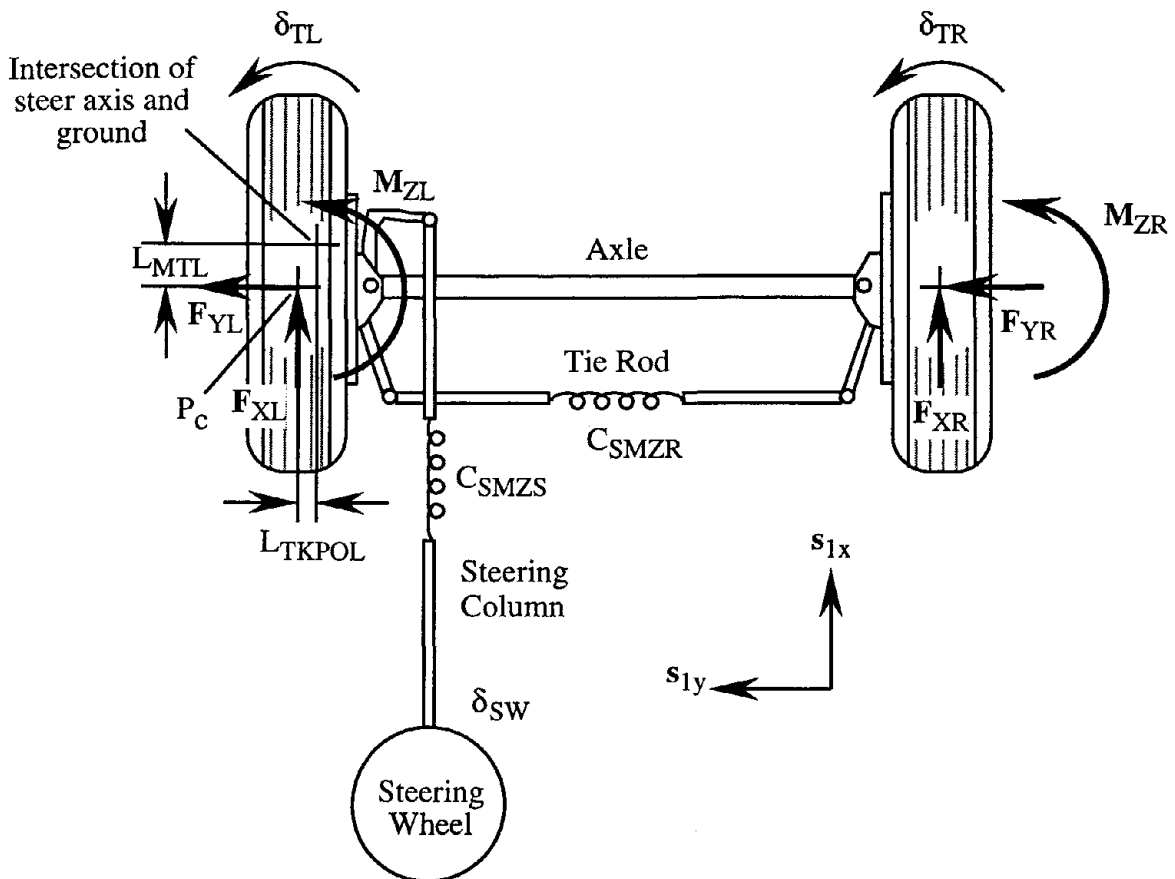


Figure E.6. Steering system.

Bump-steer

When an axle translates vertically, the geometry of the steering linkages can induce some steering. This effect is known as *bump-steer*. Bump-steer is assumed to be the same for the left and right wheels, and it is modeled with a single coefficient, R_{BMPSTR} , that has units of degrees of steer per unit of axle jounce. Axle jounce is defined as the translational coordinate of the axle, $\text{tq}(A_1)$. This is equal to the amount that the axle CG has translated vertically (in the \mathbf{a}_{1Z} direction) from its nominal location in body A_{1r} (i.e., at a distance H_{CGA} above the ground). Bump-steer is thus given by

$$\delta_b = R_{\text{BMPSTR}} \text{tq}(A_1) \quad (5)$$

Ackerman-steer

If the front of the leading unit is supported in the air by its frame (such that the front wheels can freely steer without the tires contacting the ground), the kinematical gains between the steering wheel and the left wheel, and the steering wheel and the right wheel, are normally nonlinear, and different from side to side. This is mostly by design, in an attempt to produce Ackerman-steer, but also due to the complex geometry of the steering linkages. VDM RoAD provides a separate table function for the kinematical steer (at zero jounce and zero roll) of each front wheel as a function of steering wheel angle. These are assumed to contain the total kinematical gains (i.e., including the steering gearbox reduction gear gain) from the steering wheel to the road. The left and right Ackerman steering gains are given by

$$\delta_{aL} = \text{str}_{aL}(\delta_{\text{SW}}) \quad \delta_{aR} = \text{str}_{aR}(\delta_{\text{SW}}) \quad (6)$$

respectively, where str_{aL} and str_{aR} denote table-lookup functions.

Wrap-steer

When the vehicle accelerates longitudinally, the axles tend to wrap (twist about the Y axis). This can cause steering, especially on the front axle. This type of steering is modeled by a wrap-steer coefficient (R_{WRAPSTR}) multiplied by the amount of axle wrap. Axle wrap is measured in degrees, and thus R_{WRAPSTR} is dimensionless. The amount of axle wrap is in turn modeled by a wrap-up compliance coefficient (C_{WRAP}) multiplied by the sum of the moments of the tire longitudinal forces about \mathbf{a}_Y . C_{WRAP} thus has the dimensions of degrees per unit of moment. The wrap-steer of the front axle is given by

$$\delta_w = R_{\text{WRAPSTR}} C_{\text{WRAP}} \left[(H_{\text{CGA}} - \Delta_{\text{TL}}) F_{\text{XL}} + (H_{\text{CGA}} - \Delta_{\text{TR}}) F_{\text{XR}} \right] \quad (7)$$

where $H_{\text{CGA}} - \Delta_{\text{TL}}$ and $H_{\text{CGA}} - \Delta_{\text{TR}}$ are the instantaneous left and right front tire radii, and F_{XL} and F_{XR} are the left and right front tire longitudinal forces, respectively.

Steering System Compliance

Compliance in the steering column and tie rod (C_{SMZS} and C_{SMZR}), respectively, decrease the steer angle at the road when the steering is resisted by self-aligning moment [3]. Steering compliances are *very* significant in determining the relation between steering

wheel angle and road-wheel steer. For example, δ_{TL} and δ_{TR} calculated for a rigid steering system ($C_{SMZS} = C_{SMZR} = \text{zero}$) can be on the order of 20% and 75% too large, respectively, even when compared with a "stiff" steering system (e.g., $C_{SMZS} = C_{SMZR} = 4 \times 10^{-5} \text{ deg/(in-lbf)}$). This can result in predicted lateral accelerations that are too large by 50% or more for a given steering wheel input.

VDM RoAD solves for the left and right steer angles due to steering system compliance using a quasi-static moment balance on the two wheels.

Note: Quasi-static means that the inertial forces and moments in the steering system are ignored — these are small and to include them would introduce unnecessary high frequency dynamics into the model.

The moment balance involves the left and right tire forces and moments (F_{XL} , F_{YL} , F_{ZL} , M_{ZL} and F_{XR} , F_{YR} , F_{ZR} , M_{ZR}), as well as the left and right caster angles, kingpin inclination angles, and the kingpin offsets at the wheel center (A_{CASTL} , A_{KPIL} , L_{KPOL} and A_{CASTR} , A_{KPIL} , L_{KPOL}).

The left and right mechanical trail (L_{MTL} and L_{MTR}) at the ground are given by

$$\begin{aligned} L_{MTL} &= (H_{CGA} - \Delta_{TL}) \tan^{-1}(A_{CASTL} - \text{pitch}(S_1)) \\ L_{MTR} &= (H_{CGA} - \Delta_{TR}) \tan^{-1}(A_{CASTR} - \text{pitch}(S_1)) \end{aligned} \quad (8)$$

where $\text{pitch}(S_1)$ is the pitch angle of body S_1 .

The left and right kingpin offsets at the ground (L_{TKPOL} and L_{TKPOR}) are given by

$$\begin{aligned} L_{TKPOL} &= L_{KPOL} - (H_{CGA} - \Delta_{TL}) \tan^{-1}(A_{KPIL}) \\ L_{TKPOR} &= L_{KPOR} - (H_{CGA} - \Delta_{TR}) \tan^{-1}(A_{KPIR}) \end{aligned} \quad (9)$$

The total steering moments acting on the left and right wheels (M_{SZL} and M_{SZR}) are:

$$\begin{aligned} M_{SZL} &= M_{ZL} - L_{MTL} F_{YL} - L_{TKPOL} F_{XL} \\ &\quad - [L_{MTL} \sin(A_{KPIL}) + L_{TKPOL} \sin(A_{CASTL} - \text{pitch}(S_1))] F_{ZL} \\ M_{SZR} &= M_{ZR} - L_{MTR} F_{YR} + L_{TKPOR} F_{XR} \\ &\quad + [L_{MTR} \sin(A_{KPIR}) + L_{TKPOR} \sin(A_{CASTR} - \text{pitch}(S_1))] F_{ZR} \end{aligned} \quad (10)$$

Finally, δ_{cL} and δ_{cR} are given by

$$\begin{aligned} \delta_{cL} &= C_{SMZS} (M_{ZL} + M_{ZR}) \\ \delta_{cR} &= \delta_{cL} + C_{SMZR} M_{ZR} \end{aligned} \quad (11)$$

Total Steer at the Road

The wheel steer angles relative to the spin axes, δ_L and δ_R , are:

$$\begin{aligned}\delta_L &= \delta_b + \delta_{aL} + \delta_w + \delta_{cL} \\ \delta_R &= \delta_b + \delta_{aR} + \delta_w + \delta_{cR}\end{aligned}\tag{12}$$

The total steer angles at the road, δ_{TL} and δ_{TR} , are equal to δ_L and δ_R , plus the effect of roll-steer

$$\begin{aligned}\delta_{TL} &= \delta_L + \delta_{roll} \\ \delta_{TR} &= \delta_R + \delta_{roll}\end{aligned}\tag{13}$$

References

1. "Vehicle Dynamics Terminology." SAE J670e, 1976.
2. "International Standard, Road vehicles - Vehicle dynamics and road-holding ability - Vocabulary." ISO 8855, 1991.
3. Durstine, J.W. "The Truck Steering System from Hand Wheel to Road Wheel." SAE Paper No. 730039, 1973.
4. Fancher, P.S., et al. "Measurement and Representation of the Mechanical Properties of Truck Leaf Springs." SAE Paper No. 800905, 1980.
5. Winkler, C.B., et al. "A Test Facility for the Measurement of Heavy Vehicle Suspension Parameters." SAE Paper No. 800906, 1980.

Appendix F — The Car Model

This appendix describes the modeling assumptions that were used to build the automotive vehicle model used in VDM RoAD. The tire model is described in Appendix F.

The description is intended for the engineer who has a multibody program and wants to simulate the essential braking and handling behavior of an automobile with full fidelity, without the overhead associated with component-level details (bushing rates, linkage geometry, etc.). The model that will be described was tested using the AutoSim multibody code generator [3,4]. AutoSim generates equations symbolically, performs coding optimizations, and generates a custom simulation program. The simulation programs obtained by AutoSim have run-time performance comparable with (and usually better than) that of a hand-coded program based on the same model.

Portions of this appendix were published in the technical paper "A Generic Multibody Vehicle Model for Simulating Handling and Braking," *Vehicle System Dynamics*, Vol. 25 supplement, 1996, pp. 599-613, by M.W. Sayers and D.S. Han.

Multibody Ingredients

Readers who are familiar with the vehicle modeling literature may be disturbed to see less information than is typical for papers presenting models. This is because the model is intended to be implemented with a multibody program that will assemble the equations. Multibody programs handle details involving state variables, coordinate transformations, and equations of motion.

The major ingredients for describing the rigid-body kinematics and dynamics are bodies, points, and vector directions. Forces will be defined in terms of magnitudes, directions, and points that lie on the lines of action. The magnitudes and directions will usually be described in terms of quantities such as position and velocity vectors that are available from the multibody program with function such as `pos` (position vector) and `vel` (velocity vector). Some of the model degrees of freedom (DOF) are handled with auxiliary user-defined state variables and equations.

Nomenclature

All of the symbols and unconventional functions that are used in this appendix are listed below. Subscripts for wheels and suspensions are omitted when an equation applies to a single wheel. Vectors are shown in bold face characters. The nominal configuration of a multibody system in this paper represents the condition when all state variables are zero. Depending on the model, the nominal configuration may or may not be in equilibrium.

Symbols

α	delayed lateral slip angle (rad)
CTC	center of tire contact (point)
δ	steering of wheel relative to body (rad)
δ_s	steering due to steering system input from the driver (rad)
δ_{sw}	steering wheel angle (rad)
δ_k	steering due to suspension kinematics (rad)
δ_c	steering due to compliance in the suspension and steering system (rad)
Δ	suspension compression (also called jounce) (m)
F_s	compressive spring force, at spring (N)
F_x, F_y, F_z	components of tire force acting at point CTC (N)
ϕ_k	roll of wheel relative to body due to suspension kinematics (rad)
ϕ_c	roll of wheel relative to body due to compliance (rad)
γ	inclination (roll) of wheel relative to ground (rad)
H_{rc}	nominal roll center height (m)
H_{wc}	nominal height of wheel center (m)
I_s	moment of inertia of wheel about spin axis ($\text{kg}\cdot\text{m}^2$)
K_s	suspension spring rate (at spring) (N/m)
K_t	tire vertical spring rate (N/m)
κ	longitudinal slip ratio for a wheel (-)
L_{tk}	nominal track width (m)
L_{wb}	nominal wheelbase (m)
L_{relax}	tire relaxation length (m)
M_{by}, M_{dy}	moment about wheel spin axis due to brake or driveline (N-m)
M_f, M_r	mass of vehicle supported by front or rear tires (kg)
M_z	tire aligning moment (N-m)
θ	pitch of the sprung mass (rad)
r_z	unit-vector perpendicular to road surface at tire contact point
R	instant rolling radius of tire (m)
R_{sg}	ratio of steering wheel angle divided by average road-wheel steer (-)
R_p	anti-pitch coefficient: ratio of longitudinal to vertical movement (-)
R_d, R_s	ratio of damper or spring compression to suspension compression (-)
s_x, s_y, s_z	unit-vectors aligned with X, Y, Z axes of vehicle body
t_x, t_y	unit-vectors aligned with X, Y axes of tire/road contact
t'_x, t'_y	tire/road unit-vectors without effect of steer
τ	auxiliary state variable: tangent of delayed lateral slip angle (-)
w_y, w_z	unit-vectors aligned with Y, Z axes in wheel plane
W_o	origin of wheel coordinate system, nominally at point CTC (point)
W_c	wheel center (point)
ω_s	auxiliary state variable: wheel spin rate (rad/sec)
v_x, v_y	scalar components of velocity vector of W_o (m/s)

X_c, Y_c, Z_c coordinates of CTC in wheel body coordinate system (m)

Multibody Functions

$dir(r)$ direction of vector r : $r/|r|$

$pos(P1, P2)$ position vector going from point P2 to point P1

$vel(P)$ absolute velocity vector of point P

Overview of Factors Affecting Vehicle Behavior

Figure F.1 shows a free-body diagram of a four-wheeled vehicle as viewed from the top. There are just three governing equations: the sum of the tire shear forces must equal the vehicle mass times its acceleration in both the vehicle X and Y directions, and the moment of those forces about the vehicle mass center must be equal to the product of the yaw acceleration and the vehicle yaw moment of inertia. Thus, the main objective of the vehicle model is to accurately predict tire shear forces.

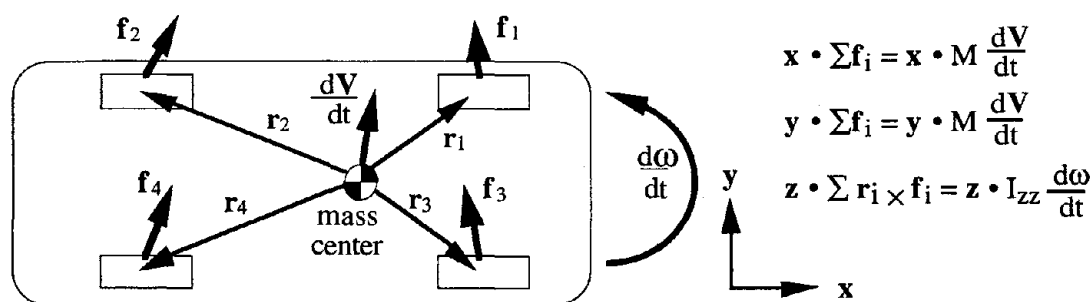


Figure F.1. Primary factors influencing vehicle system motions.

A vehicle is also subject to aligning moments in the tire contact patches. The aligning moment has a negligible direct effect on the vehicle yaw, but, due to steering compliance, it can be a significant factor in determining the all-important shear forces. Another behavior that influences the vehicle response involves the rotary motion of the car body in roll and pitch. Mechanical energy is transferred as the vehicle pitches and rolls, and these motions contribute to the vehicle transient response.

Besides the tire/road interactions, the only forces and moments acting on the vehicle are due to aerodynamic effects. They have a secondary influence, but are relatively easy to add to multibody models.

Rigid Body Kinematics

The model is based on a rigid body that represents the main body of the vehicle and has six DOF. An additional four bodies are added, each with a single translational DOF, to

account for the vertical movements allowed by the suspensions. The wheel bodies are positioned such that the origins of their local coordinate systems are nominally at the locations of the centers of tire contact (see Figure F.2). Longitudinally, the origins of the front and rear wheels are separated by the vehicle wheelbase, L_{wb} . Laterally, they are separated by the vehicle front and rear track widths, $L_{tk,f}$ and $L_{tk,r}$.

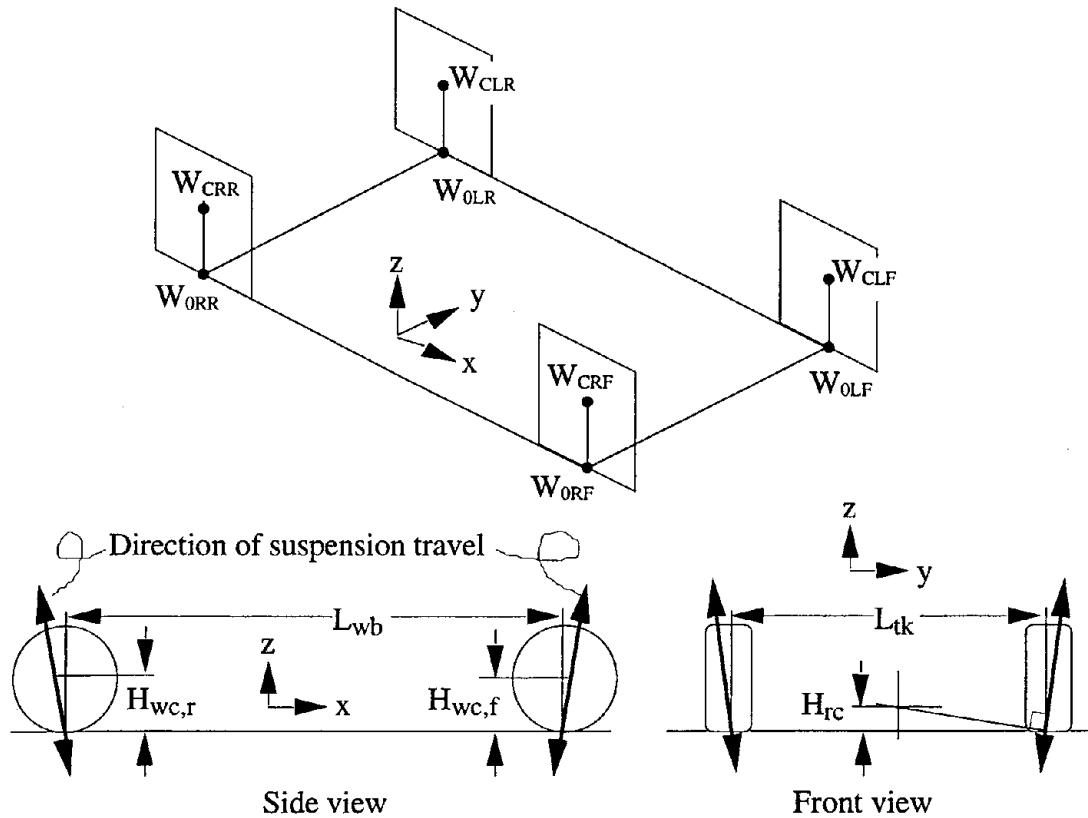


Figure F.2. Locations and movements of wheels.

Ignoring, for the moment, the influence of compliances in the suspension and steering system linkages, each wheel center of a real vehicle follows a trajectory through 3D space, relative to the car body, as the suspension moves up and down. Due to the kinematics of the suspension, the trajectory is usually not purely vertical. For most vehicles, the wheels move out laterally as the suspensions are compressed, such that track width increases with suspension compression. The wheels also move out longitudinally, such that wheelbase increases with suspension compression.

The direction of the wheel trajectory (relative to the main body) determines how tire shear forces in the ground plane are transmitted to the vehicle body through reaction forces in the suspension linkages. In hand-written equations, roll and pitch moments due to suspension reaction forces have been written with coefficients with names such as anti-roll, anti-pitch, anti-dive, anti-squat, and jacking [1,2].

The multibody model accounts for the interaction between tire shear forces and roll and pitch moments so long as the movement is constrained to follow the proper path. A

simple approximation is to assume the movement is in a straight line, as shown in Figure F.2. Using an axis system based in the vehicle sprung mass (s_x , s_y , s_z), the directions of the movements of the four wheels are:

$$\begin{aligned} \text{left-front: } \text{dir}(s_z + \frac{2H_{rc,f}}{L_{tk,f}} s_y + R_{p,f} s_x) & \text{right-front: } \text{dir}(s_z - \frac{2H_{rc,f}}{L_{tk,f}} s_y + R_{p,f} s_x) \\ \text{left-rear: } \text{dir}(s_z + \frac{2H_{rc,r}}{L_{tk,r}} s_y - R_{p,r} s_x) & \text{right-rear: } \text{dir}(s_z - \frac{2H_{rc,r}}{L_{tk,r}} s_y - R_{p,r} s_x) \end{aligned} \quad (1)$$

where secondary subscripts f and r indicate parameters for the front and rear.

In the basic suspension analyses, the roll kinematics are analyzed to define a point called a roll center [5]. For compatibility with this convention, the inclination of the wheel movement in the roll direction is defined by the ratio of the roll center height to the half-track distance. For pitch, a single coefficient (R_p) is used.

Masses and Inertias

The user of a vehicle model must provide mass and inertia parameters for the bodies in the model. For the wheel bodies, one may set the moments of inertia to zero, and locate the mass centers at the wheel centers, nominally a height H_{wc} above the ground. The mass of each wheel body should be set to that portion of vehicle mass supported by the tire that is considered to move with the wheel. This value is commonly called the unsprung mass, and usually includes some of the mass of the suspension elements.

The mass of the main body, called the sprung mass, must be set to the mass of the entire vehicle minus the unsprung masses. The inertia properties are also required, including the XZ product of inertia. (Due to lateral symmetry, the XY and YZ products are usually zero.) It is much easier to measure inertia properties for the entire vehicle than for the body alone. The multibody program can be made to calculate the mass and inertia properties of the sprung mass from measurements made for the entire vehicle. This is done by adding four more bodies and giving them negative masses. These four bodies should be placed at the same locations as the wheel body mass centers. However, their masses should be set to the negative values of the unsprung masses, and they should be fully constrained with respect to the main vehicle body (i.e., zero DOF). The multibody program, in accounting for the full constraint of these four bodies, will in effect subtract the masses and inertia properties, bringing the mass and inertias of the main body down to those of the sprung mass alone.

Suspension Force Effects

Movement of a wheel along the line of motion allowed by the suspension kinematics is affected by suspension springs, dampers, bump stops, and anti-sway bars. In each case,

some of the force generated by a component (e.g., a spring) acts to move the wheel, affecting the transfer of mechanical energy to and from the sprung mass. In addition, some of the force is reacted at other points or in other directions that do not move and therefore cannot affect the transfer of mechanical energy. For example, consider the spring shown in the suspension of Figure F.3. If the wheel moves vertically an amount of Δ relative to the body, the spring is compressed by a lesser amount, say, for example, $R_s \Delta$, where R_s is a coefficient that defines the mechanical advantage of the spring relative to the wheel. The spring exerts a force F_s on the lower control arm. Some of the force is reacted at the connection to the body, and some is reacted at the wheel by the vertical tire force, as shown in the figure. Conservation of work requires that the change in force at the wheel center multiplied by its movement must be equal to the change in spring force, multiplied by its change in compression. Thus, the effect of the spring at the wheel is $R_s F_s$. A similar analysis can be made for the damper, using a different ratio R_d .

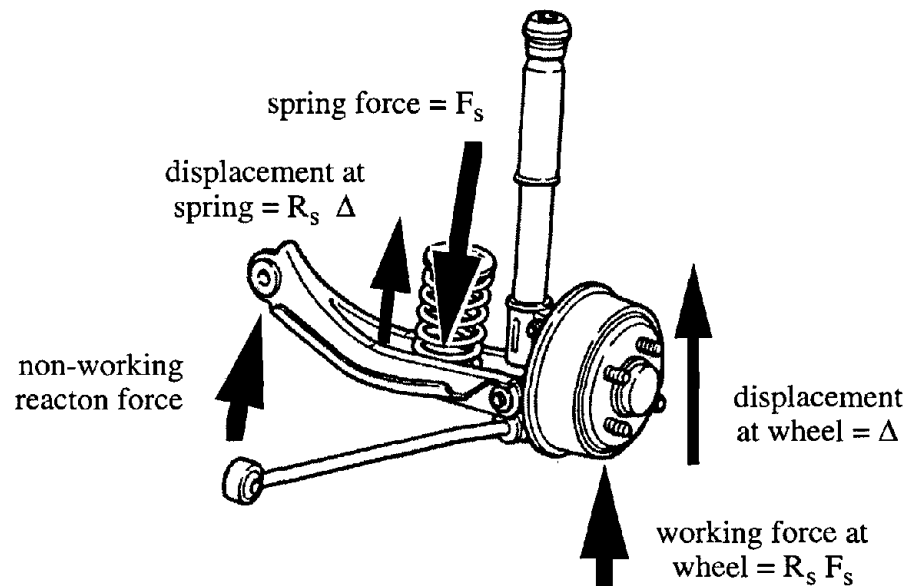


Figure F.3. Mechanical advantage of suspension component.

This principle of mechanical advantage is used to include components such as springs and dampers in the vehicle dynamics model without directly modeling their points of attachment or the complex suspension linkage geometry. The effect of a suspension component at the wheel is calculated in three steps:

1. multiply the suspension compression (measured at the wheel) by the kinematic ratio to determine the compression at the component,
2. apply a known functional relationship (e.g., spring force vs. compression) to determine the force generated by the component, and

- multiply the component force by the kinematic ratio to obtain an effective vertical force at the wheel.

For a linear spring, the three steps can be combined to define an effective spring rate at the wheel: $K_s R_s^2$. For nonlinear relations, it is necessary to perform all three steps. For example, consider the addition of the suspension spring for the left-front wheel. In AUTOSIM, the force is added with a command called `add-line-force`, with arguments shown in Table 1 for the spring force.

A similar treatment is made for the shock absorber, using the derivative of the suspension displacement, the damper mechanical advantage R_d , and a functional relation between damper force and stroke rate. In general, different geometric ratios are needed for the suspension spring, the damper, and the bump stop. Different ratios are also used for front and rear, but the same ratios are used between left and right wheel on the same axle.

Table 1. Adding a spring force to the model.

Property	Value	Description
Point 1	P1: Point on sprung mass	Point on which force acts
Point 2	P2: Point on wheel	Point in body taking the reaction force
Direction	s_z	Direction of force vector
Magnitude	$R_s F_s(R_s s_z \cdot \text{pos}(P1, P2))$	Magnitude of force vector

The effect of the anti-sway bar is modeled with a linear spring between the two wheels linked by the bar. The two points in the `add-line-force` command are on the two wheels, the direction of the force is s_z , and the magnitude is a spring rate multiplied by the vertical movement difference between the two points.

References

- Jindra, F. "Mathematical Model of Four-Wheeled Vehicle for Hybrid Computer Vehicle Handling Program," National Highway Traffic Safety Administration, DOT HS 801800, 1976.
- R. Wade Allen, Henry T. Szostak et. al., Vehicle Dynamic Stability and Rollover, National Highway Traffic Safety Administration, DOT HS 807 956, June 1992.
- Kortüm, W. Sharp, R.S., eds., *Multibody Computer Codes in Vehicle System Dynamics*, Lisse, Swetz and Zeitlinger, 1993.
- Sayers, M.W., "Symbolic Vector/Dyadic Multibody Formalism for Tree-Topology Systems." *Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 6, Nov/Dec 1991, 1240-1250.
- Gillespie, T.D., *Fundamentals of Vehicle Dynamics*, SAE, Warrendale, PA, 1992.

Appendix G — Tire Models

This appendix describes the tire models used in VDM RoAD.

The interaction of the tires with the ground is modeled by longitudinal force (F_x), lateral force (F_y), vertical force (F_z), and aligning moment (M_z). F_x , F_y , F_z , and M_z are applied to the axles and reacted by the ground, as shown in Figure G.1. Overturning moment (M_x) and rolling resistance moment (M_y) have insignificant influence on vehicle braking and handling behavior and are ignored. F_z is determined by assuming the tire behaves as a linear spring-damper when deformed vertically. F_x , F_y , and M_z are more complicated to compute. Their calculation can be broken into two basic steps.

1. F_x , F_y , and M_z are modeled as table functions of two independent variables. The data for these table functions is supplied by the VDM RoAD user in the form of carpet plot data. Linear interpolation/extrapolation is used to evaluate the table functions. F_x is modeled as a function of F_z and longitudinal slip rate (κ), $F_x(F_z, \kappa)$. F_y and M_z are modeled as table functions of F_z and lateral slip angle (α), $F_y(F_z, \alpha)$, and $M_z(F_z, \alpha)$, respectively.
2. Combined slip theory is used to adjust the values of F_x , F_y , and M_z (as computed in step 1) to account for the force and moment coupling that occurs during maneuvers in which there is both braking and turning.

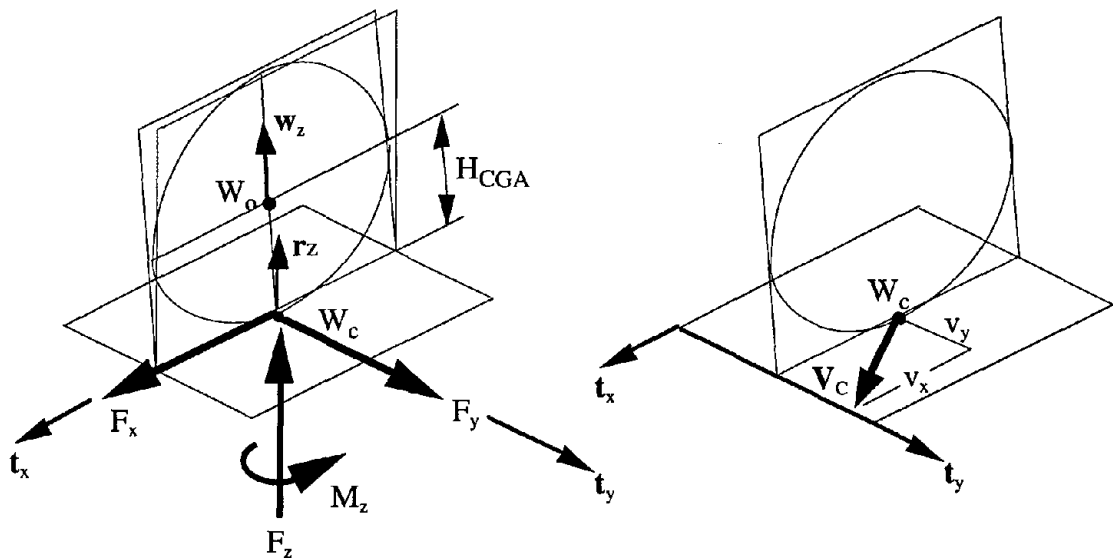


Figure G.1. Tire points and axes.

Note: F_x , F_y , and M_z also depend upon tire inclination (γ). However, with the solid axle suspensions of most trucks, inclination angles

are determined by the roll of the axle relative to the ground and are very small prior to first tire lift-off. Thus, the effect of inclination angle is neglected in VDM RoAD models.

Tire/Wheel Kinematics

Most of the arithmetic operations in the expressions for tire forces and moments arise from deriving expressions for various kinematical quantities associated with the tires/wheels. It is necessary to (1) define a point where the tire forces act on the multibody model, (2) establish the unit vector directions for the tire X and Y directions (\mathbf{t}_x and \mathbf{t}_y , respectively) given the ground normal (\mathbf{r}_z), and (3) determine expressions for κ and α . These steps are explained below. The expressions are given for an arbitrary \mathbf{r}_z as a function of the location of the center of tire contact. However, for a flat, level surface, \mathbf{r}_z is simply \mathbf{n}_z .

Center of Tire Contact

The tire model in VDM RoAD (and most other tire models) makes use of a point called the center of tire contact (CTC) for the definition of the location of the tire on the ground, the calculation of slip angles, and the point of application of tire forces. In the nominal configuration, in which the vehicle is at rest in equilibrium on a flat, level surface, CTC corresponds to the point W_C shown in Figure G.1.

CTC remains in the ground plane as the wheel moves up and down. For a flat surface, the kinematical expressions for calculating the position and velocity of CTC are well defined, but extremely complex. In nearly all vehicle dynamics models that have been developed by hand (without multibody programs), approximations to CTC for a flat surface have been used. The exact expressions can be obtained easily with the aid of a multibody program. However, the complexity of the exact equations requires many mathematical operations per time step during numerical integration. Most of the complexity arises from terms that are negligible (such as small changes in the vertical position of CTC in the frame of the axle (body A), and vertical components of the velocity of CTC that contribute little to the slip angle).

When the ground surface is permitted to take on an arbitrary shape, calculating the location of CTC becomes even more complex. This is because \mathbf{r}_z is a function of the global location of CTC, but at the same time, \mathbf{r}_z determines the instantaneous tire deflection, which in turn affects the location of CTC. (The general solution of the location of CTC on an arbitrarily shaped surface requires an iterative solution at each time step.) To obtain shorter run times and avoid the complexities associated with the precise definition of CTC, VDM RoAD uses a point W_C fixed in body A, as described below.

Wheel Plane

Each tire on an axle has an associated wheel plane that is perpendicular to the wheel spin axis, \mathbf{w}_Y . For the front axle, the wheels are steered with angles δ_L and δ_R relative to the axle Y axis. The spin axis for a wheel is:

$$\mathbf{w}_Y = \cos(\delta) \mathbf{a}_Y - \sin(\delta) \mathbf{a}_X \quad (1)$$

For all axles but the front, the steer is zero, which makes \mathbf{w}_Y the same as \mathbf{a}_Y .

Point W_C , fixed in body A, is located in the wheel plane and is nominally coincident with the precise definition of CTC (see Figure G.1). The X, Y, and Z coordinates of W_C in the frame of body A are:

$$\begin{aligned} W_{CX} &= 0 \\ W_{CY} &= \pm \frac{L_{TK}}{2} \pm \frac{L_{DUAL}}{2} \end{aligned} \quad (2)$$

$$W_{CZ} = -H_{CGA}$$

Here, L_{TK} is the axle track to the center plane of the dual-tire set and L_{DUAL} is the dual tire spacing. The algebraic signs of the terms in the Y coordinate depend on the side of the axle (left: $+\frac{L_{TK}}{2}$, right: $-\frac{L_{TK}}{2}$) and whether the wheel plane is located in the inner or outer tire (inner: $-\frac{L_{DUAL}}{2}$, outer: $+\frac{L_{DUAL}}{2}$).

Ground Geometry

The coordinates of the absolute position of W_C in the global frame N are:

$$\begin{aligned} X_C &= \mathbf{n}_X \cdot \text{pos}(W_C) \\ Y_C &= \mathbf{n}_Y \cdot \text{pos}(W_C) \\ Z_C &= \mathbf{n}_Z \cdot \text{pos}(W_C) \end{aligned} \quad (3)$$

VDM RoAD models are based on the assumption of a ground surface that is smooth and level. However, the formulation is easily extended to work with a 3D ground surface, where the ground normal \mathbf{r}_Z is determined as a function of the location of point W_C :

$$\mathbf{r}_Z = X_{RZ} \mathbf{n}_X + Y_{RZ} \mathbf{n}_Y + Z_{RZ} \mathbf{n}_Z \quad (4)$$

where

$$[Z_{GROUND}, X_{RZ}, Y_{RZ}, Z_{RZ}] = \text{func}(X_C, Y_C) \quad (5)$$

and $\text{func}(X_C, Y_C)$ is a function of X_C and Y_C that returns the four variables shown in braces.

Vertical Tire Deflection

Vertical tire deflection (Δ_T), and its time derivative, are needed to calculate F_z . Δ_T is the distance between point W_c and the road, in the plane of the wheel:

$$\Delta_T = \frac{(Z_{\text{GROUND}} - Z_C) \mathbf{r}_z \cdot \mathbf{n}_z}{\mathbf{r}_z \cdot \mathbf{a}_z} \quad (6)$$

F_x , F_y , and F_z act on the axle through a point that is nominally coincident with W_c , but which moves vertically by the displacement Δ_T in the \mathbf{a}_z direction. This movement of the force application point provides a slight correction to the moment arm of the forces.

Tire Axes

The tire X and Y axes, \mathbf{t}_x and \mathbf{t}_y , lie in the plane of the road ($\mathbf{t}_z = \mathbf{r}_z$). \mathbf{t}_x is defined as being perpendicular to the wheel spin axis. Thus,

$$\mathbf{t}_x = \frac{\mathbf{w}_y \times \mathbf{r}_z}{|\mathbf{w}_y \times \mathbf{r}_z|} \quad \mathbf{t}_y = \frac{\mathbf{r}_z \times \mathbf{t}_x}{|\mathbf{r}_z \times \mathbf{t}_x|} \quad (7)$$

Wheel Spin and Longitudinal Slip Rate

The wheel spin, ω , is described with a differential equation:

$$\dot{\omega} = - \frac{F_x (H_{\text{CGA}} - \Delta_T) + \text{sign}(M_{\text{YBK}}, \omega)}{I_w} \quad (8)$$

where $H_{\text{CGA}} - \Delta_T$ is the instantaneous tire radius (moment arm of F_x about the wheel center), M_{YBK} is the braking torque, and I_w is the polar moment of inertia of the spinning wheel.

κ is calculated as

$$\kappa = \frac{\omega H_{\text{CGA}}}{V_x} - 1 \quad (9)$$

The wheel spin is integrated to calculate the wheel rotation angle needed for making animations of the rotating wheels.

Lateral Slip Angle

The definition of α for each tire is based on the velocity of point CTC. The slip angles of the inner and outer tires of a dual-wheel assembly are nearly the same. Therefore, a single slip angle is defined for both. It is developed using the velocity of a point P_{slip} , defined for a wheel plane midway between the two wheel planes of the dual wheel assembly. The coordinates of P_{slip} in the frame of body A are:

$$P_{\text{slip}X} = 0 \quad P_{\text{slip}Y} = \pm \frac{L_{TK}}{2} \quad P_{\text{slip}Z} = -H_{CGA} \quad (10)$$

The slip angle is defined in terms of the X and Y velocity of P_{slip} , V_X and V_Y , expressed in the ground plane:

$$V_X = t_x \cdot \text{vel}(P_{\text{slip}}) \quad V_Y = t_y \cdot \text{vel}(P_{\text{slip}}) \quad (11)$$

where t_x and t_y are defined for the midway wheel plane that contains point P_{slip} . The instant slip angle is the arc-tangent of the ratio: V_Y / V_X .

Tire Relaxation

Tires develop shear forces in response to deformation of the tire structure. The forces do not develop instantaneously, but build as the tire rolls [2]. For example, Figure G.2 shows how F_Y builds in response to a step change in slip angle.

Two methods are commonly used for including the tire lag in a vehicle model: (1) use a tire model with the dynamics built in, or (2) use a static (steady-state) tire model with a separate filter to account for the lag. The second approach is used in VDM RoAD because it offers two practical advantages. First, it allows the use of any static tire model from the literature independently of the method used to introduce lag. Second, it simplifies the calculation of the kinematical variables used as inputs to the tire model. Lag is introduced into the slip angle such that the instantaneous response calculated for the lagged slip angle yields the lagged side force and aligning moment.

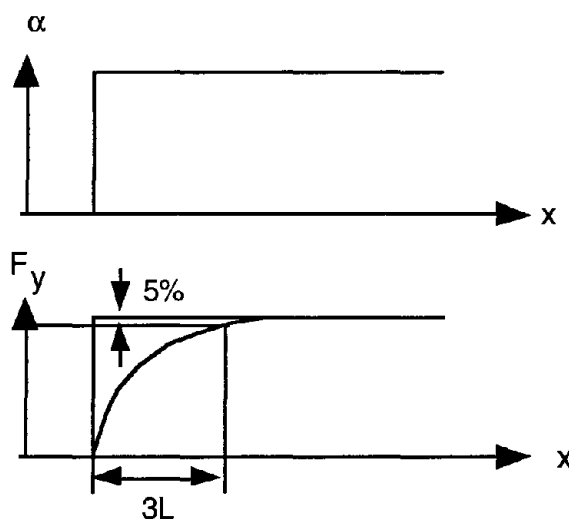


Figure G.2. Tire relaxation.

A method described by Bernard [1] is used to account for the lag in tire response. In this method, α is defined as the arc tangent of an auxiliary state variable, τ .

$$\alpha = \tan^{-1}(\tau) \quad (12)$$

A state variable is added for each wheel and defined with a first-order differential equation:

$$\begin{aligned} \frac{d\tau}{dt} &= \frac{|V_x|}{L_{RELAX}} \left(\frac{V_y}{|V_x|} - \tau \right) \lambda \\ &= \frac{V_y - |V_x| \tau}{L_{RELAX}} \lambda \end{aligned} \quad (13)$$

$$\text{where } \lambda = \begin{cases} 1, & |V_x| \geq V_{CUTOFF} \\ \min \left(50, \frac{V_{CUTOFF}}{\max(|V_x|, \epsilon)} \right), & |V_x| < V_{CUTOFF} \end{cases}$$

The parameter L_{RELAX} is the tire relaxation length. The absolute value of V_x is used in Equation 13 to maintain continuity in case the vehicle spins out and V_x assumes a negative value. The factor λ is introduced to avoid peculiar behavior at near-zero speeds. V_{CUTOFF} is a cut-off speed that can be set by the user, and ϵ is a small positive number used to avoid division by zero.

Tire Forces and Moments

Once expressions are obtained for the needed kinematical quantities, the tire force and moment magnitudes can be calculated. F_z must be calculated first, because F_x , F_y , and M_z depend upon it.

Vertical Tire Force

Vertical tire force is proportional to the tire deflection:

$$F_z = \max(0, F_{ZSTAT} + K_T \Delta_T) \quad (14)$$

where F_{ZSTAT} is the static axle load divided by the number of tires (2 or 4) and K_T is the vertical tire stiffness.

The above definition of F_z establishes the force to be in equilibrium when the vehicle is in the nominal configuration. The use of the \max function prevents the magnitude of the force from going negative when the tire leaves the ground.

VDM RoAD models also place a linear damping force in parallel with the above spring force to help smooth the simulated vehicle response. The force is implemented with an IF statement so that when F_z given in Equation 14 is less than a threshold, the damping force is zero. This is done to eliminate damping when the tire is off the ground. The damping rate is set internally to $K_T/2000$ and cannot be changed by the user.

Combined Slip Theory

Under a pure longitudinal or lateral slip condition, F_x , F_y , and M_z are equivalent to the result of simple linear interpolation of the carpet plot data. When there is both longitudinal and lateral slip, the combined slip theory of Pacejka's Magic Formula is used to modify the values obtained by table lookup (see Equations 88-93 in [3]).

First the longitudinal and lateral slips are combined to get total theoretical (combined) slip:

$$\sigma_{\text{total}} = \sqrt{\sigma_x^2 + \sigma_y^2} \quad (15)$$

where

$$\sigma_x = -\frac{\kappa}{1+\kappa} \quad \sigma_y = \frac{\tan(\alpha)}{1+\kappa} \quad (16)$$

The theoretical slips are then normalized by peak slip values, $\sigma_{x\text{max}}$ and $\sigma_{y\text{max}}$. Peak slip values are those that cause peak F_x and F_y . They are functions of F_z and are obtained by linear interpolation and a search algorithm.

$$\sigma_{\text{total}}^* = \sqrt{\sigma_x^{*2} + \sigma_y^{*2}} \quad (17)$$

where

$$\sigma_x^* = \frac{\sigma_x}{\sigma_{x\text{max}}} \quad \sigma_y^* = \frac{\sigma_y}{\sigma_{y\text{max}}} \quad (18)$$

The equivalent longitudinal and lateral slips are calculated from the normalized total theoretical slip,

$$\kappa' = \frac{\sigma_{\text{total}}^* \cdot \sigma_{x\text{max}} \cdot \text{sign}(\sigma_x)}{1 + \sigma_{\text{total}}^* \cdot \sigma_{x\text{max}} \cdot \text{sign}(\sigma_x)} \quad (19)$$

$$\alpha' = \tan^{-1}[\sigma_{\text{total}}^* \cdot \sigma_{y\text{max}} \cdot \text{sign}(\sigma_y)] \quad (20)$$

Using the equivalent longitudinal and lateral slips, the so-called "base-curves" are obtained by means of linear interpolation of the tabular data.

$$\begin{aligned} F_{x0} &= FX(F_z, \kappa') \\ F_{y0} &= FY(F_z, \alpha') \end{aligned} \quad (21)$$

The base-curves are then modified to account for the anisotropic properties of the tire-road friction.

$$\begin{aligned} F_{X0}^* &= F_{X0} - \varepsilon(F_{X0} - F_{Y0}) \cdot \left(\frac{\sigma_Y^*}{\sigma_{total}^*} \right)^2 \\ F_{Y0}^* &= F_{Y0} - \varepsilon(F_{Y0} - F_{X0}) \cdot \left(\frac{\sigma_X^*}{\sigma_{total}^*} \right)^2 \end{aligned} \quad (22)$$

where $\varepsilon = \sigma_{total}^*$ for $\sigma_{total}^* < 1$ and $\varepsilon = 1$ for $\sigma_{total}^* > 1$.

The longitudinal and lateral tire forces are then calculated by

$$\begin{aligned} F_X &= F_{X0}^* \cdot \frac{\sigma_X}{\sigma_{total}} \\ F_Y &= F_{Y0}^* \cdot \frac{\sigma_Y}{\sigma_{total}} \end{aligned} \quad (23)$$

and the aligning moment is calculated by

$$M_Z = \frac{MZ(F_Z, \alpha')}{F_{Y0}} \cdot |F_Y| \quad (24)$$

STI Tire Model

The VDM RoAD models include two methods for calculating F_x , F_y , and M_z . In addition to the method described above, a version of the STI model is also supported.

The STI tire model is a modified version of the Calspan tire model. It deals with combined slip and takes the longitudinal slip rate κ , lateral slip angle α , tire camber angle γ , and vertical tire load F_z as independent variables. It is an empirical model built around a polynomial *Saturation Function*. The implementation of the STI tire model is based on a set of formulae in a DOT report. [4]

Sequence Of Calculations

The vehicle simulation is run by numerically integrating a set of ordinary differential equations. At the beginning of each time step, the values of all of the state variables are known, including the τ variables. The various quantities necessary for the tire equations are calculated in the following sequence:

1. Compute F_z , which depend only on the state variables.
2. Compute F_x , F_y , and M_z with the static tire model (table lookup/combined slip theory) using τ .

3. Compute δ , including the effect of steer compliances coupled with tire actions of F_x , F_y , and M_z .
4. Determine t_x and t_y .
5. Calculate terms that depend on t_x and t_y :
 - Apply forces F_x and F_y to the axles.
 - Calculate the time derivatives of τ .

Notes on Sign Conventions

- Positive κ generates positive F_x .
- Positive α generates negative F_y .
- Positive (small) α generates positive M_z .

References

1. Bernard, J.E., Clover, C.L., "Tire Modeling for Low-speed and High-speed Calculations," SAE Paper No. 950311, 1995.
2. Gillespie, T.D., *Fundamentals of Vehicle Dynamics*, SAE, Warrendale, PA, 1992.
3. Pacejka, H., et al. "Shear Force Development by Pneumatic Tyres in Steady State Conditions: A Review of Modelling Aspects." *Vehicle System Dynamics*, Vol. 20, 1991.
4. Szostak, H.T., et al. "Analytical Modeling of Driver Response in Crash Avoidance Maneuvering Volume II: An Interactive Tire Model for Driver/Vehicle Simulation" U.S. DOT Report DOT HS 807 271, 1988.

Appendix H — Road Geometry

This appendix describes the road geometry used in the VDM RoAD models for describing road geometry.

Right-Handed Axis Systems

The describing dynamical equations for a vehicle are time-based differential equations. Solving them provides coordinates of critical points in the vehicle based on a global X-Y-Z coordinate system. The road design data from the IHSDM data file is also tied to a global X-Y-Z coordinate system. The global system is needed by the underlying CAD system to perform calculations and to prepare graphic views of the design.

The IHSDM file, as defined by the A/E Group at the Geometrics Laboratory at TFHRC, includes X, Y, and Z coordinates for transition points along the centerline of a road design.

The VDM RoAD software uses the global system from the IHSDM file to locate the vehicles, linking the vehicle and road data together through the common global coordinate system.

The only restrictions on the coordinate system used in VDM RoAD are the same as in most 3D CAD packages: the X, Y, and Z axes must be orthogonal and must be oriented according to the right-hand rule for axes, illustrated in Figure H.1.

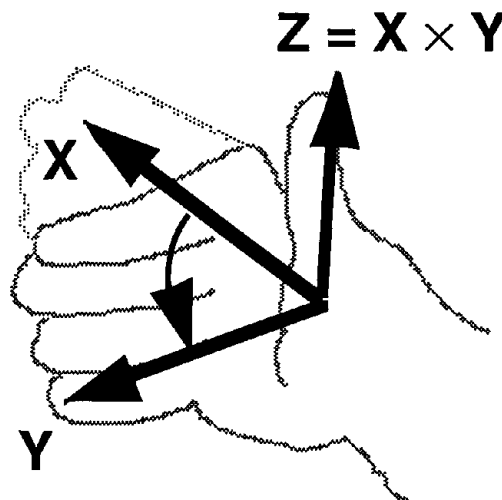


Figure H.1. A right-handed axis system.

Orient your right hand so your fingers point in the direction of the X axis when extended, and point in the direction of the Y axis when folded.

When oriented in this way, your thumb points in the direction of the Z axis.

For example, if Z is defined as pointing up (away from the center of the earth) and X is defined as pointing east, then Y must be pointing north.

The S-L-Z Coordinate System

The global X-Y-Z coordinate system is theoretically sufficient for locating any point of interest in the road, and also any point of interest in the vehicle at a specific instant of time. However, the X-Y-Z system is not necessarily the most convenient. For example, it is desirable to locate the vehicle using station number (S) and lateral distance (L) from the design centerline, rather than global X and Y coordinates.

The data in the IHSDM file define a continuous, curved reference design line (typically the centerline) in 3D space. The 2D horizontal geometry is defined by projecting the 3D line onto the global X-Y plane, as shown in Figure H.2. Station number is defined as the distance along the 2D projection.

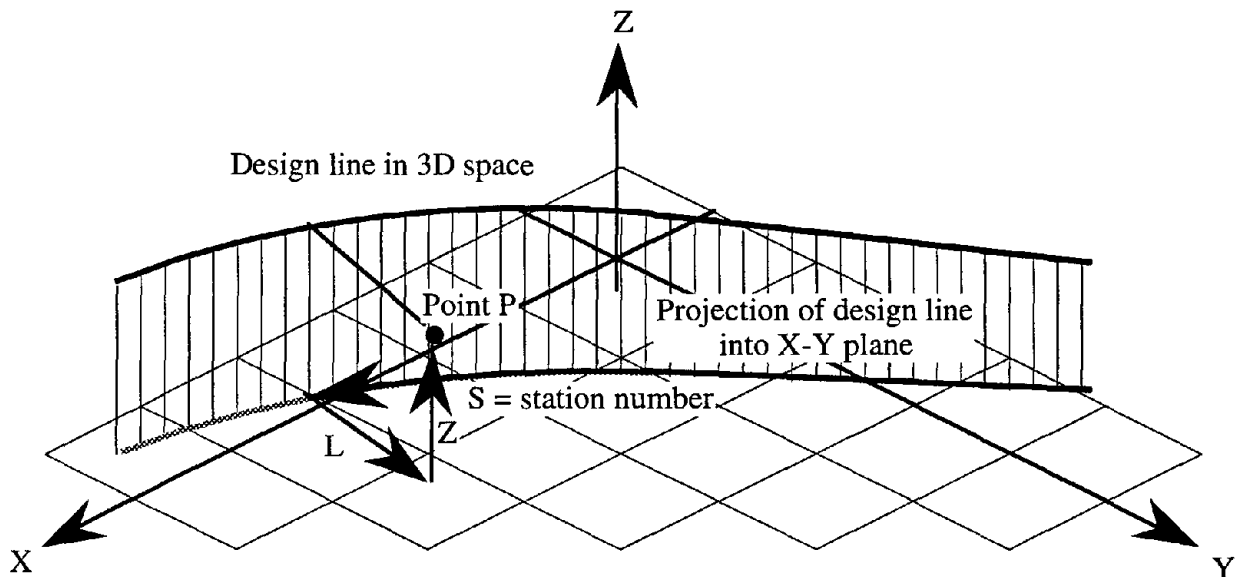


Figure H.2. The S-L-Z coordinate system.

Note: If there is no vertical geometry, the design line and the X-Y projection are one and the same.

The location of an arbitrary point P in 3D space requires a coordinate system with three independent coordinates. For the X-Y-Z coordinate system, the three coordinates are X, Y, and Z, each defined as the distance from the origin along a line parallel with the named axis. To locate point P, one would:

1. start at the origin and then go X meters along the X axis,
2. then go Y meters parallel with the Y axis, and, finally,
3. go up Z meters, parallel with the Z axis.

The road coordinate system in VDM RoAD involves three coordinates named S, L, and Z. The Z coordinate is the same as in the X-Y-Z system. The other coordinates, S and L, are based on the design centerline. To locate point P in the road coordinate system, one would:

1. start on the projection of the road design line into the X-Y plane at station $S=0$ and then travel along the projection of the design line S meters,
2. then go laterally L meters, along a line in the X-Y plane that is perpendicular to the projection of the design line, and finally,
3. go up Z meters, parallel with the Z axis.

The right-hand rule requires that L be positive for positions left of the projection of the design line (facing in the direction of increasing S). Points to the right of the design line have negative L coordinates.

Integration of the IHSDM geometric design data in VDM RoAD requires that X-Y-Z coordinates be translated to S-L-Z, and vice versa. For example, the animations in VDM RoAD require that road geometry, often given in S-L-Z coordinates, are expressed in X-Y-Z coordinates. The computation of tire forces in VDM RoAD require that, given X and Y, Z and the change in Z with X and Y are computed. This requires the intermediate step of translating X-Y coordinates into S-L coordinates before computing Z. The following sections describe the translation between the two horizontal coordinate representations and their use in computation of Z.

Horizontal Geometry

The road design data are read from an IHSDM data file and reduced to the minimum set of intermediate quantities needed to describe horizontal design. These quantities define the road geometry in a manner that requires less computation “in the loop” during the simulations, and eliminates redundancy required when mixing vertical and horizontal design data. A distinct set of these quantities is required each time the road geometry changes. That station number at the point where a change occurs is termed the *critical station*. The quantities are:

- N_h : The number of changes in horizontal road geometry. The quantities that follow are tabulated, with N_h values each.
- S_i : Station number at critical station i. (m)
- X_i : Absolute x-coordinate at station S_i . (m)
- Y_i : Absolute y-coordinate at station S_i . (m)

R_j : Radius of curvature for constant radius curves. (m) A negative value implies a turn to the left. Zero implies no constant radius curve between station S_j and S_{j+1} .

Note: The sign convention for the radius is contrary to the right-handed convention used for the coordinate systems. It is used to maintain compatibility with the radius as defined in the IHSDM files.

S_{pi} : Spiral angle/(spiral length squared). (deg/m/m) (Zero implies no spiral between station S_j and S_{j+1} .)

Note: The VDM RoAD simulation programs do not include the code to interpret spirals yet. However, the spiral data are included to help with compatibility if spirals are added in the future.

COS_{HDj} : Cosine of heading angle at station S_j .

SIN_{HDj} : Sine of heading angle at station S_j .

Translating From S-L to X-Y Coordinates

The process of transforming S-L coordinates into X-Y values is direct. For any pair of S-L values, there is a single pair of corresponding X-Y coordinates. If S (current station) and L (lateral offset from the road centerline) are given, X and Y are computed using the following algorithm. Figure H.3 shows the geometry.

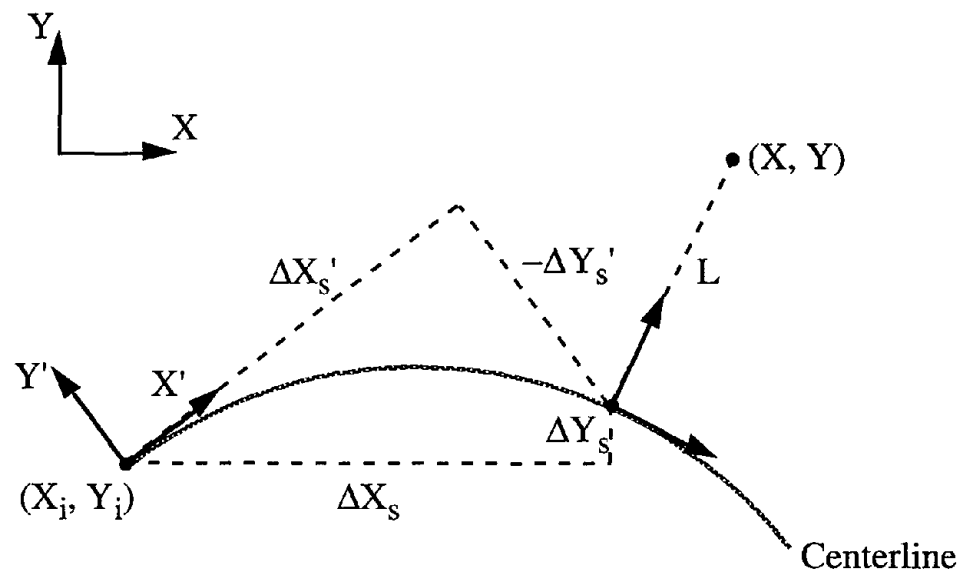


Figure H.3. Computation of X-Y from S-L.

1. Find the index i for the critical station that precedes S . This is done using a simple table-lookup procedure.
2. Compute the distance from the X and Y coordinates of the last critical station (X_i , Y_i) to the point of interest along S in global X and Y coordinates. Store these as ΔX_s and ΔY_s :

2.1 Compute the distance from the last critical station to S and store this as ΔS .

$$\Delta S = S - S_i$$

2.2 Compute distances in local coordinates, oriented in the frame of reference of an observer moving along the road centerline.

Straight Road:

$$\Delta X_s' = \Delta S$$

$$\Delta Y_s' = 0$$

Constant Radius Curve:

$$\Delta X_s' = R_j \cdot \text{SIN}(\Delta S/R_j)$$

$$\Delta Y_s' = R_j \cdot (\text{COS}(\Delta S/R_j) - 1)$$

Spiral:

$$\Delta X_s' = \Delta S - 0.3046174 \cdot 10^{-4} \cdot S_{P_i}^2 \cdot \Delta S^5 + 0.4295915 \cdot 10^{-9} \cdot S_{P_i}^4 \cdot \Delta S^9$$

$$\Delta Y_s' = 0.5817767 \cdot 10^{-2} \cdot S_{P_i} \cdot \Delta S^3 + 0.1265851 \cdot 10^{-6} \cdot S_{P_i}^3 \cdot \Delta S^7$$

2.3 Transform $\Delta X_s'$ and $\Delta Y_s'$ into global coordinates:

$$\Delta X_s = \Delta X_s' \cdot \text{COS}_{HD_i} - \Delta Y_s' \cdot \text{SIN}_{HD_i}$$

$$\Delta Y_s = \Delta X_s' \cdot \text{SIN}_{HD_i} + \Delta Y_s' \cdot \text{COS}_{HD_i}$$

3. Compute the contribution of L (lateral offset from centerline) to X and Y. Store these as ΔX_l and ΔY_l .

3.1 Find $\partial X'/\partial S$ and $\partial Y'/\partial S$ at station S (in local coordinates, oriented in the frame of reference of an observer moving along the road centerline):

Straight Road:

$$\partial X'/\partial S = 1$$

$$\partial Y'/\partial S = 0$$

Constant Radius Curve:

$$\partial X'/\partial S = \text{COS}(\Delta S/R_j)$$

$$\partial Y'/\partial S = -\text{SIN}(\Delta S/R_j)$$

Spiral:

$$\partial X'/\partial S = 1 - 0.1523087 \cdot 10^{-3} \cdot S_{P_i}^2 \cdot \Delta S^4 + 0.38663235 \cdot 10^{-8} \cdot S_{P_i}^4 \cdot \Delta S^8$$

$$\partial Y'/\partial S = 0.17453301 \cdot 10^{-1} \cdot S_{P_i} \cdot \Delta S^2 + 0.8860957 \cdot 10^{-6} \cdot S_{P_i}^3 \cdot \Delta S^6$$

3.2 Transform $\partial X'/\partial S$ and $\partial Y'/\partial S$ into global coordinates:

$$\partial X/\partial S = \partial X'/\partial S \cdot \text{COS}_{HD_i} - \partial Y'/\partial S \cdot \text{SIN}_{HD_i}$$

$$\partial Y/\partial S = \partial X'/\partial S \cdot \sin_{HD_i} + \partial Y'/\partial S \cdot \cos_{HD_i}$$

3.3 Compute ΔX_1 and ΔY_1 :

$$\Delta X_1 = -L \cdot \partial Y/\partial S$$

$$\Delta Y_1 = L \cdot \partial X/\partial S$$

4. Sum all contributions to X and Y in global coordinates:

$$X = X_j + \Delta X_s + \Delta X_1$$

$$Y = Y_j + \Delta Y_s + \Delta Y_1$$

Translating From X-Y to S-L Coordinates

Converting from X-Y coordinates to determine S-L values is not as direct as the first transform. One complication is that there can be multiple values of S-L that correspond to a single point in X-Y space. For example, the center of a design curve could be equally well defined using any value of S along the curve, with the L value equal to the negative radius (the negative sign arises because the S-L convention is right-handed and the IHSDM convention is not).

If X and Y in global coordinates are known, S (current station) and L (lateral offset from the road centerline) are computed using the following step-by-step procedure. This procedure is iterative. If the value of S computed for a given critical station falls outside the range of its geometric characteristics (that is, if S is less than S_i or greater than S_{i+1}), the computations are repeated for the next critical station. Figure H.4 shows the geometry.

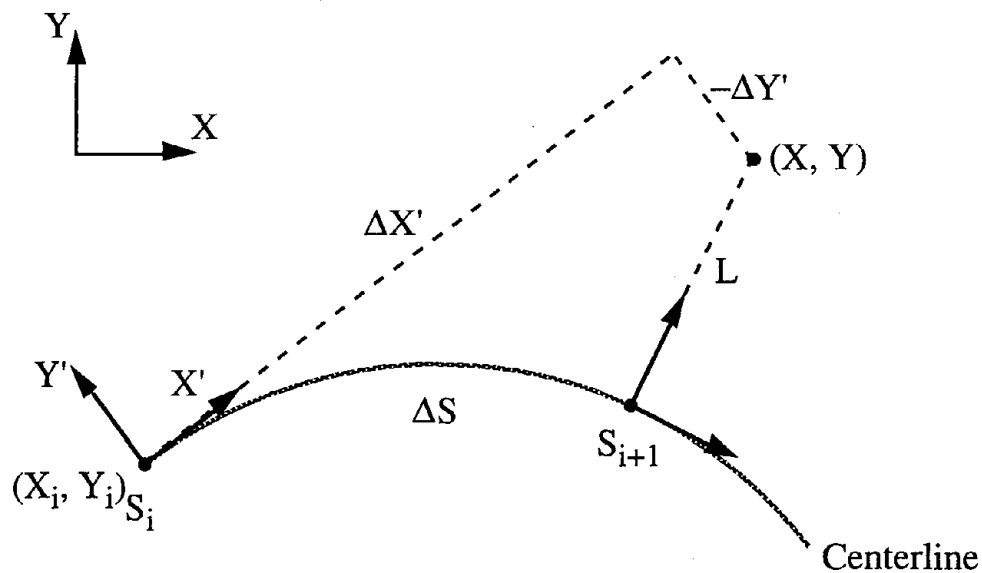


Figure H.4. Computation of S-L from X-Y.

1. Compute the X and Y distance from critical station i to the point of interest in global coordinates.

$$\Delta X = X - X_i$$

$$\Delta Y = Y - Y_i$$

2. Transform the distances to a local coordinate system, oriented in the frame of reference of an observer moving along the road centerline.

$$\Delta X' = \Delta X \cdot \cos_{HDi} + \Delta Y \cdot \sin_{HDi}$$

$$\Delta Y' = -\Delta X \cdot \sin_{HDi} + \Delta Y \cdot \cos_{HDi}$$

3. Compute the value of S implied by the geometry at critical station i .

- 3.1 Resolve $\Delta X'$ and $\Delta Y'$ into ΔS (distance from the last critical station to the station implied by X and Y).

Straight Road:

$$\Delta S = \Delta X'$$

$$L = \Delta Y'$$

Constant Radius Curve:

$$\Delta S = |R_i| \cdot \text{sign}(\Delta X') \cdot \tan^{-1} \left(\frac{\Delta X'}{(|R_i| + \text{sign}(R_i) \cdot \Delta Y')} \right)$$

$$L = \text{sign}(R_i) \cdot \left(\sqrt{(|R_i| + \text{sign}(R_i) \cdot \Delta Y')^2 + \Delta X'^2} - |R_i| \right)$$

Where the *sign* function returns -1 for a negative argument and 1 for a positive argument.

- 3.2 Compute S.

$$S = S_j + \Delta S$$

4. Check S. If S falls within the two critical stations that bound the geometric properties used to compute S and L, stop. If not, increment i and repeat the calculations.

If $S_j \geq S < S_{i+1}$ stop.

If not, set $i = i + 1$ and repeat steps 1-4.

Vertical Geometry

The road design data are read from an IHSDM data file and reduced to the minimum set of intermediate quantities needed to describe vertical design. These quantities define the road geometry in a manner that requires less computation “in the loop” during the simulations, and eliminates redundancy required when mixing vertical and horizontal

design data. The data are split into (1) quantities that define the Z coordinate at every point along the road centerline, and (2) quantities that define changes in elevation as a function of lateral distance (L) from the road center line. A distinct set of these quantities is required each time the road geometry changes.

In order to simulate a vehicle going over a 3D road surface, it is necessary to be able to compute the vertical coordinate of the road surface as a function of X and Y, and to also compute the slope of the road with respect to X and Y ($\partial Z/\partial X$ and $\partial Z/\partial Y$).

The quantities for defining the elevation of the center line are:

N_v : The number of changes in vertical geometry of the road center line. The quantities that follow are tabulated, with N_v values each.

S_{vi} : Station number at critical station i. (m)

Z_i : Absolute z-coordinate at station S_{vi} . (m)

L_{vi} : Vertical curve length. (m) (Zero implies no vertical curvature between station S_{vi} and S_{vi+1}).

B_{gi} : Back grade at station S_{vi} . (m/m) If station S_{vi} begins a vertical curve, this is the grade that precedes the curve.

F_{gi} : Forward grade after station S_{vi} . (m/m) If station S_{vi} begins a vertical curve, this is the grade at the end of the curve (at station $S_{vi} + L_{vi}$).

The quantities for defining changes in elevation with lateral movement from the road center line are:

N_{cs} : The number of changes in cross slope or lane width. The quantities that follow are tabulated, with N_{cs} rows each.

S_{csj} : Station number at critical station i. (m)

N_l : The number of lanes -1. The quantities that follow are tabulated, with N_l columns each.

Thus, for each of N_{cs} critical stations, N_l values of the following quantities must be stored, where $i = 1, N_{cs}$ and $j = 1, N_l$:

$L_{LEj,i}$: L coordinates for lane edges. (m)

$\Delta Z_{LEj,i}$: ΔZ values at lane edges. (m) This represents the vertical climb with lateral movement across each lane.

$(\partial Z/\partial L)_{j,i}$: Cross slope values for each lane. (m/m)

The following quantities are pre-computed to make interpolation of Z and $\partial Z/\partial S$ more efficient:

$(\partial L_{LE}/\partial S)_{j,i}$: Used for interpolating L_{LE} vs. S.

$(\partial \Delta Z_{LE}/\partial S)_{j,i}$: Used for interpolating ΔZ_{LE} vs. S.

$((\partial Z/\partial L)/\partial S)_{j,i}$: Used for interpolating $\partial Z/\partial L$ vs. S .

The procedure for finding Z , $\partial Z/\partial S$, and $\partial Z/\partial L$, given X and Y are:

1. Use X and Y to find S and L using the procedure given above. Retain the index that represents the critical station in horizontal geometry that precedes S as k . (This will be used in step 9.)

2. Find the last critical station that precedes the given S and store its Z coordinate position (Z_i). This is done using a simple table-lookup procedure that yields a value for i .

3. Compute the distance from the last critical station to S and store this as ΔS .

$$\Delta S = S - S_{vi}$$

4. Compute the Z coordinate (Z_s) along the road centerline and $\partial Z/\partial S$:

On a section of constant slope ($L_{vi} = 0$):

$$Z_s = Z_i + B_{gi} \cdot \Delta S$$

$$\partial Z/\partial S = B_{gi}$$

On a vertical curve ($\Delta S < L_{vi}$):

$$Z_s = Z_i + B_{gi} \cdot \Delta S + 0.5 \cdot (F_{gi} - B_{gi}) \cdot \Delta S^2 / L_{vi}$$

$$\partial Z/\partial S = B_{gi} + (F_{gi} - B_{gi}) \cdot \Delta S / L_{vi}$$

Following a vertical curve ($\Delta S \geq L_{vi}$):

$$Z_s = Z_i + F_{gi} \cdot \Delta S + 0.5 \cdot (B_{gi} - F_{gi}) \cdot L_{vi}$$

$$\partial Z/\partial S = F_{gi}$$

5. Find the lane number j implied by L using a simple search routine.

If

$$L > L_{LE N_1,i} + \Delta S \cdot (\partial L_{LE}/\partial S)_{N_1,i}$$

Then $j = N_1$.

Otherwise, set j such that:

$$L_{LE j-1,i} + \Delta S \cdot (\partial L_{LE}/\partial S)_{j-1,i} \leq L < L_{LE j,i} + \Delta S \cdot (\partial L_{LE}/\partial S)_{j,i}$$

6. Compute the cross slope at X , Y .

$$\partial Z/\partial L = (\partial Z/\partial L)_{j,i} + \Delta S \cdot ((\partial Z/\partial L)/\partial S)_{j,i}$$

7. Compute the change in elevation due to lateral offset from the center line.

$$\Delta Z = \Delta Z_{LE j,i} + \Delta S \cdot (\partial \Delta Z_{LE}/\partial S)_{j,i} + (\partial Z/\partial L) \cdot (L_{LE j,i} + \Delta S \cdot (\partial L_{LE}/\partial S)_{j,i})$$

8. Sum up final value for Z :

$$Z = Z_s + \Delta Z$$

9. Transform $\partial Z/\partial S$ and $\partial Z/\partial L$ into global X-Y-Z coordinates.

$$\partial Z/\partial X = \partial Z/\partial S \cdot \cos_{HDk} - \partial Z/\partial L \cdot \sin_{HDk}$$

$$\partial Z/\partial Y = \partial Z/\partial L \cdot \cos_{HDk} + \partial Z/\partial S \cdot \sin_{HDk}$$

Appendix I — The Steer Controller

This appendix describes the theory and application of an algorithm used to control a road vehicle to follow a prescribed path. The technique was developed by Charles MacAdam of UMTRI in 1980 [1, 2] and has been used in various forms in many computer programs since then. In recent research for the Federal Highway Administration (FHWA), the algorithm has been streamlined and re-formulated to work with roadway centerline geometry.

Optimal Control Theory

The algorithm is intended to provide optimal control for a linear system:

$$\dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u} \quad (1)$$

$$\mathbf{y} = \mathbf{C} \mathbf{x} + \mathbf{D} \mathbf{u} \quad (2)$$

where \mathbf{x} is an array of n state variables, \mathbf{u} is a control input, \mathbf{y} is an output variable of interest, and \mathbf{A} , \mathbf{B} , \mathbf{C} , and \mathbf{D} are matrices with constant coefficients. The control objective is to determine the optimal value of \mathbf{u} to causes the predicted output $\mathbf{y}(t)$ to match a target trajectory $\mathbf{y}_{\text{target}}(t)$, over some previewed time T .

In the above equations, \mathbf{A} is an $n \times n$ matrix. In the general case, \mathbf{u} and \mathbf{y} could be arrays involving more than one control and/or output variable. However, in this derivation, the only case considered is the one in which \mathbf{u} and \mathbf{y} are scalars. Thus, \mathbf{B} is a $n \times 1$ matrix and \mathbf{C} is a $1 \times n$ matrix. There is a further simplification, namely, that the output \mathbf{y} does not depend explicitly on \mathbf{u} . Thus, the \mathbf{D} matrix is not used.

If the system has initial conditions \mathbf{x}_0 at time $t=0$ and a constant control input \mathbf{u} , its time response has the known solution:

$$\mathbf{x}(t) = e^{\mathbf{A}t} \mathbf{x}_0 + \int_0^t e^{\mathbf{A}\eta} \mathbf{B} \mathbf{u} \, d\eta \quad (3)$$

The term $e^{\mathbf{A}t}$ is an $n \times n$ matrix called the state transition matrix. Each coefficient in the matrix is the portion of state variable i at time t that is linearly related to state variable j at time 0 . It can be calculated with a power series, with eigenvectors, or by numerical integration. (As will be described later, numerical integration is the method used in the driver model.) The product of the state transition matrix ($e^{\mathbf{A}t}$) and the array of initial conditions (\mathbf{x}_0) is an array of length n with the part of each state variable at time t due to the initial conditions of the system at $t=0$. This is called the free response. The integral defines a contribution to each state variable due to the constant control \mathbf{u} over the time interval. It is called the forced response.

Combining Equations 2 and 3 gives the response of the output variable:

$$y(t) = C x = C e^{At} x_0 + C \left[\int_0^t e^{A\eta} d\eta \right] B u \quad (4)$$

A free-response array F (a $1 \times n$ matrix) is defined to simplify some of the following notation. F relates the state variables at time 0 to the resulting output variable y at time t .

$$F(t) = C e^{At} \quad f_i(t) \equiv \frac{\partial y(t)}{\partial x_i(0)} \quad (5)$$

A control-response scalar g is also defined, to relate the control input u over the interval t to the output variable y at time t .

$$g(t) = C \left[\int_0^t e^{A\eta} d\eta \right] B \quad g(t) \equiv \frac{\partial y(t)}{\partial u} \quad (6)$$

Note that $g(t)$ is related to $F(t)$ by integration:

$$\begin{aligned} g(t) &= \left[\int_0^t C e^{A\eta} d\eta \right] B \\ &= \left[\int_0^t F(\eta) d\eta \right] B \end{aligned} \quad (7)$$

The response equation, re-written using the newly introduced terms, is:

$$y(t) = F(t) x_0 + g(t) u \quad (8)$$

To determine the optimal control, a quadratic performance index J is defined,

$$J = \frac{1}{T} \int_0^T \left\{ y_{\text{target}}(t) - y(t) \right\}^2 W(t) dt \quad (9)$$

where $W(t)$ is an arbitrary weighting function. A control u is considered optimal if it minimizes J —the squared deviation of response variable $y(t)$ relative to the target function $y_{\text{target}}(t)$. Because J is quadratic, the minimum occurs when the derivative $\partial J / \partial u$ is zero. The value of u that minimizes J can be found by substituting Equation 8 into 9 and taking the partial derivative of J with respect to u .

$$J = \frac{1}{T} \int_0^T \left\{ F(t) x_0 + g(t) u - y_{\text{target}}(t) \right\}^2 W(t) dt \quad (10)$$

$$\frac{\partial J}{\partial u} = 0 = \frac{2}{T} \int_0^T \left\{ F(t) x_0 + g(t) u - y_{\text{target}}(t) \right\} g(t) W(t) dt \quad (11)$$

Solving for u gives the following:

$$u = \frac{\int_0^T \left\{ y_{\text{target}}(t) - F(t) x_0 \right\} g(t) W(t) dt}{\int_0^T g(t)^2 W(t) dt} \quad (12)$$

In practice, the integrals over T can be replaced with finite summations:

$$u = \frac{\sum_{i=1}^m \{y_{\text{target}_i} - F_i x_o\} g_i W_i}{\sum_{i=1}^m g_i^2 W_i} \quad (13)$$

where the time dependencies of Equation 12 are replaced with an index i . The meaning here is that index i applied to F , g , W , and y_{target} refers to the value at time $t=iT/m$, where m is an integer number of intervals within the preview time T .

A further simplification will be made in the application below; the arbitrary weighting function will be set to unity. The simplified equation is shown below for future reference.

$$u = \frac{\sum_{i=1}^m \{y_{\text{target}_i} - F_i x_o\} g_i}{\sum_{i=1}^m g_i^2} \quad (14)$$

Application to Vehicle Control

The general control method is programmed in the UMTRI driver model to generate a steering wheel angle in a vehicle simulation program. The algorithm flow is shown in Figure I.1. Given a target path, described in terms of X and Y coordinates, the driver model computes a steering wheel angle given the current state of the vehicle.

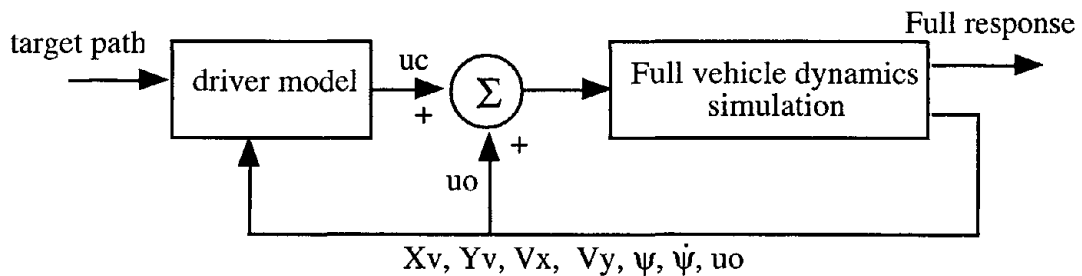


Figure I.1. Algorithm flow.

The vehicle dynamics simulation is usually a complex nonlinear model involving thousands of calculations and tens of degrees of freedom (DOF). It moves on a 3D surface following a prescribed path that may be simple or complex. The start time of the simulation is usually set to zero, and the stop time may be set to anything from a few milliseconds to several minutes.

The complexity of the vehicle simulation is largely hidden from the driver model. In the world of the driver model, the current time is defined as zero, the vehicle is described

with a simple model having two DOF, and the driver control is calculated to optimize the vehicle response over the preview time T . As shown in the figure, the driver model requires only seven variables as feedback. They are: the X and Y coordinates of the front axle of the vehicle (X_v and Y_v), the vehicle-based longitudinal and lateral components of the velocity vector (V_x and V_y), the yaw angle and yaw rate (ψ and $\dot{\psi}$), and the steer control due to factors outside the driver's direct control (u_o). (The term u_o usually represents steering due to suspension kinematics and compliance.)

The driver control algorithm can be divided into three types of calculations:

1. it synthesizes the target path over the preview time,
2. it calculates the optimal steer u to minimize deviations of the path of a point (the center of the front axle) from a target path and subtracts u_o to obtain the steering needed by the driver, u_c , and
3. it delays the driver steering control u_c by a constant time τ_d , to simulate a well-known dynamic characteristic of human drivers.

Station

The target path is normally provided as a sequence of N_{pts} X and Y coordinates. Station (also called station number) is the distance along a reference line, typically a road centerline. In a road design, station is a spatial independent variable. For any given value of station, there is a unique set of X , Y , and Z coordinates. There may also be other values associated with that position, such as superelevation, road width, etc.

A simplification is made in the driver model that the path is composed of straight lines connecting points defined as a sequence of X - Y values. The X - Y values are specified at run time. The starting value of S is set to match the initial X value:

$$S_1 \equiv X_1 \quad (15)$$

Subsequent values, from $i=2$ to N_{pts} , are calculated with the recursive equation:

$$S_i = S_{i-1} + [(X_i - X_{i-1})^2 + (Y_i - Y_{i-1})^2]^{1/2} \quad (16)$$

This calculation is done when the X and Y values are read as inputs. After the calculation, each point in the path is represented with three numbers (S , X , Y) instead of two (X , Y). Because the path connecting two points is assumed to be a straight line, the partial derivatives $\partial X/\partial S$ and $\partial Y/\partial S$ can be calculated with finite-difference equations:

$$\begin{aligned} \left. \frac{\partial X}{\partial S} \right|_i &= \frac{\Delta X_i}{\Delta S_i} = \frac{X_{i+1} - X_i}{S_{i+1} - S_i} \\ \left. \frac{\partial Y}{\partial S} \right|_i &= \frac{\Delta Y_i}{\Delta S_i} = \frac{Y_{i+1} - Y_i}{S_{i+1} - S_i} \end{aligned} \quad (17)$$

The forward-difference form of Equation 17 means that the i^{th} derivative applies for the line segment after the i^{th} point (between points i and $i+1$). Table lookup routines are used to provide X , Y , $\partial X/\partial S$, and $\partial Y/\partial S$ as functions of S at various places in the driver

model. For station numbers outside the range of the table, the last values of $\partial X/\partial S$ and $\partial Y/\partial S$ should be held constant to extrapolate X and Y as needed.

Because S is treated as the independent variable for determining the current path geometry, it is essential to know the current value of S . Figure I.2 shows the relationship between the last station number S_0 , with corresponding coordinates X_0 and Y_0 , and the current vehicle location as defined by the coordinates X_v and Y_v .

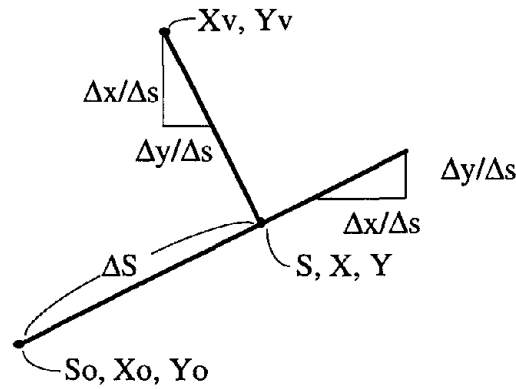


Figure I.2. Geometry used to calculate new station.

The new station S is

$$S = S_0 + \Delta S \quad (18)$$

where

$$\Delta S = \frac{(X_v - X_0) \frac{\Delta x}{\Delta s} + (Y_v - Y_0) \frac{\Delta y}{\Delta s}}{\left(\frac{\Delta x}{\Delta s}\right)^2 + \left(\frac{\Delta y}{\Delta s}\right)^2} \quad (19)$$

To calculate the optimal steer control u_c with Equation 14, the target position is needed at each point being considered in the summation. The value of S obtained with Equation 18 is the station number of the vehicle at its current position. The station for a target location is

$$S_{\text{target},i} = S + \frac{iV_x T}{m} \quad (20)$$

where $i = 1, \dots, m$, and V_x is the forward vehicle speed, assumed within the world of the driver model to be constant.

Target Position

The controller calculations are made using a special axis system, shown in Figure I.3. At time $t=0$, the origin of the X and Y axes coincide with the center of the vehicle front axle, and the axes are aligned with the longitudinal and lateral axes of the vehicle. In the world of the driver controller, the vehicle movements will be predicted relative to these axes. The axes are fixed in the inertial reference, and are rotated from the inertial axes by the vehicle yaw angle, ψ . The initial lateral displacement of the vehicle and the initial yaw angle are both zero in the driver model axis system.

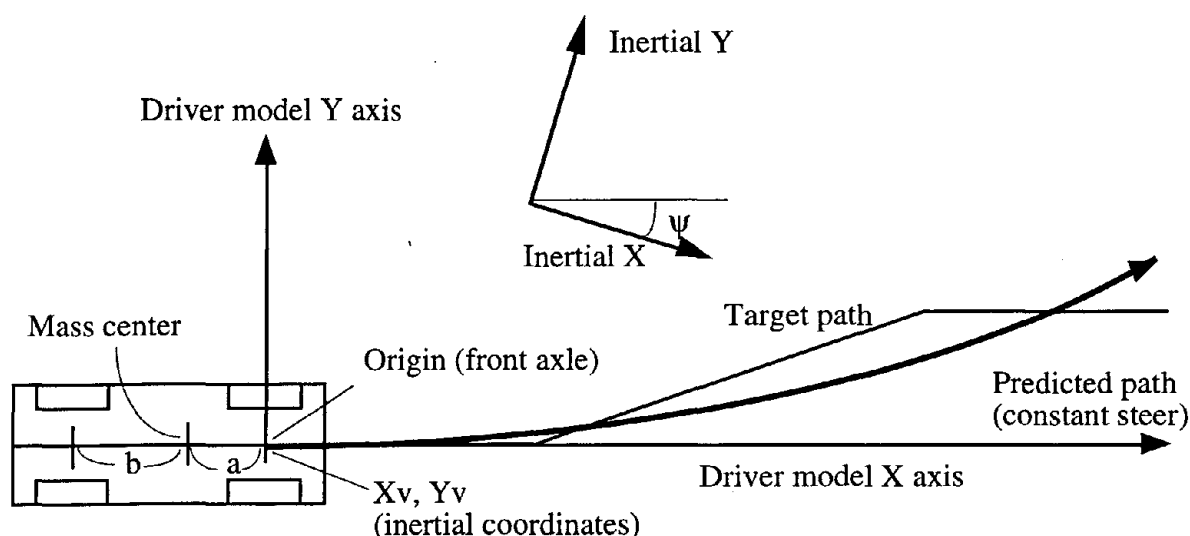


Figure I.3. Axis system of driver model.

The target lateral translation in this coordinate system is calculated by first getting the inertial X and Y coordinates of the path as functions of the station at the target location (S_{targ}), and then applying the transformation

$$Y_{\text{targ}} = [Y(S_{\text{targ}}) - Y_V] \cos(\psi) - [X(S_{\text{targ}}) - X_V] \sin(\psi) \quad (21)$$

Calculating the Optimal Control

Within the world of the driver model, the vehicle always is located at the origin of the axes shown in Figure I.3, the time is 0, and the target path is known from time zero to the preview time, T . The vehicle speed (V_x) is constant and the vehicle is assumed to be described by four state variables:

- $x_1 = Y$ coordinate of the vehicle mass center, in the driver model axis system,
- $x_2 = \text{Yaw angle of vehicle, in the driver model axis system,}$
- $x_3 = V_y$, the lateral component of velocity in the vehicle axis system, and
- $x_4 = \text{Yaw rate.}$

The A and B matrices are defined as follows:

$$A = \begin{bmatrix} 0 & 0 & 1 & V_x \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{-(C_f + C_r)}{MV_x} & \frac{C_r b - C_f a}{MV_x} - V_x \\ 0 & 0 & \frac{C_r b - C_f a}{I_{zz} V_x} & \frac{-(C_f a^2 + C_r b^2)}{I_{zz} V_x} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ C_f \\ C_r a \\ I_{zz} \end{bmatrix} \quad (22)$$

where C_f and C_r are tire cornering stiffness coefficients for the front and rear axles, M is the total vehicle mass, a is the distance from the front axle to the mass center, b is the distance from the rear axle to the mass center, V_x is the forward component of vehicle speed, and I_{zz} is the polar moment of inertia of the vehicle in yaw. The control u is the steer angle of the front wheels. (Terms can be added for four-wheel steer.)

The output variable of interest is the lateral position of the front axle. Thus, the C matrix is defined as:

$$C = [1 \quad a \quad 0 \quad 0] \quad (23)$$

The task performed by the driver model is to calculate a new value of steering angle at the front wheels as the simulation proceeds, using Equation 14 and the target location from Equation 21. Consider the terms needed to make the calculation. The solution involves a summation over m intervals (m is presently programmed as 10). For a given speed, the relation between interval i and time is fixed ($t=iT/m$). Each time the driver model is called, it is provided x_0 and the information needed to determine y_{targ} for each interval. The coefficients F_i , g_i , and g_i^2 depend only on speed, and do not need to be re-computed unless the vehicle speed has changed.

The free-response coefficients in the array F_i define the lateral position of the vehicle at the end of interval i due to non-zero initial conditions. Although the internal 2-DOF vehicle model has four state variables, the choice of axis systems simplifies the calculations. Figure I.3 shows that the initial values of x_1 (lateral coordinate Y) and x_2 (yaw angle ψ) are identically zero in the axis system of the driver model. Therefore, only two coefficients are needed in array F_i : f_{1i} accounts for initial lateral velocity (x_3), and f_{2i} accounts for initial yaw rate (x_4).

The coefficient f_{1i} represents the value of y at the end of interval i , for an initial value of $x_3 = 1$. The values are calculated for $i=1, \dots, m$ using numerical integration. The initial values of all state variables except x_3 are set to zero and x_3 is set to unity. The 2-DOF model is then simulated using an Euler integrator from $t=0$ to $t=T$, and values of the lateral position are saved at the m locations used in the summation. The process is repeated to determine the values of f_{2i} , except that the initial conditions for the 2-DOF model are that $x_4=1$ and all other variables are 0.

The free-response coefficients in F_i are also used to compute the control response coefficients g_i , using Equation 7. (The coefficient g_i defines the deviation of the vehicle at the end of interval i due to a sustained steer angle of unity.) An extra state variable is added whose derivative is y . Its integral at the end of each interval i is multiplied by B_3

for the initial condition of $x_3 = 1$, and its integral for the initial $x_4=1$ is multiplied by B_4 . The weighted sum is g_i .

Equation 14 is used to determine the optimal control steer. The steer due to factors other than the driver is provided as u_o , and subtracted from the optimal control to obtain the steer needed by the driver, u_c . The driver steer angle, applied to the front wheels, is divided by a steering gear ratio to obtain the corresponding angle at the steering wheel.

Transport Delay

The steer angle from the driver, u_c , is given a pure transport delay to simulate the neuromuscular delay of a human driver. Previous research has shown the transport delay to be an important parameter in determining the dynamics of the closed-loop man-machine system [1].

References

1. MacAdam, C.C. "Application of an Optimal Preview Control for Simulation of Closed-Loop Automobile Driving," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 11, June 1981.
2. MacAdam, C.C. "An Optimal Preview Control for Linear Systems," Journal of Dynamic Systems, Measurement, and Control, ASME, Vol. 102, No. 3, Sept. 1980.

Appendix J — Model Output Variables

The main output of the vehicle simulation programs in VDM RoAD is a pair of files (extensions = ERD and BIN) with time histories of variables computed during the simulation. By convention, the two files together are called an ERD file. See Appendix A for details of the ERD format.

The simulation is performed by calculating new values of vehicle variables as functions of simulated time. The ERD file contains a large table of numbers, where each column corresponds to a different variable, and each row corresponds to a value of time. The values from this file are read by the plotter and animator programs in VDM RoAD.

Types of Output Variables

The ERD file contains variables that fall into three groups:

1. Vehicle dynamic variables needed to create animations for viewing vehicle motions.
2. Additional vehicle variables that characterize its behavior during the run.
3. Variables that define the instant positions of the wheels on the road, along with the properties of the road surface at those positions.

Although the simulation is based on elapsed time, and the ERD file is organized by discrete time steps, all variables can be plotted against any other variable. Station number is included, and most of the pre-configured plot setups in VDM RoAD involve plots of variables versus station number.

Animator Variables

In order to locate and orient a rigid body in 3D space, six independent variables are needed: X-Y-Z coordinates and three rotation angles. For example, the car model has five independent moving parts (the body and four wheels). Each requires six variables to animate. Therefore 20 variables in the ERD file are needed to produce full animations. For the truck models, there is some overlap because wheels on the same axle “share” some variables.

The coordinate variables begin with the letters X, Y, and Z. The rotation variables begin with the names Yaw, Pitch, Roll, and Rot. Pitch and Rot both involve the same conceptual rotation (about a Y axis), however, the Rot variables apply to the rolling wheels that have units of revolution, rather than degrees.

Vehicle Variables

There are potentially hundreds of variables that are of interest to engineers studying vehicle dynamics. However, for VDM RoAD, only those variables that would be of interest to road designers are included in the ERD files. Even with this qualification, there are a lot of variables.

All vehicle controls that involve the driver are available for plotting: the steering wheel angle (*Steer_SW*), the target speed (*VxTarget*), and the brake input (*Prbk_con*).

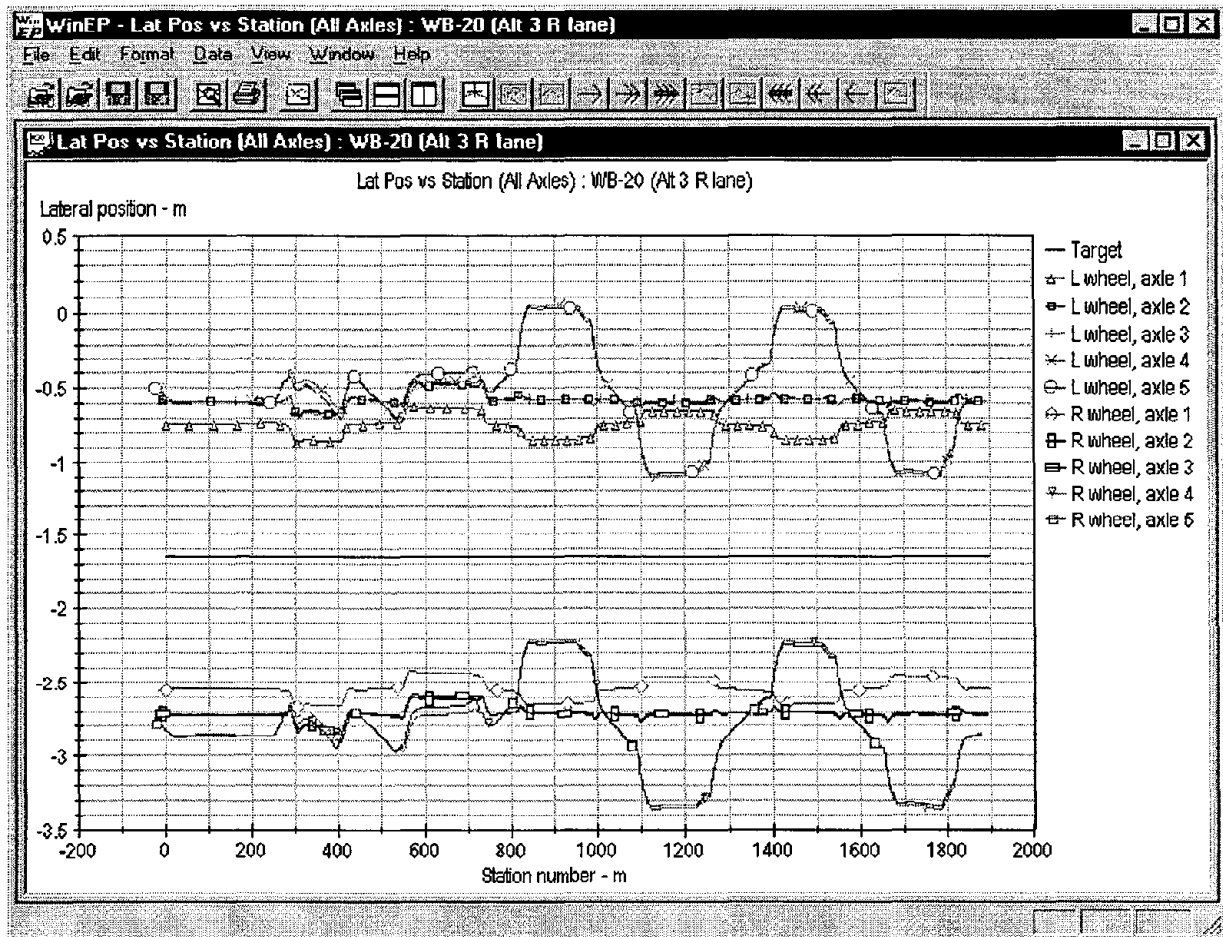
The forces and moments acting between the ground and the vehicle are characterized with variables beginning with *Fx* (longitudinal force), *Fy* (lateral force), *Fz* (vertical force), and *Mz* (aligning moment). For dual wheel assemblies, the forces for the individual tires are summed. For the car model, aerodynamic forces and moments are also present, and these also begin with the symbol *F* (forces) and *M* (moments).

Angular velocity variables begin with the letters *AV*. Acceleration variables begin with *Ax* (longitudinal) or *Ay* (lateral). Speed variables begin with *Vx* (longitudinal) or *Vy* (lateral). Wheel spin is computed with units of forward speed (km/h) by multiplying the angular rotation (rad/sec) with the effective rolling radius, and converting to km/h.

Tracking and Road Geometry Variables

The vehicle dynamics calculations are based on a global coordinate system with *X*, *Y*, and *Z* axes. The road geometry is based on a curved coordinates system *S*, *L*, and *Z*, that follows the road design line (typically at the center). *S* is station number and *L* is the lateral distance to the left of the design line, perpendicular to it. The *S* position of each wheel is computed and stored in the output file, with variables beginning with *Stat*. These variables can be used to create cross-plots of wheel-related variables (e.g., tire forces) as functions of station number.

The lateral coordinates are also available and are associated with variables beginning with *Lat*. In addition, the target lateral position is available (*Lat_pos*). Plots of these variables show the paths taken by the wheels of the simulated vehicle in an efficient graphical form, as shown below.



Lateral tracking as a function of station number.

The figure shows that for a target tracking (vehicle centerline) of -1.65 m, the front wheels deviate laterally by a range of about 0.25 m, as the driver controller model in VDM RoAD steers to attempt to follow the specified path. The rear trailer axle moves over a wider range of about 1 m.

The output file also has the Z coordinate of the road surface associated with each wheel (variables beginning with Zrd) and the longitudinal (dZ/dS) and lateral slopes (dZ/dL).

Creating Lists of Outputs

The following lists of output variables were created from within the software for a car and a five-axle combination vehicle (three-axle tractor with two-axle semitrailer).

To create a similar list for other vehicle models, do the following:

1. Go to the Runs library, and find a run involving the vehicle of interest.
2. If the run has not already been made, click the **Run** button.

3. Click the **Plot** button to bring up WinEP.
4. Select the menu item **Save Channel List (ctrl-S)** from the **File** menu.
5. A **Save File** dialog box will come up.
6. Enter a name for the text file and locate it in a directory where you can find it later.
7. Click the **Save** button to create the file.
8. You will be returned to the WinEP graphics window.
9. Exit WinEP.

Car Model

Listing J.1 lists all of the plot variables that are produced by the car simulation program in VDM RoAD.

Listing J.1. Output variables available for the car simulations.

File: C:\VDM_ROAD\RUNS\541.ERD

File generated by WinEP: 11/21/97 10:17:32

Channel Names

1. AVz : Vehicle yaw rate - deg/s
2. Ax : Vehicle longitudinal accel. - g's
3. Ay : Vehicle lateral accel. - g's
4. Ay_road : Input (road) lateral accel. - g's
5. Az : Vehicle vertical accel. - g's
6. BetaAero: Aerodynamic slip angle - deg
7. dZ/dL_LF: LF road cross slope at wheel - -
8. dZ/dL_LR: LR road cross slope at wheel - -
9. dZ/dL_RF: RF road cross slope at wheel - -
10. dZ/dL_RR: RR road cross slope at wheel - -
11. dZ/dS_LF: LF road long. slope at wheel - -
12. dZ/dS_LR: LR road long. slope at wheel - -
13. dZ/dS_RF: RF road long. slope at wheel - -
14. dZ/dS_RR: RR road long. slope at wheel - -
15. Fxaero : Longitudinal aerodynamic force - N
16. Fx_LF : LF tire longitudinal force - N
17. Fx_LR : LR tire longitudinal force - N
18. Fx_RF : RF tire longitudinal force - N
19. Fx_RR : RR tire longitudinal force - N
20. Fyaero : Lateral aerodynamic force - N
21. Fy_LF : LF tire lateral force - N
22. Fy_LR : LR tire lateral force - N
23. Fy_RF : RF tire lateral force - N
24. Fy_RR : RR tire lateral force - N

- 25. Fzaero : Vertical aerodynamic force - N
- 26. Fz_LF : LF tire vertical load - N
- 27. Fz_LR : LR tire vertical load - N
- 28. Fz_RF : RF tire vertical load - N
- 29. Fz_RR : RR tire vertical load - N
- 30. Lat_Targ: Target lat. pos. from c-line - m
- 31. Lat_WLF : Lateral position of LF wheel - m
- 32. Lat_WLR : Lateral position of LR wheel - m
- 33. Lat_WRF : Lateral position of RF wheel - m
- 34. Lat_WRR : Lateral position of RR wheel - m
- 35. Mxaero : Rolling aerodynamic moment - N-m
- 36. Myaero : Pitching aerodynamic moment - N-m
- 37. Mzaero : Yawing aerodynamic moment - N-m
- 38. Mz_LF : LF tire aligning moment - N-m
- 39. Mz_LR : LR tire aligning moment - N-m
- 40. Mz_RF : RF tire aligning moment - N-m
- 41. Mz_RR : RR tire aligning moment - N-m
- 42. Pbrk_con: Brake control input - kPa
- 43. Pitch : Body pitch - deg
- 44. RFric_LF: Friction used at LF tire - -
- 45. RFric_LR: Friction used at LR tire - -
- 46. RFric_RF: Friction used at RF tire - -
- 47. RFric_RR: Friction used at RR tire - -
- 48. Roll : Body roll - deg
- 49. Roll_E : Body roll (Euler) - deg
- 50. Roll_WLF: LF wheel inclination angle - deg
- 51. Roll_WLR: LR wheel inclination angle - deg
- 52. Roll_WRF: RF wheel inclination angle - deg
- 53. Roll_WRR: RR wheel inclination angle - deg
- 54. Rot_WLF : Rotation of LF wheel - rev
- 55. Rot_WLR : Rotation of LR wheel - rev
- 56. Rot_WRF : Rotation of RF wheel - rev
- 57. Rot_WRR : Rotation of RR wheel - rev
- 58. Station : Station number at front axle - m
- 59. Stat_WLF: Station number at LF wheel - m
- 60. Stat_WLR: Station number at LR wheel - m
- 61. Stat_WRF: Station number at RF wheel - m
- 62. Stat_WRR: Station number at RR wheel - m
- 63. Steer_SW: Steering-wheel angle - deg
- 64. Vx : Vehicle longitudinal velocity - kph
- 65. VxTarget: Target longitudinal velocity - kph
- 66. Vy : Vehicle lateral velocity - kph
- 67. Vz : Vehicle vertical velocity - kph
- 68. XcamptF : X coord. of front camera point - m
- 69. XcamptR : X coord. of rear camera point - m
- 70. Xo : X coordinate of car ref. point - m

71. X_Design: X coordinate of centerline - m
 72. X_WLF : X coordinate of LF wheel center - m
 73. X_WLR : X coordinate of LR wheel center - m
 74. X_WRF : X coordinate of RF wheel center - m
 75. X_WRR : X coordinate of RR wheel center - m
 76. Yaw : Vehicle yaw - deg
 77. Yaw_WLF : Yaw angle of LF wheel - deg
 78. Yaw_WLR : Yaw angle of LR wheel - deg
 79. Yaw_WRF : Yaw angle of RF wheel - deg
 80. Yaw_WRR : Yaw angle of RR wheel - deg
 81. YcamptF : Y coord. of front camera point - m
 82. YcamptR : Y coord. of rear camera point - m
 83. Yo : Y coordinate of car ref. point - m
 84. Y_Design: Y coordinate of centerline - m
 85. Y_WLF : Y coordinate of LF wheel center - m
 86. Y_WLR : Y coordinate of LR wheel center - m
 87. Y_WRF : Y coordinate of RF wheel center - m
 88. Y_WRR : Y coordinate of RR wheel center - m
 89. ZcamptF : Z coord. of front camera point - m
 90. ZcamptR : Z coord. of rear camera point - m
 91. Zo : Z coordinate of car ref. point - m
 92. Zrd_LF : LF road elevation at wheel - m
 93. Zrd_LR : LR road elevation at wheel - m
 94. Zrd_RF : RF road elevation at wheel - m
 95. Zrd_RR : RR road elevation at wheel - m
 96. Z_WLF : Z coordinate of LF wheel center - m
 97. Z_WLR : Z coordinate of LR wheel center - m
 98. Z_WRF : Z coordinate of RF wheel center - m
 99. Z_WRR : Z coordinate of RR wheel center - m

Truck Models

Listing J.2 lists all of the plot variables that are produced by the truck and combination simulation programs in VDM RoAD. This listing is for a three-axle tractor towing a two-axle trailer. Variables that do not exist in other models (for example, those related to axle 5) are of course not contained in the output ERD files.

Listing J.2. Output variables available for the car simulations.

File: C:\VDM_ROAD\RUNS\555.ERD

File generated by WinEP: 11/21/97 10:18:13

Channel Names

1. AVz : Tractor yaw rate - deg/s
 2. AVz_2 : Semitrailer yaw rate - deg/s
 3. Ax : Tractor longitudinal accel. - g's

4. Ax_2 : Semitrailer longitudinal accel. - g's
5. Ay : Tractor lateral accel. - g's
6. Ay_2 : Semitrailer lateral accel. - g's
7. Ay_road : Input (road) lateral accel. - g's
8. dz/dL_L1: Road cross slope at L1 wheel - m
9. dz/dL_L2: Road cross slope at L2 wheel - m
10. dz/dL_L3: Road cross slope at L3 wheel - m
11. dz/dL_L4: Road cross slope at L4 wheel - m
12. dz/dL_L5: Road cross slope at L5 wheel - m
13. dz/dL_R1: Road cross slope at R1 wheel - m
14. dz/dL_R2: Road cross slope at R2 wheel - m
15. dz/dL_R3: Road cross slope at R3 wheel - m
16. dz/dL_R4: Road cross slope at R4 wheel - m
17. dz/dL_R5: Road cross slope at R5 wheel - m
18. dz/dS_L1: Road long. slope at L1 wheel - m
19. dz/dS_L2: Road long. slope at L2 wheel - m
20. dz/dS_L3: Road long. slope at L3 wheel - m
21. dz/dS_L4: Road long. slope at L4 wheel - m
22. dz/dS_L5: Road long. slope at L5 wheel - m
23. dz/dS_R1: Road long. slope at R1 wheel - m
24. dz/dS_R2: Road long. slope at R2 wheel - m
25. dz/dS_R3: Road long. slope at R3 wheel - m
26. dz/dS_R4: Road long. slope at R4 wheel - m
27. dz/dS_R5: Road long. slope at R5 wheel - m
28. Fx_L1 : Long. force, L wheel, axle 1 - N
29. Fx_L2 : Long. force, L wheel, axle 2 - N
30. Fx_L3 : Long. force, L wheel, axle 3 - N
31. Fx_L4 : Long. force, L wheel, axle 4 - N
32. Fx_L5 : Long. force, L wheel, axle 5 - N
33. Fx_R1 : Long. force, R wheel, axle 1 - N
34. Fx_R2 : Long. force, R wheel, axle 2 - N
35. Fx_R3 : Long. force, R wheel, axle 3 - N
36. Fx_R4 : Long. force, R wheel, axle 4 - N
37. Fx_R5 : Long. force, R wheel, axle 5 - N
38. Fy_L1 : Lateral force, L wheel, axle 1 - N
39. Fy_L2 : Lateral force, L wheel, axle 2 - N
40. Fy_L3 : Lateral force, L wheel, axle 3 - N
41. Fy_L4 : Lateral force, L wheel, axle 4 - N
42. Fy_L5 : Lateral force, L wheel, axle 5 - N
43. Fy_R1 : Lateral force, R wheel, axle 1 - N
44. Fy_R2 : Lateral force, R wheel, axle 2 - N
45. Fy_R3 : Lateral force, R wheel, axle 3 - N
46. Fy_R4 : Lateral force, R wheel, axle 4 - N
47. Fy_R5 : Lateral force, R wheel, axle 5 - N
48. Fz_L1 : Vertical load, L wheel, axle 1 - N
49. Fz_L2 : Vertical load, L wheel, axle 2 - N

50.	Fz_L3	: Vertical load, L wheel, axle 3 - N
51.	Fz_L4	: Vertical load, L wheel, axle 4 - N
52.	Fz_L5	: Vertical load, L wheel, axle 5 - N
53.	Fz_R1	: Vertical load, R wheel, axle 1 - N
54.	Fz_R2	: Vertical load, R wheel, axle 2 - N
55.	Fz_R3	: Vertical load, R wheel, axle 3 - N
56.	Fz_R4	: Vertical load, R wheel, axle 4 - N
57.	Fz_R5	: Vertical load, R wheel, axle 5 - N
58.	Lat_L1	: Lateral position of L1 wheel - m
59.	Lat_L2	: Lateral position of L2 wheel - m
60.	Lat_L3	: Lateral position of L3 wheel - m
61.	Lat_L4	: Lateral position of L4 wheel - m
62.	Lat_L5	: Lateral position of L5 wheel - m
63.	Lat_pos	: Target lat. pos. from c-line - m
64.	Lat_R1	: Lateral position of R1 wheel - m
65.	Lat_R2	: Lateral position of R2 wheel - m
66.	Lat_R3	: Lateral position of R3 wheel - m
67.	Lat_R4	: Lateral position of R4 wheel - m
68.	Lat_R5	: Lateral position of R5 wheel - m
69.	Maux1	: Auxiliary roll moment, Axle 1 - N-m
70.	Maux2	: Auxiliary roll moment, Axle 2 - N-m
71.	Maux3	: Auxiliary roll moment, Axle 3 - N-m
72.	Maux4	: Auxiliary roll moment, Axle 4 - N-m
73.	Maux5	: Auxiliary roll moment, Axle 5 - N-m
74.	Mx_H	: Hitch X moment on tractor - N-m
75.	My_H	: Hitch Y moment on tractor - N-m
76.	Mz_H	: Hitch Z moment on tractor - N-m
77.	Mz_L1	: Aligning moment, L wheel, axle 1 - N-m
78.	Mz_L2	: Aligning moment, L wheel, axle 2 - N-m
79.	Mz_L3	: Aligning moment, L wheel, axle 3 - N-m
80.	Mz_L4	: Aligning moment, L wheel, axle 4 - N-m
81.	Mz_L5	: Aligning moment, L wheel, axle 5 - N-m
82.	Mz_R1	: Aligning moment, R wheel, axle 1 - N-m
83.	Mz_R2	: Aligning moment, R wheel, axle 2 - N-m
84.	Mz_R3	: Aligning moment, R wheel, axle 3 - N-m
85.	Mz_R4	: Aligning moment, R wheel, axle 4 - N-m
86.	Mz_R5	: Aligning moment, R wheel, axle 5 - N-m
87.	Pbrk_con	: Brake control input - kPa
88.	Pitch	: Tractor pitch - deg
89.	Pitch_2	: Semitrailer pitch - deg
90.	Pitch_A1	: Pitch of front axle - deg
91.	Pitch_H1	: Hitch pitch - deg
92.	Ro1Ab_A1	: Absolute roll of Axle 1 - deg
93.	Ro1Ab_A2	: Absolute roll of Axle 2 - deg
94.	Ro1Ab_A3	: Absolute roll of Axle 3 - deg
95.	Ro1Ab_A4	: Absolute roll of Axle 4 - deg

- 96. RolAb_A5: Absolute roll of Axle 5 - deg
- 97. Roll : Tractor roll - deg
- 98. Roll_2 : Semitrailer roll - deg
- 99. Roll_E : Tractor roll - deg
- 100. Roll_E_2: Semitrailer roll - deg
- 101. Roll_H1 : Hitch roll - deg
- 102. Rot_L1W : Rotation of L wheel, axle 1 - rev
- 103. Rot_L2W : Rotation of L wheel, axle 2 - rev
- 104. Rot_L3W : Rotation of L wheel, axle 3 - rev
- 105. Rot_L4W : Rotation of L wheel, axle 4 - rev
- 106. Rot_L5W : Rotation of L wheel, axle 5 - rev
- 107. Rot_R1W : Rotation of R wheel, axle 1 - rev
- 108. Rot_R2W : Rotation of R wheel, axle 2 - rev
- 109. Rot_R3W : Rotation of R wheel, axle 3 - rev
- 110. Rot_R4W : Rotation of R wheel, axle 4 - rev
- 111. Rot_R5W : Rotation of R wheel, axle 5 - rev
- 112. Station : Station number at front axle - m
- 113. Stat_L1 : Station number at L1 wheel - m
- 114. Stat_L2 : Station number at L2 wheel - m
- 115. Stat_L3 : Station number at L3 wheel - m
- 116. Stat_L4 : Station number at L4 wheel - m
- 117. Stat_L5 : Station number at L5 wheel - m
- 118. Stat_R1 : Station number at R1 wheel - m
- 119. Stat_R2 : Station number at R2 wheel - m
- 120. Stat_R3 : Station number at R3 wheel - m
- 121. Stat_R4 : Station number at R4 wheel - m
- 122. Stat_R5 : Station number at R5 wheel - m
- 123. Steer_SW: Steering wheel angle - deg
- 124. StrT_L : L total steer - deg
- 125. StrT_R : R total steer - deg
- 126. Vx : Tractor long. velocity - kph
- 127. VxTarget: Target longitudinal velocity - kph
- 128. Vx_2 : Semitrailer long. velocity - kph
- 129. Vx_L1 : Rolling speed of L wheel, axle 1 - kph
- 130. Vx_L2 : Rolling speed of L wheel, axle 2 - kph
- 131. Vx_L3 : Rolling speed of L wheel, axle 3 - kph
- 132. Vx_L4 : Rolling speed of L wheel, axle 4 - kph
- 133. Vx_L5 : Rolling speed of L wheel, axle 5 - kph
- 134. Vx_R1 : Rolling speed of R wheel, axle 1 - kph
- 135. Vx_R2 : Rolling speed of R wheel, axle 2 - kph
- 136. Vx_R3 : Rolling speed of R wheel, axle 3 - kph
- 137. Vx_R4 : Rolling speed of R wheel, axle 4 - kph
- 138. Vx_R5 : Rolling speed of R wheel, axle 5 - kph
- 139. Vy : Tractor lateral velocity - kph
- 140. Vy_2 : Semitrailer lateral velocity - kph
- 141. XcamptF : X coord. of front camera point - m

142. XcamptR : X coord. of rear camera point - m
143. Xo : X position of Tractor - m
144. Xo_2 : X position of Semitrailer - m
145. X_A1 : X position of Axle 1 - m
146. X_A2 : X position of Axle 2 - m
147. X_A3 : X position of Axle 3 - m
148. X_A4 : X position of Axle 4 - m
149. X_A5 : X position of Axle 5 - m
150. X_Design: X coordinate of centerline - m
151. X_WLF : X Position of L wheel, axle 1 - m
152. X_WRF : X Position of R wheel, axle 1 - m
153. Yaw : Tractor yaw - deg
154. Yaw_2 : Semitrailer yaw - deg
155. Yaw_H1 : Hitch yaw - deg
156. Yaw_WLF : Yaw angle of L wheel, axle 1 - deg
157. Yaw_WRF : Yaw angle of R wheel, axle 1 - deg
158. YcamptF : Y coord. of front camera point - m
159. YcamptR : Y coord. of rear camera point - m
160. Yo : Y position of Tractor - m
161. Yo_2 : Y position of Semitrailer - m
162. Y_A1 : Y position of Axle 1 - m
163. Y_A2 : Y position of Axle 2 - m
164. Y_A3 : Y position of Axle 3 - m
165. Y_A4 : Y position of Axle 4 - m
166. Y_A5 : Y position of Axle 5 - m
167. Y_Design: Y coordinate of centerline - m
168. Y_WLF : Y Position of L wheel, axle 1 - m
169. Y_WRF : Y Position of R wheel, axle 1 - m
170. ZcamptF : Z coord. of front camera point - m
171. ZcamptR : Z coord. of rear camera point - m
172. Zo : Z position of Tractor - m
173. Zo_2 : Z position of Semitrailer - m
174. Zrd_L1 : Road elevation at L1 wheel - m
175. Zrd_L2 : Road elevation at L2 wheel - m
176. Zrd_L3 : Road elevation at L3 wheel - m
177. Zrd_L4 : Road elevation at L4 wheel - m
178. Zrd_L5 : Road elevation at L5 wheel - m
179. Zrd_R1 : Road elevation at R1 wheel - m
180. Zrd_R2 : Road elevation at R2 wheel - m
181. Zrd_R3 : Road elevation at R3 wheel - m
182. Zrd_R4 : Road elevation at R4 wheel - m
183. Zrd_R5 : Road elevation at R5 wheel - m
184. Z_A1 : Z position of Axle 1 - m
185. Z_A2 : Z position of Axle 2 - m
186. Z_A3 : Z position of Axle 3 - m
187. Z_A4 : Z position of Axle 4 - m

Appendix J

Model Output Variables

- 188. Z_A5 : Z position of Axle 5 - m
- 189. Z_WLF : Z Position of L wheel, axle 1 - m
- 190. Z_WRF : Z Position of R wheel, axle 1 - m