ENGINEERING

GETTING STARTED WITH LOTUS SUSPENSION ANALYSIS

VERSION 5.04

The information in this document is furnished for informational use only, may be revised from time to time, and should not be construed as a commitment by Lotus Cars Ltd or any associated or subsidiary company. Lotus Cars Ltd assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

This document contains proprietary and copyrighted information. Lotus Cars Ltd permits licensees of Lotus Cars Ltd software products to print out or copy this document or portions thereof solely for internal use in connection with the licensed software. No part of this document may be copied for any other purpose or distributed or translated into any other language without the prior written permission of Lotus Cars Ltd.

©2015 by Lotus Cars Ltd. All rights reserved.

CONTENTS

1 - INTRODUCING LOTUS SUSPENSION ANALYSIS

2 - GETTING STARTED

3 - INTERACTIVE TEMPLATE MODIFICATION

4 - EXTENDED TRAVEL OPTIONS

5 - ADDITIONAL FEATURES

6 - COMMAND REFERENCE CARD

7 - COMPLIANT ANALYSIS

8 - GRAPHICAL VIEW DISPLAY TOOL

9 - USER FORMATTING RESULTS

10 - LOCAL COORDINATE SYSTEMS

11 - BALL JOINT RESULTS DISPLAY

12 - INTERNAL OPTIMISER

13 – INTERACTIVE TEMPLATE BUILDER MODULE

14 - USER TEMPLATES (1)

15 - USER TEMPLATES (2)

16 - EXTERNAL APPLICATION AUTO-SEARCH AND LOAD

17 - COMMAND MODE OPERATION

18 - BATCH FILES

19 - REPORT FILES

20 - USER LANGUAGE

21 - FULL SUSPENSION MODEL

22 - IMPORTING/EXPORTING HARD POINTS

About This Guide

Welcome to Lotus Engineering Suspension Analysis. This product will allow you to design and analyse the vehicle suspension hard points to achieve the required suspension characteristics. The optional addition of compliant bushes and operating forces allow compliant characteristics to be calculated and bushes tuned to obtain the desired behaviour.

What You Need to Know

This guide assumes the following:

- Lotus Suspension Analysis is installed on your computer or network and you have permission to execute the relevant Lotus modules.

- The necessary password file is installed to allow you to run the necessary modules.

You have a basic understanding of vehicle suspension mechanisms, their loading regimes and functional operating requirements.

1 Introducing Lotus Suspension Analysis

1.1 Overview

This chapter introduces you to the Lotus Suspension Analysis Tool and explains the normal uses for it. It also introduces the tutorials that we've included in this guide to get you started working with Lotus Suspension Analysis (LSA).

This chapter contains the following sections:

1.2 What is Lotus Suspension Analysis?

LSA is a design and analysis tool that can be used for both the initial layout of a vehicle suspensions hard points, and also the design and orientation of suspension bushes for the tuning of the compliant behaviour.

Models are created and modified through a 3d-viewing environment. This allows hard points and bushes to be ‘dragged’ on screen and graphical/numerical results updated in ‘real time’. A template-based approach to the modelling allows users to create their own suspension models, supplementing the ‘standard’ suspension templates provided.

1.3 Normal Uses of Lotus Suspension Analysis

LSA is used by both designers and analysts alike for the layout of the suspension hard point positions, in order that the required kinematic behaviour is achieved. Any number of results can be displayed graphically, (e.g. Camber angle, Toe angle), against bump motion, roll motion or steering motion. These results are updated in ‘real time’ as the suspension hard points are moved. The inclusion of compliant bushes to the kinematic model allows the tuning of bush properties to be carried out, to achieve required compliant response for items such as lateral force steer.

1.4 Overall Concepts

LSA has two main display and analysis modes, 2D and 3D, and it is possible to import a 2D model into 3D.

Suspensions can be articulated in individual bump/rebound, roll and steering modes or a combination mode that allows all three articulation types to be mixed. The steering modes are relevant to the 3D mode only.

LSA uses templates to identify specific 3D suspension types. These templates define the number of parts, the number of points and connectivity of the parts. A large number of ‘standard’ templates are include with the installation, whilst users can create their own or modify existing ones to model kinematic suspension types not catered for.

3D models can be built as corner, axle or full vehicle suspension models.

LSA can be used just in Kinematic mode, (i.e. rigid bodies with ball joints), or in compliant mode where the deflection due to bushes is added to the kinematic results on an incremental basis, (note that the compliant module is licensed additionally to the kinematic module). The compliant mode includes modal analysis and forced damped capability.

1.5 Coordinate system

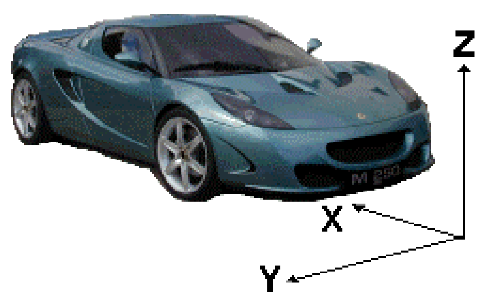
The LSA co-ordinate system is a right-handed system the origin of which must be in front of the car and coincide with the vehicle longitudinal centre line.

Y-axis is across the car track, and the +ve direction being towards the right side when sitting in the car. Suspensions can be defined as right side or left side as required.

X-axis is along the vehicle wheel base and positive toward the rear of the car.

Z-axis is the vertical height and positive upwards.

When inputting suspension hard point data you must ensure that all co-ordinates are consistent with the origin you have selected and be aware that all suspension hard point output generated by LSA will be relative to that origin. The only restrictions are that the X-Z plane must pass through the centre of the car and the origin must be in front of the car. The co-ordinate system origin need not be coincident with the ground plane.



LSA coordinate System

1.6 Default Sign convention

Camber - Inclination of the wheel plane to the vertical, negative when the wheel leans in at the top

King Pin Angle - The front view angle between the steering axis and the vertical. Positive when the steering axis leans inwards at the top.

Toe - Angle between the plane of the wheel and the forward direction, positive if the front of the wheel is “toed in” toward the centre of the car.

Castor - The angle in side view between the steering axis and vertical. Positive when the top of the steering axis is inclined toward the rear.

Steering Lock - Linear Y-axis displacement of the steering rack. Positive steering lock can produce negative or positive toe depending if the steering rack is in front or behind the steering axis.

Roll - Right hand rule applied to the vehicle positive x-axis. When sitting in the car roll to the left is positive.

The default sign conventions can be modified by a user, to suit local requirements. These local sign conventions can include a sign change, a scale and shift terms. The user sign conventions are saved to the local INI file.

1.7 About the Tutorials

The remainder of this guide is structured around a series of tutorials that introduce you to the features of Lotus Suspension Analysis. Each tutorial builds on what was learnt in those before it and are thus linked such that the user should work through them in the order presented. The essential steps required to complete the tutorial have been bulleted as shown below.

- ***Essential steps in the tutorial are bulleted and italic.***

To save time you can skip through the text and only do the essential steps. The rest of the text gives a more complete description.

2 Getting Started

2.1 Overview

This chapter introduces the main features of the product and gives a summary of its base functionality.

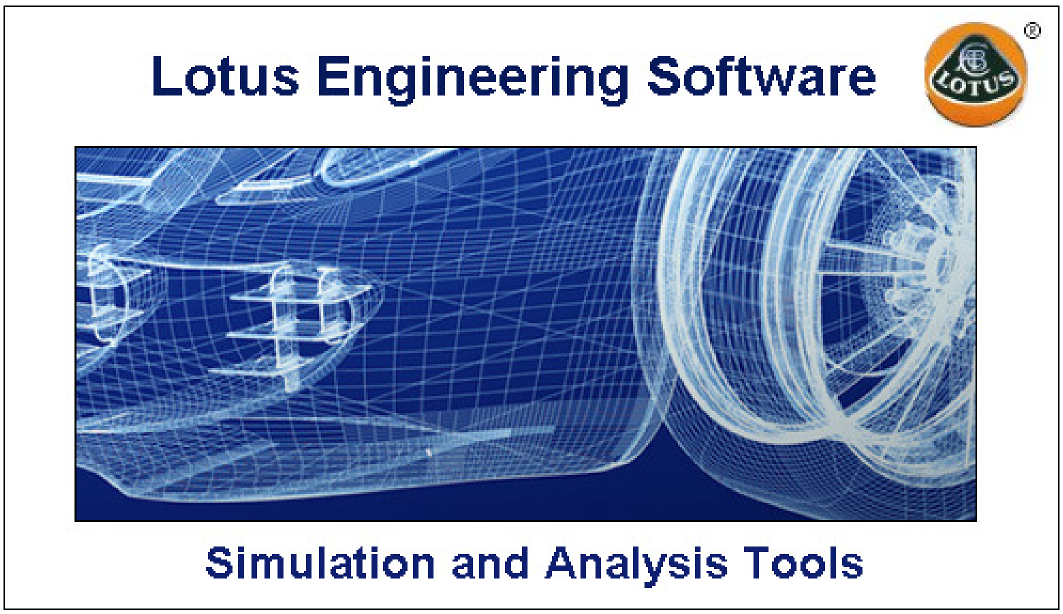
This chapter contains the following sections:

2.2 Starting the Application

To start Lotus Suspension Analysis from the main **Start** menu, point to **Programs** and then **Lotus Engineering Software**, then **Lotus Suspension Analysis (Interactive)**. If the program fails to start or the menu item is missing from your start menu, firstly confirm that the software has been installed correctly. You can browse for the application directly, the executable file name is *Shark.exe*. As the program starts, the start up 'splash' screen will be displayed, before the main application window is opened.

Start LSA from windows start menu

- ***Start / Programs / Lotus Engineering Software / Lotus Suspension Analysis (Interactive)***

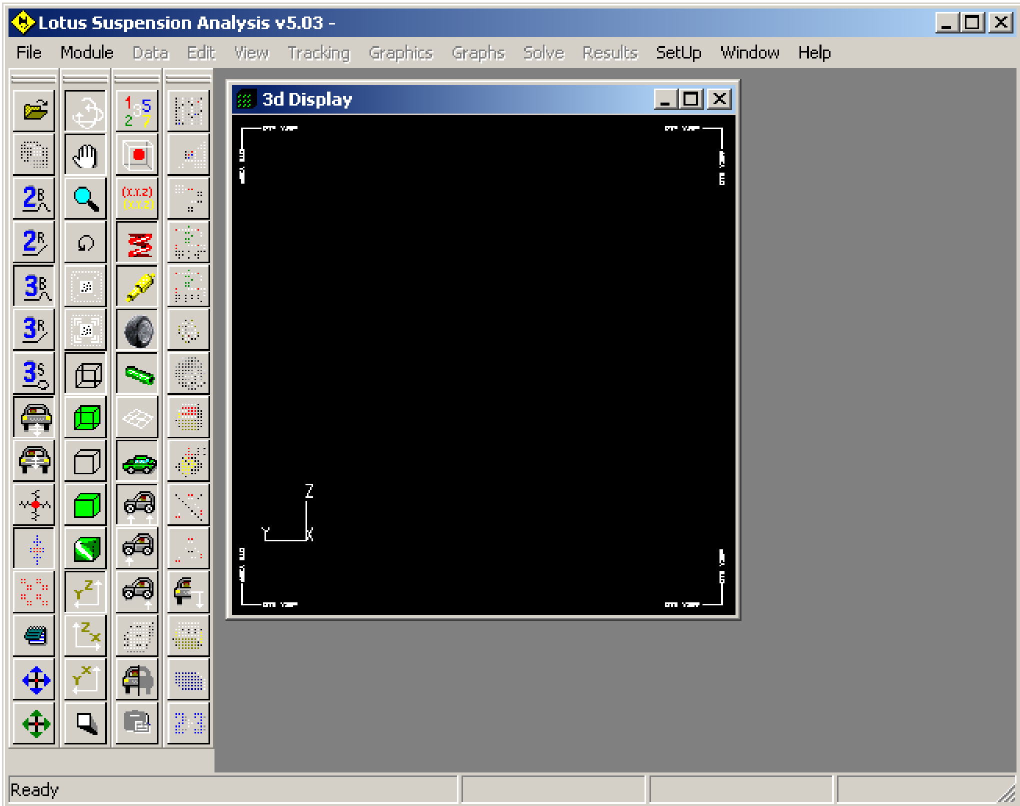


Start-up Splash Screen

On start-up, the application will open with an empty 3D display window. A number of the menus and icons are disabled until either a new model has been started or an existing model has been loaded.

The settings of both the display and analysis modes is initially set either by the defaults, (if not previously run), or by the settings saved to the 'ini' file from the previous run.

Note: The start-up procedure may differ from the text above as a function of your local installation. If in doubt check with your local IT support personnel.



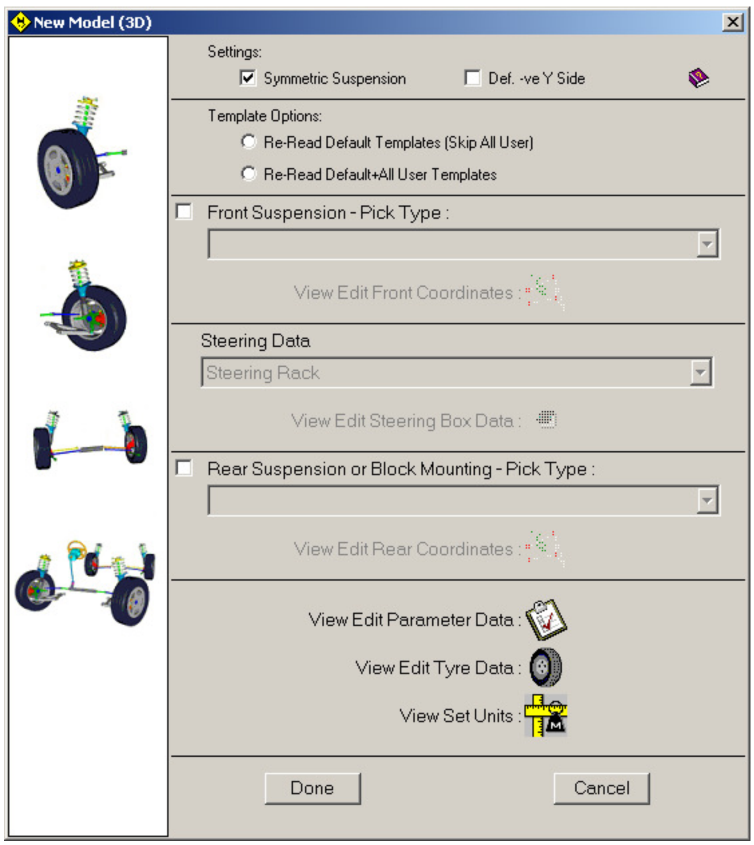
Layout of application

Additional context specific menu items are used throughout the application and can be accessed by clicking the right mouse on the window/graph of interest.

2.3 Creating a New Model

To create a new model, select the *File / New* menu option from the main menu bar, (note that we are in 3D module and will thus be creating a new 3D model. Creating a new model in the 2D module works in exactly the same way). The '*new model*' dialogue box is then displayed.

- ***File / New***



New Model Dialogue Box

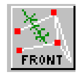
The dialogue box allows you to pick the required suspension type for the front, rear or both. For our example we will consider a model with only the front. Suspensions are modelled in LSA based on specific template types. Depending whether the template has been built with provision for a steering attachment point, it will dictate if it appears in the list of available front suspension types (all defined templates are listed in the rear suspension list).

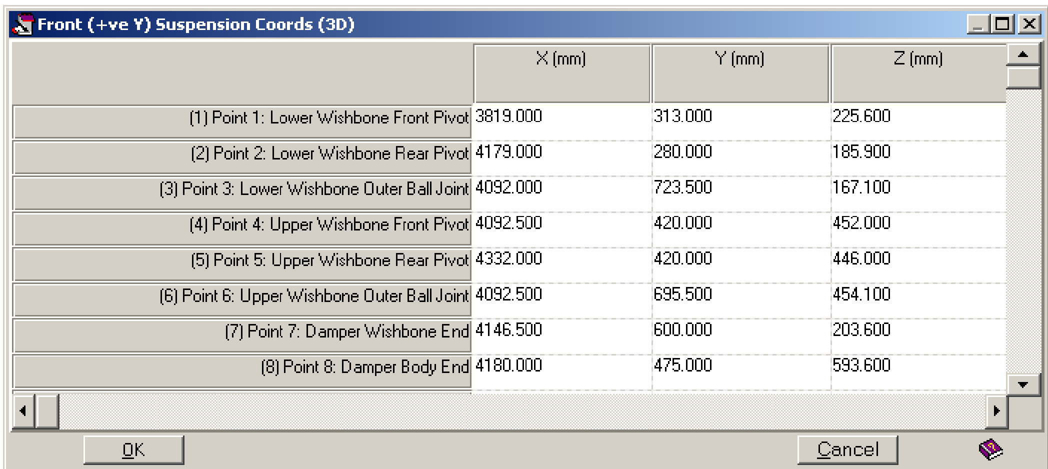
- ***Check 'Front Suspension- PickType'***

- ***From the front suspension drop down box select 'Type 1: Double Wishbone, damper to lower wishbone'***

- ***From steering type drop down box select 'steering rack'***

Once you have selected the front suspension type, the 'View/Edit Front Co-ordinates' icon becomes enabled, allowing you to change the default hard point co-ordinate values.

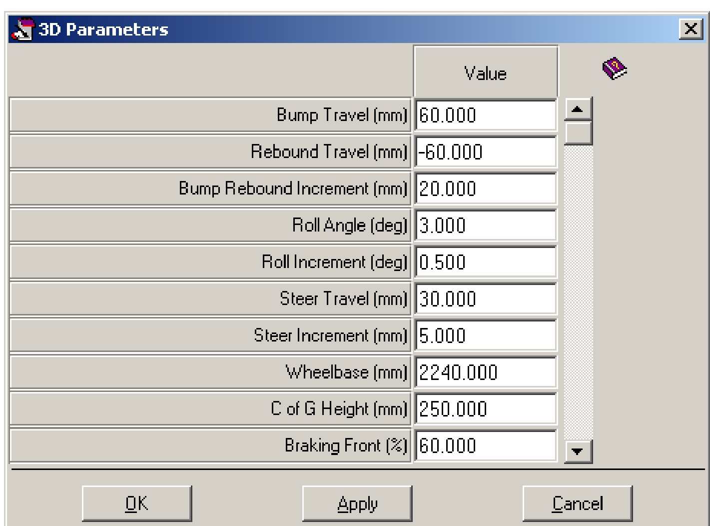
➢ ***Click 'View/Edit Front Co-ordinates'***  ***to inspect the front suspension co-ordinate. Once done click OK to accept defaults***



Type 1 Default hard points display

For a front suspension, you can choose between a conventional steering rack or two types of steering box (a steering box requires additional hard point data to be defined). We will stick with the more normal steering rack.

From the 'new' dialogue box we can also view/change the 'Parameter' data associated with the model (such as wheelbase, c of g height, bump travel, brake split etc, and geometric data associated with the tyre). All of the model properties can also be modified at a later stage as required.

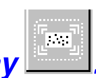


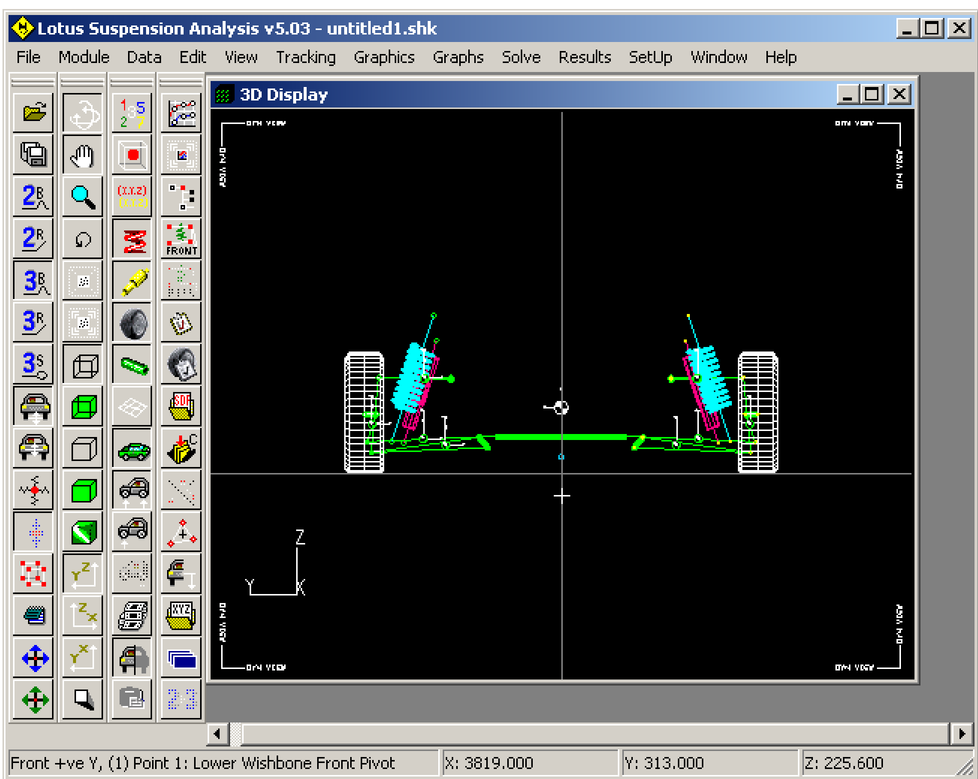
Parameter Data Listing

To complete the creation of a new front suspension model, click ‘Done’. This will now enable all the previously ‘greyed-out’ menus and icons. The created model is now displayed in the ‘3D display’ window.

- ***Click ‘Done’ to open the model.***

Now that we have a model we will set up the 3D display. The first time LSA is opened, the default view settings will be applied. Subsequently, each time that LSA is closed, the current view settings are saved and will be used the next time LSA is opened (the default settings can only be restored by deleting the LSA initiation file ’SHARK.INI’ from the installation directory). Note this location may change with specific user installations and use more than one INI file.

- ***‘View / Screen Display / Static Only’***
- ***‘Graphics / Point Limits’, ensure neither ‘visible’ nor ‘use’ have a tick mark next to them.***
- ***Display both sides of the suspension by left clicking on the ‘Display Both Sides’ tool***  ***.***
- ***Auto scale the view with ‘Autoscale Display***  ***.***

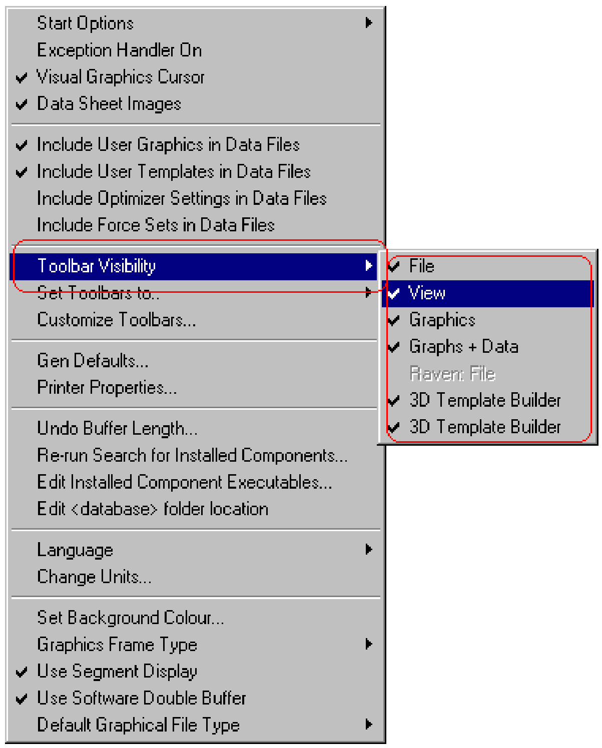


Screen shot of new front suspension model

2.4 Manipulating the Graphical View

Use the Setup menu from the menu bar to display only the ‘View’ toolbar.

➢ ***From the ‘SetUp’ menu, setup the toolbars so that only ‘View’ Toolbar is displayed.***



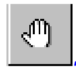
Selecting View Toolbar Visibility’s from SetUp menu

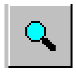
The suspension 3D display interface has two modes: ‘Dynamic viewing’ for manipulating the view, and ‘Edit’ mode for modifying the suspension geometry. The left mouse button is clicked on the dynamic view icon to toggle between ‘Viewing’ and ‘editing’ mode. In viewing mode markers are displayed in each corner of the 3D suspension display window.

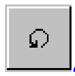
➢ ***Toggle the ‘dynamic view icon’***  ***so that the viewing mode is selected, i.e. markers displayed in each corner of the suspension 3D display window.***

The graphical display is manipulated through the mouse cursor and buttons. It allows you to rotate, translate and zoom in/out by the combination of holding the left mouse button down whilst moving the mouse. Specific menu options exist for ‘autoscale’, pick centre, and setting the view to orthogonal projections.

If you are in the ‘edit’ mode, selecting any one of the dynamic viewing options will change the mode to dynamic viewing. Alternatively, selecting the dynamic view icon will cycle between edit and view modes.

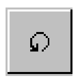
- ***Change to 'Translate view'***  ***. Select a point on the 3D-suspension window with the left mouse button, hold down and drag.***

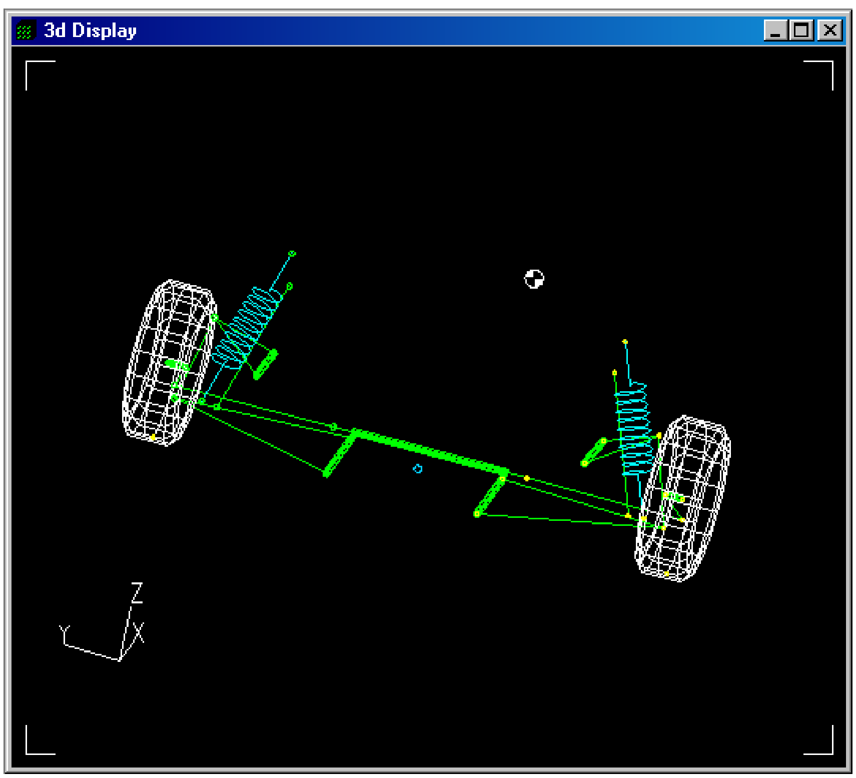
- ***Change to 'Scale view'.***  ***Select with left mouse button, hold down and move down to zoom in, up to zoom out.***

- ***Change to 'Rotate view'***  ***. Select with left mouse button, hold down and move to rotate view. Picking towards the centre rotates the eye point around the object, picking towards the edge rotates around the object axis.***

When in dynamic view mode, the right mouse button will cycle through the three dynamic view types: zoom, translate and rotate.

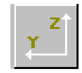
In some situations, it is desirable to make frequent use of a particular user defined view. To achieve this, LSA can save user define views for latter use.

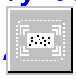
- ***Use the rotate view tool***  ***to set a non-orthogonal view***



'Pictorial' view of front suspension

- ***Select the menu item 'View / Saved Views / Save' and enter a name for the view, click ok***

- ***Restore the front view by selecting the front view icon*** 

, then the 'Autoscale display' icon '***'***

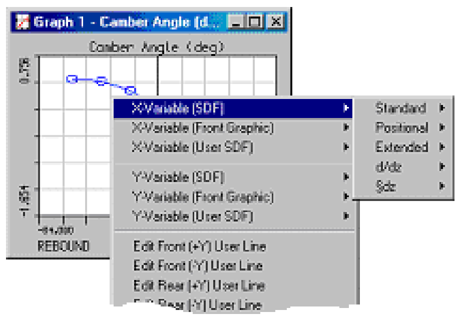
- ***The saved view can be used via the 'View' main menu. 'View / Saved Views -/ Recall Saved' and select the saved view from the list***

- ***To proceed, re-set the suspension display to front view and ensure the view is fitted to the display window.***

2.5 Displaying Graphical Results

Graphs are used to display analysis results for any of the calculated results. To open a graph, select *Graphs / New-Open*. The created window will show the current model results for a particular parameter, e.g. camber angle. To change the displayed parameter for a particular graph select the graph with the right mouse button and pick the required parameter from the displayed list. The right mouse menu also contains options for setting axis scales and general viewing options such as zoom and autoscale.

Any number of graphs can be open at the same time, the positions and sizes of which can be modified and saved by the user for future use. Some exporting options are also available, as *export to Excel…*



Graph showing right mouse menu

The default setting for each of the x-y graphs is to display the original 'as calculated' x and y data values. For each individual graph the user can choose to alternatively plot the data as the '*derivative'* or the '*integral'* of the original calculated data. Because of the nature of integration, an assumption needs to be made about the intercept value. The integral display assumes a zero value for the first plotted value. Changing the display type from data, derivative and integral also impacts the displayed value of the 'ride derivative value'. When the graph plot is set to 'data', the ride derivative value lists the derivative value at the static position. When the graph plot is set to 'derivative' or 'integral', the ride derivative value lists the y value at the static position.

If you have a number of x-y graphs displayed and wish to produce a hard copy of them all, you can print them in one simple menu selection. The *Graphs / Print All* menu has a number of sub options that perform multiple prints with a specified number of prints per page. These are 1,3,4,6 and 8 to a page.

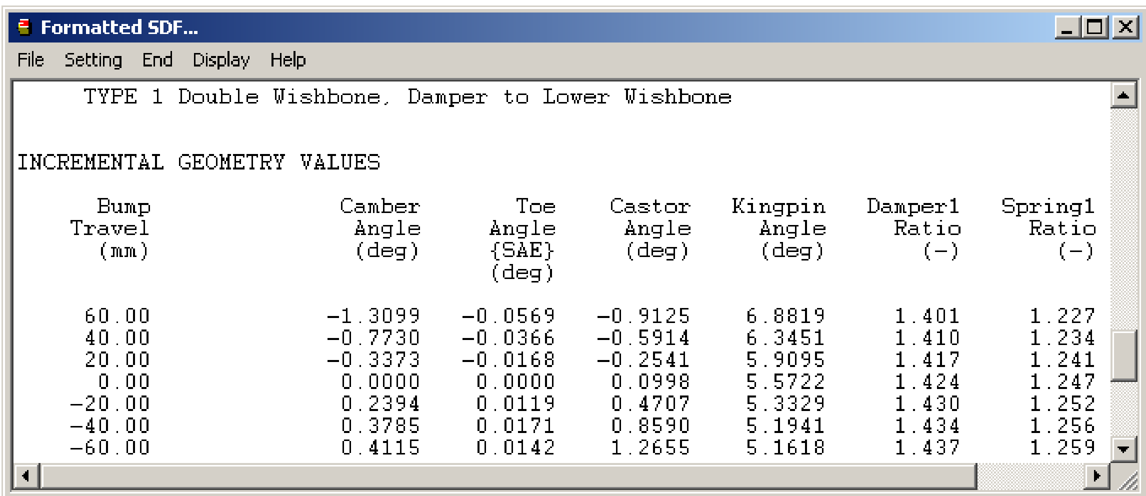
- ***Open a graph, and set it to display camber angle by 'Right Click / Y-Variable (SDF) / Standard / Camber Angle'.***

- ***Open two more graphs for Toe Angle and Castor Angle and arrange the windows, so you can view the suspension 3D display and each graph simultaneously.***

2.6 Displaying Text Results

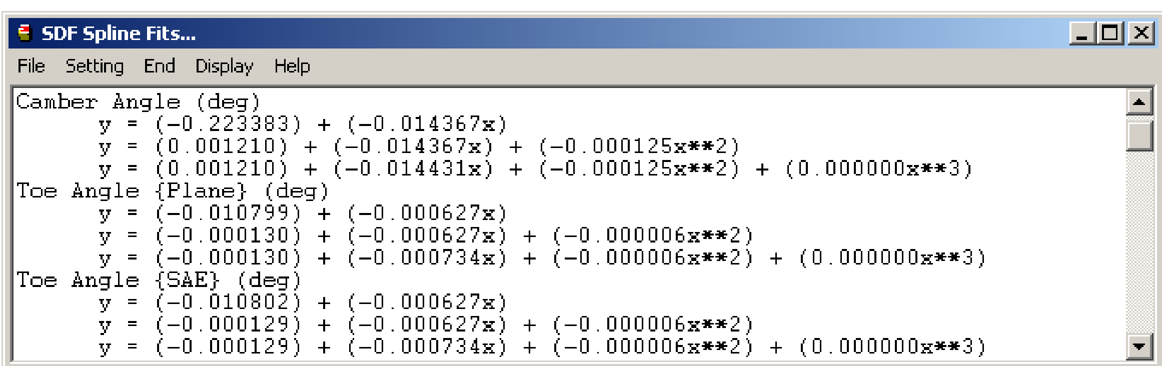
The text results for the currently defined suspension model can be displayed in a scrollable text window, *Results / Formatted SDF…* This lists an echo of the input data and tabulated/headed suspension derivatives. This provides a convenient reporting medium for numerically summarising the suspension properties.

- ***From the main menu select 'Results / Formatted SDF…'. When done inspecting results, close the text results window***



Sample Formatted SDF Display

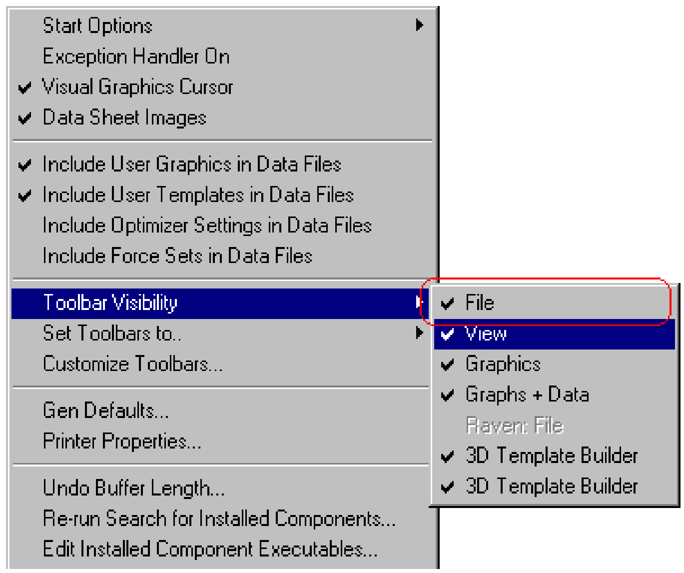
The text results can also be listed as a series of spline fits rather than tabulated data. The user has control of which spline to list, and the power of the spline fits. This provides a method of exporting suspension properties to external spline based full vehicle handling applications.



Sample Spline Results Display

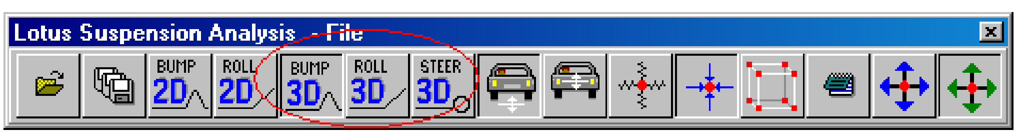
2.7 Bump, Steer and Roll Kinematics

➢ ***Display the File Toolbar from the SetUp menu by selecting 'SetUp / Toolbar Visibility / File'***



Displaying the File Toolbar

The suspension articulation type can be bump/rebound, roll or steering. Steering articulation is applicable to 3D front suspension models only. The articulation type can be changed via the relevant toolbar icons, or the Module / Shark pull down menu options.



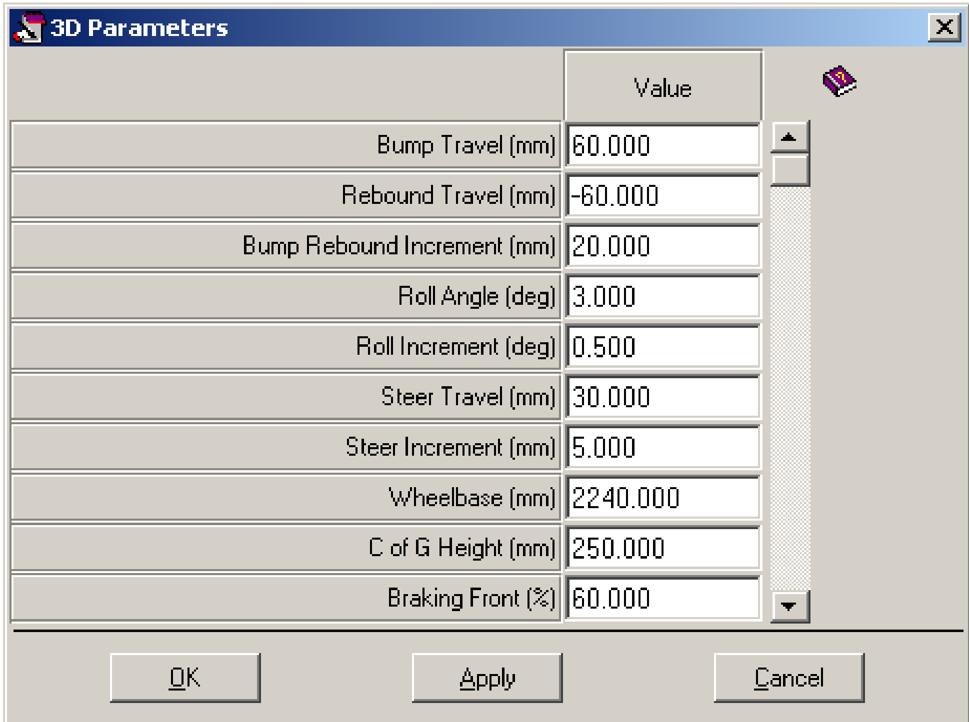
3D articulation type icons ringed

Changing the articulation type will change any displayed result graphs to show the same variables, but over the new articulation motion range (roll, bump, steer). Graph y-axis scales may need to be re-set to show the new results.

➢ ***In turn select each of the '3D bump' '3D Roll, and '3D Steer' articulation Icons on the File Toolbar, and note how the results displayed on each graph change for each motion type.***

An additional combined bump, roll and steering mode is available. Users define each point separately through an interactive display.

Each articulation type range is controlled by user-defined limits. These can be changed via the '*Data / Parameters*' main menu. They can also be set for specific articulation positions.



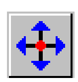
Parameters Data Display

Now we can move the suspension hard points and see the effect on the suspension kinematics

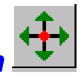
.

2.8 Points editing

Suspension hard points can be modified by one of three ways, using input edit boxes, using the keyboard arrow keys, or by dragging hard points with the mouse.

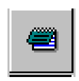
- ***Click the 'Set to Joggle Mode' icon***  ***on the File Toolbar and left click on one of the left inboard suspension ball joints.***
- ***Select from the menu 'Tracking / All' so that both up/down and left/right arrows are displayed over the suspension point you have selected.***
- ***Manipulate the suspension hard point by holding down the 'Ctrl' key, then pressing the arrows on the keyboard.***

As the suspension point is moved, the suspension geometry graphs will be continuously updated and the position of the suspension roll centre will move on the 3D display. The roll centre is displayed as a blue circular dot on the display. Toggling between 3D bump, steer and roll using the 'File' Toolbar icons will update the graphs for each of the kinematic motion types.

- ***Select the 'Set To Drag Mode' icon*** 

- ***Now left click and hold a suspension hard point. This can now be dragged around the screen and the results will be continuously updated on the graphs as you move the suspension.***

The suspension hard points can be moved in the front plan and side views. The suspension is modelled symmetrically: any change to one side is mirrored on the opposite hard point.

- ***Select the 'Set to Edit' icon***  ***on the File toolbar and left click on a lower inboard suspension point.***

- ***In the edit box add 5 mm to the 'y' co-ordinate and click 'OK'. The suspension will now have moved to the new location.***

When in edit mode, 'tracking lines' are drawn to indicate the current 'tracking' direction(s). This is not relevant to the hard point-editing mode as tracking only applies to the dragging and joggle edit modes. The right mouse button will cycle through the available tracking direction options. A similar action is achieved by selecting the mouse icon from the 'view' toolbar.

Hard point joggling operates in a similar way to dragging, with regard to available directions. The drawn joggle symbol indicates the number of joggle directions available. To use joggle select either *Ctrl + Arrow* Key for coarse joggle or *Shift + Arrow* Key for fine joggle. The joggle fine size is a tenth of the coarse size, the coarse size can be set via SetUp / Gen Defaults…

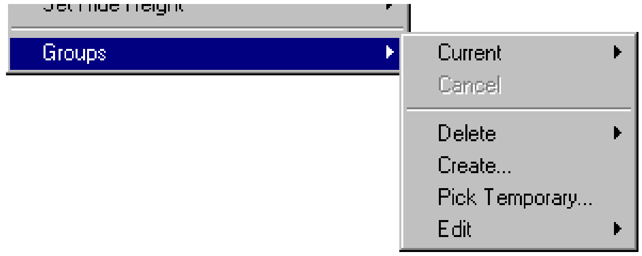
For a full description of the suspension hard point editing options, refer to the help file, 'Overview – Hard Point Editing' and 'Overview – Hard Point Dragging'.

LSA can also be set up to retain the length of suspension parts when modifying the suspension. In this mode the whole suspension moves to satisfy the new hard point location, without changing any suspension part lengths.

- ***From the menu bar, select 'Edit / Change mode / Retain Parts'. Now try dragging suspension hard points with the mouse.***

- ***Return 'Change mode' to 'Change Point Positions'***

It is also possible to move a group of points, by using the options provided under the *Edit* menu:



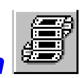
Groups options

➢ ***Try to create a group and use it as current to move some points together with the drag mode.***

2.9 Animation Suspension Kinematics

The suspension can now be animated to give a movie of the suspension movement in bump, roll and steer. The movement of the roll centre is also displayed in the animation, and the suspension hard points can be edited during the animation.

➢ ***Display the graphics tool bar by using the SetUp menu 'SetUp / Toolbar Visibility / Graphics'***

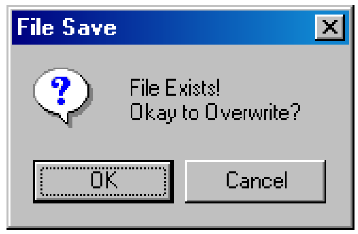
➢ ***Left click the 'Animate mechanism' icon***  ***on the graphics toolbar.***

➢ ***Whilst the animation is displayed, switch between bump, roll and steer modes.***

➢ ***As the suspension is animating, try dragging suspension hard points and see the effect on the roll centre location in roll, bump and steer.***

2.10 Saving Data Files

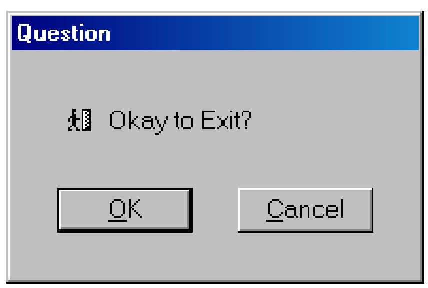
Models can be saved in the conventional way using the *File / Save* or *File / SaveAs* menu items. You will always be warned about overwriting existing model files. Data files will include all suspension hard point data, compliant bush properties and model parameters. What it does not necessarily include is the template definition. A data file can refer to the template via an entry number, if the 'include User Templates in Data File' option is not checked. For further information on the definition and storing of suspension templates, see the template sections in this document.



Okay to Overwrite dialogue

2.11 Closing the Application

To close the program select '*File / Exit'* from the main menu, and then confirm the 'okay to exit' prompt. Alternative methods to close the application include the conventional 'X' from the windows top right corner, Alt+F4 or close from the main windows top left menu. In addition, the 'esc' key will close the application (subject to accepting the prompt).



Okay to Exit Prompt

3 Interactive Template Modification

3.1 Overview

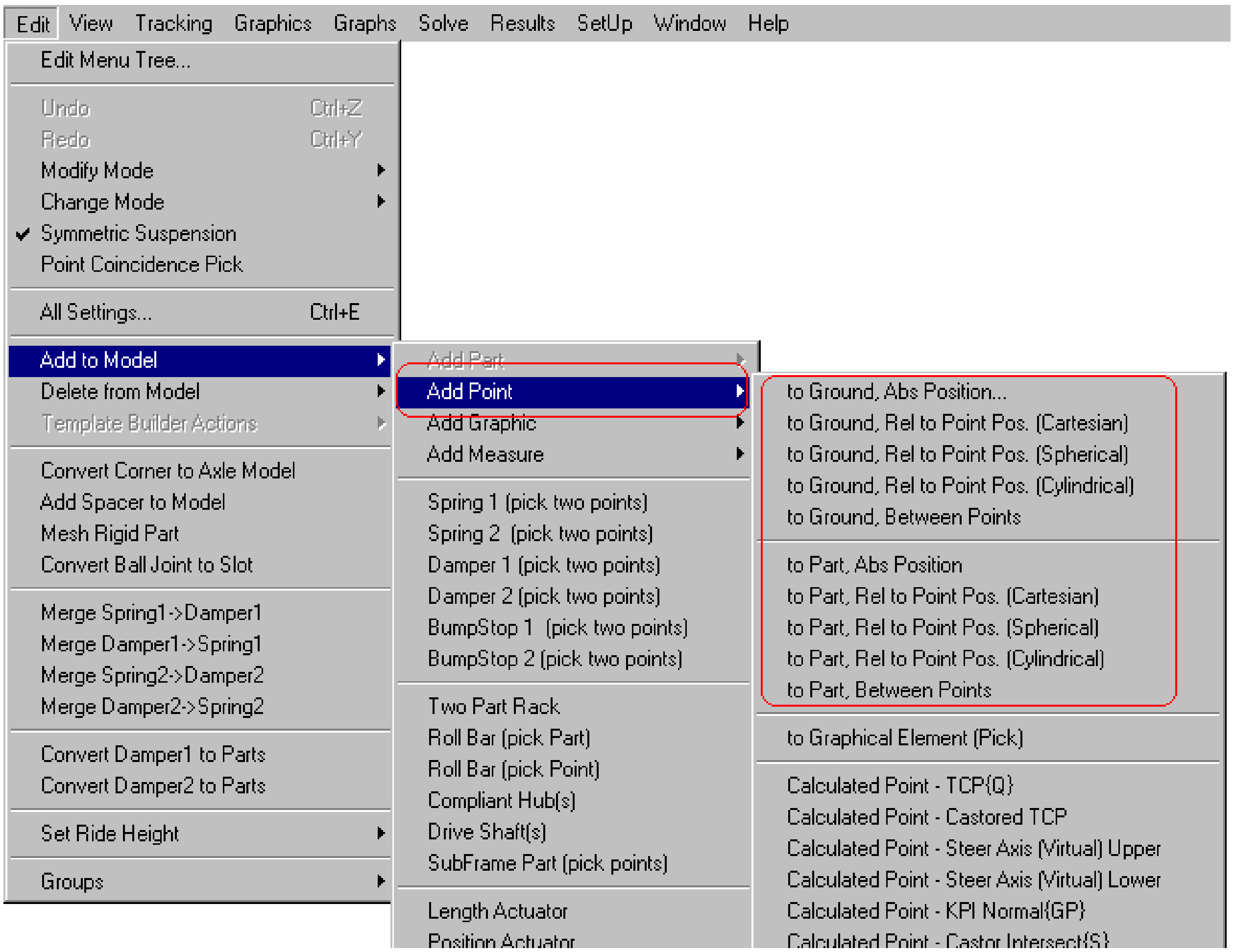
A number of convenience menus exist in the ‘Edit’ menu that modify the template without the user being required to open the template editor. They operate on the current visible model with mouse selections and keyboard inputs to provide simple methods by which some of the more popular template modifications can be easily performed.

To improve the functionality of simple model template editing, a number of menu options were added to the main ‘Edit’ pull down menu that support some of the more common template editing activities. It should be remembered that ‘graphics’ elements also form part of the template definition. In a similar way, ‘graphics’ can be added interactively through the ‘Graphics / Add’ main menu, to modify the template graphics definition.

This chapter contains the following sections:

3.2 Adding Points to the Template

A number of menu options are available to add a hard point to the current model's template. They are sub-divided into three categories: adding the point to 'Ground', adding the point to 'Part', and adding 'Calculated' points.



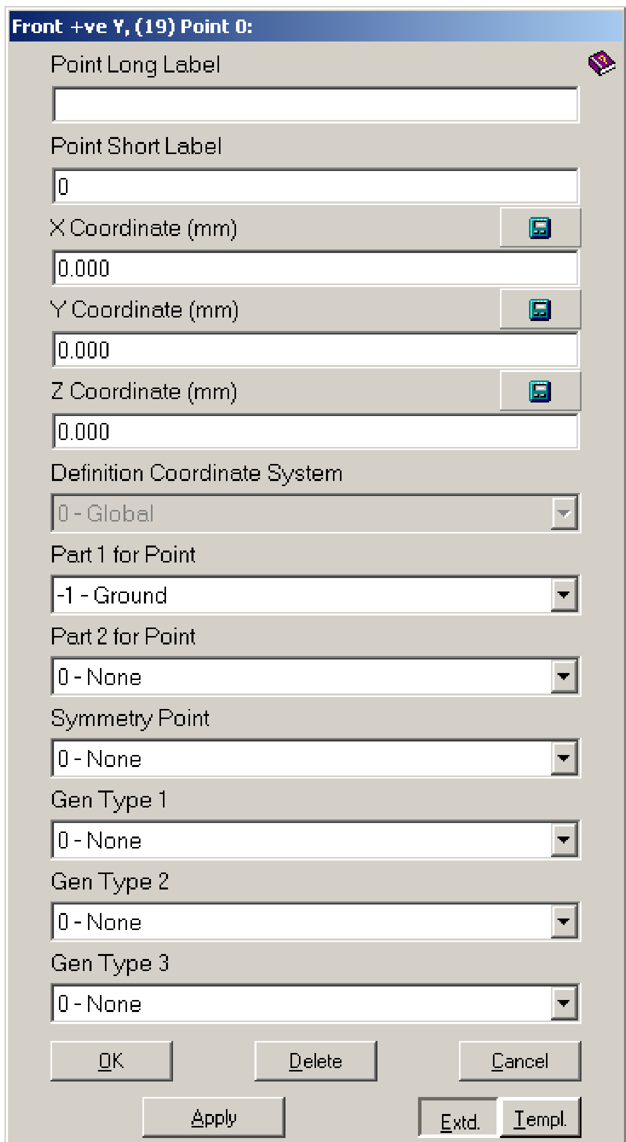
Adding Point – Menu Options

Adding point to Ground

Five alternatives are given for adding a point to the ground: an absolute position, relative to an existing point, and between two existing points.

1) Add Point / to Ground, Abs Position

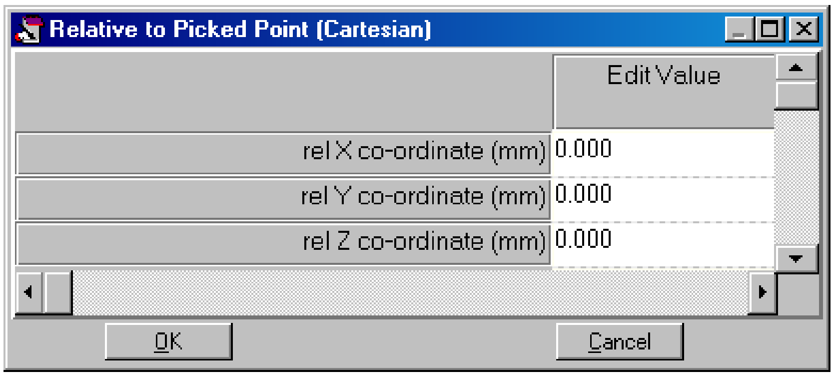
For this option simply complete the displayed dialogue box giving, label, template local point No. and global co-ordinates of the point. The ‘part 1 for point’ box should be left as ‘-1 Ground’ and ‘part 2 for point’ should be left as ‘0 – None’.



Adding Point to Ground Abs position

The new point will be added to the template and the re-drawn graphics will show the new point.

2) Add Point / to Ground, Rel to Point Position (Cartesian)
For this option, the graphical display is re-drawn with just hard points associated with 'ground' being drawn. With the left mouse button, select the required hard point that you will define the relative position to (you can cancel this action by using the right mouse button). Having selected the hard point, enter the required relative co-ordinates for the new point in the listed dialogue box.



Adding Point to Ground Relative position

The new point will be added to the template and the re-drawn graphics will show the new point. To change the label and template point No. assigned to this new point, change to edit mode and select it for editing in the normal way.

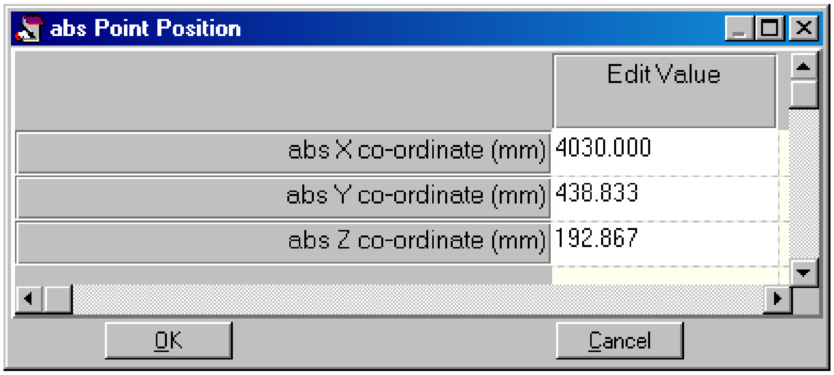
Similar options are given for adding points using Spherical and Cylindrical definition methods.

5) Add Point / to Ground, Between Points
For this option, the graphical display is re-drawn with just hard points associated with 'ground' being drawn. With the left mouse button, select the two existing points between which the new point should be added (you can cancel the picking action by pressing the right mouse button). The new point will be added to the template midway between the two picked points.
The re-drawn graphics will show the new point. To change the label and template point No. assigned to this new point, change to edit mode and select it for editing in the normal way.

Adding point to Part

Five alternatives are given for adding a point to a part: absolute position, relative to an existing point, and between two existing points.

1) Add Point / to Part, Abs Position
For this option the graphical display is re-drawn with Part labels turned 'on'. The user must then pick the required parts centre that you wish to add the point to. Then enter the required absolute co-ordinates of the new point into the displayed dialogue box.



Adding Point to Part, Absolute position

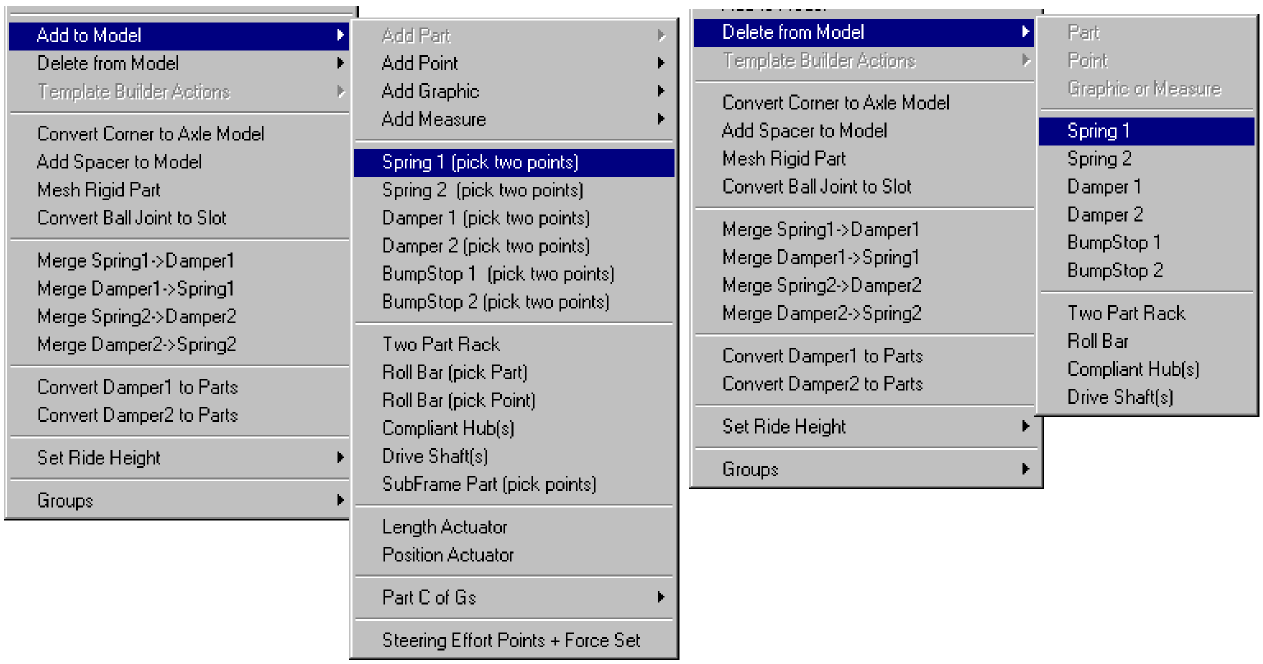
The new point will be added to the template and the re-drawn graphics will show the new point. To change the label and template point No. assigned to this new point, change to edit mode and select it for editing in the normal way.

2) Add Point / to Part, Relative to Point Position (Cartesian)
For this option, the graphical display is re-drawn with Part labels turned 'on'. The user must then pick the required parts centre that you wish to add the point to. The graphical display is then re-drawn again showing only the selected part and its associated points. Now, pick the required point that the new one is to be defined relative to. Finally, enter the required relative co-ordinates of the new point into the displayed dialogue box. As with the previous options, the right mouse button can be used to cancel the action. The new point will be added to the template and the re-drawn graphics will show the new point. To change the label and template point No. assigned to this new point, change to edit mode and select it for editing in the normal way.

5) Add Point / to Part, Between Points
For this option, the graphical display is re-drawn with Part labels turned 'on'. The user must then pick the required parts centre that you wish to add the point to. The graphical display is then re-drawn again, showing only the selected part and its associated points. Now pick the required two points that the new one is to be placed between. As with the previous options, the right mouse button can be used to cancel the action at any time. The new point will be added to the template and the re-drawn graphics will show it. To change the label and template point No. assigned to this new point, change to edit mode and select it for editing in the normal way.

3.3 General Addition and Deletion

This collection of menu options provides easy addition and deletion of some specific model elements. Each is discussed individually below.



General Addition and Deletion Menu Options

Spring 1: If a template does not have a ‘spring1’ element, it can be added by picking the two end points with the left mouse button. The first pick is for the ‘ground’ point (normally referred to as the upper point), whilst the second picked point is the suspension end (normally referred to as the lower point). The template is modified and the graphics re-drawn showing the new model. The right mouse button can be used to ‘cancel’ the picking actions. You cannot add a ‘Spring 1’ element if one is already defined in the model (delete first if you want to modify the connections but don’t want to use the template editor).

Spring 2: Functions exactly the same as for Spring 1 above.

Damper 1: If a template does not have a ‘damper1’ element, it can be added by picking the two end points with the left mouse button. The first pick is for the ‘ground’ point (normally referred to as the upper point), whilst the second picked point is the suspension end (normally referred to as the lower point). The template is modified, and the graphics re-drawn showing the new model. The right mouse button can be used to ‘cancel’ the picking actions.

Damper 2: Functions exactly the same as for Damper 1 above.

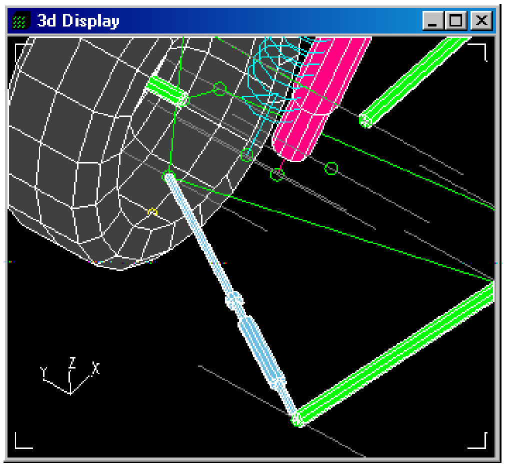
Bump Stop 1: If a template does not have a ‘bumpstop1’ element, it can be added by picking the two end points with the left mouse button. The first pick is for the ‘ground’ point (normally referred to as the upper point), whilst the second picked point is the suspension end (normally referred to as the lower point). The template is modified and the graphics re-drawn showing the new model. The right mouse button can be used to ‘cancel’ the picking actions.

Bump Stop 2: Functions exactly the same as for Bump Stop 1 above.

Note: For non-linear bump stops, some resultant forces calculated using the user defined bump stop curve, will be incorrect if the compliant displacements are ‘large’. This is due to the compliant solver linearizing the rate at a certain kinematic position, to compute the force it will apply to the system. Once this force is applied, the compliant displacement if large will change the kinematic position, leading to a new operating point of the bump stop (thus rate change), which is not taken into account to find a new bump stop force. To reduce this effect set the tyre vertical rate to a high value to stop the large displacements.

Each of these preceding four items can be removed from a template using the appropriate ‘delete’ menu entry. Note that the delete does not remove the associated points. Only the tag of the points (in ‘file/edit template/settings’, ‘gen type’ column) is deleted (ex: ‘damper 1 to suspension’). To delete a point use the standard edit method and the optional ‘delete’ button.

Length Actuator: A length actuator can be added to interactively control the length between two points. Pick the required end points with the left mouse button. The graphics will be updated to indicate the change in the template. To change the properties of the added actuator, change to edit mode and pick the actuator.



Length Actuator Added to Model – Blue Element

Position Actuator: A position actuator can be added to interactively control the location of a hard point. Should only be applied to points attached to ground (i.e. the body). A single point pick is required with the left mouse button. The graphics is then re-drawn, to indicate the change in the template. To edit the properties of the actuator, change to edit mode and pick the actuator.

Part C of G's: Three menu options are available to add a C of G point to a defined part. These C of G points are used in specific calculations such as 'Un-sprung corner Weight', 'modal analysis' and 'Forced-damped response'. The three 'adds' functionality is identical to that for the 'add point to part' discussed earlier. Remember that C of G's are only visible in compliant mode and only then if the specific visibility is set to 'on'. To change the C of G properties set to 'edit' mode and pick the required C of G point.

3.4 Adding Calculated Points

This collection of menu options provides simple selection of calculated points that can be added to the model. Each is discussed individually below.

Calculated points, as the name implies, are derived from the positions of other defined model points. They are thus neither editable nor defined directly. They can however be used for graphical display either individually or referred to by graphical elements.

TCP – The Tyre contact point with the ground plane. The kinematic position of this point is based on the lowest point of a zero thickness disc around the stub axle. This 'low' point is recalculated at each increment.

Castored TCP – The incremental position of where the original 'static' TCP point has moved to. Hence the term castored as the original point will roll around the new point of contact and hence at increments is different from the TCP point above.

Steer Axis (Virtual) Upper – For steerable templates that do not have a single ‘tagged’ upper ball joint, this calculated point is a point on the ‘virtual’ steer axis above the upper locations. It is normally coupled with the next item to graphically draw a virtual steer axis.

Steer Axis (Virtual) Lower – For steerable templates that do not have a single ‘tagged’ lower ball joint, this calculated point is a point on the ‘virtual’ steer axis below the lower locations. It is normally coupled with the previous item to graphically draw a virtual steer axis.

KPI Normal – Intersection point on the Kingpin Axis of the normal between the Kingpin axis and the wheel spindle axis.

Castor Intersect – Intersection point of the steering axis with the ground plane.

Spindle Normal – Intersection point on the wheel spindle axis of the normal between the Kingpin axis and the wheel spindle axis.

Spindle/Damper Normal – Intersection point on the wheel spindle axis of the normal between the Damper axis and the wheel spindle axis.

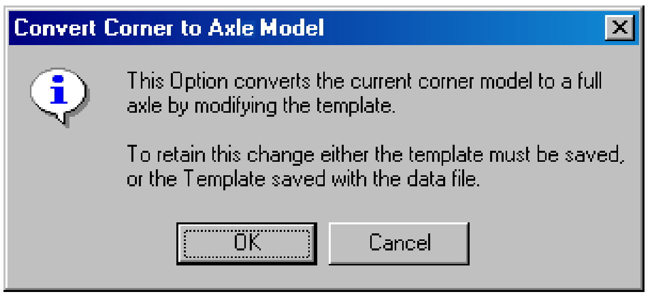
Damper Normal – Intersection point on the Damper Axis of the normal between the Damper axis and the wheel spindle axis.

3.5 Merging Springs and Dampers

A number of the ‘standard’ templates have springs and dampers defined separately. In the case of ‘coil-over’ spring/damper units, where they can use the same hard points, these ‘merge’ convenience function allow the user to communize on one pair of the points. The unused pair of hard points is removed from the template.

3.6 Converting Corner to Axle Model

Most ‘standard’ templates for independent suspension are for a single corner. In some instances it is necessary to model a complete axle. For examples, this would be required when modelling anti-roll bars, sub frames, compliant steering racks. This single click menu option mirrors all points and parts across to the other side, modifying the template to set up the new connections and bushes.

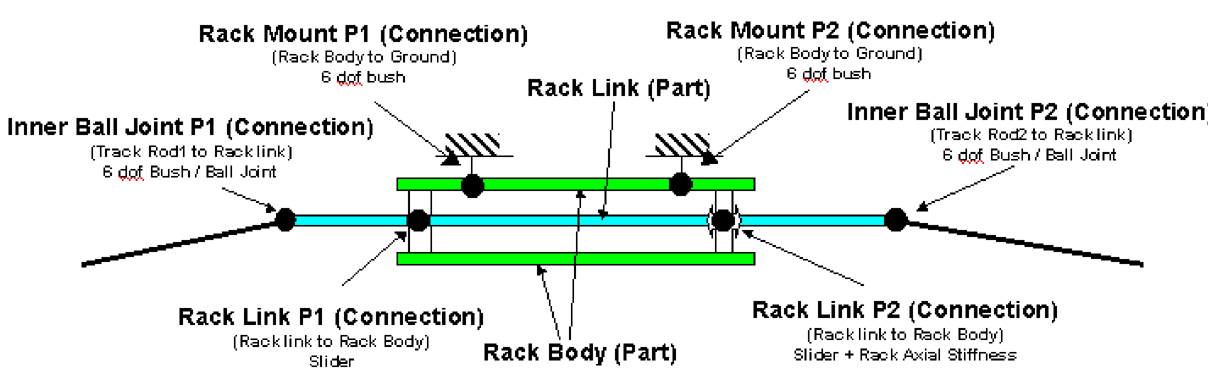


Converting to Axle Model Prompt

The new model is redraw reflecting the change to the template.

3.7 Add Two Part Rack to Model

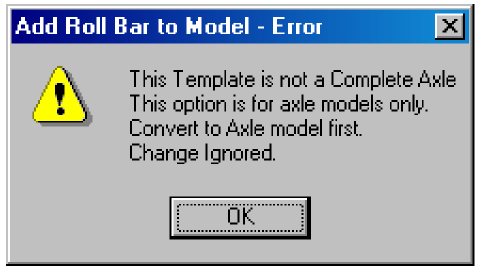
For full compliant analysis there is often a requirement to have the rack connecting left and right hand suspensions for force transfer. This single click menu option adds a compliant rack to the existing models front template. It can only be applied to a full axle model, (this is checked for as part of the action). The rack parts are connected between the two inner track rod points, these will need to have already been ‘tagged’ in the template (all standard front suspension templates have this, since this is what identifies it as a ‘front’ suspension). The model structure for the two-part rack is shown below.



Two Part Rack Model

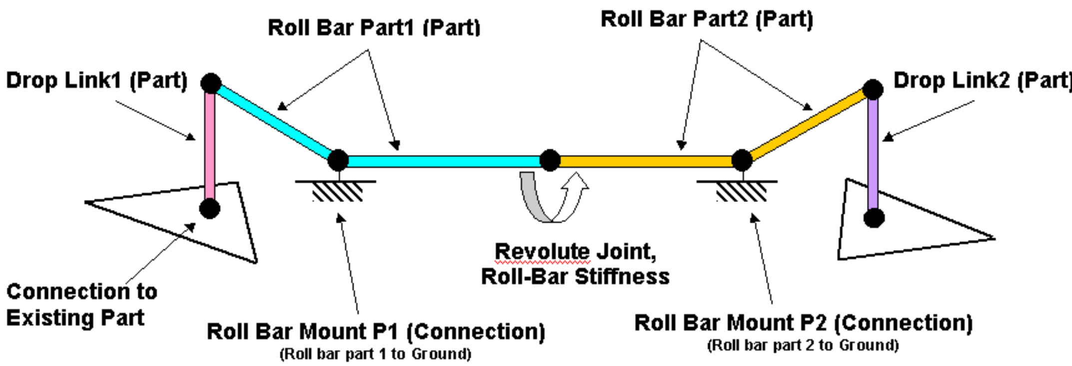
3.8 Add Roll Bar to Model

Including an anti-roll bar to a template requires that the current template is modelled as a full axle (so convert to a full axle if not yet done so).

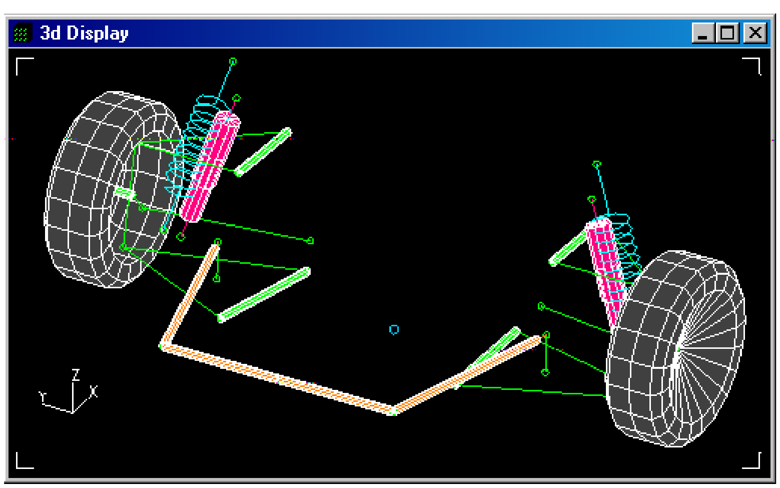


Full Axle Warning – Adding Anti-Roll Bar

On adding the roll bar, the graphics will be re-drawn to show part centres. You must select the part that the roll-bar drop link will be attached to with the left mouse button. Having selected the part defines the absolute co-ordinates of the attachment point. Once entered, the template is modified.



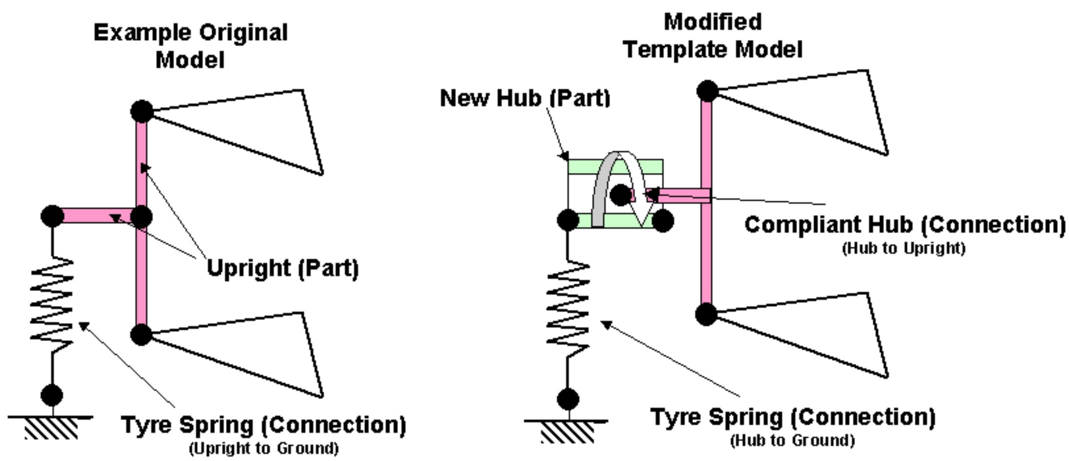
Anti-Roll Bar Model



Anti-Roll Bar Added to Model

3.9 Add Compliant Hub to Model

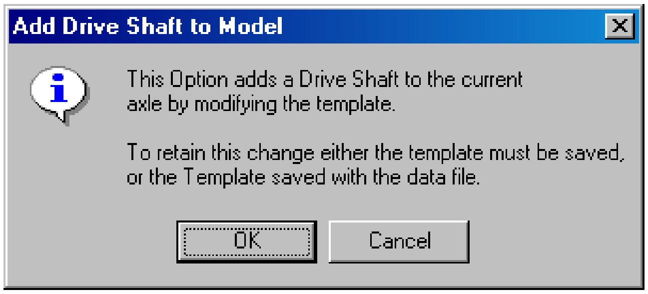
The ‘Add compliant hub’ option provides a simple menu selection route to including hub compliance into the existing model template. It adds a new part, the ‘wheel/Hub’ between the upright and ground. Two new points are added one for the new parts C of G position and the other for the connection point. The compliant hub is modelled with a single bush, rather than the more physical two bushes (i.e. the inner and outer bearings), as hub compliance values are usually measured as a single stiffness number. In compliance mode, if no bush stiffness values are provided, the default ‘Stiff’ values are applied to both axial and rotational stiffness. As part of the template modification performed by this option, the wheel centre point and stub axles points properties are changed such that they are associated with the new ‘hub’ part rather than the original ‘upright’ part.



Compliant Hub Model

3.10 Add Drive shaft to Model

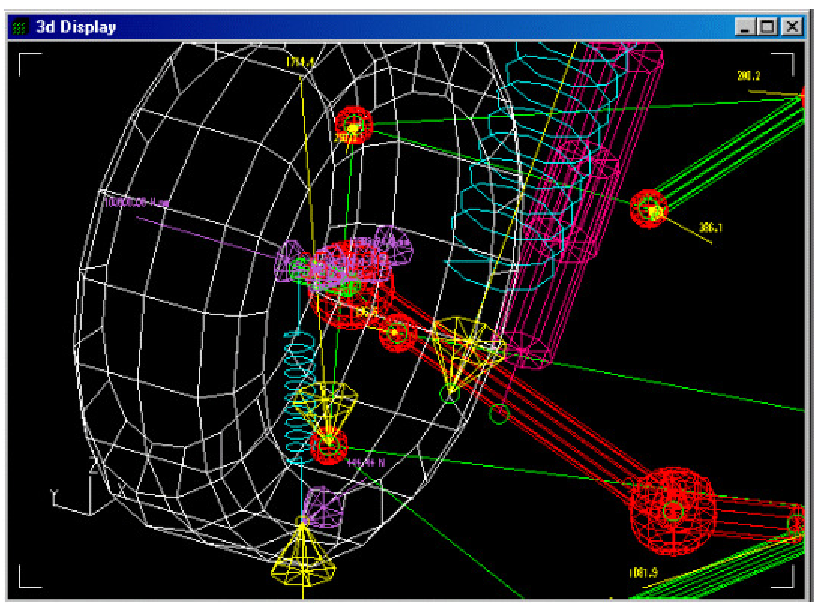
To provide a mechanism by which drive shaft loads can be applied to a model, a representation of the drive shafts is needed. This option adds the necessary points and modifications to the template such that the solver can calculate the associated drive shaft loads and torque’s that should be applied to the upright. This option does not add any parts and as such does not change the main kinematic solution of the mechanism. The changes only involve ‘tagging’ of the stub axle point to indicate the outer CV joint centre, and the addition of two points to identify the drive shaft inner and centre and axis. Four graphical elements are added for visualization, (they do not imply physical parts in the model i.e. they have no mass), connecting the ‘tagged’ and new points.



Running the 'Add Drive Shaft' option

Drive shaft loads include the definition of a torque value (one for each corner), and an optional 'loss' table that defines the efficiency of the inner and outer joints, based on the instantaneous joint angles. A separate solver switch controls the application of drive shaft loads to the model.
An additional option is involved with drive shaft loads, and is intended to allow users to choose between brakes on and off when loading via the drive shafts. Drive shaft loads are shown on the graphical display in a unique colour.

Three drive shaft types are available. The first is a fixed length drive shaft, with plunge taken on the inboard sliding joint. The second is a fixed length drive sahaft, with the plunge taken on the outboard end. The third is a varying length drive shaft, the inboard joint position being fixed and the plunge accommodated in a sliding joint between two parts of the shaft.

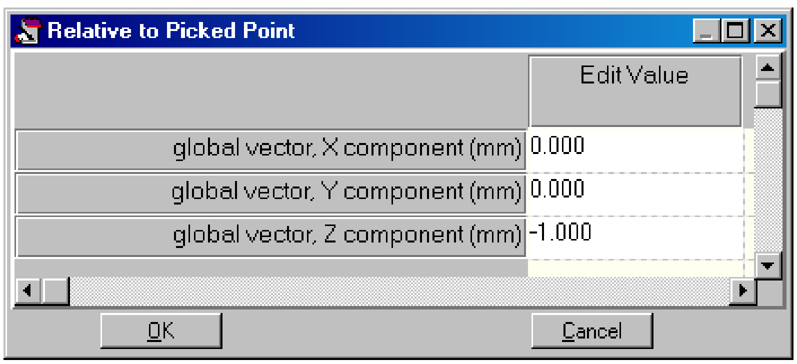


Drive Shaft added – Drive Shaft loads shown

3.11 Add Spacer to Model

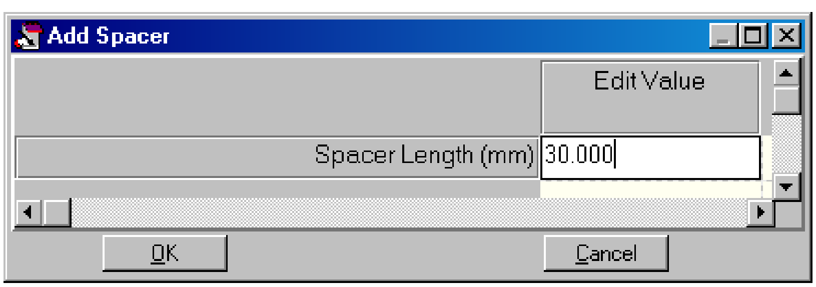
This option adds a ‘spacer’ to the model. A ‘spacer’ is intended to represent a physical part in a suspension that is used to statically adjust the suspension settings. It can be a spacer inserted between two moving parts, (such as wishbone to upright), or a spacer between a moving part and the body (ground). They can be considered as ‘shims’. Spacers have properties not just of connection but length and orientation. Both of these properties can be interactively edited/dragged. Spacers can also be added to the Component Toolbox for use in component mixing/adjustment, (see separate description of the component toolbox).

To add a spacer, the graphics are re-drawn showing part centres, as the first pick must be the part that the spacer is to be associated with. Having picked the required part centre with the left mouse button, the display is re-drawn to show only the selected part and its associated points. Again with the left mouse button, select the required point that the spacer is to be attached at. The right mouse button can be used to cancel this action at any point. Each spacer requires a global vector to be specified that defines the orientation of the spacer axis. Enter the global vector when prompted.



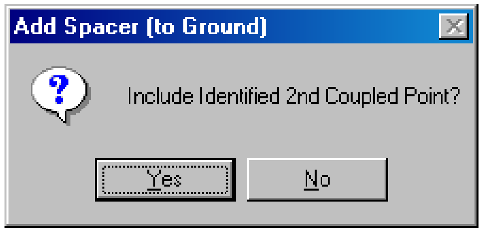
Defining the Spacer Global Vector

Having defined its orientation you will also need to specify its length. A zero length would imply no change to the model whilst a positive length will not only complete the addition of a spacer and modify the template it will also rebuild the current model by inserting the defined length spacer at the selected point and orientation.

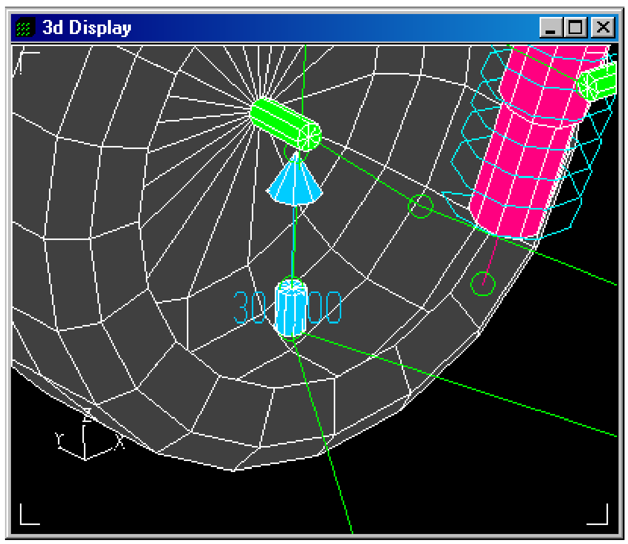


Defining the Spacer Length

The graphics are re-drawn and the model updated to reflect the change. One other option exists with spacers. For those that are attached between point and ground (body), the model is checked to see if there is a second point that could be considered to operate with the picked point. An example of this is a wishbone pivot axis, having picked one point, the second is identified as linked, and the option given of applying an identical spacer to each.



Confirming a Linked Spacer



Single non-linked Spacer Added to Model

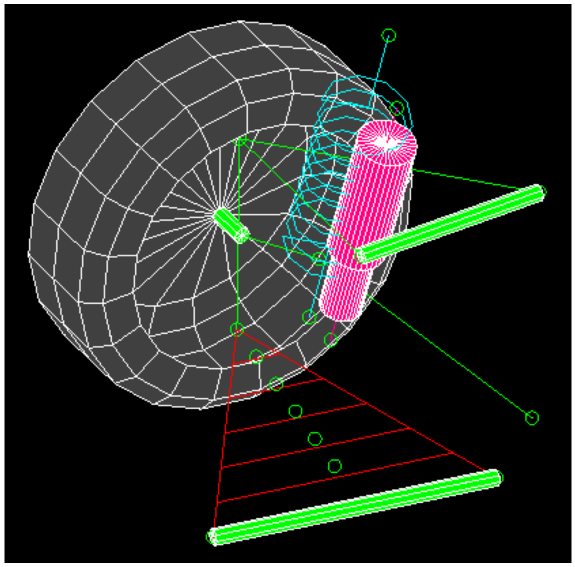
3.12 Mesh a Rigid Part

This utility is used to create a pseudo flexible part. It does so without any additional burden on the kinematic solver, (i.e. the number of unknowns/equations in the main kinematic solution is not altered). The meshing takes the selected part and subdivides it into ‘n’ smaller parts, each connected in sequence to the previous part by three bushes. Two of the bushes are termed ‘zero stiffness’ whilst the third is given translation and rotational stiffness values to control the overall part flexibility.

As a simple example create a new double wishbone front suspension model and select the option, *Edit / Mesh Rigid Part*, them select the ‘Lower Wishbone’ part centre. To mesh a part you need to identify three points which define how the part will be ‘meshed’. The first two points picked set the start points of the mesh with the mesh then proceeding in the direction of the third point. Thus for our lower wishbone we will pick the front inner pivot and the rear inner pivot as the first two points, then the outer ball joint as the third point. To complete the mesh you need to set the number of Sub-parts, enter 6, and the x, y and z offsets from the first picked point, (leave these as zero). Leaving them at zero places the 'stiff' connection points on an axis midway between the first two points. If you define actual non-zero values, then the 'stiff' axis position is based on the specified offsets.

As with all template modifications, to retain this template change either save the template with the model file or save the template to a user/custom file.

The modified model should look similar to that shown in the screen shot below.

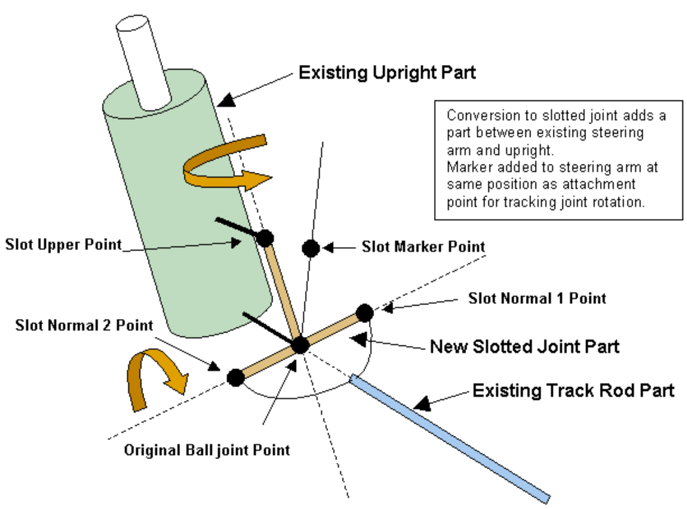


Example Meshed Rigid Part

The stiffness properties of the meshed part are then set by editing individual bush values just like any other bush.

3.13 Convert Ball Joint to Slot

This very specific option is the conversion of a simple ball joint (normally an outer track rod) to a reduced degree of freedom ‘slotted’ joint. The slotted joint is analogous to a special case of the universal joint. This menu option requires the user to select the ball joint to convert, then modifies the template by adding an additional part between the two original parts (i.e. between the track rod and the upright). Connections between the original parts and the new part are made such that the new joint operates as though the ball part is constrained by a ‘slot’ on the original track rod part.



Slotted Joint

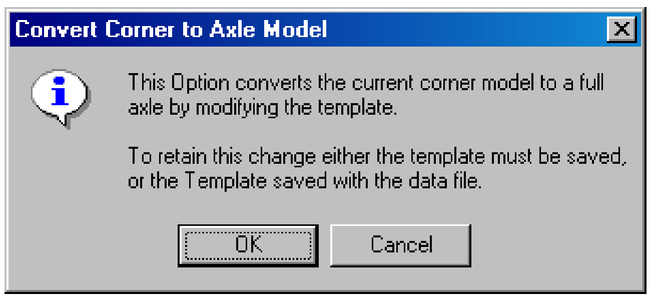
3.14 Exercise 1: Modifying a Models Standard Template

In this exercise we will use some of the interactive template modifying menus, described to build a more complex model, starting with the standard single corner double-wishbone. Open a new model using the front template type 1.

- ***File / New, Select Front, Set to Template type 1.***

We want to add an anti roll bar to the model as well as a two part steering rack. For both of the items, we need a full axle model: we will first convert the standard corner model into a full axle.

- ***Edit / Convert Corner to Axle Model.***

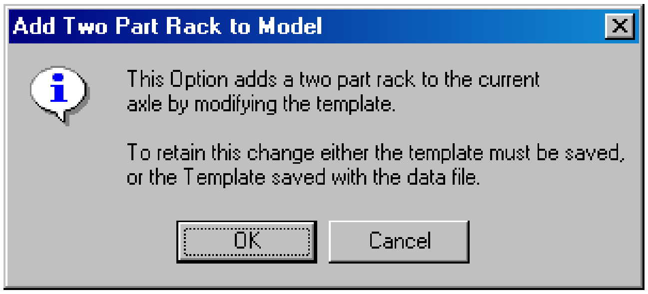


Converting to Full Axle model

Re-scale the graphics to show the full axle model. Converting to full axle with symmetry enabled will automatically identify and 'tag' symmetry points in the template. You can test this by dragging one of the hard points.

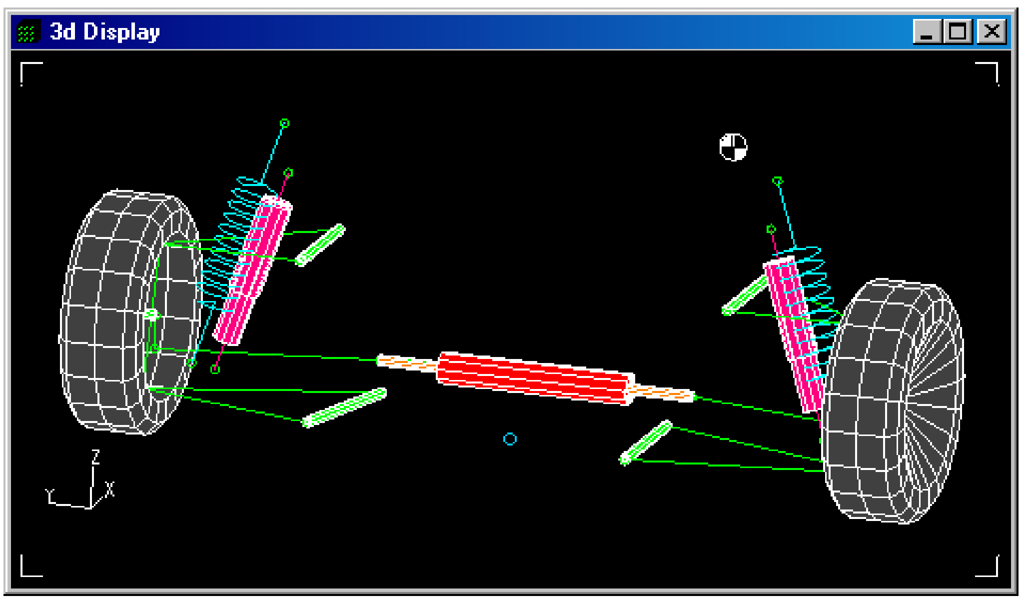
We will now add the two-part rack to the model.

➢ ***Edit / Add to Model / Two Part Rack.***



Adding the two part rack

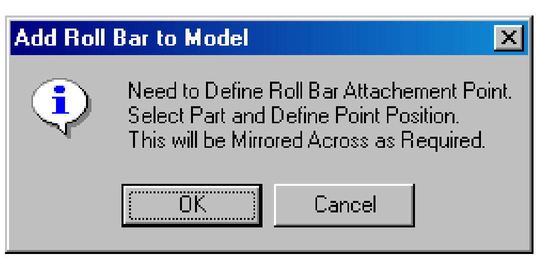
Your model should now look similar to that shown below.



The modified model

The last part of this exercise is the addition of the anti-roll bar. To do this we need to identify the part that the roll bar drop link will attach to.

➢ ***Edit / Add to Model / Horizontal Roll Bar (pick part).***

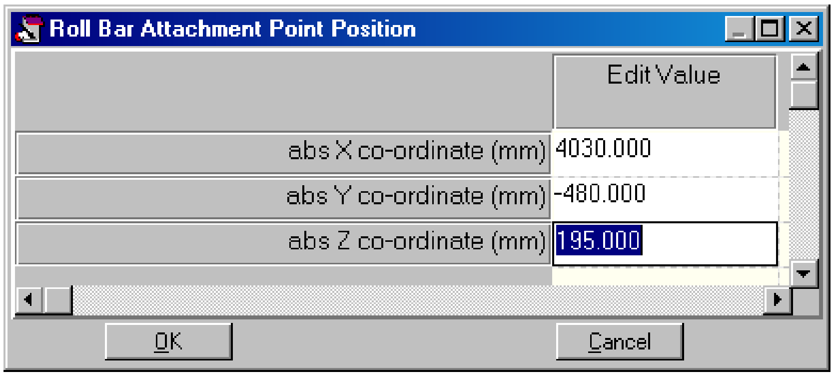


Adding the Roll Bar

➢ ***Ok.***

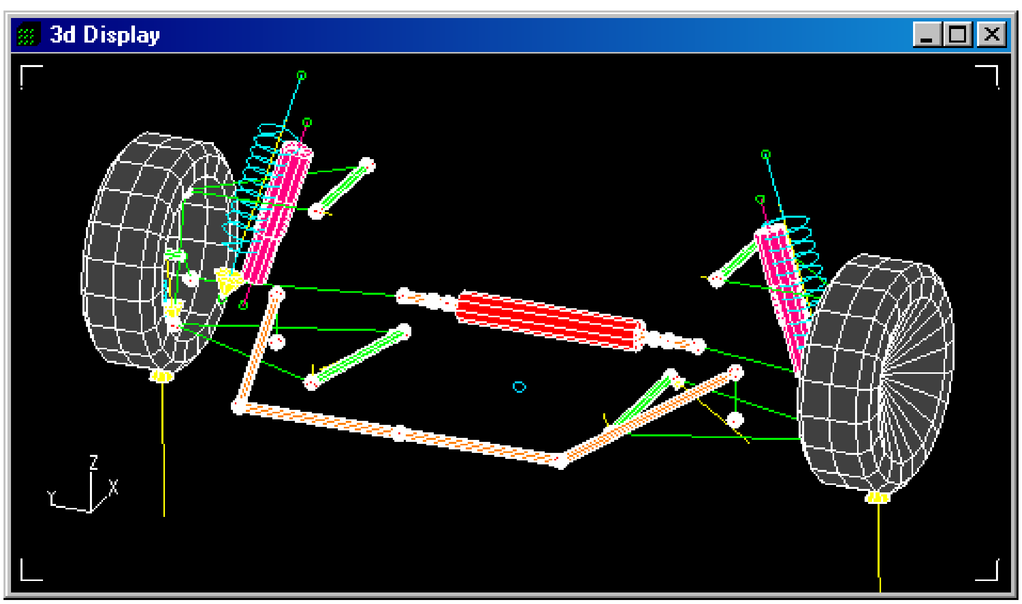
For this exercise, we will attach the roll bar to the lower wishbone. With the left mouse button, select the displayed part centre for the 'lower wishbone'. Note that as you hover the part centre, the far left of the status bar lists the part No. that you are currently on. It should say, "Front, Part 1 centre".

➢ ***Pick Part 1 Centre with the left mouse.***



Editing the connection co-ordinates

Now set the co-ordinates of the attachement point to be 4030, -480 and 195 and confirm the creation prompt. Your model should look similar to the screen shot below. Remember that these template changes would need to be saved either to a separate custom template file, user template, or saved with the data file.



Completed modified template

3.15 Exercise 2: Working with Spacers

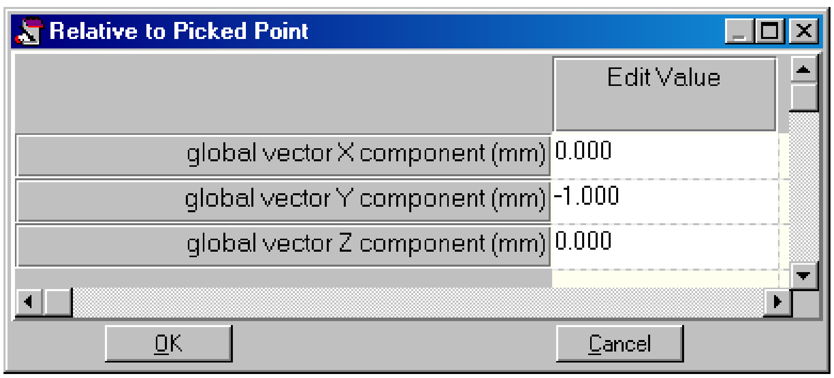
We will briefly add some spacers to a double wishbone template. To first revert back to the default double wishbone template, run the following menu option (note the example screen shots shown here are for a –ve y side corner model).

➢ ***File / Re-Read Default Templates (Skip All User).***

This will replace out recently modified template with the default double wishbone. We will first add a spacer to the lower wishbone pivot(s).

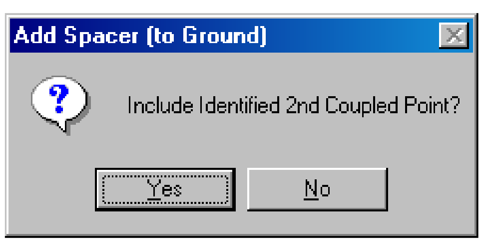
➢ ***Edit / Add Spacer to Model.***

Pick the lower wishbone in the same way as with adding the anti roll bar, by picking its now displayed Part 1 centre. Then pick Point 1, the lower wishbone front pivot. Now define the orientation of the spacer as 0, -1, 0



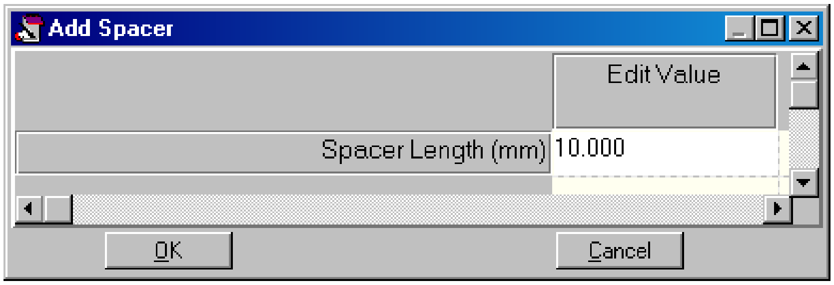
Setting the spacer orientation

For the lower wishbone front pivot, the application identifies a second dependent coupled point (point 2, lower wishbone rear pivot). The displayed prompt identifies that this is a coupled point, and the user needs to confirm that they are to be coupled (effectively means that two identical coupled spacers will be added to the model). Confirm the coupling, pick Yes.



Setting the spacer orientation

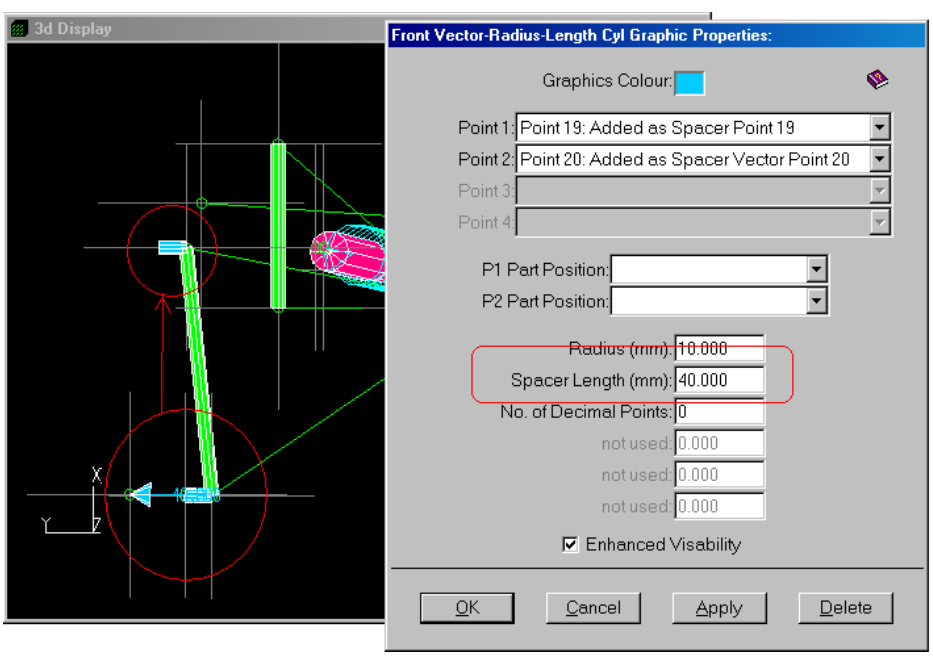
The final property of the spacer is its length, for now accept the default value of 10.0 mm then select Ok. The graphic should update to show two spacer cylinders and the spacer vector.



Defining the Spacer Length

Adding a spacer in this way will force the two hard points (points 1 and 2) to be 10mm further outboard, with all other suspension lengths remaining unchanged. Thus the suspension outboard points will be modified as the suspension is rebuilt, based on the two new lower wishbone points. The length and orientation properties of this spacer can be edited just like any other property in the model.

Try changing to 'edit' mode, and selecting the centre of the graphic cylinder representing the spacer with the left mouse button (remember to use the status bar prompt to check what you are selecting). Change the 'Spacer Length' property to 20, 30 and 40 use the 'apply' button to view the impact as you make the changes.



Editing the spacer (ringed) properties

4 Extended Travel Options

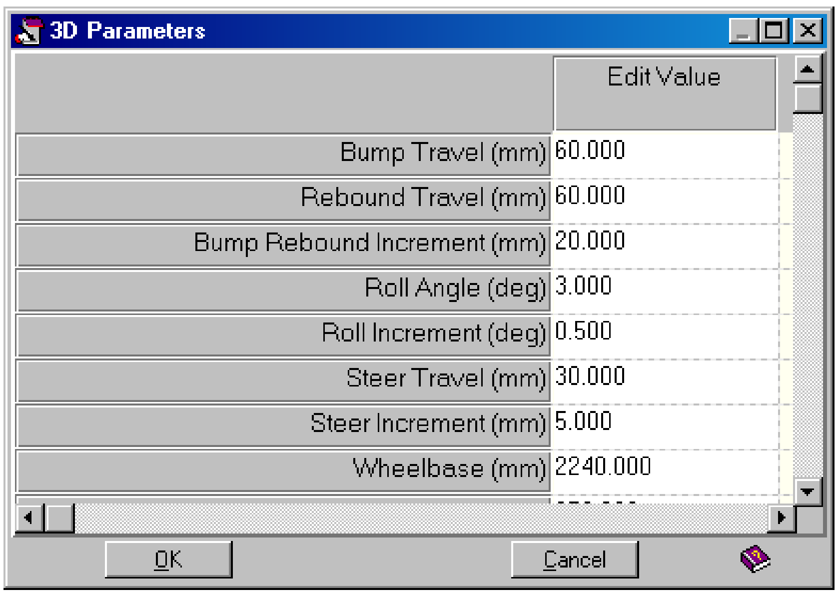
4.1 Overview

This chapter describes the use of the extended travel options. These are extensions to the standard bump, roll and steer displacement modules, as well as an alternative combined motion module.

This chapter contains the following sections:

4.2 Introduction

The default method of controlling the displacement of the defined model is to select the required displacement type from bump (and rebound), roll, steering or combined. Each of these displacement types in standard form is defined by a maximum displacement(s) and a step size. In the case of bump/rebound, each direction has its own maximum displacement, whilst for roll and steer their maximum value is mirrored (i.e. taken as +ve and –ve).



Setting the Standard Displacement Limits – 3D Parameters

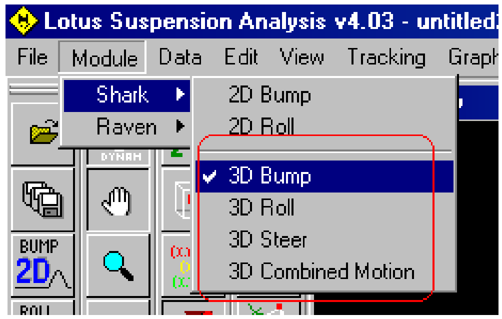
For Bump displacement, the default travel is set as the vertical movement of the ground plane (or body).
For Roll displacement, the travel is set as the roll angle of the body about the static roll centre axis.
For Steer displacement, the default travel is the linear motion of the steering rack inner ball joint. For steering boxes, it is the rotation angle of the steering arm.

Options are available to change these default travel settings. The bump displacement, when not in moving body mode, can be the ground point (default), the wheel centre, the lower ball joint or the upper ball joint (see *'solve/motion'*).

In addition to the above option, the displacements can be either the change in value (default), or the new absolute position value. This applies to Bump and steering articulation types (and hence also the combined mode).

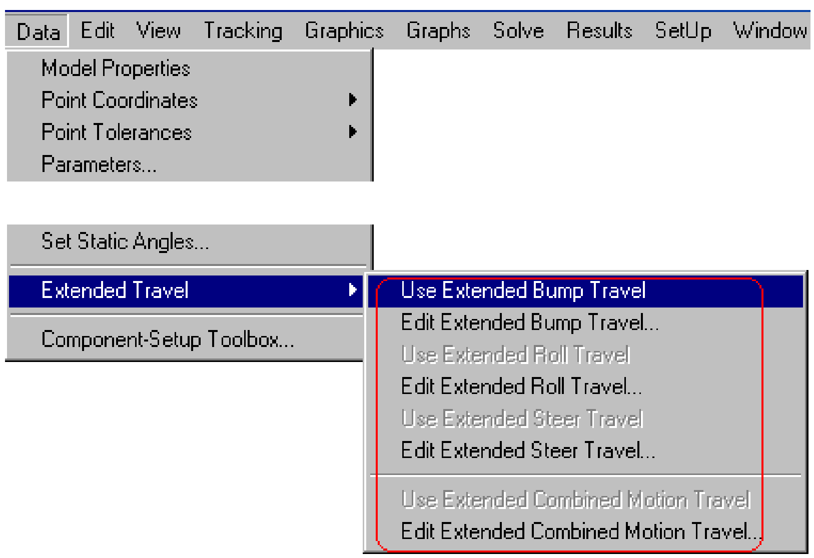
4.3 Setting the Extended Module Type

Each of these displacement modules is set through the 'Module' section of the main menu bar.



Setting the Displacement Module Type

Each of the three 'standard' displacements and the combined mode has an extended option, where rather than use the limit and step size value, you define the number of displacement points and their actual values. To enable the extended mode select from the data pull down menu the required extended module switch.

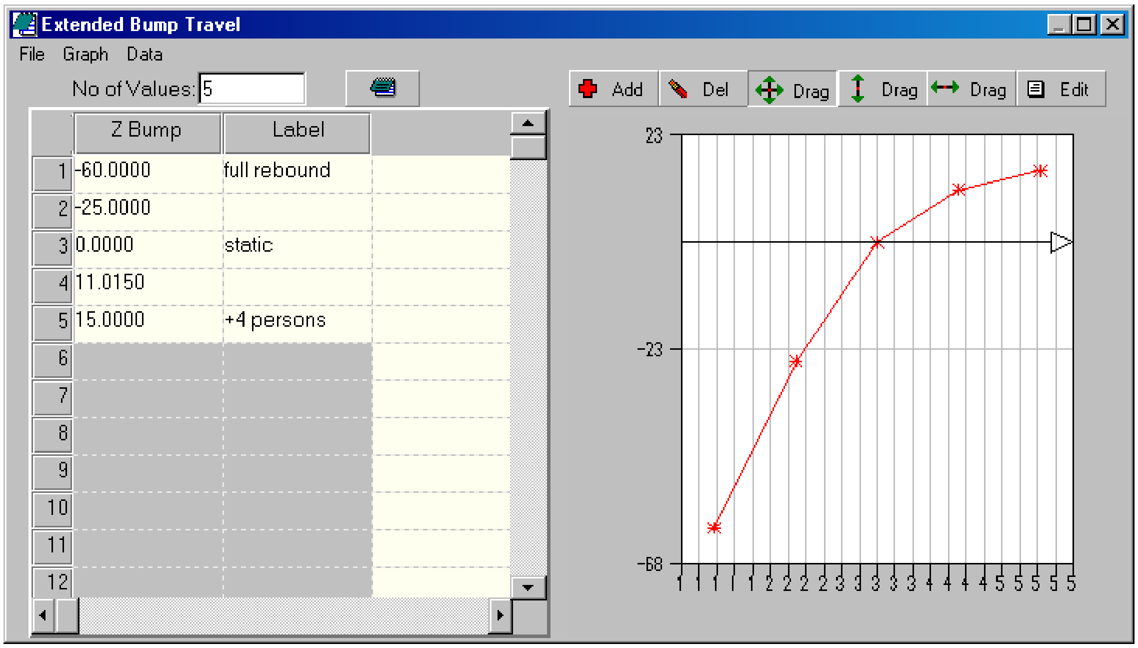


Enabling the Extended Option

The extended menu option is 'checked' when enabled.

4.4 Defining Extended Module Displacements

To define the extended travel positions, select the relevant ‘Edit Extended *** Travel’ option from the Data menus. In the opened dialogue box set the number of required points and their values. Each point can also be given a label to aid identification and use in result’s display.



Editing Extended Bump Travel

For Roll and Steer extended modes, you only need to define the +ve travel values. They will be repeated for the –ve side. With bump/rebound travel you define the +ve and –ve motions separately, where +ve is the bump direction and –ve is the rebound direction. In the combined mode, you can have different bump displacements on each wheel, thus two columns are given for bump travel, Zbump1 and Zbump2.

Extended Travel data settings are a property of the model and are thus saved with the model data file. They are not saved to the INI file.

4.5 Exercise: Extended Bump Travel

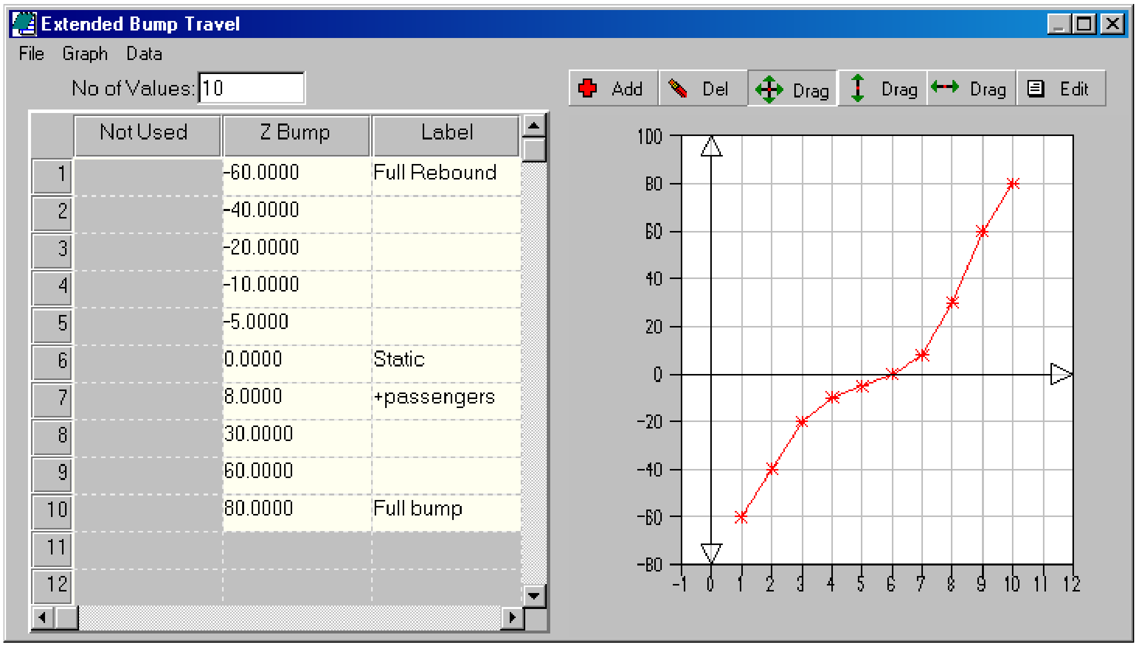
In this short exercise we will take a standard model and define a set of user defined bump solution points, using the ‘Extended Bump Travel’ option.

Open a new front model using suspension type 3 and open two graphs for Toe and Camber, set to 3d Bump travel (you shouldn’t need the commands for this any more).

As normal, note the even spacing of the results on the graphs. We will now switch to user defined bump travel having first entered the required numbers.

➢ ***Open the Extended bump travel data. Menu Data / Extended Travel / Edit Extended Bump Travel.***

Enter 10 defined displacements into the display as illustrated in the screen shot below. Note that you do not use column 1 and that you can enter point labels into column 3. Try experimenting with the graph 'dragging'.

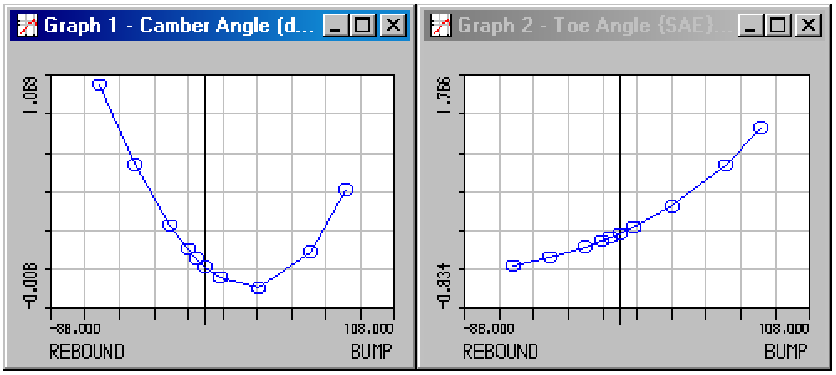


Editing the Extended Bump travel

To use these user defined extended travel points, ensure you are in 3D bump mode, and the enable the extended bump travel option.

➢ ***Switch on Extended Bump Travel, select Data / Extended Travel / Use Extended Bump Travel.***

The graphical display should change to indicate the non-even spread of calculated points. You will need *Graphs / visibility / Point Symbols* to be 'on' to appear the same as the images below.



Graph Results, Extended bump travel

Remember that this extended data is saved with the model file and not to the local INI file.

Repeat the above exercise, but try and use the *Solve / Motion / Ground Plane Options / Z Displacement as Position* and *Solve / Motion / Ground Plane Options / move Wheel Centre* to generate the same plots. (Tip: you need to make the user displacement values have the absolute Z value of the wheel centre as before).

5 Additional Features

5.1 Overview

This chapter describes some features relative to various functionalities of Shark such as graphs, tolerances, or user's configuration.

This chapter contains the following sections:

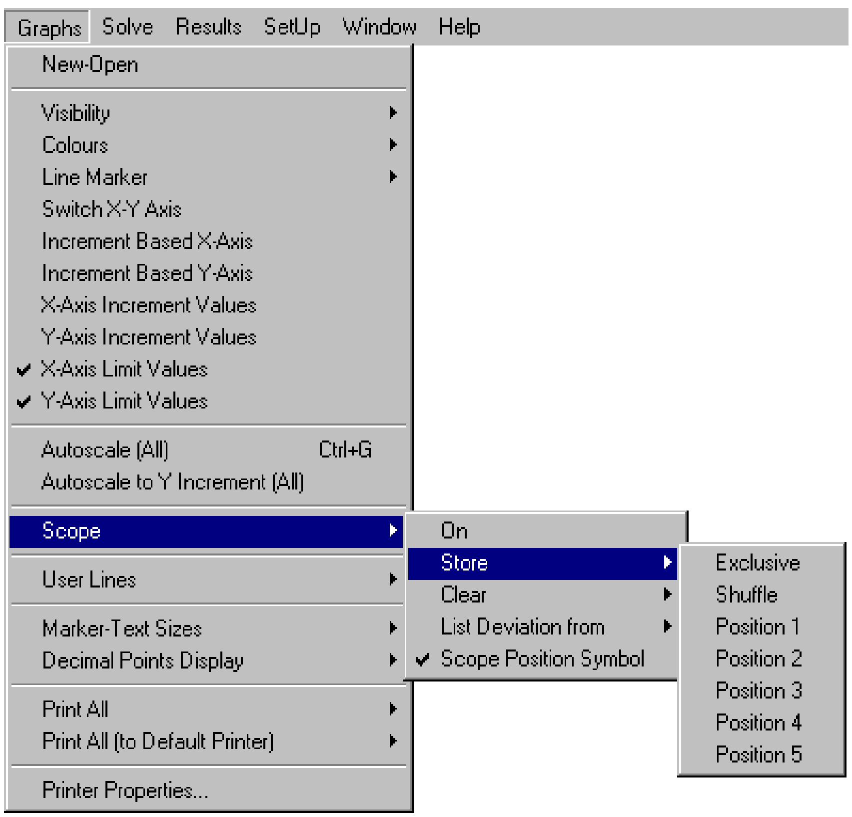
5.2 Graph’s Scopes

Three different types of lines can be displayed in the graphs:

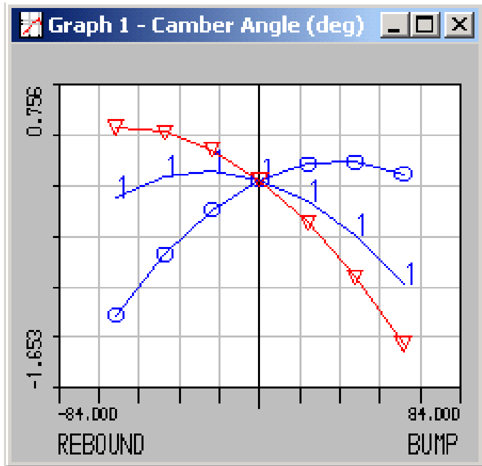
- The data line is the current hard points results (blue line with circles by default).

- The user line is an editable curve principally for visually identifying the required targets for the design (red line with triangles).It can be created by selecting *Graphs / User Lines / Copy Front-2D Data to User* for example.

- The scope lines are for saving incremental results, to enable comparison of subsequent changes to the stored plots (blue line with numbers).
It can be created by using the menu *Graphs / Scope / Store*. Scope lines are stored in positions 1 to 5. An exclusive option is available to store the current to position one and empty the others, as well as an option to grab the current line into scope position one having first shuffled any other scope lines down one position.



Scope options

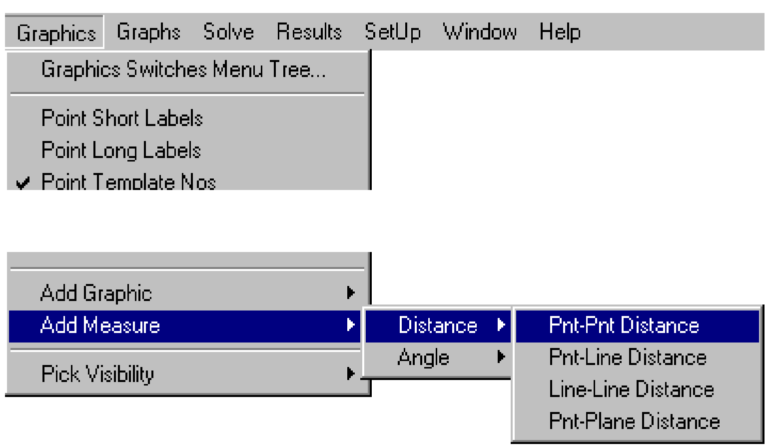


Example graph showing all three line types displayed

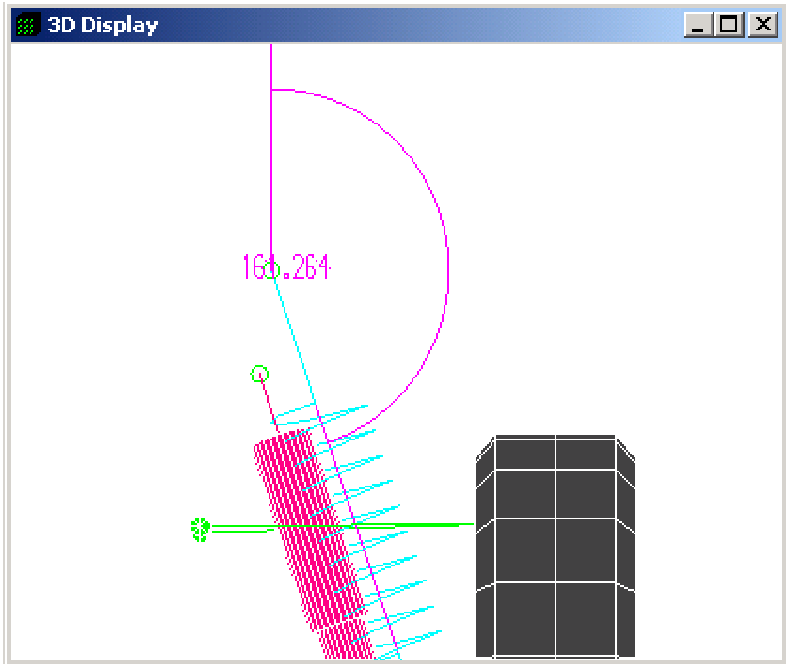
The deviation between the Data Line and the current Scope and User lines can be listed as a numerical sum. The displayed value is the cumulative sum of the difference for each calculated position. To display these values use *Graphs / Visibility Deviation Values*. The scope line used for the difference number can be changed to any of the five positions.

5.3 Graphical measure

User can manually add various measures such as angle, distance, vector cross product.



Adding a measure



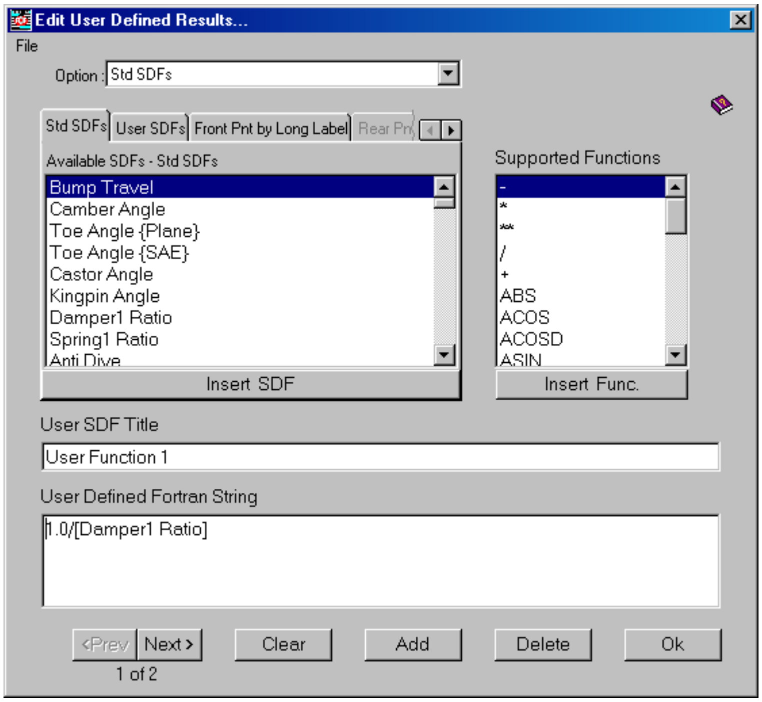
Measuring the angle between spring axis and z axis

5.4 User Defined Results

User created results can be built up using simple mathematical relationships from standard SDF's, point positions and point forces. They can also make use of other user defined SDF's but in this case, they would need to be defined in a suitable sequence (i.e. they can only reference one that has already been defined).

To open the user SDF tool, select the menu *Results / Edit User Defined Results…* , the displayed tool has two selection list areas one for variables and one for standard maths functions. The variables list has 12 sub-sections that relate to standard results, point positions, and point forces.

Each user function has a title string used for all menus and results displays, and a describing function string. The function string uses a simple Fortran style construction to build up a definition. Users can step through the defined user results via the 'prev' and 'next' icons.



User Defined Results Tool

To build up the function for a user result, either type the require function directly into the display, or, to assist in achieving the necessary formatting, use the ‘Insert Field’ and ‘Insert Func.’ Buttons. These will add the brackets and correct field syntax.

Some examples of simple functions are given below. Note the use of the ’[‘ brackets to identify variables, and the ‘(‘ brackets to set mathematical precedence.
As with all math functions, user should ensure they do not try and perform illegal calculations, such as dividing by zero, square root of a negative number or raising negative numbers to non-integer power.

a) The ratio of castor angle to kingpin angle;

    [Castor Angle]/[Kingpin Angle]

b) The distance between two points (note extensive use of ABS to ensure stability);

```
SQRT(  (ABS([frontP3X]-[frontP5X]))**2.0
     + (ABS([frontP3Y]-[frontP5Y]))**2.0
     + (ABS([frontP3Z]-[frontP5Z]))**2.0 )
```

(as a test you can compare this against a graphical plot of the distance between two points added as a graphical)

c) The ratio of the X force to the resultant force at a point.

    [Lower wishbone front pivotFX]/[Lower wishbone front pivotFR]

d) Using an earlier user function (No. 1) in a later user function

    2.0*[U1]*COSD([Camber Angle])

5.5 Exercise: Producing a User Defined SDF

In this exercise we will use the functionality of the user defined SDF tool to create our own SDF result. The example will create the equation below.

length = || P1 - P6 ||

Where;

P1 is the lower wishbone front pivot joint
P6 is the upper wishbone outer ball joint
||…|| indicates the magnitude

We will use default suspension type 1 for this exercise:

- ***File / New... Set to Front Suspension, Type 1 Double Wishbone***

Open the User SDF edit tool.

- ***Results / Edit User Defined Results.***

Create a new SDF via the 'Add' button. You can modify the title if required.

- ***Change to the 'Front Pnt by No.' tab. Select the point 1 and click on the 'Vector' button.***

This will insert the text [frontP1V] text in the display. You could have typed this in directly but use of the function buttons ensures the correct nomenclature.

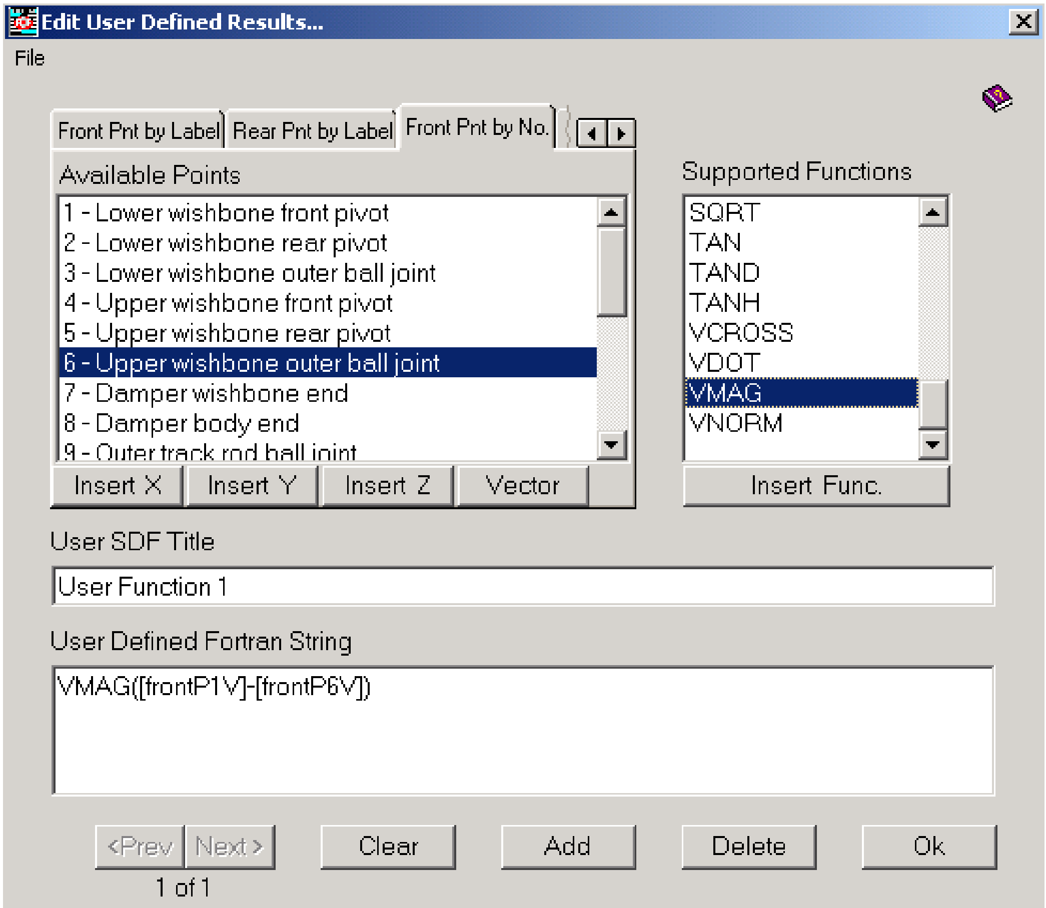
- ***After putting a minus, still in the 'Front Pnt by No.' tab, select the point 6 and click on the 'Vector' button.***

As it stands this formula will produce a vector result, i.e. it is the vector from point 6 to point 1 and has three components. Because we can only plot single values and not vectors to check the results, we will turn this into the magnitude of the vector (effectively the distance between the two points).

- ***Position your cursor at the beginning of the function string and from the 'Supported Functions' list, scroll down to find the VMAG entry. Select this with the left mouse button, and add to the function string with the 'Insert Func.' button.***

We need to make a small syntax correction because we have added the VMAG() function after the vector we need to move the trailing bracket from the VMAG function to the end of the string. Perform this editing using normal keyboard edit strokes.

Your function should now look the same as shown below.



User Function entered

Once complete, select the 'Ok' button to save and close the edit tool.
To plot this result, open a new graph, and from its right mouse menu, select Y-Variable (User SDF) / User Function 1 to display the created function.

We can now check this result by adding a graphical 'distance' element between the two points.

- ***Graphics / Add Measure / Distance / Pnt-Pnt Distance. Select point 1 and point 6 with the mouse to generate the graphical element.***

We can now open another new x-y graph, and plot this graphical element's distance as a check.

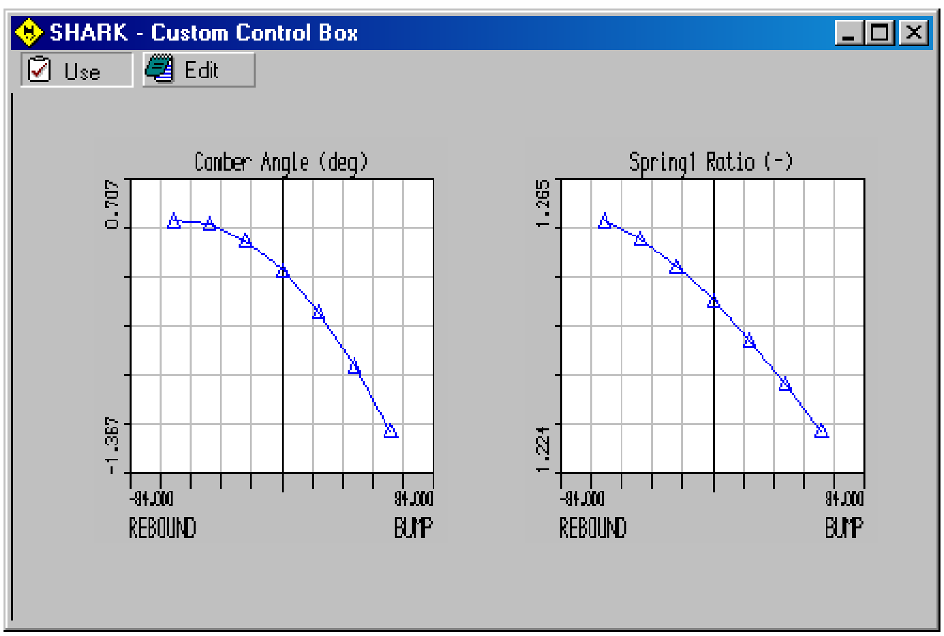
- ***Graphs / New-Open then from the new graphs right mouse menu select Y-Variable (Front Graphic) / frontG10 Pnt-Pnt Distance: Distance (mm).***

5.6 Custom Control Box

The custom control box provides a user customizable method of creating not only data entry dialogue boxes, but also opportunities to provide a formatted graphical results display.

These custom control displays are accessed through the *Window* main menu. Two menus allow you to open the existing defined custom controls or create a new one. Originally intended as a way for users to build custom data entries their use has been extended to display results graphs in the same way that you would add a button or a data entry.

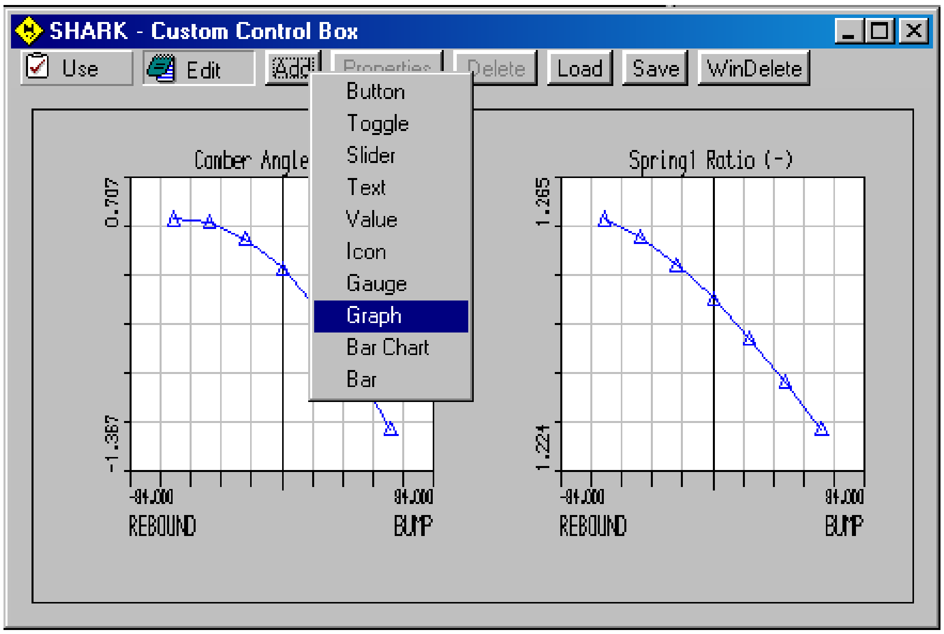
Custom controls are saved as part of the users INI file, so are reusable objects. They can also be shared directly with other users with a specific external file read/save. They could also be shared via a global common INI file used with some network installations.



Example User Control Box – Two Graphs – Use Mode

The switch between ‘use’ mode and ‘edit’ is performed by selecting the required icon button at the top left of the display. In ‘edit’ mode additional icons buttons are enabled to allow controls to be added to the display, existing controls modified or files to read/saved. The control window ‘delete’ button is also available in ‘edit’ mode and will remove a user control permanently from the display and the subsequent local INI file.

Editing is performed through pick and drag type actions, with individual control properties displayed in a simple property sheet. Thus a selected element can be resized by picking the corners, moved by picking in the middle (or with the keyboard through the Ctrl+arrow or Shift+arrow keys). Multiple picks created by selecting an area enable layout options such as align, equal height, equal size to be applied.



In ‘edit’ mode – Showing the Add control types

The controls that can be added are shown in the above figure. Remember to check the right mouse for menu options. The one specific to formatted results is limited to ‘Graph’. This graphical element behaves in the same way as the normal x-y graphs in terms of their appearance and menu options. In some ways they are tied to the main x-y graphs, as a particular SDF shares its graph limits between plot types. They thus also respond to changes in the main *Graph* pull down menu options such as Point Symbol visibility.

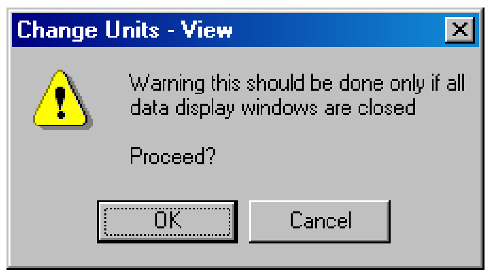
The properties of a graph ‘control’ element include its position and size together with options to defined the SDF variable (same list as for normal x-y results graphs) and local graph axis settings. The local graph axis settings are important since if they are not specified (i.e. all set to zero), the axis properties are inherited from the main display x-y results graphs, and any attempt to retain ‘standard’ graph axes through these plots would otherwise be lost.

In ‘use’ mode, these control windows can be printed or saved to a graphical file.

5.7 Set view units

Users can control the units that data and results are displayed. Relevant units are split into four groups: Angle, Length, Mass and Force. Each group has a number of alternatives to the default setting including a 'user' defined option.

To change the view display units select the menu *SetUp / Change Units…* you are warned that this should be done with all displays closed. This is because all data is stored internally in a consistent constant set of units. The changes made through this utility only affect how values are displayed, thus data values are scaled in and out of a display, using the appropriate scale factor.

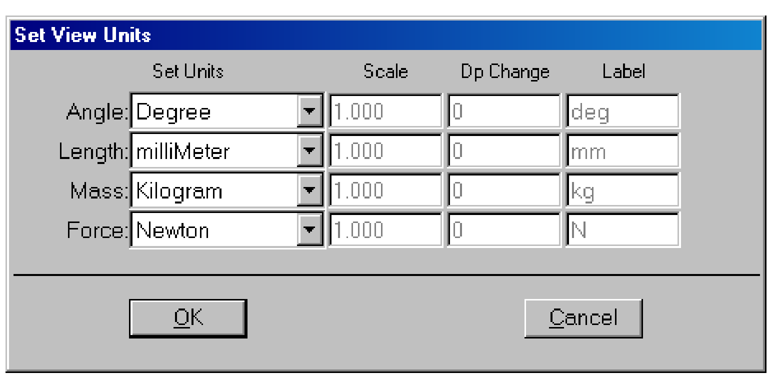


Opening the Units Display Window- Warning Message

The default setting for each units group is as follows (bracketed options shown);

Angle: Degree, (Radian, milliRadian, Minute, User-Defined)
Length: milliMeter, (Meter, User-Defined)
Mass: Kilogram, (User-Defined)
Force: Newton, (decaNewton, User-Defined)

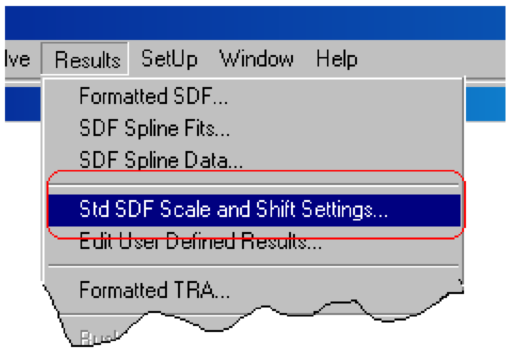
Each unit option has an associated scale factor, label, and a correction of the number of displayed decimal points. A user defined option needs to provide these three data variables. The decimal point correction ensures that a suitable number of significant figures are shown on general data entry (most graphs and results display have their own decimal point settings, so they are not affected by changes of these values).



Editing the Display Units

5.8 Standard SDF Scale and Shift Settings

All calculated SDF results have a definition; part of it is a sign convention. Since some users may have a different sign convention for a particular parameter, and also possibly a slightly different definition of the value itself, some control of these is available. The menu option *Results / Std SDF Scale and Shift Settings* opens a spread sheet that lists a Scale value and a Shift value, for each calculated result. Each corner can be set independently to give capability of both Left/Right and Front/Rear corrections.



Opening the Scale and Shift Spread Sheet

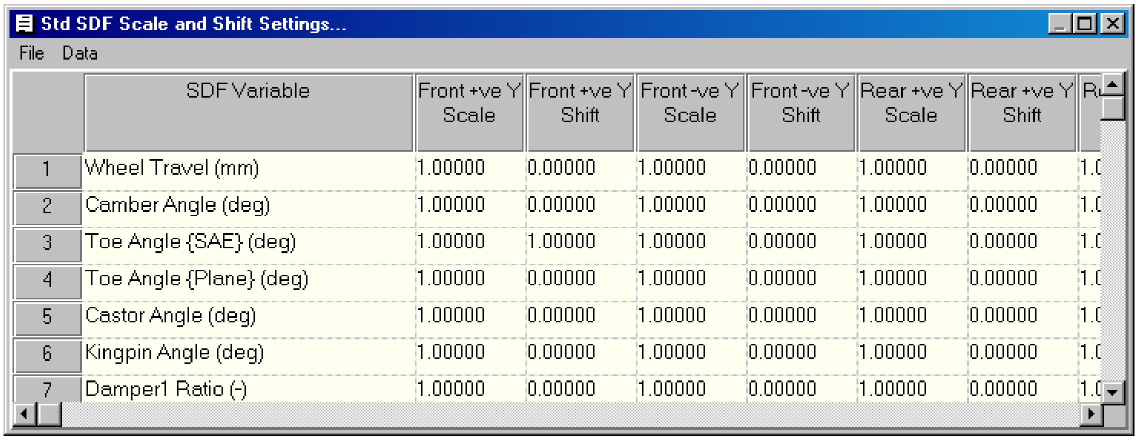
The hard coded default is for all scale values to be unity (1.0) and the shift values to be zero (0.0). This setting can be re-applied via the local menu option *Data / Set to Default 1.0/0.0*. All settings are saved to the users INI file such that changes are automatically reapplied every time the application is opened.

Some user installations have alternative settings from the 1.0/0.0 defaults that are automatically applied as part of their specific installation. These are hard coded alternatives and not part of the INI files settings. They would still be overwritten by any alternative settings in the INI file, in the normal way.

To aid transfer of these ‘scale and shift’ settings, users can save/read them to/from, a separate data file via the local *File* menu options.

The scale and shift is applied in the order shift first followed by scale, i.e.

New Value = Scale x (Old Value + Shift)


Std SDF Scale and Shift Settings...
File Data

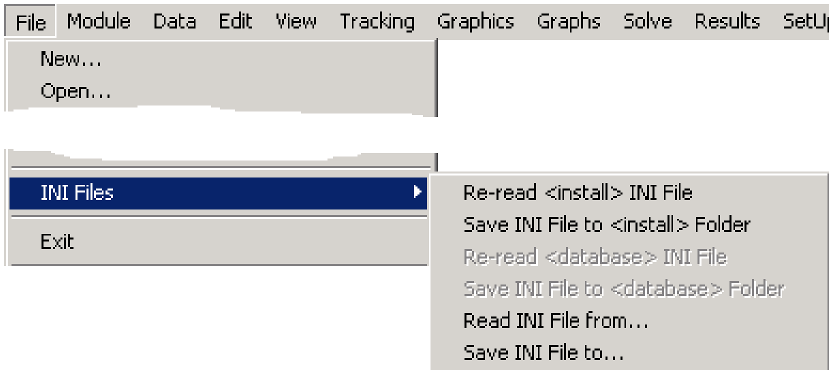
	SDF Variable	Front +ve Y Scale	Front +ve Y Shift	Front -ve Y Scale	Front -ve Y Shift	Rear +ve Y Scale	Rear +ve Y Shift	R
1	Wheel Travel (mm)	1.00000	0.00000	1.00000	0.00000	1.00000	0.00000	1.0
2	Camber Angle (deg)	1.00000	0.00000	1.00000	0.00000	1.00000	0.00000	1.0
3	Toe Angle {SAE} (deg)	1.00000	1.00000	1.00000	0.00000	1.00000	0.00000	1.0
4	Toe Angle {Plane} (deg)	1.00000	0.00000	1.00000	0.00000	1.00000	0.00000	1.0
5	Castor Angle (deg)	1.00000	0.00000	1.00000	0.00000	1.00000	0.00000	1.0
6	Kingpin Angle (deg)	1.00000	0.00000	1.00000	0.00000	1.00000	0.00000	1.0
7	Damper1 Ratio (-)	1.00000	0.00000	1.00000	0.00000	1.00000	0.00000	1.0

Editing the Standard Scale and Shift Settings

5.9 Ini files

The INI file contains all the user specific settings. This file is read in each time the program is started. It is updated/overwritten with the current settings when the program performs a normal exit. The program has hard coded defaults for all these settings, which are overwritten by the user settings when the INI file is read. Thus to revert back to the hard coded ‘factory’ defaults a user could delete the INI file prior to opening the application. For the standard installation the INI file is written to the “Windows®” folder (i.e. C:\WINNT), this means that individual users on the same machine could not have their own unique settings. Conversely it also meant that no two machines could expect to be set-up in the same way. An optional INI file is looked for in the <database> folder this is looked for and loaded if found prior to reading looking for the local windows INI file.

For the user specific server installation the INI file process has an additional INI file step on startup. This provides a method that can support both a common setting on all machines and all individual user settings on the same machine. How?



ini files options

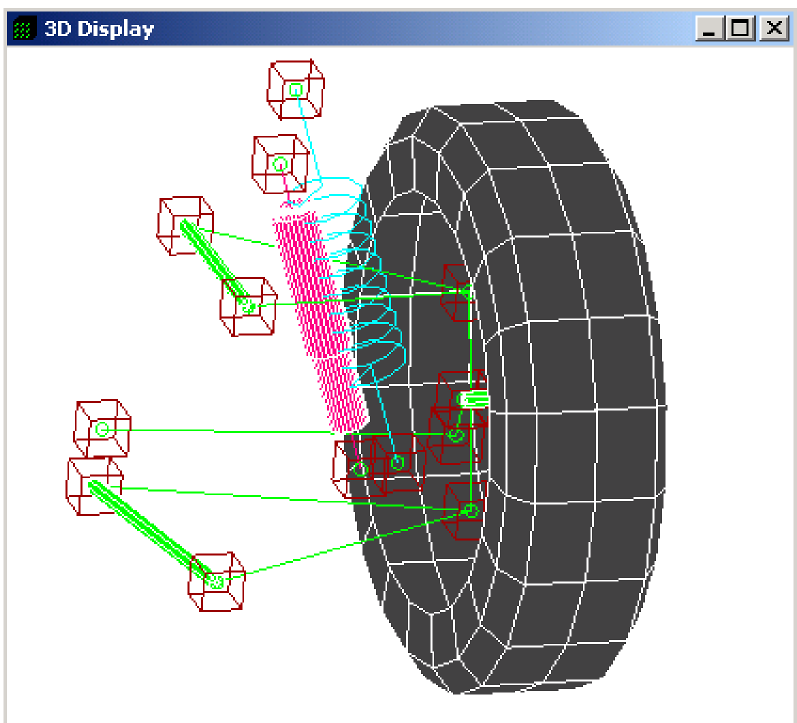
The specific server installation uses a central server for the software install, (i.e. the software is not installed on individual machines). This single location means that an INI file can be placed in this <install> folder that is read by all users. Because this install is seen as a 'fixed' file in that it is part of the initial install and then remains unchanged (primarily due to its location) a second system wide INI is required in a more flexible location. This second system wide INI file is to be identified by the <database> location. This <database> INI file can be modified by an expert user, to set common properties and settings for all users, unlike the <install> INI file which is fixed. Neither of these are written to on program close! But menu options exist to be able to write to them. These become the first two of the three INI files read. The application then looks in the specific user's directory on the "Homedrive" in the "Homepath" for the users unique INI file. This is the INI file that is overwritten when the user closes the program and thus stores their specific variation of the default server settings. An example of where "Homedrive" and "Homepath" would point to is "C:\Documents and Settings\myusername\shark.ini". Note that the standard installation uses the 'Windows' environment variable for this folder location.

Users can revert back to the system wide server default settings either by deleting their local copy of "shark.ini" prior to opening the application, or once the application is open selecting the menu option "File/Re-Read <install> INI File" or "File/Re-Read <database> INI File". Note that this three step process will still mean that a users individual settings are still machine specific, but they will start with the same server specific defaults on any other machine. The users could copy their own individual INI file on to a new machine if they wished to preserve all their settings.

5.10 Point Tolerances

Hard point 'limit boxes' can be switched on, theses boxes are set to allow only a user specified amount of travel in a specific direction. Thus when switched on, a point, (or a group point), cannot be dragged outside of its limit box. These boxes could perform one of two functions, firstly they could be set to represent packaging limitations, or secondly to indicate production tolerances. In the second case the program can run a tolerance analysis for the chosen hard point at all extremes of the limit box, the spread on the chosen derivatives is displayed on the current graphs.

The display of limit boxes have three settings, 'On', 'Off' but visible and finally 'Off' and invisible. There is no functional difference between the last two, it merely assists the clarity of the display by removing the additional graphical lines.



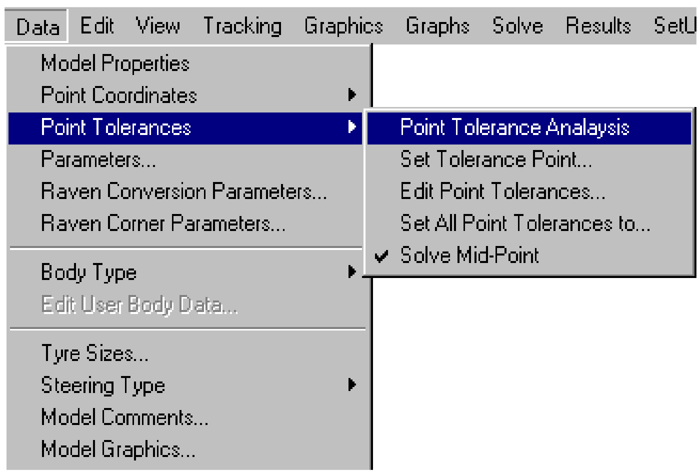
3D Graphic Display showing Limit Boxes as On

To control the status of Limit boxes use the pull down menu Graphics / Point Limits sub menu to set as Visible or to set as Use, (note that in this context use means 'On'. Un-checking Use will turn limit boxes off but remain visible, whilst un-checking Visible will set limit boxes to 'off' irrespective of the current setting).

The first use of the 'Limit Box' is as a constraint on how far a hard points position can be moved in any direction whilst joggling or dragging.

If limit boxes are in use then you cannot 'Joggle' or 'drag' a point such that it is moved outside of the limit box. Limit boxes are defined as separate +/- distances in each of the three axes, (or two for the 2D module), i.e. a total of six values for the 3D module and four for the 2D module.
If limit boxes are not in use, (visible or not), when a points position is changed by any of the edit modes, (edit, joggle or drag), the limit box is enlarged if the new position falls outside the currently defined points limits.
Because of this individual point editing, each suspension hard point has its own 'Limit Box' dimensions. These can be individually re-set using the Data / Point Tolerances / Edit Point Tolerances… menu, identify the required axle and point, and finally edit the values.

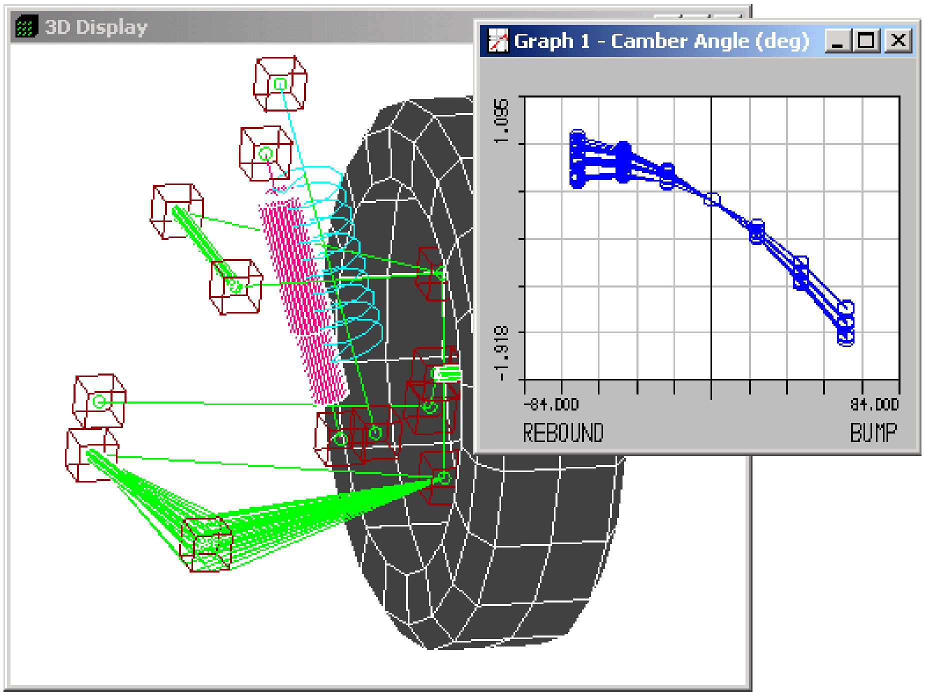
To re-set the limit boxes for all point in one step, select Data / Point Tolerances / Set All Point Tolerances To… menu and edit the required values, (note that you do not need to enter the negative directions as a –ve value, this is assumed).



Point Tolerances menu

The second use of the 'Limit Box' is as a design/manufacturing tolerance analysis tool. This is used in conjunction with the Data / Point Tolerances / Point Tolerance Analysis option to display on the graphs the spread of the current derivative over the defined limit box.

Tolerance analysis is applied to a single point at a time, the suspension being solved for its current position, each corner and each mid point of the limit box cube, (total of 27 positions for the 3D module). Before being able to run the tolerance analysis the analysis hard point needs to be identified, (select from tree style selection box). Subsequent tolerance runs will not request for the analysis hard point as by default the previously selected point will be used. To change to a different tolerance point use the Data / Point Tolerances / Set Tolerance Point… menu and identify the new point.

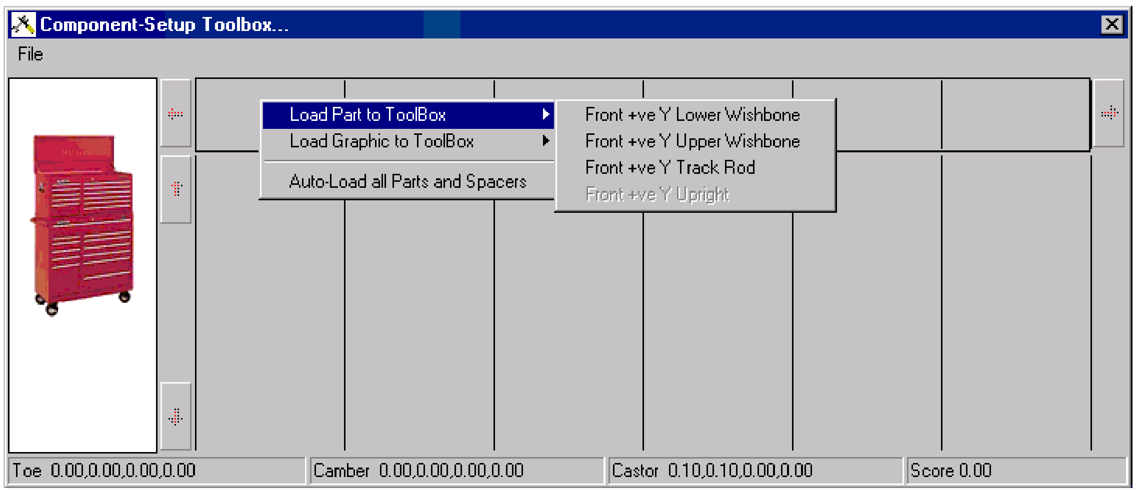


Example tolerance analysis Graphics and Graph displays

With tolerance analysis switched 'on' the model can be dynamically viewed and/or edited in exactly the same way as normally. Because of the increased number of solution loops the refresh time will be significantly increased. Once a tolerance point has been defined you can switch between tolerance on/off either via the menu Data / Point Tolerances / Point Tolerance Analysis or the equivalent toolbar icon.

5.11 Component-setup Toolbox

The Component toolbox is a utility that allows the user to create a library of “standard” alternatives for each part in the current models template(s). Each part in the toolbox has a characteristic length and the alternatives have different length properties. The parts that can appear in the toolbox are Wishbones, Tie Rods and Spacers, (currently uprights with their potential for up to six defining lengths are not included). The toolbox can thus be used to investigate the effect on suspension derivatives when mixing these alternative standard components.

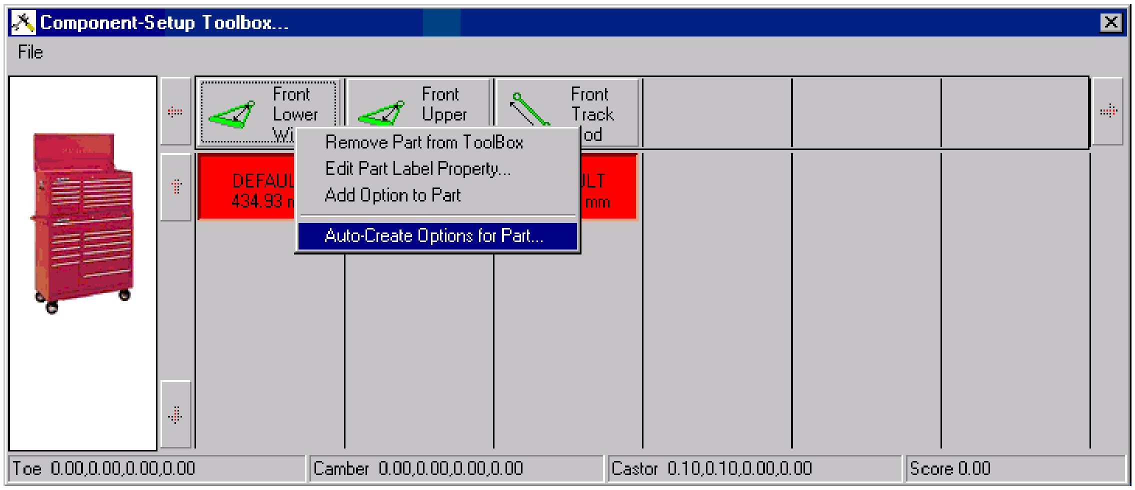


Initial Opened empty Toolbox – Select to Add Parts

To open the toolbox utility select the appropriate menu from the ‘data’ pull down menu. On initial opening it will be empty. You can add as many or as few of the current parts to the toolbox. To add parts to the toolbox select on the top horizontal panel with the left mouse. Here you can select individual parts from a list or use the ‘Auto-load all Parts and Spacers’ option to load all valid parts.

This initial add will place the part in the toolbox and add one ‘alternative’ for each added part, this alternative being the baseline as extracted from the current model. If you select on the parts top header box menu options are given to Remove the part from the toolbox (and any alternative options of it), Edit the label used for the part, add an option to the part or Auto-create a range of alternative options for this part.

Along the bottom of the toolbox can be seen the current model the values for Toe, Camber and Castor angle for each corner and the total deviation from any defined/open SDF graph user lines. These values are given to assist in later options presented that include running the optimizer to minimize the deviation or auto-setting a part length to match static angles.



Parts Added – Alternative options menu shown

Adding an option to a part places a new entry in the column beneath the default option. Each option has the same set of menu options, to modify its label, its value, to remove it from the toolbox, to make the option current or to auto-adjust its length to match the target static value for either Toe, Camber or Castor. Users can thus add as many options as required to Parts with the properties perhaps that reflect the physical alternatives available, and then mix these options to assess overall impact on all relevant suspension derivatives.

To make a particular part option current, pick it with the mouse and select the 'make option current' menu item. This option will then be shown as indented and in 'red'.

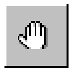
After a number of part option changes you may require to adjust a tie-rod or toe-link to reset one of the main static angles. To do this pick an option from the required part with the left mouse and select the 'Adjust Option length to re-set static Toe'. Note that at the end of the menu option is the target static value for this particular angle. Similar menus exist for camber and castor. The current static angles are displayed in the status bar at the bottom of the dialogue box. Obviously using the 'adjust option length' on a parts option will change its length property.

The component toolbox utility can be left open whilst the existing model is modified in the normal ways. This can lead to the situation where the 'baseline' part options no longer have the correct length properties. If required they can be re-aligned using the local menu option 'File / Align (All) Baseline Length Properties with Model'. This will reset the baseline option lengths as necessary for all loaded parts.

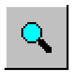
The internal optimizer can be used to sort through the available part options to identify which combination of options gives the best (lowest) score. The scoring is performed in exactly the same way as the normal optimization process, see The Internal Optimizer. To run through this loop, (in which every permutation is tried), select the local menu item 'File / Run Optimizer Scoring on Toolbox Options'.

6 Command Reference Card

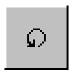
Change between dynamic **Viewing** mode and dynamic **Point Picking**. Shortcut **Alt+Z**



Change to **Translate** view. Select with left mouse button, hold down and drag.

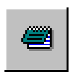


Change to **Scale** view. Select with left mouse button, hold down and move down to zoom in, up to zoom out.

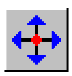


Change to **Rotate** view. Select with left mouse button, hold down and move to rotate view. Picking towards the centre rotates the eye point around the object, picking towards the edge rotates around the eye to object axis.

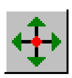
when in dynamic view mode, the **Right Mouse** button will cycle through the three dynamic view types



Point-pick mode set to **Edit**. Selecting a point with the left mouse button brings up a dialogue box to edit the points values.



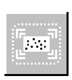
Point pick mode set to **Joggle**. Selecting a point draws the ‘joggle’ symbol on the selected point. **Ctrl+Arrow** keys joggles the points position in coarse steps, **Shift+Arrow** keys joggles the point in fine steps. Joggle is affected by current tracking direction.



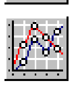
Point pick mode set to **Drag**. Select a point with the left mouse button hold down and drag along current tracking axis direction(s).

(when in dynamic point pick mode, the **Right-Mouse** button will cycle through the **Tracking** directions)

Cycles through the available tracking directions. The current **Tracking** direction(s) is indicated on screen by visible lines drawn through each hard point, along the relevant axis.



Autoscales the 3D graphical display. Shortcut **Ctrl+A**.

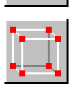


Opens a new **Graph**. The displayed variable can be changed by selecting the graph with the **Right-Mouse** button, and picking the required parameter.

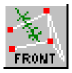
Animates the suspension over the currently selected articulation type, Bump, Roll or Steer.

Displays the Suspension Derivatives listing (**SDF**’s) in a pop-up dialogue box.

Displays the suspension **Co-ordinates** for a specified bump and steer position in a pop-up dialogue box.

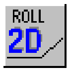


Enables the **Tolerance** analysis. In the first instance the point to run the tolerance analysis on must be defined.

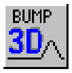


Display for editing the current front/rear suspension hard points.

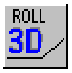
Change solver to **2D Bump** module.



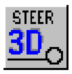
Change solver to **2D Roll** module.



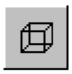
Change solver to **3D Bump** module.



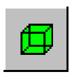
Change solver to **3D Roll** module.



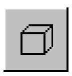
Change solver to **3D Steer** module.



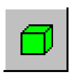
Set view mode to **Wire Frame**.



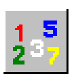
Set view mode to **Solid Fill**.



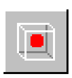
Set view mode to **Hidden Line**. This view mode only works correctly with the graphics frame type set to OpenGl, (see View/Graphics Frame Type).



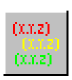
Set view mode to **Depth Buffered** solid fill. This view mode only works correctly with the graphics frame type set to OpenGl, (see View/Graphics Frame Type).



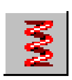
Toggles the visibility of the hard **Point Reference No's.**



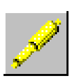
Toggles the use of the **Limit Bboxes** for the hard points. Limits can also be either visible or invisible. Turning limit boxes on will always make them visible.



Toggles the visibility of the hard point (x,y,z) **Co-ordinates.**

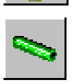


Toggles the visibility of the **Spring** enhanced graphics display. When off spring is drawn as a simple line.



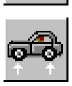
Toggles the visibility of the **Damper** enhanced graphics display. When off damper is drawn as a simple line.

Toggles the visibility of the **Wheel** enhanced graphics display. When off wheel is drawn as a simple line.



Toggles the visibility of the **Pivot Axes** enhanced graphics display. When off is drawn as simple line.

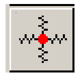
Toggles the visibility of the any **Body** graphics currently selected, (see Data / Body Type).



Displays both **Front** and **Rear** suspensions, (if in current model).

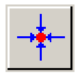
Displays **Front** suspension only.

Display **Rear** suspension only.



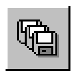
Toggle 3D compliance solver. Toggles between compliant joints with resultant forces and pure kinematic joints.

Toggles between **Display both sides** of the suspension and right-hand side only.



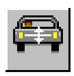
Toggle 3D Compliance Forces. Toggles the model to display compliance forces such as Spring Forces, External Forces, Compliance Forces ETC.

Open Existing File. Opens the standard Windows browser to locate and open a existing saved data file.

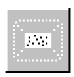


Save File As. Opens the standard Windows browser to save the current model to disk.

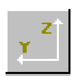
Set to Moving Ground Plane. Sets the bump displacement mode to moving ground plane.



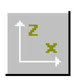
Set to Moving Body. Sets the bump displacement mode to moving body.



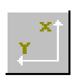
Graphics Zoom. Zooms in to the picked area of the graphics display.



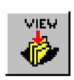
Standard Y-Z View. Sets the graphic view to the standard 'front' view. This is the Y-Z plane.



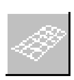
Standard X-Z View. Sets the graphic view to the standard 'side' view. This is the X-Z plane.



Standard X-Y View. Sets the graphic view to the standard 'plan' view. This is the X-Y plane.

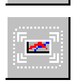


View Store. Saves the current graphical view. Give a label to identify for later retrieval.

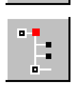


Graphics Plane Visibility. Sets the visibility of the ground plane grid graphical representation.

Copy to Clipboard. Copies the current graphical display to the Windows clipboard for paste actions.

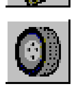


Autoscale All Graphs. Autoscales x and y axes for all open graphs.



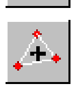
Open Property Tree. Opens the model property tree display. Alternative method for listing and editing model data.

Edit Parameter Data. Opens the parameter data edit. dialogue

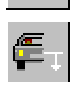


Edit Tyre Data. Opens the tyre data edit dialogue.

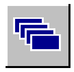
Save Point Co-ordinates. Saves the current model point co-ordinates. Give a label for later retrieval.



Create Point Group. Creates a new points group. Identify the points and associated group label.



Set Ride Height. Modify the model points by setting a change in the ride height.



Cascade All Windows. Re-arrange all open graphs and graphical display into equal size cascaded display.

7 Compliant Analysis

7.1 Overview

Up to now we have only considered the pure kinematics of the suspension system, considering it rigid. In this chapter, the suspension arms are still modelled as rigid members, but the bushes joining the suspension arms are compliant. This module is also used for force calculation.

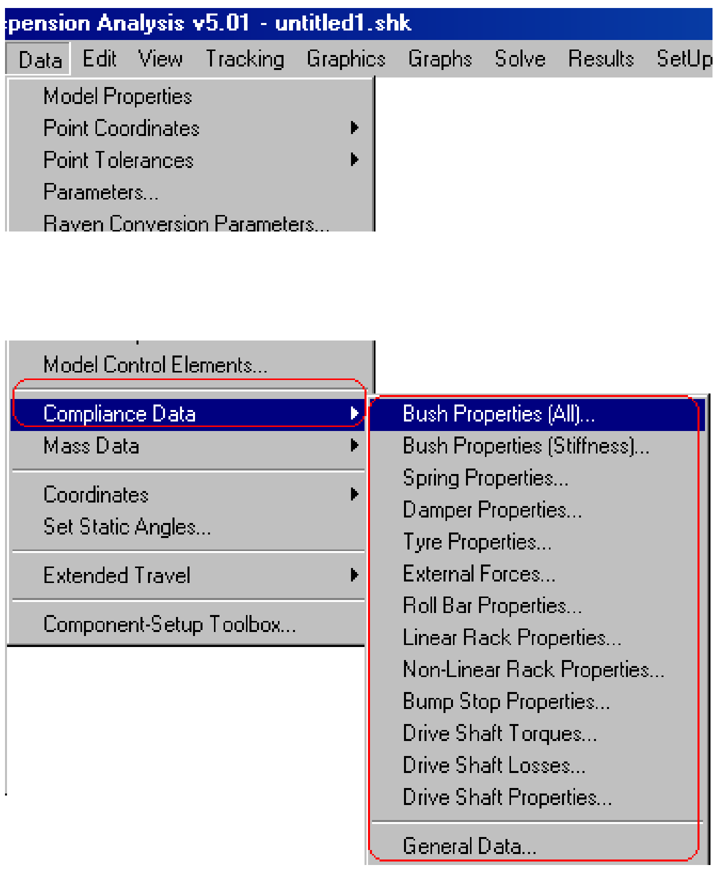
Lotus Suspension Analysis solves compliant bushes by superposition of first, kinematic solution, and second, locally linearised compliant bush solution. We use that approximation as, for any particular suspension displacement, the effect of bush compliance is assumed to give a small perturbation of the position of the suspension members. Therefore the system equations defining the bush compliance can be used in their linear form, by assuming that the change in angle between suspension members due to compliance is very small. This allows suspension compliance results to be computed quickly, and real-time updated onscreen as the suspension design is manipulated with mouse and keyboard input.

This chapter contains the following sections

7.2 Introduction

The data associated with the compliance module can be thought of as either describing how forces are applied to the model (i.e. spring properties, external forces, drive shaft loads etc), or how the model reacts those loads via its elastic members (i.e. bush properties, roll bar properties etc).

This chapter gives an overview of each of the data sections as they appear through the interface.

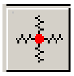


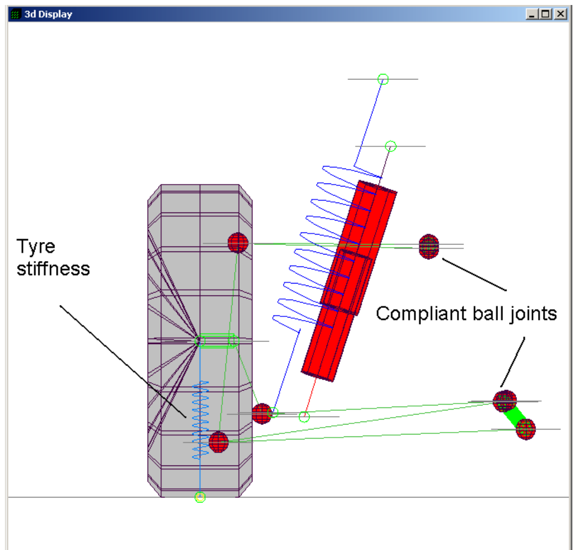
Compliance Data Menu Options

7.3 Enabling compliance calculation

For this tutorial, we will work with a front suspension model to illustrate compliant bush analysis.

➢ **Use the 'File' menu and select 'New'.** ***Create a new default 'type 1' double wishbone front suspension model.***

➢ ***Turn on compliance calculation by clicking on 'Toggle 3D Compliance Solver'*** 

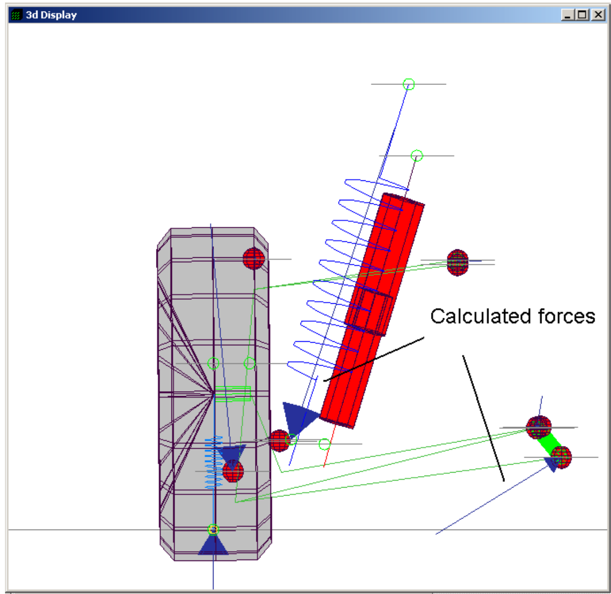


Screen shot of front suspension with compliant joints.

The ball joints are now represented by red wire frame spheres, denoting that the compliant solver module is active. You will also note that the tyre stiffness has also been included in the solver, and it is graphically represented by a spring connecting the stub axle to the contact patch. Although the suspension forces are calculated, the joints remain rigid until their individual compliance is activated. It is now possible to load external forces onto the model, and view the resulting loads on each of the points within the model. To start, we will add the force due to spring compression.

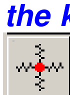
➢ ***Load the model with the spring force by selecting 'Solve / Spring Kinematic Displacement Force' and Spring Rate. Ensure all other forces are removed from the model by un-checking; 'External Forces', 'Roll Bar Kinematic Displacement Force' and 'Bump Stop Kinematic Dispalcement Force' in the 'Solve' menu.***

➢ ***If not already displayed, the force vectors can be added to the graphical display from the menu by selecting 'Graphics / Compliance visibility / Calculated Forces'.***



Calculated compliance forces due to spring force

Arrows now appear on the model representing the load vector exerted by the suspension members on each ball joint/bush in the model. To compare the results of the kinematic solution to the combined kinematic and compliant model, the deflection due to force/load can be magnified by using the '*Deformed Geometry Scale'* on the display tool.

- ***Toggling between the kinematic and compliant model using the 'Toggle 3D compliance tool'***  ***now shows a clear difference between the two models due to deflection of the tire.***

To animate the deflection due to compliance select 'View / Screen Display / Deformed Geometry' from the main menu bar and turn animation 'on'.

7.4 Bush Properties

Bush properties specify the orientation and values applied to a point modelled as a general six degree of freedom bush. By arranging the orientation and values, it can simulate special cases such as 'Ball Joint', 'Slider' or 'hinge'. This is the approach taken by the Shark compliant solver. All joints can be by default replaced by 'stiff' ball joints (three high translation stiffness values and zero for the three rotational stiffness'), this is sufficient for the compliant solver to calculate forces at points.

Strut templates are dealt slightly differently, with the upper and lower slider joints having high translational stiffness in only two directions (to allow the slider to move), and bush axes aligned along the slider axis.

The orientation of bush axis system is by two points, the local z-axis and a point in the local x-z plane. These two points can be defined in absolute terms, relative terms or by another hard point. The advantage of using another hard point is that for the example of the strut slider a local bush z-axis will follow the strut top axis point.

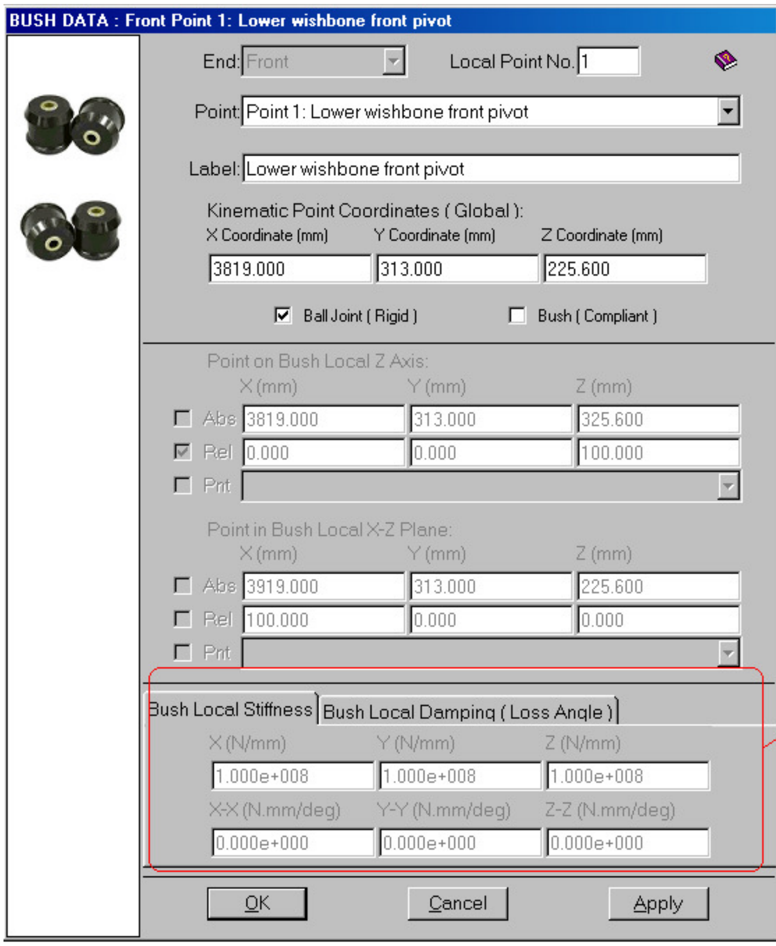
Bushes can be toggled between ‘Rigid’ and ‘Compliant’. These terms are relative since both are actually compliant, but the rigid one uses the default ‘ball joint’ rigid stiffness value for its three translational stiffness’. Default applied stiffness values are all editable.

The other values editable on the ‘Bush Stiffness’ property sheet are the local damping values. Damping values are used within the solution of the Forced-damped analysis. For bushes they are given in terms of Loss angles (deg).

Bushes that are identified as ‘compliant’ can be optionally included/excluded by individual force set. Thus should you require a bush to act as though it was a locked damper for one particular load case, it can be defined such that it is only included into the compliance stiffness matrix for the specified load case. Remember that because the compliance is applied on top of the kinematic step solution, the mechanism will still kinematically displace to the defined position irrespective of this ‘locked’ compliance bush.

- ***Set the view to the ‘x-y view’ then select ‘Set to Edit mode’***  ***and click on the front lower wishbone pivot.***

A bush data table will be brought up displaying the current ball joint/ bush setting and the stiffness of the bush in all directions.



Ball joint stiffness
Note: -

Large x, y, z values
Zero x-x, y-y, z-z values

Ball Joint and bush editing tool

➢ ***Select 'Bush (compliant)'. The greyed out boxes are now active and the bush axis are set relative to the centre of the bush.***

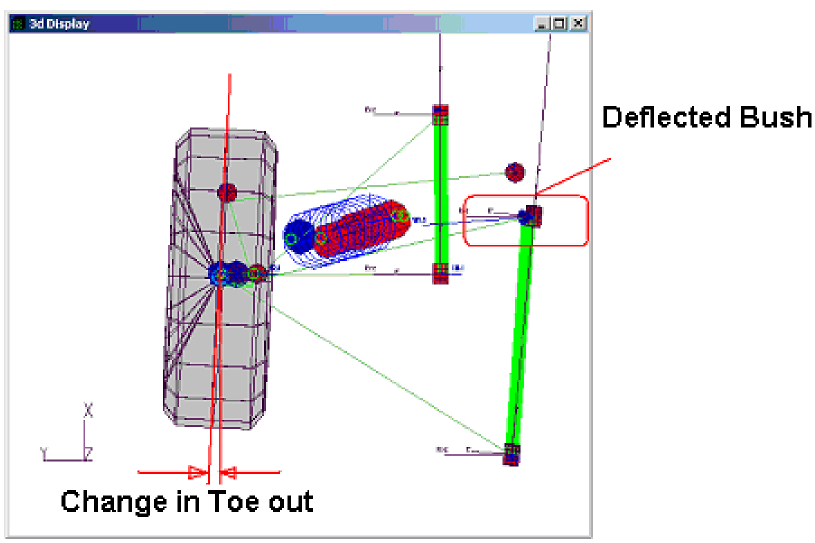
To define bush compliance, the local co-ordinate system of the bush first needs to be specified. This is achieved by specifying a point that lies on the local z axis, and a second point that lies within the local x-z plane of the local co-ordinate system. These two points and the location of the bush uniquely define a local co-ordinate system. These two points can be defined in a number of ways.

1) Absolute (Abs). Defines a point in the absolute global co-ordinate system.
2) Relative (Rel). Defines a point along the axes of the global co-ordinate system but relative to the location of the bush.
3) Point (Pnt). Uses another suspension point.

➢ ***From the 'point' drop down select 'Point 1: Lower Wishbone Front Pivot' and set the bush to 'Bush (compliant)'.***

- ***Set the 'Point on Bush Local Z-axis' to Pnt and select 'point 2: Lower Wishbone Rear Pivot'.***
- ***Set the 'Point on Bush Local x-z Plane' to relative and select x = 0, y = 100, z = 0***
- ***Accept default bush stiffness and click 'Apply'***
- ***From the 'point' drop down select 'Point 2: Lower Wishbone Rear Pivot' and set the bush to 'Bush (compliant)'.***
- ***Set the 'Point on Bush Local Z-axis' to Pnt and select 'Point 1: Lower Wishbone Front Pivot'.***
- ***Set the 'Point on Bush Local x-z Plane' to relative and select x = 0, y = 100, z = 0***
- ***Accept default bush stiffness and click 'OK'***

A slight deflection of the lower wishbone rear bush has occurred due to the effect of compliance. Also note that this slight bush movement has caused toe out of the wheel, illustrating the effect of the small deflection caused by this compliance, and the resultant effect on the suspension system as a whole.

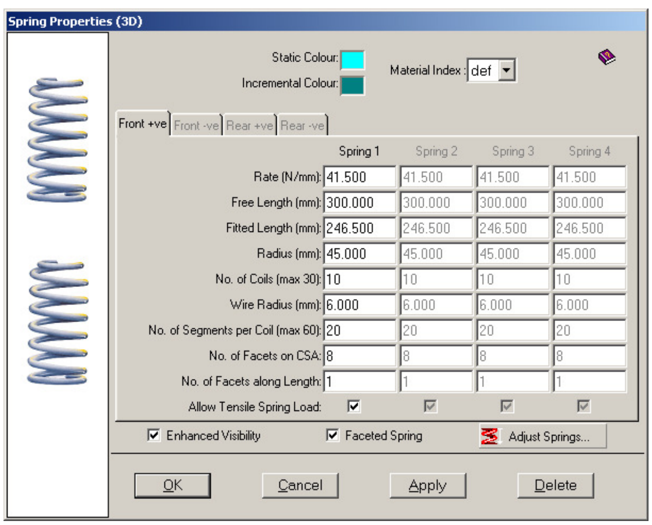


Deflection of compliant bush due to spring force

7.5 Spring Properties

Spring properties are used to identify the force applied by each spring in the model (thus it should not be thought of as a connection property like a bush stiffness). The springs are limited to linear elements, forces being defined by a linear rate and the free and fitted length. At each displacement position, the change in spring length is determined and added such that an overall incremental spring load can be calculated. This is then applied to the appropriate part and point along the defined orientation.

For a single corner template, it can have two springs on the one corner (listed in the property table as Spring 1 and Spring 2), and these may be asymmetric left to right. In the case of a full axle template, only one spring per corner is available (spring 1 on one side and spring 2 on the other). Because they are individual spring entries, they are not automatically linked such that if you are looking for a symmetric case, both need to be modified at the same time.

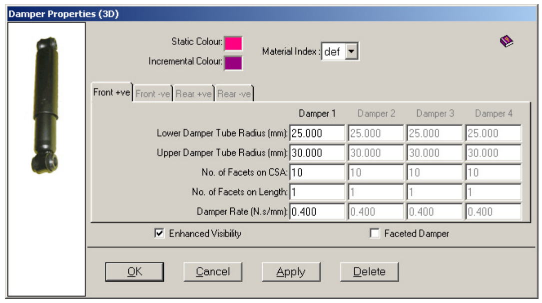


Spring Properties Data Edit

7.6 Damper Properties

Damper properties are limited to a single damping rate (in N.s/mm). They are only linked to the 'Forced-damped' results.

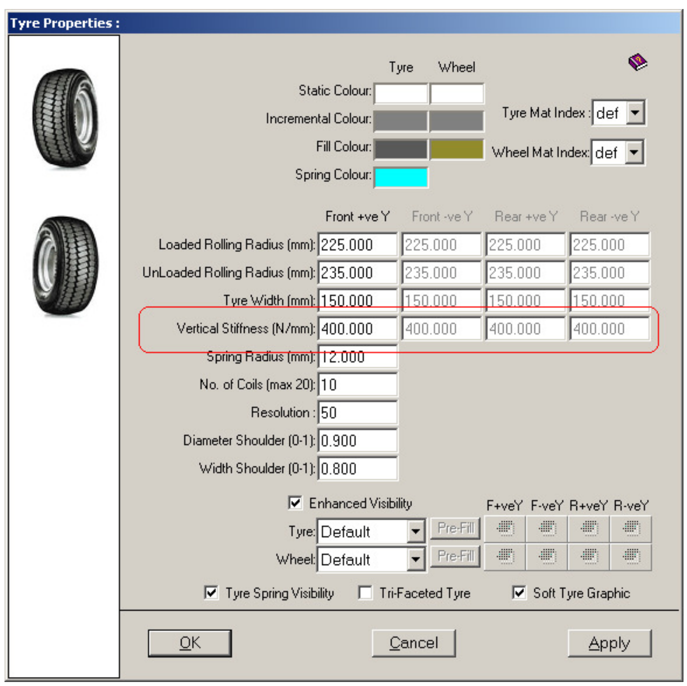
As with spring properties, for a single corner template it can have up to two dampers on the one corner (listed in the property table as Damper 1 and Damper 2), and these may be asymmetric left to right. In the case of a full axle template, only one damper per corner is available, (damper 1 on one side and damper 2 on the other). Because they are individual damper entries, they are not automatically linked such that if you are looking for a symmetric case, both need to be modified at the same time.



Damper Properties Data Edit

7.7 Tyre Properties

The tyre properties data sheet covers a number of geometry data variables. As most are self-explanatory and not related to compliance they will not be discussed here. From a compliance property perspective, only the tyre vertical stiffness (N/mm) value is of interest. This linear stiffness is used as the stiffness for the vertical connection between the upright and the ground. Both symmetry and asymmetry is supported.



Tyre Properties Data Edit – Stiffness ‘Ringed’

7.8 External Forces

This data section defines external forces that are applied to the model. Multiple forces can be arranged into sets to represent standard load cases. Forces are applied to a part, defined by magnitude and phase. The position and orientation of the force vector is controlled by the position of the head and tail. The position of these points is by absolute global position or relative to a hard point.

Force sets can be added or deleted, in the same way that forces can be added and removed from the forces sets. The concept of the 'Default' force set is used. This is the force set that is shown on the Graphical display, and the one that is used to calculate the displayed results and graphs. Some formatted results have the option to locally re-set the current/default force set; this would be part of the local formatting and thus should be obvious to the user. The 'default' set is indicated/selected by the button at the top right hand corner of the display.

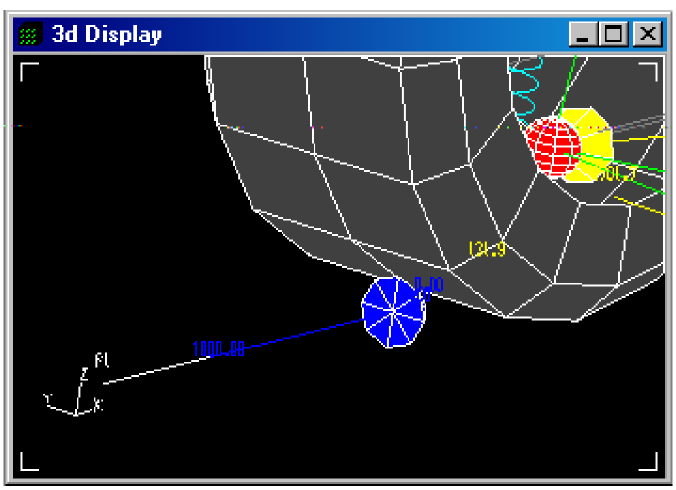
Each force set has individual solver settings for; including the spring pre-load, drive shaft loads and braked/un-braked hub switch. This is to provide full control on 'standard' load cases.

A number of load cases are hard coded into the software the settings for which can be re-applied via the local *Data* menu option.

The settings for the force sets 1-n are saved to the INI file, whilst the force set 0 is saved with the model file. This is because default force sets are considered to be comparable with 'company standard tests' whilst force set zero was considered to be a local model force set.

Each force set has an on/off switch as does each force within a set. These force set switches control their inclusion in the compliance coefficients table. The individual force switches provide control over the individual force sets, without the need to delete a force from the set.

External forces appear on the model graphics, the settings for which are controlled through the *Graphics / Compliance Visibility* sub menus.

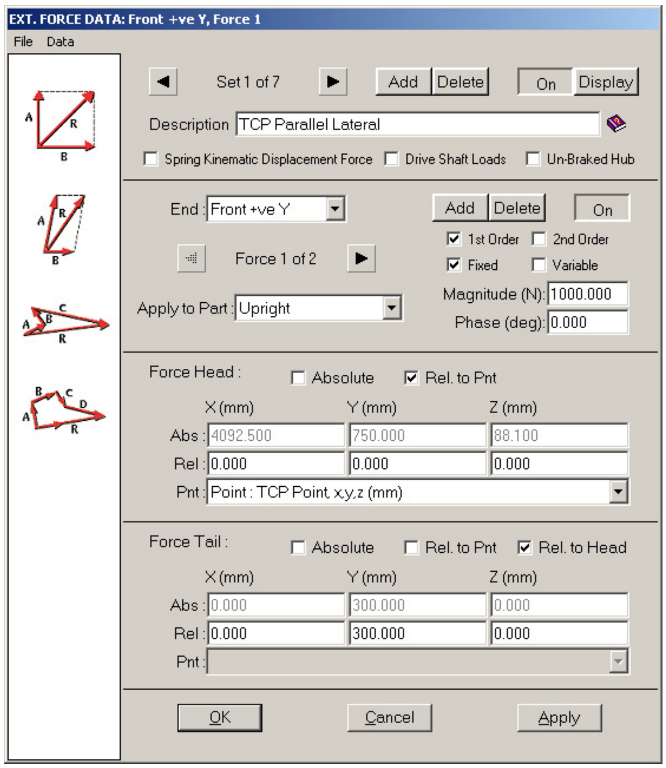


3D External Forces Display

LSA does not simultaneously apply more than one force set at a time, however more than one force set can be turned on, and LSA will then calculate results for each force set applied separately. The suspension graphic display window always displays the compliant results for the default force set that is specified by the user.

- ***From the main menu select 'Data / Compliance Data / External Forces'.***

Now both *'Spring force'* and *'External forces'* should be active.



External Force Data edit screen

Set up a 1000N inward acting lateral load applied to the upright acting at the centre of the front contact patch.

- ***Select 'Set 0', 'User Definable Default Set'***

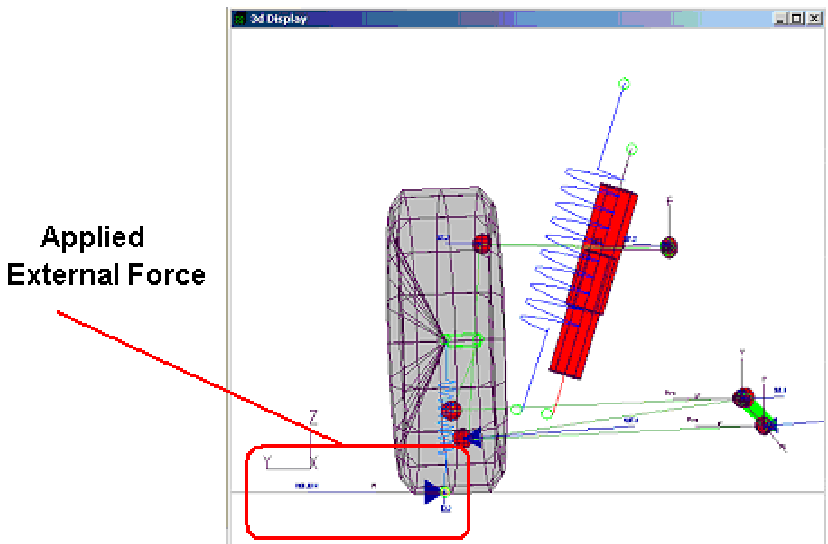
- ***Click the 'Add' button next to the force selector to add a force to the force set.***

- ***Check that front right is selected for 'end' and select 'upright' in the 'Apply to Part' drop down box.***

- ***Give the force a magnitude of 1000 N.***

The force can be turned on and off using the ‘On‘ button in the second box. Multiple forces can be added to each component in turn by adding additional forces in the same Force Set.

- ***Define the point where the force is acting by setting up the ‘Force Head’ so that it acts at the ‘Tyre Contact Patch x, y, z (mm)’ by selecting it from the ‘Pnt’ drop down box.***
- ***Define the ‘Force Tail’ relative, Rel, to the Force Head at y = 100.***
- ***Click ‘OK’ to close the external force dialog box when done.***



External force applied to upright at Contact Patch.

As with the bushes axes, each of the force vectors can be defined using two reference points. The force head can be defined as acting at a pre-defined point, selected from the drop down menu, or by giving it an absolute value expressed as a global Cartesian co-ordinate.

The force tail can also be set using several reference points. It can be defined as an absolute point defined in 3D space, as a point relative to another selected point, or can be given a co-ordinate relative to the force head.

We will now add more forces to ‘force set 0’ using table 6.1 for input data. Once each of the external forces have been included in the model, the effect of these forces both individually and combined can be calculated.

➢ ***Use table 6.1 to setup 4 forces in force 'Set 0'.***

Force	End	Apply to part	Magnitude (N)	Force Head	Force Tail (Rel.To Head)
1	Front Right	Upright	1000	Tyre Contact Patch	Y = 100
2	Front Right	Upright	1000	Tyre Contact Patch	X = -100
3	Front Left	Upright	1000	Tyre Contact Patch	Y = 100
4	Front Left	Upright	1000	Tyre Contact Patch	X = -100

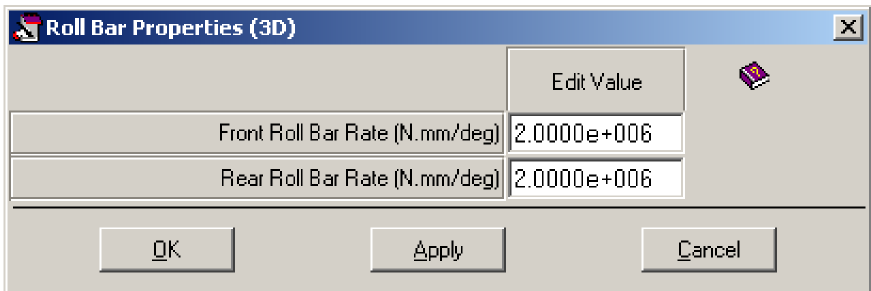
Table 7.1

➢ ***Turn both halves of the model on using the***  ***button.***

The model now reflects the systems reaction to a series of forces that would typically be exerted as the vehicle enters a right hand corner under slight braking. Similar forces acting on the system due to other conditions can be added in other force sets to represent say, braking or accelerating.

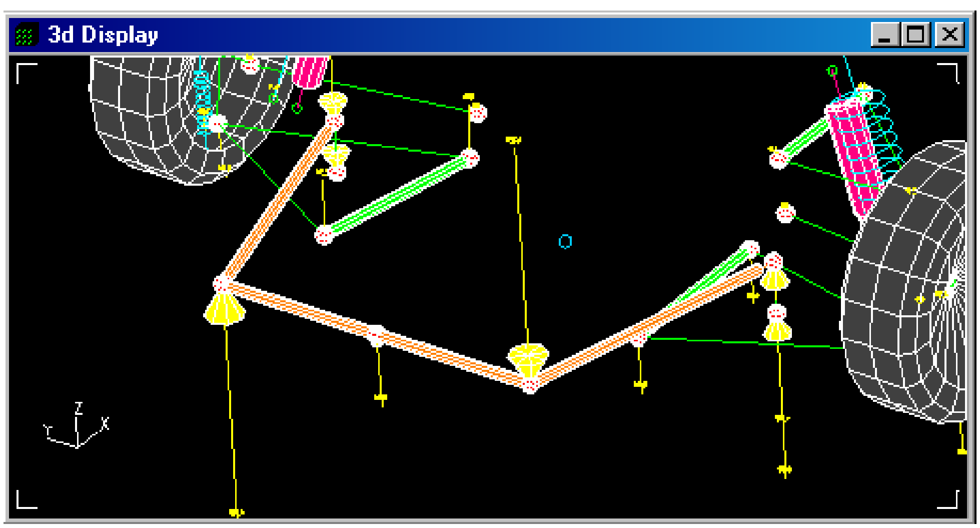
7.9 Roll Bar Properties

A template that has an anti-roll bar will use the properties defined in this section. The property is the torsional stiffness of the bush that is applied to the roll-bar revolute. It is used to represent the roll-bar stiffness.



Roll Bar Stiffness – Data Entry

A roll bar revolute joint is 'tagged' through the template as general type 34. With this tag in compliant mode, this bush will be given the roll bar stiffness as its local stiffness property. Users do not need to specifically make this bush 'compliant' (i.e. leave as rigid) for it to have the roll bar rate.

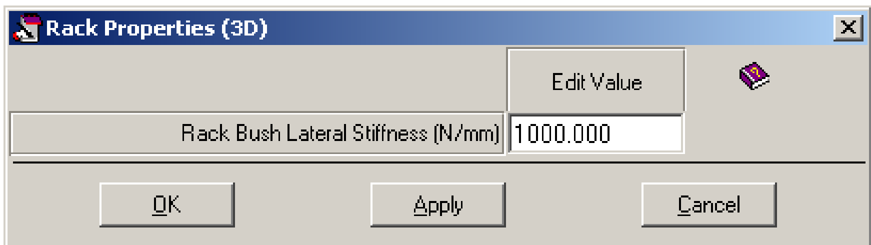


Example Roll Bar Model

If you chose to define the revolute bush properties directly, that will overwrite this default roll bar rate.

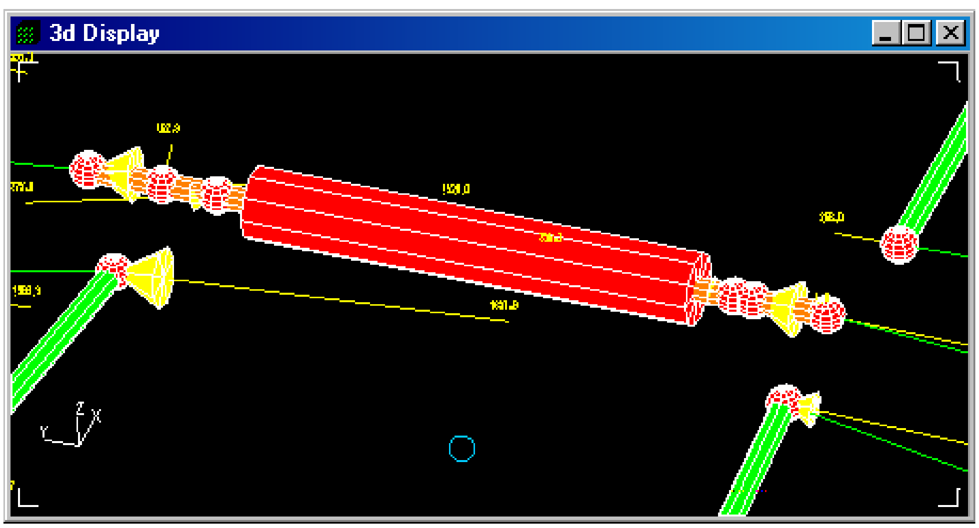
7.10 Linear Rack Properties

The linear rack properties define the axial stiffness that is applied to the specific bush in the model. It is identified by the template settings as general type 15 ‘Rack Lateral Mount Point’ (see section ‘Interactive Template Modification’ for a description of the rack two-part model).



Bush Lateral Stiffness Data Entry

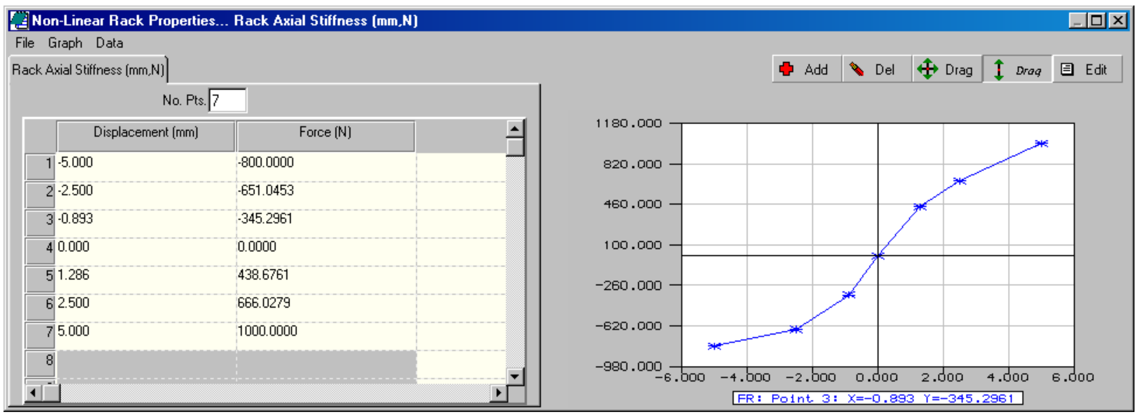
The stiffness is applied to the ‘tagged’ point without needing to define it as a ‘compliant’ point, i.e. it should be left as ‘rigid’.



Example Two Part Rack Model

7.11 Non-Linear Rack Properties

The non-linear rack properties are an optional setting for the rack properties in the previous section. The solver option '*Non-linear Rack Bush'* will enable this option for the tagged rack lateral mount bush. The properties are a look-up table of 'Force' versus 'Displacement'. This option adds an additional calculation loop to correct for the linear rack displacement, by the addition of corrective forces to bring the rack displacement in-line with the non-linear property spline.



Non-linear Rack Bush Properties

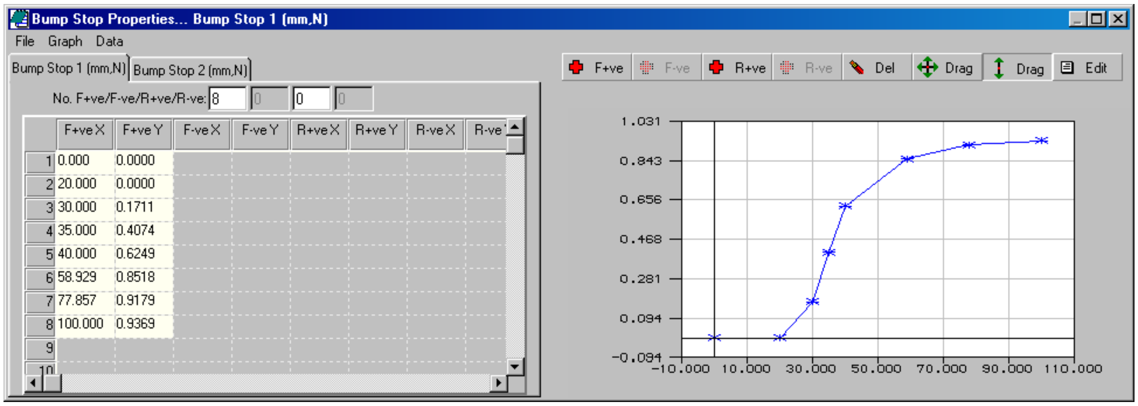
7.12 Bump Stop Properties

Bump stop properties define the magnitude of the bump stop force applied to the model, thus it is not a connection property like a bush, but can be considered to be more like an external force or spring force. Its inclusion is optionally set by two solver switches, which control the inclusion of the bump stop preload and the bump stop rate.

Bump stop locations and orientations are not defined independently. They are considered as additional forces defined by their own attachment points. They can be separate from the spring axis.

Bump stop properties are defined by splines of Force versus Displacement. The use of a spline allows for non-linear bump stop properties to be modelled, although at each increment, the local linear stiffness is determined and used within the overall stiffness matrix.

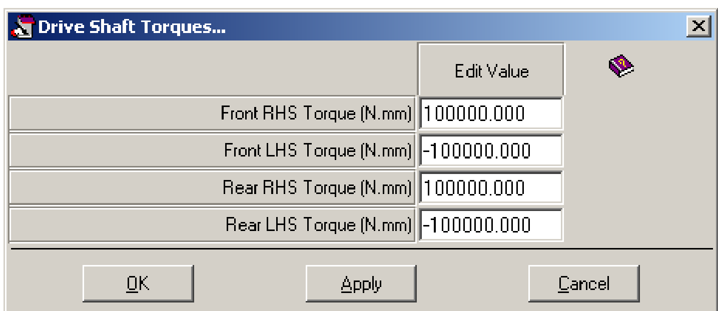
Bump stop properties can be set for both bump and rebound travel. So that this can simulate not only a bump stop but also a rebound stop.



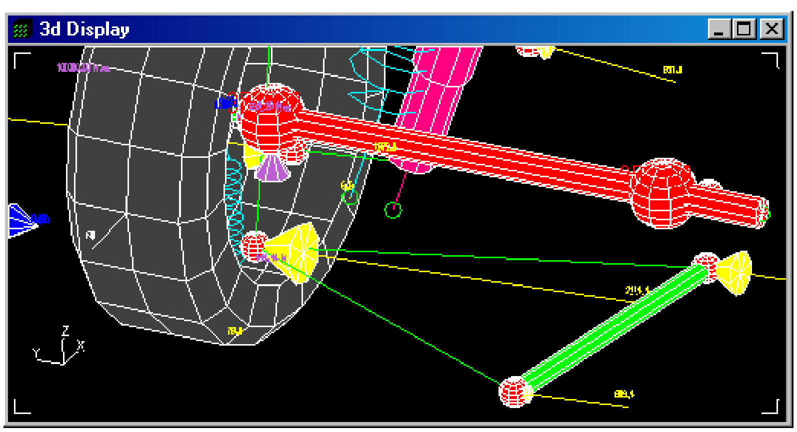
Bump Stop Properties – Non Linear Spline Example

7.13 Drive Shaft Torques

For model templates that include drive shafts, torsional loads are applied to the inboard joint. The geometry of the drive shaft is used to identify the forces and moments that are applied to the actual model upright parts (remember that drive shafts are only graphical elements, not connected parts). The data in this section defines the torque that will be applied to each corner (note use of –ve sign).



Drive Shaft Torque Properties



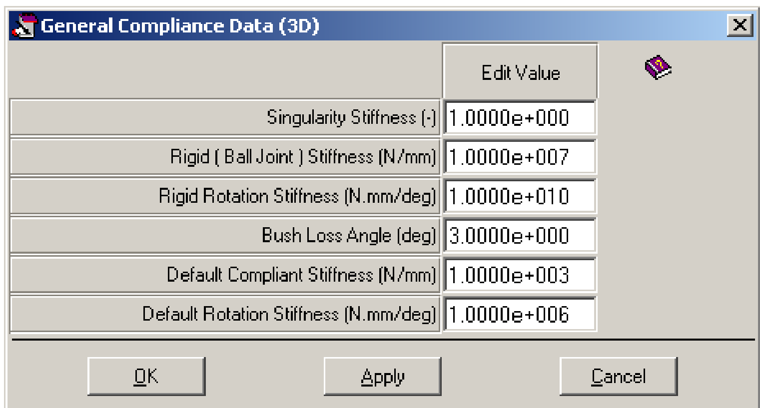
Example Drive Shaft Model

7.14 Drive Shaft Losses

This data section allows for the inclusion of non-linear drive shaft losses based on a function of the joint angle. The loss term is defined as the percentage loss for the applied torque. Separate loss curves can be optionally defined for both the inner and outer joint. Each corner also has its own pair of splines.

7.15 General Data

This data section lists a number of default stiffness numbers applied by the solver to undefined bushes and/or special ‘tagged’ joint cases. Each item is discussed below.



General Compliance Data

Singularity Stiffness: (default value 1.0 N/mm). The compliant solver uses this value to remove any singularities it identifies in the solver. An example of this is a track rod that has the default ‘rigid’ ball joints at both ends. It will have an unconstrained degree of freedom in the rotational ‘spin’ axis. The solver will apply the singularity stiffness to the stiffness matrix to remove this singularity.

Rigid (Ball Joint) Stiffness: (default value 1.0e8 N/mm). Any bush in the model that is set to ‘rigid’, i.e. does not have user defined stiffness values, is treated by the solver as a ball joint. This value is used for the three translational stiffnesses, the three rotational values are set to zero.

Rigid Rotation Stiffness: (default value 1.0e10 N.mm/deg). The rigid rotation stiffness is used by the solver for certain joint types to remove a rotational degree of freedom. These joints include lateral rack bush local z-z, Spacer all three rotations, Rigid Axle revolute, all three rotations, Roll bar revolute, local x-x and local y-y and, Hub compliance all three rotations.

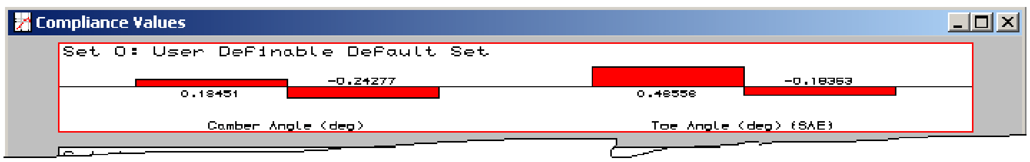
Bush Loss Angle: (default value 3.0 deg). It is the default value for bush damping. All undefined bushes are pre-filled with this value.

Default Compliant Stiffness: (default value 1.0e3 N/mm). It is the default value for all compliant bush translational stiffness'. When a bush is switched from 'rigid' to 'compliant' its translational values are pre-filled with this value.

Default Rotation Stiffness: (default value 1.0e6 N.mm/deg). It is the default value used for some specific bush types whose actual stiffness values have not been set by the user. These bush types are leaf spring hanger points local x-x and local y-y and part mesh points all three rotations.

7.16 Displaying compliance results

➢ ***Select 'Results / Compliance Bar Values' from the main menu bar.***

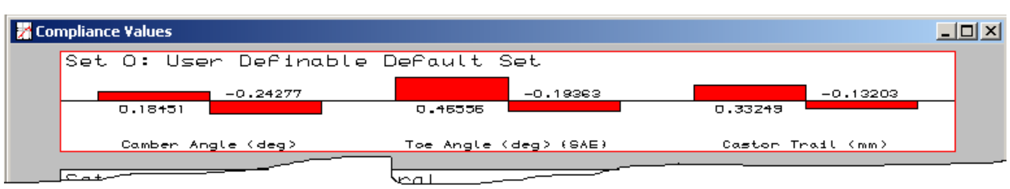


Compliance Values

The Compliance Values show the change in the displayed values due to application of the force Set. By default, the two displayed variables are Camber angle and Toe Angle, but other variables can be added by left clicking on the force set background, then right clicking, selecting *'Add Extra Variable'*. The variable displayed on any particular graph can also be changed by left clicking on the particular graph, then right clicking to select the variable from the '*Y-Variable'* menu. Each force set can be turned off as required, or all force sets can be turned on by right clicking and selecting *Turn Force Set 'Off'* and *Turn All Force Sets 'On'* respectively. Similarly, the spring force set can also be added or removed. Note that the Graphic suspension display will only show the deformed suspension geometry for the user selected default force set.

➢ ***Add the 'Castor Trail' to the Force Set 0 results by first left click on the background of the force set 0 display and right click 'Add Extra Variable'.***

➢ ***Left click on the new graph then Right click and select 'Y-Variable (SDF) / Standard / Castor Trail (mm)'.***



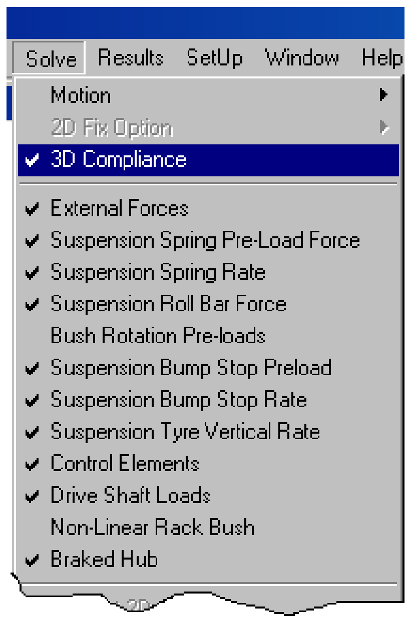
Compliance Variables relevant to External Forces

Compliance data can also be displayed numerically in the form of 'Bush Forces', 'Bush Deflections' and 'Bush Rotations' by selecting *Results / Bush Forces…* , *Results / Bush Deflections…* or *Results / Joint-Bush Rotations* from the menu bar.

The list of values display deflection and forces for all three modes i.e. Bump, Steer and Roll. In addition, they are displayed over a range for each of these three modes.

7.17 Solver Options

The solver uses a number of switch settings to define the exact nature of the solution. This section details these switches and their effect on the solution. These solver settings are saved to the INI file.



Solver Settings – Menu Items

Motion: For bump/rebound travel, the user can choose to move either the body or the ground plane. This is primarily a 'visual' display switch as the majority of the calculated result will be unaffected by this switch setting. It will however change the reported co-ordinates for points and positional SDF's. A subset of menu options is available when in 'Ground Plane'. These include the ability to define the bump motion not just by the ground plane, but also optionally switch to defining bump displacement of the wheel centre, lower outer ball joint or the upper outer ball joint.

Both the bump and steering displacements can be defined, instead of the default relative value, by an absolute position value.

3D Compliance: This controls the main switch for the additional compliant calculations. The compliance solver is required for specific results. These include point forces, bush displacements, modal frequencies, forced damped response.

External forces: This global switch turns all external forces on/off. It is a setting that sits on top of the individual force and force set settings. It doesn't change the individual force set switches just works in series with them.

Suspension Spring Pre-Load Force: Switches the spring pre-load on/off. It controls the application of the force to the model associated with the kinematic compression of the spring. This is not considered to be an external force and is thus not controlled by the external force switch. Pre-load is considered on a displacement step by step basis. Thus it is not just zeroed at the static position, it is zeroed at all kinematic positions. The suspension spring rate is still included in the stiffness matrix (see item below).

Suspension Spring Rate: Switches the spring rate on/off for inclusion in the main stiffness matrix. With this switch set to ‘off’, the solver will also exclude the spring pre-load, irrespective of the specific pre-load solver setting.

Suspension Roll-Bar Force: Switches the contribution of the roll bar on and off. This only has an effect on a compliant solution in the Roll displacement mode, and assumes that a roll bar is included in the model. It works by optionally including the anti roll bar revolute joint stiffness in the stiffness matrix.

Bush Rotation Pre-loads: It is a solver switch that optionally adds the forces due to rotational pre-loads of the bushes. This is only effective if a bush has a defined rotational stiffness (so no impact on ball joints). You don’t define a pre-load value, it is calculated from the current kinematic displacement from the static position. Thus by definition, all pre-loads would be zero at the static position.

Suspension Bump Stop Pre-Load: Switches the bump stop pre-load on/off. It controls the application of the force to the model associated with the kinematic compression of the bump stop. This is not considered to be an external force and is thus not controlled by the external force switch. Pre-load is considered on a step by step displacement basis. Thus it is not just zeroed at the static position, it is zeroed at all kinematic positions. The bump stop spring rate is still included in the stiffness matrix (see item below).

Suspension Bump Stop Rate: Switches the bump stop rate on/off for inclusion in the main stiffness matrix. With this switch set to ‘off’, the solver will also exclude the bump stop pre-load, irrespective of the specific pre-load solver setting.

Suspension Tyre Vertical Rate: Allows the user to remove the connection of the suspension upright to the ground. This connection is a vertical stiffness based on the tyre’s vertical rate. It would normally be left ‘on’.

Control Elements: Models that make use of control elements for modifying length and position via local control loops can be optionally included in the solution, such that their impact can be assessed without having to remove them from the model.

Drive Shaft Loads: Optionally includes drive shaft loads. The switch will allow a user to quickly understand the impact of the drive loads on the characteristics.

Non-Linear Rack Bush: With this option 'off', the bush tagged as the rack lateral location is taken has having a single linear stiffness. With this option enabled, the non-linear look up table is employed to have force dependant stiffness value.

Braked Hub: Specifies how the torsional load is reacted by the suspension. The default setting is 'braked', and in this form the wheel torsion load is reacted by the upright as if the brakes are on. If drive shafts are included in the model, changing this switch to un-braked will have wheel torsional loads reacted by the drive shafts. If the drive shaft is not included in the model changing this switch will have no effect on the results.

7.18 Exercise: Drive Shaft Forces

To demonstrate an element of the compliance data, we will add a drive shaft to a standard model and review the impact on a point's force.

Open a new front model using default template type 3.

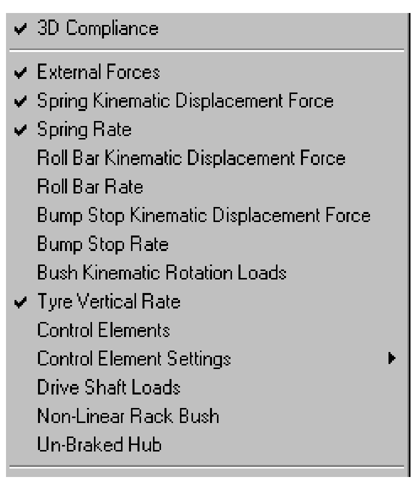
- ***File / New select front and suspension type 3 Steerable MacPherson strut. Select 'Done' to create the model.***

We will look at forces in the lower ball joint. To plot these, create a user SDF function for the resultant force at the lower ball joint and open a graph to display the result.

- ***Set to bump mode, with 60mm for both bump and rebound travel.***

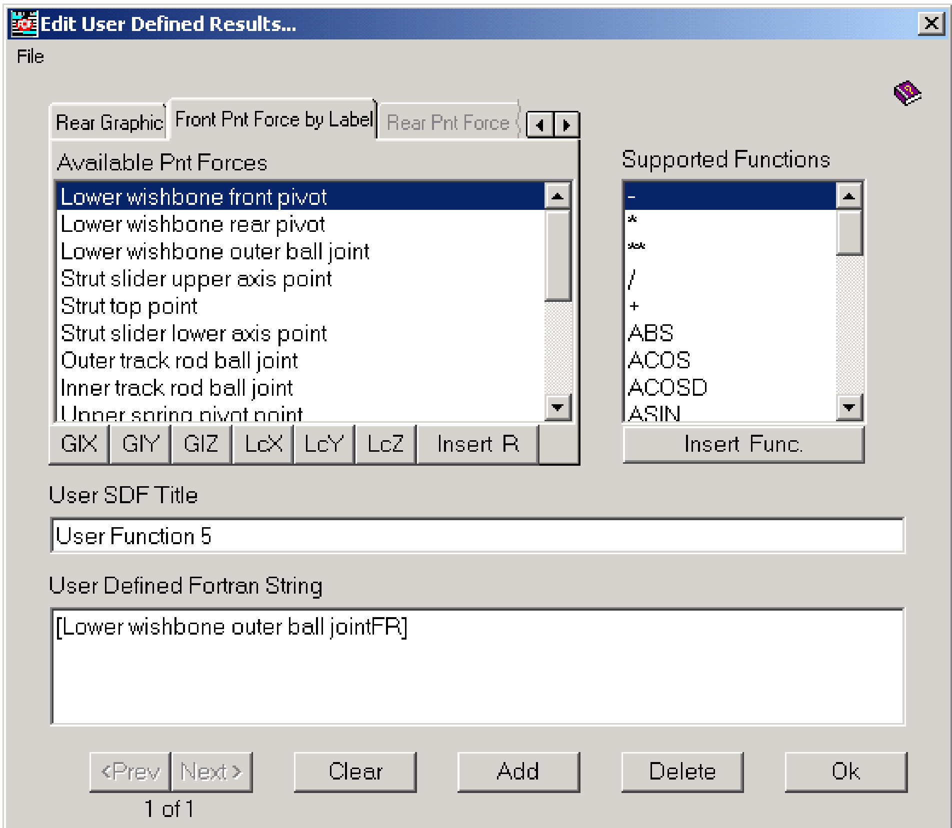
Remember you will need to switch 'compliance' on, so that Shark calculates forces. The force applied to the suspension is the spring pre-load, which is reacted by the tyre's vertical stiffness.

- ***Make sure your solver options are as following:***



Solver options

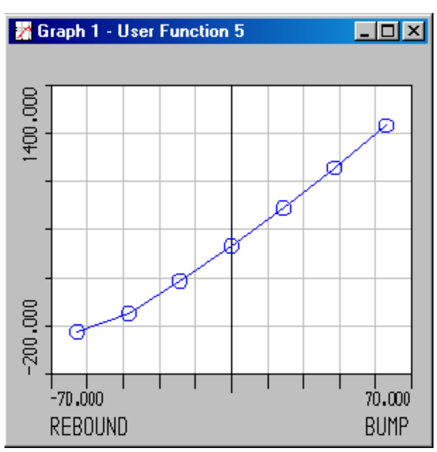
➢ ***Open user SDF tool: Results / Edit User Defined Results. Add a new function and select from the 'Front Pnt Force by Long Label' tab the 'Lower wishbone outer ball joint'. Click on 'Insert R' and then 'Ok'***



Editing the Force user function string

Now open a new graph and change the y variable to the new user function.

➢ ***Graphs / New-Open, from right mouse menu select, Y-Variable (User SDF) / User Function n. Autoscale graph as required.***



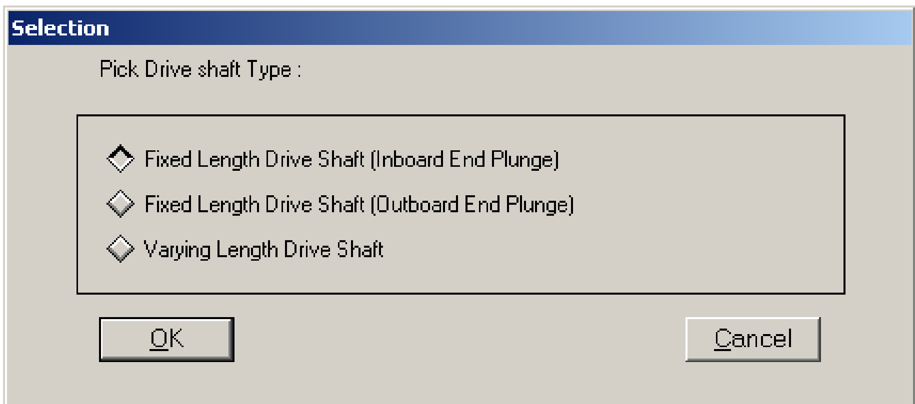
The Lower wishbone force – no drive shaft load

To provide a comparison of the forces with and without the drive shaft loads we will use the graph scope store option.

➢ ***Graphs / Scope / Store / Exclusive.***

To enable us to compare with the drive shaft we need to modify the template. Using the appropriate edit option add a drive shaft to the current models template.

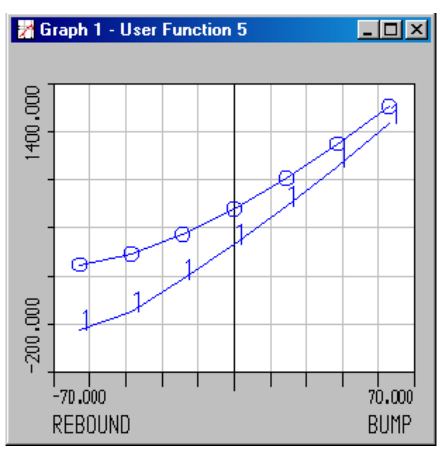
➢ ***Edit / Add to Model / Drive Shaft(s).***



Select the Type of Driveshaft

Three drive shaft types are offered, for this example use the conventional fixed length shaft with inboard end plunge and accept the changes to the template.

The displayed model should update to show the drive shaft graphics. If still in compliance mode and provide the solver option Solve / Drive Shaft Loads is enabled then the x-y graph display should also a change due to the drive shaft torque's.



The Lower wishbone force – Comparison

8 Graphical View Display Tool

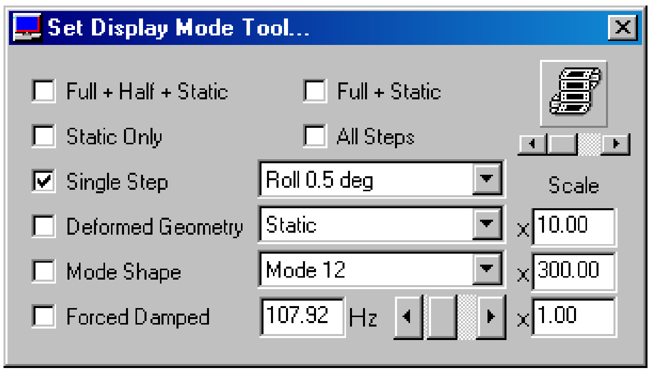
8.1 Overview

This chapter briefly covers the Graphical View display tool. This simple dialogue box consolidates into a single dialogue tool, a number of menus, and settings that affect the graphical display.

This chapter contains the following sections:

8.2 Introduction

The Graphical view display tool is a convenience dialogue box tool that combines a number of individual menus and settings of the 3D graphical view into a single display. To open the tool, select *View / Set Display Mode Tool*. This dialogue box can be left open during all normal use of the application. It does not need to be closed to have a view change applied. Some of the items presented on this display tool also have an equivalent separate menu entry in the *View* main menu.



Display Mode Dialogue Tool

8.3 Simple Modes

The top 'tick' boxes allow the user to choose between four non-animated 'step' display options. Options are:

Full + Half +Static: The display draws the suspension at the static position, the two full travel positions and two mid-travel positions.

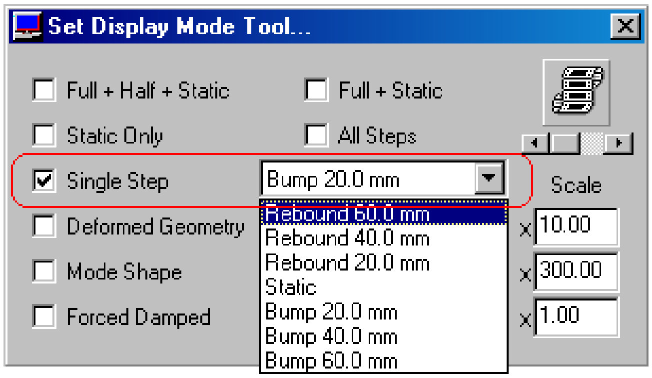
Full +Static: The display draws the suspension at the static position and the two full travel positions.

Static Only: The display draws the suspension at the static position only.

All Steps: The display draws the suspension at all calculated positions.

8.4 Single Step

The first selection box, and its associated ‘tick’ box, sets the display to a single calculated step. This is similar to the ‘static only’ option above, but it is available for all calculated steps (note that the list changes with displacement mode).



Selecting a Single Step Option – (Bump/Rebound list shown)

Setting this option and selecting from the available selections changes the graphical display, to show the selected single position. Note that animating the view will behave in the same way as when in one of the ‘simple’ modes above. The display is animated through all the calculated displacement positions.

8.5 Deformed Geometry

The deformed geometry option is only available in compliant mode. In the same way as the ‘single step’ option above, the list box gives all the calculated displacement positions (again it changes with displacement mode), and the user selects the particular position to display. The deformed geometry option applies a user defined scalar to the compliant displacements. The model is drawn as (kinematic displacements) + (compliant displacements x scalar). When animating the deformed geometry, the display shows only the selected displacement position with the scalar varying, such that the compliant displacement component is animated.

8.6 Mode Shape

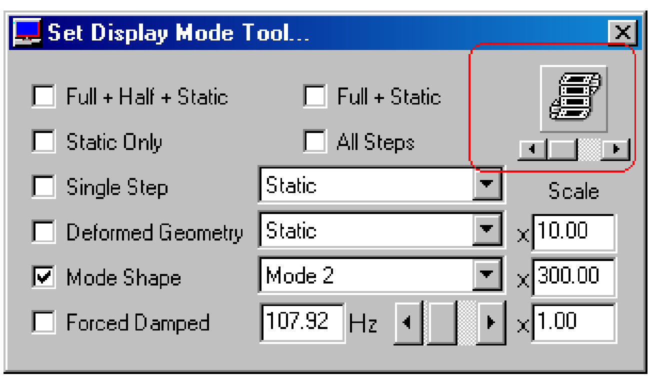
The mode shape option is only enabled in compliant mode. An individual selected mode shape is drawn with a user supplied scaling factor. The scaling factor is simply to control the magnitude of the modal displacement, such that it can be viewed/interpreted on the screen. On animation, the scalar is varied from zero to the defined value to animate the modal shape.

8.7 Forced Damped

The forced-damped option is only enabled in compliant mode. It is also required to turn on compliance forces, and to tick ‘solve/external forces’. The different external forces configurations are under ‘data/compliance data/external forces…’ (see exercise).
The Forced-damped response is drawn at the specified frequency and using the defined scalar. On animation, the scalar is varied from zero to the defined value, to animate the forced damped response at the selected frequency. The frequency setting can either be edited in directly, or set by the slider and its associated arrows.

8.8 Animation

The display tool also has a slider control for setting the animation speed. The animation can be switched ‘on’ and ‘off’ through the icon at the top right corner.



Display Tool – Animation slider and switch ‘ringed’

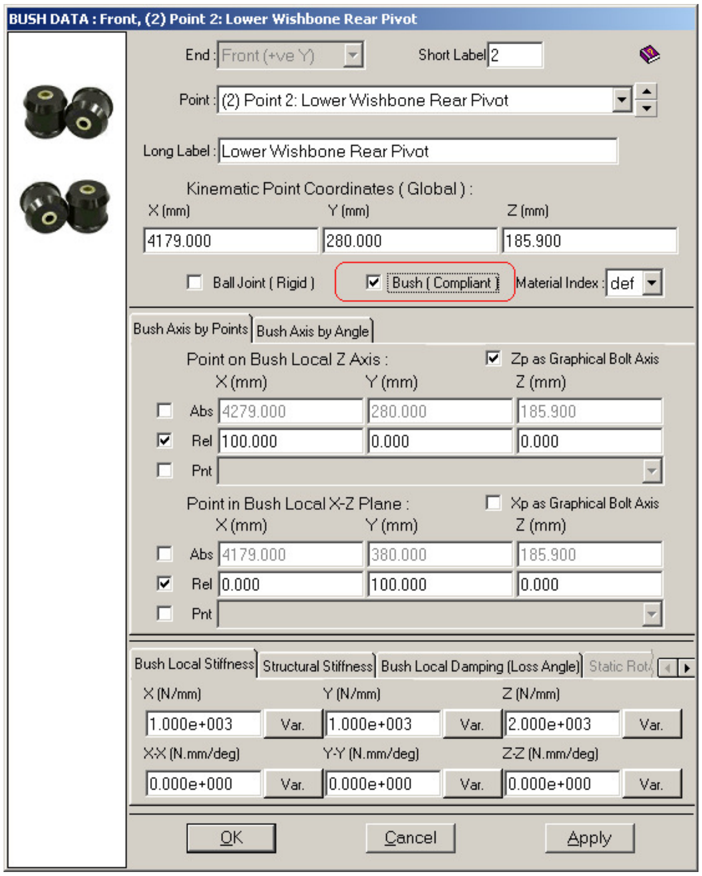
8.9 Exercise: Using the Display Mode Tool

To put into practice some of the items covered above, open a new model based on a front suspension template type 1.

➢ ***File / New select front and suspension type 1 Double wishbone, select ‘Done’ to create the model.***

Turn on compliant mode and edit the two lower wishbone points to be compliant.

➢ ***Solve / 3D Compliance. In edit mode pick in turn the two lower wishbones points and change them from ' Rigid' to ' Compliant'.***



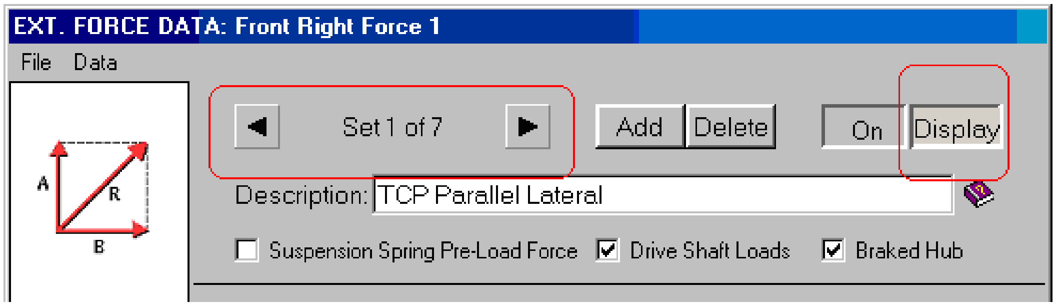
Setting the Lower wishbone points to 'bushed'

Open the display mode tool and experiment with the 'Single Step' option.

➢ ***View / Set Display Mode Tool. Change to 'Single Step' and select different options from the associated list box. What happens if you change to Roll or Steer?***

Turn on the compliance mode (if off) and add some external forces to the display. The simplest way to do this is to use the default force sets, and change set to be default.

➢ ***Data / Compliance Data / External Forces…Change to 'Set 1' and select the 'display' button to make it appear on the graphics screen.***

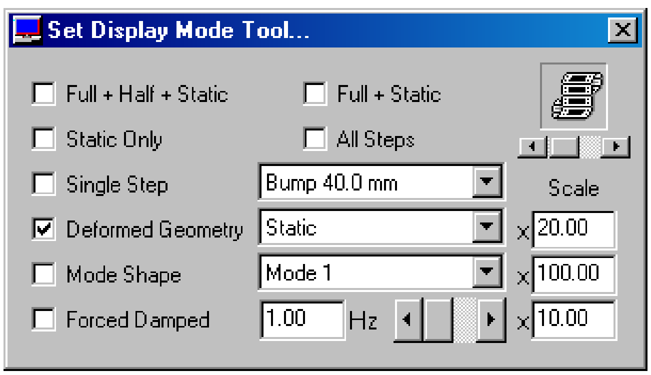


Setting Force set 1 as Default

Turn compliance mode 'on' and ensure external forces are 'on'.

- ***Solve / 3D Compliance and Solve / 3D External Forces both need to be checked.***

Now we can use the 'deformed geometry' option on the display mode tool. Select 'Deformed Geometry' and enter a scale factor of 20. If you now animate the sequence, you can view the scaled displacement of the model.



Setting Display Mode to 'Deformed Geometry'

For interest review the Mode shape and Forced Damped options.

9 User Formatting Results

9.1 Overview

A number of the result sections can be either fully or partially formatted by the user, to display a specific set of relevant results. These format settings are saved as part of the INI file. For some result sections a number of format layouts can be created and saved under different labels. These different result layouts lend themselves to defining a range of ‘standard’ results formats.

Because individual users require the ability to change the appearance of the results, the user can format a number of text and graphical results displays. In some cases, a range of format options can be created for a particular result section, being labelled in such a way as to from a set of ‘standard’ results formats that given a shared INI file could become company wide standard results formats.

The following sections cover each of the result sections that can be formatted beyond simple header switching, or decimal point display. They are all sections that a user, as a minimum, can control which values are shown, and have an element of control over the appearance as well.

This chapter contains the following sections:

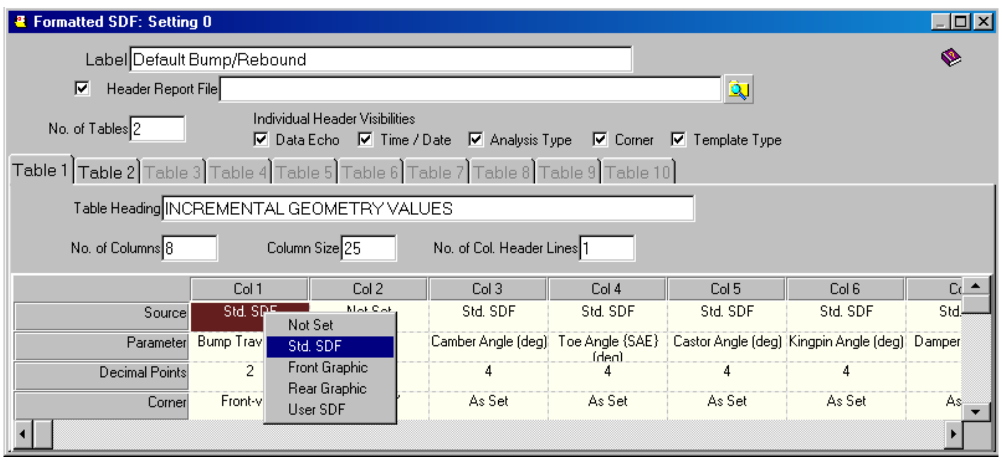
9.2 Formatted SDF

Results / Formatted SDF…

The Formatted SDF results have full format control, each format stored in a numbered and labelled slot. The first four slots 0,1,2 and 3 have a hard coded format setting, these can be overwritten by settings in the INI file, that relate to each of the four default displacement modes. The four hard-coded format settings are set up to mimic the original fixed format version outputs of each displacement mode.

Each format slot is selected via the local 'Setting' menu. This lists the available format settings. Note that by default, empty undefined slots are labelled as 'Not Defined'. A local switch, independent of the format, controls which end(s) are plotted.

To change the setting of a format set, first via the *'Setting'* menu, select the required format slot number. Now open the format editor via the local *Display / Edit Current Setting'* menu.



Format Editor – Formatted SDF Results

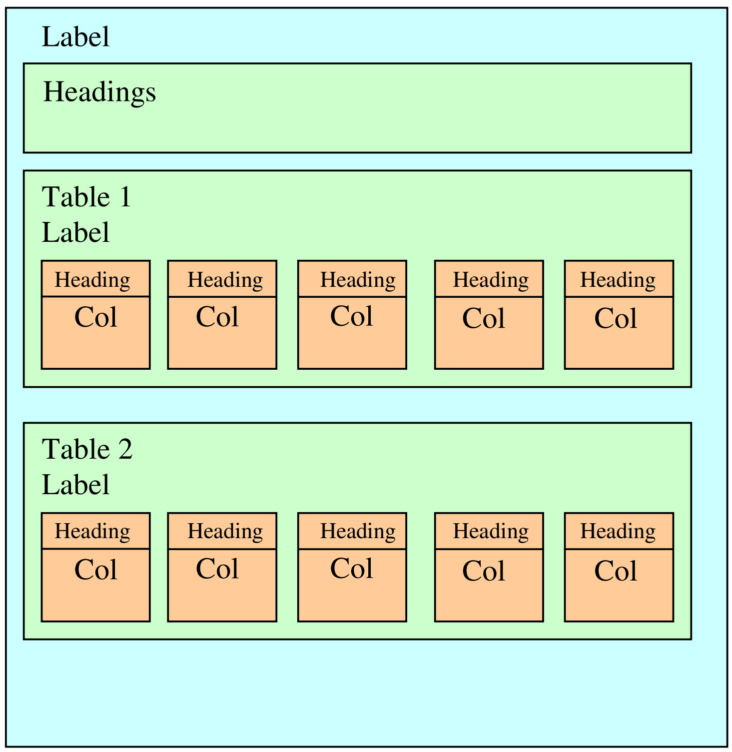
The format properties start with the label string for this format setting, (this is the description that will appear on the menu). Enter as required.

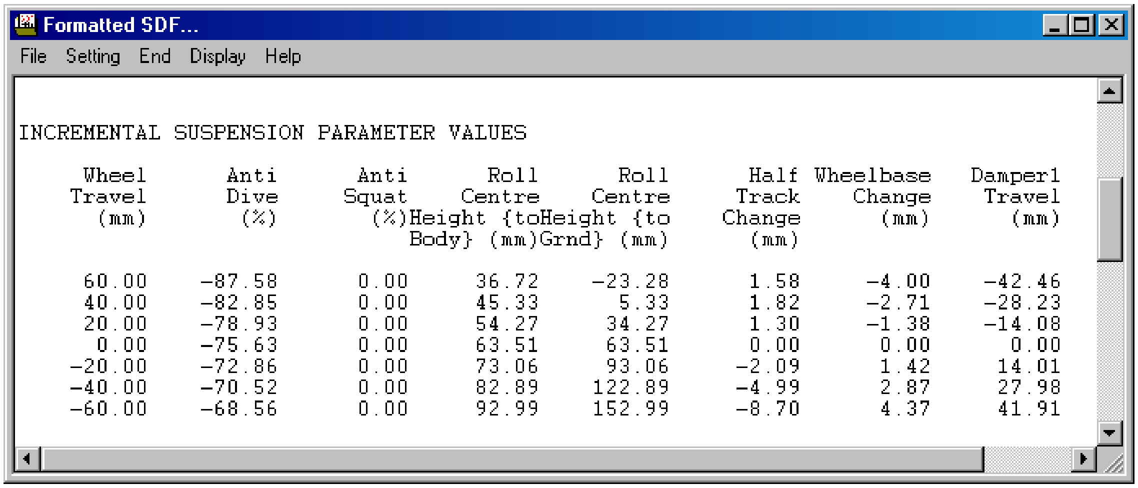
The top of each page of the results has a number of optional header lines. These include; a copy of the input hard point co-ordinates, the run time and date, the analysis displacement type, the corner number and the template type. Each of these can be optionally made visible.

The next format property setting is the number of tables. Each table is added sequentially to the results listing and has its own defined column data settings. Set the required number of tables, (maximum 10). Each table then has its own spread sheet page to define the contents of the table.

Table properties start with a heading string, (define as required), the number of columns, (each column can be thought of as a result), the column width defined as a number of characters and the number of header lines on the columns. The column width and number of header lines applies to all columns in the table.

For each column in a table it has a ‘source’ setting, ‘parameter’ from the defined ‘source’ the number of decimal points to use and the corner offset. ‘Source’ is the group that the SDF variable is in, the options are Standard SDF, Front Graphic, Rear Graphic and User Defined SDF. This setting can also be set to ‘Not Set’, this would produce an empty column. Once ‘Source’ is set the available ‘Parameter’ options list is changed to reflect the selected ‘source’. The ‘Corner’ option provides the ability to list in the same spreadsheet the result from both left and right hand sides. This is achieved by having the one of the columns being ‘-ve Y’ with the other being set to ‘+ve Y’.

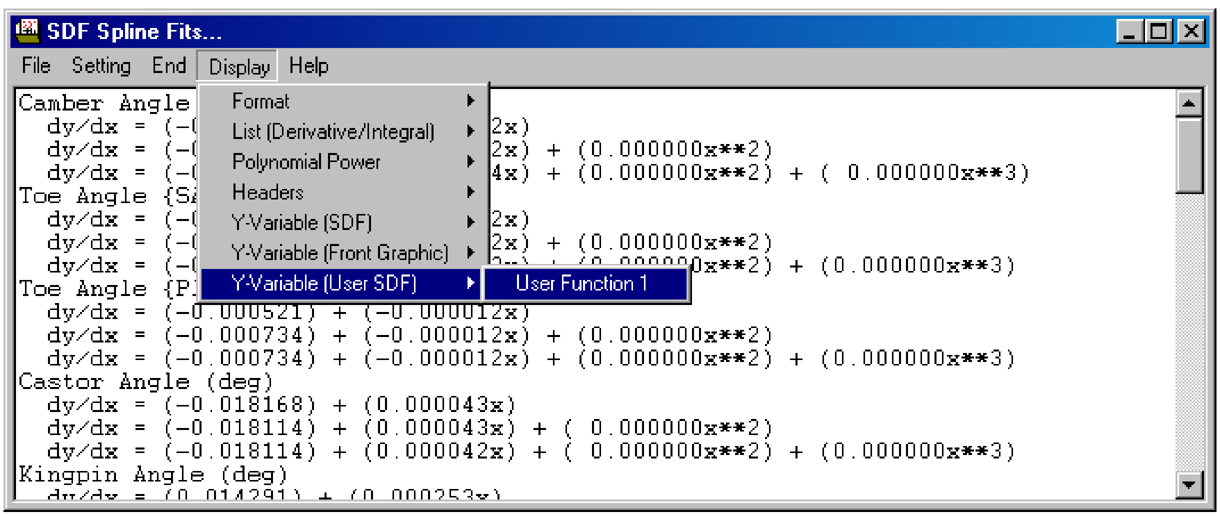




Example formatted SDF

9.3 Spline Fits

Spline fits display mathematical fits to the raw SDF data. The format functionality for Spline Fits is similar to the previous item: settings are saved with an identifying label to a slot number. The format settings consist of a number of options that are set via the *Display* menu items.



SDF Spline Fits – Format Options

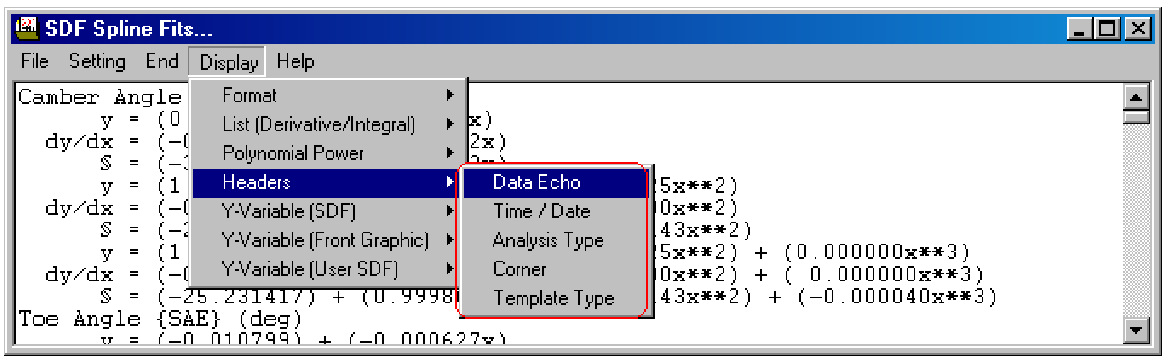
To edit/define the format select the required slot from the list. Then from the *display* menu select the required options.

Format: Sets the numerical display form of the spline fits coefficients. Select between real number display with differing Nos of decimal points, or Exponent displays with differing Nos of significant figures.

List: Sets the individual visibility for the fits to the original data curve, its derivative and the integral. Note these are three individual visibility options not an individual exclusive option.

Polynomial Power: Sets the visibility for the individual fits polynomial order. These are individual choice options, not a mutually exclusive selection, thus this option together with the previous makes visible up to 6 curve fits for each enabled variable.

Headers: Sets five individual header visibility switches, Data Echo, Time / Date, Analysis Type, Corner and Template Type.



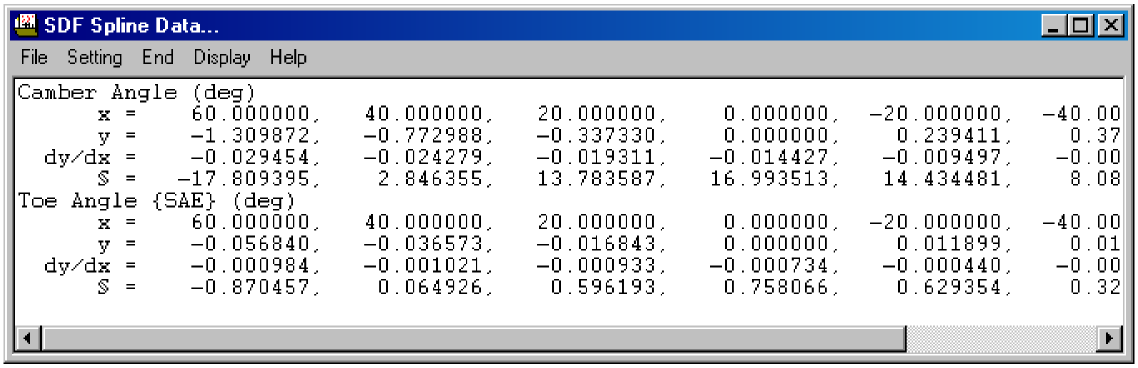
Setting the Header Visibility's

The visibility of each individual SDF is then controlled by the next three sub menus, each of which has a list of their associated variables. These three sub menus are Standard SDF, Graphics (Front and/or Rear) and User Defined SDF. The x-variable can also be defined via its set of sub menus.

All settings are saved to the users INI file.

9.4 Spline Data

This results section is very similar to the previous one, but rather than listing polynomial fits to the results it lists the results values themselves. This includes listing the derivatives and integrals. Settings are not shared between 'Spline Fits' and 'Spline Data'. Each has its own separate group of format settings.



Example SDF Spline data display

9.5 Formatted Point Forces

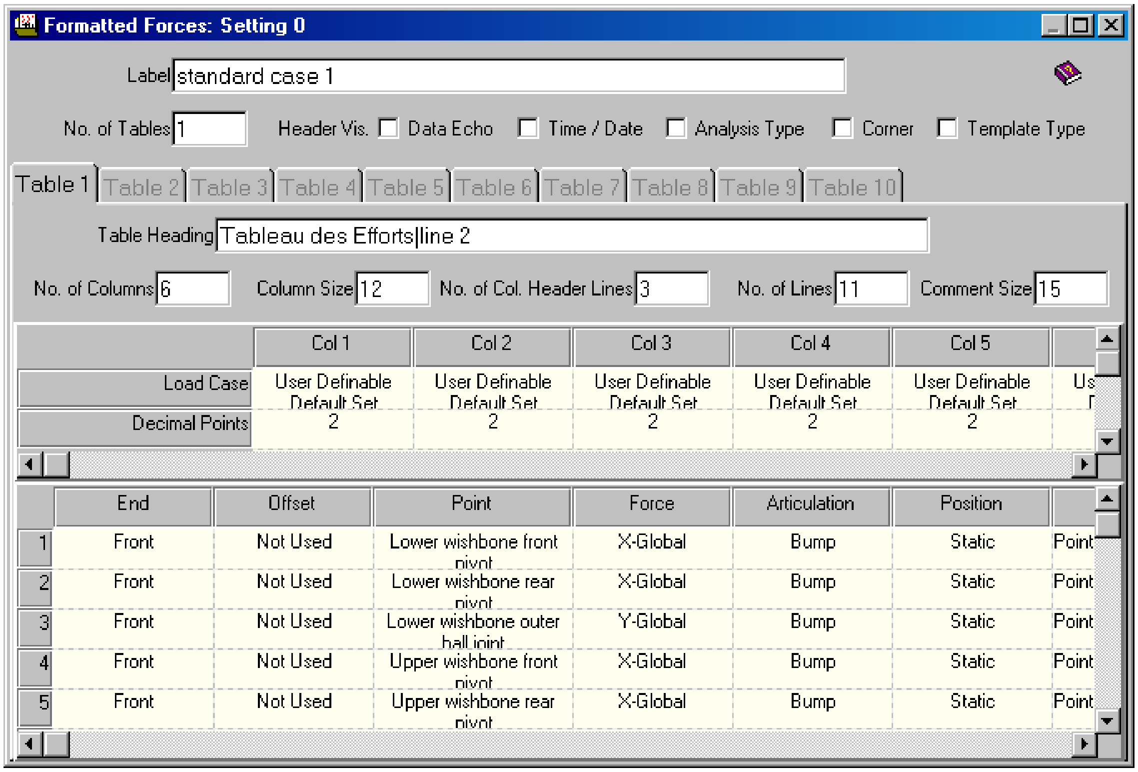
These compliance results allow a user to fully format the point forces results display. As with previous result sections, settings are given a slot number and label. This enables different standard settings to be created, saved in the INI file, and potentially used subsequently by other users through shared INI file settings.

To edit/change a particular setting select the required set from the ‘Setting’ list. Then open the *Display / Edit Current Settings* to display the formatting tool. The setting label used on the menus can be edited here. Point force results are arranged in tables. At the top of the edit tool is the definition of the number of tables, and also settings for the overall header visibilities.

Each table has a heading string and a set of unique format settings. The table overall settings include the number of Columns, the column size in terms of the number of characters, the number of lines to use for the column headers, the maximum number of lines in the columns, and the column comment character width placed at the beginning of the line.

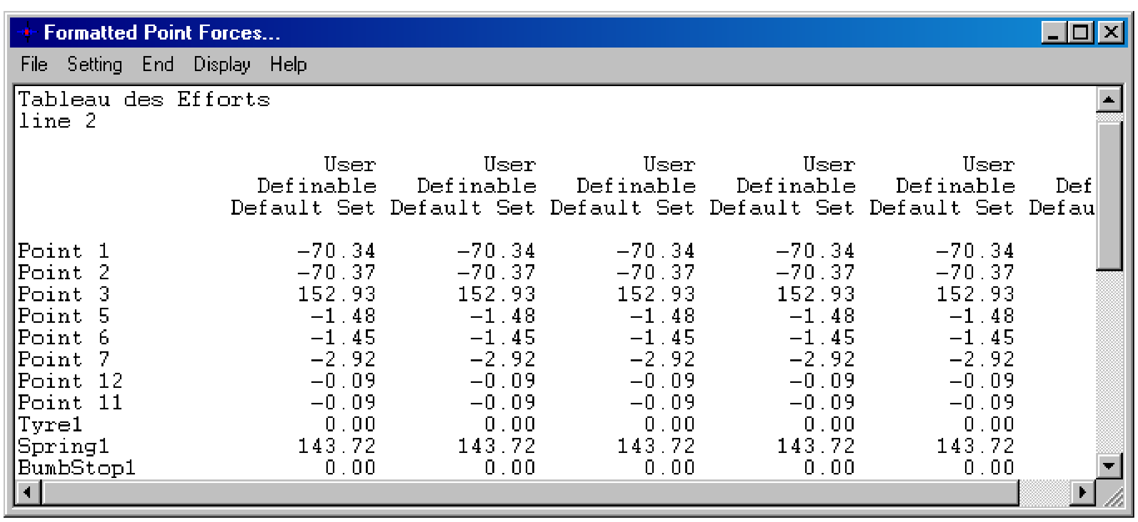
Some settings are then listed as properties of each column, whilst the lower ones are listed as properties of each line. The column properties specify which default force set to use for the column and also the number of decimal points to use for the listed numbers.

The ‘by-line’ properties list the end to use, (ignore offset reserved for future use), the hard point to use (or special force case), the force component, the model displacement type (i.e. bump, roll, steer or combined), the actual displacement position and the comment string at the beginning of the line.



Point Forces Editing Tool

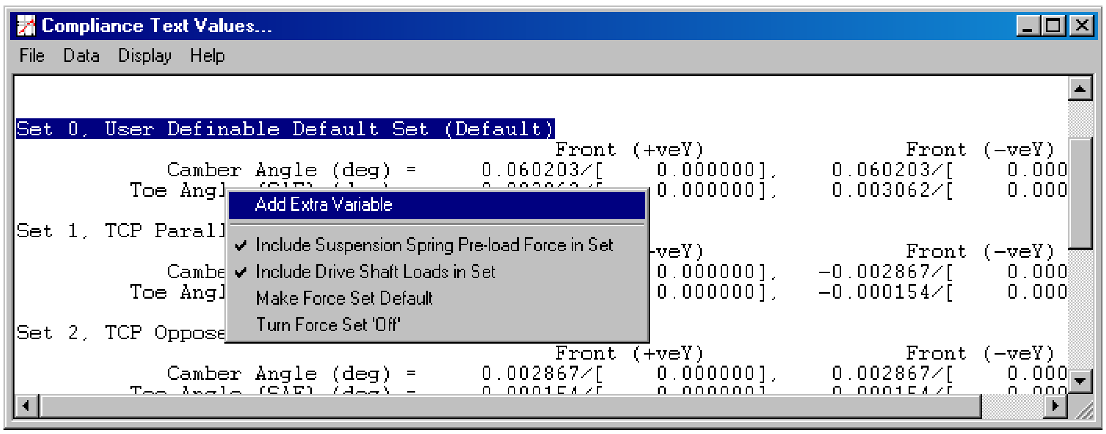
The formatted point forces display has a range of plotting options to produce hard copies or text files saved to the local discs.



Sample Formatted Results Display

9.6 Compliance Text Values

The compliance text values display is slightly different from the other result sections in that the editing of the formatting is performed by selecting on the actual display itself rather than through a separate tool. A pop up menu is displayed when you click on the display with the right mouse button. Move around the display with the left mouse button.



Example right mouse 'pop-up' menu

The compliance text values is a direct numerical equivalent of the 'Compliance Bar Values' graphical display. They share the same settings for which there is only one set. So unlike the early results sections different settings can not be saved to unique slots.

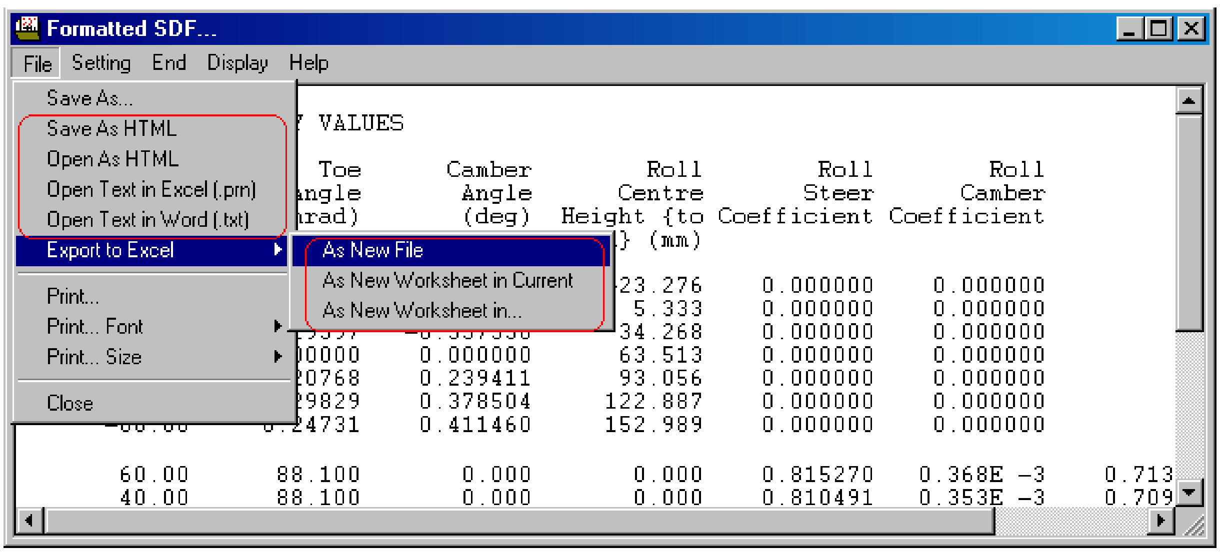
Results are listed by force set for each of the enabled variables. The value listed is the difference in the selected variable between the kinematic and compliant values (for this load case). These 'coefficients' are only listed for the static position, so are thus not affected by displacement mode.

To make a change to the result of a particular force set, move the text cursor to the area of that set, (use the left mouse button or keyboard arrows), then use the right mouse to pull up the menu options. This will include options to switch the 'default' force set (remember this is the one displayed on the 3D graphics), turn the force set 'off' (i.e. remove it from the results list) or add a new SDF variable to the listed results. The hard coded defaults have just Camber and Toe displayed for each force set.

Some local format menus are provided under the *Display* menu to set the numerical display format, control header visibility's and control the visibility of the target values. Target values can be edited through the local *Data* sub menu options.

9.7 Exporting

It is recognised that, for a software application to be popular and easy to use, it should integrate with other tools that are considered standard. This particularly applies with an application operating on a PC running a Windows® operating system needing to integrate with Microsoft's desktop products.
When LSA is first run after the installation, on start-up, the program will check for the presence of the executables. The search is performed through the registry. Additional menu items will then become visible/enabled where relevant (no changes are made to the registry).
If software products such as Mathworks Matlab are installed after LSA, the search for installed components needs to be re-run in order to initialize the link. The menu item *Set-Up / Re-run Search for Installed Components…* is used to rerun the search.



Example external link menus – Formatted SDF Results

The following summarizes the available links:

Open as Text in Word (.txt):

- File, Text Editor
- Results, Formatted SDF
- Results, SDF Spline Fits
- Results, SDF Spline data
- Results, Formatted TRA
- Results, Bush Deflections
- Results, Joint-Bush Rotations
- Results, Formatted Point Forces
- Results, List All Point Coords for User Position
- Results, List a Point Coords at All Positions
- Results, List All Point Coords at a Position
- Results, Compliance Text Values

Results, Report Batch File Display

Open as Text in Word (.rtf):
Results, Report Batch File Display

Open as HTML
Results, Formatted SDF
Results, SDF Spline Fits
Results, SDF Spline data
Results, Formatted TRA
Results, Bush Deflections
Results, Joint-Bush Rotations
Results, Formatted Point Forces
Results, List All Point Coords for User Position
Results, List a Point Coords at All Positions
Results, List All Point Coords at a Position
Results, Compliance Text Values
Results, Report Batch File Display

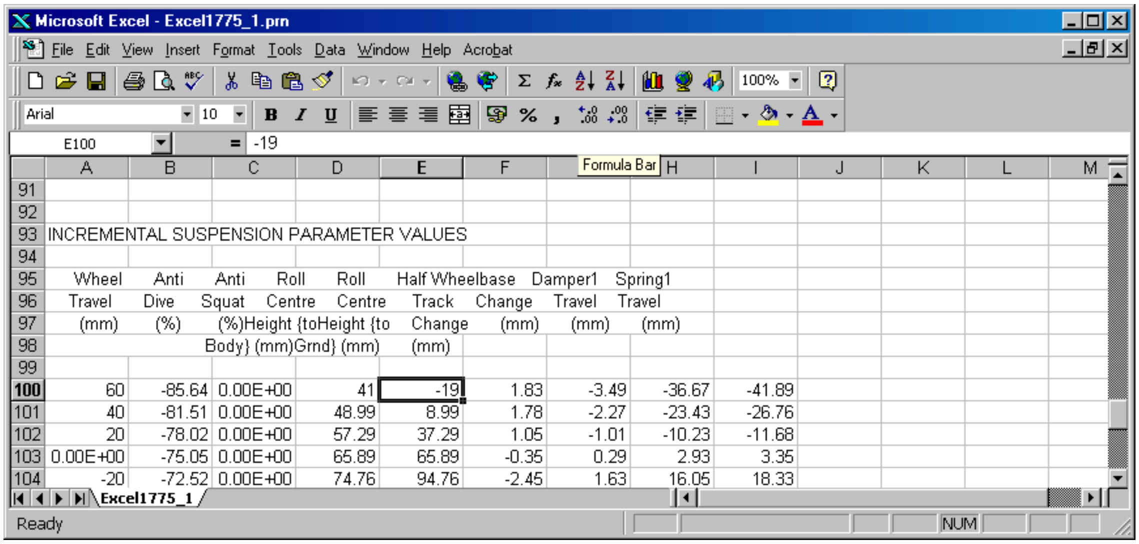
Save as HTML
Results, Formatted SDF
Results, SDF Spline Fits
Results, SDF Spline data
Results, Formatted TRA
Results, Bush Deflections
Results, Joint-Bush Rotations
Results, Formatted Point Forces
Results, List All Point Coords for User Position
Results, List a Point Coords at All Positions
Results, List All Point Coords at a Position
Results, Compliance Text Values
Results, Report Batch File Display

Open Text In Excel (.prn) (and Export to Excel Options)
File, Text Editor
Results, Formatted SDF
Results, SDF Spline Fits
Results, SDF Spline data
Results, Formatted TRA
Results, Bush Deflections
Results, Joint-Bush Rotations
Results, Formatted Point Forces
Results, List All Point Coords for User Position
Results, List a Point Coords at All Positions
Results, List All Point Coords at a Position
Results, Compliance Text Values
Results, Report Batch File Display
X-Y Graph (right mouse menu)

Open in Matlab

X-Y Graph (right mouse menu)

Below is shown an example of a formatted SDF result, opened directly in Excel.



	A	B	C	D	E	F	G	H	I
91									
92									
93	INCREMENTAL SUSPENSION PARAMETER VALUES								
94									
95	Wheel	Anti	Anti	Roll	Roll	Half Wheelbase	Damper1	Spring1	
96	Travel	Dive	Squat	Centre	Centre	Track	Change	Travel	Travel
97	(mm)	(%)	(%)Height {to	Height {to	Change	(mm)	(mm)	(mm)	
98			Body} (mm)	Grnd} (mm)	(mm)				
99									
100	60	-85.64	0.00E+00	41	-19	1.83	-3.49	-36.67	-41.89
101	40	-81.51	0.00E+00	48.99	8.99	1.78	-2.27	-23.43	-26.76
102	20	-78.02	0.00E+00	57.29	37.29	1.05	-1.01	-10.23	-11.68
103	0.00E+00	-75.05	0.00E+00	65.89	65.89	-0.35	0.29	2.93	3.35
104	-20	-72.52	0.00E+00	74.76	94.76	-2.45	1.63	16.05	18.33

Example Open as Text in Excel – Formatted SDF Results

To increase the functionality of the Excel export, a number of additional options are available. These give greater control on where the Excel data appear. These three extra options are available wherever the standard 'Open in Excel' option is given.

The first option is to export the data to a new File, whilst similar to the 'Open in Excel' it does make use of Excels' own import filter to split the data up into individual values.

The second option is to add the data as a new worksheet in the 'current' excel file. For this, there must be a currently open excel session. If more than one file is open, the 'current' file is that identified by Excel as its active session.

The third option is to add the data as a new worksheet in an existing file. This file must not already be open; it will subsequently be opened in Excel as part of this 'add' operation.

9.8 Exercise: Example Formatted SDF

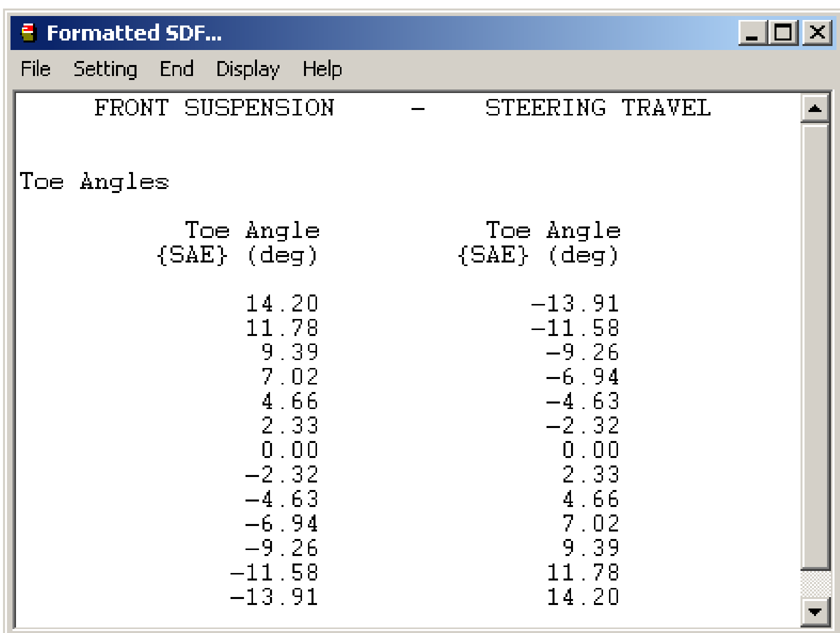
In this exercise we will produce a user formatted display for the Formatted SDF results. To produce some numbers to plot open a new front model using default template type 1.

- ***File / New, select front suspension type 1 double wishbone***
- ***Module / 3D Steer***

Open the formatted SDF results display and change to results set number 4 and then open the current settings for set 4.

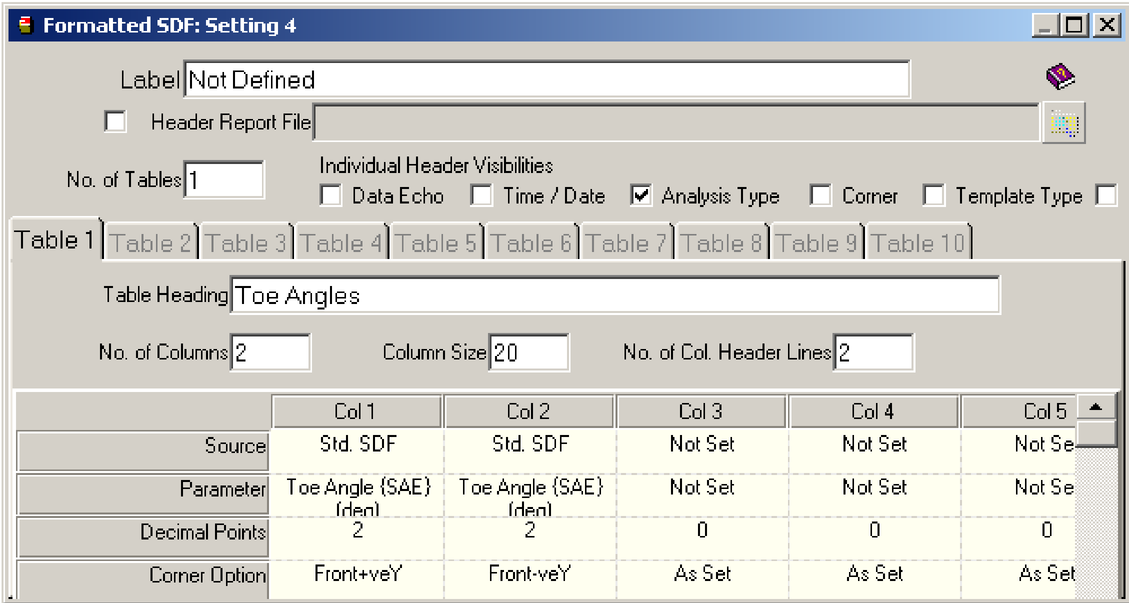
- ***Results / Formatted SDF…***
- ***Use Local menu, Setting / Set 4 Not Defined***
- ***Use Local menu, Display / Edit Current Setting…***

We will create a simple display with one table that just lists the toe angle for the two front wheels (so only the front 'End'), see example screen shot below.



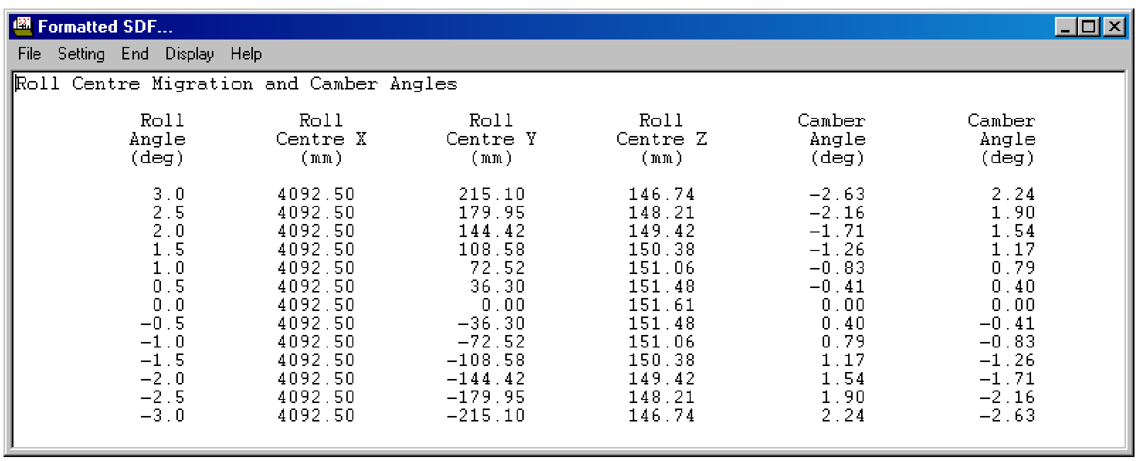
Formatted SDF example

With the edit settings display open, fill in the settings necessary to produce the required display. The solution is shown on the next page.



Format solution

Try producing the settings for the output shown below.



Roll Centre Migration and Camber Angles

Roll Angle (deg)	Roll Centre X (mm)	Roll Centre Y (mm)	Roll Centre Z (mm)	Camber Angle (deg)	Camber Angle (deg)
3.0	4092.50	215.10	146.74	-2.63	2.24
2.5	4092.50	179.95	148.21	-2.16	1.90
2.0	4092.50	144.42	149.42	-1.71	1.54
1.5	4092.50	108.58	150.38	-1.26	1.17
1.0	4092.50	72.52	151.06	-0.83	0.79
0.5	4092.50	36.30	151.48	-0.41	0.40
0.0	4092.50	0.00	151.61	0.00	0.00
-0.5	4092.50	-36.30	151.48	0.40	-0.41
-1.0	4092.50	-72.52	151.06	0.79	-0.83
-1.5	4092.50	-108.58	150.38	1.17	-1.26
-2.0	4092.50	-144.42	149.42	1.54	-1.71
-2.5	4092.50	-179.95	148.21	1.90	-2.16
-3.0	4092.50	-215.10	146.74	2.24	-2.63

Example roll format output

10 Local Coordinate Systems

10.1 Overview

This chapter describes the concept of local coordinate systems, how to create them and convert points such that they are defined in a local coordinate system rather than the global system. A solver option that makes use of local coordinate systems to have moving markers is also discussed.

This chapter contains the following sections:

10.2 Introduction

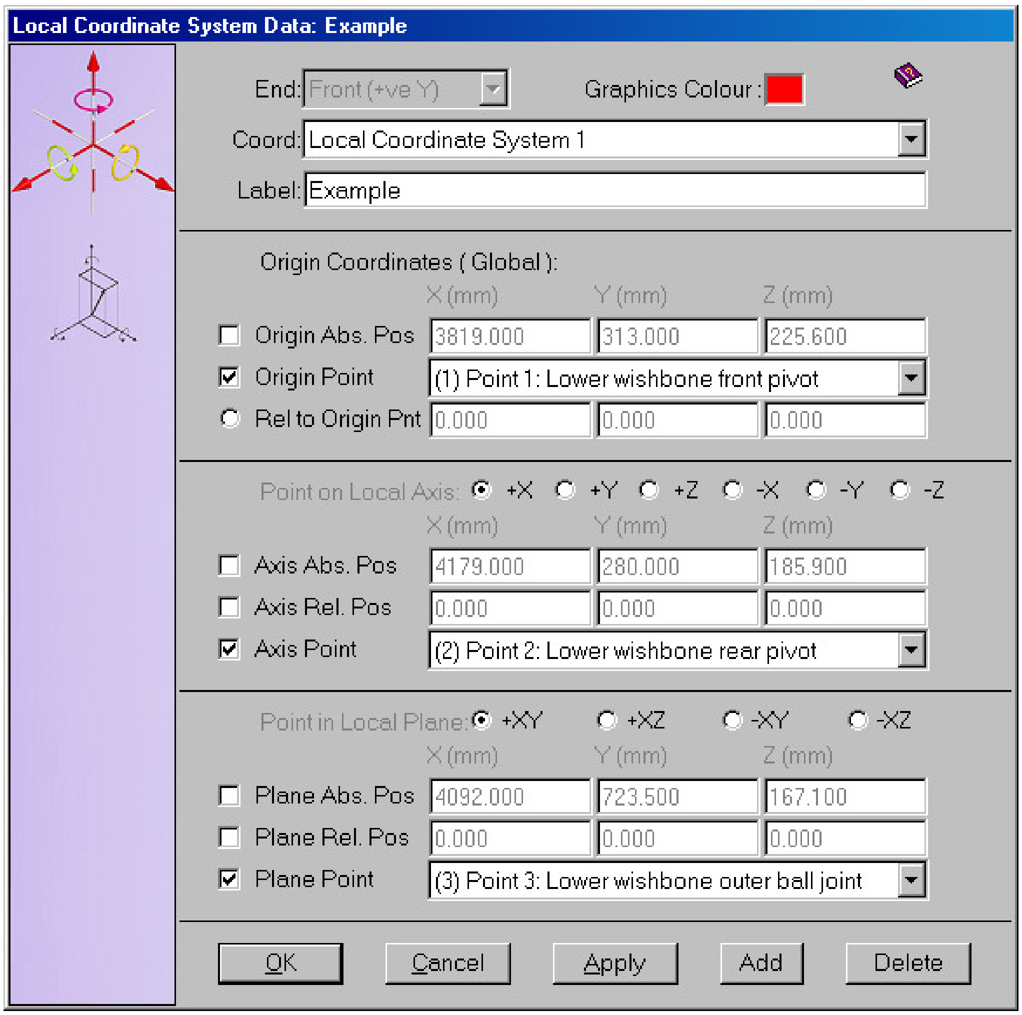
Local coordinate systems have been included primarily to locate the initial static position of a hard point in a local rather than global system. The local coordinate system is defined using existing model points to locate the origin, and axis directions. Points defined in local coordinate systems are solved and treated in exactly the same way as any other model point. It is only the initial placement of the point that is different.

The static global position of a point defined in a local coordinate system will change if its local coordinate system is moved or modified. Thus you can for example use this feature to place a number of points on a part into the same local coordinate system then move the coordinate system origin to relocate the part in the model. This achieves a different effect to that from the *Edit / Change Mode / Retain Parts* action which impacts on all points in the model and is limited to moving ‘ground’ points.

A solver option is available to make use of local coordinate systems, to have moving markers that are incrementally located by the local coordinate system, rather than fixed on a part. This is useful for applying external forces to a model at a point such as a trailed TCP point that is required to stay a fixed trailed distance from the incremental tyre contact point but rotate with toe angle. This solver option needs to be set through the main template editor.

10.3 Creating a Local Coordinate System

To create a local coordinate system open the coordinate system display via main menu entry *Data / Local Coordinate Systems…* select the 'add' button and provide the definition data.



Creating a Local Coordinate System

The properties of the local coordinate system are:

Label: Used in menu selection boxes and graphical display to help identify specific coordinate systems.

Origin Coordinates (Global): Identifies the position of the local coordinate systems origin in the global axis system. This can be entered as a position, or relative to another point in the model. Obviously setting the relative dimensions to zero will place the origin directly at the selected point.

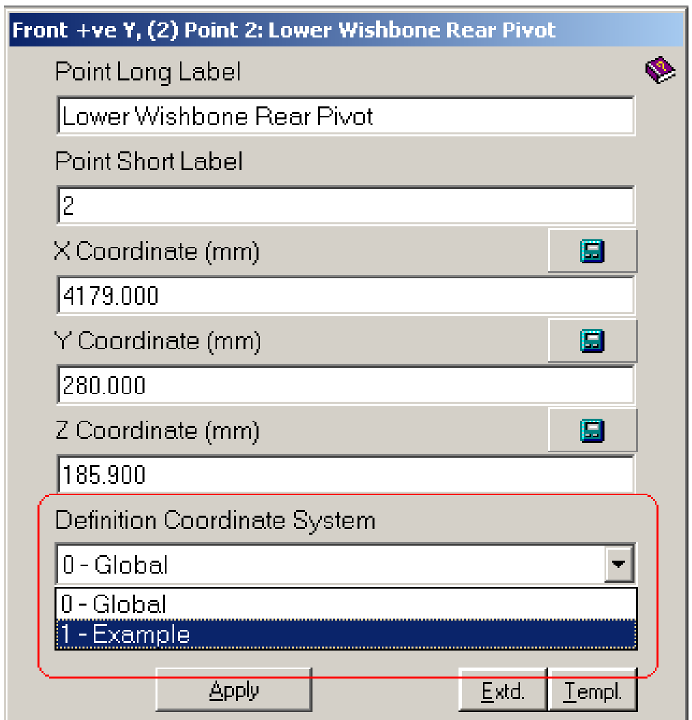
Point on Local Axis: Identifies a point on the chosen axis (select from x, y, z, -x, -y and -z). Define its position either as a global position, or relative to the origin using global offsets or relative to another point in the model. Again in the relative to point option setting the relative dimensions to zero will place the axis point directly at the selected point.

Point in Local Plane: Identifies a point in the chosen plane (four plane options are presented that vary depending on which axis has been selected for the above item). Define its position either as a global position, or relative to the origin using global offsets or relative to another point in the model. Again in the relative to point option setting, the relative dimensions to zero will place the plane point directly at the selected point.

Any number of local coordinate systems can be used in a model. This date editor also allows you to delete the currently selected system. Care should be taken when deleting coordinate systems as this may affect points that currently still use this local system.

10.4 Changing a point to use a Local Coordinate System

To change a point to use a local coordinate system, change to edit mode and select the required point from the graphical display. This needs to be done in the 'kinematic' mode, since in 'compliant' mode the bush property editor is displayed rather than the point editor. The point edit display has a selection box that identifies the coordinate system to use for its definition.



Changing to a Local Coordinate System

To change from global to local, simply select the 'definition coordinate system' selection box, and pick the required system. As you switch between one system and another, the points' x, y and z definition values are changed to retain the resultant global position of the point. This ensures that a model remain valid as you switch definition system. When defined in a local system, the point editing utilities change the label to indicate that it is a 'local' dimension.

Note that it is possible to recursively define a point in a local coordinate system when the point is also used to define the local coordinate system. This must be avoided.

10.5 Exercise: Using a local coordinate system

In this exercise, we will create a local coordinate system for the lower wishbone of a conventional double wishbone, and switch other points on the part into this new coordinate system.

Create a new model, using template type 1 ‘double wishbone, damper to lower wishbone’. Set to Kinematic mode, (i.e. compliance ‘off’).

Create a local coordinate system, label ‘Example’, with its origin at point 1 ‘Lower wishbone front pivot’. Put the point on the local ‘Z’ axis as point 2 ‘Lower wishbone rear pivot’.

Set the point in the local ‘X-Z’ plane as relative to the origin, 0.0, 0.0, 100.0

Select the ‘Ok’ button to create the local coordinate axis. The graphic display will now show the new axis symbol. If it does not appear, check you have created it and that the visibility settings are ‘on’, *Graphics / Enhanced Visibility / Settings / Local Coordinate System Axes.*

Now edit point 3 ‘Lower wishbone outer ball joint’, and change the definition coordinate system to use our recently created local system (note the change in labels and the x, y and z values. Repeat for the damper attachment and spring attachment points.

To test the impact of our newly defined points, change to ‘drag’ edit mode and drag point 2 along the y-axis. Note how our modified points pivot with the local coordinate system. Repeat the dragging with point 1, you will see the same sort of effect.

Finally create a temporary group that includes both point 1 and 2, then dragging either point 1 or 2 has the other local points translating with the dragged movement (‘Edit / Groups / Pick temporary’).

11 Ball Joint Results Display

11.1 Overview

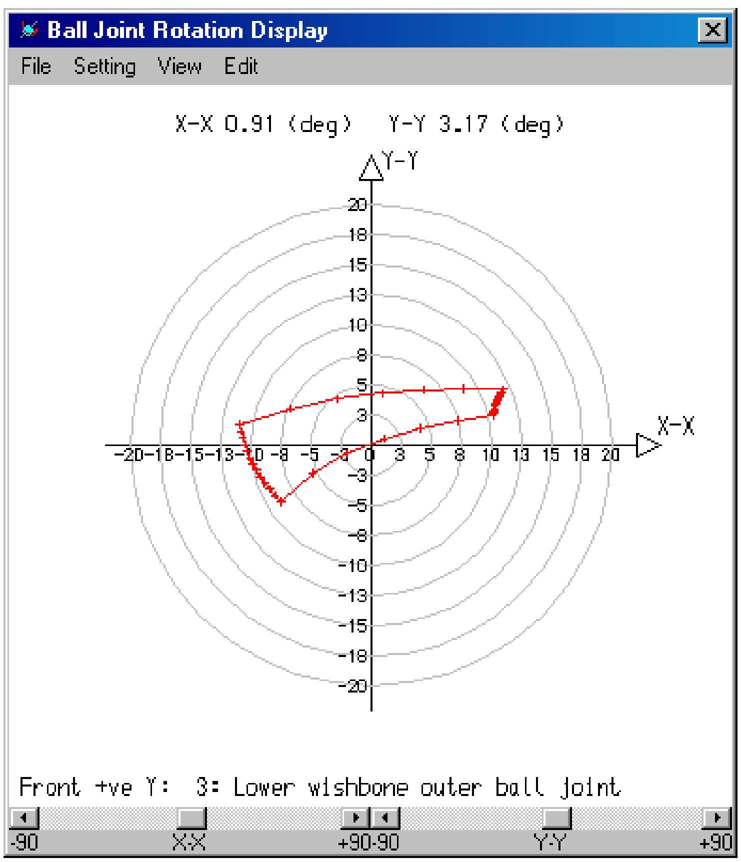
This chapter describes the use of the ball-joint results display tool. A range of options allows individual ball joint rotations to be plotted for the prescribed combined wheel motion displacement mode. Rotations can be referred to global axes, local axes or specified marker points.

This chapter contains the following sections:

11.2 Introduction

The ball joint results display shows, on an individual joint by joint basis, the articulation of the selected joint for the current wheel motion definition. Rotations are displayed graphically on an 'x-y target' display (z axis is out of the page) with rotations being relative to either the global axes, local bush axes or two specified axis markers. To open the display, select the *Results / Ball Joint Target Rotations…* menu item. Note that this option is only available in compliant mode. Because of the 'off-line' calculations performed with this module it may take a few seconds for the dialogue to initially open.

The format for the display is saved as a 'set' of details. Up to 25 different sets can be pre-formatted and saved as part of the INI file. A format includes the displayed point, the axis system, graphical axes properties and auto-centring options.



Example Ball joint display – Global axes

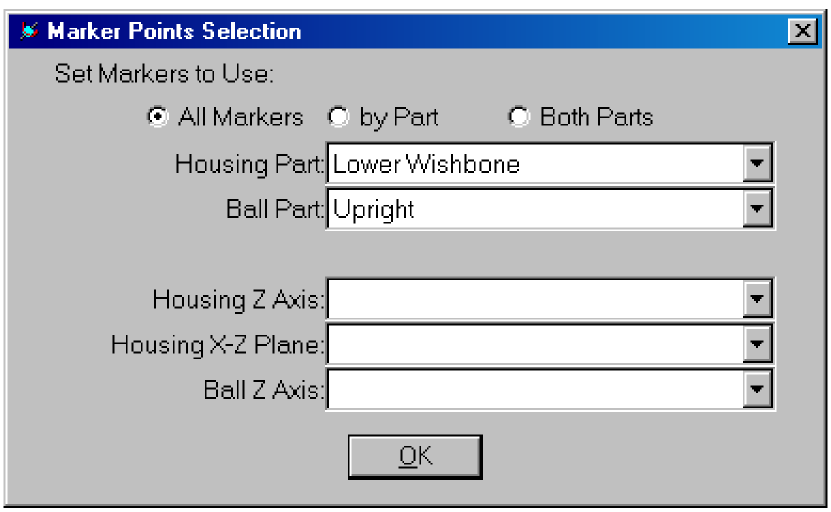
11.3 Setting the Displayed Point

With the display open, the details of the displayed point is shown across the bottom of the graph. To change to another point select the local menu *View / Set Point* and select the required point from the presented list of joints. Note that the list only presents points within the template that are 'tagged' as bushes. Because of the 'off-line' calculations performed this may take a few seconds for the display to refresh.

Alternative method of changing the displayed point is to pick an alternative pre-defined setting (assuming others have been defined).

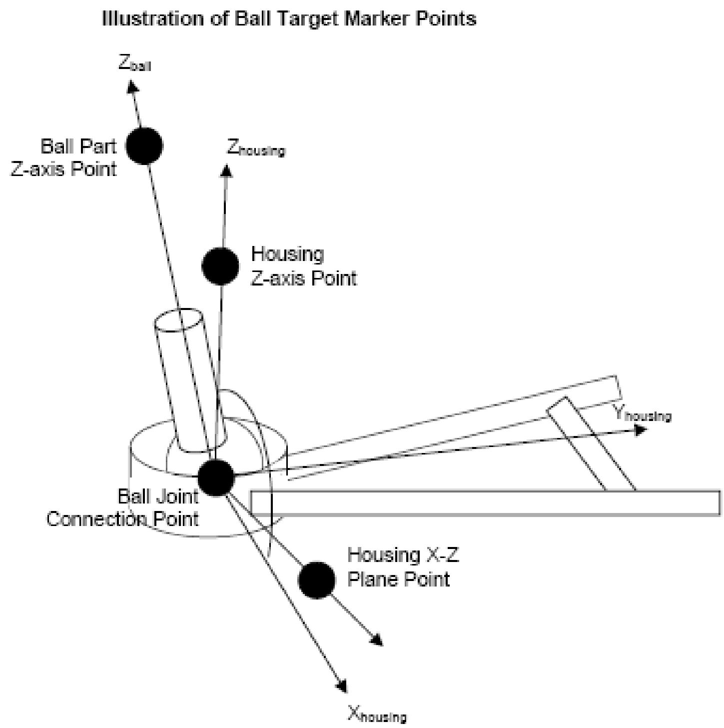
11.4 Changing the Reference Axes

The display shows the angular displacement of the ball joint in the x and y directions referenced to a z-axis. The default setting is for these axes to be aligned with the model global axis system. The *View* menu allows the display to be switched, to be relative to the bushes local axis system (which the user can define as a property of the associated bush). The axis system can also be set to use three points in the model to be local axis markers. With this option you can also control the parts that the housing and the ball are associated with.



Setting the Local Axes markers

If needed, extra points can be added to the template, for use as dedicated axis markers. The available markers in the selection boxes can be filtered to list just those associated with the part, both parts, or all markers.



Local Axes markers

11.5 Hard Copy

This results display supports the normal printing options. These include Copying to clipboard, Printing (with prompt), Printing to the default printer or Export to Excel.

11.6 Auto-Centring

The calculated points display can be re-positioned about the z-axis with centring shifts in both X and Y directions. This centring can be automatically applied to put the displayed locus centrally in the display, or the user can edit the values directly as a single correction.

The current centre offset values are displayed at the top of the display. As well as direct editing, they can also be modified via the two sliders located at the bottom of the display.

12 Internal Optimiser

12.1 Overview

This chapter describes the use of the internal optimiser to provide a numerical method of identifying an 'optimum' solution. The requirements for this optimum solution are defined as a series of kinematic and/or compliant targets. The solution can be achieved by moving identified points within a defined limited space, and/or varying bush properties, again within a limited variation.

This chapter contains the following sections:

12.2 Introduction

The optimiser is a fully integrated solution technique for quickly and efficiently identifying 'optimum' solutions. The approach used is not based on a design of experiment or Taguchi type techniques. It instead uses a combination of scoring, sensitivity and trends to identify the optimum.

The scoring technique is used to convert the deviation of the current solution from all desired characteristics and values to a single number (or score). The objective of the optimisation process is to reduce this score to zero.

The score is determined by summing the deviation of each point on the calculated curve (e.g. camber angle against bump travel) or value when compared to a defined target curve. Any number of characteristics can be included in this way, the summation of each made with reference to a weighting factor. It does rely on the user knowing what target responses they require for each included characteristic. Increasing the number of characteristics does not significantly alter the analysis time of the optimisation process.

The variables for the optimisation are identified as point positions and/or bush properties. Whilst the optimisation can include both kinematic and compliant variables they are more normally performed in isolation of each other. Each included variable has a user-defined range to contain the solution within sensible bounds. Also, within these bounds, a minimum resolution is specified to limit the number of solution steps. The number of variables included directly affects the run time for the optimisation, so it should be kept to the minimum possible.

The optimiser assesses each variable to identify a sensitivity value. This is the amount the score changes over the defined range at the specified step size. This then sets the order that each variable is adjusted to produce its minimum value before moving on to the next variable in the sensitivity order list. The user can run the optimisation either from the most to the least sensitive or more successfully from the least sensitive to the most sensitive.

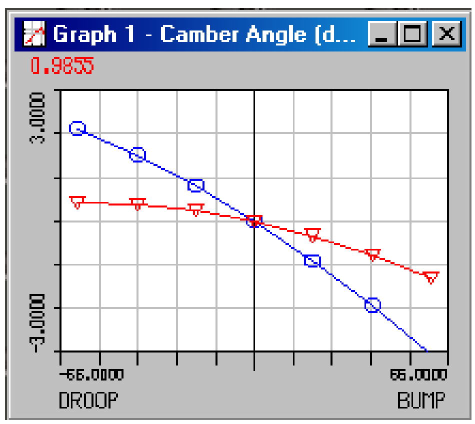
Thresholds can be set to ignore variables whose sensitivity is significantly less than the most sensitive, as well as a solution acceptance tolerance on getting the deviation score down to zero.

At the end of the optimisation process, the user is presented with the original score and the best position score. The user can then optionally accept the new 'optimised' solution, or return to the original models data.

12.3 Setting Kinematic Targets via User Lines

The kinematic targets for the optimiser, for items such as camber, toe and roll centre height are defined via user lines. These are defined for each of the required suspension derivatives graphs. Note that graphical element properties such as ‘distance’, cannot be used for the optimisation, as you cannot define user lines for them.

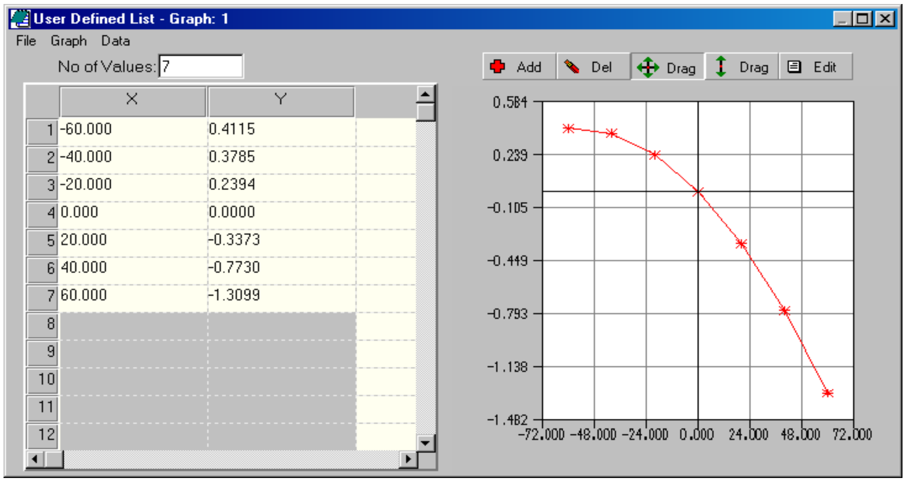
User lines can be defined directly or extracted from user line data sets. These data sets are managed through a series of menu options that allow the user to add, edit or remove user line data from data sets. Each dataset can have any number of complete sets of user lines stored in them, and any number of datasets can be referenced by the application at any one time. These data set files are stored by their full pathname in the users ini file, such that they are searched for at program start up, and added to the relevant menu tree. In a similar manner to 'add', datasets can be ‘removed’ from the available list. The dataset files can be shared between users, either through a central server type location or by local copies of the same file.



Example Graph showing User Line and deviation 'score'

A convenience menu function is provided, *Graphs / User Lines / Copy Front-2D data to User* for copying the current result line values to the user line. This menu function performs a copy not just for the currently visible graph parameters, but all available results (some 45 different parameters). Thus for a given target model, this function provides a complete set of user line curves, that could be stored/added to a dataset for use as a future target for an alternative model.

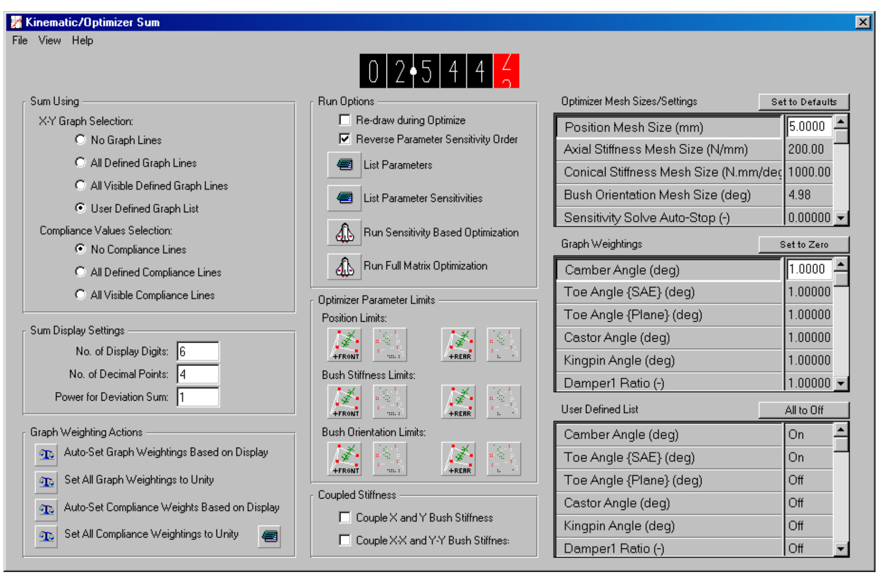
Individual graphs user lines can be edited directly through the right mouse menu option on the chosen graph. Note that user lines are stored and edited by corner. This is to support asymmetric model options.



Example user line edit

12.4 Weighting Settings for Kinematics

Each graph/curve used in summing the total deviation from its respective user line is included via its own weighting factor. This allows the user to balance the numerical differences, between say an angle such as camber, and a distance such as roll centre height. A number of convenience functions are available to automatically set these, but before we can look at these, we need to review how we specify which graphs we wish to include in the optimisation process. It doesn't necessarily follow that: if a particular graph is visible, it will be included in the optimisation score.



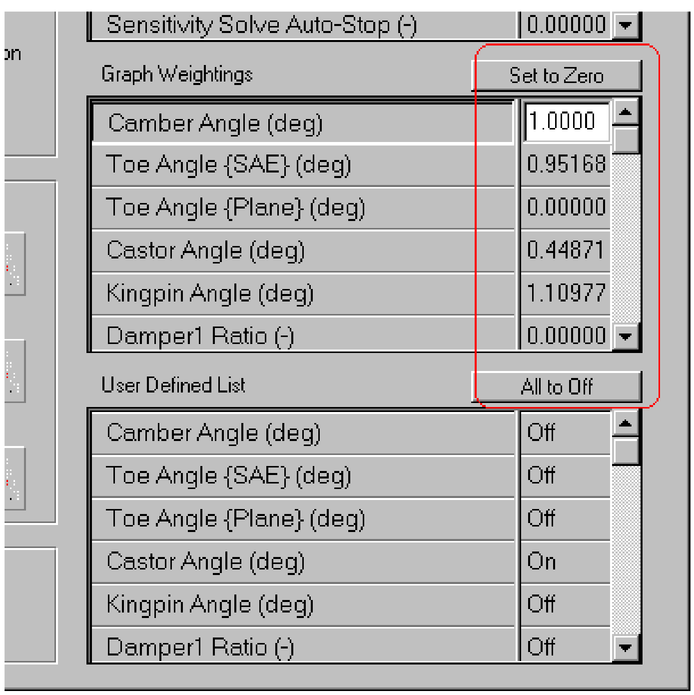
Optimiser Display Screen

All optimiser settings are accessed through the *Solve / Display Optimizer* display. This display can be with or without 'details'. The screen shot illustrates the 'with details' display.

Through this display, you can specify which graphs to include into the sum. The top left corner shows the 'sum using' options. We can choose to include No. Graph lines, All defined (i.e. any with a user line specified visible or not), All visible defined graph lines, (i.e. as per the previous option, but must also be visible), or a user defined list. The user-defined list is set graph by graph in the lower right hand corner. Note that the top left hand box also has the compliance sum options that we will cover later.

Once we have identified which Graph selection to include, we can then set the appropriate weightings. The 'Graph weightings' panel shows the current value for each graph. We can either edit them directly, or use one of the menu options (lower left). This will either set them all to zero, all to unity or to a weighted value based on the individual graphs y-range. The latter option attempts to provide a convenience-weighting regime without recourse to lengthy editing of each graphs value. These weighting numbers will directly influence the optimisation process, in both the order and final achieved model. When using this auto-weight option, it is important to have the graph y-axes correctly set. This may be nothing more than using the graph autoscale option or may involve more specific axis setting.

You can also set some X-dist weightings, by editing the values accessed with the associated button next to Graph Weightings. The X values go from –1 (lower limit) to +1 (upper limit), 0 being the static position. The Y values define the shape of the curve. The appropriate X-weightings are then extrapolated from these values, and used to put a weight on all the parameters at some specific position. For example, a curve with a point at x=1 and y=0 will lead to the multiplication of all the parameters at full deflexion by 0.



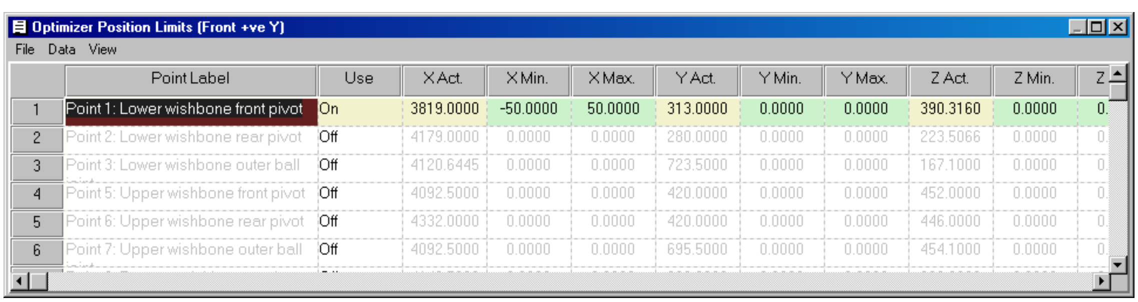
Graph weightings ringed - Set by Auto-weight routine

12.5 Identifying Positional Optimiser Parameters

Having specified how we are to determine the targets and summation, we need to identify which hard point positions we intend to move. We can set positional parameter limits by individual corner. In the simplest single corner symmetrical model we would only be interested in one corners points set.

Accessing the positional parameters settings

Within each corner set, we identify which points are to be included with a simple on/off selection. On the same line, we also define the limits of motion in each axis for the selected point. Menu options are available to set the limits as the current position, or the current points tolerance limits. As a display option, we can either view the parameter limits as absolute or deltas.

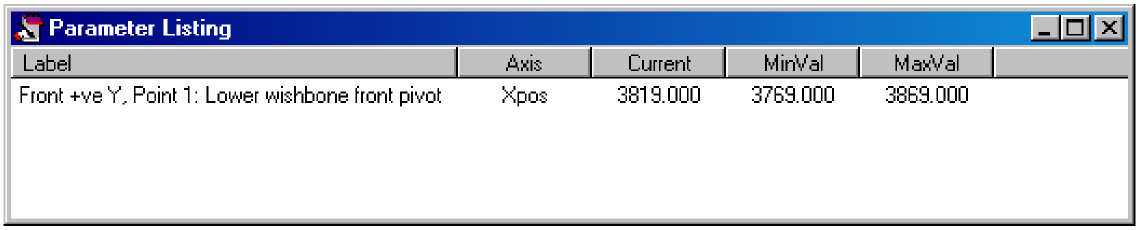


	Point Label	Use	X Act.	X Min.	X Max.	Y Act.	Y Min.	Y Max.	Z Act.	Z Min.	Z
1	Point 1: Lower wishbone front pivot	On	3819.0000	-50.0000	50.0000	313.0000	0.0000	0.0000	390.3160	0.0000	0.
2	Point 2: Lower wishbone rear pivot	Off	4179.0000	0.0000	0.0000	280.0000	0.0000	0.0000	223.5066	0.0000	0.
3	Point 3: Lower wishbone outer ball	Off	4120.6445	0.0000	0.0000	723.5000	0.0000	0.0000	167.1000	0.0000	0.
4	Point 5: Upper wishbone front pivot	Off	4092.5000	0.0000	0.0000	420.0000	0.0000	0.0000	452.0000	0.0000	0.
5	Point 6: Upper wishbone rear pivot	Off	4332.0000	0.0000	0.0000	420.0000	0.0000	0.0000	446.0000	0.0000	0.
6	Point 7: Upper wishbone outer ball	Off	4092.5000	0.0000	0.0000	695.5000	0.0000	0.0000	454.1000	0.0000	0.

Example Parameter setting - Point 1 selected viewed as deltas

The above screen shot shows just point 1 as having been selected, and only the x-axis will be considered, since the limits for y and z are set to 0. This display has been switched to show delta's and thus the zero's imply no change from the current actual value.

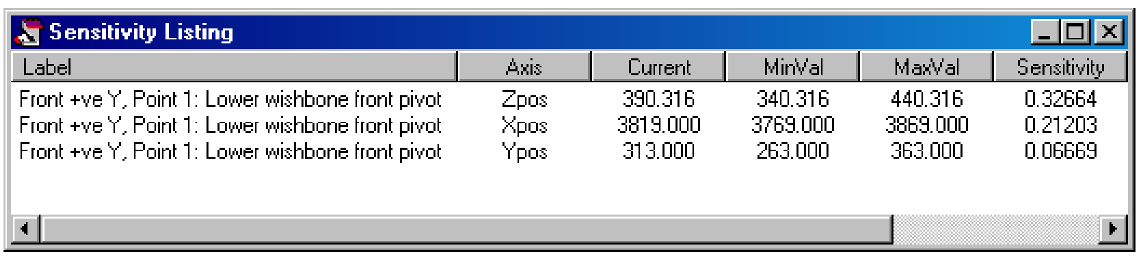
Once we have set all our required parameters, we can list them through the 'List parameters' button. Our simple case of a single parameter is shown below.


Parameter Listing

Label	Axis	Current	MinVal	MaxVal
Front +ve Y, Point 1: Lower wishbone front pivot	Xpos	3819.000	3769.000	3869.000

Parameter listing - single point position single axis

In a similar display, we can list the parameters in sensitivity order. This can be useful in identifying parameters that have a small impact on the solution and they can thus either be removed from the list, or used to identify a value for the 'sensitivity ignore ratio'; the objective being to reduce the number of parameters to produce a quicker solution time.


Sensitivity Listing

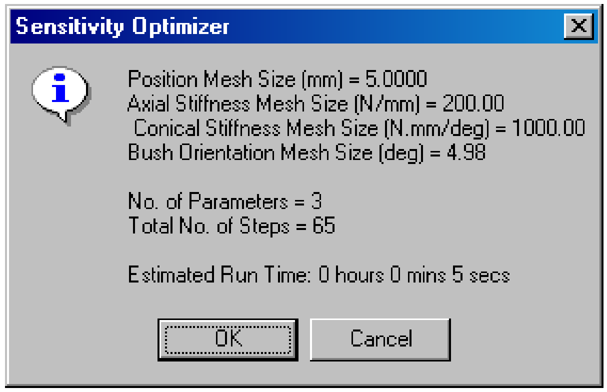
Label	Axis	Current	MinVal	MaxVal	Sensitivity
Front +ve Y, Point 1: Lower wishbone front pivot	Zpos	390.316	340.316	440.316	0.32664
Front +ve Y, Point 1: Lower wishbone front pivot	Xpos	3819.000	3769.000	3869.000	0.21203
Front +ve Y, Point 1: Lower wishbone front pivot	Ypos	313.000	263.000	363.000	0.06669

Sensitivity listing - three parameters shown, Y insensitive.

12.6 Running a Kinematic Optimisation

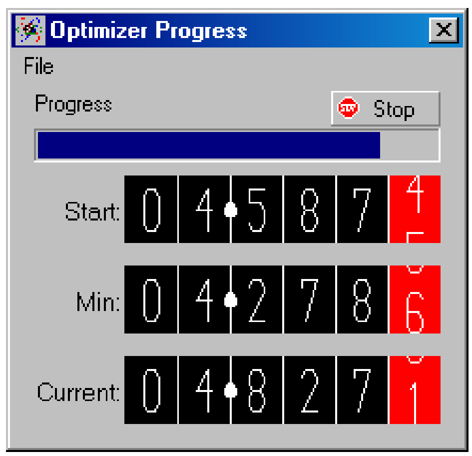
With the targets defined and the parameters specified, we can run the optimiser. As stated previously, the solver will run through each parameter in a sensitivity-based order, selecting the setting for each parameter that gives the lowest deviation. This positional setting for minimum score is retained as the solver moves to the next parameter. During this process the user can view the intermediate steps in the graphical window as well as the animated score. Two optimisation options are available, one of which is a simple full matrix solution. This full matrix option is impractical for all but the most simple cases (simple in terms of few variables). The normal solution method is the 'Sensitivity Based Optimization'.

When you first select the optimisation button, a summary display is shown, listing the step size resolutions, number of parameters, total number of steps and an estimate of the run time.



Optimisation summary

Selecting Ok will start the optimisation and display the progress display shown below. The display shows the progress and three scores. The scores are the value at the start, the minimum value found so far and the score for the last iteration.



Progress display

12.7 Kinematic Tutorial

To illustrate the previous discussion, we will run a simple example involving three graph curves and three parameters.

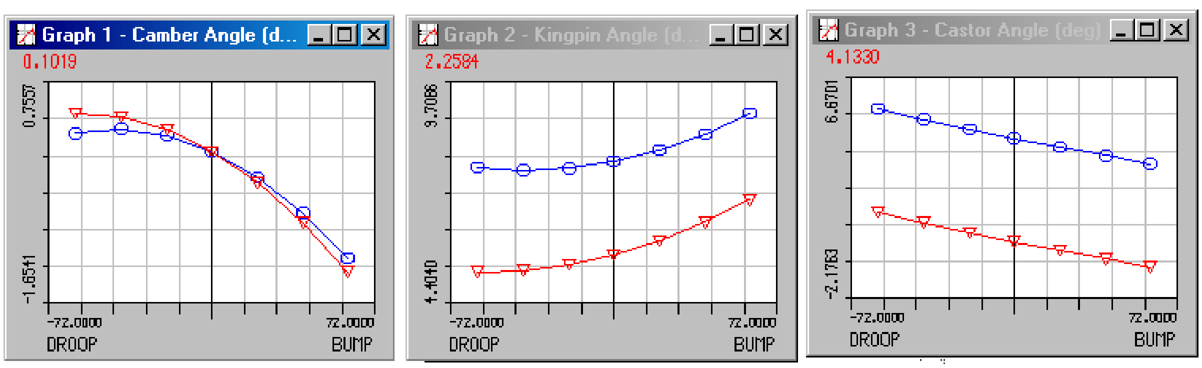
Create a new single corner front model based on default type 1.

Open three graphs, Camber Angle, Kingpin Angle and Castor Angle, *Graphs / New*. Ensure you are set to 3d bump module, and that bump/rebound is set to +-60mm with a 20mm increment, *Data / Parameters*.

Use the graph convenience menu item to pre-fill the user lines with the current data values, *Graphs / User Lines / Copy Front-2D data to User*.

Edit point 3, the outer lower ball joint to a new position, X=4072, Y=733.5, Z=177.0

Before opening the optimiser display autoscale the graphs. They should now appear as below. Note deviation scores visible.



Graph screen shot for tutorial

Open the Optimiser display, *Solve / Display Optimiser*, and set the display to show details. For this simple example, we will use a fine positional increment, so set 'Position Mesh Size' to 1.0 mm.

We are going to have point 3's X,Y and Z position as variables so open the relevant 'Position Limits' display and set them 'on' and range to the current tolerances. Check by comparing your parameter list with that shown below.

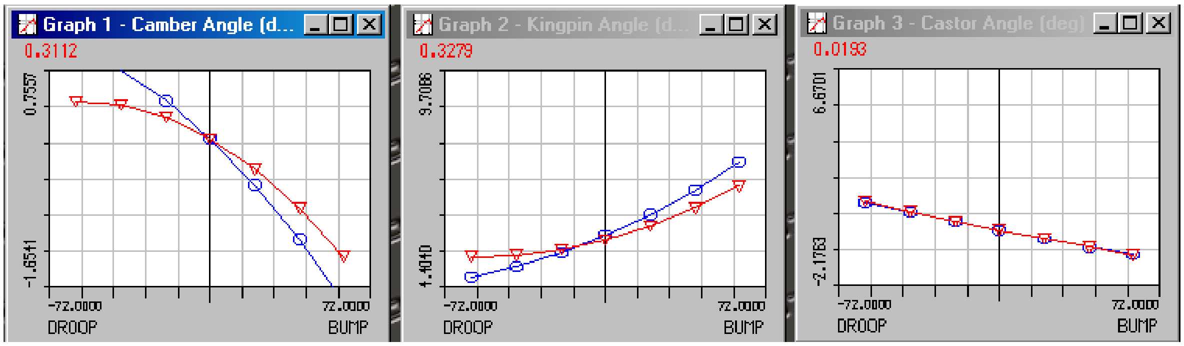
Parameter Listing

Label	Axis	Current	MinVal	MaxVal
Front +ve Y, Point 3: Lower wishbone outer ball...	Xpos	4072.000	4067.000	4117.000
Front +ve Y, Point 3: Lower wishbone outer ball...	Ypos	733.500	698.500	748.500
Front +ve Y, Point 3: Lower wishbone outer ball...	Zpos	177.000	142.100	192.100

Parameter Listing

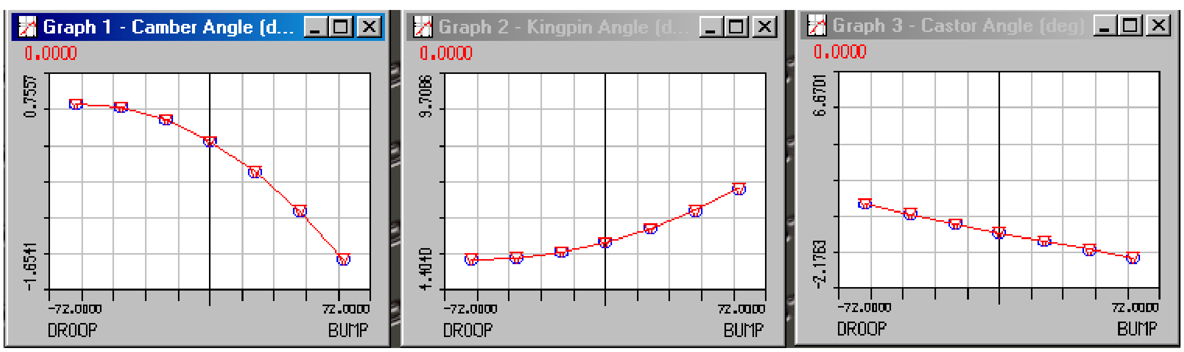
Before we can run the analysis, we need to define what weightings we will apply. Initially, try the 'Auto-Set weightings based on display' option. This should give 1.0 (camber), 0.2724 (castor) and 0.4543 (kingpin) for our three graphs.

Try running the optimiser with these weightings…



Auto-Weightings Result - Score = 0.4654

The auto-weightings do not produce a very effective solution. In particular, the Camber curve is not very close. Try re-running, but this time double the camber weighting, i.e. 2.0, 0.2724, 0.4543. No need to reload the original values, we can re-run from this new position.

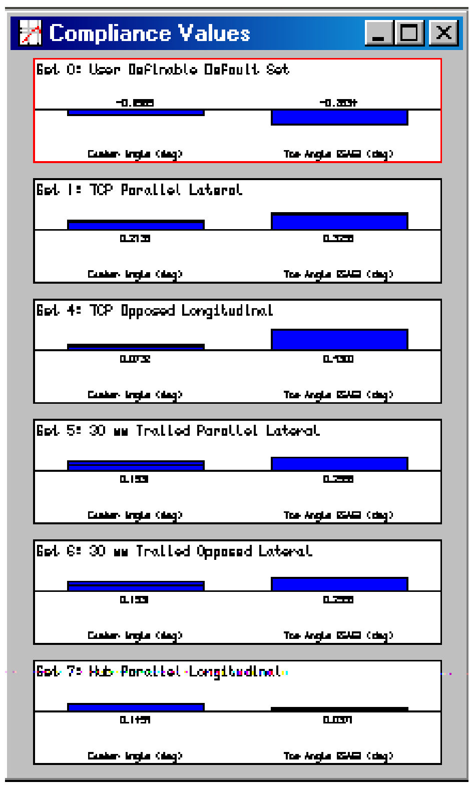


Revised Weighting Graphs

In this case we get a much-improved match, and a score of zero. Are the co-ordinates for point 3 the same as originally? Would we have got here in one step had we set the weightings to 2.0 etc. from the start?

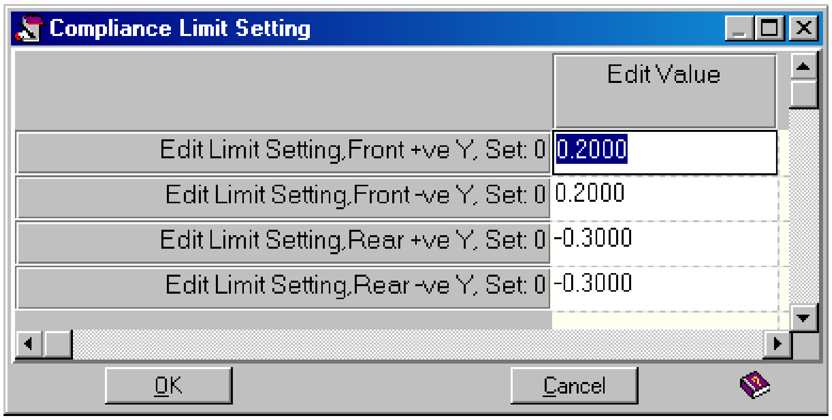
12.8 Setting Compliant Targets via Compliance Coefficients

To extend the use of the optimiser into the use of tuning compliant properties, such as bush stiffness and orientation, we need to have an additional compliant results scoring method, similar to that used with our derivative graphs. We can of course use the graph curves user lines as targets for compliant bush optimisation, but it is more normal to use the 'Compliance Values' bar chart display (*Results / Compliance Bar Values*), as this allows us to use compliant specific results under a range of different load cases.



Example Screen Shot - Compliance Values

Each bar on the display has an associated target or limit setting. These are drawn as horizontal lines across each bar display. We can use the sum of differences from these 'limit' settings to control our optimisation process in a similar way to the x-y graph user lines. To set the limit values for a particular bar, select it with the right mouse button, and then edit as required. Alternatively, to edit them all through a single display, select the menu option *Edit All Line Limits/Scale/Weights…,* from the right mouse menu on the compliance values graph.



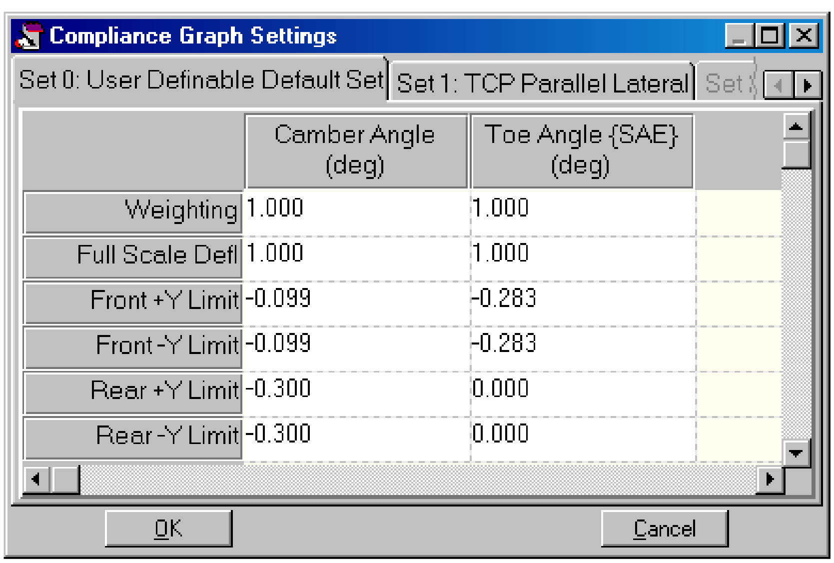
Limit Setting for Individual Bar

In a similar manner to user line setting, a convenience menu option is available, *Set All Limit Values to Current* from the right mouse menu, to take the current values as required limits.

No database system exists for these limit settings, but they are saved into the users ini file.

12.9 Weighting Settings for Compliance Targets

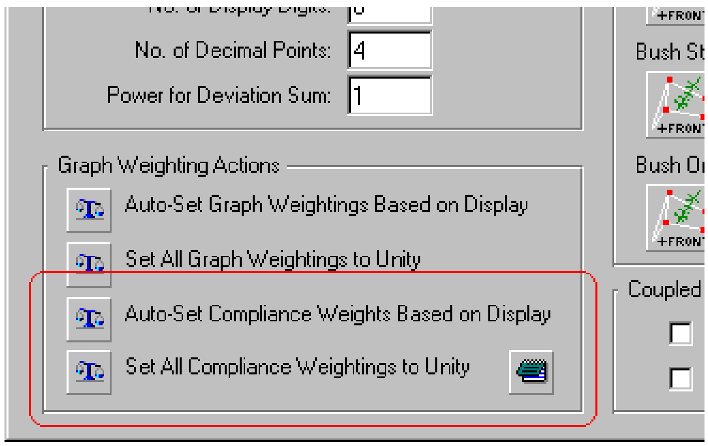
As with the kinematic user lines, we need to apply a weighting factor to each compliant bar graph. We can either edit them individually using the right mouse menu item *Edit Weighting Setting…* or use the *Edit All* option.



Setting Weighting Values via the edit all display

Individual weighting values are applied to each separate bar in each load set, although the same value is used for each corner within a single bar variable, though each corner can have a separate target/limit.

The optimiser display has two convenience options for auto-setting the bar weighting values. The lower left panel has *Auto-Set Compliance Weights Based on Display* and *Set All Compliance Weightings to Unity*. They function as implied to provide one-click settings for the weightings.



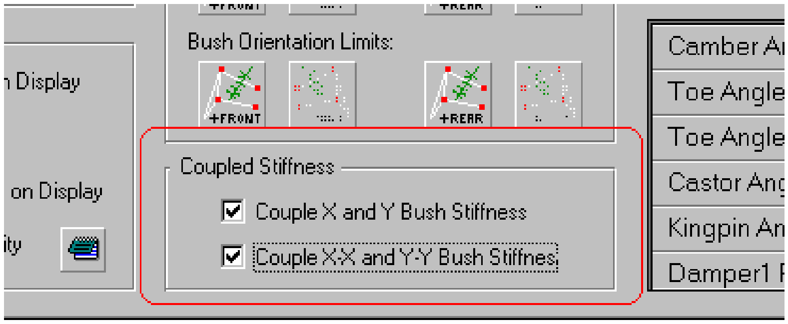
Compliance Weighting Settings

12.10 Specifying Compliant Optimiser Parameters and Limits

The compliant properties available as parameters for the optimiser are the bush X, Y and Z axial stiffness, the bush X-X, Y-Y and Z-Z rotational rates, and the bush orientation. The bush orientation is controlled by the optimiser through an Euler sequence of local axis rotations about the X, Y and Z axes.

Obviously, compliant optimisation can only be applied to model points that are either a connection to ground, or a connection between parts. They must also have been identified as 'compliant' rather than 'rigid'.

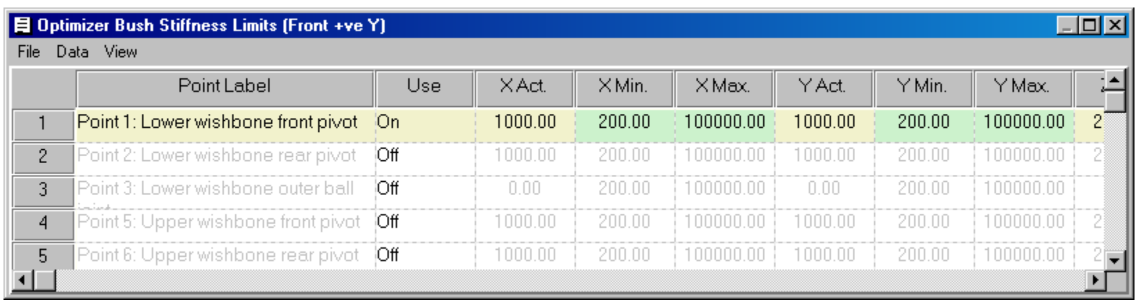
Because of the often symmetric nature of bushes, special options are provided to couple the X and Y stiffness (i.e. no need to set both the X and Y axial stiffness as a parametric variable just change one and the other will be automatically made the same). A similar option is available for the x-x and y-y rotation rates.



Coupling the X and Y Stiffness parameters

To select the required bushes to optimise, select the 'Bush Stiffness Limits' icon. Within this display, toggle on the required bushes, and set the x, y and z limits in a similar manner to that performed previously for point position.

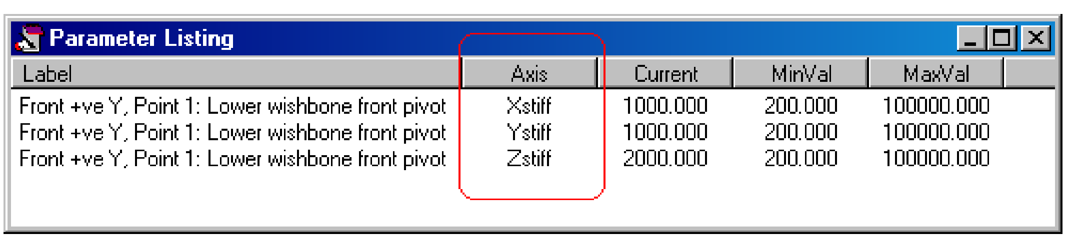
You can pre-fill all the stiffness values using the two convenience menu items, *Pre-Fill Min/Max Using Defined Values*, and *Pre-Fill Using Actual Values*. The first of these two options requires the user to specify what the min and max limits should be.



	Point Label	Use	X Act.	X Min.	X Max.	Y Act.	Y Min.	Y Max.
1	Point 1: Lower wishbone front pivot	On	1000.00	200.00	100000.00	1000.00	200.00	100000.00
2	Point 2: Lower wishbone rear pivot	Off	1000.00	200.00	100000.00	1000.00	200.00	100000.00
3	Point 3: Lower wishbone outer ball	Off	0.00	200.00	100000.00	0.00	200.00	100000.00
4	Point 5: Upper wishbone front pivot	Off	1000.00	200.00	100000.00	1000.00	200.00	100000.00
5	Point 6: Upper wishbone rear pivot	Off	1000.00	200.00	100000.00	1000.00	200.00	100000.00

Single Bush enabled, limits set for x, y and z

Once all the required parametric bush properties are defined, they can be viewed via the list parameters option. The axis will be identified as 'Xstiff', 'Ystiff' and 'Zstiff' as applicable.

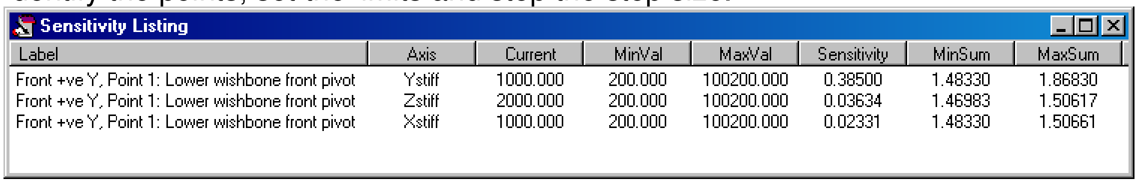


Label	Axis	Current	MinVal	MaxVal
Front +ve Y, Point 1: Lower wishbone front pivot	Xstiff	1000.000	200.000	100000.000
Front +ve Y, Point 1: Lower wishbone front pivot	Ystiff	1000.000	200.000	100000.000
Front +ve Y, Point 1: Lower wishbone front pivot	Zstiff	2000.000	200.000	100000.000

Parameter listing, stiffness variables indicated.

We can similarly review sensitivities of the defined parameters. The step size used between minimum and maximum stiffness is defined in the top right panel entries. Here you define the step sizes for both the axial and rotational stiffness.

The same process can be applied to set up bush rotations as parametric variables. Identify the points, set the limits and step the step size.



Sensitivity Listing

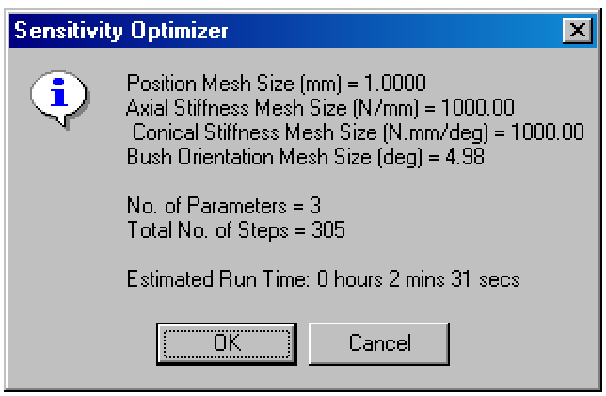
Label	Axis	Current	MinVal	MaxVal	Sensitivity	MinSum	MaxSum
Front +ve Y, Point 1: Lower wishbone front pivot	Ystiff	1000.000	200.000	100200.000	0.38500	1.48330	1.86830
Front +ve Y, Point 1: Lower wishbone front pivot	Zstiff	2000.000	200.000	100200.000	0.03634	1.46983	1.50617
Front +ve Y, Point 1: Lower wishbone front pivot	Xstiff	1000.000	200.000	100200.000	0.02331	1.48330	1.50661

Sensitivity listing - note min and max sums given

12.11 Running a Compliant Optimisation

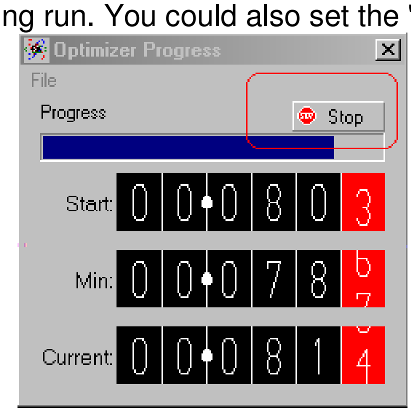
With the variables, limits and parameters defined, we cannot run the optimisation in exactly the same way as for the previous kinematic only example.

Because this is a compliant solve, the solution run times will be longer than previously. Check the total number of steps and run time for sensible values before submitting the optimisation run. You may need to revise the step size to reduce the run time.



Check run time before submitting

You can stop an optimiser run before it finishes, and still have the option to take the lowest score solution already found. This can be useful when a suitable solution has already been found, in a long run. You could also set the 'Auto-Stop' value higher.



Stop button identified

13 Interactive Template Builder Module

13.1 Overview

This chapter describes the interactive template builder module. This module provides the simplest method of generating a new template or modifying an existing one, being completely graphical in its implementation. The alternative method of direct template editing through the template spread sheet, (see the following chapters), can be used interspersed with the interactive module.

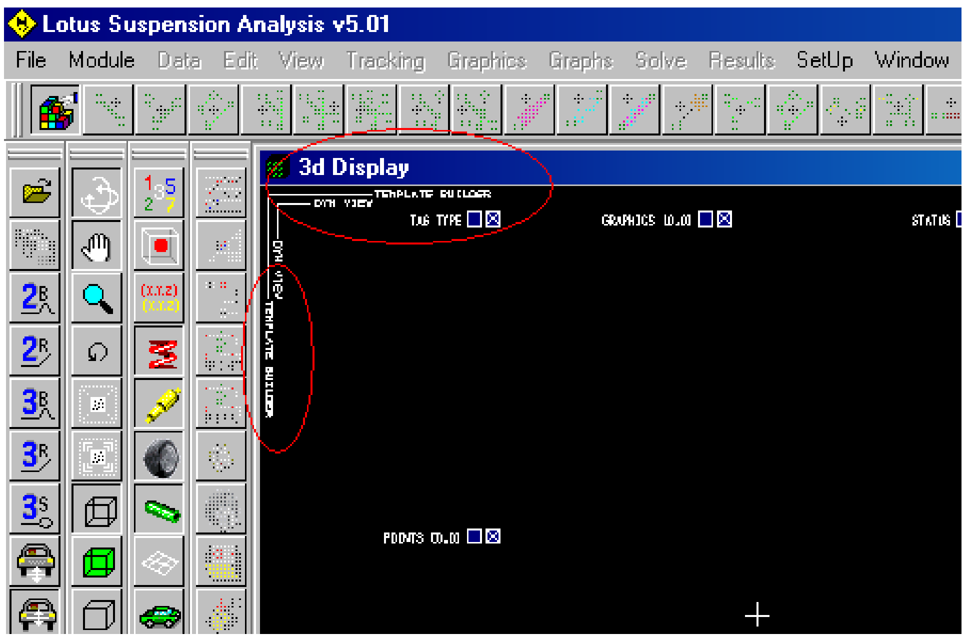
A number of the commands and actions that can be used within this module are also available when in the other 3D modules (i.e. bump), for use in the general modification of an existing models template. These were discussed in Section 3.0 “Interactive Template Modification”. The main difference between this module and previous template editing is that the user can work with all the elements of a template, so the user can add and delete parts. These parts can be joined together or joined to ground. Parts that are joined can be split for modification.

Whilst in this module no results are calculated as the model is not being solved over any particular displacement case. This makes for a far more stable environment in which to make template modifications without the risk of programs errors due to invalid or incomplete templates being solved.

This chapter contains the following sections:

13.2 Template Builder Screen Layout

The template builder mode is selected from the module main menu *Module / Shark / 3D Template Builder*. When switching to this module two toolbars become active that have icons for the main template builder commands, (note that because toolbars can be redefined by the user they may not appear exactly as shown in this document). The builder icons will be greyed out until a new template is opened (they will however be enabled if a model had already been loaded).



In the Interactive Template Builder

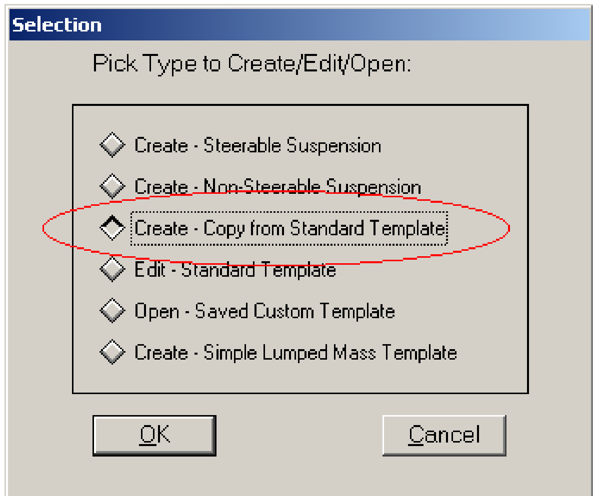
When in template builder mode the graphical display indicates this by showing the 'Template Builder' label in each corner of the display.

To illustrate the layout of the template builder screen and to review the 'New' template options we will generate a new template using the 'Copy from standard template' option. This creates a new template but from the start point of an existing defined template.

- ***Change to the Interactive 3D Template Module. Module / Shark / 3D Template Builder.***

- ***Create a new template. File / New.***

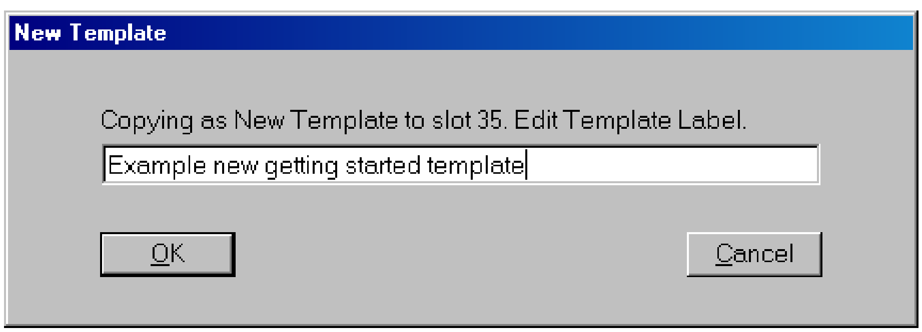
Five options are presented, three of which generate a new template, '*Steerable*', '*Non-steerable*' and '*Copy from Standard Template*'. The only difference in selecting between the steerable and non-steerable is that the steerable will function as though it is a front suspension whilst the non-steerable is edited as thought it were a rear suspension. The last two options open an existing template for further editing/modification.



New Template Creation Options

- ***Select 'Copy from Standard Template'.***

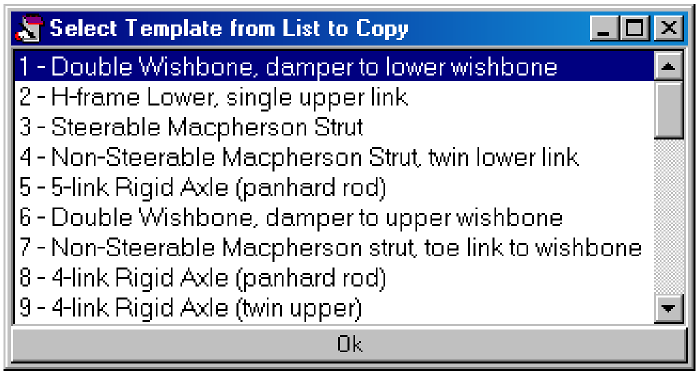
Each new template has a unique slot number, and takes the next free position. This 'Slot No.' is shown on the title of the 'New Template' label entry and will appear in the status bar once in the template builder mode.



Seting the Template Label for the New Template

- ***Enter a suitable label for the new template and Select 'Ok'.***

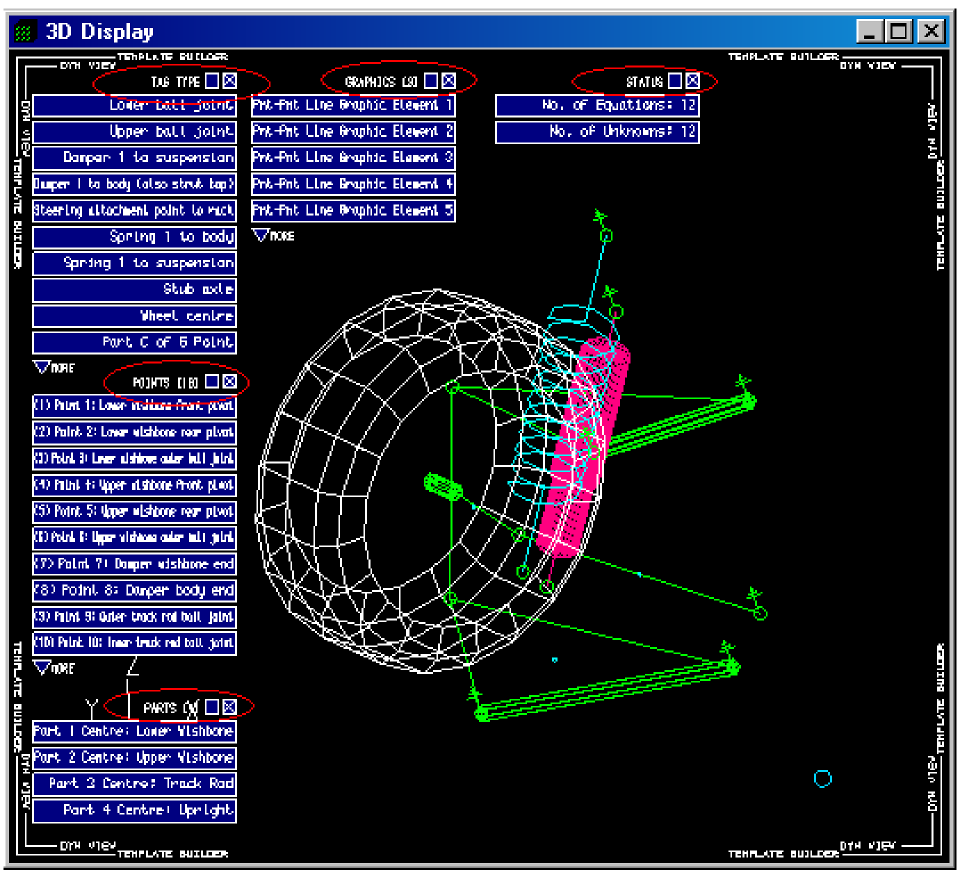
For this example we will select from the template list presented by the 'Copy from Standard Template' option. We will use standard template 1, "Double Wishbone, damper to lower wishbone".



Selecting the required Template to copy for the New Template

➢ ***Select standard template No. 1 from the list presented and Select 'Ok'.***

The Graphical display will refresh to show the selected template. The graphical display will show lists for 'Tag Type', 'Points', 'Parts', 'Graphics' and 'Status'. These lists are unique to the template builder module and provide the means by which entities can be individually selected and thus modified/deleted.



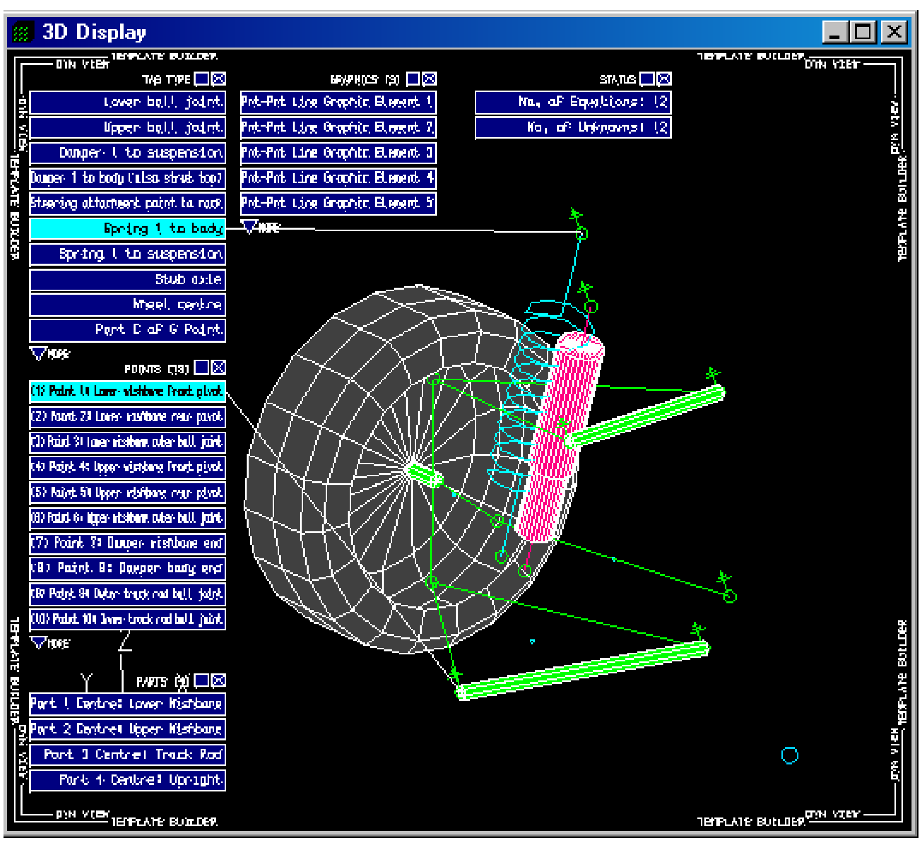
Individual Element Lists Highlighted

The lists have a number of selection actions. The visibility of the list can be turned on/off by picking with the mouse the box at the top right of each list. The second box at the top of each list controls the visibility of the ‘grey’ location lines. These lines connect each entry in the list to the appropriate position on the model.

If a list has more entries than can be viewed in the specified space then the hidden items can be scrolled into view by selecting the ‘triangle’ at the bottom left of the list. A similar triangle will appear at the top left if items are scrolled off the top of the list.

The position and size of each list can be modified by selecting the relevant ‘hot spot’ with the mouse. These hot spots are indicated by the cursor changing appearance to either the ‘position’ cursor or the ‘size’ cursor. The ‘position’ hot spot is at the top of the list in the middle, whilst the ‘size’ hot spot is similarly in the middle of the list but at the bottom.

Individual items in a list can be selected / un-selected with the mouse. The selected item is highlighted and a line drawn to the relevant screen position. For ‘Tag Type’ and ‘Points’ this screen position is the relevant points position in the model, for ‘Parts’ this screen position is the average of all associated active points on the part. For ‘Graphics’ the screen position is the location of the graphics’ hot spot.



Selection of Items from List, Note ‘to ground’ symbol

For the ‘status’ list only two items are given. The first when selected shows the location of all the defined equations and their equation type, whilst the second item shows the location in the model of the unknowns. When editing of the template is complete these two items should always be equal for a valid template.

13.3 Template Builder Individual Menu Options

Having selected an entry from a list, the right mouse then lists a context sensitive set of appropriate menu options. Some of these ‘listed’ menu options duplicate the general options that are available from the toolbars, such as for the ‘Part’ list the right mouse menu includes the ‘add part’ section. Other items in the ‘list’ menu duplicate some of the ‘standard’ edit functions, for example the ‘Points’ list menu has an ‘edit point properties’ option that is the same as the normal graphical point edit by picking the point directly when in ‘Edit’ mode. The objective of these right mouse menus is to collate all relevant menus in to one convenient menu.

The ‘Tag Type’ List is the summary of which points in the model are prescribed to have a specific function, simple examples of this would be the spring upper and lower points. This ‘tagging’ allows the solver to efficiently determine specific suspension type results, for example tagging of the spring points allows spring travel and spring ratio to be calculated. The right mouse menu options for Tagging allow you to remove a tag, change the point the tag is applied to, change the tag type for the listed point or add a new tag type by picking a type and then point. Tag types are split in to three groups, *Standard set 1*, *Standard set 2* (which is normally the same as in standard set 1 but applied to the opposite corner in a full axle model) and *Extended*.

13.4 Building a New Template, Example 1

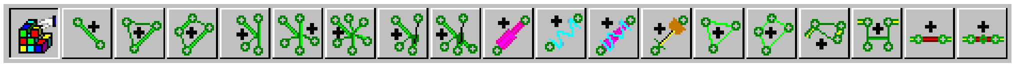
We will now build a new template from scratch to illustrate the use of the toolbar part icons and also the interactive template builder actions. We will create a double wishbone suspension similar to the template No. 1 that we just used. First create a new steerable template.

- ***Create a new steerable template, File / New and Select ‘Steerable Suspension’ and then Select ‘Ok’.***

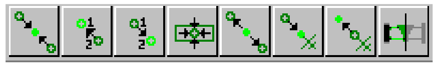
Provide a suitable label.

- ***Enter a label for the new template and then Select ‘Ok’.***

The template builder toolbar icons should now be selectable. If these toolbars are not visible they can be turned ‘on’ via the *SetUp / Toolbar Visibility* menu, look for the two *3D Template Builder* entries. Remember that to find out the function of an icon hover over it for the toolbar bubble help.



Template Builder Toolbar - Part Icons



Template Builder Toolbar - Action Icons

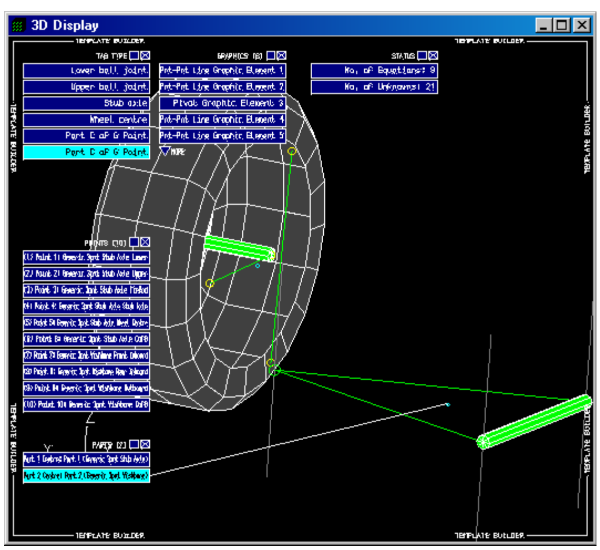
We need to add four parts: an upright, the upper wishbone, the lower wishbone and a steering arm. Start by adding a three-point stub axle to the model. With the default icons this is the 5th icon in the 'Parts' toolbar, alternative use the main menu *Edit / Add to Model / Add Part / 3 Point Stub Axle.*

- ***Add a three point stub axle to the template, Edit / Add to Model / 3 Point Stub Axle.***

This adds one part, six points, five tag types and three graphics. Note that when a part is added all the points added with the part are placed into a temporary group, such that if you 'drag' one of the points the others are dragged by the same amount. As usual to cancel the group, such that points can be moved individually, select *Edit / Groups / Cancel* or select the equivalent icon.

Now add the first wishbone (lower) and drag it down in Z to a suitable position. Either use the 3 Point Wishbone toolbar icon, (third icon on the parts toolbar). Alternatively use the main menu *Edit / Add to Model / Add Part / 3 Point Wishbone.*

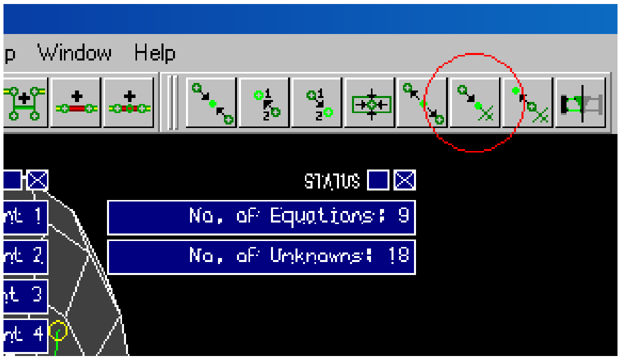
- ***Add a three wishbone to the template, Edit / Add to Model / 3 Point Wishbone, and drag it down in Z to a suitable position.***



Two parts added – wishbone dragged down as a group

We will now use some of the template builder 'actions' to create some connections. First connect the two inboard points to ground. To do this we select the appropriate toolbar icon or the main menu *Edit / Template Builder Actions / Join Point to Ground (at Point)*. Thus select the icon then select the first of the two wishbone inboard points, its graphic will change to indicate that it is connected to ground. Repeat the icon selection and then pick the second of the two wishbone inboard points. Note that as you make the connection the number of unknowns reduces by 3.

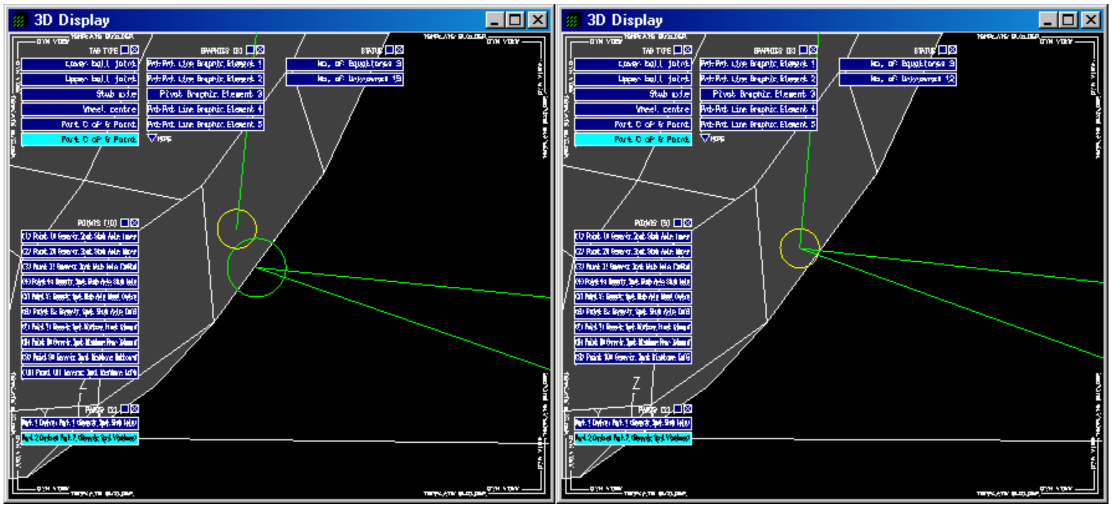
➢ ***Connect the two wishbone inboard pivot points to ground, Edit / Template Builder Actions / Join Point to Ground (at Point). Repeat for both inboard points.***



Join Point to Ground (at Point) - Template builder icon

We now need to join the outer ball joint of the wishbone to the lower ball joint of the stub axle. There are three methods of joining two parts together, the mean position of the two points, at the position of the first point or at the position of the second point. For this example we will joint the parts at the mean of the two position.

➢ ***Connect the wishbone outboard ball joint to the lower ball joint of the stub axle, Edit / Template Builder Actions / Join Part to Part (at mean of two points).***



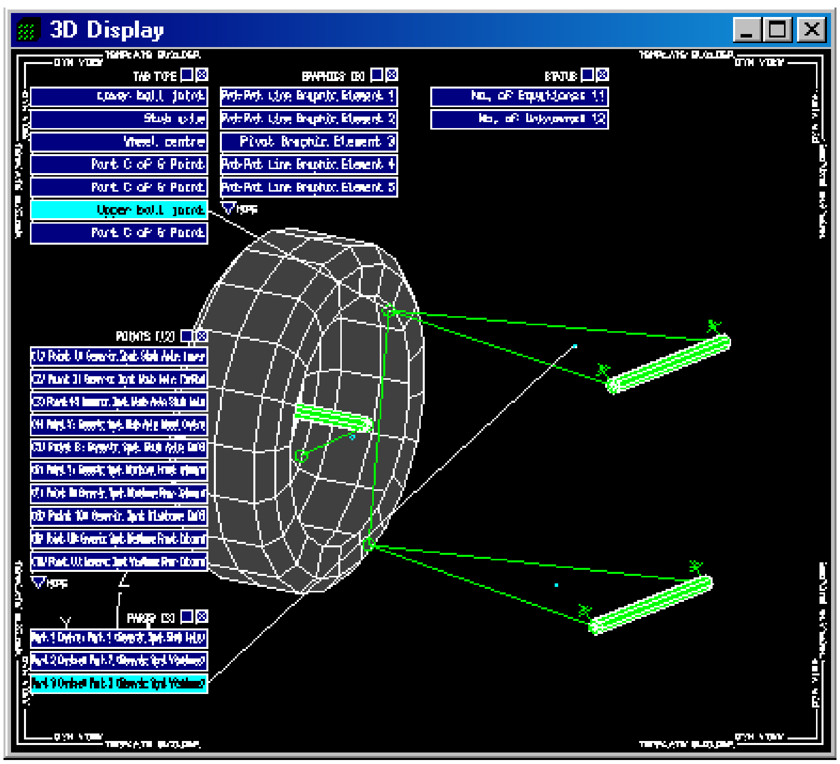
Wishbone Outer Ball Joint joined to upright – At Mean of two points

We now need to repeat the previous steps for the upper wishbone. So add a new 3 point wishbone, drag it up in Z to an appropriate position, connect the inboard points to ground and the out board point to the stub axle upper pivot.

- ***Add a three wishbone to the template, Edit / Add to Model / 3 Point Wishbone, and drag it up in Z to a suitable position.***

- ***Connect the two wishbone inboard pivot points to ground, Edit / Template Builder Actions / Join Point to Ground (at Point). Repeat for both inboard points.***

- ***Connect the wishbone outboard ball joint to the upper ball joint of the stub axle, Edit / Template Builder Actions / Join Part to Part (at mean of two points).***

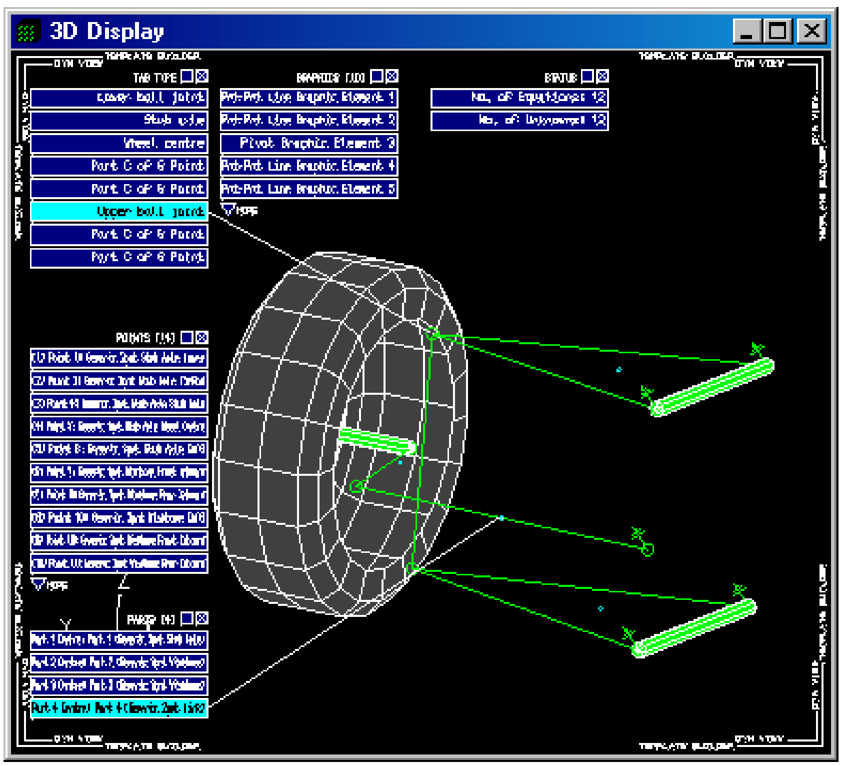


Upper Wishbone Added to Template and Joined to Ground and Stub Axle

Your template should look similar to that in the above screen shot and have 12 unknowns and 11 equations. We still need to add the steering tie rod check for additional 'Tag Types' and include the spring and damper.

To add the last part we will include a 2-point link to the template. Either use the second toolbar parts icon or the *Edit / Add to Model / Add Part / Two Point Link.* Drag it forward in X to a suitable position. Connect the inboard end to ground and then join the outboard end to the stub axle at the mean of outboard end and the stub axle steering point (i.e. last free point on the stub axle).

- ***Add a two point link to the template, Edit / Add to Model / 2 Point Link, and drag it forward in X to a suitable position.***
- ***Connect the inboard end of the link to ground, Edit / Template Builder Actions / Join Point to Ground (at Point).***
- ***Connect the link outboard ball joint to the steering point of the stub axle, Edit / Template Builder Actions / Join Part to Part (at mean of two points).***



Final Template prior to spring/damper being added

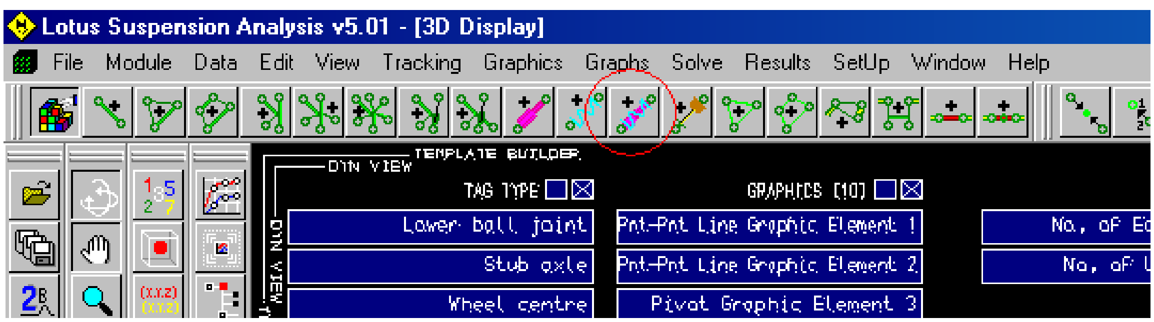
We could now use the template at this stage. The status indicates 12 unknowns and 12 equations, so it is balanced and thus solvable.

To retain the use of this new template we need to save it, either using one of the Template file options, or it could get saved just as part of a data file, (provided that the option *Setup / Include User Templates in Data Files* is enabled). More normally we would save this new template using one of the *File / Template File Options* sub menus. For our example we will save it as a single ‘custom’ template. This is a file that just contains this template and if we wanted to use it in future runs of the program it would first need to be read-in via the *File / Add Custom Templates from File…* menu option. Save the template to a custom file as ‘sample_template.dat’.

- ***Save the template as a custom template file, File / Template Builder File Options / Save Custom Template. Set file name as ‘sample_template.dat’***

If you wanted to load a saved custom template back into the template editor for further modifications you would, in the Template builder module, use the *File / New* menu item as though your where creating a new template, but use the option *Open – Saved Custom Template*. The saved template is read in to enable further changes to be made. The template builder module *File / New* options list also includes an option that allows you to select an existing standard template for modification, but rather than create this as a new template in the next free slot it modifies it in its current location. As with all user template modifications they must be saved using one of the available options if the changes are to be retained for future re-use.

Before we use this template we should add some additional components to increase the functionality. The most commonly added will be the spring and damper. We will add these as a common component, this allows us to add them in one go.



Add ‘Spring / Damper’ Toolbar Icon

To add the combined spring damper select the icon as indicated above, or the menu *Edit / Add to Model / Add Part / Add Spring-Damper*. Note that this template builder add part action is functionally different from that used in the interactive model editing option, *Edit / Add to Model / Spring 1 (pick two points)*, which as the menu suggests requires you to pick the two existing points for which the spring should be attached to. The menu option we are using is only available in the template builder mode as it adds the two points required to mark the upper and lower connections, but does not connect them with any particular part. This would thus not be valid in the normal interactive model editing. You could of course use the interactive model editing option in the template builder but you would first need to add the two points for picking.

We will use the *Edit / Add to Model / Add Part / Add Spring-Damper* menu option which places the spring/damper unit on the graphic. As a variation on the normal we will be connecting the spring and damper on to the stub axle rather than one of the wishbones. So drag the upper and lower points to suitable positions prior to making the connections.

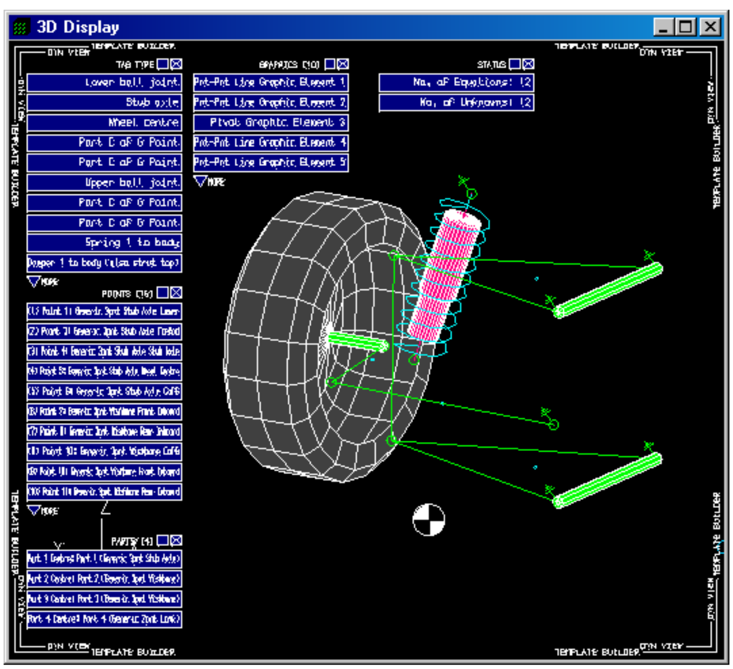
- ***Add the combined Spring/damper unit to the model, Edit / Add to Model / Add Part / Add Spring-Damper. Then drag the upper and lower connection points to suitable positions for connecting the lower point to the stub axle.***

The spring/damper upper point we can connect to ground in the same way that we did for our other 'ground' points, (remember you may prefer to think of 'ground' as the body since this is what is implied by the analytical term 'to ground').

- ***Connect the Spring/Damper upper point to 'Ground'. Edit / Template Builder Actions / Join Part to Ground (at Point)***

To make the connection of the spring/damper unit on to the stub axle we could use the method we have employed previously of joining two parts by selecting two points. In this instance we don't have a point on the stub axle that we could connect to, the existing points all being used to define joints with the other suspension parts. We could create a new point on the stub axle using one of the *Edit / Add to Model / Add Point / to Part…* menu options and then join this point to the lower spring/damper point with the *Join Part to Part* option, but there is a more convenient method available. We will make the connection with *Join Point to Part (at the Point)* menu. Thus we only need to select the part we wish to join to and the point we wish to join to it.

- ***Connect the Spring/Damper lower point to the stub axle. Edit / Template Builder Actions / Join Point to Part (at the Point)***



Template with spring/damper added

We now have a complete template that can be used for modelling. Before proceeding re-save the template as a custom template.

- ***Re-Save the template as a custom template file, File / Template File Options / Save Custom Template. Set file name as 'sample_template.dat'***

To use this as a model we can simply switch to one of the 3D modules such as 3D Bump. The display will change to indicate that we are no longer in the template mode and the model will now solve over the defined articulation range. Confirm this by displaying an x-y graph of the calculated camber angle.

- ***Change to the 3D Bump module and display a camber angle graph. Module / Shark / 3D Bump, Graphs / New-open***

Note that you can freely switch between the 3D-bump mode and the template builder module to make further changes to the template as required.

13.5 Building a New Template, Example 2

As a second example of using the interactive template builder we will create a Mcpherson Strut type suspension. This simple 4 part template requires only three actual 'add part' actions since the Strut/Upright adds two parts but is a single 'add part' action. Thus we need to add the following parts from the *Edit / Add to Model / Add Part* options; 3 Point Strut, 3 Point wishbone and 2 Point Link.

Once the parts have been added and positioned at appropriate locations, to complete the template we must carry out a number of steps. 1) We need to connect the strut top, wishbone inner points and the steering link inner points to ground. 2) Join the wishbone outer ball joint to the strut lower ball joint. 3) Joint the steering link outer ball joint to the strut steering arm.

- ***Change to template builder mode, Module / Shark / 3D Template Builder***

- ***Create new template, File / New, select 'Create – Steerable Suspension', give this new template a suitable label, "Mcpherson strut example".***

- ***Add a 3 Point Strut Part, Edit / Add to Model / Add Part / 3 Point Strut***

- ***Add a 3 point wishbone and drag it down in Z to an appropriate position, Edit / Add to Model / Add Part / 3 Point Wishbone.***

- ***Add a 2 point link and drag it forward in X to an appropriate position, Edit / Add to Model / Add Part / 2 Point Link.***

- ***Connect the Strut top to Ground, Edit / Template Builder Actions / Join Part to Ground (at Point).***

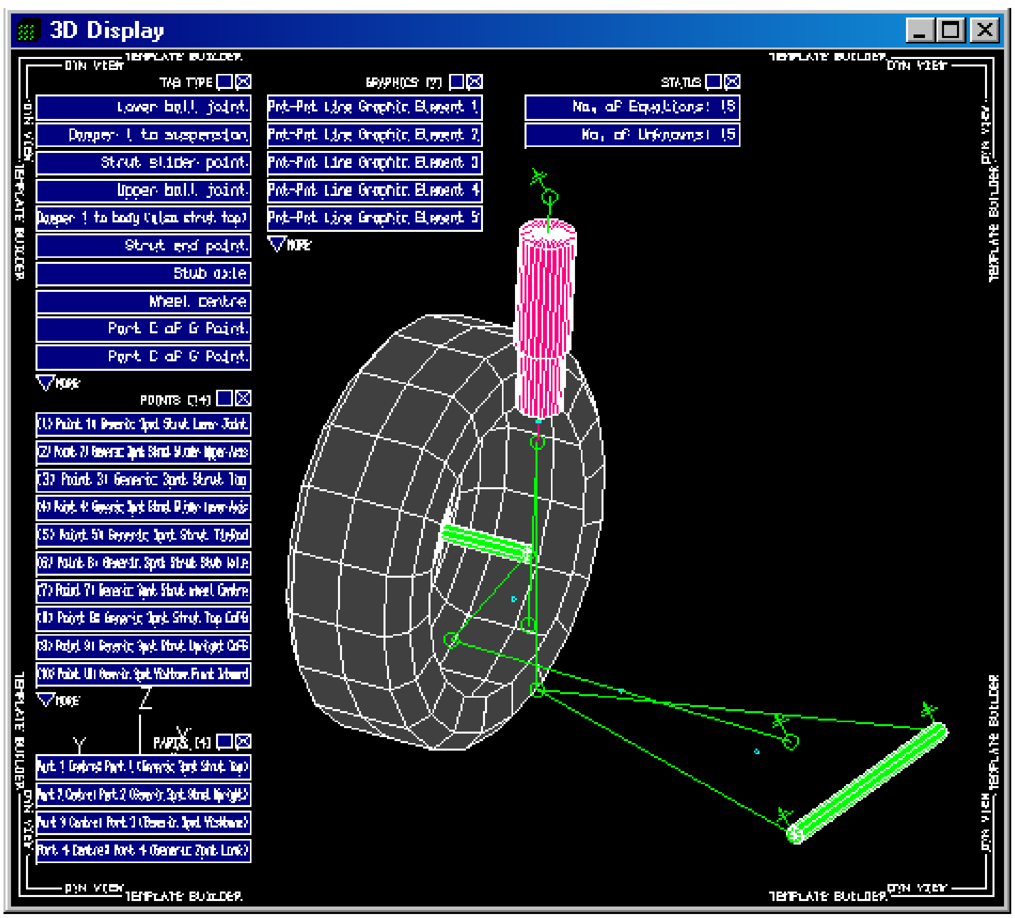
- ***Connect the Wishbone Inner pivots to Ground, Edit / Template Builder Actions / Join Part to Ground (at Point).***

- ***Connect the Steering link inner pivot to Ground, Edit / Template Builder Actions / Join Part to Ground (at Point).***

- ***Joint the Wishbone outer ball joint to the lower ball joint of the Strut, Edit / Template Builder Actions / Join Part to Part (at mean of two points).***

- ***Joint the Steering link outer ball joint to the steering arm ball joint of the Strut, Edit / Template Builder Actions / Join Part to Part (at mean of two points).***

You should now have a valid template. As always check the status, you should have 15 unknowns and 15 equations.

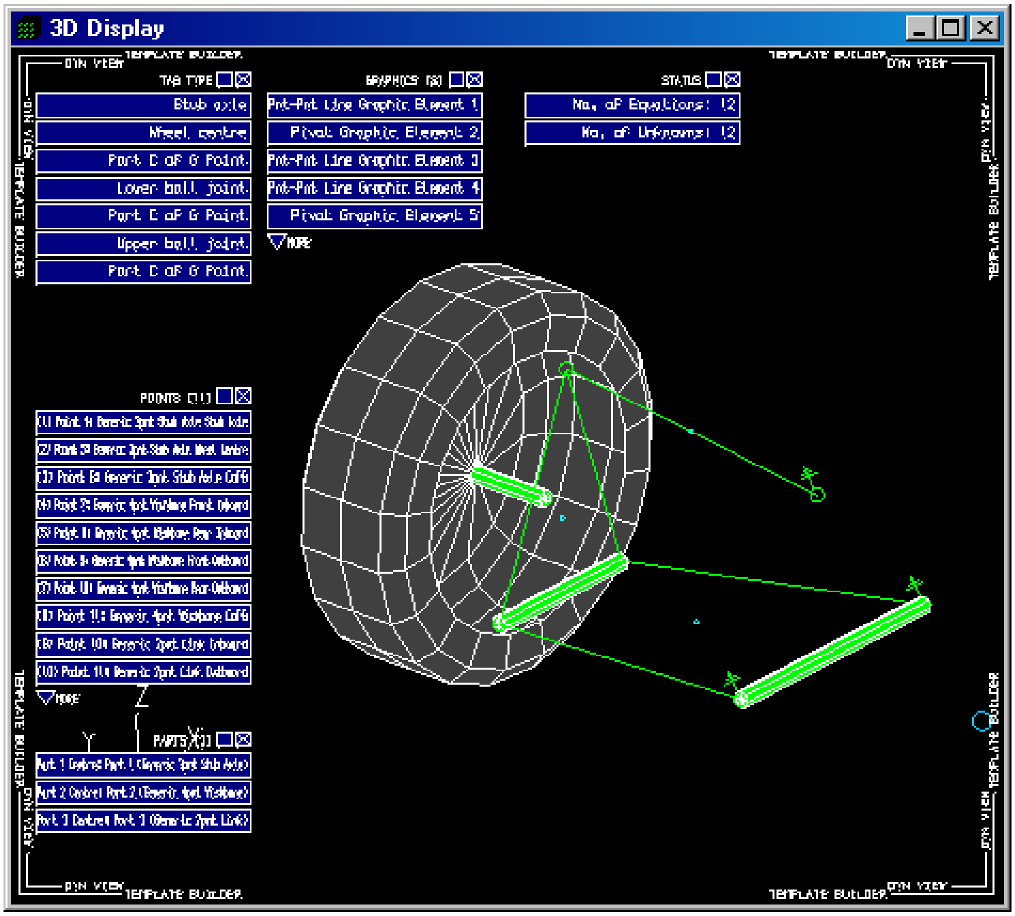


McPherson Strut Template – No Spring in template

Try adding the spring to complete this template.

13.6 Exercise: Non-Steerable Single Upper Link Suspension

For an exercise try and produce the template shown below. It is a non-steerable suspension with a single upper link and extended lower wishbone.

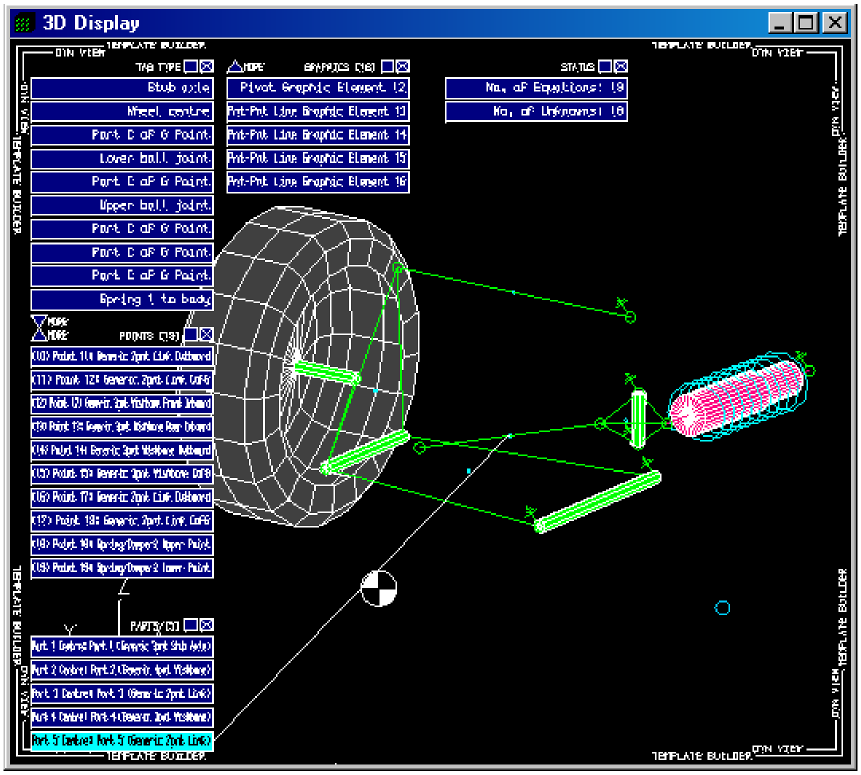


Non-Steerable template exercise

Hints: This 11 point 3 part template uses the following parts, 3-point stub axle, 4-point wishbone and a 2-point link. The normal steering link connection on the stub axle is re-positioned by connecting it to the other lower wishbone outer ball joint.

You should end up with a 12 equation / 12 unknown template.

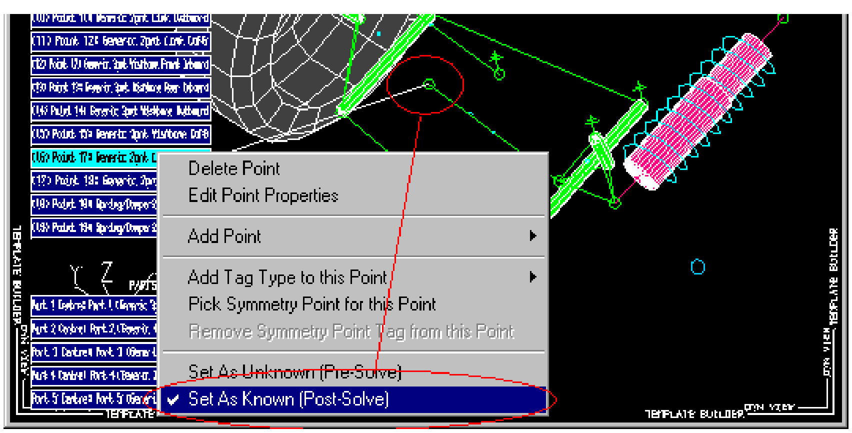
As an extension to this exercise try and add the spring/damper unit operated on by a push rod from the lower wishbone. You will need to add two more parts, a three-point wishbone and a two-point link as well as the spring/damper unit.



Push Rod Spring-Damper Modification

These changes produce a new template with 18 equations and 18 unknowns. Since the additional push-rod mechanism does not control the kinematic motion of the wheel as it is articulated, the extra points we have added can be post-solved. Post solving can be used to remove unknowns from the main solution process, thus leading to a quicker solution. We can make this change direct from the interactive template editor.

To return to our previous 12 unknowns we need to switch the two unknown points (since each unknown point adds three unknowns) from 'unknown' to 'known' via the relevant right mouse menu. The two points to switch are the ends of the push-rod.



Changing the 'unknown' Type

Having made the changes to solver type you should be back to 12 unknowns and 12 equations. Test your template in the 3D bump module and the Roll module.

14 User Templates (1)

14.1 Overview

This chapter describes the structure of the model Templates used by Lotus Suspension Analysis to define and solve each suspension type. This then details how users can create their own templates, and make them available as extra ‘default’ templates, or share them with other users.

This chapter contains the following sections:

14.2 Description of User Templates

All suspension types used in LSA (Shark) are defined using a template structure. This structure uses a combination of data sets separated up into 'Parts', 'Points', 'Settings' and 'Graphics'. A significant number of the most common suspension types are 'hard coded' into the application as defaults, but the user can modify these 'defaults' or add their own through the template-editing tool, (*File / Edit Templates)*.

Each template is given a location number in the templates' database (for example the default double wishbone suspension is stored in location number 1). The template is also given a descriptive string to make recognition of each type easier.

For a server installation, on program start up the template file "_User_Template.dat" is searched for in the <install> folder. If it is found any template definitions identified within it are loaded and will either overwrite an existing default template (if the template number is already used by a default entry) or fill an empty slot number.

If a Database Folder is defined, on program start up the template file "_User_Template.dat" is searched for in the <database> folder. If it is found any template definitions identified within it are loaded and will either overwrite an existing loaded template data.

Template definitions can be modified by individual users, thus individuals may have completely different definitions using the same template slot number. To provide a robust definition method, the specific template definition can be optionally included in the models data file. By definition this implies that when a model file is read in it can redefine a template specification, (and also that any subsequent use of the same template number will be similarly affected until the program is restarted or the templates reset).

Extra 'custom' templates can be loaded at any time. Custom templates would normally have be a pre-saved set (or single) template, that may be required for occasional use but don't warrant being added to the automatically loaded "_User_Templates.dat" file.

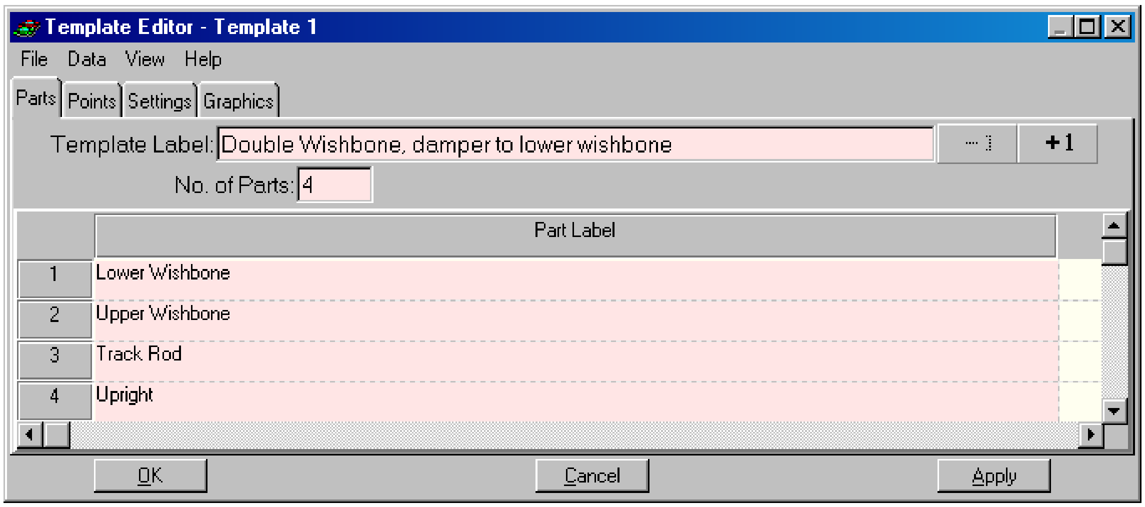
To allow users to return to a set of known template definitions, menu options are provided that will re-set the template definitions to the hard coded ones only, or the hard coded ones plus the system user templates in the "_User_Templates.dat" file, (if it exists).

Data values in the various data sets within the template editor are grouped into four further categories. These are 'compulsory', level 1, level 2 and level 3. Each category is identified by a colour. These are used to illustrate which values **must** be defined by the user, and which can be filled via the 'auto-fill' routines. The auto-fills can be used to assist in identify missing template values, through a series of menu options.

14.3 The ‘Parts’ Data-Set

The Parts data-set defines how many parts are in the suspension corner (or axle) model. Parts include wishbones, tie rods, uprights and rockers. There is no need to add a part for the vehicle body (this is taken as part number –1), spring/dampers or the tyre. Each of these is implicitly included later by ‘Point’ definition.

The only data field for a part is an identifying label. This label is used in the other data sets to ease part recognition/selection.



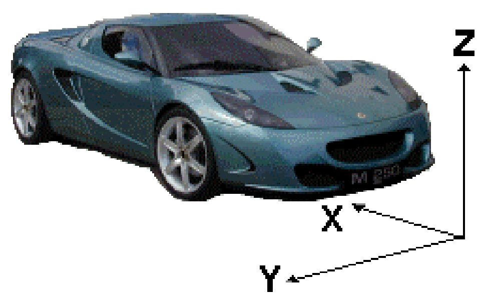
’Parts’ Data Set editing – Default template 1 shown

Part Numbers and Part Labels for the defined model can be displayed on the model using the visibility switches, *Graphics / Part Nos* and *Graphics / Part Labels.*

All data values in the ‘Parts’ data set are in the ‘Compulsory’ category.

14.4 The 'Points' Data Set

The Points data set defines how many 'hard points' are included in the suspension corner/axle model. The data fields for each point are a Point short label/No., a descriptive label (again used to aid identification/selection), and the default x, y and z co-ordinate values. The default co-ordinate values are those that are applied to a 'new' model using the particular template. The 'SHARK' co-ordinate system is a right handed system with the Y-axis across the car track, the origin of which is assumed to be on the vehicle centre line, and the +ve direction being towards the offside suspension (Right hand Corner sitting in car). The X-axis is along the vehicle wheelbase, normally with the origin in front of the vehicle with the +ve direction towards the rear. The Z-axis is the vertical height, which for the 3D mode can be at any height position. The +ve direction is taken as upwards.



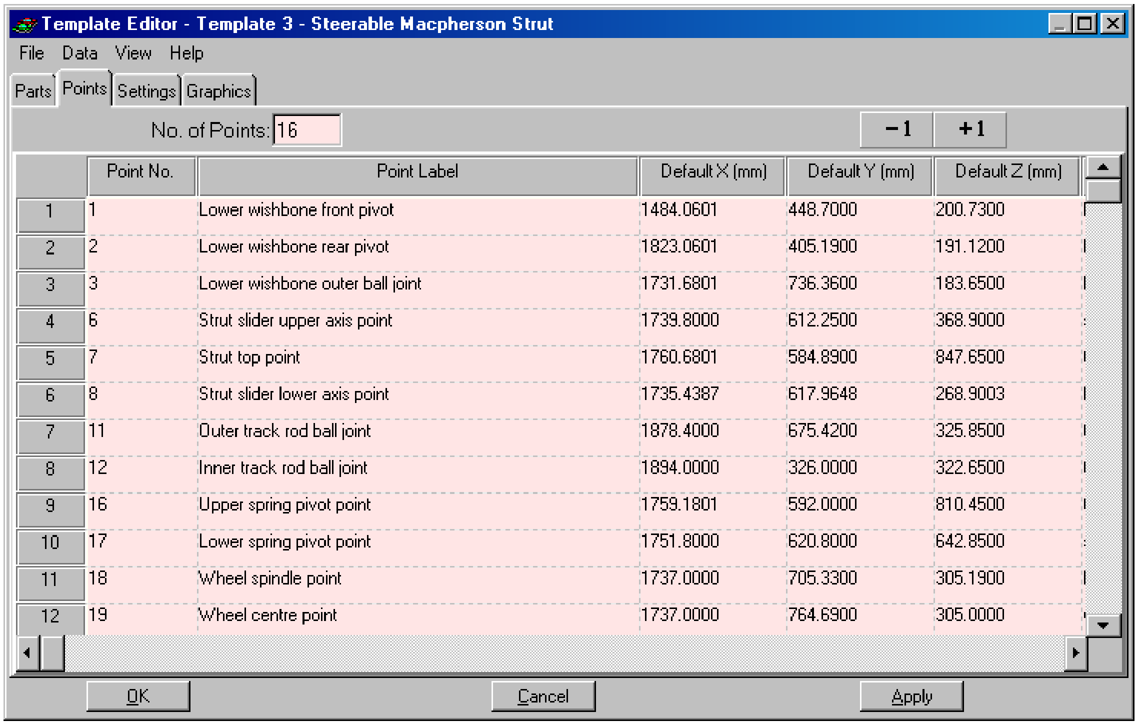
'Shark' Coordinate System

Connections between parts are made at points (i.e. wishbone ball joint). These connection points should only be entered once in the list of points (the association to two parts is made later). Remember to add points for springs and damper connection points. You need to include points for both the suspension attachment point of the spring (or damper) and the attachment to the body.

If you have a combined spring/damper unit that employs common attachment points, you don't need to define the spring and damper points separately. We can associate both the spring and damper to the same connection point later.

For strut type suspensions, three points define the strut: the strut top mount point, and two further points on the slider axis. These later two points would normally be placed at the upper and lower bearing centres to ensure the correct moments/forces. These three points will always be forced to lie along a single 3D line and thus get updated/corrected by the application.

Two points are always used to define the wheel spindle axis. The first is the wheel centre and the second is a point on the wheels rotational axis. This second spindle point can be either inboard or outboard of the wheel centre, although it is usually placed inboard of the wheel centre since its is the normal convention to draw between these two, a graphical element representing the spindle shaft.



'Points' Data Set editing – Default template 3 shown

Although we have given each point a descriptive label, they can also be referenced by the Point No, which is actually a 'short' string. This short string is limited to a maximum of 8 characters. This 'short' string can be numbers, characters or a mix. If numbers, it doesn't need to be the same as its numerical position in the list. The use of this short label as a number is somewhat historical relating back to earlier version of the program, when points were only referenced by number, rather than by label. The visibility of both 'number' and 'label' on the model is set using the visibility switches, *Graphics / Point Nos.* and *Graphics / Point Labels.*

You do **not** define a point for the tyre contact point. This is calculated automatically from the tyre properties and the wheel spindle points. It can be added to the template as a 'calculated' point, should its visibility be required.

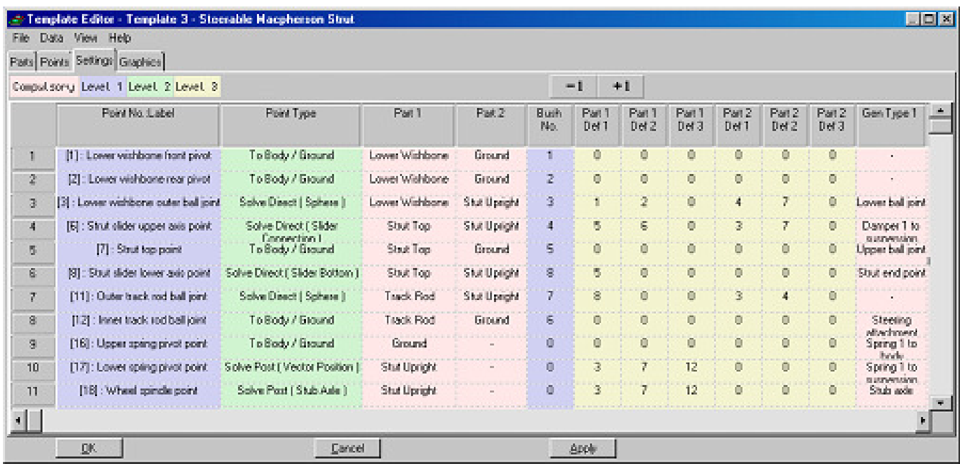
All data values in the 'Points' data set are in the 'Compulsory' category.

The 'Points' data set does have some optional identification labels that are used for the import/export options to some specific file formats. These can be seen in the right hand columns of the spreadsheet.

Points can be defined in a template as a function of another point's position. In the simplest form, it can be made the same by replacing the value with the string [P1X]. This will then always use the x co-ordinate from point 1 for this defined points position (need not also be x). This string can be extended to include simple maths equations, i.e. [P1X+10.0], or involve more than one point, i.e. [(P1X+P2X)/2.0]. These relationships are held as you drag points in the model.

14.5 The ‘Settings’ Data Set

The ‘Settings’ data set has a row entered for each point defined in the ‘Points’ data set. It is through the Settings data set that we identify the association of points to a part (or Parts), the solution method to solve for each point, a bush No., and any special function that the point has. Some of these data fields are compulsory others may be filled through the ‘auto-fill’ options. It should be noted that the ‘auto-fill’ routines are not foolproof, and thus should be applied with care. Auto-fill routines should only be used once all the compulsory data fields have been filled.

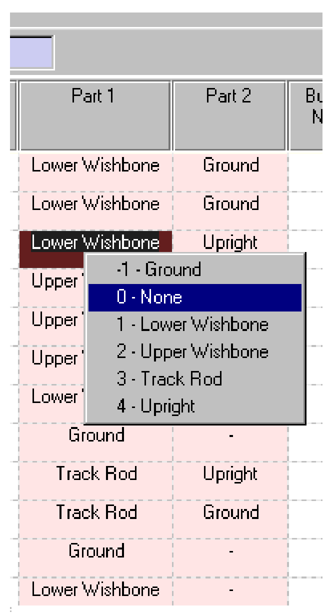


	Point No. Label	Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3	Gen Type 1
1	[1] : Lower wishbone front pivot	To Body / Ground	Lower Wishbone	Ground	1	0	0	0	0	0	0	-
2	[2] : Lower wishbone rear pivot	To Body / Ground	Lower Wishbone	Ground	2	0	0	0	0	0	0	-
3	[3] : Lower wishbone outer ball joint	Solve Direct (Sphere)	Lower Wishbone	Strut Upright	3	1	2	0	4	7	0	Lower ball joint
4	[6] : Strut slider upper axis point	Solve Direct (Slider Connection)	Strut Top	Strut Upright	4	5	6	0	3	7	0	Damper 1 to suspension
5	[7] : Strut top point	To Body / Ground	Strut Top	Ground	5	0	0	0	0	0	0	Upper ball joint
6	[8] : Strut slider lower axis point	Solve Direct (Slider Bottom)	Strut Top	Strut Upright	8	5	0	0	0	0	0	Strut end point
7	[11] : Outer track rod ball joint	Solve Direct (Sphere)	Track Rod	Strut Upright	7	8	0	0	3	4	0	-
8	[12] : Inner track rod ball joint	To Body / Ground	Track Rod	Ground	6	0	0	0	0	0	0	Steering attachment
9	[16] : Upper spring pivot point	To Body / Ground	Ground	-	0	0	0	0	0	0	0	Spring 1 to body
10	[17] : Lower spring pivot point	Solve Post (Vector Position)	Strut Upright	-	0	3	7	12	0	0	0	Spring 1 to suspension
11	[18] : Wheel spindle point	Solve Post (Stub Axle)	Strut Upright	-	0	3	7	12	0	0	0	Stub axle

’Settings’ Data Set editing – Default template 1 shown

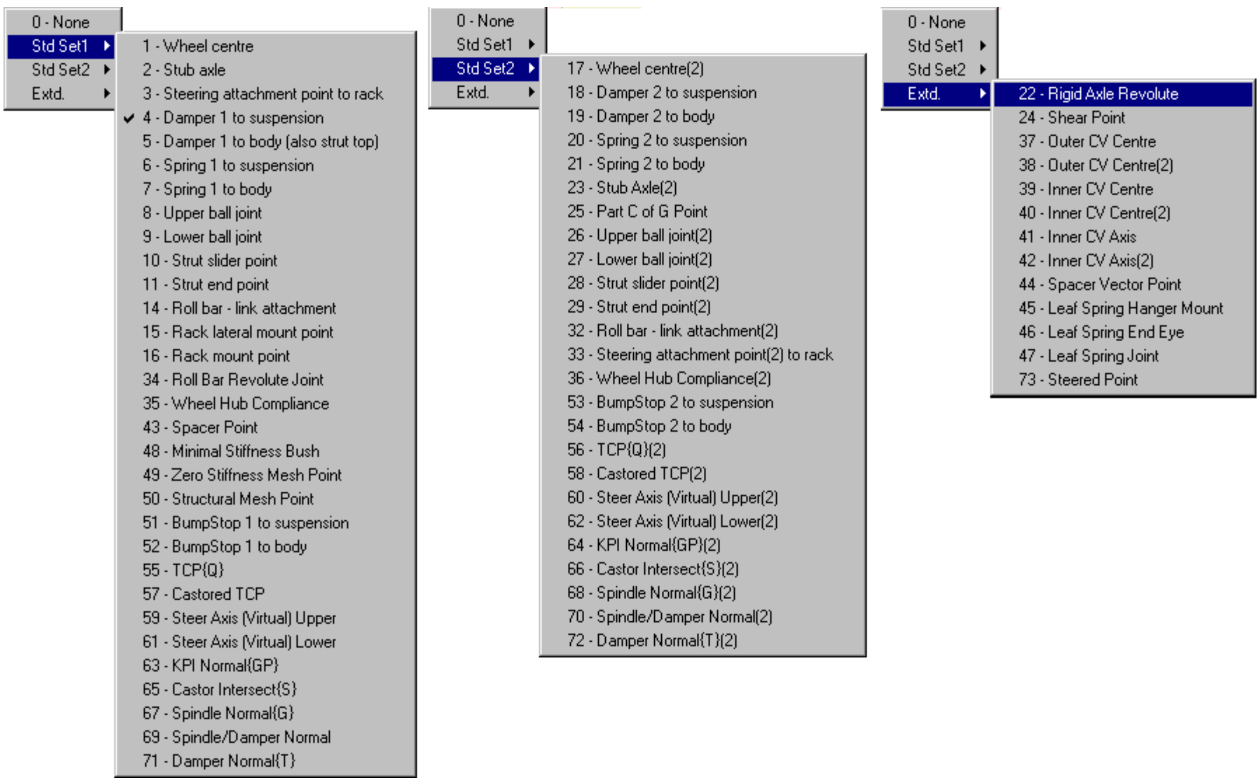
14.6 ‘Settings’ - Parts

Columns 3 (‘part 1’) and 4 (‘part 2’) define the parts that a point is associated with. These are compulsory data fields. A point can at most be associated with two parts (this would be a connection between parts and requires a bush), and a minimum of one. In the case of a point associated to one part, always fill column 3 in preference to column 4. Note that additional part options, ‘Ground’ and ‘None’ are added to the available parts list.



14.7 ‘Settings’ - General Types

Column 12 (‘Gen Type 1’) which, whilst it is a compulsory data field, it can be left blank (the equivalent of ‘none’ from the menu options). This column defines whether a point performs one of the ‘General Types’ functions. These functions identify to the solver and the template builder that a point has specific properties. Column 13 (‘Gen Type 2’) is a repeat of column 12 and is used if a point has more than one General Type classification. There are currently 98 different ‘General Types’. The major ones are discussed below.



General Types Menu

0 = None: Identifies that the point has no specific type. This could include points such as wishbone attachment points to ground.

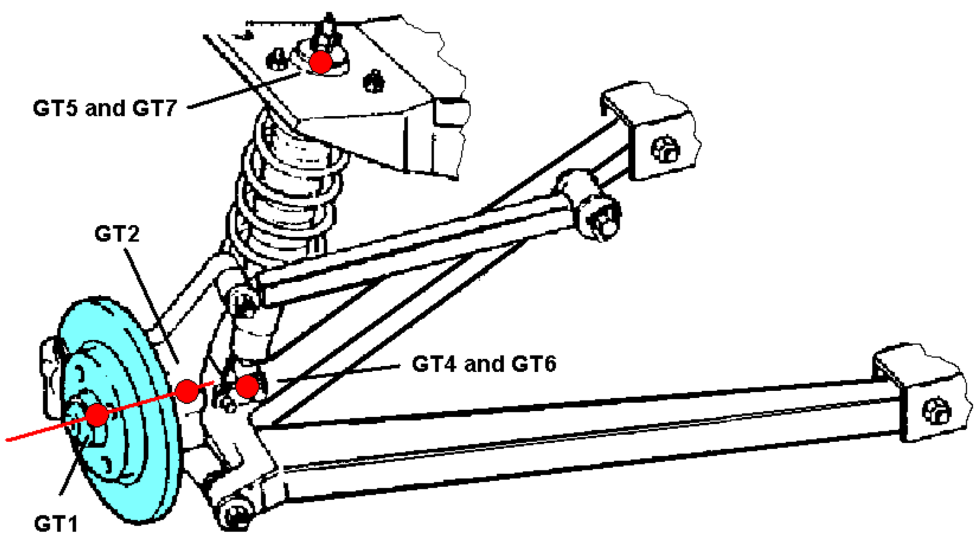
1 = Wheel Centre: One point must always be identified as ‘wheel centre’. Identifies that the point is the wheel centre.

2 = Stub Axle: One point must always be identified as ‘stub axle’. Identifies that the point is the second point on the wheel spindle which together with the ‘wheel centre’ define the stub axle and hence all wheel related parameters.

3 = Steering Attachment Point to Rack: This optional point identifies that the template has a point at which steering input can be applied, (either rack or steering box). It is also used to determine whether the template will appear in the list of available ‘Front’ suspensions on the *File/New* option, (all templates appear in the list of ‘Rear; suspensions irrespective of this general type).

4 = Damper 1 to Suspension: This optional point identifies that the specified point is the position in the template at which the damper is attached to the moving suspension. This is used to determine damper travel, damper ratio etc. To work correctly it will need general type 5 to also be defined. This would also be used to identify the point at the upper bearing position on the slider of a McPherson-strut suspension. The reference to ‘Damper 1’ in the description is a recognition for potential future expansion to include multiple damper templates.

5 = Damper 1 to Body (also Strut Top): This optional point identifies that the specified point is the position in the template at which the damper is attached to the body. It is used in conjunction with general type 4 above. This would also be used to identify the point at the top of the slider for a McPherson-strut suspension.

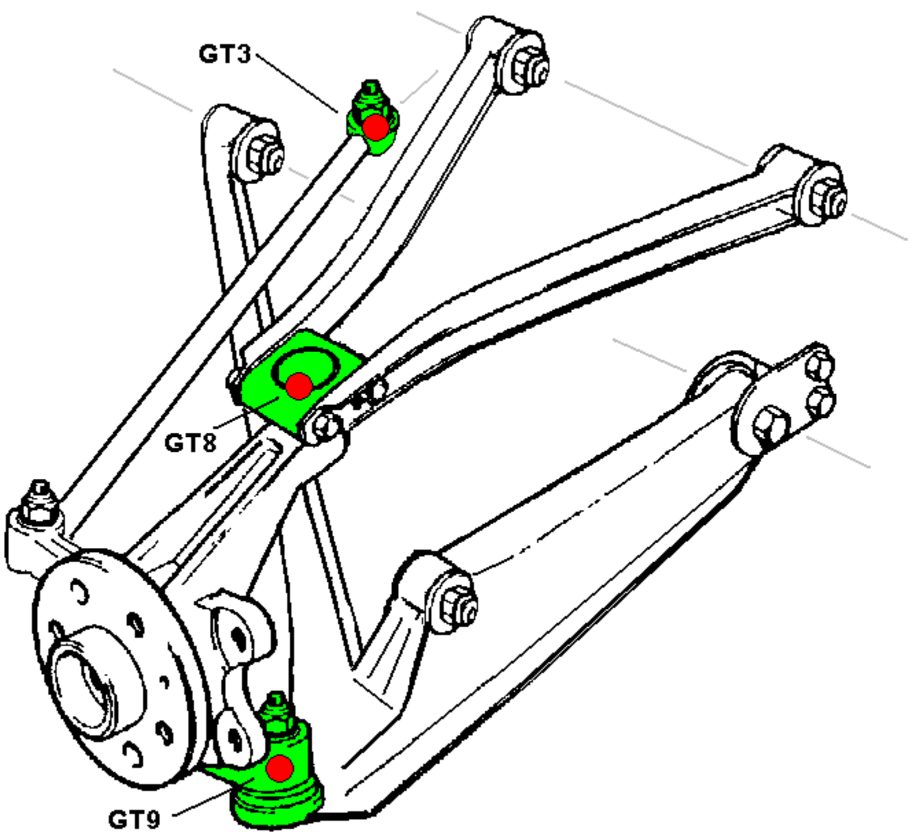


Example Rear Suspension – Showing General Types 1,2,4,5,6 and 7

6 = Spring 1 to Suspension: This compulsory point identifies that the specified point is the position in the template at which the spring is attached to the moving suspension. This is used to apply suspension spring forces to the template and determine spring travel, spring ratio etc. To work correctly it will need general type 7 to also be defined. The reference to ‘Spring 1’ in the description is a recognition for potential future expansion to include multiple spring templates.

7 = Spring 1 to Body: This optional point identifies that the specified point is the position in the template at which the spring is attached to the body. It is used in conjunction with general type 6 above.

8 = Upper Ball Joint: This optional point identifies that the specified point is the upper ball joint on the steering axis. Whilst this is optional if it exists and can be identified this will improve the calculation speed/accuracy. Together with general type 9 it defines the steering axis. If a steering axis can not be defined but a steering attachment point has been identified, (see default template No. 20), then additional calculations are used involving a small steering perturbation to identify the 'effective' steering axis. This normally involves suspension types that have twin outer ball joints rather than a single one.

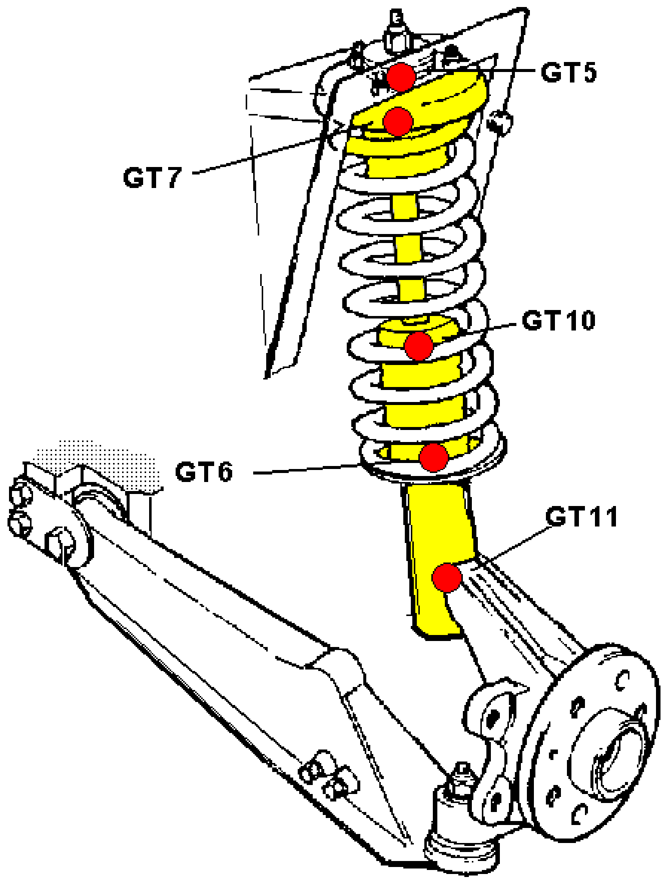


Example Front Suspension – Showing General Types 3,8 and 9

9 = Lower Ball Joint: This optional point identifies that the specified point is the lower ball joint on the steering axis, (see comments for type 8 above).

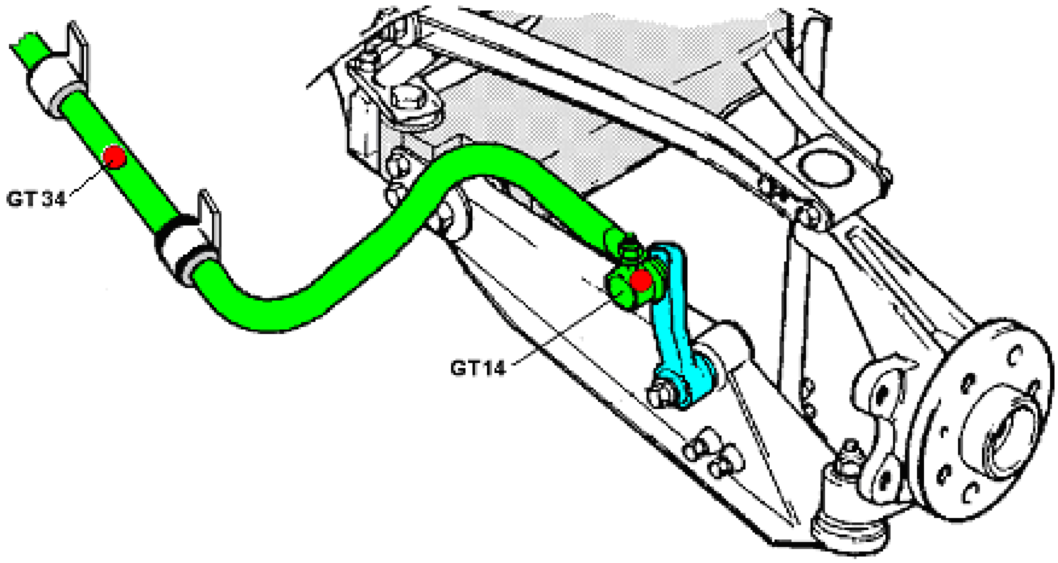
10 = Strut Slider Point: This optional point identifies that the template is a strut type suspension and the specified point is the strut body upper bearing position. This should not be confused with the strut top mount, (which is identified via general type 5). This point moves with the strut body and is used as the sliding connection between the slider and the strut body, (see also type 11 below).

11 = Strut End Point: This optional point identifies that the template is a strut type suspension and the specified point is the strut slider end bearing. This point moves with the strut slider and is used as the sliding connection between the slider and the strut body.



Example McPherson Strut – Showing General Types 5,6,7,10 and 11

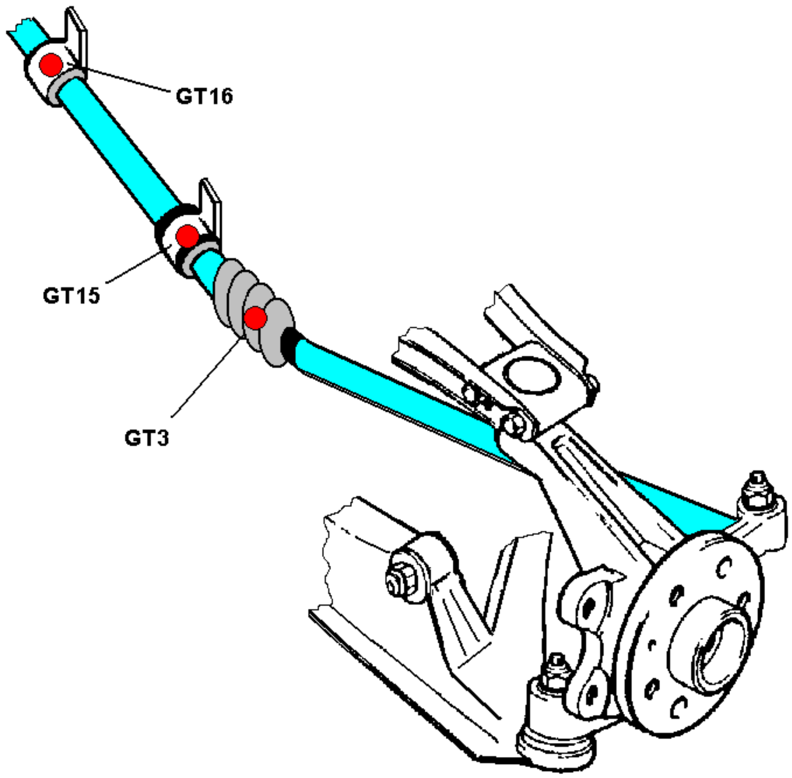
14 = Roll Bar – Link attachment: This optional point identifies the point where the roll bar drop link is attached to the anti-roll bar. This is used to identify the amount of twist in the bar and hence the force applied to the suspension.



Example Front Suspension – General Types 14 and 34

15 = Rack Lateral Mount point: This optional point identifies the point as being a the rack attachment point that also takes the axial reaction load. User defined bush properties should be defined accordingly.

16 = Rack Mount Point: This optional point identifies the point as being the second rack attachment point. No axial load is carried by this point when using the default bush settings.



Example Front suspension – General Types 3, 15 and 16

17 = Wheel Centre(2): This optional point identifies the point as being the second wheel centre for rigid axle and full axle templates.

18 = Damper 2 to suspension: This optional point identifies the point as being the second dampers' attachment point to the suspension. It may be a second damper on a single corner model or the opposite side on a full axle model.

19 = Damper 2 to body: This optional point identifies the point as being the second dampers' attachment point to the body. As for point 18 above it may be one of two cases.

20 = Spring 2 to suspension: This optional point identifies the point as being the second springs' attachment point to the suspension. It may be a second spring on a single corner model or the opposite side on a full axle model.

21 = Spring 2 to body: This optional point identifies the point as being the second springs' attachment point to the body. As for point 20 above it may be one of two cases.

22 = Rigid axle revolute: This optional point identifies the point as being a revolute joint at the centre of a rigid axle. It is required for rigid axle templates to enable them to pre-solve in kinematic mode when in roll mode.

23 = Stub Axle(2): This optional point identifies the point as being the second wheel stub axle point. This is needed for full axle models.

24 = Shear Point: Used just for twist beam suspensions to identify the different pivot point position used in bump and roll. (Optional).

25 = Part C of G Point: Used to identify a point as being the C of G point for its primary part. It is normal for this point to not be used except as the C of G point, i.e. no involved in any joints. (Optional).

26 = Upper Ball Joint(2): Identifies a point as being the upper ball joint for the steering axis on full axle templates only. This must be a connection between two parts to conform with the concept of a steering axis. It is an optional setting in that if it (and the lower ball joint) are not defined the steering axis is determined via a small perturbation of the steering input mechanism. If it can be defined it will lead to faster solution times than the small perturbation method. (Optional).

27 = Lower Ball Joint (2): Identifies a point as being the lower ball joint for the steering axis on full axle templates only. This must be a connection between two parts to conform with the concept of a steering axis. It is an optional setting in that if it (and the upper ball joint) are not defined the steering axis is determined via a small perturbation of the steering input mechanism. If it can be defined it will lead to faster solution times than the small perturbation method. (Optional).

28 = Strut Slider Point(2): Sets the point for a Macpherson strut suspension type that is considered to be the location of the top bush for the strut for full axle templates only, (attached to the strut body). (Required for Struts).

29 = Strut End Point(2): Sets the point for a Macpherson strut suspension type that is considered to be the location of the strut lower bush for full axle templates only, (attached to the strut slider). (Required for Struts).

32 = Roll Bar – Link Attachment(2): Identifies the point as being the second connection between the roll bar drop link and the suspension. (Optional). Roll bars can only be added to full axle templates so a template must have both this and point 14 defined.

33 = Steering Attachment Point(2) to Rack: T Identifies which suspension link end point should be used for the steering input from the rack or steering box for the second end in a full axle model only. See also point 3 above. This point should be the inboard end of the track rod, i.e. link point connected to body or rack. (Optional). For a compliant rack to be added to the model this point must be defined together with point 3 above.

34 = Roll Bar, Revolute Joint: Identifies the point as being the centre point of a two part roll bar. In kinematic mode this is treated as a simple revolute allowing roll motion. In compliant mode the roll bar stiffness is applied to this point to simulate the effect of the roll bar stiffness. (Optional). Roll bars can only be added to full axle templates so a template must have this point and points 14 and 32 defined.

35 = Wheel Hub Compliance: Identifies the point as representing the hub compliance. This forms the connection between the upright and the hub/wheel. The solver uses this to apply suitable default stiffness values to the associated bush when the user as left it as ‘rigid’. This is necessary since a rigid has no rotational stiffness and would lead to unconstrained rotations of the hub.

36 = Wheel Hub Compliance (2): For full axle models identifies the point as representing the hub compliance for the other side. This is used by the solver in an identical manner to general type 35 above.

37 = Outer CV Centre: Tags the point in the template as being the Outer CV joint centre. This is used in conjunction with general types 39 and 41 to fully define the centre positions and orientations of the drive shaft. In most instances where the drive shaft has been added using the convenience menu the general type 37 is assigned to the stub axle point, (general type 2). The wheel centre is also used with this point to define the axis orientation of the output shaft.

38 = Outer CV Centre (2): Identical function as general type 37 above, but for the opposite side with full axle models.

39 = Inner CV Centre: Tags the point in the template as being the Inner CV joint centre. This is used in conjunction with general types 37 and 41 to fully define the centre positions and orientations of the drive shaft.

40 = Inner CV Centre (2): Identical function as general type 39 above, but for the opposite side with full axle models.

41 = Inner CV Axis: Tags the point in the template as being a point on the Inner CV joint axis. The axis uses this point and the inner joint centre to define the axis orientation of the input shaft. This is used in conjunction with general types 37 and 39 to fully define the centre positions and orientations of the drive shaft.

42 = Inner CV Axis (2): Identical function as general type 41 above, but for the opposite side with full axle models.

43 = Spacer Point: Identifies the point as being associated with a spacer. This enables certain solver functionality for the point including local solving by length and checking for an associated spacer vector point. As with all general type allocations it also assists in filling the template values when users use the ‘auto-fill’ options.

44 = Spacer Vector Point: Identifies the point as being the vector point for a spacer. This is used to identify the direction of the spacer’s ‘length’ vector for any subsequent changes in the spacer length applied to that spacer.

45 = Leaf Spring Hanger Mount: Together with general types 46 and 47 this point identifies and defines the equivalent leaf spring model. It is used by the solver to apply suitable default stiffness values when in compliant mode and when the bush properties are undefined (i.e. set as rigid). This particular general type is for the connection between the spring hanger and the body/ground.

46 = Leaf Spring End Eye: Together with general types 45 and 47 this point identifies and defines the equivalent leaf spring model. It is used by the solver to apply suitable default stiffness values when in compliant mode and when the bush properties are undefined (i.e. set as rigid). This particular general type is used on either end of the spring, so the connection between the spring eye and the spring hanger or the spring eye and the body/ground.

47 = Leaf Spring Joint: Together with general types 45 and 46 this point identifies and defines the equivalent leaf spring model. It is used by the solver to apply suitable default stiffness values when in compliant mode and when the bush properties are undefined (i.e. set as rigid). This particular general type is used on the joints within the spring, so the connection between the spring parts.

49 = Zero Stiffness Mesh Point: The meshing of a rigid part make use of two general types, 49 and 50 to deal with the graphical drawing and application of suitable default stiffness values. The zero stiffness point will have by default zero stiffness values applied to it by solver unless specifically defined by the user. They are zeroed to allow the 'structural mesh point' to control the meshed part flexibility.

50 = Structural Mesh Point: The meshing of a rigid part make use of two general types, 49 and 50 to deal with the graphical drawing and application of suitable default stiffness values. The Structural mesh point will have by default high translational and rotational stiffness values applied to it by solver unless specifically defined by the user.

51 = Bump stop 1 to Suspension: Identifies the point as being the suspension end of the bump stop 1.

52 = Bump stop 1 to body: Identifies the point as being the body end of the bump stop 1.

53 = Bump stop 2 to Suspension: Identifies the point as being the suspension end of the bump stop 2.

54 = Bump stop 2 to body: Identifies the point as being the body end of the bump stop 2.

55 to 72 = Specific Calculation Points: All these points are positions in the model that are classed as 'Calculated', that is their position is calculated directly from other points in the template are not therefore defined directly. They cannot be edited or manipulated but can be used in the template to locate graphics.

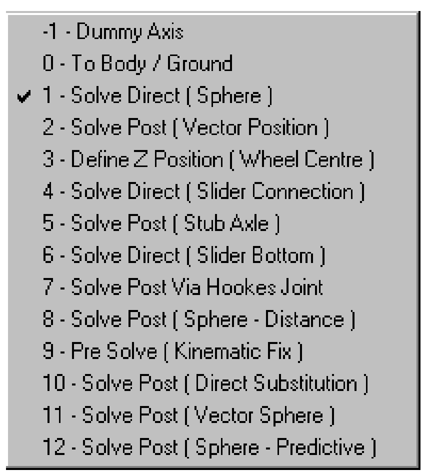
73 = Steered Point: A specific option that identifies a point attached to the body as one that should move with steering articulation. Alternative methods such as local co-ordinate systems or point definition strings would not achieve this behaviour as they only set the initial position.

14.8 ‘Settings’ - Bushes

The settings data-set is where we define the number of bushes in the model, and assign a bush number to each connection (column 5). A bush is added at any point that is a connection between two parts, and at a point that is the connection of a part to ground. Bush Nos. are primarily used for internal identification and need not be sequential.

14.9 ‘Settings’ – Point Solution Type

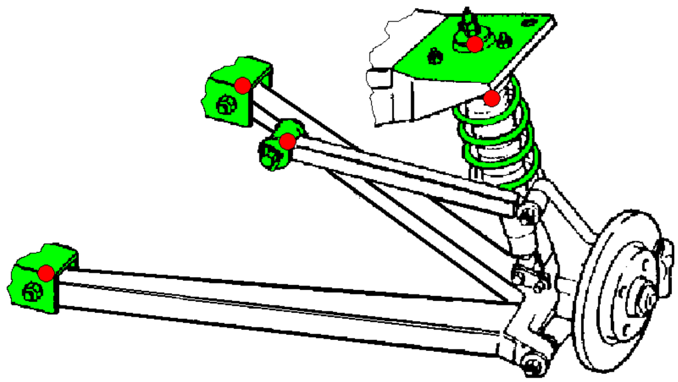
Column 2 (‘Point Type’) defines for each point the solution type used to solve its incremental position. This controls the number of equations, and the values used in solving the set of simultaneous equations. Some solution definitions can be post solved. That is determined after the set of simultaneous equations are solved. Fourteen different point solution types are available although by far the most common is the use of the ‘sphere’ equation. A description of each solution type is given below. Depending on the solution type selected the values required in columns 6 to 11 alter. These level 3 data fields identify the relevant points for the selected solution type.



Point ‘Type’ options list

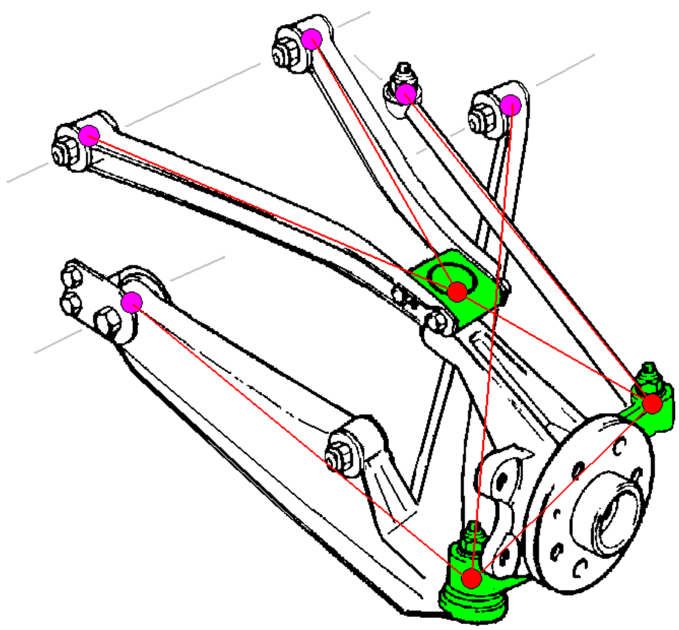
-1 = Dummy Axis Point: This option would not normally be used. It was added to deal with a specific issue identified for backward compatibility with default template type 12. It had in its original formulation an additional undefined point used to restrain the upright and enable the use of a steering knuckle. It is not envisaged that users would need to use this special option, as it is perfectly feasible to build this suspension template without recourse to this point solution type.

0 = To Body/Ground: This sets the point solution type as being pre-filled, (by bump, rebound or roll articulation) and hence no equations are added to the solver for this point. No arguments are required in columns 6 to 11 for this solution type. Points that use this type are usually those that are attached directly to the body.



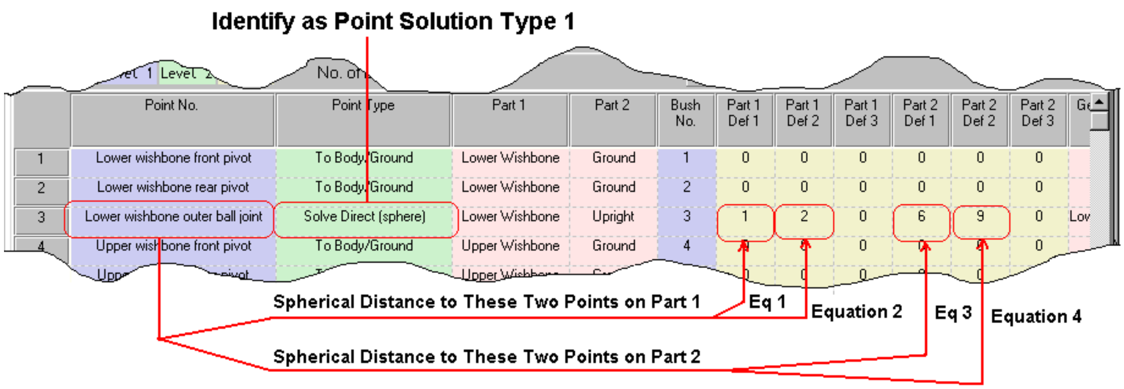
Typical Point Type 0's, To Body/Ground

1 = Solve Direct (sphere): This sets the point solution type to be based on the equation of a sphere. A spherical equation is added to the list of simultaneous equations for each point referenced in columns 6 to 11. The sphere equation controls the 3D distance between two points. The two points being the current rows point and the column 6 to 11 data field values.



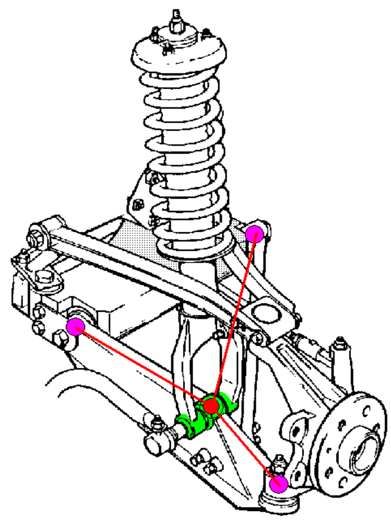
Typical Double Wishbone Type 1's, Solve direct (Sphere)

Data entry in columns 6 to 11 can be completed using the 'auto-fill' routines. The auto fill routines tend to add duplication but these are checked for and ignored by the solver.



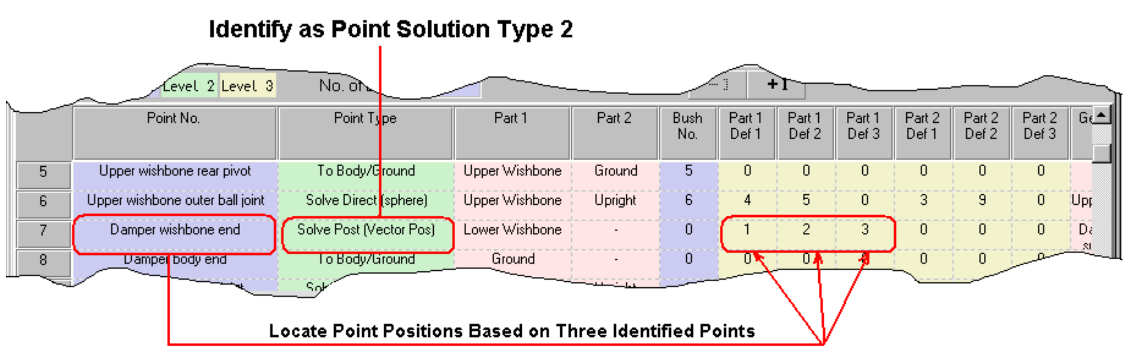
Screen Shot for settings on solution type 1, Adds 4 Equations

2 = Solve Post (Vector Position): This sets the point solution type to be based on a fixed position relative to three other points on the same body. It is a 'post' simultaneous equation solution calculation and as such has no control on the suspension articulation. Typical cases where this solution type is applied would be the case of a simple damper attachment to a wishbone. By definition it requires a minimum of four points to be associated with the relevant part. It is used where possible in preference to type 1 above, to increase the overall calculation speed by not adding three more sphere equations to the list. This solution type should not be used on push-rod or pull-rods since additional mechanism positions rely on it.



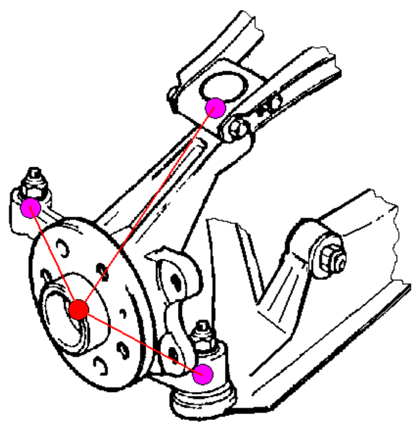
Example Solver Point Type 2

Define in columns 6,7 and 8 the three point No's of points on the same part as this point. No numbers are required for columns 9 to 11.

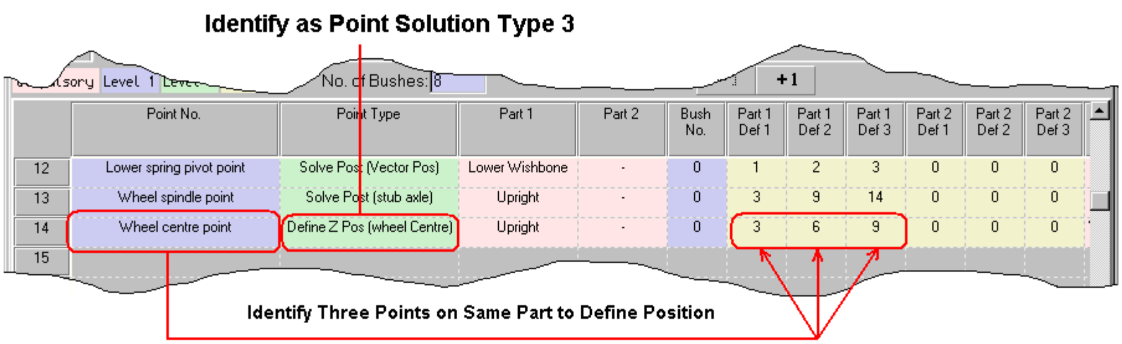


Screen Shot for settings on solution type 2

3 = Define Z Position (Wheel Centre): This point solution type is specific to wheel centre points. A point that has already been identified as the wheel centre would be given this point solution type by the ‘auto-fill’ routines. It requires in columns 6,7 and 8 three other reference points to be identified on the same part. One of these points should not be the wheel spindle point since that point is always post calculated and therefore not available to use in defining the wheel centre. This point solution type adds one equation to the simultaneous equation list. No values are required in columns 9,10 and11.



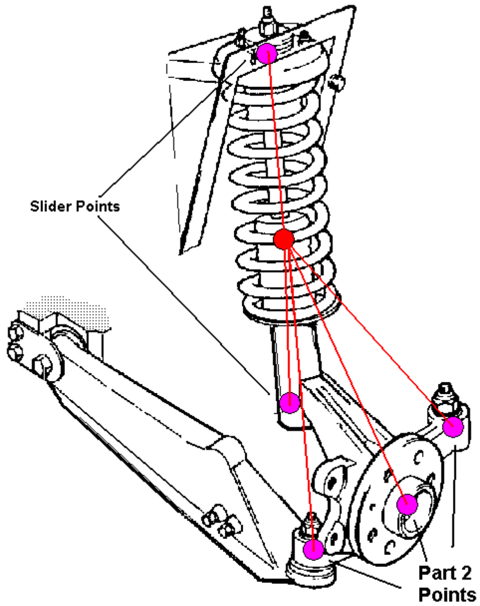
Example Solver Point Type 3, Typical point dependency



	Point No.	Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3
12	Lower spring pivot point	Solve Pos (Vector Pos)	Lower Wishbone	-	0	1	2	3	0	0	0
13	Wheel spindle point	Solve Post (stub axle)	Upright	-	0	3	9	14	0	0	0
14	Wheel centre point	Define Z Pos (wheel Centre)	Upright	-	0	3	6	9	0	0	0
15											

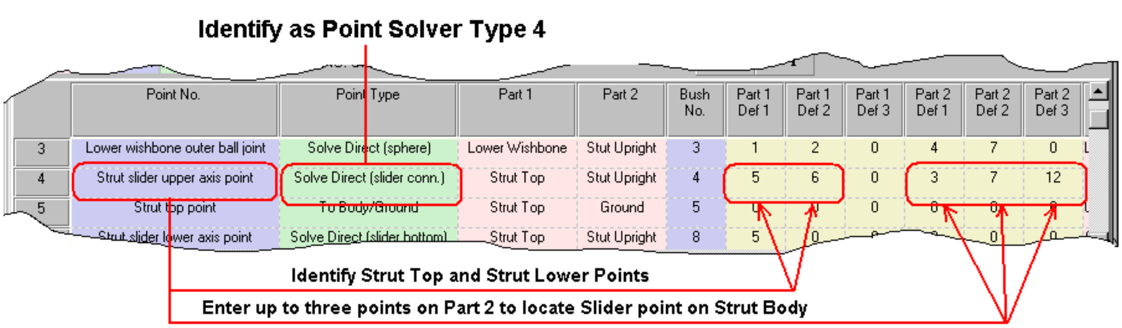
Screen Shot for settings on solution type 3, Adds 1 Equation

4 = Solve Direct (Slider Connection): This point solution type is specific to Strut slider points. A point that has already been identified as the 'Strut slider point', (general type 10), would be given this point solution type by the 'auto-fill' routines. It requires in columns 6 and 7 the two other strut slider points to be identified, (general types 5 and 11). This adds two equations to the simultaneous equations list.



Example Solver Point Type 4, Shows both part definitions

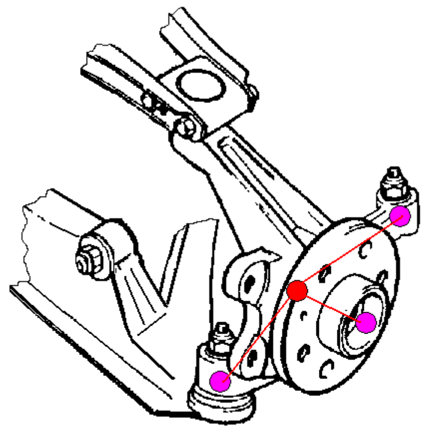
No value is required for column 8. In columns 9,10 and 11 point No's are required for up to three other points on the Strut Upright part, that define the location of it on that part. Each point number adds a sphere constraint equation between it and the strut slider point. The auto-fill routine will always add three points, any subsequent duplication in spherical constraints is ignored.



	Point No.	Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3
3	Lower wishbone outer ball joint	Solve Direct (sphere)	Lower Wishbone	Stut Upright	3	1	2	0	4	7	0
4	Strut slider upper axis point	Solve Direct (slider conn.)	Strut Top	Stut Upright	4	5	6	0	3	7	12
5	Strut top point	To Body/Ground	Strut Top	Ground	5	[illegible]	[illegible]	0	[illegible]	0	[illegible]
	Strut slider lower axis point	Solve Direct (slider bottom)	Strut Top	Stut Upright	8	5	0	[illegible]	[illegible]	0	0

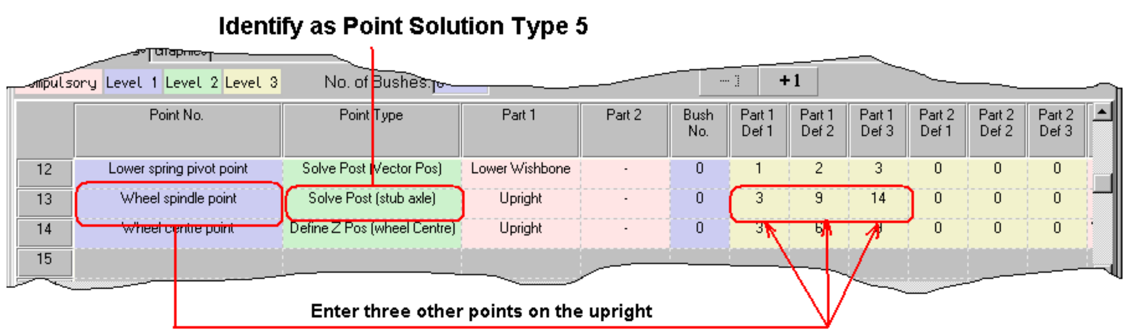
Screen Shot for settings on solution type 4, Adds 5 Equations

5 = Solve Post (Stub axle): This point solution type is specific to the point general type 2 'Stub Axle'. A point that has already been identified as the 'Stub axle point', (general type 2), would be given this point solution type by the 'auto-fill' routines. It requires three points to be defined through columns 6,7 and 8 that are also on the upright part. This is a post simultaneous solution method and so does not add any equations to the list.



Example Solver Point Type 5, Shows typical three point definition

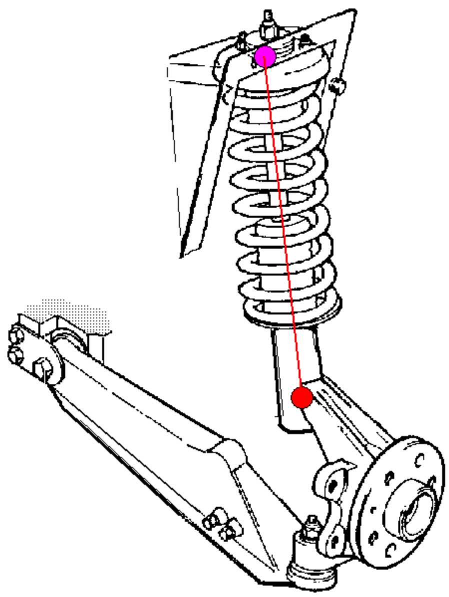
The normal point selection would include the wheel centre point. Because this is a Post simultaneous equation solver type it can use any three points on the body that have already been solved in the main simultaneous equation solution, this is typically major defining points such as all ball joints. It can not use points that are also post solved since depending on the solver order sequence they may not yet have been evaluated.



	Point No.	Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3
12	Lower spring pivot point	Solve Post (Vector Pos)	Lower Wishbone	-	0	1	2	3	0	0	0
13	Wheel spindle point	Solve Post (stub axle)	Upright	-	0	3	9	14	0	0	0
14	Wheel centre point	Define Z Pos (wheel Centre)	Upright	-	0	3	6	8	0	0	0
15											

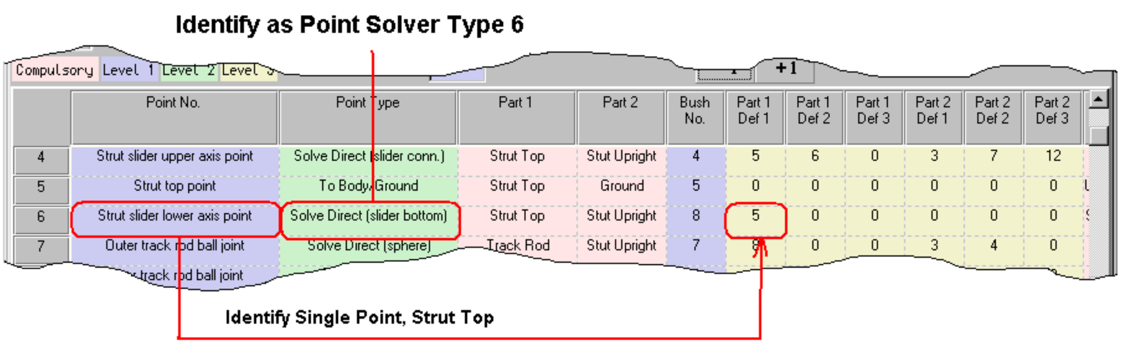
Screen Shot for settings on solution type 5

6 = Solve Direct (Slider Bottom): This point solution type is specific to Strut End Points. A point that has already been identified as the 'Strut End Point', (general type 10), would be given this point solution type by the 'auto-fill' routines. It requires in column 6 the strut top point to be identified, (general types 5). This adds one equation to the simultaneous equations list.



Example Solver Point Type 6, Shows single point identification

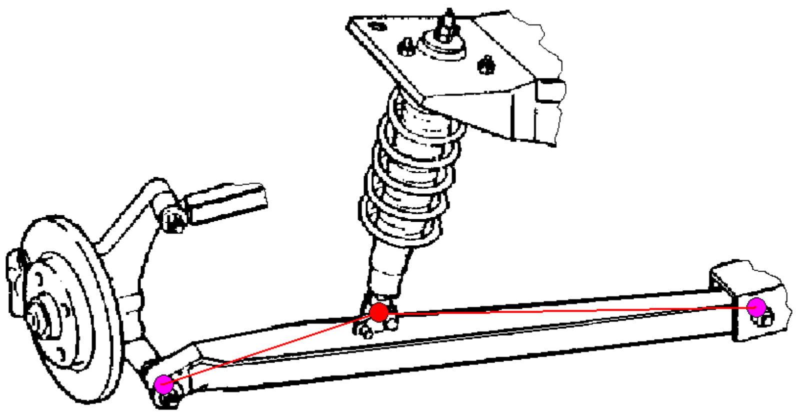
No values are required for columns 7 and 8 or columns 9,10 and 11.



	Point No.	Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3
4	Strut slider upper axis point	Solve Direct (slider conn.)	Strut Top	Stut Upright	4	5	6	0	3	7	12
5	Strut top point	To Body/Ground	Strut Top	Ground	5	0	0	0	0	0	0
6	Strut slider lower axis point	Solve Direct (slider bottom)	Strut Top	Stut Upright	8	5	0	0	0	0	0
7	Outer track rod ball joint	Solve Direct (sphere)	Track Rod	Stut Upright	7	8	0	0	3	4	0
	...track rod ball joint										

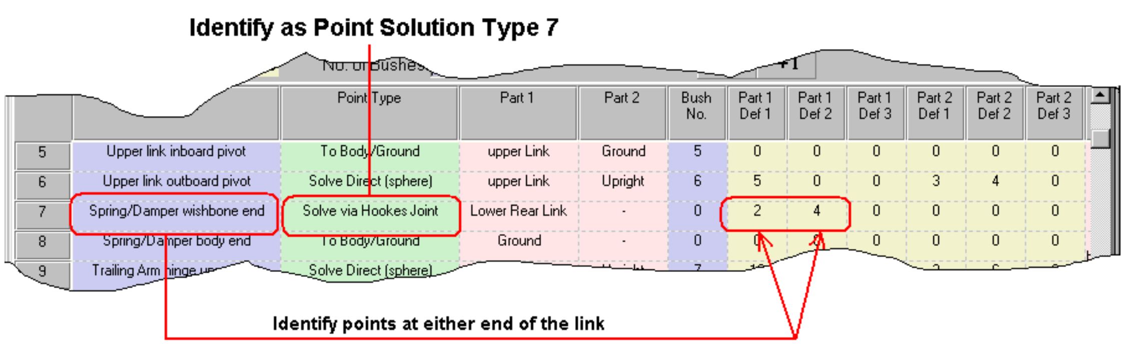
Screen Shot for settings on solution type 6, Adds 1 Equation

7 = Solve via Hookes Joint: This point solution type is a post calculation solution type that was added to handle the specific case of an under constrained kinematic model. The normal use for this solution type is when a spring or damper is attached to a tie rod. Kinematically a tie rod having just two attachment points provides a single spherical constraint to the model but in itself it has a degree of freedom left, rotation about the axis joining the two ends. When a damper is attached to this link, unless it is placed exactly on the tie rod axis, this degree of freedom means that the new position of the damper attachment cannot be solved. By placing a Hookes joint at one end of the tie rod this rotational degree of freedom is removed and a kinematic solution can be identified.



Example Solver Point Type 7, Two points identified

The use of a Hookes joint in the kinematic model does not effect the fully bushed compliant solution. The rotational degree of freedom in the complaint case is taken out via suitable bush properties. An example of the Hookes joint point solver can be found in default template 19.

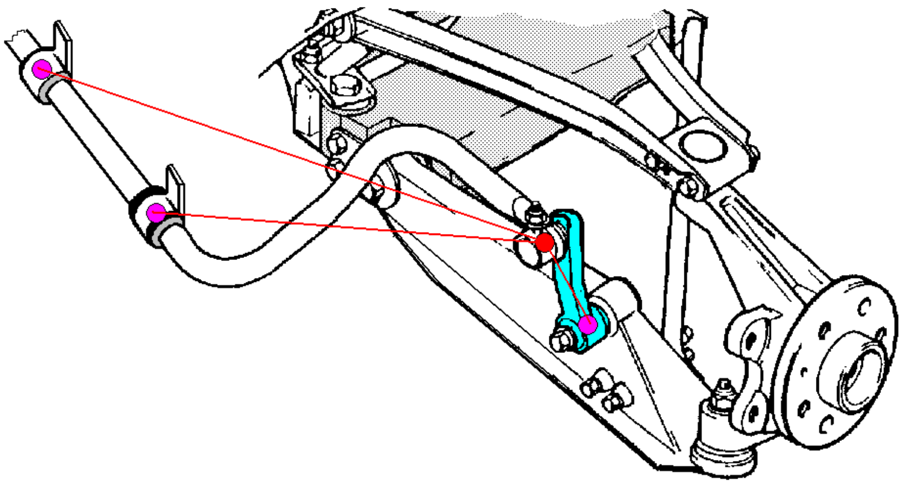


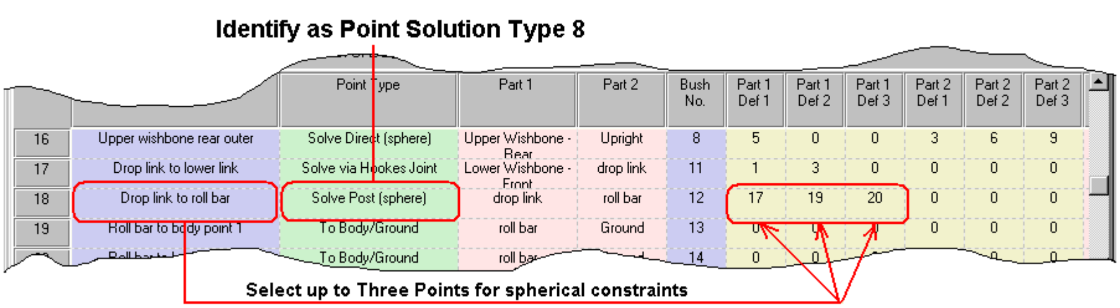
		Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3
5	Upper link inboard pivot	To Body/Ground	upper Link	Ground	5	0	0	0	0	0	0
6	Upper link outboard pivot	Solve Direct (sphere)	upper Link	Upright	6	5	0	0	3	4	0
7	Spring/Damper wishbone end	Solve via Hookes Joint	Lower Rear Link	-	0	2	4	0	0	0	0
8	Spring/Damper body end	To Body/Ground	Ground	-	0	0	0	0	0	0	0
9	Trailing Arm hinge ...	Solve Direct (sphere)	[illegible]	[illegible]	7	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Screen Shot for settings on solution type 7, Example template 19

8 = Solve Post (Sphere): This point solution type is a post calculation solution type and as such adds no equations to the simultaneous equation list. It is used as an alternative to point solution type 2 in special cases. It is always preferable to use point solution type 2 rather than this one. Since the sphere equation always has more than one solution which can lead to errors when two solutions are similar.

Default template 23 uses this solution type for the anti roll bar drop link attachment. As with the conventional 'Solve Direct (sphere)' solution type up to three points are listed in columns 6,7 and 8 a spherical constraint equation added for each point number. All points entered must be on the same part as the point being solved. No data points need be defined in columns 9,10 and 11.





		Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3
16	Upper wishbone rear outer	Solve Direct (sphere)	Upper Wishbone - Rear	Upright	8	5	0	0	3	6	9
17	Drop link to lower link	Solve via Hookes Joint	Lower Wishbone - Front	drop link	11	1	3	0	0	0	0
18	Drop link to roll bar	Solve Post (sphere)	drop link	roll bar	12	17	19	20	0	0	0
19	Roll bar to body point 1	To Body/Ground	roll bar	Ground	13	0	0	0	0	0	0
[illegible]	[illegible]	To Body/Ground	roll bar	[illegible]	14	0	0	[illegible]	[illegible]	0	0

Screen Shot for Settings on solution type 8

9 = Pre-Solve (Kinematic Fix): A pre main solver option calculation. Requires no defining points since the point is assumed to be inactive in kinematic mode. It remains fixed to the part it is defined on (normally ground or a ground fixed part). It is used to add additional compliance effects for parts such as rack mounts and sub frames that are assumed to have no kinematic effect but are included in the compliance matrix.

10 = Solve Post (Direct Substitution): A post solver option calculation. Is used to identify all points that belong to the ‘calculated’ group. That is points with General Type tags of 55 to 72. Requires no defining points since the points general type is sufficient to identify the solution.

11 = Solve Post (Vector Sphere): Used to solve the position of a drive shaft inner CV type point. A post solver option calculation that requires three points to be identified, Itself, its axis point (that together with itself identifies the axis that the inner joint slides along and the point for the outer drive shaft centre. the distance between the inner and outer shaft centres is held constant whilst the inner CV ‘plunges’ along the specified axis.

12 = Solve Post (Sphere Predictive): This post solver calculation is identical to Solution type 2 with the exception of how the solution choice from the two possible options is made. It takes the same option of the two as was identified for the static case. This can be a useful option when type 2 has an instability due to inversion.

13 = Solve By Local Co-ordinate System: This post solver calculation identifies that the points position will be set by a local co-ordinate system. The point must already have been be defined in a local co-ordinate system outside of the template editor.

14 = Derive From Graphical Element: This post solver calculation indicates that the points position is derived from a graphical element in the model, for example a sphere centre. Thus this solver type is normally set outside of the template editor as you add a point via a graphical element pick.

15 = Solve Direct (Sphere-Body Seeded): This solution type is identical to type 1, being a solution type based on the equation of a sphere. (see type 1 for full definition). The only difference is that at each kinematic solution step the previous steps solution is seeded by the change in body z motion as the start value for this new step.

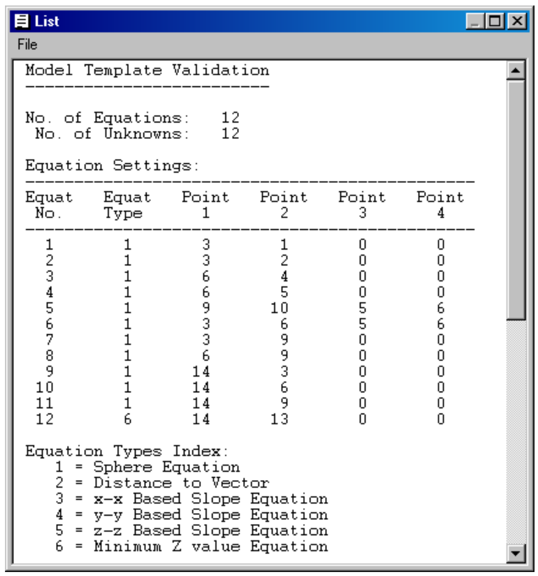
16 = Solve Post (Mono-Shock): This post solver calculation is a specific solution case used for mono-shock suspension templates. It uses four indicated points, an example of which is used in template type 34.

17 = Solve Post (Sliding Pivot): This post solver calculation is a specific solution case that is again part of the mono-shock suspension. It uses three indicated points, an example of which is used in template type 34.

18 = Solve Post (Sphere – Averaging Distance): This post solver calculation uses a sphere solution but seeds on an average of body and wheel travel.

15.10 Template Validation

The definition of a template is used to identify unknown point positions (x, y and z), and the number of equations. For a successful template definition, the number of equations must equal the number of unknowns. A utility is provided to test the unknown to equations count. From the template window, select *Data / Run Validation Test…* The produced display lists the number of unknowns, and the number of equations. Each equation is listed by type and with the associated point numbers.



```
List
File
Model Template Validation
-------------------------

No. of Equations:   12
 No. of Unknowns:   12

Equation Settings:
----------------------------------------------
Equat   Equat   Point   Point   Point   Point
 No.    Type      1       2       3       4
----------------------------------------------
  1       1       3       1       0       0
  2       1       3       2       0       0
  3       1       6       4       0       0
  4       1       6       5       0       0
  5       1       9      10       5       6
  6       1       3       6       5       6
  7       1       3       9       0       0
  8       1       6       9       0       0
  9       1      14       3       0       0
 10       1      14       6       0       0
 11       1      14       9       0       0
 12       6      14      13       0       0

Equation Types Index:
   1 = Sphere Equation
   2 = Distance to Vector
   3 = x-x Based Slope Equation
   4 = y-y Based Slope Equation
   5 = z-z Based Slope Equation
   6 = Minimum Z value Equation
```

Example Template validation for default template type 1

This list can be compared with expected equations to debug new template problems. Remember that it will not list 'post' calculation unknowns or equations, only those that are solved as the list of simultaneous equations. If we compare the validation list with the template 'settings', we can identify the source of each equation. The validation of template 1 shows 12 unknowns. Looking at the conventional double wishbone suspension, we can identify these (we will ignore additional points that are post calculated);

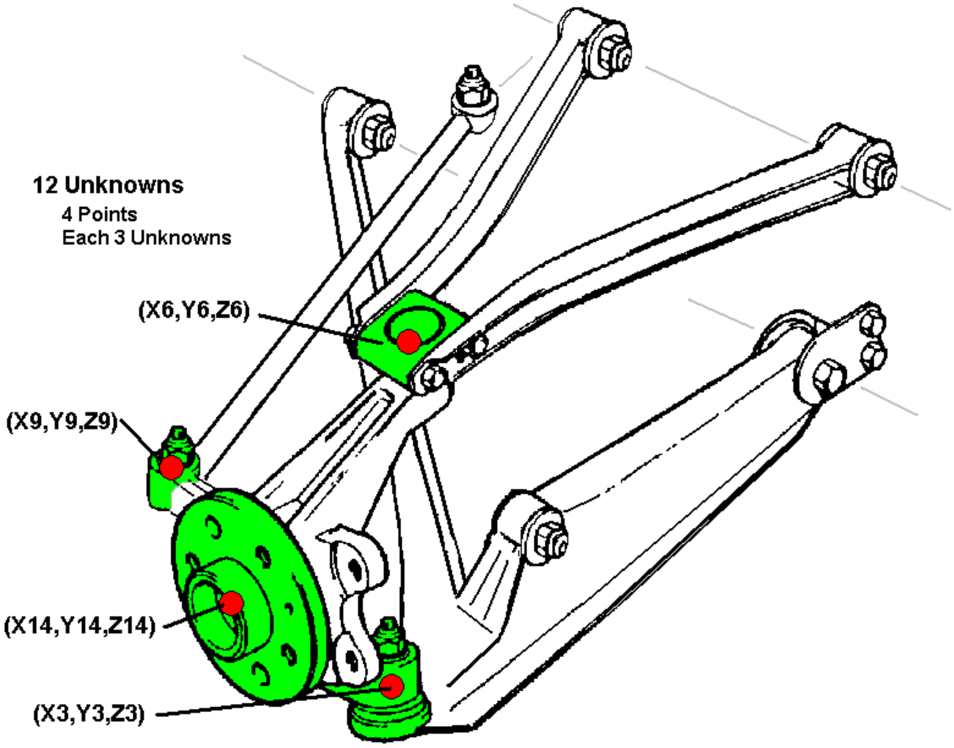
Lower Wishbone Outer Ball Joint, X, Y and Z co-ordinates, (x3, y3, z3)

Upper Wishbone Outer Ball Joint, X, Y and Z co-ordinates, (x6, y6, z6)

Steering Arm Outer Ball Joint, X, Y and Z co-ordinates, (x9, y9, z9)

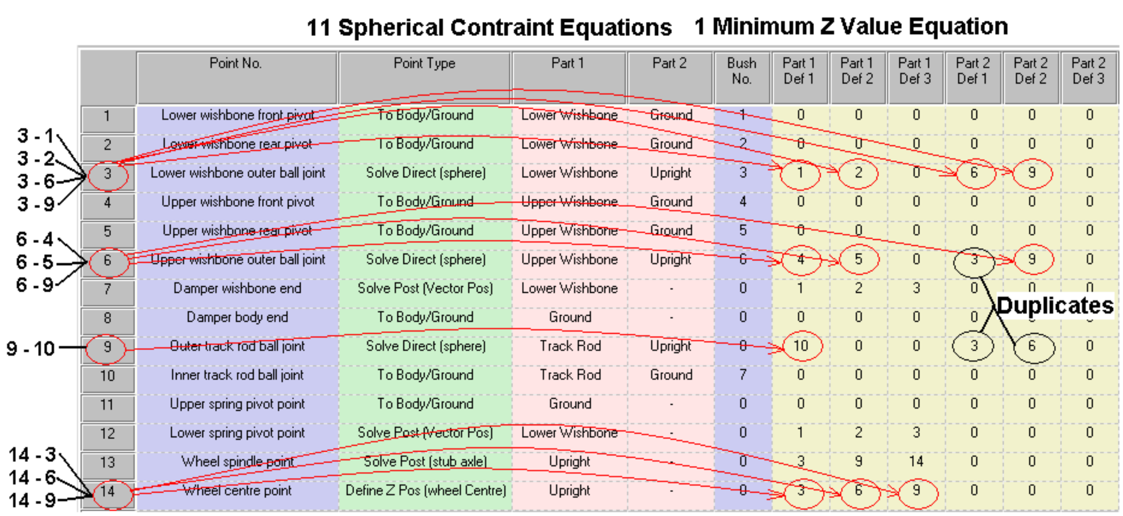
Wheel Centre Point, X, Y and Z co-ordinates, (x14, y14, z14)

These 12 unknowns are illustrated on the image below.



Default Template Type 1, 12 unknowns indicated

If we now look at the data in the 'Settings' display we can identify the origin of the 12 equations used to solve for the 12 unknowns.



	Point No.	Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3
1	Lower wishbone front pivot	To Body/Ground	Lower Wishbone	Ground	1	0	0	0	0	0	0
2	Lower wishbone rear pivot	To Body/Ground	Lower Wishbone	Ground	2	0	0	0	0	0	0
3	Lower wishbone outer ball joint	Solve Direct (sphere)	Lower Wishbone	Upright	3	1	2	0	6	9	0
4	Upper wishbone front pivot	To Body/Ground	Upper Wishbone	Ground	4	0	0	0	0	0	0
5	Upper wishbone rear pivot	To Body/Ground	Upper Wishbone	Ground	5	0	0	0	0	0	0
6	Upper wishbone outer ball joint	Solve Direct (sphere)	Upper Wishbone	Upright	6	4	5	0	3	9	0
7	Damper wishbone end	Solve Post (Vector Pos)	Lower Wishbone	-	0	1	2	3	0	0	0
8	Damper body end	To Body/Ground	Ground	-	0	0	0	0	0	0	0
9	Outer track rod ball joint	Solve Direct (sphere)	Track Rod	Upright	8	10	0	0	3	6	0
10	Inner track rod ball joint	To Body/Ground	Track Rod	Ground	7	0	0	0	0	0	0
11	Upper spring pivot point	To Body/Ground	Ground	-	0	0	0	0	0	0	0
12	Lower spring pivot point	Solve Post (Vector Pos)	Lower Wishbone	-	0	1	2	3	0	0	0
13	Wheel spindle point	Solve Post (stub axle)	Upright	-	0	3	9	14	0	0	0
14	Wheel centre point	Define Z Pos (wheel Centre)	Upright	-	0	3	6	9	0	0	0

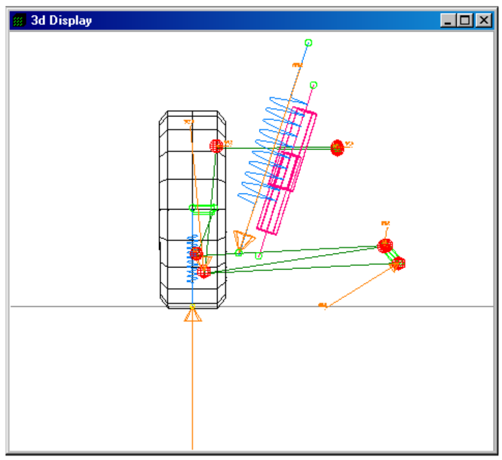
Screen shot of settings for default template type 1

Compare the paired numbers with those given in the validation list: they will match. Minimum Z equation added automatically for wheel centre point. Three duplicate spherical constraints ignored by the solver.

15.11 Exercise 1, Modifying an Existing Template

As a simple exercise in modifying a template, we will take default template 1 and change the component that the spring/damper is attached too. To validate that we have changed the connection point of the spring, we will use the compliant analysis force display.

Open a new front-end only model, and use default template type 1. Turn the compliance ‘on’ and note the calculated force display (see front view screen shot below).



Standard templates calculated forces

Now modify the points ‘Damper wishbone end’ and ‘lower spring pivot point’ such that they are attached to the upright rather than the lower wishbone.

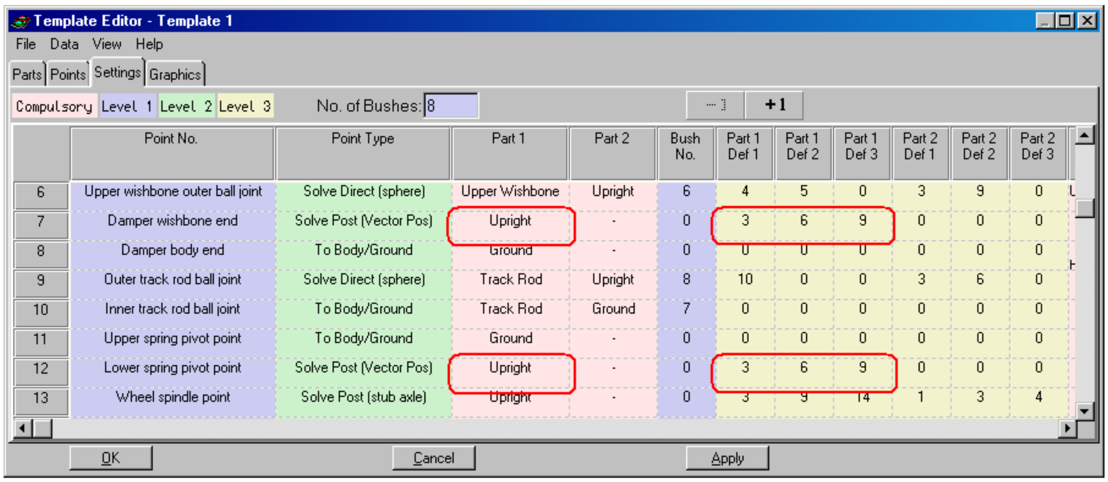
Remember to set any required changes to the new points in columns 6 to 11. For the purpose of this exercise do not use the ‘auto-fill’ routines we will cover these later.

Make the require template changes and confirm the difference by checking for a change in the calculated forces.

Hint: Use the ‘apply’ button to apply the changes to the current model (you can also press ‘ok’).

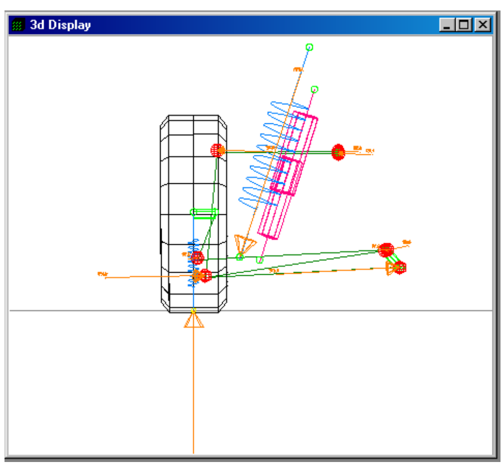
Exercise: Solution

You should have made changes to the ‘Part 1’ settings for points 7 and 12. You will also have needed to change columns 6, 7 and 8 for both of them to reference three points (other than themselves), on the upright. Could pick any three from points 3, 6, 9 or 14.



	Point No.	Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3
6	Upper wishbone outer ball joint	Solve Direct (sphere)	Upper Wishbone	Upright	6	4	5	0	3	9	0
7	Damper wishbone end	Solve Post (Vector Pos)	Upright	-	0	3	6	9	0	0	0
8	Damper body end	To Body/Ground	Ground	-	0	0	0	0	0	0	0
9	Outer track rod ball joint	Solve Direct (sphere)	Track Rod	Upright	8	10	0	0	3	6	0
10	Inner track rod ball joint	To Body/Ground	Track Rod	Ground	7	0	0	0	0	0	0
11	Upper spring pivot point	To Body/Ground	Ground	-	0	0	0	0	0	0	0
12	Lower spring pivot point	Solve Post (Vector Pos)	Upright	-	0	3	6	9	0	0	0
13	Wheel spindle point	Solve Post (stub axle)	Upright	-	0	3	9	14	1	3	4

Modified template 1, Changes ringed



Revised model illustrating the change in calculated forces

If this was required to be a permanent change, we would probably change the default x,y,z co-ordinates for points 7 and 12 to make them more realistic and change some of the point descriptions to be inline with our new template.

Unless we save this modified template, our changes will be lost when we close the application. In this instance we do not want to retain the changes to this default template. In the next section we will look at creating and saving new templates such that they are available for future use.

15 User Templates (2)

15.1 Overview

In this chapter, we extend the previous chapters description of the data structure for templates, by creating a new template and making this available for subsequent analysis runs.

This chapter contains the following sections:

15.2 Template Sources

Each template is stored in a particular template 'slot' (normally referred to as template type). So the conventional double wishbone suspension, with damper attached to the lower wishbone, is stored in slot 1 by the hard coded default templates.

Each saved model data file references the appropriate template slot via this number. The data file itself does not necessarily have a copy of the template structure, only the slot number (you can optionally save the template(s) with the data file *SetUp / Include User Templates in Data Files)*. Thus it is important to remember that if you make a change to a template the safest approach is to save it to a new 'free' slot number.

Templates are stored at four levels. Each subsequent level can overwrite the current slot template with a new template, if it defines a template slot number that is already defined by a previous level.

The first template source level is the 'hard-coded defaults'. These are built into the program at compile time and thus whilst they can be modified or over-written by subsequent levels, (as identified above), they cannot be directly changed by the user. The 'hard-coded defaults' at release version 5.01a are:

Slot 1 Double Wishbone, damper to lower wishbone

Slot 2 Lower H frame, single upper link

Slot 3 Steerable Macpherson Strut

Slot 4 Non-Steerable Macph Strut, two lower ball joints, tie to ground

Slot 5 5-Link Rigid Axle (Panhard Rod)

Slot 6 Double Wishbone, damper to upper wishbone

Slot 7 Non/Steerable Macpherson Strut, toe link to lower wishbone

Slot 8 4-Link Rigid Axle (Panhard Rod)

Slot 9 4-Link Rigid Axle (Twin Upper)

Slot 10 Trailing Arm upper and lower rear links

Slot 11 Semi Trailing Arm

Slot 12 Steerable Twin Parallel Wishbones and Knuckle

Slot 14 Double Wishbone Push Rod to Damper

Slot 15 Double Wishbone, Rocker Arm Damper

Slot 16 Non-Steerable Lower 'A' Arm with Toe Link

Slot 17 Double Wishbone, Push Rod, Mono-shock

Slot 18 Double Wishbone, Upper Toe Link, Drop 'S' Link

Slot 19 Hinged Trailing Arm, Twin lower Link

Slot 20 Double Wishbone, Twin Outer Ball Joints

Slot 21 5-Link Rigid Axle (Watts Linkage)

Slot 22 Double Wishbone, Twin Outer Ball Joints, Spring Front

Slot 23 Double Wishbone, Anti-Roll Bar

Slot 24 Steerable Macpherson Stut, Twin Outer Ball Joints

Slot 25 Double Wishbone, Twin Lower Outer Ball Joints

Slot 26 Double Wishbone, Damper to Lower Wishbone, Compliant Rack

Slot 27 Steerable Macpherson Strut, Twin Lower Link

Slot 28 4-Link Rear, Transverse Control Link

Slot 29 Twist Beam – Twin Wheel

Slot 30 Generic 5-Link Rear

Slot 31 Leaf Spring Rigid Axle (Panhard Rod)

Slot 32 5-link Rigid Axle (offset wheels)

Slot 33 McPherson Strut with Steerable Hub

The second template source level is the 'User defined Templates'. The user defined templates are stored in a specific file (_user_templates.dat), in the same folder the software is installed in. On program start-up, this file is checked for, and if found, any user defined template information stored in it is loaded into the application. As discussed above, if the same slot number as one of the 'hard coded defaults' is used, the user template will overwrite the hard coded default. This file will only exist if the user has previously selected to save a template from within the template editor.

The third template source level is the 'Custom Templates'. These are stored by the user into a user-defined file (disc location, folder location and name). They are only loaded from the file into the relevant template slots when the user scans for and reads the required file. Thus custom templates can be stored in any number of separate files, these files can be passed between users or stored on a central repository. As identified previously, slot numbers referenced by the custom templates will overwrite any existing default or user template definition. If you use custom templates, the template properties must be loaded before you load a data file that references this custom template.

The fourth template source is from within a loaded data file. If the template has been saved with the data file, this will be loaded into its respective slot when the data file is opened. Note that these changes to the template will continue to reside in the loaded slot, until it is overwritten, or the program is restarted, or one of the *File / Re-Read* menu options is run.

Slot No.	1 - Hard-coded defaults	2 - User Defined Templates	3 – Custom Templates	4-Data Files
1	Double Wishbone, damper to lower wishbone.			
2	H-frame Lower, single upper link.			
3	Steerable Macpherson Strut.			
4	Non-Steerable Macpherson Strut, twin lower link.			
5	5-Link Rigid Axle (Panhard Rod)			
6	Double Wishbone, damper to upper wishbone.			
7	Non-Steerable Macph strut, toe link to wishbone.	Would overwrite hard coded	Would overwrite defined	
8	4-Link Rigid Axle (Panhard Rod)			
9	4-link Rigid Axle (Twin Upper)		Would overwrite hard coded	
10	Trailing Arm, upper and lower rear links.			
11	Semi Trailing Arm.			
12	Steerable Twin Parallel Wishbones and knuckle.			
13		Free slot to use		
14	Double Wishbone, Push Rod to damper.			
15	Double Wishbone, Rocker arm damper.			
16	Non Steerable lower A with toe link.			
17	Double Wishbone, pushrod mono-shock.			
18	Double Wishbone, Upper toe link + S link.			
19	Hinged Trailing Arm, Twin Lower Link.			
20	Double Wishbone, twin outer ball joints.			
21	5-Link Rigid Axle (Watts Linkage)			
22	Double Wishbone, twin outer ball joints Spring front.			
23	Double Wishbone, twin outer ball anti roll bar.			
24	Steerable Macpherson Strut, twin outer ball joints.			
25	Double Wishbone, twin lower outer ball joints.			
26	Double Wishbone, damper to lower, comp rack.			
27	Steerable Macpherson Strut, Twin lower Link			
28	4-Link Rear, Transverse Control Link			
29	Twist Beam – Twin Wheel			
30	Generic 5-Link Rear			
31	Leaf Spring Rigid Axle (Panhard Rod)			
32	5-link Rigid axle (offset wheels)			
33	Macpherson Strut with steerable hub			
34….		Free slot to use		
….		Free slot to use		
…50		Free slot to use		

Schematic of Template Levels and Slot Nos.

The schematic above illustrates the templates levels. ‘Cyan’ shows the ‘free’ slots where new ‘user’ or ‘custom’ templates could be stored. The ‘Red’ boxes indicate example slots for ‘user’ and ‘custom’ templates that would overwrite the ‘Hard-coded’ defaults. The ‘mauve’ box indicates an example ‘custom’ template slot that would have overwritten the ‘Red’ box ‘user’ template.

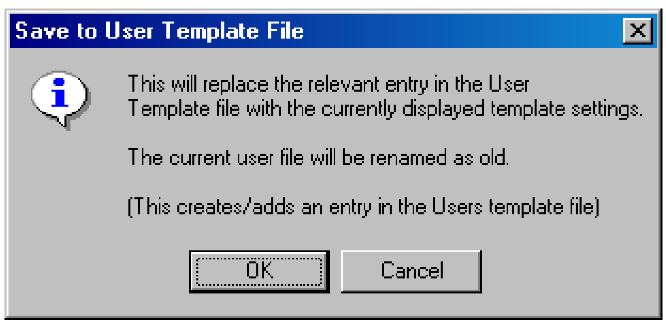
15.3 Storing and Saving Templates

As identified previously, the user can save templates either as ‘User’ templates, ‘Custom’ templates, or within the data file. Saving templates involves writing the specified template slot number to a file.

The ‘user’ templates file is a predefined file name and location, whilst the ‘custom’ templates are saved to user specified files. ‘Custom’ templates can be saved either singularly or as a complete capture of all currently defined templates, including unaltered ‘default’ templates.

15.4 Saving to the User Templates File

A template is saved to the user templates file from the template editor. Open the template editor, *File / Edit Templates*. You would then display the required template by selecting the ‘+’ and ‘-‘ icons. Once displayed, select the menu *Data / Save Template to User File*. You will be warned about the data change and overwrite existing users file, (if found).



Saving current template to user template file

The user templates file, (_User_Templates.dat), will only exist if a previous template has been saved to the user templates file. If the user templates file already contains an entry for the slot number selected, this will be replaced with the current settings. If no entry exists in the User templates file for the selected slot number then this template definition will be added to the User templates file.

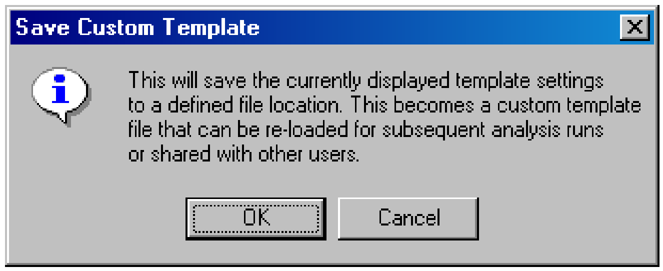
When adding/over writing the contents of the User templates file the existing file is renamed to _User_Templates.dat.old.

If you choose to delete the user templates file, all templates revert back to the hard coded settings.

At any time during a program run, you can revert back to the hard coded defaults + saved user templates by selecting the main menu item, *File / Re-Read Default +User Templates*. If the user templates has been deleted this will effectively set all template settings back to the hard-coded defaults.

15.5 Saving Custom template Files

A modified template data settings can be saved to a separate file as a ‘Custom’ template. On subsequent analysis runs this file would need to be re-read before this custom template can be used. From within the template editor, the currently displayed template can be saved to a custom file by selecting *Data / Save Custom Template…* Confirm the information message and use the file browser to identify the required file location and name (this could be on a remote/central server).



Save Custom Template file message

Templates loaded from custom files will overwrite any existing data in the specified slot number.

It is also possible to take a ‘snap-shot’ copy of all defined template data settings, and save them to a single ‘custom’ templates file. From the main menu, select *File / Save Custom Templates (All)*, then use the displayed file browser to locate the required file location and name.

On subsequent runs, if you require to use a Custom template, these must be re-loaded as, unlike ‘user’ templates, they are not automatically loaded when starting the application. To load a ‘custom’ template from the main menu, select *File / Add Custom Templates…* Use the displayed file browser to locate the required custom templates file.

15.6 Creating New Templates

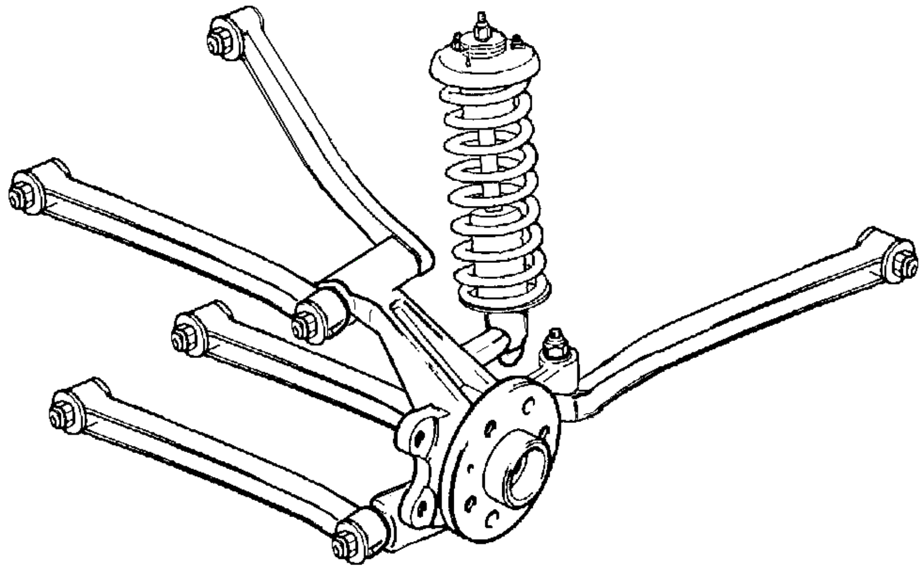
If you need to create a new template, you would normally pick a currently free ‘slot’ to avoid overwriting an existing template. In the template editor, step through until you find a Slot identified as ‘Not Defined’.

To assist in building a template that is only a slight change from an existing one, you can pre-fill your new template from an existing one, by using the *File / Fill current Template From /…* menu option.

We will create a new template completely from scratch via the following exercise.

15.7 Exercise 2 – Creating a New Template

As an exercise in understanding the structure of user defined templates, we will create from scratch a new template for the generic five link rear suspension illustrated below.



Example five link rear suspension

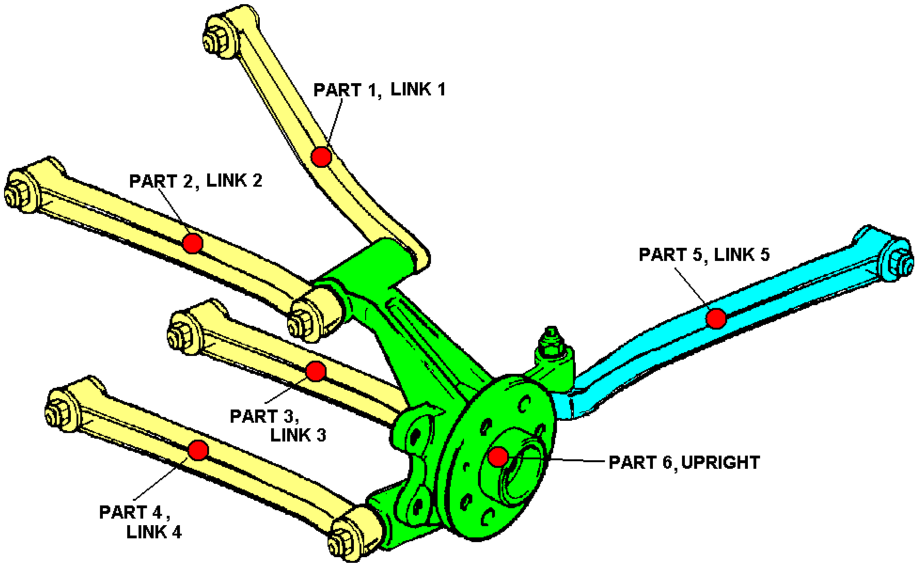
To create the new template, open the template editor (*File / Edit Templates).* Find a convenient empty slot (the screen shots shown with this examples use slot number 8). But you may wish to select a free slot such as number 44.

Step 1: Starting at the ‘Parts’ tab we need to;

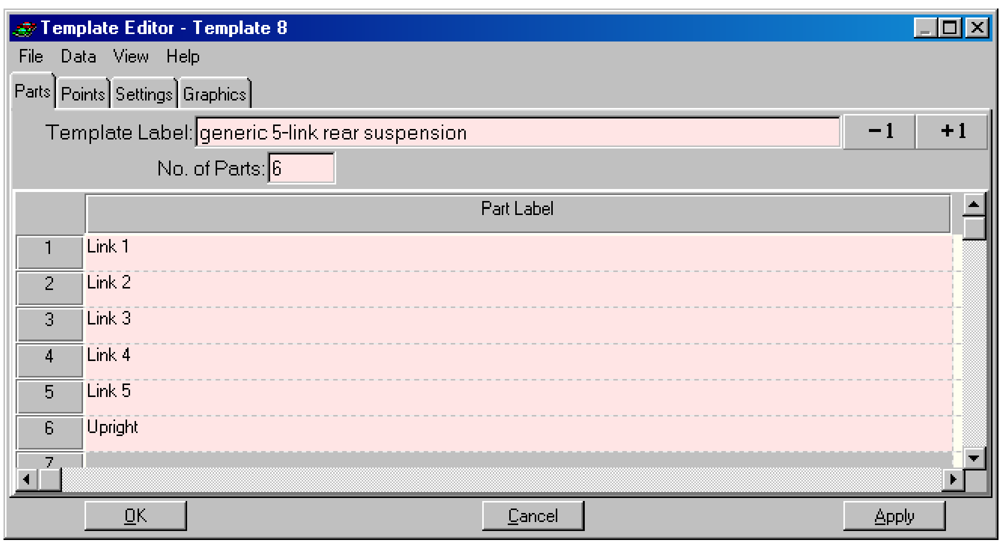
1) Give the template a label
2) Identify the number of parts
3) Give each part a descriptive label.

No need to define a part for ‘Ground’.

See overleaf for example ‘Parts’ data values.



Identification of Parts, Six in Model



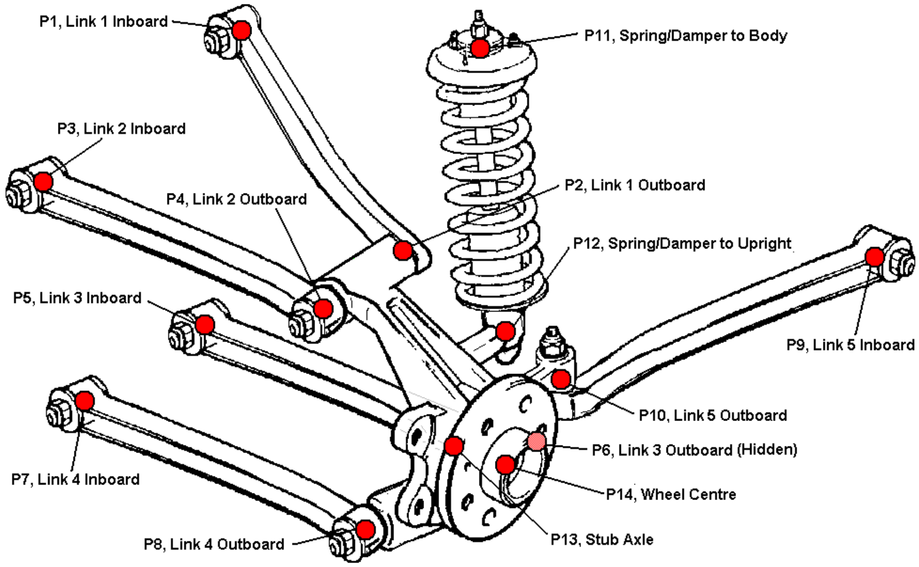
Screen shot with 'Parts' data values entered

Step 2: Change to the 'Points' tab. Now we need to;

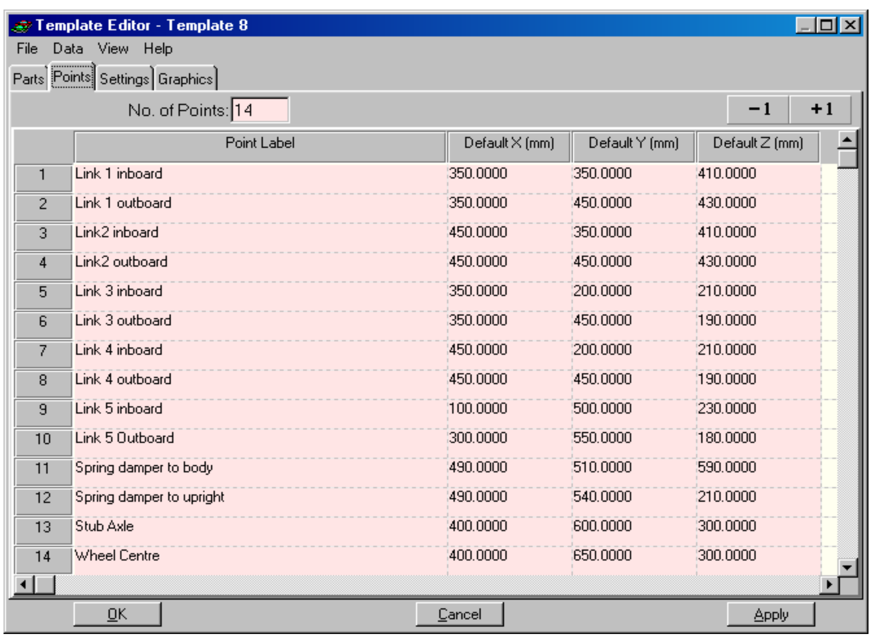
1) Identify the number of points
2) Give each point a short label
3) Give each point a descriptive label
4) Supply the default x, y and z values for each point.

Remember you don't need to define a point for the tyre contact point and we can make the spring-damper attachment points as coincident.

See overleaf for example 'Points' data values.



Identification of Points, 14 in Model



	Point Label	Default X (mm)	Default Y (mm)	Default Z (mm)
1	Link 1 inboard	350.0000	350.0000	410.0000
2	Link 1 outboard	350.0000	450.0000	430.0000
3	Link2 inboard	450.0000	350.0000	410.0000
4	Link2 outboard	450.0000	450.0000	430.0000
5	Link 3 inboard	350.0000	200.0000	210.0000
6	Link 3 outboard	350.0000	450.0000	190.0000
7	Link 4 inboard	450.0000	200.0000	210.0000
8	Link 4 outboard	450.0000	450.0000	190.0000
9	Link 5 inboard	100.0000	500.0000	230.0000
10	Link 5 Outboard	300.0000	550.0000	180.0000
11	Spring damper to body	490.0000	510.0000	590.0000
12	Spring damper to upright	490.0000	540.0000	210.0000
13	Stub Axle	400.0000	600.0000	300.0000
14	Wheel Centre	400.0000	650.0000	300.0000

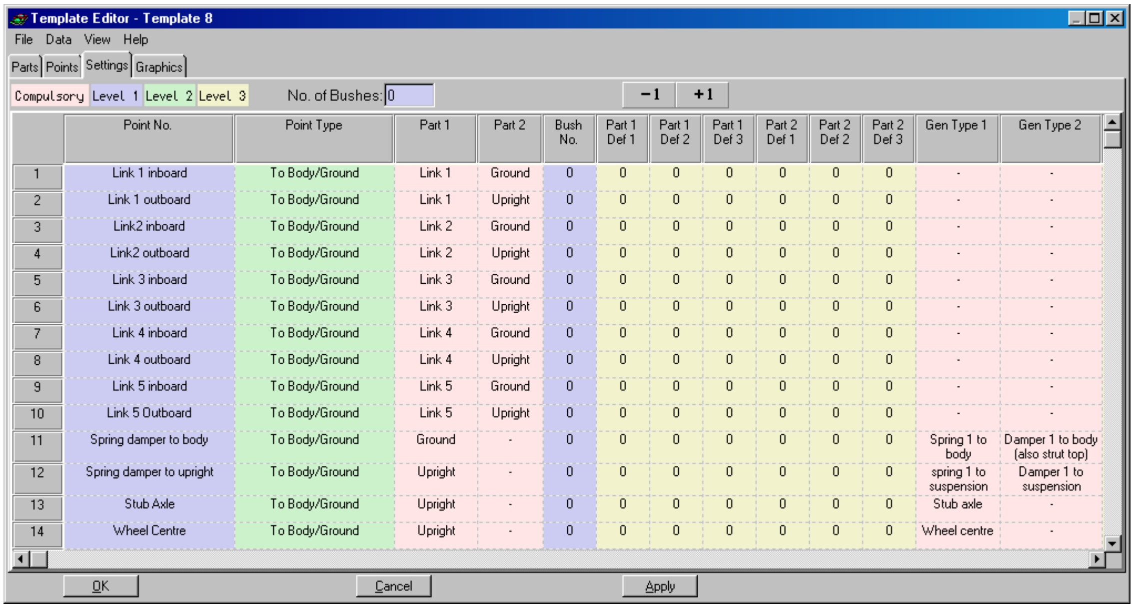
Screen shot with 'Points' data values entered

Step 3: Change to the 'Settings' tab. Now we need to:

1) Identify the 'Part 1' and 'Part 2' settings for each point.
2) Define any General types for each point.

Remember any point that is a connection between two parts, or a connection between a part and ground, requires two parts to be defined.

See overleaf for example point 'Settings'.



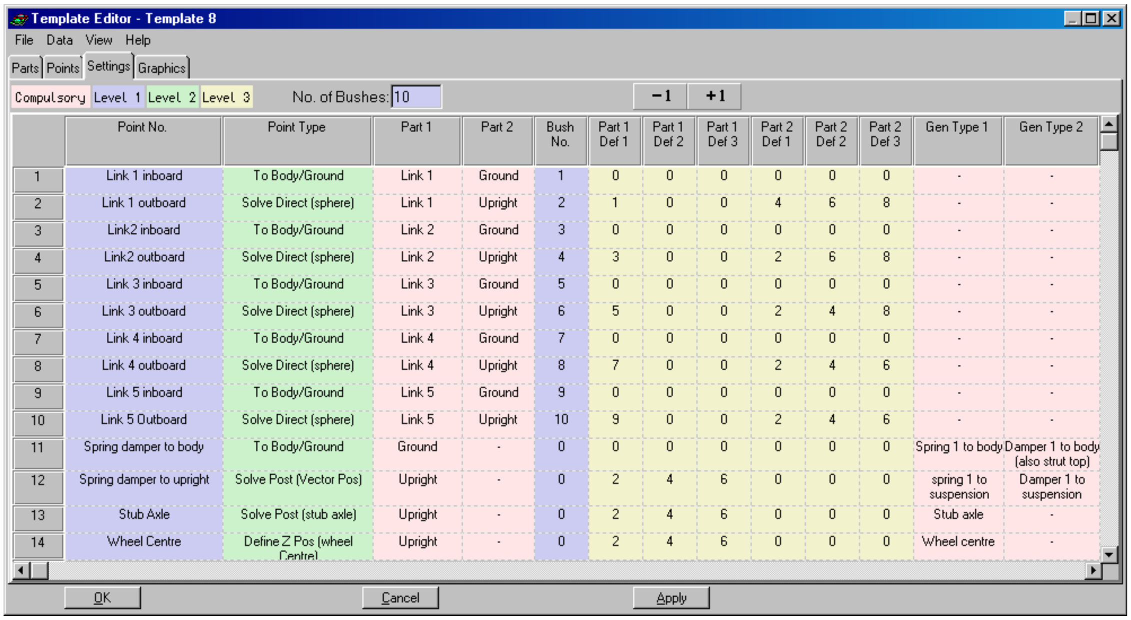
	Point No.	Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3	Gen Type 1	Gen Type 2
1	Link 1 inboard	To Body/Ground	Link 1	Ground	0	0	0	0	0	0	0	-	-
2	Link 1 outboard	To Body/Ground	Link 1	Upright	0	0	0	0	0	0	0	-	-
3	Link2 inboard	To Body/Ground	Link 2	Ground	0	0	0	0	0	0	0	-	-
4	Link2 outboard	To Body/Ground	Link 2	Upright	0	0	0	0	0	0	0	-	-
5	Link 3 inboard	To Body/Ground	Link 3	Ground	0	0	0	0	0	0	0	-	-
6	Link 3 outboard	To Body/Ground	Link 3	Upright	0	0	0	0	0	0	0	-	-
7	Link 4 inboard	To Body/Ground	Link 4	Ground	0	0	0	0	0	0	0	-	-
8	Link 4 outboard	To Body/Ground	Link 4	Upright	0	0	0	0	0	0	0	-	-
9	Link 5 inboard	To Body/Ground	Link 5	Ground	0	0	0	0	0	0	0	-	-
10	Link 5 Outboard	To Body/Ground	Link 5	Upright	0	0	0	0	0	0	0	-	-
11	Spring damper to body	To Body/Ground	Ground	-	0	0	0	0	0	0	0	Spring 1 to body	Damper 1 to body (also strut top)
12	Spring damper to upright	To Body/Ground	Upright	-	0	0	0	0	0	0	0	spring 1 to suspension	Damper 1 to suspension
13	Stub Axle	To Body/Ground	Upright	-	0	0	0	0	0	0	0	Stub axle	-
14	Wheel Centre	To Body/Ground	Upright	-	0	0	0	0	0	0	0	Wheel centre	-

Initial Settings for Part 1, Part 2 and Gen Type values.

We can now use the auto-fill routines to complete the other required data values. Alternatively you may want to try completing them by hand and then check them with the auto-fill settings. To use the auto-fill select *Data / Test Auto Fill* and if you then select *Level 1* you can note the changes level by level.

Using the *Test* auto-fill routines allows you to visually review the differences, and accept or reject the changes.

Repeat the Test Auto-fills for Levels 2 and 3, such that you have a fully defined template (check with results on following page).



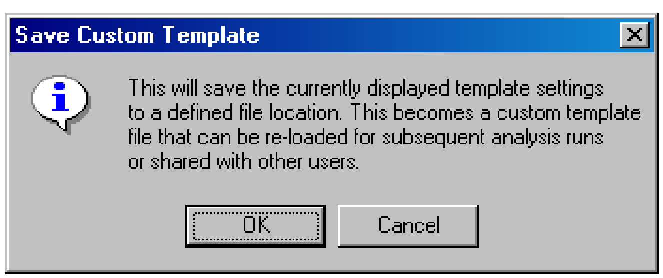
	Point No.	Point Type	Part 1	Part 2	Bush No.	Part 1 Def 1	Part 1 Def 2	Part 1 Def 3	Part 2 Def 1	Part 2 Def 2	Part 2 Def 3	Gen Type 1	Gen Type 2
1	Link 1 inboard	To Body/Ground	Link 1	Ground	1	0	0	0	0	0	0	-	-
2	Link 1 outboard	Solve Direct (sphere)	Link 1	Upright	2	1	0	0	4	6	8	-	-
3	Link2 inboard	To Body/Ground	Link 2	Ground	3	0	0	0	0	0	0	-	-
4	Link2 outboard	Solve Direct (sphere)	Link 2	Upright	4	3	0	0	2	6	8	-	-
5	Link 3 inboard	To Body/Ground	Link 3	Ground	5	0	0	0	0	0	0	-	-
6	Link 3 outboard	Solve Direct (sphere)	Link 3	Upright	6	5	0	0	2	4	8	-	-
7	Link 4 inboard	To Body/Ground	Link 4	Ground	7	0	0	0	0	0	0	-	-
8	Link 4 outboard	Solve Direct (sphere)	Link 4	Upright	8	7	0	0	2	4	6	-	-
9	Link 5 inboard	To Body/Ground	Link 5	Ground	9	0	0	0	0	0	0	-	-
10	Link 5 Outboard	Solve Direct (sphere)	Link 5	Upright	10	9	0	0	2	4	6	-	-
11	Spring damper to body	To Body/Ground	Ground	-	0	0	0	0	0	0	0	Spring 1 to body	Damper 1 to body (also strut top)
12	Spring damper to upright	Solve Post (Vector Pos)	Upright	-	0	2	4	6	0	0	0	spring 1 to suspension	Damper 1 to suspension
13	Stub Axle	Solve Post (stub axle)	Upright	-	0	2	4	6	0	0	0	Stub axle	-
14	Wheel Centre	Define Z Pos (wheel Centre)	Upright	-	0	2	4	6	0	0	0	Wheel centre	-

Data Fill Complete to Level 3 using auto-fill routines.

We could now use this template to create a new model, using the *File / New* option in the normal way (remember: because we have not identified an input steering point, it will only appear in the *Rear* suspension list).

If we did use this template in its current form, because we haven't associated any graphics with this template, we would only see the hard points, tyre graphic and spring/damper graphic. We need to define template graphics.

Before we cover template graphics, save the currently defined values for our new template to a user file. From the template editor, select *Data / Custom Template Save*. Confirm the 'save' event and locate the required file folder and name using the standard browser



Save custom template message.

15.8 Template Graphics

When a template is used within a model, each has its own associated 3D graphics. Apart from the generic graphics items, such as the wheel/tyre, spring, and damper, the remainder such as links and wishbones are specified by the fourth data set of the template specification 'Graphics'.

Currently nine basic graphics primitives are available: Line, Cylinder, Circle, Sphere, Facet, Plane, Distance, Component and Angle. Each of these graphical types is drawn based on hard point co-ordinates. Each primitive type has its own set of property values, and some may be defined in a number of ways. A full list of the current options is given below, with further details of some specific examples following this list.

Line Graphic Classes:

Pnt-Pnt Line: Adds a new Line graphical element to the selected ends' template. Two hard point picks are required, points need not be on the same part.

Pnt-Vector Line: Adds a new Line graphical element to the selected ends' template. Three hard point picks are required, a line is drawn through the first point who's direction is set by the vector defined by the second and third picks, points need not be on the same part. The first and second picks can be the same point. The line is drawn to a global clipped length.

Pnt-Xvector Line: Adds a new Line graphical element to the selected ends' template. One hard point pick is required, a line is drawn through the picked point in the global X axis direction. The line is drawn to a global clipped length.

Pnt-Yvector Line: Adds a new Line graphical element to the selected ends' template. One hard point pick is required, a line is drawn through the picked point in the global Y axis direction. The line is drawn to a global clipped length.

Pnt-Zvector Line: Adds a new Line graphical element to the selected ends' template. One hard point pick is required, a line is drawn through the picked point in the global Z axis direction. The line is drawn to a global clipped length.

Pnt-Plane-Norm: Adds a new Line graphical element to the selected ends' template. A line is drawn through the selected point in a direction normal to the selected plane. The plane is identified by three point picks. The line is drawn to a global clipped length.

Pnt-UserVector: Adds a new Line graphical element to the selected ends' template. A line is drawn through the selected point in a direction defined by a user vector. The line is drawn to a global clipped length.

Pnt-Vector^Vector Line: Adds a new Line graphical element to the selected ends' template. A line is drawn through the selected point in a direction defined by the cross product of two defined vectors. The line is drawn to a global clipped length.

Cylinder Graphic Classes:

Pivot: Adds a new Pivot graphical element to the selected ends' template. Two hard point picks are required, both points need not be on the same part.

Tube: Adds a new Tube graphical element to the selected ends' template. Two hard point picks are required, both points need not be on the same part.

Vector-Radius-Length: Adds a new cylinder graphical element to the selected ends' template. Drawn through the selected point in a direction defined by the second and third point picks. The radius and length of the cylinder are defined directly.

Pnt-Vector-Radius-Length: Adds a new cylinder graphical element to the selected ends' template. Drawn through the selected point in a direction defined directly by the user. The radius and length of the cylinder are also defined directly.

Circle Graphic Classes:

Pnt-Pnt-Pnt: Adds a new Circle graphical element to the selected ends' template. Three hard point picks are required through which is drawn a circle, both the circle centre and radius are calculated and displayed as part of the graphical display.

Cntr-Rad-Norm: Adds a new Circle graphical element to the selected ends' template. Three hard point picks are required. The circle is drawn centred at the first point of a defined radius and who's normal is defined by the second and third picks. The first and second picks can be the same point.

Cntr-Pnt-Plane: Adds a new Circle graphical element to the selected ends' template. Three hard point picks are required. The circle is drawn centred at the first point and is drawn through the second point, (i.e. defines the radius), in a plane that contains the third picked point. All picked points must be different.

Pnt-Normal: Adds a new Circle graphical element to the selected ends' template. Three hard point picks are required. The circle is drawn through the first point about the defined normal vector. All picked points must be different. The derived circle centre and radius is drawn as part of the graphical element display.

Sphere Graphic Classes:

Pnt-Pnt Radius: Adds a new Sphere graphical element to the selected ends' template. Two unique hard point picks are required. The sphere is centred at the first pick and the radius is set by the second pick.

Pnt Radius: Adds a new Sphere graphical element to the selected ends' template. One hard point pick is required. The sphere is centred at the pick and given the radius specified by the user.

Pnt-Pnt Dia: Adds a new Sphere graphical element to the selected ends' template. Two unique hard point picks are required. The sphere is centred at the mid point of the two picks, the radius being half the distance between them.

Pnt-Pnt-Pnt-Pnt: Adds a new Sphere graphical element to the selected ends' template. Four unique hard point picks are required. The sphere is drawn through the selected four points. Four points will define a unique sphere who's calculated radius and centre position is identified as part of the drawn graphical element.

Facet Graphic Classes:

Pnt-Pnt-Pnt Facet: Adds a new Triangular Facet graphical element to the selected ends' template. Three hard point picks are required; points need not be on the same part.

Pnt-Pnt-Pnt-Pnt Facet: Adds a new four-nodded Facet graphical element to the selected ends' template. Four unique hard point picks are required; points need not be on the same part. Whilst points need not be in a plane, any facet drawn of non-planar nodes is not fully define

Plane Graphic Classes:

Pnt-Pnt-Pnt Plane: Adds a plane graphical element to the selected ends' template. Three unique hard point picks are required; points need not be on the same part. All plane elements are drawn clipped to a global value, (which the user can edit).

Pnt-X-Y Plane: Adds an X-Y plane graphical element to the selected ends' template drawn through the selected pick. All plane elements are drawn clipped to a global value, (which the user can edit).

Pnt-X-Z Plane: Adds an X-Z plane graphical element to the selected ends' template drawn through the selected pick. All plane elements are drawn clipped to a global value, (which the user can edit).

Pnt-Y-Z Plane: Adds an Y-Z plane graphical element to the selected ends' template drawn through the selected pick. All plane elements are drawn clipped to a global value, (which the user can edit).

Pnt-UserVector Plane: Adds an plane graphical element to the selected ends' template drawn through the selected pick. The orientation of the plane is controlled by two user defined vectors. All plane elements are drawn clipped to a global value, (which the user can edit).

Distance Graphic Classes:

Pnt-Pnt Dist: Adds a point to point distance graphical element to the selected ends' template. Any two hard point picks are required; both points must be on the same suspension corner. The display shows the total distance between the two points.

Pnt-Line Dist: Adds a point to line distance graphical element to the selected ends' template. Any three hard point picks are required; all points must be on the same suspension corner. The last two picks define the required line. The display shows the total perpendicular distance between the point and the line.

Line-Line Dist: Adds a minimum distance between two lines graphical element to the selected ends' template. Any four hard point picks are required; all points must be on the same suspension corner. The first two picks define one line whilst the last two picks define the other required line. The display shows the minimum normal distance between the two lines as a total distance.

Pnt-Plane Dist: Adds a points' distance from a plane as a graphical element to the selected ends' template. Any four hard point picks are required; all points must be on the same suspension corner. The first point is the required point whilst the last three picks define the required plane. The display shows the normal distance between the point and the plane as a total distance.

Components Graphic Classes:

Pnt-Pnt Comps: Adds a point to point distance graphical element to the selected ends' template. Any two hard point picks are required; both points must be on the same suspension corner. The display shows the distance between the two points in its x, y and z components.

Pnt-Line Comps: Adds a point to line distance graphical element to the selected ends' template. Any three hard point picks are required; all points must be on the same suspension corner. The last two picks define the required line. The display shows the perpendicular distance between the point and the line in its x, y and z components.

Line-Line Comps: Adds a minimum distance between two lines graphical element to the selected ends' template. Any four hard point picks are required; all points must be on the same suspension corner. The first two picks define one line whilst the last two picks define the other required line. The display shows the minimum normal distance between the two lines in its x, y and z components.

Pnt-Plane Comps: Adds a points' distance from a plane as a graphical element to the selected ends' template. Any four hard point picks are required; all points must be on the same suspension corner. The first point is the required point whilst the last three picks define the required plane. The display shows the normal distance between the point and the plane in its x, y and z components.

Angle Graphic Classes:

Pnt-Pnt-Pnt Angle: Adds an angle between three identified points as a graphical element to the selected ends' template. Any three hard point picks are required; all points must be on the same suspension corner. The middle picks is the point for which the angle is given. The display shows the angle created by the three point picks in degrees.

Pnt-Pnt Z-Axis Angle: Adds an angle between two identified points and the Z-axis as a graphical element to the selected ends' template. Any two hard point picks are required; all points must be on the same suspension corner. The first pick is the point for which the angle is drawn at. The display shows the angle created by the two point picks in degrees.

Pnt-Pnt Z-Axis X-X Angle: Adds a rotation angle of a vector defined by two identified points and the Z-axis as a graphical element to the selected ends' template. The angle is the rotation angle from the Z-axis around the X-X axis. Any two hard point picks are required; all points must be on the same suspension corner. The first pick is the point for which the angle is drawn at. The display shows the angle created by the two point picks in degrees.

Pnt-Pnt Z-Axis Y-Y Angle: Adds a rotation angle of a vector defined by two identified points and the Z-axis as a graphical element to the selected ends' template. The angle is the rotation angle from the Z-axis around the Y-Y axis. Any two hard point picks are required; all points must be on the same suspension corner. The first pick is the point for which the angle is drawn at. The display shows the angle created by the two point picks in degrees.

Pnt-Pnt X-Axis Angle: Adds an angle between two identified points and the X-axis as a graphical element to the selected ends' template. Any two hard point picks are required; all points must be on the same suspension corner. The first pick is the point for which the angle is drawn at. The display shows the angle created by the two point picks in degrees.

Pnt-Pnt X-Axis Z-Z Angle: Adds a rotation angle of a vector defined by two identified points and the X-axis as a graphical element to the selected ends' template. The angle is the rotation angle from the X-axis around the Z-Z axis. Any two hard point picks are required; all points must be on the same suspension corner. The first pick is the point for which the angle is drawn at. The display shows the angle created by the two point picks in degrees.

Pnt-Pnt X-Axis Y-Y Angle: Adds a rotation angle of a vector defined by two identified points and the X-axis as a graphical element to the selected ends' template. The angle is the rotation angle from the X-axis around the Y-Y axis. Any two hard point picks are required; all points must be on the same suspension corner. The first pick is the point for which the angle is drawn at. The display shows the angle created by the two point picks in degrees.

Pnt-Pnt Y-Axis Angle: Adds an angle between two identified points and the Y-axis as a graphical element to the selected ends' template. Any two hard point picks are required; all points must be on the same suspension corner. The first pick is the point for which the angle is drawn at. The display shows the angle created by the two point picks in degrees.

Pnt-Pnt Y-Axis Z-Z Angle: Adds a rotation angle of a vector defined by two identified points and the Y-axis as a graphical element to the selected ends' template. The angle is the rotation angle from the Y-axis around the Z-Z axis. Any two hard point picks are required; all points must be on the same suspension corner. The first pick is the point for which the angle is drawn at. The display shows the angle created by the two point picks in degrees.

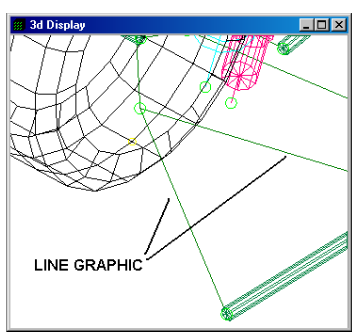
Pnt-Pnt Y-Axis X-X Angle: Adds a rotation angle of a vector defined by two identified points and the Y-axis as a graphical element to the selected ends' template. The angle is the rotation angle from the Y-axis around the X-X axis. Any two hard point picks are required; all points must be on the same suspension corner. The first pick is the point for which the angle is drawn at. The display shows the angle created by the two point picks in degrees.

'Line' Graphic:

This is a simple line joining two hard points. It could be representing a simple link or be part of a wishbone, the graphics of which are built up using a number of 'lines'. The properties of a 'Line' are;

Point 1	Hard point at start of line, (pick from list).
Point 2	Hard point at end of line, (pick from list).
Position 1	Sets association with either first or second part, (if applicable).
Position 2	Sets association with either first or second part, (if applicable).
Property 1	Offset in Global 'x' from Point 1 x-value
Property 2	Offset in Global 'y' from Point 1 y-value
Property 3	Offset in Global 'z' from Point 1 z-value
Property 4	Offset in Global 'x' from Point 2 x-value
Property 5	Offset in Global 'y' from Point 2 y-value
Property 6	Offset in Global 'z' from Point 2 z-value
Colour	Optional colour setting, (numerical 1-n).

The 'position' settings are used to show compliance deflections. A hard point that is associated with two parts, (by virtue of being the connection between them), can have in compliance solution mode two positions. The ability to pick either the first or second parts allows a visual representation of deflection of a bush by the separation of two graphics points.

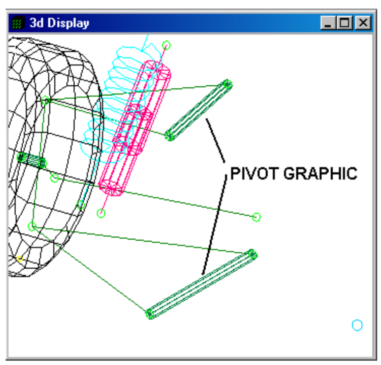


Examples of 'Line' graphics

‘Cylinder - Pivot’ Graphic:

This is a simple cylinder joining two hard points. Normally used to indicate a pivot axis, (also added automatically joining the wheel centre and the spindle point). The properties of a ‘Pivot’ are;

Point 1	Hard point at start of line, (pick from list).
Point 2	Hard point at end of line, (pick from list).
Position 1	Sets association with either first or second part, (if applicable).
Position 2	Sets association with either first or second part, (if applicable).
Property 1	Offset in Global ‘x’ from Point 1 x-value
Property 2	Offset in Global ‘y’ from Point 1 y-value
Property 3	Offset in Global ‘z’ from Point 1 z-value
Property 4	Offset in Global ‘x’ from Point 2 x-value
Property 5	Offset in Global ‘y’ from Point 2 y-value
Property 6	Offset in Global ‘z’ from Point 2 z-value
Colour	Optional colour setting, (numerical 1-n).

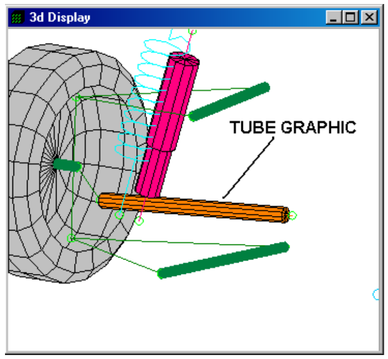


Examples of ‘Pivot’ Graphics

‘Cylinder - Tube’ Graphic:

This is a tube joining two hard points. Normally used to indicate a bar or tubular link. The difference between this and the ‘Pivot’ graphic is that the properties of the tube allow you to control the diameter and the axial offsets from the ends. The properties of a ‘Tube’ are;

Point 1	Hard point at start of line, (pick from list).
Point 2	Hard point at end of line, (pick from list).
Position 1	Sets association with either first or second part, (if applicable).
Position 2	Sets association with either first or second part, (if applicable).
Property 1	Outer diameter of tube.
Property 2	Offset of tube start from point 1 position, along the tube axis.
Property 3	Offset of tube end from point 2 position, along the tube axis.
Colour	Optional colour setting, (numerical 1-n).

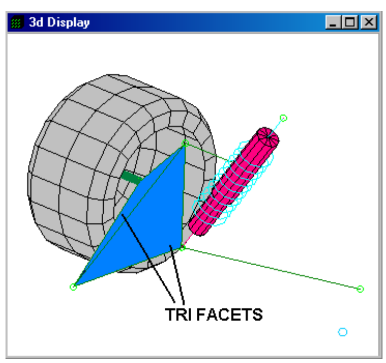


Example of 'Tube' Graphics.

'Facet – Pnt-Pnt-Pnt' Graphic:

This is a triangular facet joining three hard points. Normally these would be on the same body and used to indicate a face of a solid link. It will only appear as a 'filled' facet when the view fill style is set to something other than wire frame. In wire frame mode only the boundary of the facet is drawn. The properties of a 'Tri Facet' are;

Point 1	Hard point at first corner of the facet, (pick from list).
Point 2	Hard point at second corner of the facet, (pick from list).
Point 3	Hard point at third corner of the facet, (pick from list).
Position 1	Sets association with either first or second part, (if applicable).
Colour	Optional colour setting, (numerical 1-n).



Examples of 'Tri Facets' Graphics

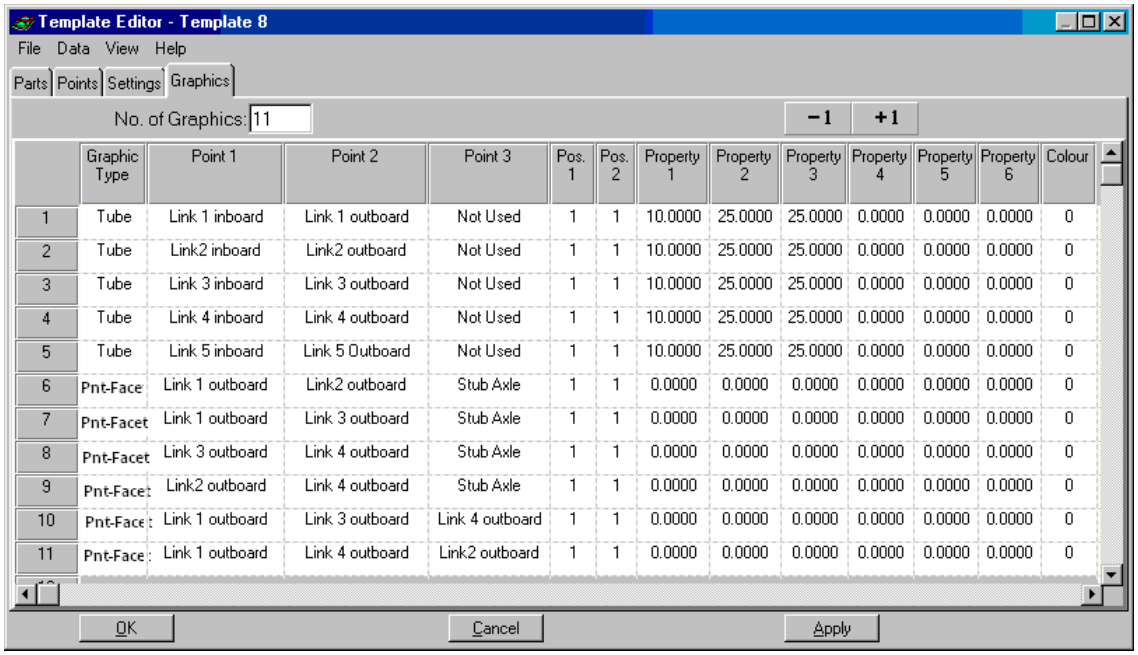
15.9 Exercise 3 – Adding Graphics to a Template

To complete the template we created earlier, we will now add some graphics to the template. Open the template editor, and select the correct template and change to the ‘Graphics’ tab. Currently, the number of graphical elements is set to zero.

For our template, add a ‘Tube’ graphic for each of the five links, and fill the upright with 6 Pnt-Pnt-Pnt facets. Remember we do not need to add graphics for the spring, damper, spindle axis or tyre as these are automatically drawn, using our point ‘gen type’ settings.

Remember to identify the required properties for each element, and select the relevant points.

Once complete, compare to the solution overleaf and re-save the completed custom template.



Graphical Element Settings for Five Link Template

The screen shot above shows the settings used, values have been entered for the tube diameters and end offsets, all other property values are left as zero.

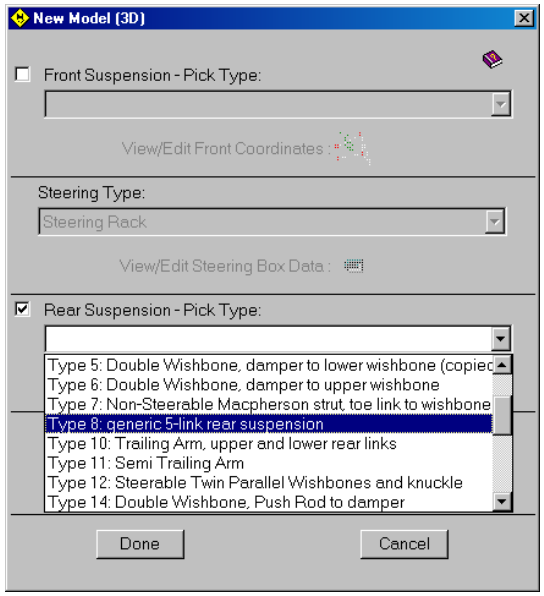
Make any alterations as necessary, and re-save your custom template. We can now use our template to create a model.

Note: You can add extra points to your templates that are used purely for applying graphical elements too. These ‘dummy’ points can be included to the template in the same way as any other point. Remember, because these points do not have any influence on the kinematic mechanism, they do not need to use one of the Pre-solve options that adds to the simultaneous equations set (since this would lead to an increase in solution time). These dummy points would normally be solved using ‘Solve Post (Vector Pos)’, Solve Post (Sphere) or ‘Solve via Hookes joint’.

Whilst the method outlined above is perfectly valid, the addition of convenience menus under sub-menu *Graphics / Add,* it is far easier to add graphical elements to templates by using the graphical interface to identify required graphics primitive, and then select points directly from the 3D view.

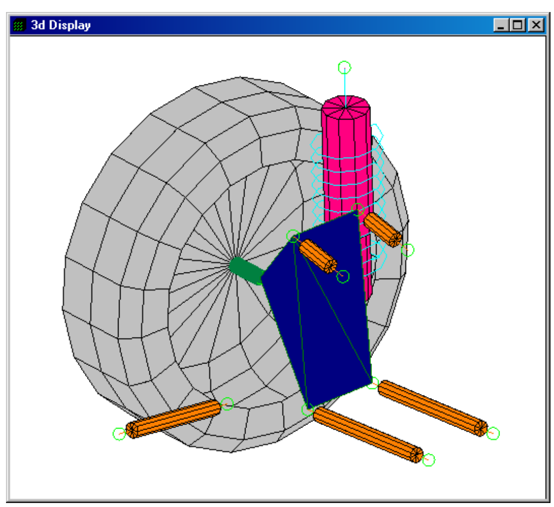
15.10 Using the New Template

We can now use our new template. Open a new model in the normal way, *(File / New*), and select our new template from the rear suspension list.



Selecting the new template.

Your new model should look like that shown below. This new template can be used as a generic template for all 5-link rear suspensions. We could, with a small change to the template, make it available for use as a steerable front suspension.



Screen Shot of Generic 5-link rear suspension template

Try modifying the template for use as a steerable front 5-link suspension.

16 External Application Auto-Search and Load

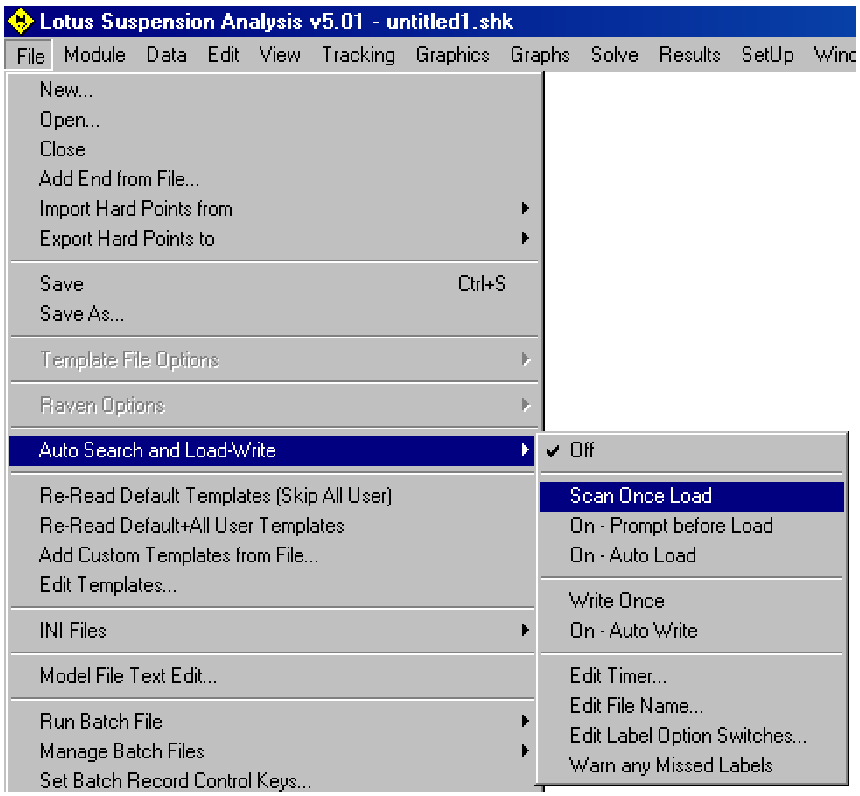
16.1 Overview

This chapter describes the use of the Auto-Search and load feature. It is a route by which an external application can update the co-ordinates of model hard points via an automatic software link. Conversely the same link can be used to export hard points from LSA out to an external application.

This chapter contains the following sections:

16.2 Introduction

The ‘auto-search and load’ function is a fully automatic software executed loop, that imports/exports hard point co-ordinates from/to some other external application. An example of this may be a CAD package that writes the new point positions of a modified suspension layout, and with Shark open at the same time, the suspension analysis model can be auto-updated, re-solved and all current graph results refreshed without any user interaction. This eliminates manually entering data and the associated opportunity for errors. Data Transfer is via an intermediary ASCII text file, and uses text strings to identify the points by comparison with the point labels.



Auto-Load Menu Options

16.3 Mode Types

The function can be run in one of five modes. ‘Scan Once Load’, ‘Prompt Before Load’, ‘Auto Load‘, ‘Write Once’ or ‘On – Auto Write’.

Off: Auto-Load is not enabled. This is the default setting

Single Scan Load: On selection of this mode, a once only scan is performed. The defined file is searched for. Each line of text in the file is then processed by identifying the string, and checking it against each point in the model’s label (and its other import labels).

Prompt before Load: This mode is a continuous scanning mode: at the defined time interval, the link file is searched for and checked to see if it has been modified since the last scan. If it has been modified, then the user is informed of the change and asked to confirm that it is okay to read and update the model. It is then opened and processed line by line in the same way as the previous option. When this mode is first set to 'On', the link file is searched for, and if found it is opened and processed.

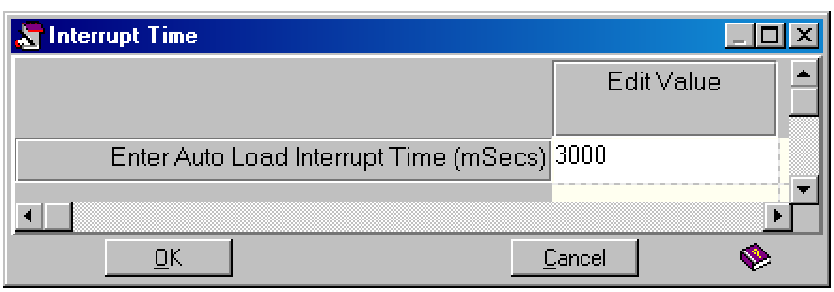
Auto Load: This mode is also a continuous scanning mode, checking for a modification of the link file at the prescribed time interval. When a change is detected, the file is processed, and the model updated without any user prompt. When this mode is first set to 'On', the link file is searched for, and if found it is opened and processed.

Write Once: This mode is the opposite of 'Load Once': instead of loading the specified file, it is written. In the case of writing, all points in the model are listed in the created file.

On – Auto Write: This mode is an extension of the Write Once mode: whenever a change is made to the LSA model (as detected by the edit buffer), the new model data is written to the defined file.

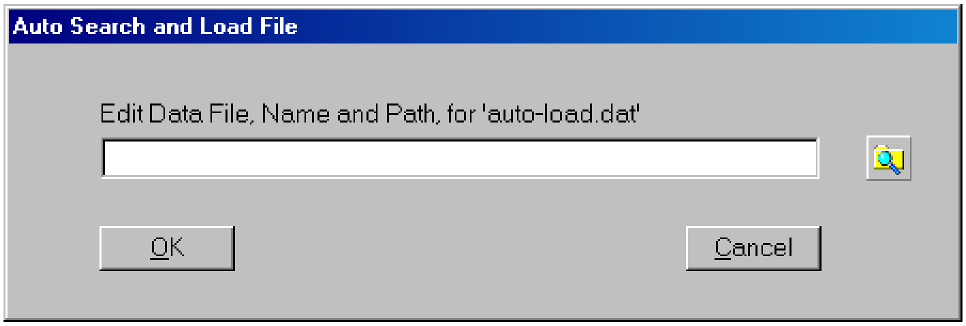
16.4 Settings

Timer: The frequency at which the scan for file modification is performed is controlled by this variable. The default setting is 3000 mSec. This setting is stored in the INI file.



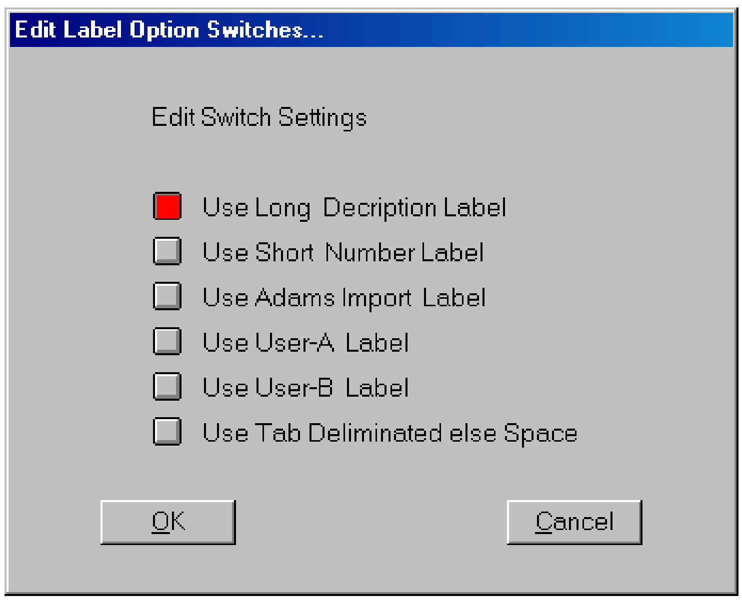
Interrupt Time Edit

File Name: Defines the full path and name of the link file used to transfer the hard point co-ordinates. This setting is stored in the INI file.



Setting the Link File full Path Name

Label Option Switches: Defines which point labels will be checked for in the scanned file. The settings for this allow either individual or multiple label cases to be scanned for.



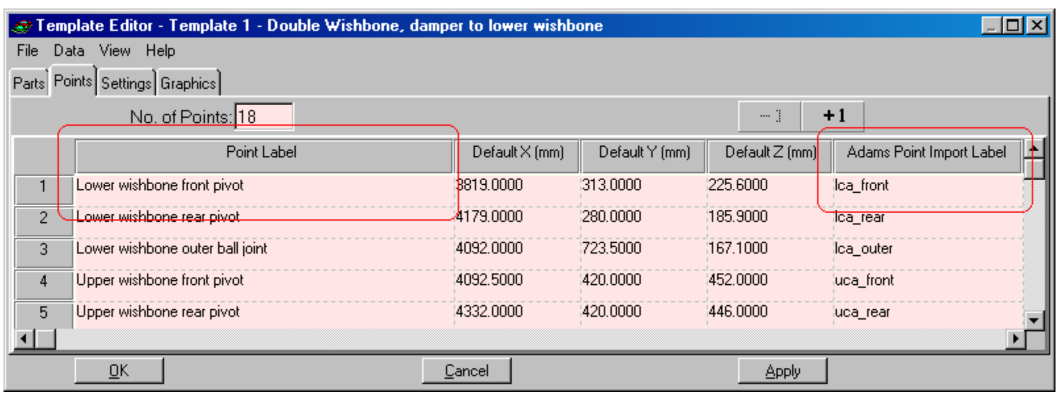
Setting the Point Labels to Scan for

Warn any Missed Labels: When enabled, this option flags the user when a line in the scanned file is not recognised by its label. The data loading is unchanged by having this setting on or off, the only change is that the user is informed that some lines have not been processed. Missed labels do not cause the import to stop: it will continue to process as many points as it can recognise.

16.5 Processing

The link file is processed on a line-by-line basis. The first record of the file should contain a version number (currently set as 1.0), this is intended to protect for future user specific requests for file formats. All subsequent lines are treated in the following way. The first part of the line should contain a string that matches the required point labels (as set by the Label Option Switches'). This must be followed by the global x value, y value and finally z-value. Note that z and y can be optionally omitted, i.e. the first value is assumed to be x, the second value if present is assumed to be y, and finally the third value if present is assumed to be z.

The point labels can be viewed/edited through the template editor: *File / Edit Templates*, select the 'Points' tab. The label 'matching' is case sensitive and requires the full label to be matched. Whilst this may seem excessively restrictive, i.e. why not allow partial match, it is rigorous and ensures parts of strings that are used in more than one label are not modified in error.



Editing the Point Labels – Template Editor

Based on interacting with the above template settings, an example 'link' file might contain the following lines;

1
Lower wishbone front pivot 3900.0 320.0 210.0
Upper wishbone rear pivot 4300.0 400.0

Note that you only need to specify as many points as required.

With any of the 'Auto-update' options turned 'on', you can continue to use LSA in the normal way, data will be changed either with or without prompting (depending on the mode selected), as you use the program.

As indicated in the 'Mode Types' section above, the processing can also be performed in the opposite direction, writing a transfer file on demand or whenever a change is applied to the LSA model.

16.6 Exercise: Running a Simple Case

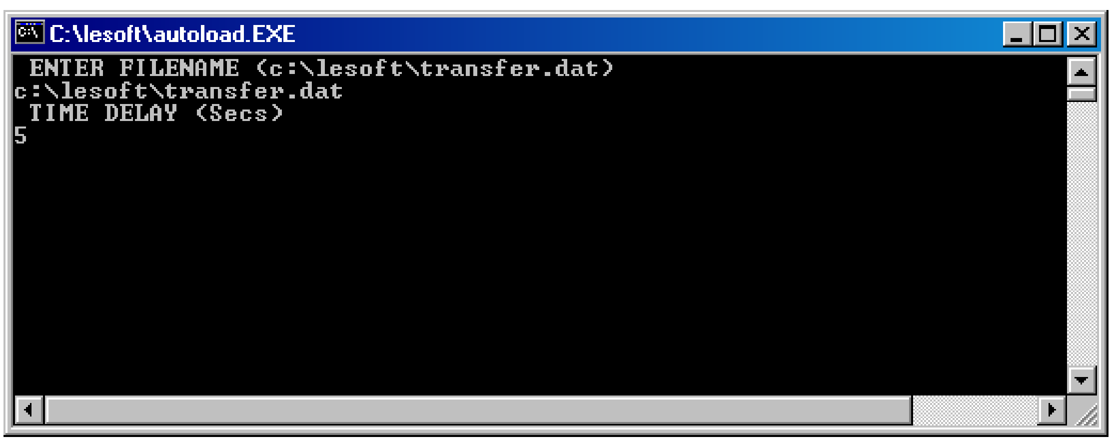
As a real example of how this works, we will use the simple executable ‘autoload.exe’ provided with the tutorial files. This file writes the co-ordinates at a user specified time interval, to a user-specified file for two points: “Lower wishbone front pivot” and “Lower wishbone rear pivot”.

- ***Open Shark and Select File / New.***
- ***Ensure “def –ve Y Side” is checked.***
- ***Select Front Suspension and set to Type 1 – Double Wishbone.***
- ***Select ‘Done’.***
- ***Open Two x-y Graphs, Camber and Toe, Autoscale the graphs.***

To run the simple executable ‘autoload.exe’, you can either run it from a ‘Windows command prompt’ or by double clicking on it through Windows Explorer. The text below assumes the later approach. This file can be found in the install sub folder datafiles, (default location c:/lesoft/datafiles/autoload.exe).

- ***From Windows explorer locate the file ‘Autoload.exe’ double click on it to start this simple external application.***

The external application opens a display and prompts for a linking filename and Time Delay. If it won’t run due to missing ‘salflibc.dll’ copy this down from the folder above.

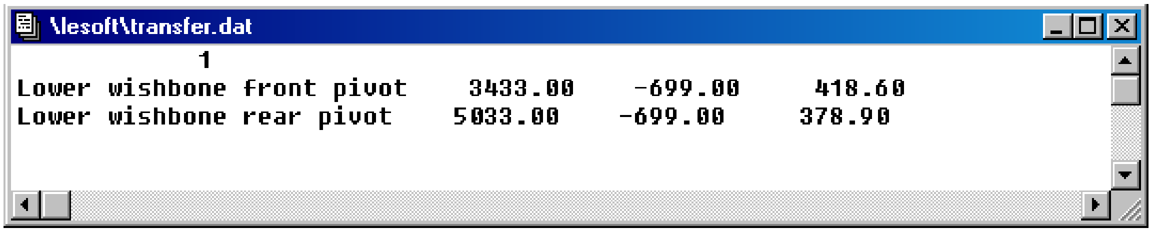


Screen shot of external application at startup

- ***Enter the filename, (select a suitable path and name), and enter the time delay as 5 seconds.***

Once you have entered this information, this simple application writes to the defined file at the defined time interval. A line appears in the display at each time increment to indicate the defined file has been updated.

- ***Leave Running.***

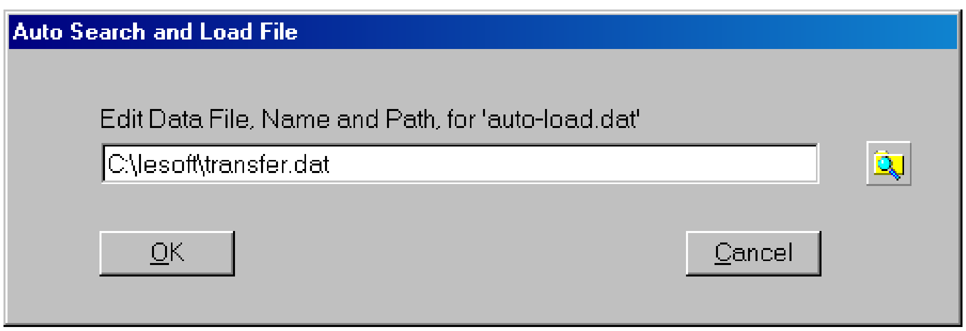


Example of data format of "Transfer.dat"

Now we need to activate the Shark end of the link.

- ***Open the Auto Search file name entry and set to your defined name.***

Hint: use menu 'File / Auto Search and Load-Write / Edit File Name'. Either locate with the browser or enter directly.



Setting the full path name for the link file

We will now turn the 'auto-search' facility 'on' and try the three alternative methods provided, 'Scan Once', 'On – Prompt before Load' and 'On – Auto Load'.

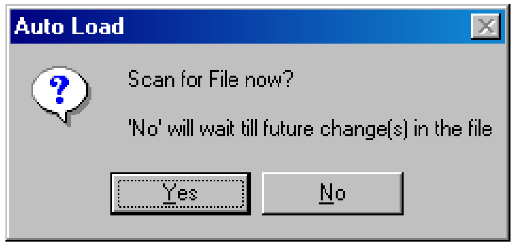
- ***Turn the Auto-Search facility on to the 'Scan Once Load' option.***

No message is given, but if the file is found, the data is scanned for and the model updated. Try repeating this a few times, to check that the external application is continuing to update the transfer file (remember you will need to leave at least the defined delay time interval for a change to occur).

Now try the second 'auto-search' option:

- ***Turn the Auto-Search facility on to the 'On –Prompt Before Load' option.***

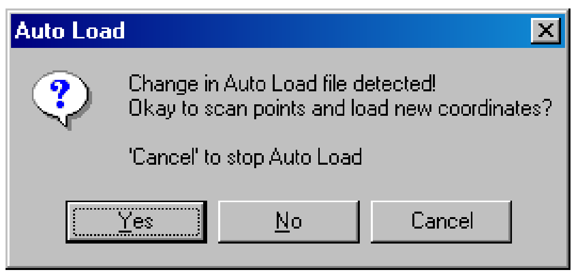
Initially you are prompted to choose whether LSA should perform an immediate scan for the transfer file, or wait until a future change occurs.



Wait for change prompt

- ***Select 'No'.***

Choosing the 'no' option means that you will wait for the next change. Once a change is detected, LSA will open a prompt to tell you that a change in the file has been noted.



Auto-load prompt

- ***Select 'Yes'.***

Choosing yes will scan the file and load the new co-ordinates. The displayed graphics and graphs should change to reflect this. Note that on the 'Auto-load' prompt, we can 'cancel' auto-load action. Next time the prompt appears we will cancel the action.

- ***Select 'Cancel'.***

Finally we will change to the fully automatic setting.

- ***Select 'Solve / Auto Search and Load-Write / On –Auto Load'.***

As before, the 'wait for change prompt' is displayed. Select either option then wait to check that the graphics and graphs now update at regular time intervals. Note that you can continue to use all the menus and functionality of the interface.

To stop the 'autoload.exe' program, simply close the windows command prompt via the top right corner 'x'.

17 Command Mode Operation

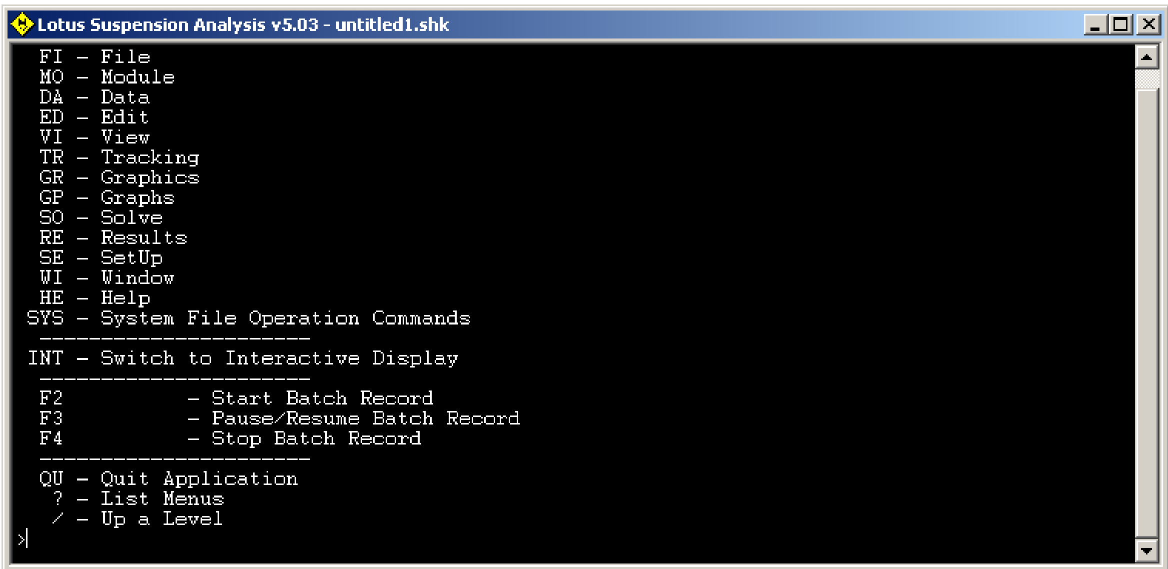
17.1 Overview

This chapter describes the use of the command mode. This mode does not rely on the main graphical user interface, but uses a simple command line display with associated short string commands to perform the analyses. The command mode is also the route towards batch files (see next chapter), and thus automated standard analysis cases.

This chapter contains the following sections:

17.2 Introduction

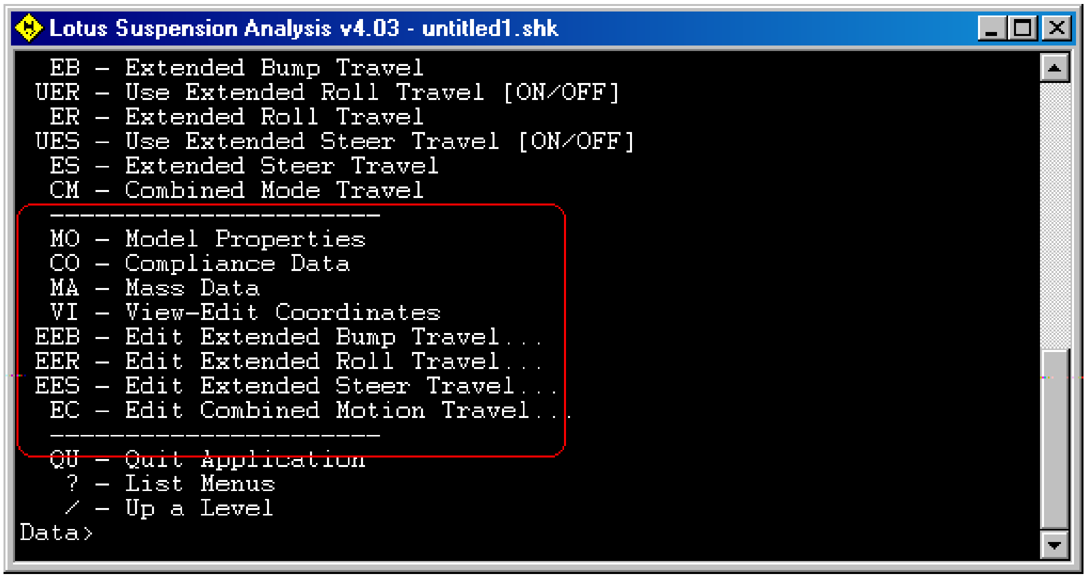
The batch mode was introduced as an alternative to the main graphical user interface. The program can switch between these two modes (command or graphical), or be started in either. Each mode has its own entry on the desktop and in the start menu.



Command Mode Display – Top Menu Shown

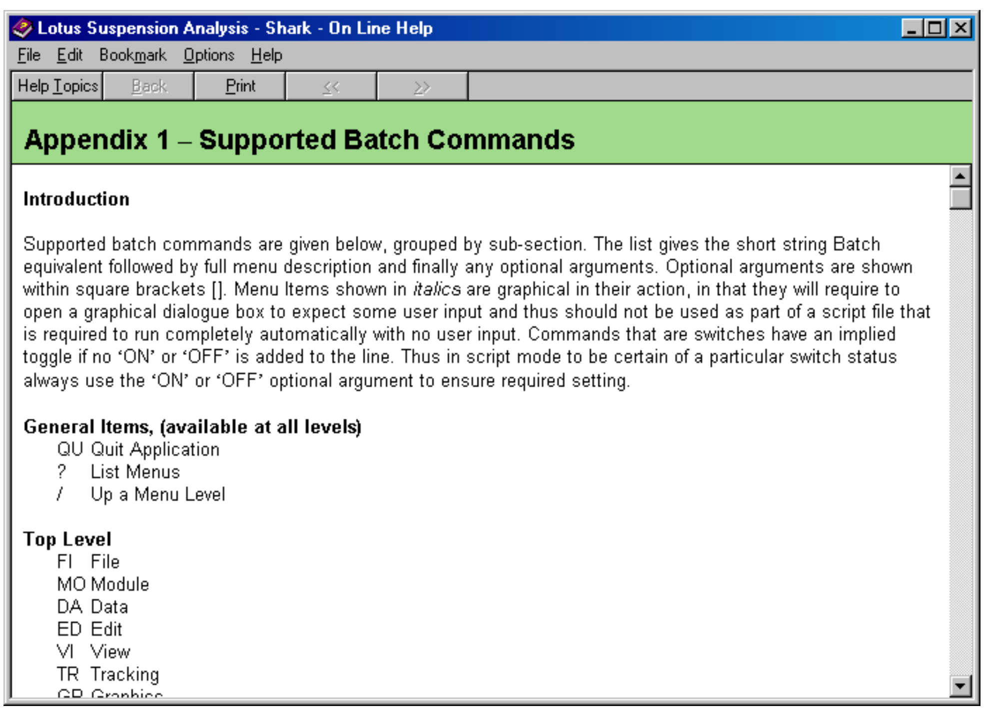
Each command in the main graphical interface has an equivalent short command string, for example FI is equivalent to 'File'. They are arranged in menu levels in the same way as the graphical interface, the user migrates up and down menu levels to the various sub menus. Command strings are mostly only 2 characters with a few being three characters long. Commands can be typed individually or strung together to provide a sequence of commands. General purpose commands have been included for listing the available menus and their short string equivalent (enter the ? character at the prompt) and for moving up a level in the command structure (enter the '/' character or enter '//' to move up to top level). The prompt string changes to indicate the current menu position and level. At the top level the simple '>' symbol is displayed.

Some commands have no non-graphical equivalent, the data being too complex to define/edit from the command line. In these cases a command is provided that opens the relevant full graphical dialogue box. Thus a command run can still access all data menus. Obviously these should not be used in an automated batch file, as they require user input. Command menu items that relate to a 'graphical' display, and hence requiring user input, are shown on the command list in a separate list, below the non-graphical options.



Command Mode Display – Example Graphical Menus 'Ringed'

Batch commands can be stored in a series of text files to provide 'standard' analysis tasks. This is covered in the next chapter. It is a natural extension of the command mode and the ability to string commands together. A complete list of the command short strings is given in appendix 1 of the on-line help.



Appendix 1 on-line help file – Supported Batch Commands

17.3 Basic Commands

Some examples of basic commands are given here. To create a new file for a rear suspension using template number 2 would look like;

```
//          ! ensure in top level
FI          ! change to File main menu
NE 2 3      ! new file template type 2 for end 3
```

This could have been strung together as;

```
//FI NE 2 3
```

To list to the command screen the formatted SDF results for end (corner) 3 using formatted set 1

```
//          ! ensure in top level
RE          ! change to Results main menu
FO          ! set to formatted SDF
LI 3 1      ! list for corner 3 using format set 1
```

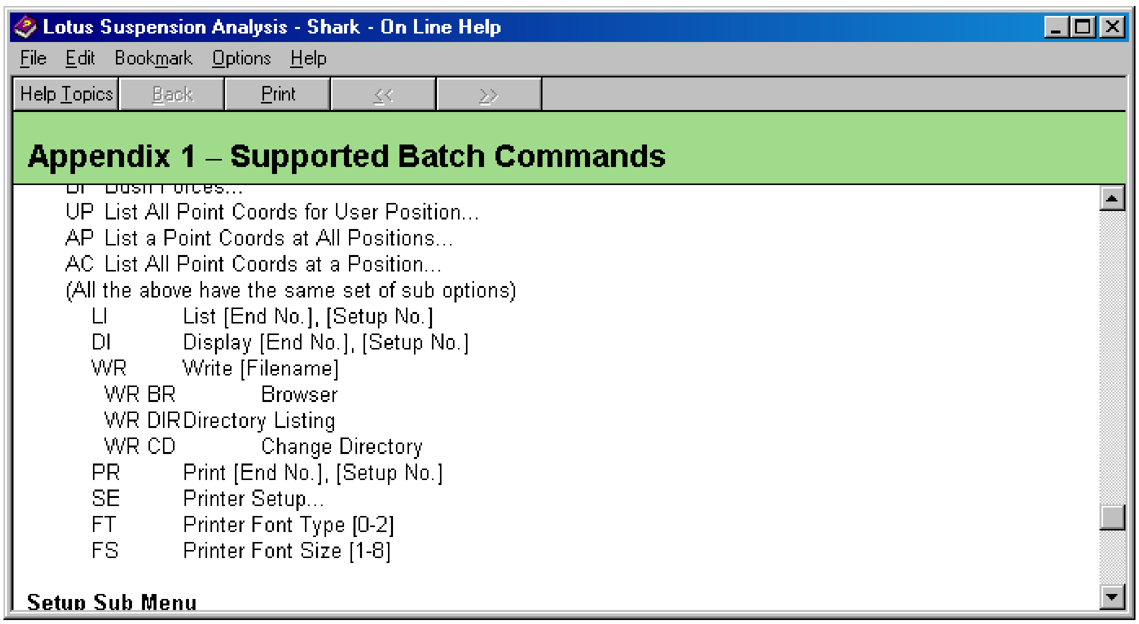
Or strung on to a single line

```
//RE FO LI 3 1
```

17.4 Command Arguments

Some commands require additional arguments. The ‘File / New’ command given above was an example of this requiring a template type number and the corner (or end) number. Note that some arguments are optional but that in some cases, the absence of an argument may be interpreted as reverting to the graphical mode. This is the case of the File/New command. If the template type and end is omitted, the application opens the File / New graphical display. In other cases, omitting an argument causes the particular command to not be applied.

The listing in Appendix 1 indicates the commands that require arguments, these argument terms are shown [ ] bracketed.



Appendix 1 on-line help file – Some Command Arguments

17.5 Switching Between Graphical and Command Modes

A user can switch between the command mode and the graphical (and back the other way), should this be required.

To open the command window from the full graphical interface, select from the main menu *File / Manage Batch Files / Open Batch Command Window*. This will open the scrollable command window, with the menu set at the top level.

If the command window was opened from the interface (in the manner described above), then to return to the graphical interface, users can close the batch window via the top right corner cross, select the ‘Esc’ key on the keyboard or at the top level enter the short command string ‘INT’.

If the application was initially opened in 'batch' or 'command' mode, and the full graphical interface has not yet been displayed, then the first two items mentioned above will close the application rather than open the graphical interface.

17.6 Exercise: Running a Command Mode Example

We will open the program in Command mode. Refer to the earlier text as to how you can do this. You may have an existing start menu entry or desktop icon or you may need to modify one of them to do this.

We will create a new model from the standard templates, run a bump and rebound solution and list the SDF spline results. Follow the sequence of commands and remember that you can use the '?' character at any time to list the available options.

- ***FI NE 1 1*** ***! opens a new model***
- ***// MO 3B*** ***! ensure set to 3D bump***
- ***// RE FI LI*** ***! list the SDF fitted spline fits***

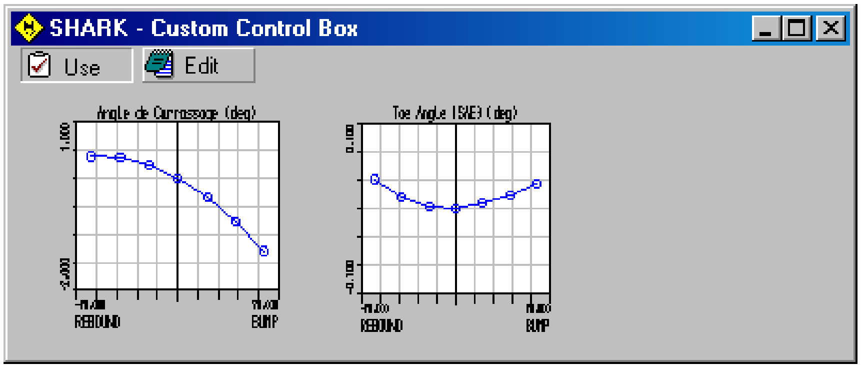
Now change the co-ordinates for a hard point and re-list the results.

- ***// DA PO LI*** ***! list the current hard points***
- ***?*** ***! show the available menu options***
- ***ED 3 4090.0 -725.0 168.0*** ***! change point 3 co-ordinates***
- ***// RE FI LI*** ***! lists the new SDF fitted spline fits***

We can create a new user formatted window to display some x-y graphs, even though we are in command mode.

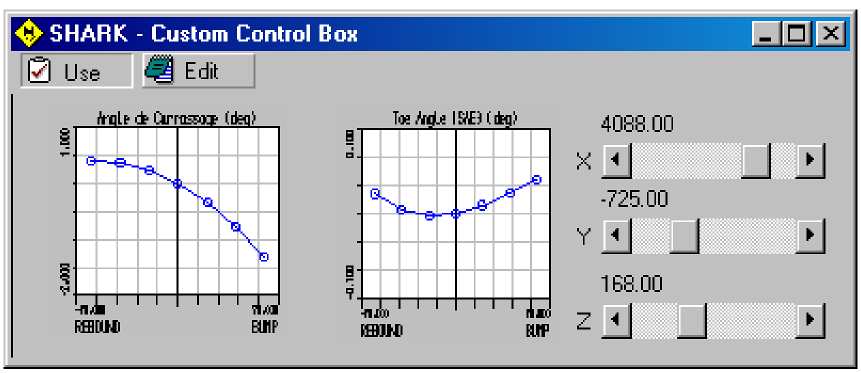
- ***// WI OP*** ***! opens a new user custom control box***

Select the Edit mode and from the right mouse menu select delete all to clear the display. Use the Add / Graph option to put two new graphs on the user window. Pick each in turn. Set the first graph to 'Camber Angle' and its Y-axis values to –2.0 to +1.0. Set the second graph to 'Toe angle (SDF)' and its Y-axis values to –0.1 to +.0.1 Then select the Use button to return to normal display of this single control window. It should look similar to the image below.



Simple two graph Custom control box

As an example of the linked functionality, we can add three sliders to this custom control box that will change the hard point x, y and z co-ordinates of the point 3 we changed earlier. Go back to 'Edit' mode and add three horizontal sliders. On their property sheet, set them to 'Point 3 X-coord', 'Point 3 Y-coord' and 'Point 3 Z-coord' with suitable slider limits and step sizes, we can use them to control the co-ordinates of this point. If we also add three 'value' displays that show the same three co-ordinate properties, we can produce a modified display similar to below.



Final two graph Custom control box with point sliders

The use of custom control boxes is also covered in section User Formatting Results.

18 Batch Files

18.1 Overview

This chapter introduces the concept of batch files. They are discussed as an extension to the Command Mode covered in a previous section. The batch file provides a method of creating and reusing ‘standard’ procedures in the analysis of suspension systems. This should provide a method by which different levels of users can all achieve the same results in both form and function in a rigorous manner.

The batch file is an extension of the short string command structure of the ‘command mode’ introduced in section Command Mode Operation. The batch file is an ASCII text file, in which each line has short string commands.

Different ‘batch files’ can be created for specific process tasks to provide a user with an independent method of modelling, solving and presenting suspension analysis results.

Batch files could be created via any standard text editor (a simple text editor is also provided within the software).

This chapter contains the following sections:

18.2 Running Batch Files from the System Prompt

Batch files can be run directly from the Command Prompt, by specifying the required file name after the ‘batch’ string. In its simplest form this might look like;

System prompt>shark.exe batch test1.bat

or
System prompt>shark.exe batch “test1.bat”

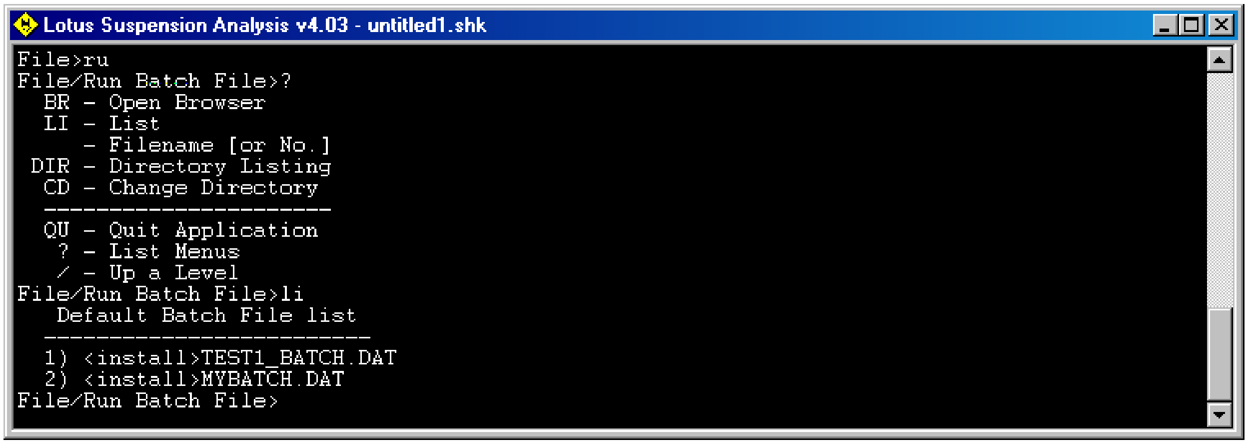
or
System prompt>shark.exe batch “<install>test1.bat”
Note: to be able to use the <install> string replacement, the “” characters are required around the file name.

Running a batch file from the Dos/Command Prompt

When a Batch files is run from the system prompt, the application will be left open unless the batch file includes the necessary quit (QU) command.

18.3 Running Batch Files in the Command Mode

Batch files can be run from within the application whilst in ‘Command Mode’ (not to be confused with the Windows Command Prompt). For example you would enter the following short command, FI RU test1_batch.dat or FI RU <install>test1_batch.dat



Running a batch file from the Command Mode

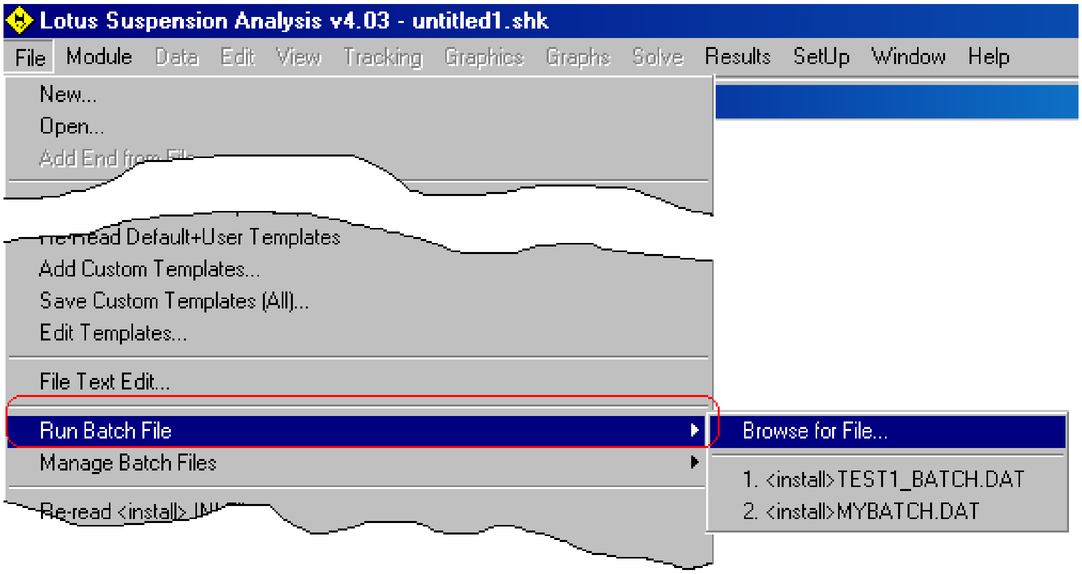
The application provides a list of standard 'batch' files. These can be added to/organised via a tool in the main graphical interface. In the command mode, you can list the standard files via the FI RU LI short string command. They are listed by number and can then be run either by using the filename, or more simply by its number.

Within the command mode, options are provided to browse (BR) for a batch file, list (DIR) current directory contents, or change (CD) the current directory. Note that for specific server based installations, the use of the '<install>' string is supported as part of the file name where '<install>' is automatically replaced with the actual installation folder location.

18.4 Using Batch Files in the Graphical Interface

Batch files can be run directly from the *File* menu. In the same way as with running in command mode a list of 'standard' batch files is given, together with the option to browse for a file. Running a batch file from the graphical interface will cause the 'command mode' scrollable display to be opened to enable the batch file to run. As it runs, the commands (and any requested lists) are echoed on to the scrollable display.

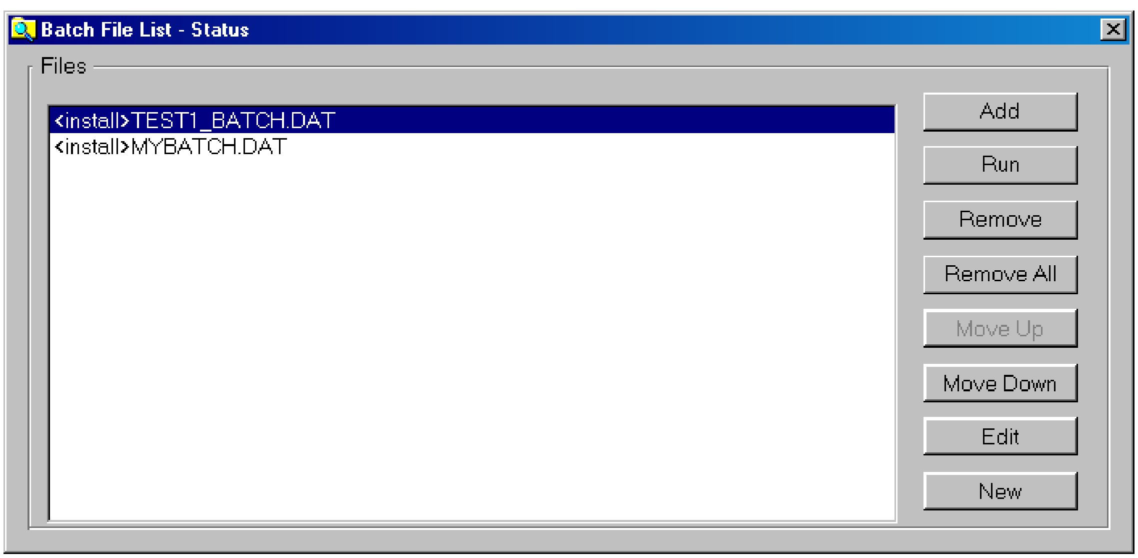
Once the batch run has finished the prompt, '* Batch Run Complete – Hit Return to Close' is displayed. This is to allow the user to scroll the display if required before it is closed. Whilst the batch file is running, the user cannot interact with the main graphical interface. It is effectively locked out until the batch run is complete and the command mode display closed.



Running a batch file from the Graphical Interface

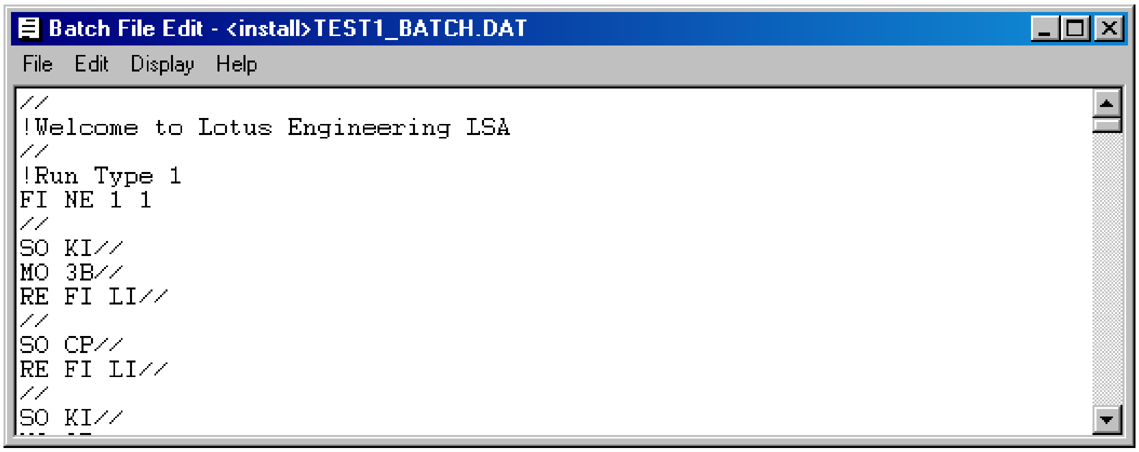
To add an existing batch file to the defaults list, either use the main menu option *File / Manage Batch Files / Add File to List…* and use the browser to locate the required file, or from the same sub menu open the *Batch File List Status…*

The 'Batch File list – status' display allows you to add other batch files in the same way as the previous item via a conventional browser. It also provides access to a number of other batch file features. These include changing the order of the files in the list, Remove (All) file, Edit a file from the list, Run a file from the list or create/edit a new batch file.



Batch File List Status display

To apply some of these actions, highlight the required file from the list: this will enable extra options. Most of these options are self-explanatory. The 'Edit' option opens a separate simple text editor to display the contents of the selected file (or empty for a 'new' file) to allow editing of the short string commands. Refer to Appendix 1 of the on-line help file for a full list of the short string commands.



Example batch file edit

Note the use of the '!' character at the start of the line allows an echoed comment to be added to the batch run.

18.5 Site Specific Batch Files

It is anticipated that individual user sites will be configured to provide a list of company wide 'standard' batch files. As a user, you may be instrumental in creating new 'standard' batch files, or you may just use batch files created by other users. In either case, you may need to understand where these batch files are stored (or could be stored) to ensure site wide access.

18.6 Exercise: Creating a Sample Batch File

We will use the internal text editor to create a simple batch file that we can then run from a Windows command line. This batch file will open a user located file, set the analysis mode, list formatted SDF results, change a model property, and list the new results. Open the application in interactive mode.

➢ ***File / Manage Batch Files / Batch File List Status***

➢ ***Select the local New button***

We will use comment lines in the batch file to add reading and act as local prompts.

Enter each line into the text editor

- ***!open the file using the browser (note assumes front model loaded)***
- ***//FI OP BR***
- ***!run the solver in 3d roll***
- ***//MO 3R***
- ***!list the formatted results***
- ***//RE FO LI 1 1***
- ***!change the rolling radius value***
- ***//DA TY ED "Rolling Radius (mm)" 245.0 1***
- ***!list the revised formatted results***
- ***//RE FO LI 1 1***
- ***!write the formatted result on c drive***
- ***// re fo wr "c:\exercise.txt" 1 1***

Then save the batch file to "exercise.dat" in a suitable folder. To test this batch file we can run it directly from the Batch File List – Status display. Highlight the saved file from the list and select the Run button.

Try running this batch file directly from the Windows command prompt. Remember to use the 'batch' argument with the batch file as the second argument.

- ***C:\…\shark.exe batch C:\…\exercise.dat***

19 Report Files

19.1 Overview

This chapter introduces the concept of ‘Report Files’: these are script files that allow a user to formulate the process of generating consistent reported output from the program. They rely on batch commands and batch files, so users should be familiar with these (see previous chapters). By combining the functionality of batch commands with additional format statements such as ‘new page’, different report formats can be merged into a single report document. In a similar way to batch files report files are run, edited and managed through a utility tool. Report files can be shared between users either through common file location or local copies of the same files. ‘Standard’ report files can be added to the interface menus and lists, and these lists are saved as part of the INI file. Reports created in this way can be sent straight to the printer or to a file. Alternatively they can be displayed in a rich text editor that provides the opportunity to edit/format the content before printing.

Report files are used to standardise and streamline the process of producing reports from the suspension analysis. They make use of batch commands and batch files to load, solve and list results, whilst additional formatting options such as line feed and new page are included to allow the creation of ‘standard’ report formats.

Report files are ASCII text files that, whilst they are similar in form to Batch files, have some specific layout and formats, and thus would not normally be edited through a simple ‘text’ editor. The interface provides a ‘manage’ tool, as with batch files, but the edit option opens a unique spreadsheet-editing tool.

This chapter contains the following sections:

19.2 Report File Items and Arguments

Report files are made up of a sequence of lines, each line defining an action, a result or some other relevant action. The following lists the available items and their associated arguments.

Single Line of Text: Adds a single line of text to the report document. Arguments are; text string, font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Line Feed and Font type.

Text file: Adds the contents of the supplied text file to the report document. Arguments are; file name, font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Line Feed and Font type.

Single Blank Line: Adds a blank line (hence implied line feed) to the report document. No arguments.

Single Space: Adds a single blank space to the report document at the current position. No arguments.

Single Character: Adds a single character to the report document at the current position and using the current font attributes. Single argument, the Character.

New Page: Adds a page break to the report document. No arguments.

Single Batch Command Line: Performs a batch command or series of batch commands. It does not add results to the report, it just allows for the required data changes, solver changes etc that may be required to enable the required results to be subsequently included. Arguments are; batch command string.

Batch Command File: In the same way as the ‘single batch command line’ this does not add results to the report. The defined batch file will contain command strings necessary to make data changes solver changes etc, so that the required results can subsequently be added to the report. Arguments are; batch command file.

Formatted SDF: Includes the specified corners Formatted SDF results in the report document using the defined format set. Arguments are; font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Font type, Corner number and format set number.

SDF Spline Fits: Includes the specified corners SDF Spline fits results in the report document using the defined format set. Arguments are; font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Font type, Corner number and format set number.

SDF Spline Data: Includes the specified corners SDF Spline data results in the report document using the defined format set. Arguments are; font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Font type, Corner number and format set number.

Bush Deflections: Includes the specified corners Bush Deflection results in the report document using the defined format set. Arguments are; font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Font type and Corner number.

Joint-Bush Rotations: Includes the specified corners Joint-Bush Rotation results in the report document using the defined format set. Arguments are; font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Font type and Corner number.

Bush Forces: Includes the specified corners Bush Force results in the report document using the defined format set. Arguments are; font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Font type and Corner number.

Formatted Point Forces: Includes the specified corners formatted point force results in the report document using the defined format set. Arguments are; font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Font type, Corner number and format set number.

List All Point Coords for User Position: Includes a list of all points for the specified corner at the defined user position. Arguments are; font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Font type, Corner number, bump travel, steer travel and roll travel.

List a Point Coords at All Positions: Includes a list of specified point for the required corner at all current solution points. Arguments are; font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Font type, Corner number and Point label (or point No.).

List All Point Coords at a Position: Includes a list of all points for the specified corner at the identified position. Arguments are; font colour, font size, Bold on/off, Italic on/off, Underline on/off, Strikeout on/off, Superscript on/off, Subscript on/off, Justify, Font type, Corner number and Position label (or position No.).

Insert User Window: Inserts a user window/control as an embedded image in the report. Arguments are; User Window No. (or User window label), Line feed and Justify.

Insert Visible Graph: Inserts a visible graph as an embedded image in the report. Only currently open graphs are available, so batch commands will need to be used to ensure they are open before being included. Arguments are; Graph No. (or SDF Label), Line Feed and Justify.

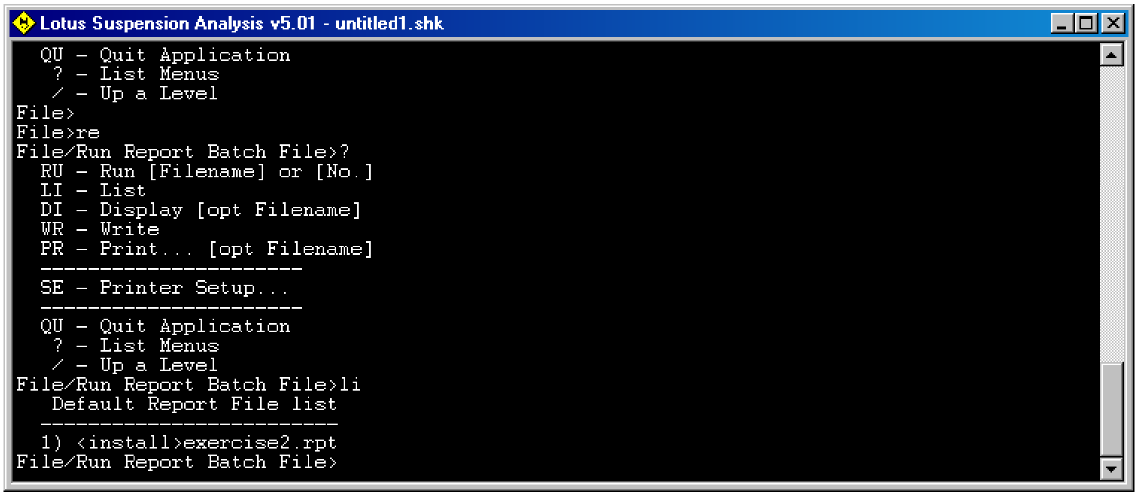
Insert Current Graphics: Inserts the current graphical view as an embedded image in the report document. Arguments are; Line Feed and Justify.

Insert Current AVI as File: Inserts the current animation sequence as an embedded AVI object in the report document, (included like this within a word document it can be viewed/animated directly from Word when the document is distributed). Arguments are; Line Feed and Justify.

Insert Ball Joint Target Rotation: Inserts the ball joint target graphical display for the specified point. Arguments are; Setting No, Corner (optional), Point No (optional), Point Short Label (optional), Line Feed and Justify.

19.3 Running a Report File in Command mode

Report files can be run from within the application whilst in ‘Command Mode’ (not to be confused with the Windows Command Prompt). For example you would enter the following short command, FI RE RU report1.rpt or FI RE RU <install>report1.rpt



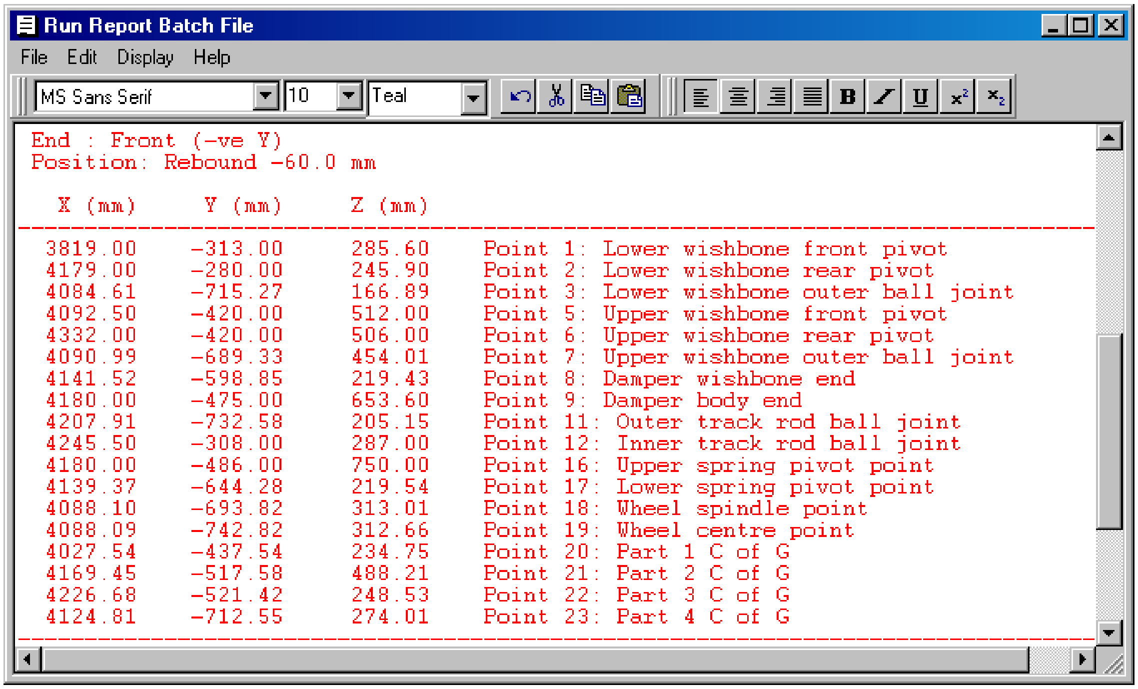
Running a report file from the Command Mode

The Run (RU) option will only run the report file. If you want to subsequently view the report file you will need to open the rich text display (DI). To print the current report, select the print (PR) command. Note that you can give both the display and print commands an optional filename, so that it will open or print using the new report file. Note that display and print does not currently support file number, it must be the file name.

The application provides a list of standard ‘report’ files. These can be added to/organised via a tool in the main graphical interface. In the command mode, you can list the standard files via the FI RE LI short string command. They are listed by number and can then be run either by using the filename or more simply by its number (see comment above on display and print options).

Within the command mode, options are provided to browse (BR) for a report file, list (DIR) current directory contents or change (CD) the current directory. Note that for specific server based installations the use of the ‘<install>’ string is supported as part of the file name where ‘<install>’ is automatically replaced by the software with the actual software installation folder location.

Printing is controlled by local printing properties, which can be edited through the local printer setup (SE) command.

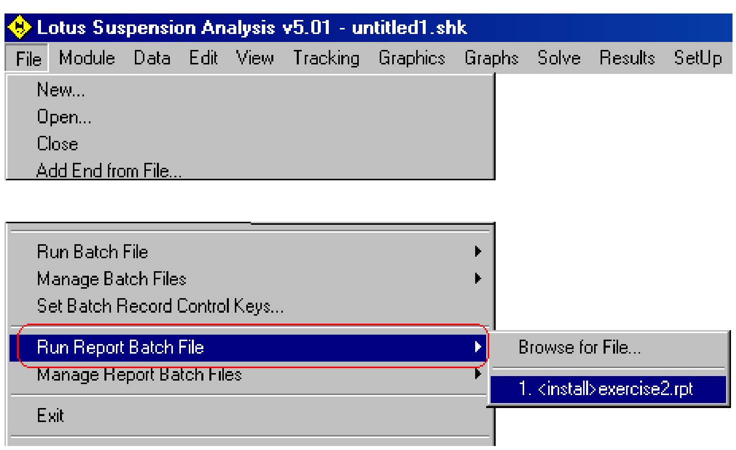


Sample Report Document – Created by a report file

19.4 Using Report Files in the Graphical Interface

Report files can be run directly from the *Results* menu. In the same way as with running in command mode, a list of 'standard' report files is given together with the option to browse for a file. Running a results file from the graphical interface will cause the 'results report' rich text display to be opened, and the created report document displayed.

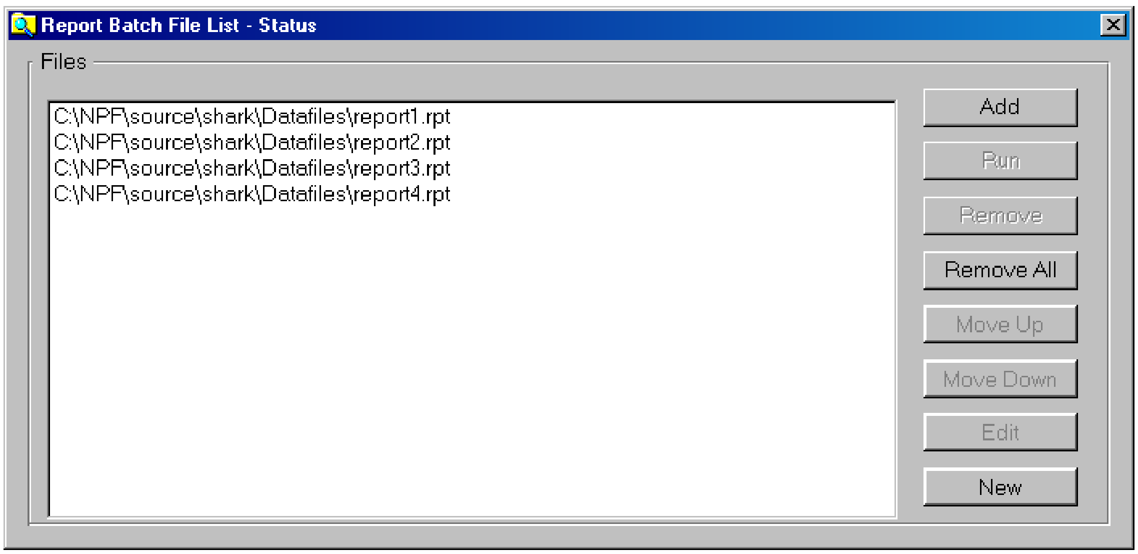
Once the report file has finished, the displayed report can be edited using the functionality of the rich text editor. Alternatively it can be sent to a printer, saved to a rich text document or opened directly in Word.



Running a report file from the Graphical Interface

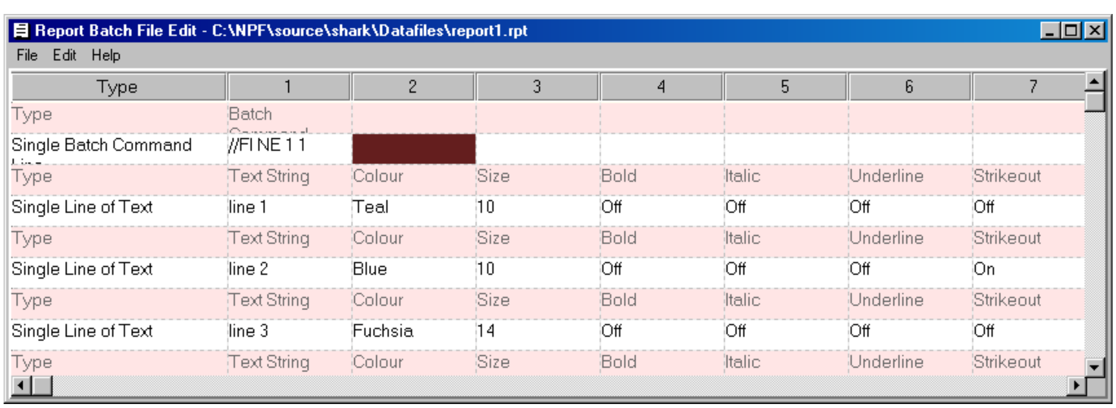
To add an existing report file to the defaults list, either use the main menu option *File / Manage Report Batch Files / Add Report File to List…*and use the browser to locate the required file, or from the same sub menu, open the *Report Batch File List Status…*

The 'Report Batch File list – status' display allows you to add other report files in the same way as the previous item via a conventional browser. It also provides access to a number of other report file features. These include changing the order of the files in the list, Remove (All) file, Edit a file from the list, Run a file from the list or create/edit a new report file.



Report Batch File List Status display

To apply some of these actions, highlight the required file from the list: this will enable the extra options. Most of these options are self-explanatory. The 'Edit' option opens a specific report file editor spread sheet to display the contents of the selected file (or empty for a 'new' file) to allow editing of the report format (see earlier discussion of feature items and arguments).



Example Report batch file edit

19.5 Site Specific Report Batch Files

It is anticipated that individual user sites will be configured to provide a list of company wide ‘standard’ report batch files. As a user, you may be instrumental in creating new ‘standard’ report batch files, or you may just use report batch files created by other users. In either case you may need to understand where these report batch files are stored (or could be stored) to ensure site wide access.

19.6 Exercise: Creating a Sample Report File

We will create a new report file that reproduces the previous chapters exercise, but instead of listing the results to the batch window, we will create a report file display.

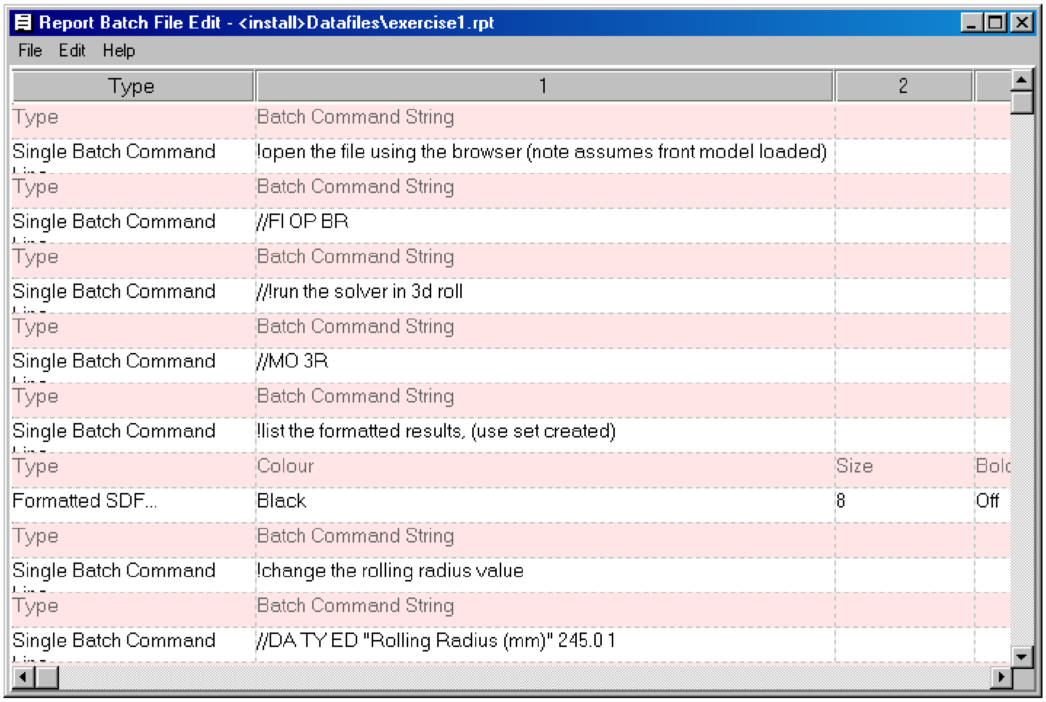
From the main interface, we will open a new report file, and use the local edit spread sheet, follow the entry below.

- ***File / Manage Report Batch Files / Report Batch File List Status…***

Select the New button to open the editor spread sheet for the new file. We will enter our batch commands in a similar way to previously, but use the ‘output’ selection for Formatted SDF rather than the batch command. Follow the steps outlined below.

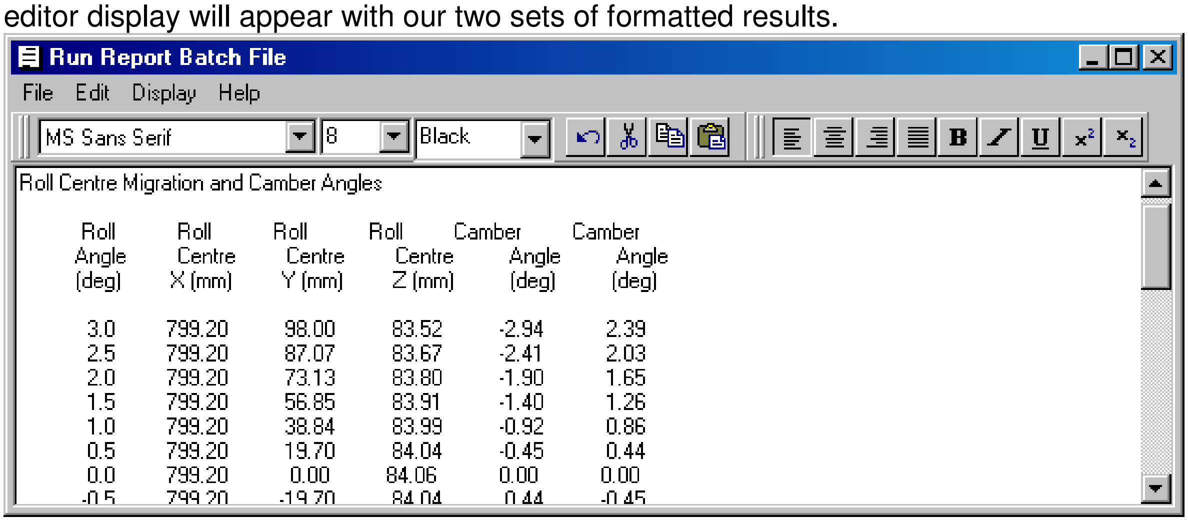
- ***Set type to ‘Single Batch Command’ and string as “//FI OP BR”***
- ***Set type to ‘Single Batch Command’ and string as “//MO 3R”***
- ***Set type to ‘Formatted SDF…’ set justify to ‘Left’ and setting to ‘set 5 Roll Exercise’***
- ***Set type to ‘Single Batch Command’ and string as “//DA TY ED "Rolling Radius (mm)" 245.0 1”***
- ***Set type to ‘Formatted SDF…’ set justify to ‘Left’ and setting to ‘set 5 Roll Exercise’***

The top half of the entry should look the same as indicated in the screen shot below. The comment lines are not compulsory.



Exercise Report batch file contents

Now save the report file to a suitable folder as “exercise1.rpt”. You will be given the option to add this report file to the list of default report files, select Yes. We can now run this report file from the status display. Highlight the file and select the run button. Select the Run – Go to Display option. Select an example file and the report rich text editor display will appear with our two sets of formatted results.



Exercise Report batch file contents

Try running some of the other ‘run’ options such as ‘open in word as rtf.

20 User Language

20.1 Overview

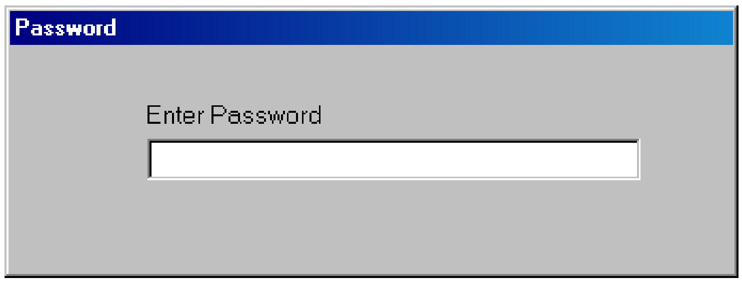
This chapter describes how users can create their own custom dictionary to replace a large number of text labels and variable names used by the interface. Because of the potentially large number of not only native languages but also local working practises, the hypothetical creation of custom dictionaries is to be carried out by the end user rather than the software vendor.

The default language for the interface is set to English. Through the *SetUp / Language / User Defined* menu option, an individual user (or customer site) can switch to their own settings. The user defined language is carried out on a string by string substitution. Each substitution required must be defined by the user and then becomes saved to the users "_Custom.Dic" file. On subsequent program re-starts, this file is searched for and reloaded automatically if found. Users only define as many of the strings as they require. The custom dictionary can also be passed to other users for sharing, or in some specific server based installations, all users might use the same custom dictionary.

This chapter contains the following sections:

20.2 Opening the Editor

To open the editor and start defining your own language/customer string settings from the *SetUp* main menu select *Setup / Language / Edit User Language*. This will open the language editor tool. In some installations, access to the editor is protected by a password. Consult your local support staff if it is the case.

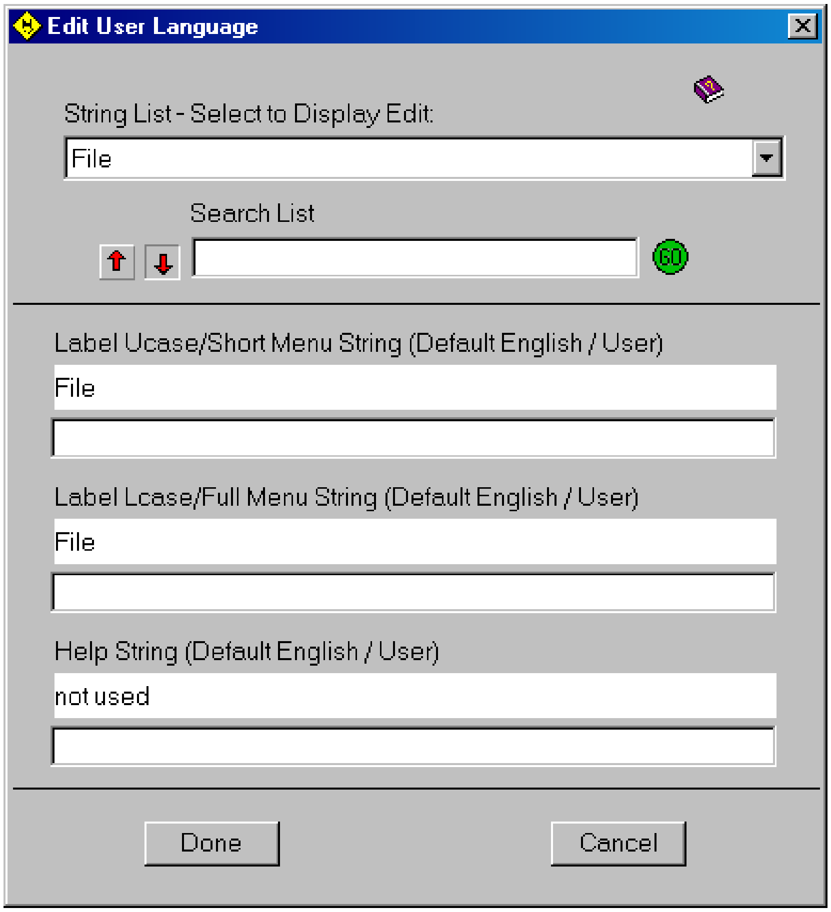


Optional Password Entry

20.3 Using the Editor

The editor tool has a simple selection box at the top, that users can scroll through every string entry used by the application (these strings has been made available to the custom language function). Each entry in this list has a maximum of three defining strings. Depending on the exact use of the string, the first two are either the UPPERCASE and lowercase options, or the Short and Long menu options. The third defining string, when used, is always the help string (this is for buttons and icons).

For each of the three strings, two text displays are shown. The top non-editable display shows the default English string whilst the second is editable and is where the user defined equivalent string should be entered.



Language Editor Tool

20.4 The Search Facility

A search facility allows the user to enter part or all of the required string to replace. The ‘go’ button then performs a search either up or down the list (depending on the arrow settings) for the next occurrence (if any) of the search string. If the search string is found, the tool displays the matched entry that can now be edited. You can repeat the search on the same string or change to another.

20.5 Implementing the User Dictionary

To save the changes, select the ‘done’ button (‘cancel’ will lose the changes). The changes will only be fully applied when you restart the program. To use the user dictionary, ensure the *SetUp / Language / User Defined* menu option is checked. This selection setting is saved in the INI file, and is thus preserved for future use.

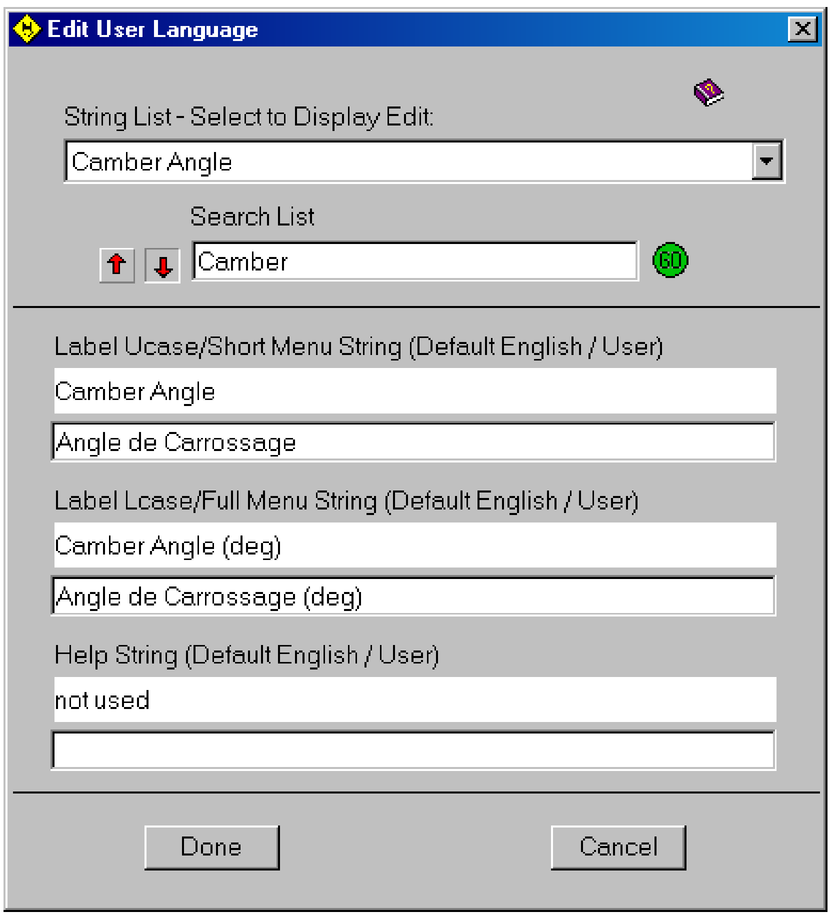
20.6 Exercise: Replacing 'Camber Angle'

As an example of how the user dictionary can be implemented, this exercise will replace 'Camber Angle' with the French equivalent 'Angle de Carrossage'.

First we will open the user language editor. If you need a password for this, either check with your site expert or for training courses, your course tutor. Remember the password file is case sensitive.

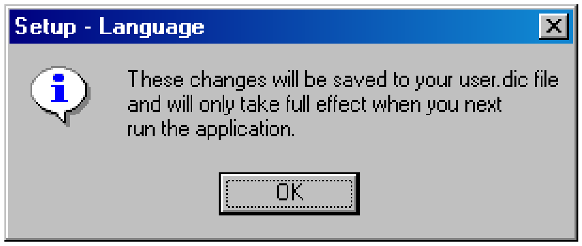
➢ ***SetUp / Language / Edit User Language.***

With the display open, enter camber in the 'search list' box, choose to search down and select Go. As each reference to camber is found via the search, change the string to the required 'angle de carrossage' for the full camber angle replacement, or just 'Carrossage' for camber on its own.



Screen Shot – User Language Equivalent of Camber angle

Repeat until all references have been suitably replaced. Then select the done button. This will only be applied to new menus, and only once the user language option has been enabled.

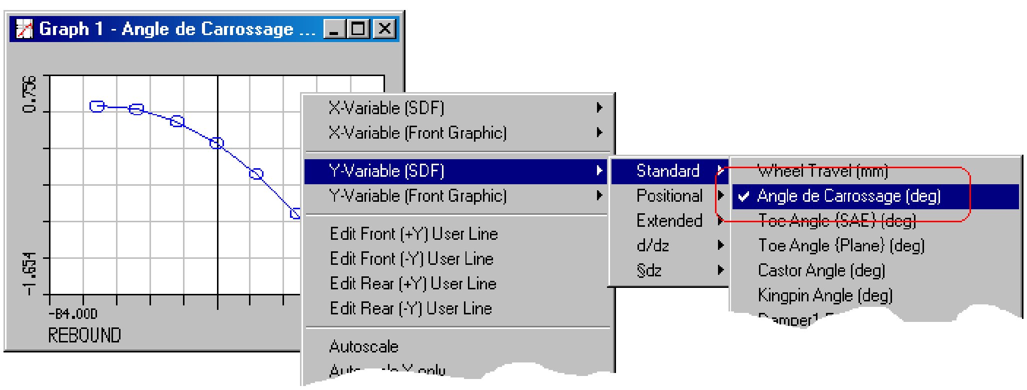


Informing the Effect of User Changes

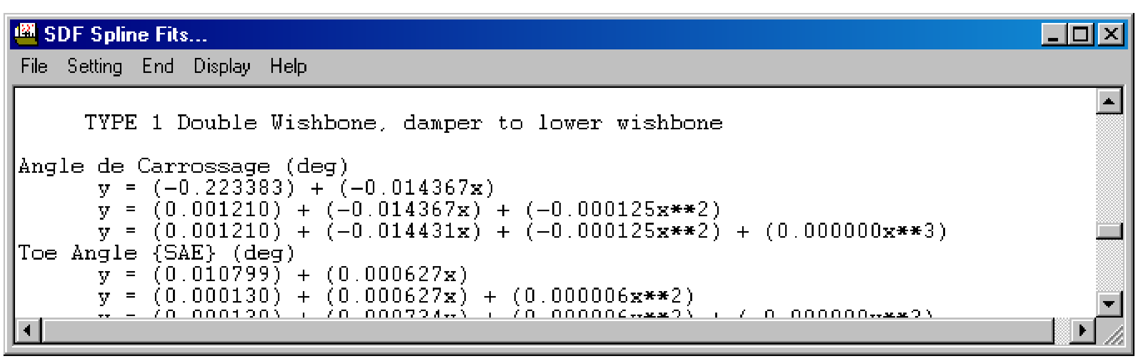
To enable the change to the user language setting, set the following menu option.

- ***SetUp / Language / User Defined.***

To test the user language implementation try opening a new x-y graph and use the right mouse list select the camber option. This should have been replaced with the supplied string.



x-y Graph Label Changed



SDF Spline fits label change

21 Full Suspension Model

21.1 Overview

In this chapter we will develop separate front and rear suspensions in two separate files, then combine the two to create a full vehicle suspension kinematic model.

A target hypothetical suspension characteristic has been set, and we will attempt to achieve a design that gives the best compromise solution. The target does not necessary represent desirable kinematics nor have other limitations such as packaging been considered. The suspension specification shown is a subset of all the dimensions / specifications required. All other data has been left as LSA default values.

This chapter contains the following sections:

21.2 Target suspension design

Front suspension: Type 1: double wish bone
Steering: Steering rack
Rear suspension: Type 1: double wish bone
Settings

	Front	Rear
Toe	0	0
Camber	-1.5	-1.5
Castor	3	0
KPI	5	5
Anti-dive	40	-
Anti-squat	-	44
Ackerman	80%	-
% braking	60%	40%
Roll centre height	75	100

Suspension travel

	Front	Rear
Bump	40	40
Rebound	40	40
Roll	2.5	2.5
Steering	30	-

Tire

	Front	Rear
Rolling radius	225	225
Width	150	150

Suspension specifications

Goals:

- Minimise roll and bump steer.
- Minimise movement of roll centre.
- Keep roll centre above ground at maximum bump.
- Camber less than zero at maximum body roll angle.
- Minimise tire scrub with bump

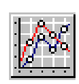
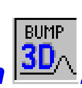
21.3 Tutorial 2.a

Open new front suspension model

- ***From the file main menu select 'File / New' to bring up the new model dialog box.***
- ***Select front suspension only with Type 1: double wishbone suspension and steering rack.***
- ***On the 'New Model' dialog left click on the 'View/Edit parameter data' icon*** ***.***
- ***Go through the list of parameters and modify: Bump travel, Rebound travel, Roll angle and Steer travel to match the data in the previous table.***
- ***Left click 'Ok' to close parameter dialog box then 'Done' to generate the new model.***
- ***To set the static toe and camber angles from the main menu select 'Data / Set Static Angles…' and input 0.0 for toe and –1.5 for camber, then click 'ok'.***
- ***Check that all of the 4 toolbars are displayed. If any are missing they can be displayed from 'setup' on the main menu.***

The model parameter and static settings have now been set up. We will now proceed to set up graphs to display the results as we manipulate the suspension geometry. Our goal is to manipulate the hard points to give kinematic motion in bump/rebound, steer and roll that gives the best compromise compared with our design goals.

Open Graphs

- ***Left click on the 'open new results graph' icon***  ***on the graphics+data toolbar to open a new results graph.***
- ***Right click on the new graph and select 'Y-Variable(SDF) / Standard / Camber angle'***
- ***Open another 5 graphs for toe angle(deg) {SAE}, castor angle (deg), King Pin Angle (deg), Roll Centre Height {To Grnd} (mm) and Half Track change. Once all the graphs are open arrange the 3d display window and the graphs so they can all be viewed simultaneously.***
- ***Select 3D Bump mode by left clicking the icon*** 

.

➢ ***Right click on each graph in turn and select 'AutoScale to Y increment' to auto scale each graph. (The axis scale and y increment size can be set to your own custom values by right clicking on the graph and selecting 'Axes Scales').***

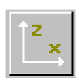
The castor angle and KPI will be modified first to achieve the specified values. For this tutorial we will modify the upper ball joint on the suspension upright.

➢ ***Turn off display of the wheels by clicking on the 'Toggle Enhanced Wheel Vis' icon*** ***.***

➢ ***Select joggle mode and left click on the top upright ball joint. You can now move the ball joint in the y direction. As you move the hardpoint (keyboard keys 'ctrl' + '->') the KPI graph will be updated. Move the ball joint to give 5 degrees KPI at zero bump as shown on the KPI graph.***

With the ball joint selected, the mouse right click will toggle through the available joggle directions in that view

➢ ***Right click on the KPI graph and select 'List data line(s)' to verify angle. Close the dialog when done.***

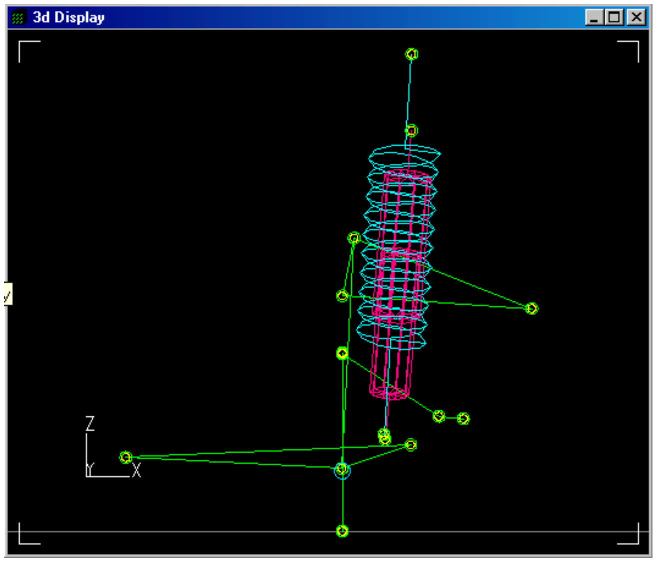
➢ ***Change the view to z-x*** 

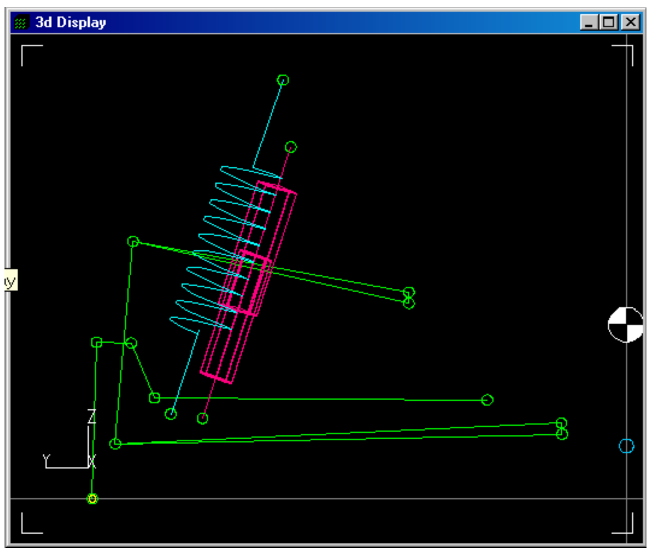
➢ ***Move the top upright ball joint in the x-direction to give 3 degrees castor angle.***

The castor and KPI are now set and you can manipulate the views to visually verify that the top of the King Pin axis is inclined inwards towards the centre line of the car and backwards toward the rear. The castor and KPI are also listed in the SDF output file available in the 'results menu'. Return to front view and 3d roll mode when done.

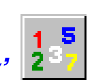
- Next we will manipulate the inboard suspension hard points to achieve desired suspension kinematics in bump and roll.

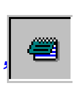
To achieve these characteristics, we will move the inboard suspension hard points on the upper and lower suspension wishbones, and the inner and outer steering ball joints. Before proceeding, experiment with moving these 6 suspension hard points and see if you can achieve a compromise between the target suspension kinematic characteristics. As movement of any one hard point can affect all of the characteristics we are interested in, you will need to iteratively move and adjust each hard point in front and side view until you have reached a best compromise. Figures below give an indication of one hard point configuration that satisfies this condition. Once you have finished experimenting, continue with the tutorial to input the suspension hard points manually.

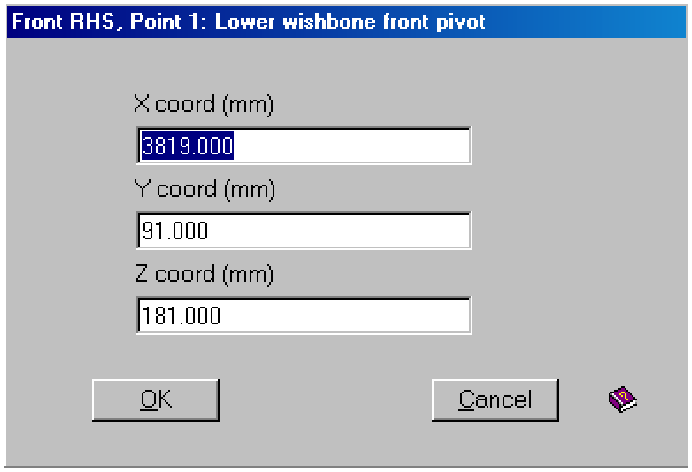




- ***Set viewing mode to front view y-z***

- ***Turn hard point numbering on by toggling the 'Point Nos Visibility'***  ***tool.***

- ***Select the 'Set to Edit Mode'***  ***icon and select the lower front inboard pivot: 'Point 1' and type in the hard point location shown.***



➢ ***Now go through each of the hard points listed in table bellow and use the same method to set the hard point locations listed in the table.***

Point	Name	X (mm)	Y(mm)	Z(mm)
1	Lower wishbone front pivot	3819	91	181
2	Lower wishbone rear pivot	4179	92	197
4	Upper wishbone front pivot	4092	308	385
5	Upper wishbone rear pivot	4332	308	369
9	Outer track rod ball joint	4214	668	233
10	Inner track rod ball joint	4245	197	230

Suspension hard points that require modification

With the suspension now located, toggle between 3D bump roll and steer modes and verify that the design criteria are satisfied.

➢ ***You can now verify yourself from the graphs and animating the suspension in steering, bump and roll that a compromise solution to the targets has been achieved.***

➢ ***From the main menu select 'File / Save As' and save the model as 'Tutorial 2a'.***

Now that we have setup the graphs, we can also save window settings to save which output graphs are displayed. The next time you use the model, you can load the windows setting file to reset the graphs.

➢ ***From the main menu select 'Window / Save Window settings To…'. Enter 'Tutorial 2a' as the file name and save the widows settings file.***

This completes tutorial one. The solution obtained is a compromise between all the desired and sometimes conflicting requirements of the suspension. You may want to experiment to see if you can improve the design. Also notice that the camber and castor angles achieved are not an exact match to the targets.

21.4 Tutorial 2.b

Open New Rear Suspension Model

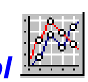
- ***From LSA menu select 'File / New'.***

- ***Check the rear suspension tick box and select Type 1 Double wishbone, damper to lower wishbone (we will add the front suspension later).***

- ***Check that the parameter data and tyre data are correct (you may need to modify Bump travel, Rebound travel, Roll angle and Steer travel to match the data in the first table) then click done on the 'New Model' dialog to open the new rear suspension model.***

Now we have a new rear suspension model open, we will setup a new graph to display % anti-squat for the rear suspension.

- ***From the main menu select 'Window / Load Windows settings From….'. Select the settings file 'Tutorial 2a' that we saved earlier and open.***

- ***Add another graph by clicking on the 'Open New Results Graph' tool***  ***and position the graph so it does not obstruct any of the other graphs.***

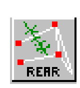
- ***Right click on the new graph and select 'Y-Variable (SDF) / Standard -/ Anti-Squat (%)'.***

- ***Auto scale the Y-axis. Right click on graph and select 'Autoscale To Y Increment'.***

The new graph settings can be now saved to the windows settings file.

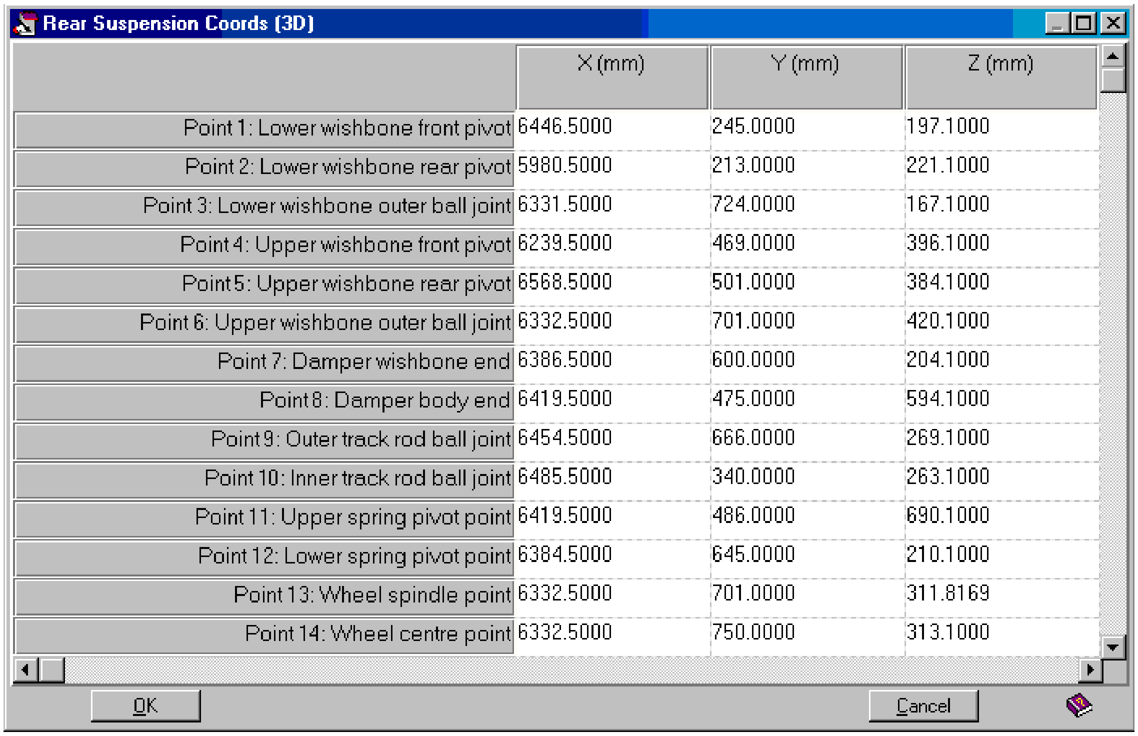
- ***From the main menu select 'Window / Save Window settings To…'. Enter 'Tutorial 2b' as the file name and save the widows settings file.***

Setup of the rear suspension follows the same procedure as the front therefore we will omit manipulating the suspension hard points and edit the rear suspension co-ordinates table directly. At this point you can experiment for yourself to see if you can meet the suspension characteristic targets, then continue from this point once you are done (you may want to save your own model before you proceed).

- ***Click on the 'View/Edit rear co-ordinates'*** 

icon to bring up the rear suspension co-ordinates table.

- ***Now you can manually edit each of the suspension hard point locations by directly editing the (X,Y,Z) co-ordinates of each hard point. Use table bellow for input data. Click 'OK' when finished.***



Rear Suspension Coords (3D)

	X (mm)	Y (mm)	Z (mm)
Point 1: Lower wishbone front pivot	6446.5000	245.0000	197.1000
Point 2: Lower wishbone rear pivot	5980.5000	213.0000	221.1000
Point 3: Lower wishbone outer ball joint	6331.5000	724.0000	167.1000
Point 4: Upper wishbone front pivot	6239.5000	469.0000	396.1000
Point 5: Upper wishbone rear pivot	6568.5000	501.0000	384.1000
Point 6: Upper wishbone outer ball joint	6332.5000	701.0000	420.1000
Point 7: Damper wishbone end	6386.5000	600.0000	204.1000
Point 8: Damper body end	6419.5000	475.0000	594.1000
Point 9: Outer track rod ball joint	6454.5000	666.0000	269.1000
Point 10: Inner track rod ball joint	6485.5000	340.0000	263.1000
Point 11: Upper spring pivot point	6419.5000	486.0000	690.1000
Point 12: Lower spring pivot point	6384.5000	645.0000	210.1000
Point 13: Wheel spindle point	6332.5000	701.0000	311.8169
Point 14: Wheel centre point	6332.5000	750.0000	313.1000

OK Cancel

Rear suspension hardpoints co-ordinate table

You can check for yourself that the new rear suspension co-ordinates give a 'good' compromise solution compared to the targets set out. To complete the full suspension model we will now proceed to add the front suspension from tutorial 2.a.

➢ ***From the main menu select 'File / Add End From File' and select the front suspension model developed in tutorial 2a.***

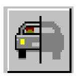
You now have a full vehicle suspension kinematic model, which satisfies the target kinematic characteristics.

When working with full suspensions, displaying both the front and rear suspensions can be visually confusing. LSA allows you to display only the front or rear suspension and also only one side of the suspension.

 Displays only the front suspension.

 Displays only the rear suspension.

 Displays both front and rear suspension.

 Toggles between displaying one side and both sides.

In this tutorial you have learnt all the basic operations for setting up, manipulating and analysing suspension kinematics. To complete this tutorial save your file and quit LSA.

➢ ***Save model as 'Tutorial 2b' and close Lotus Suspension Analysis to end tutorial 2.***

22 Importing/Exporting Hard Points

22.1. Overview

This chapter describes the use of a text recognition based system of importing hard point geometry to (and from) the Adams® Subsystem file. This section describes the process through which users can successfully transfer data between the application and external file format.

The option to extract data to/from data files from other applications is currently supported for one specific case. It is an ASCII text file, but but has its own unique format. The format supported is the Msc Software Adams sub system file.

Export is supported through the use of text string recognition. Each point in the LSA model has a matching text label in the exported data file. This text string equivalence is extended to include, bush axis points, part properties, bush properties and general model parameters such as rolling radius.

These tables of equivalent text strings are edited by the user, and saved within the INI file such that they are available for future re-use. The use of standard labels within models and subsystem files aids in the reusability of these settings.

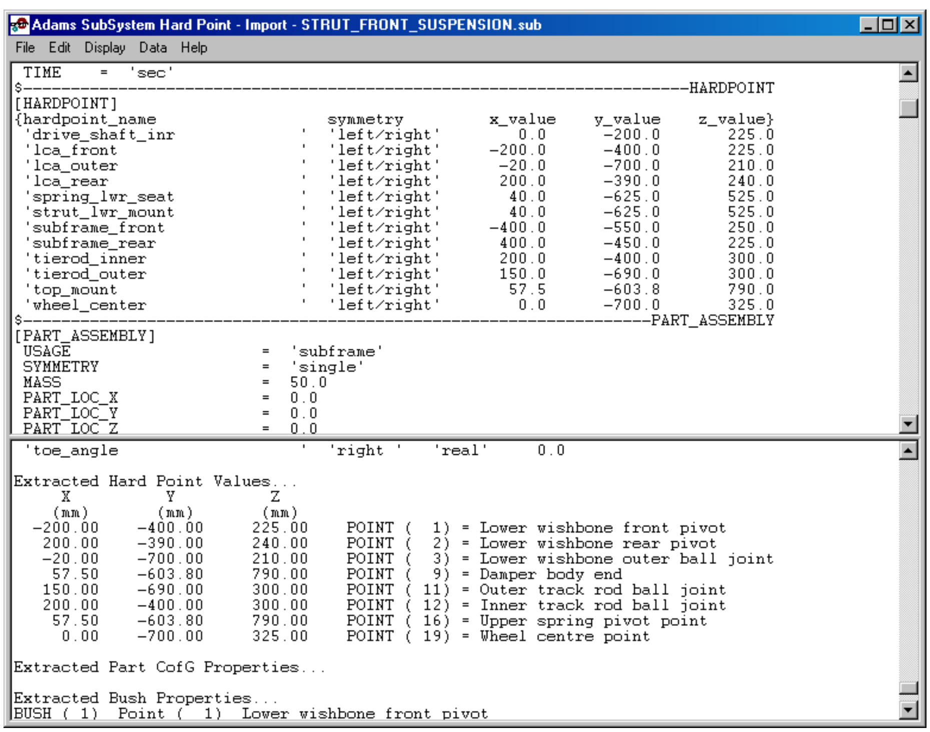
These import and export options only become enabled once a model has been specified via the new or the open commands.

This chapter contains the following sections:

22.2. Import and Export to an Adams Sub System

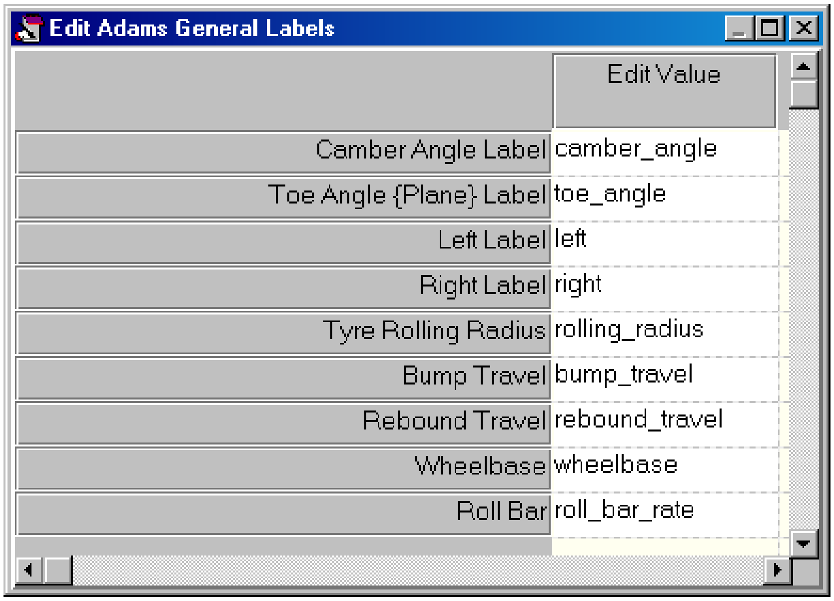
The use of this import export filter is obviously going to be limited to users who have a licensed copy of Adams. The data transfer is between LSA and the sub-system file normally having the “.sub” file extension. To open the split window display use the *File / Import Hard Points from / Adams Subsystem* or *File / Export Hard Points to /Adams Subsystem* as required. The opened display shows a split screen with two text area display. The upper area will be where the opened subsystem file is displayed, whilst the lower display for import will show a preview of the extracted text sections and extracted values. For export it will show a preview of the modified subsystem file with the new number substitutions.

For importing data, the normal order of actions would be, first ensure all the text strings are set for each section that you intend to import. The sections are; Hard Points, Bush Axis points, Part Mass Properties, Bush Stiffness and Parameter. Each section has its own set of strings to edit under the local *Data* menu entry. Secondly, open the Adams sub-system file that you require to extract the data from, via the local *File / Open (subsystem)* command. The subsystem file should now be displayed in the top display area. You can now preview the values that will be extracted from the subsystem file using the local *File / Import Hard Points (Preview),* the extracted numbers are previewed in the lower display area (note this is an optional step you can go straight to the next item). To complete the data import, select the local menu *File / Import Hard Points*. The display will close and the import number used to update the model.



Example Adams subsystem import screen shot – preview shown

As well as the import function having the matching strings, users can apply local shifting, scaling and axis switching to the data values.



Editing the 'General' label strings

The export option follows the same process as the import, except that the preview displays the modified subsystem file, rather than the extracted data values.

GROUP HEADQUARTERS

Group Lotus Plc
Potash Lane
Hethel, Norwich
NR14 8EZ
United Kingdom

Tel: 01953 608000
Fax: 01953 608300
Email Cars: carsales@lotuscars.com
Email Eng: engsales@lotuscars.com
Email Group: group@lotuscars.com