

Technical Newsletter

Available on-line in the EDC website Library at www.edccorp.com

HVE & HVE-2D - Ongoing Developments

HVE & HVE-2D Version 6 Now Available

Version 6.00 was released on May 20, 2008, and has been shipped as an update to all users with current Update/Technical Support policies. This release provides many new features including AutoConnect and the Steer Degree of Freedom model in *EDSMAC4*, vehicle mesh tessellation in *EDSMAC*, major enhancements to the *DyMESH* 3-D collision model and also new vehicles added to the *EDVDB* vehicle database.

Users who have not installed their updates should do so as soon as possible. Feel free to contact EDC Customer Service by phone at 503.644.4500 or by email to support@edccorp.com for assistance.

HVE V 6.1: Focus on HVE-2D Features

Development is already underway on exciting new features for the next version of *HVE* and *HVE-2D*. Also good news: Version 6.1 is scheduled for the 4th quarter of 2008. New features include enhancements to the user interface especially for the benefit of *HVE-2D* users, and also extensions to physics capabilities for the benefit of all users.

Here are just a few of the upcoming features:

- Users will be allowed to change the vehicles used in an existing event and also to add or remove vehicles from an existing event. Previously, the user was required to create a new event in order to change the vehicle or the number of vehicles in an event.
- The Driver Tables will offer shortcuts allowing the user to quickly duplicate rows or to apply a value to all columns in the same row.
- The visualization of skidmarks will be enhanced using a variable skidmark width calculated according to current slip angle, inclination angle and width of the tire.
- Visualizing the traveled path of each vehicle in a simulation will be easily performed using a new option to display tire tracks of a vehicle moving across the environment.

- The modeling of connections between vehicles will be enhanced by providing for attributes for maximum articulation angle, connection stiffnesses for roll, pitch and yaw, plus also assigning a breakaway force and torque that results in separating the vehicles.
- More extensive paths for the *HVE* Driver Model to follow can be created using a data entry table which allows for a large number of targets. Additionally, the attempted path can be visualized in the Event Viewer.
- Assigning Set-up options, such as braking inputs, wheel damage or tire blowout for towed vehicles will be easier for the user as the names of the towed vehicles will now be included and directly selectable in the Event Objects List.

This new version will be released within the next six months. Visit www.edccorp.com for further information.

Validations

During the Users Group meeting at the 2008 *HVE* Forum, a number of users expressed great interest in completing validation studies to demonstrate how well the results from the *SIMON* vehicle dynamics simulation program and the *DyMESH* 3-D collision model compare to a variety of real-world experiments.

If you would like to participate in these efforts, or would like to contribute existing data from well-instrumented collision or handling experiments, please contact EDC Customer Service to be directed to the appropriate organizer for your topic of interest.

2009 HVE Forum - February 23 - 27, 2009

Make plans now to attend the 2009 *HVE* Forum at the Orlando Marriott Downtown on February 23 - 27, 2009. The 2009 *HVE* Forum will offer important workshops that every user of *HVE* and *HVE-2D* should attend. Workshop schedules, descriptions and registration materials will soon be available to download from the 2009 *HVE* Forum page of the EDC website.

Technical Session

Based on comments made by attendees at the 2008 HVE Forum *DyMESH* workshop, it is clear that users want to understand how *DyMESH* works. This Technical Session provides a detailed overview of the *DyMESH* collision simulation model. A high-level description of the new features incorporated into *DyMESH* was included in the previous (Winter 2008) Newsletter.

DyMESH Procedures

DyMESH calculates vehicle-fixed collision forces and moments acting between a pair of vehicles, or a vehicle and the environment, during their collision. Multi-vehicle collisions are handled because *DyMESH* is included in a calculation loop designed such that all possible colliding vehicle pairs are evaluated. This design also allows simulation of simultaneous collisions involving more than two vehicles.

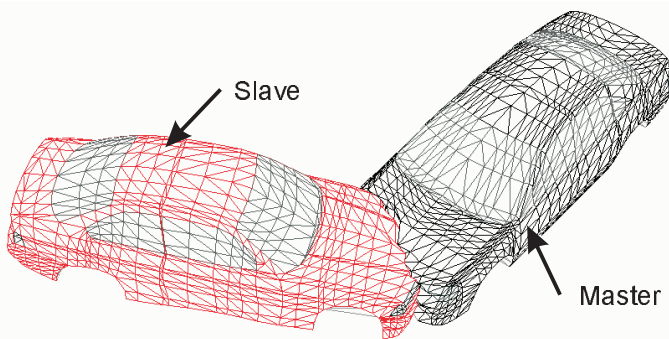


Figure 1 - Master and Slave vehicle meshes

When *DyMESH* executes, one of the vehicles is the *master* vehicle and the other vehicle is the *slave* vehicle (see Figure 1). Damage and resulting collision force are calculated for each vertex on the slave vehicle mesh. An equal and opposite force is then applied to the master vehicle. These forces are then summed in order to calculate the total collision force and moment components acting at the sprung mass CG.

DyMESH is part of the *SIMON* simulation model. Like all simulations, forces and moments are calculated at small, user-defined integration timesteps, usually 0.001 seconds. The current collision forces and moments acting on the sprung mass are then included along with suspension, aerodynamic and inter-vehicle connection (for vehicles towing trailers) forces and moments in the equations of motion.

The pair of master and slave vehicles is swapped each time *DyMESH* is executed. Therefore, the forces and moments are based on one vehicle for a given timestep, and the other vehicle for the next timestep. The forces and moments are averaged over two timesteps to prevent biasing the results due to a difference in vehicle stiffnesses.

What follows is a detailed, step-by-step description of the procedures executed by *DyMESH* during each calculation timestep.

Vehicle To Earth

All collision detection between vehicle meshes is done in the earth-fixed coordinate system. Thus, the earth-fixed vertex coordinates (both damaged and undamaged) for both vehicles must be calculated. The normal, center coordinates and area for each damaged polygon must be calculated as well.

In general, earth-fixed coordinates for vertices and polygon centers are calculated as follows:

$$\bar{S}_{Earth} = \bar{S}_{CG} + [A] \times \bar{S}_{Vehicle}$$

where

- [A] = vehicle to earth transformation matrix
- S_{CG} = earth-fixed CG location
- $S_{Vehicle}$ = vehicle-fixed coordinates
- S_{Earth} = earth-fixed coordinates

Vertex Earth Velocity

It is also necessary to calculate for both vehicles the earth-fixed velocity components at each damaged vertex.

$$\vec{V}_{Earth} = \vec{V}_{CG} + [A] \times \vec{V}_{Vehicle}$$

where

- [A] = vehicle to earth transformation matrix
- V_{CG} = earth-fixed CG velocity
- $V_{Vehicle}$ = vehicle-fixed vertex velocity
- V_{Earth} = earth-fixed vertex velocity

This result will later be used in calculating the inter-vehicle friction.

Calculate Normals

We have already calculated the normal for each polygon. We now calculate the normal for each damaged vertex. This is done using the average of the normals of all neighboring polygons (those that share the specified vertex; Figure 2).

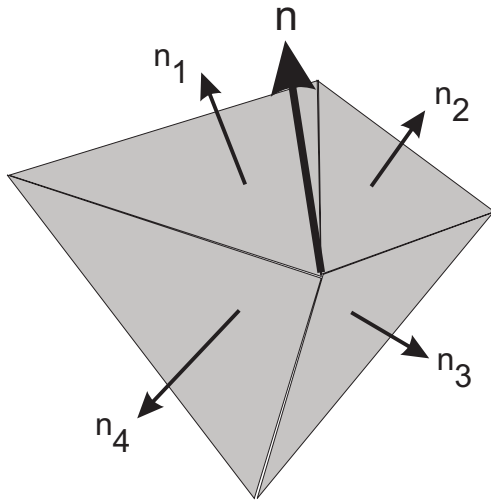


Figure 2 - Direction of vertex normal, n , is average direction of all neighboring polygon normals, n_i .

Calculate Bounding Box

The dimensions of a bounding box (Figure 3) into which the damaged vehicle fits are calculated. These dimensions are simply the minimum and maximum vehicle-fixed x , y and z vertex coordinates of the damaged vehicle mesh.

Find Relevant Geometry

This routine converts the bounding boxes for both vehicles into earth-fixed coordinates and then calculates the mathematical intersection of those bounding boxes. The resulting volume (Figure 4) represents the space inside of which vehicle mesh damage can occur. A list of *relevant polygons* for both

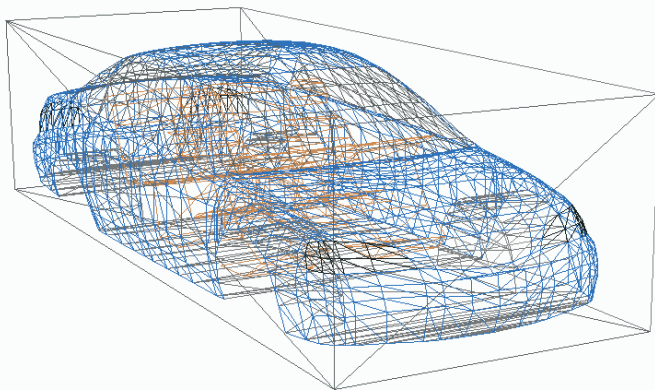


Figure 3 - The vehicle bounding box defines the current exterior dimensions of the vehicle mesh.

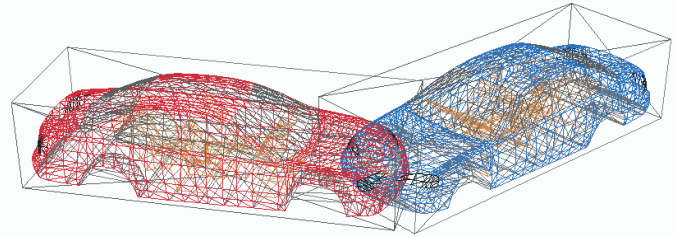


Figure 4 - Relevant geometry is located inside the intersection of the two vehicle bounding boxes

vehicles inside of this volume is created. From this list a *compact* list of *relevant vertices* is created (a *compact* list consists only of vertices inside the volume intersection; this results in a significant increase in calculation efficiency).

Build Working Vertex List

Next, we build a list of master vehicle vertices inside of the slave vehicle bounding box, and a list of slave vehicle vertices inside of the master vehicle bounding box.

Reset Vertex Data

This function simply resets all vertex damage parameters to zero or FALSE for master and slave vehicles.

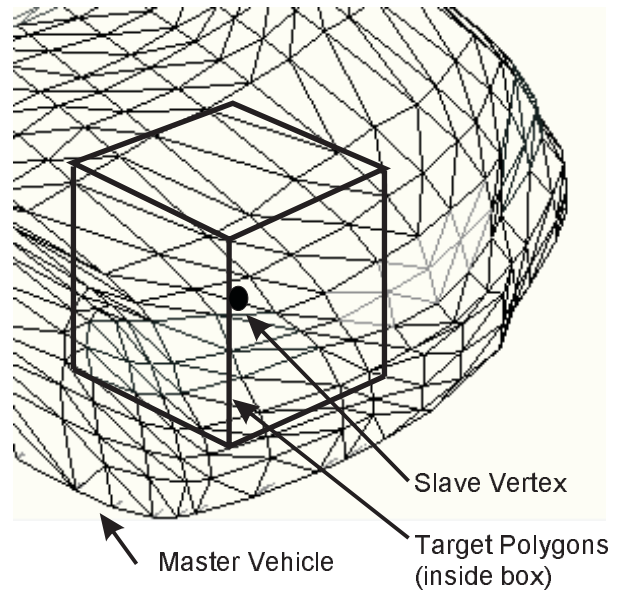


Figure 5 - Target polygons on the master vehicle are found inside a user-defined box size centered around the vertex on the slave vehicle.

Create Polygon List

Next, create a list of *target* polygons on the master vehicle (Figure 5). Target polygons are those having a polygon vertex within a distance defined by the user-entered value of *BoxSize*, centered around the slave vehicle vertex. Then, repeat the process for the slave vehicle.

Find Master Surface

This is the meat of the contact algorithm; it is where the vertex displacement is determined.

- Calculate an approximate damage direction for vertex displacement for each vehicle based on the area-weighted average of polygon normals for each of its relevant polygons, then average these directions for both vehicles so they are opposite in direction.
- Using the relevant vertex list, create a list of vertices on each vehicle that are inside the other vehicle's relevant polygons using the approximate damage direction as the basis. Then, for each relevant vertex in this list, refine the damage direction by averaging the normals of the penetrated relevant polygons, weighted by relevant vertex area ($1/3^{rd}$ of each vertex neighbor polygon).
- Calculate in the refined damage direction the slave polygon displacement if the master polygon's vertices are penetrating the slave polygon.
- Update current slave vertex earth-fixed coordinates.

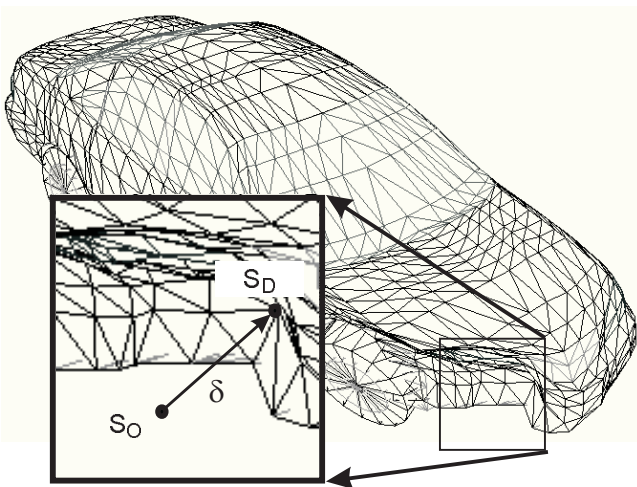


Figure 6 - Current vertex deformation, δ , is the difference between original vertex coordinate, S_O , and current (damaged) vertex coordinate, S_D

Update Deformation

Now calculate the deformation, δ , for each penetrated slave vertex in earth-fixed coordinates. Figure 6 shows this is simply

$$\delta = |S_O - S_D|$$

where

- S_O = Original Earth-fixed vertex coord
- S_D = Current earth-fixed vertex coord

Slave Node Force

For each deformed slave vertex, the current earth-fixed vertex force, F_V , in the direction of deformation is calculated. While deformation is increasing, the vertex force is

$$F_V = (A' + B'\delta) * Area$$

During rebound, the force is

$$F_V = (A' + B'(\delta - (10 - r)\delta_{max})) * Area$$

where

- A' = A/C
- B' = B/C
- C = Stiffness Conversion Fctor
- $Area$ = Area associated with vertex
- r = Relaxation Length

Slave Node Friction

The relative velocity between the deformed slave vertex and the point of penetration on the master polygon is calculate in earth-fixed coordinates,

$$\vec{V}_{Rel} = \vec{V}_{Poly, Master} - \vec{V}_{Vert, Slave}$$

The projection of this velocity on the master polygon's surface is calculated,

$$VdotN = \vec{V}_{Rel} \cdot \vec{N}_{Poly}$$

$$\vec{V}_{Tan} = \vec{V}_{Rel} - VdotN * \vec{N}_{Poly}$$

The magnitude of the slave vertex force normal to the master polygon surface is

$$FdotN = \vec{F}_{Vert} \cdot \vec{N}_{Poly}$$

The friction force in the plane of the master polygon surface in the direction opposing the sliding velocity is

$$\vec{F}_{Tan} = F_{dotN} * \mu * \vec{U}_{TanVel}$$

Finally, the total slave vertex force in earth-fixed coordinates is calculated as the vector sum of the normal and friction force components,

$$\vec{F} = \vec{F}_N + \vec{F}_{Tan}$$

Sum Forces and Moments

Next, the earth-fixed force is transformed back into vehicle-fixed coordinates so vertex forces can be summed and vehicle-fixed moments calculated. The first step in this process is to calculate vehicle-fixed coordinates for damaged vertices on the master and slave vehicles,

$$\vec{S}_{Vehicle,Slave} = (\vec{S}_{Earth,Slave} - \vec{S}_{CG,Slave}) [A]_{Slave}$$

$$\vec{S}_{Vehicle,Master} = (\vec{S}_{Earth,Slave} - \vec{S}_{CG,Master}) [A]_{Master}$$

Next, the earth-fixed vertex forces for the slave vehicle are transformed to vehicle-fixed and summed,

$$\vec{F}_{Vehicle,Slave} = \sum_{i=0}^n \vec{F}_{Earth,Slave} [A]_{Slave}$$

Based on slave vertex force, the associated vehicle-fixed force (Newton's 3rd Law) on the master vehicle is calculated,

$$F_{Vehicle,Master} = \sum_{i=0}^n -F_{Earth,Slave} [A]_{Master}$$

Finally, the vehicle-fixed moments are calculated for both vehicles,

$$\vec{M}_{Vehicle,Slave} = \vec{F}_{Vehicle,Slave} \times \vec{S}_{Vehicle,Slave}$$

$$\vec{M}_{Vehicle,Master} = \vec{F}_{Vehicle,Master} \times \vec{S}_{Vehicle,Master}$$

These vehicle-fixed forces and moments are then included in the generalized equations of motion, as described earlier, to calculate the current vehicle-fixed acceleration.

Rebound Damaged Vertices

The last step is to provide restitution into the model. This is done by calculating a *grow-back distance* for

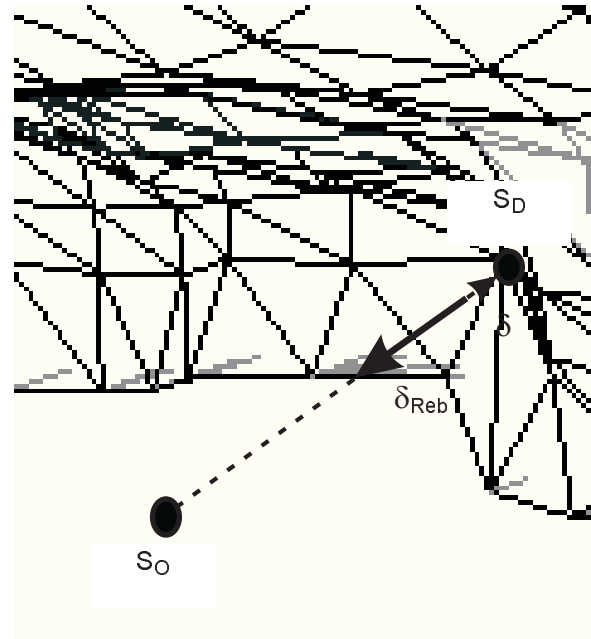


Figure 7 - Restitution is modeled by allowing a displaced vertex to grow back towards its original location

each damaged slave vertex. Each damaged vertex is moved back toward it's original (undamaged) location as shown in Figure 7,

$$\delta_{Reb} = \delta - (10 - r) * \delta$$

where

- r = user-entered *Relaxation Length*
- δ = current vertex deformation
- δ_{Reb} = rebound distance

Note that when $r = 0$, there is no grow-back. Thus there is no restitution force and the collision approaches a plastic impact. When $r = 1$, there is 100% growback. This produces a very large restitution force and the collision approaches an elastic impact. It should be understood, however, that the value of r does not directly assign the coefficient of restitution for the collision.

The above description provides a very detailed look at *DyMESH*. Our experience thus far has been extremely positive. *DyMESH* 2.0 is more robust than its predecessor and runs significantly faster (benchmarks show up to 10x faster!). User's are encouraged to rerun their previous *SIMON/DyMESH* events to see just how much faster their events execute. Call EDC Customer Service and let us know your experience.

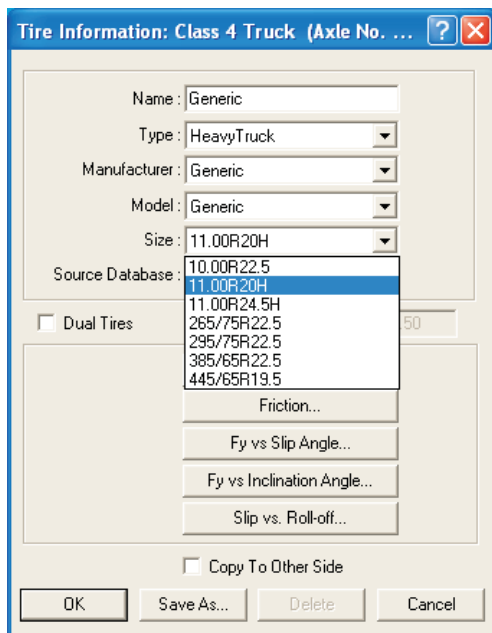
Creating Custom Size Tires for Vehicles in HVE

A question often asked of our Technical Support Engineers is about creating custom size tires for use on user-built vehicles in HVE. For example:

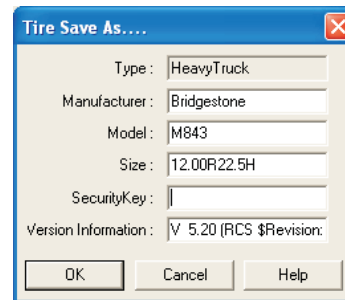
Q. I have a case involving a cement truck that has Bridgestone M843 4-rib/lug tires with a tread pattern for local on/off road use. The tire size is 12R22.5 and I have physical specifications for this tire. Can I create my own custom tire size in HVE?

A. Yes, you can. The basic procedure involves selecting the nearest size tire from the tire database, editing the Physical Data and Friction Data values appropriately to correspond to your actual tire, then saving this tire into the User Database with the proper name and size designation of the tire. When this tire is later selected to fit to a vehicle, HVE will read the designation for the tire size and display the tire accordingly. Since the tire is saved in your user.DB, it can be easily applied to all wheel positions on a vehicle and also applied to other vehicles in the future.

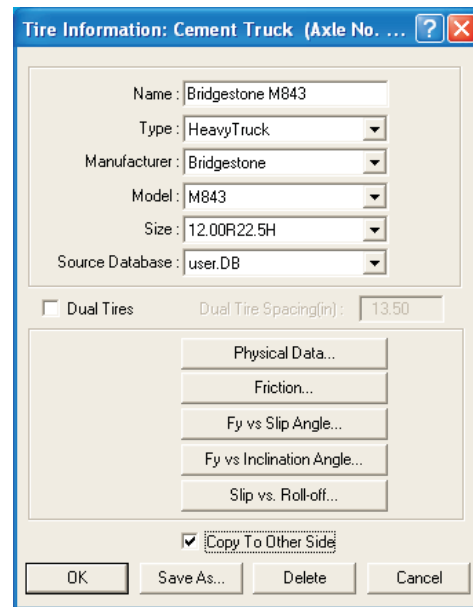
Tires can be created for any vehicle available in a HVE simulation or reconstruction. If you have questions about creating custom tires, please contact EDC Technical Support for assistance.



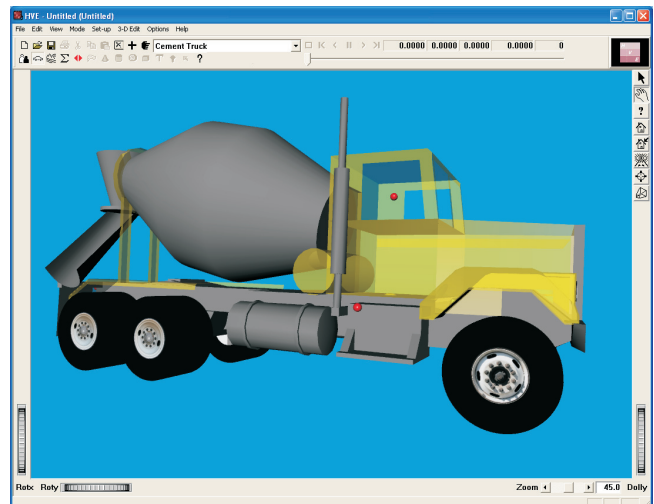
Tire Information dialog showing the Generic Heavy Truck tires available to fit to the vehicle. In the procedure, the user selects the one closest to the actual tire as a starting point for their edits of the Physical and Friction Data.



Tire Save As dialog after the user has edited the fields for Manufacturer, Model and Size. Pressing the OK button will save this modified tire into user.DB for later use.



Tire Information dialog showing the selection of the Bridgestone M843 tire the user created for the vehicle in this example. Note that the source indicates user.DB.



Vehicle with the proper tires fitted to the vehicle, along with the appropriate wheel images for the front and rear axles of the vehicle.

HVE and HVE-2D F.A.Q.

This section contains answers to frequently asked questions submitted to EDC Technical Support staff by HVE and HVE-2D users.

Q: I need to simulate an accident involving a vehicle with a steerable rear axle. How can I change my selected vehicle to have a steerable rear axle?

A: To change from the default front steering axle to a rear steering axle, follow these steps:

1. Display the vehicle in the Vehicle Editor and click on the green steering wheel/column icon displayed inside the car on the driver's side of the vehicle. In the dialog that appears, note the drop down list at the top indicating "Steering Data for" and the field next to it indicating "Axle 1".

2. To disable the front axle steering, uncheck the check box for "Steerable Axle" for Axle 1. Note that the data fields below are now greyed out.

3. Now change the drop down list to read "Axle 2" and check the check box for "Steerable Axle". Note that the data field below for "Steering Gear Ratio" is now enabled. YOU MUST ENTER an appropriate steering gear ratio for this axle, rather than leaving the default value of 0.00. If you do not enter a non-zero number, you will immediately receive an error message when you try to run an event using this vehicle. You also need to make sure the *Suspension Type* of this rear axle is shown as Solid Axle, rather than Independent.

4. Click OK to close the dialog and accept your change from the front steering axle to a rear steering axle.

5. Set-up and run your simulation. When entering Driver Controls for steering, note that your inputs in the tables are now indicated for Axle 2.

Q: I have installed HVE Version 6.00 and I notice that in an EDSMAC4 simulation involving multiple vehicles, the wheels on a vehicle at its rest position still appear to be rotating when other vehicles are still in motion. The vehicle at rest has 0.0 velocity reported in Key Results, but the wheel spin velocity is greater than zero? Why?

A. This issue was inadvertently caused by a change in EDSMAC4 and SIMON physics programs to improve the simulation of vehicles at very low velocities. The way to stop this unexpected spinning of the wheel images is to set the Termination Condition for Linear Velocity to a number less than 0.5 mph or even 0.0 and then reset and run your event to completion. With this small change, when the vehicle velocity drops to zero, so will the wheel spin velocity, thereby stopping the spinning of the wheel image.

NOTE: Our developers are currently working on the patch to EDSMAC4 and SIMON to correct this issue. This patch will be available for HVE v6.00 users to download from the Support, Downloads section of the EDC website at www.edccorp.com.

Q: I'm trying to accelerate a vehicle in reverse in EDSMAC (or EDSMAC4, EDSVS or EDVTS). I can get the vehicle to start with an initial velocity backwards, but when I provide throttle inputs the vehicle changes direction and starts accelerating forwards instead of accelerating in reverse. What do I do?

A. In the physics programs EDSVS, EDVTS, EDSMAC and EDSMAC4, the throttle inputs are for forward acceleration of the vehicle. That forward acceleration input overcomes the initial rearward velocity of the vehicle and results in the vehicle changing direction and accelerating in the forward direction. In the physics programs EDVSM, EDVDS and SIMON, the user can use the drivetrain model to put the transmission of the vehicle into reverse, so that any throttle inputs (at the pedal) result in the vehicle accelerating in reverse.

Q: How can I add vehicles or change the vehicle types in an existing event?

A. At this time, you need to create a new event with the correct number of vehicles or the correct vehicle types. It is not possible to go back and try to add more vehicles to the existing event, or to go back to the Vehicle Editor and try to change the types of vehicles already used in the existing event. Good news: Version 6.1 will allow users to add, delete and/or change vehicles in an existing event. Read about HVE V 6.1 on the front page of this Newsletter.

Q: I'm using the new HVE Version 6 and running a SIMON with DyMESH 2.0 simulation involving a vehicle rolling over on the terrain model. At the beginning of the event, I receive an error message indicating: "Event termination, DyMESH MaxNabor is too small" How do I fix this issue?

A. The error message indicates that either your vehicle geometry or your environment terrain geometry has a vertex that is used in creating more than 80 polygons. To correct this issue, you need to simplify the geometry either using a polygon reduction routine in your CAD program or by rebuilding your model with less polygons.

Visit the Support section of www.edccorp.com for the latest Downloads and answers to F.A.Q.'s.

EDC Training Courses

EDC Reconstruction & EDC Simulations

EDC offers an excellent training course on the use of the EDC reconstruction program, *EDCRASH*. Both new and long-time users of *EDCRASH* agree that the *EDC Reconstruction* course is extremely beneficial and challenging.

EDC also offers an excellent training course on the use of EDC simulation programs, such as *EDSMAC*, *EDSMAC4*, *EDSVS* and *EDVTS*. The *EDC Simulations* course offers the fastest way to learn what you really need to know – how to efficiently use the program and get the right results.

These one-week courses are designed to fully investigate the program's inner workings. Lectures are full of helpful hints gained from years of experience. During the course, students will use the programs (e.g. *EDCRASH*, *EDSMAC4*) in either the *HVE* or *HVE-2D* simulation environment to complete several workshops highlighting the capabilities of the programs.

HVE Forum

The *HVE* Forum is an excellent opportunity for *HVE* and *HVE-2D* users to jump to a new level of ability. By participating in workshops, attendees brush up on their present skills, learn new techniques, and learn how to use the latest advancements in the software. The *HVE* Forum also presents a great opportunity to meet other users and expand your network of resources.

Engineering Dynamics Corporation Training Course Schedule

EDC Simulations

Los Angeles, CA January 5 - 9, 2009

Miami, FL November 2009

EDC Reconstruction

Los Angeles, CA January 2010

Miami, FL November 10 - 14, 2008

Theoretical & Applied Vehicle Dynamics

Upon Request TBA

2009 HVE FORUM

Orlando, FL February 23 - 27, 2009

Vehicle Dynamics

Theoretical and Applied Vehicle Dynamics extends the theory of the basic SAE course and includes direct applications using several vehicle simulation programs (e.g. *SIMON*, *EDVSM*) within the *HVE* simulation environment, as well as a solid theoretical background for such simulations. The course is focused towards vehicle design engineers and safety researchers with an interest in a greater understanding of vehicle dynamics and automotive chassis systems development.

Hands-on Training

Intensive hands-on training on how to use your *HVE* or *HVE-2D* system software, physics programs and databases is available. Contact EDC Customer Service for more information about bringing this two-day on-site course to your office.

Course Registration

You may register for a course by contacting EDC Customer Service at 503.644.4500, or by email to training@edccorp.com. You can also visit the Training pages on our website and download a registration form. All courses are eligible for Continuing Education Units and ACTAR credits. See you at our next course!

DiscoverHVE.com

Collision Engineering Associates has launched a website where subscribers can learn more about using *HVE* by viewing basic, intermediate and advanced video tutorials. Members can also participate in on-line discussion groups covering various topics such as tire side-wall impacts, video output, importing vehicles and scenes, using humans in *HVE*, and known issues or work-arounds. Visit www.DiscoverHVE.com or contact Collision Engineering Associates at 480.655.0399 for more information.

Engineering Dynamics Corporation
8625 SW Cascade Blvd, Suite 200
Beaverton, Oregon 97008 USA
Phone 503.644.4500 / FAX 503.526.0905
Email: info@edccorp.com
Website: www.edccorp.com

EDVAP, EDCRASH, EDSMAC, EDSMAC4, EDCAD, EDSVS, EDVTS, EDHIS, EDVSM, EDVDS, EDGEN, EDVDB, HVE, HVE-2D, HVE Brake Designer and GetSurfaceInfo() are trademarks of Engineering Dynamics Corporation. All Rights Reserved.

SIMON and DyMESH (Patent number 6,195,625) are registered trademarks of Engineering Dynamics Corporation. All Rights Reserved.

GATB, GBF, DiscoverHVE.com and PhotoIntoHVE are trademarks of Collision Engineering Associates, Inc.