Case Study: Efficient visualization of physical and structural properties in crash-worthiness simulations

Sven Kuschfeldt and Thomas Ertl*
University of Erlangen
Computer Graphics Group

Michael Holzner†
Bayerische Motoren Werke AG
Car Body Design and Engineering

Abstract

Numerical finite element simulations of the behavior of a car body in frontal, side or rear impact collision scenarios have become increasingly complex as well as reliable and precise. They are well established as a standard evaluation tool in the automotive development process. Both, the increased complexity plus the advances in computer graphics technology have resulted in the need for new visualization techniques to facilitate the analysis of the immense amount of data originating from such scientific engineering computations. Expanding the effectiveness of traditional post-processing techniques is one key to achieve shorter design cycles and faster time to market. In this paper we describe how the extensive use of texture mapping and new visualization mappings like force tubing can considerably enhance the post-processing of structural and physical properties of car components in crash simulations. We show, that using these techniques both the calculation costs and the rendering costs will be reduced and the quality of the visualization will be improved.

1 INTRODUCTION

The analytical process in the development of prototype vehicles uses finite element analysis (FEA) with the primary goal to do as much development as possible with analytical tools and to minimize hardware-prototype testing. Taking into account the growing knowledge and experience of the simulation engineers and the increasingly reliable and stable calculation software the absolute simulation time for modeling, computing and investigating a complete crash model could be significantly decreased. However, a shift of the proportions between pre-processing, analysis and post-processing has to be noted with the post-processing stage turning out to become the most time consuming activity performed by the simulation engineers.

The changes in the proportions of relative simulation time and the development of computer graphics technology in the last few years has resulted in the need for new visualization techniques to facilitate the analysis of crash-worthiness simulations, since they offer a high potential to optimize the product development process and to improve the quality of post-processing. The study of realistically simulated crash scenarios involves very large data sets. FE models typically consist of 60 time steps with about 400,000 triangles each. The FEA data comprise not only spatial displacements of nodes and relationships between nodes and elements, but also physical properties of the surface structure such as stresses or plastic strains. Besides these physical properties, other information characterizing the deformation behavior of the structure, e.g. the intrusion of components into the passenger cell, are of special interest. Also the analysis of the flux of forces along longitudinal structures is very important.

In this case study we describe the productive application of visualization techniques in computer aided engineering, especially in crash simulations, utilizing recent research results in the field of texture mapping techniques. Additionally, we introduce new techniques for the visualization of the structural and physical behavior of finite element car crash models. First, we show the disadvantages of traditional methods currently used to visualize such properties. Thereafter, we describe the use of texture mapping for achieving a more efficient visualization, considering the recent developments of hardware and graphics subsystems, which allow the application of texture mapping without additional performance costs. Then, a new technique, called force tubing, is introduced which can be used to analyze the force flux through car components. Finally, we present our conclusions and suggestions for further investigations.

2 CRASH VISUALIZATION USING TRADITIONAL POST-PROCESSORS

Considering the progress of scientific visualization in various areas during the last decade, it becomes obvious that the application of 3D visualization techniques to finite element analysis has not been a primary focus [2, 5]. Nevertheless, the use of commercial visualization packages is now well established, i.e. in the automotive industry. In the case of crash analysis, these traditionally employed post-processors have been designed to manage the enormous amount of simulation data on workstations with limited memory by performing animations of wire-frame meshes and polygonal representations of the simulated crash models.

Physical properties like plastic strains on FE surfaces have been visualized through color coding of the polygons representing the surface elements. The strain values of the elements are read from the simulation output file. After being scaled, they serve as an entry into a HSV color table. The HSV values are converted into RGB values (requiring additional computation) and they are assigned to the corresponding vertices in the polygonal model. Unfortunately, this approach does not result in colors that appear equally spaced. One pair of adjacent colors in a HSV color table may seem nearly identical while another pair seems significantly different [9].

Therefore, surface areas with nearly equal strain values within predefined ranges are visualized using iso-contouring and color bands. Usually, the edges between two iso-contours intersect a certain amount of polygons representing the elements of the surface. The intersection points between contour edge and polygon must be calculated, and additional polygons with different colors have to be created on both sides of the contour line. Using this technique results in more costly computational and rendering procedures.

In many situations the discretization of the FE structure has to be displayed in combination with scalar or vectorial values. However, since the quadrilateral FE elements are subdivided into triangles during rendering, the grid is visualized deploying a hidden

---

[*] Universität Erlangen, Informatik 9, Am Weichselgarten 9, 91058 Erlangen, Germany, email: kuschfel@informatik.uni-erlangen.de

[†] BMW AG, Entwicklung Karosserie, 80788 München, Germany, email: Michael.Holzner@bmw.de

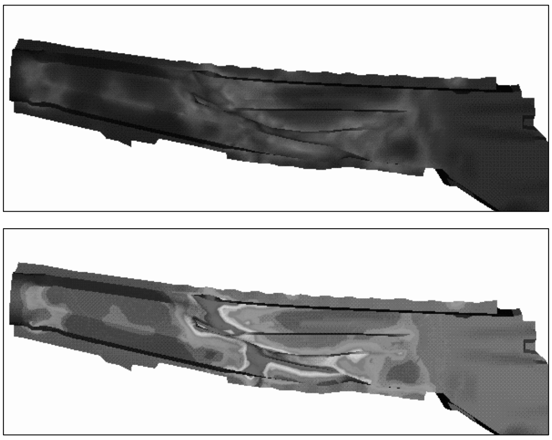

Figure 1: Mapping of strain values using plain color mapping (upper picture) and using iso-contours (lower picture).

wire-frame geometric representation or combining wire-frame and shaded polygons, respectively. In a first step, the polygonal model is rendered and in an additional step, the lines representing the element borders are drawn. This means, that the geometry must be rendered twice.

In order to calculate the forces that act inside a car component, section force calculations are performed. Using traditional post-processors, first a section must be defined and positioned within the component. Then, the section force is calculated, and finally a diagram is popped up showing the section force as a sum of the forces of the elements influenced by the section plane. The diagram contains information about the progression of the force at this position during the crash. For the investigation of the whole component many different sections have to be positioned within the component. Obviously, this way of investigation involving the successive evaluation of various 1D line drawings is very inconvenient and time consuming.

3 EFFECTIVE USE OF TEXTURES

3.1 Mapping of physical properties

Texture mapping as first introduced by Catmull [1] is now a well established and widely used technique in computer graphics (see the survey by Heckbert [4]). Traditionally, texture mapping has been used to add realism to computer generated images. The color of the object, onto which the texture is applied, is modified at each pixel by a corresponding color from the texture image. This is obtained performing a transformation of the texture from texture space into object space. Hardware support for texture mapping, i.e. fast interpolation and mipmapping in the texture memory is now widely available from high-end graphics workstations of various vendors down to PCs. A texture can be thought of not only as an image, but also as a lookup table [3]. We use a color table texture to visualize the values of maximum plastic strains at the surface of a component with less calculation and rendering costs in comparison to traditional visualization methods. Considering the perceptional drawbacks of the HSV color system a more reasonable approach is the use of a color table derived from CIE Lab color space [9]. The entries of the color table are mapped into floats between 0 and 1 and serve as coordinates of the one-dimensional texture. The strain values are scaled down and are used as entries into this texture map. The color values of the texture are RGB values, the texture coordinates represent values of the CIE Lab color system and scaled strain values likewise. The strain values are transformed directly into RGB values, and a conversion between the different color spaces is omitted.

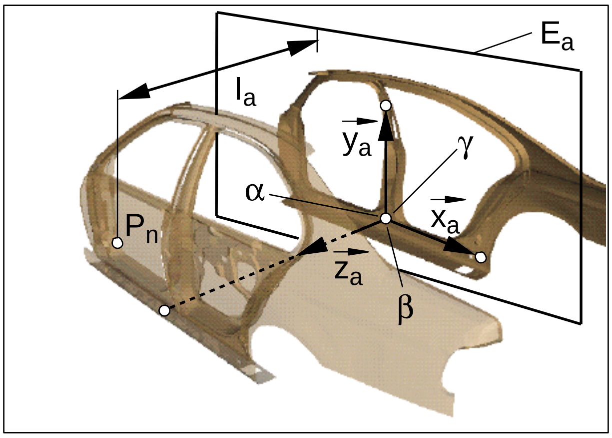

Figure 2: The reference coordinate system for the intrusion computation with initial RCS-angles, reference plane and initial distance of the node P_n.

Another approach for the use of one-dimensional textures is the creation of iso-contours. Using a texture with less colors and sharp borders between the colors automatically creates iso-contours on the textured model without any calculation of intersection points between the contour edges and the polygons of the model [10]. No additional polygons have to be rendered. The use of iso-contouring with textures for the visualization of strain values on the surface of a car component is shown in Figure 1.

3.2 Visualization of intrusion values

Whereas in the case of the perception of plastic strains on model surfaces the analysis using iso-contours will be qualitatively improved, the use of such contours is absolutely necessary in order to analyze the intrusion of components of the passenger cell in case of a frontal or side impact collision. Iso-contours show, at which parts of the deformed structure the intrusion of the passenger cell is acceptable and how far it is away from the acceptable limit. The behavior and the values of the intrusions influencing the components are determined using one-dimensional texture coordinates representing the difference between the initial position of the nodes of the FE structure and the position at any time step of the crash simulation in relation to a reference plane. The reference plane E_a must be defined within a reference coordinate system (RCS), which is created for the initial state, i.e. in the undeformed structure. First, three nodes are defined within a reference component, that will not be deformed and that will move minimally in relation to the whole passenger cell. The three points define the x-axis $\overrightarrow{x_a}$ and the y-axis $\overrightarrow{y_a}$ of the RCS. The fourth point, which defines the z-axis $\overrightarrow{z_a}$, is referenced by a node of the structure, that will be deformed by the intrusion and is to be investigated. The angles between the coordinate axes of the RCS, which we call initial RCS-angles α, β and γ, are calculated and stored. The origin and the normal of E_a are defined by the origin of the RCS and $\overrightarrow{z_a}$. Now, the initial distances between all nodes of the structure to be investigated and E_a have to be computed (see Figure 2). During the crash, the origin, the x-axis $\overrightarrow{x_v}$ and the y-axis $\overrightarrow{y_v}$ of the RCS are defined by the moving positions of the first three nodes. Because the deformation of the reference component is almost zero, both axes are practically tied together. For each time step, following operations are performed:

- The orientation of the z-axis $\overrightarrow{z_v}$ is calculated using α, β and γ.

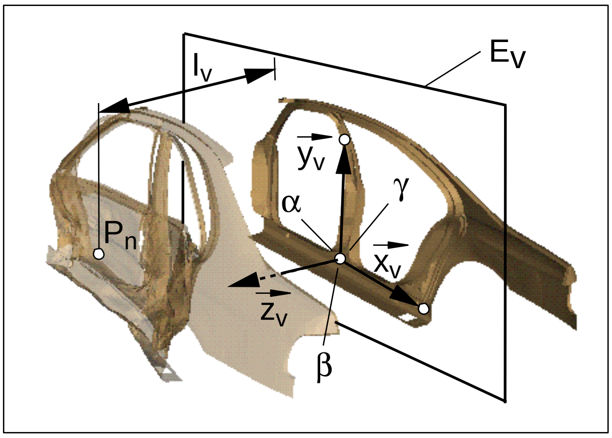

Figure 3: Computation of the intrusion value of the node P_n during the crash.

- For each node, the distance I_v to the new reference plane E_v, which is defined by the moved origin and $\overrightarrow{z_v}$, is calculated (see Figure 3).
- The difference between I_a and I_v is calculated and scaled down to values between 0 and 1, with respect to a predefined range of interest. The scaled values serve as texture coordinates of the vertices of the deformed structure.

If the acceptable limit is changed or the intrusion has to be investigated using a broader or tighter range of interest, no additional computational effort is necessary, only the texture definition has to be adapted. The use of iso-contouring showing the intrusion of the car body during a side impact collision is shown in Figures 8 and 9.

3.3 Wire-frame mapping

Besides the described property mapping, textures can be used to improve the understanding of the shape of complex structures [8, 7], in our case the grid of the finite element models. This is achieved through mapping of a texture, which paints the borders onto each element of the FE model. The main goal is to eliminate the increase of the rendering cost induced by the rendering of the shaded structure and the additional drawing of the wire-frame lines. We use a two-dimensional texture, which is represented by a white image with a black border. During the visualization of the FE models the different element types are converted into triangles. In the case of crash simulations, usually 90 per cent of the elements are four-sided, 10 per cent of the elements are three-sided. The coordinates of the corners of the texture image are assigned to the corresponding vertices of the polygons to be rendered. If the element is three sided, an additional vertex with the same spatial coordinates and the same normal like the third vertex is created. The fourth texture coordinate of the image is assigned to this new vertex. Since an efficient visualization requires the creation of triangle strips from the polygonal model, common vertices of adjacent polygons must have the same texture coordinates. Therefore, the texture coordinates are mirrored at adjacent elements (See Figures 4 and 5). We achieve a consistent line thickness of the element borders both at small elements and at large elements using mipmap filtering. The rendering speed is increased up to 200 per cent in comparison to the traditional method, because no additional drawing of lines is necessary. Figure 10 shows the FE mesh of a dummy model visualized with wire-frame mapping using a texture.

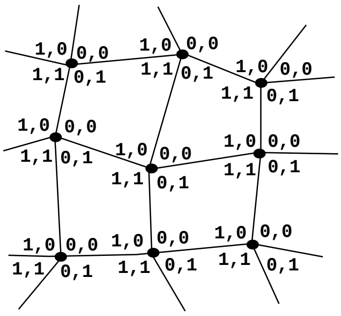

Figure 4: Texture coordinates of the vertices without mirroring of the texture.

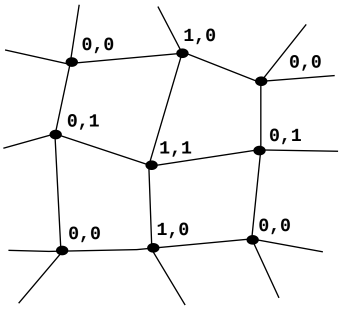

Figure 5: Texture coordinates of the vertices after mirroring of the texture.

4 FORCE FLUX VISUALIZATION WITH FORCE TUBES

During a car collision, each component of the car body is stressed in a different manner. Some parts absorb very high forces, other parts transfer the forces to the passenger cell. The determination of the structural components, through which the main forces are flowing, enables the engineer to design car components of the body structure more accurately in order to achieve an optimal behavior of the passenger cell. Since the longitudinal structures within the front part of a car body play an important role in order to increase the ability of the body to absorb forces in a frontal crash, it is necessary to detect and understand the force progression within these components (Figure 11 outlines, where the longitudinal structures are situated). We developed a new technique for the visualization of the force flux through longitudinal structures. The idea is to position an additional tubular geometry next to the deforming structure, whose radius variation visually relates to the local longitudinal force. We calculate the section forces at predefined sections for each time step of the simulation. These section forces are then displayed just like water flow in a flexible tube. Certain parts of the tube are expanding, when the longitudinal force through the corresponding part increases, whereas other parts of the tube are constricting, showing a decrease of force in the structure. Using this technique, the engineer is able to detect, through which parts of the car body the forces are transfered, and how these components must be modified in shape or material. Often, forces are not guided through structures, which should absorb them. This results in an increased risk of influencing the passenger cell. Using the tubing method, the behavior of longitudinal structures can be analyzed by investigating their deformation and simultaneously their ability to absorb forces.

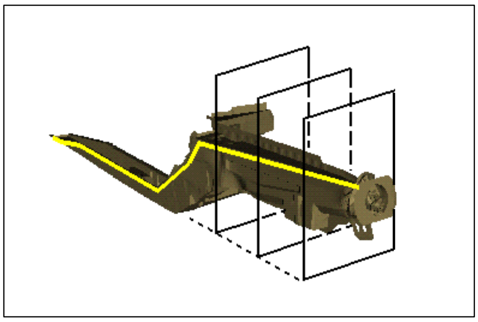

Figure 6: Generation of intersection planes along a trace line.

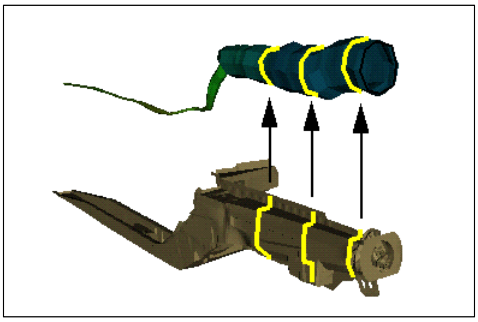

Figure 7: Force tube generation using section force values.

4.1 Computation of the force flux

If we want to analyze the force flux through a structure, we must compute a certain amount of section forces. The intersection planes are positioned perpendicular to the structure along a trace line, that follows the shape of the longitudinal (see Figure 6). There is one intersection plane next to each node of the line. At each time step of the crash, the planes' origins and normals are updated, and for each plane a list of intersected elements is created. During the FE simulation, the stresses influencing the elements were computed. We use these values for the calculation of the section forces, that affect the element nodes lying on the normal side of the plane. The forces are accumulated over the elements of the list. Finally, the vector component of the accumulated force vector that is parallel to the plane's normal, is computed. Usually, for each longitudinal structure, more than a hundred section forces must be calculated during each time step to obtain the necessary force values that represent the force flux through the component.

4.2 Building the force tube

The centers of gravity of the force tube are defined by a line parallel to the trace line of the structure. The distance of the tube to the trace-line can be chosen by the user. Thus, the tube can be positioned next to the longitudinal structure with a reasonable spacing. Several cylinders are positioned around this tube midpoint-line. The number of rings is equal to the number of sections, and every ring represents one section force. The position of a ring in the tube is corresponding to the position of the section on the trace line while the diameter is dynamically related to the value of the scaled section force (see Figure 7). Along the circle of each cylinder a number of points is created serving as vertices for polygons that connect the rings and form the tube. The vertices of the tube can be color coded according to the sign of the calculated section force value, displaying tension or compression forces. Figures 12 and 13 show the left longitudinal structure and the corresponding force tube during a frontal crash.

5 CONCLUSIONS

The applications mentioned above show, that the use of texture mapping is a powerful method to improve the quality and the ease of the visual analysis of crash simulations. Using efficient mapping techniques which require a minimum of memory, we ensure, that our applications can be run by a great number of engineers working on mid-range workstations with limited texture memory. Combining these applications with our new technique for the visualization of force fluxes through a car body structure we achieve a significant improvement over conventional post-processing tools. Together with polygon reduction algorithms the presented visualization methods are sufficiently interactive on even huge FE models, thus forming a basis for virtual reality applications [6]. Further fields of investigation are the extension of the force tubing mechanism for the visualization of bending and torsion moments affecting the car components as well as the insertion of recently introduced methods for the visualization of vector data [11] into productive applications in the field of computer aided engineering and design.

References

[1] E. Catmull. *A Subdivision Algorithm for Computer Display of Curved Surfaces.* PhD thesis, University of Utah, Salt Lake City, UT, 1974.

[2] R.S. Gallagher, R.B. Haber, G. Ferguson, D. Parker, W. Stillman, and J. Winget. Applying 3D Visualization Techniques to Finite Element Analysis. In *IEEE Visualization 1991*, pages 330–335, 1991.

[3] Paul Haeberli and Mark Segal. Texture Mapping as a Fundamental Drawing Primitive. In *Proc. of the Fourth Eurographics Workshop on Rendering, Paris, France, June, 1993*, pages 259–266, 1993.

[4] P.S. Heckbert. Survey of Texture Mapping. *IEEE Computer Graphics and Applications*, 6(11):56–67, November 1986.

[5] G. D. Kerlick and E. Kirby. Towards interactive Steering, Visualization and Animation of Unsteady Finite Element Simulations. In *IEEE Visualization 1993*, pages 374–377, 1993.

[6] S. Kuschfeldt, M. Schulz, T. Ertl, T. Reuding, and M. Holzner. The use of a virtual environment for fe analysis of vehicle crash worthiness. In *IEEE Virtual Reality Annual International Symposium 1997*, page 209, 1997.

[7] W. E. Lorensen. Geometric Clipping Using Boolean Textures. In *IEEE Visualization 1993*, pages 268–274, 1993.

[8] P. Rheingans. Opacity-modulating Triangular Textures for Irregular Surfaces. In *IEEE Visualization 96*, pages 219–225, 1996.

[9] L. R. Seidenberg, R. B. Jerard, and J. Magewick. Surface Curvature Analysis using Color. In *IEEE Visualization 1992*, pages 260–267, 1992.

[10] M. Teschner and Ch. Henn. Texture mapping in technical, scientific and engineering visualization. Technical report, SGI, August 1995.

[11] Boris Yamrom and Keneth M. Martin. Vector field animation with texture maps. *IEEE Computer Graphics and Applications*, 15(2):21–24, March 1995.

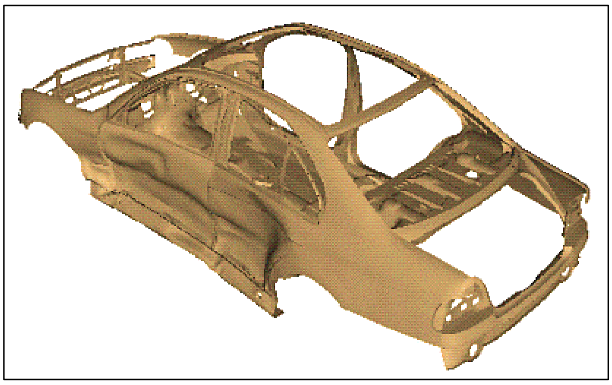

Figure 8: Intrusion of the car body during a side impact collision

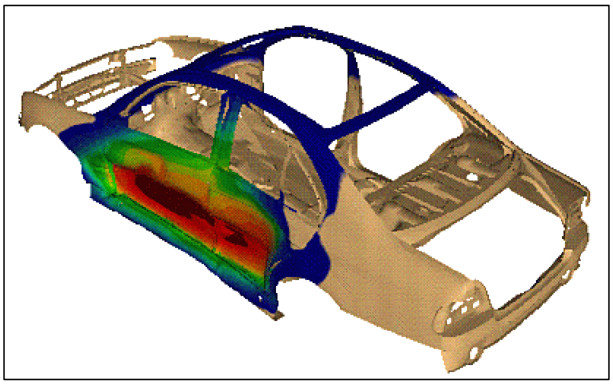

Figure 9: Intrusion visualization using a 1-D texture

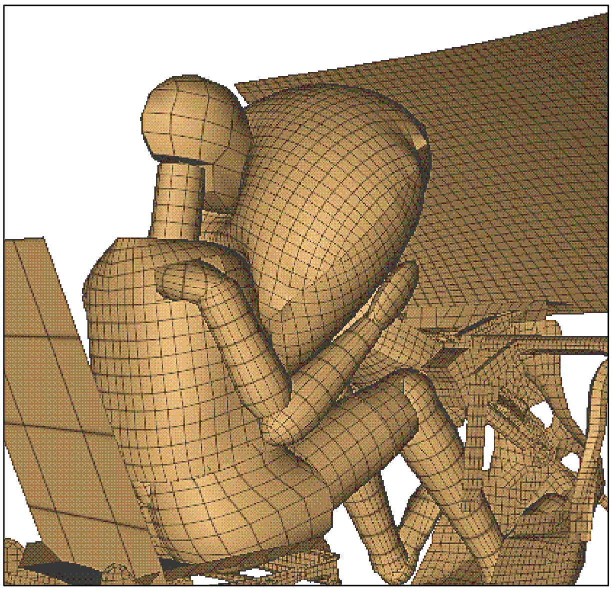

Figure 10: Wire-frame mapping using a 2-D texture

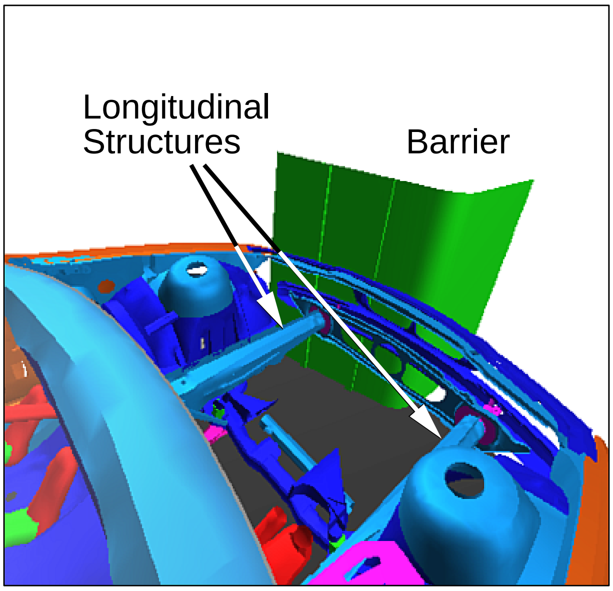

Figure 11: Longitudinal structures in the engine compartment

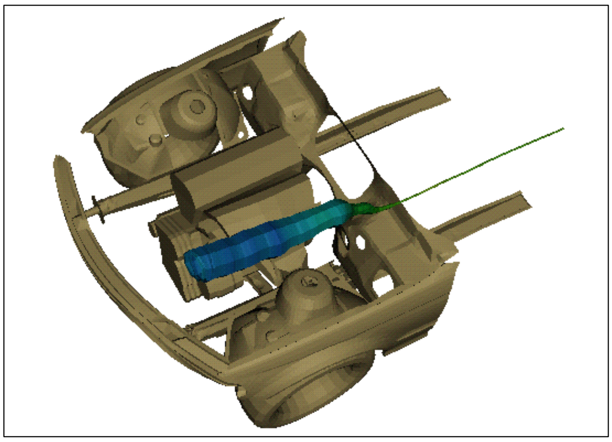

Figure 12: Force tube depicting the force flux through a longitudinal structure (20 milliseconds after the crash)

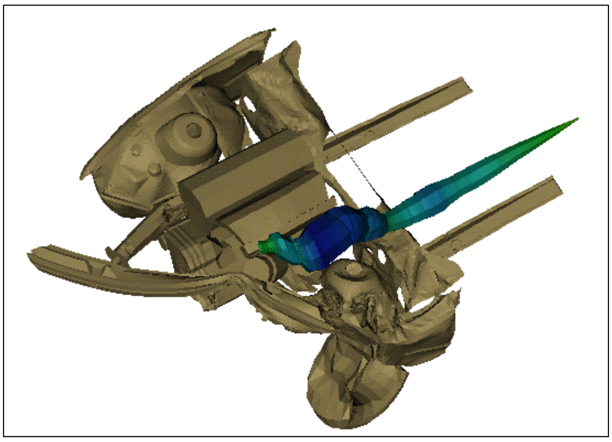

Figure 13: Same structure as in (e), but 104 milliseconds after the crash