

Interaktive Visualisierung neuer Elemente im virtuellen Automobil–Crashversuch

D. Rose N. Frisch T. Rühr T. Ertl

Universität Stuttgart · Institut für Informatik
Abteilung Visualisierung und interaktive Systeme*

Zusammenfassung

Im Rahmen des BMBF-Projekts *Autobench* wurden für den virtuellen Automobilentwurf und -crashversuch neue finite Elemente (FE) Vernetzungsstrategien, die eine unabhängige Bauteilvernetzung zulassen, erforscht. Parallel dazu wurden neuartige Elementtypen entwickelt, die einerseits zur Bauteilverbindung, andererseits zum Beispiel zur Abbildung Crash-unkritischer Fahrzeuggeometrie oder aber Sensor- und Messpunkten dienen. Im vorliegenden Artikel werden geeignete Wege für die Visualisierung solcher Komponenten und eine intuitive Benutzerschnittstelle vorgestellt.

1 Einleitung

Bei der Karosserieentwicklung von Kraftfahrzeugen spielt die passive Sicherheit eine bedeutende Rolle. Um die Gefahr von Verletzungen der Fahrzeuginsassen bei einem Unfall zu verringern, werden aufwändige Crashtests durchgeführt. Diese Tests sind sogar gesetzlich vorgeschrieben. Dabei werden mit viel Aufwand Fahrzeugprototypen hergestellt, noch bevor an eine Serienproduktion zu denken ist. Mit diesen Prototypen werden dann diverse Unfallhergänge rekonstruiert, um die Verletzungsgefahr der Insassen einschätzen und optimieren zu können.

Dank der modernen leistungsfähigen Rechenanlagen können solche Crashtests auch zuverlässig als Computersimulation durchgeführt werden. Dies spart viel Zeit und Geld bei der Entwicklung neuer Fahrzeugmodelle und bringt den Automobilherstellern dadurch Wettbewerbsvorteile. Außerdem kann durch Simulation eine viel größere Zahl von Tests durchgeführt werden als dies real möglich wäre. Die realen Crashtests können auf die gesetzlich vorgeschriebene Zahl reduziert werden und dienen gleichzeitig zur Überprüfung der Simulationsergebnisse.

Die Berechnung von Crashsimulationen geschieht nach der Methode der Finiten Elemente. Die finiten Elemente stellen eine Diskretisierung der Fahrzeugdaten dar. Durch die sogenannte Vernetzung werden die CAD Daten in FE Daten umgewandelt. Dabei werden zum

*Breitwiesenstr. 20–22, 70565 Stuttgart, Web: <http://wwwvis.informatik.uni-stuttgart.de/>
E-Mail: {rose|frisch|ruehrts|ertl}@vismail.informatik.uni-stuttgart.de

Beispiel durch Splines dargestellte gekrümmte Oberflächen in stückweise lineare finite Elemente aufgeteilt.

Bauteile, welche keine tragende Rolle spielen, brauchen nicht vernetzt zu werden. Ihre Masse wird bei der Berechnung auf die umliegenden Bauteile übertragen. Dieser Vorgang, als Massentrimm bezeichnet, geschah bislang manuell. Neuerdings werden spezielle Massenelemente verwendet, welche die Trägheitseigenschaften des nichttragenden Bauteils beinhalten. Die FE Simulationssoftware bezieht diese Massenelemente dann in die Berechnung ein.

Bis vor einigen Jahren wurde das Fahrzeug als Ganzes vernetzt. Der Nachteil dabei war, dass für jede Änderung eines Bauteils durch den CAD-Entwicklungsingenieur eine Neuvernetzung des Fahrzeugs erforderlich war. Deshalb ist man übergegangen zur unabhängigen Vernetzung von Bauteilen.

Waren einzelne Bauteile bislang durch gemeinsame Netzknoten miteinander verbunden, mussten für die unabhängig vernetzten Bauteile spezielle Verbindungselemente eingeführt werden. So wurden Verbindungselemente eingeführt, die Schweißpunkte, Schweißnähte und Klebeverbindungen repräsentieren. Das BMBF Projekt *Autobench*, an dem mehrere deutsche Automobilhersteller und Softwarezulieferer sowie Forschungseinrichtungen beteiligt waren, zählte zu seinen Themen auch die Problematik der unabhängig vernetzten Bauteile.

Im Rahmen dieses Projektes wurde an der Universität Stuttgart ein Visualisierungs-Tool für FE-Netze erstellt, das auch zahlreiche Möglichkeiten des Editierens bietet. So lassen sich auch Verbindungselemente zwischen Bauteilen visualisieren, erstellen und verändern. Das Visualisierungs-Tool trägt den Namen *crashViewer* und kann sowohl die Eingabedaten für die FE Simulation, als auch die Ergebnisdaten dreidimensional interaktiv darstellen [KHSE98]. Modifikationen sind jedoch nur bei den Eingabedaten sinnvoll, etwa um Bauteile oder Bauteilverbindungen zu editieren.

Der vorliegende Beitrag bezieht sich auf aktuelle, bislang unveröffentlichte Entwicklungen in der Visualisierung und Bearbeitung von speziellen Elementen für die Crash-Simulation. Nach Massentrimm und Schweißlinien werden die Sensorpunkte vorgestellt. Sensorpunkte dienen dazu, bestimmte Größen, insbesondere die Beschleunigung lokal zu messen. Durch den Vergleich zwischen Simulation und echtem Test können so Aussagen über die Richtigkeit der Simulation gemacht werden. Allerdings verhalten sich Fahrzeuge beim Crash nicht ganz deterministisch. Minimale Unterschiede der Ausgangskonfiguration können das Ergebnis signifikant verändern [SE01]. Zwar wird versucht, durch geeignete Bauweise das Crash-Verhalten möglichst deterministisch zu halten, dennoch muss für die Überprüfung der Korrektheit eine größere Zahl von Simulationen und realen Tests herangezogen werden.

Die neuen Elementtypen wurden im *crashViewer* zeitlich parallel mit deren Einführung in den FE Simulationscode implementiert. Da der Hersteller des Simulationscodes ebenfalls am *Autobench-Projekt* beteiligt war, konnte bei der Spezifikation der Datenaustauschformate mitgewirkt werden, um auch eine effiziente Verarbeitung und Visualisierung durch *crashViewer* zu gewährleisten.

2 Massentrimm

Bei der Crash-Simulation nach der FE-Methode wird das Fahrzeug, genauer: die einzelnen Bauteile, durch ein Gitter aus linearen Elementen dargestellt. Nichttragende Teile, zum Beispiel Batterie, Kabelbaum, diverse Aggregate und Betriebsflüssigkeiten brauchen nicht durch finite Elemente dargestellt werden. Durch das Vernachlässigen der Geometrie von nichttragenden Teilen werden also Rechnerressourcen gespart.

Bis jetzt wurden solche nichttragenden Teile durch den Berechnungsingenieur im Rahmen des Massentrimms auf benachbarte Bauteile „verschmiert“, d.h. die Masse der umliegenden Teile wurde anteilmäßig erhöht. So wurde das Fehlen der nichttragenden Teile im Modell näherungsweise kompensiert. Da nun das Simulationsprogramm die Möglichkeit bietet, nichttragende Teile als speziellen Elementtyp in der Eingabedatei zu spezifizieren, entfällt das zeitaufwändige und fehlerträchtige Verschmieren der Masse von Hand. Die neuen Elemente werden als *strukturlose Massen*, englisch *nonstructural masses* bezeichnet.

Für die Simulation relevant sind bei den strukturlosen Massen bloß die Masse und die Bauteile, auf die diese Masse verteilt wird. Will man die strukturlosen Massen jedoch visualisieren, so ist es notwendig zumindest deren Position, also den Schwerpunkt des nichttragenden Teils zu kennen. Ferner ist es wünschenswert, dass die Visualisierung einen Hinweis auf die Bauteilform gibt, die der strukturlosen Masse zugrundeliegt. Da aber weder Geometrie noch Schwerpunkt bei der Simulationsberechnung verwendet werden, ist es erstrebenswert, diese Information möglichst kompakt zu speichern. Eine Möglichkeit wäre, die Geometrie als Bounding Box vereinfacht darzustellen. Bei einer Komponente wie dem Kabelbaum wäre dies jedoch nachteilhaft, da hier die Bounding Box sehr groß ist und praktisch das gesamte Fahrzeug umschließt.

In Absprache mit den Anwendern, den Konstruktionsingenieuren, wurde deshalb vereinbart, Informationen über die Trägheitseigenschaften abzuspeichern. Von Interesse sind dabei die Hauptträgheitsmomente und die Hauptträgheitsachsen. Daraus kann der Konstruktionsingenieur Rückschlüsse ziehen über die Art des dargestellten Teils und dessen Einfluss auf das Crash-Verhalten. Hauptträgheitsmomente und -achsen können kompakt im sogenannten Trägheitstensor abgespeichert werden. Der Trägheitstensor ist eine symmetrische 3×3 Matrix, es sind also nur 6 Werte abzuspeichern [GHSW99]. Der Trägheitstensor ist für die meisten Bauteile bereits bekannt, kann aber auch ohne großen Aufwand berechnet werden. Die Hauptträgheitsmomente erhält man, indem man die Eigenwerte des Trägheitstensors berechnet. Die Eigenvektoren des Trägheitstensors sind die Hauptträgheitsachsen.

Zusammen mit dem Schwerpunkt (x_s, y_s, z_s) des nichttragenden Bauteils sind also $3 + 6 = 9$ Fließkommawerte zu speichern, welche bloß der Visualisierung dienen und nicht in die Simulation eingehen. Diese Werte werden als Zeichenkette kodiert im Titelfeld der strukturlose Masse abgespeichert, so dass sie die Simulation nicht beeinflussen. Bei der Visualisierung mit *crashViewer* können sie jedoch ausgewertet und editiert werden.

In der Theorie der Trägheitsmomente spielt das Trägheitsellipsoid eine Rolle, dessen Symmetrieachsen mit den Hauptträgheitsachsen übereinstimmen. Daher liegt es nahe, das Trägheitsellipsoid als Glyph zur Visualisierung der strukturlosen Massen heranzuziehen. Die Darstellung durch ein Ellipsoid bringt jedoch einige Nachteile mit sich. Insbesondere

sind die Hauptachsen nicht immer leicht zu erkennen. Außerdem benötigt eine ansprechende Darstellung des Ellipsoids am Bildschirm eine feine Tessellierung, was sich negativ auf die Bildwiederholrate auswirkt. Deshalb wird die Darstellung durch ein Ellipsoid nur als Option angeboten.

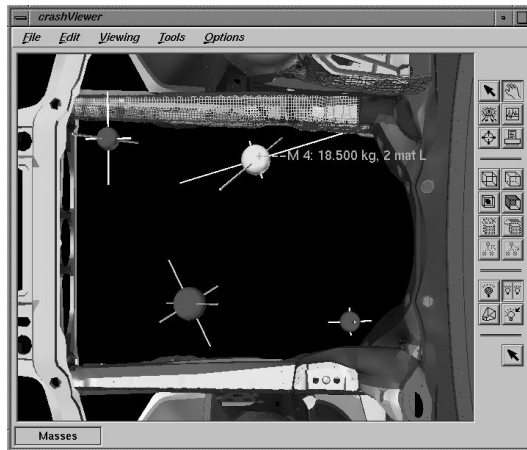


Abbildung 1: Visualisierung von vier strukturlosen Massen durch 3D-Glyphen. Für die jeweils selektierte Masse werden die betroffenen Bauteile als Drahtgitter visualisiert.

Zusätzlich zu den Trägheitsachsen wird auch die Masse des dargestellten Bauteils visualisiert. Dies geschieht durch eine Kugel, deren Mitte dem Bauteilschwerpunkt entspricht und deren Volumen zu der dargestellten Masse proportional ist. Diese Kugel benötigt keine feine Darstellung, da es bloß auf das Volumen ankommt. Durch das Kugelvolumen lässt sich sehr intuitiv die Masse visualisieren. Diese Darstellung wäre nicht möglich im Falle der Verwendung eines Trägheitsellipsoids, da sich die beiden Geometrien gegenseitig verdecken würden.

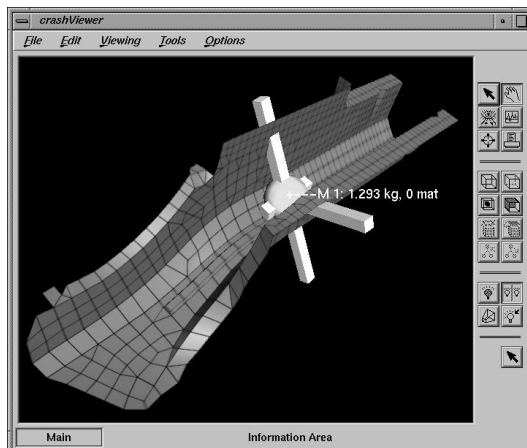


Abbildung 2: Visualisierung eines Bauteils zusammen mit dessen Trägheitseigenschaften.

Standardmäßig wird ein dreidimensionales Achsenkreuz für die Visualisierung der Trägheitseigenschaften verwendet. Das Kreuz wird dargestellt durch drei längliche, zueinander senkrechte Quader. Die Orientierung der Quader stimmt mit den Hauptträgheitsachsen überein. Die Quaderlänge ist proportional zu den Hauptträgheitsmomenten. Der Mittelpunkt der Quader befindet sich im Schwerpunkt des nichttragenden Bauteils (Abbildung 1).

Der Benutzer hat die Möglichkeit, neue strukturlose Massen anzulegen. Die Parameter der neuen oder bereits vorhandenen Massen, nämlich Schwerpunkt, Trägheitstensor und Masse können interaktiv editiert werden. Die Bauteile, auf die die Masse verschmiert werden soll, können spezifiziert werden und werden visuell hervorgehoben. Für Testzwecke kann auch im *crashViewer* die Masse auf die betreffenden Bauteile verschmiert werden. Auch können die Trägheitseigenschaften eines beliebigen Bauteils berechnet und visualisiert werden (Abbildung 2).

3 Verbindungen – Schweißnähte

Als Erweiterung zu den sich schon länger im praktischen Einsatz befindenden Schweißpunkten und den Klebeverbindungen können im *crashViewer* nun auch Schweißnähte hinzugefügt werden. Schweißpunkte [FRSE01] – oder bei mehreren aneinandergereihten Schweißpunkten: Schweißpunktlinien – definieren eine Verbindung zwischen benachbarten Bauteilen und sind nulldimensional. Klebeverbindungen [FRSE02] stellen eine flächige, zweidimensionale Verbindung zwischen Bauteilen her. Diese beiden Verbindungsarten können nur innerhalb von Bauteilflächen, in der Regel an speziell dafür konstruierten Flanschen eingesetzt werden.

Im Gegensatz dazu bilden Schweißnähte eindimensionale Linienzüge, welche auch Bauteilkanten miteinander verbinden können. Mittlerweile wird dieser Verbindungstyp von den neuesten Crashsimulationsprogrammen unterstützt und muß somit nicht mehr mittels einer Schweißpunktlinie nahe der Bauteilkanten nachgebildet werden. Andere Konstellationen, wie zum Beispiel Schweißen auf Stoß, konnten bisher überhaupt nicht simuliert werden.

Insgesamt treten bei den Nähten drei unterschiedliche Verbindungs-Konstellationen auf: Kante \leftrightarrow Kante, Kante \leftrightarrow Fläche und Fläche \leftrightarrow Fläche. Die beiden ersten Fälle orientieren sich immer an Bauteilgrenzen und gehören zu den sogenannten *Edge Links*. Die Erzeugung eines solchen *Edge Links* erfolgt mittels dreier Mausklicks: der erste legt das eine der beiden zu verschweißenden Bauteile fest, der zweite definiert zugleich das andere Bauteil und den Startpunkt der Schweißnaht, und der letzte Klick bestimmt das Ende der Schweißnaht. Generell ist die Definition einer Naht auch mittels lediglich zweier Mausklicks möglich. Die Erzeugung aller verwandten Verbindungsarten verläuft jedoch ebenfalls wie oben beschrieben. Um die Bedienung möglichst intuitiv und konsistent zu halten, fiel die Entscheidung daher zugunsten des 3-Klick-Verfahrens.

Im Gegensatz zu den bisherigen Verbindungstypen, welche – wie oben erwähnt – hauptsächlich an Flanschen zum Einsatz kommen, kann bei *Edge Links* nicht die automatische Flanschdetektion des *crashViewers* verwendet werden. Statt dessen werden für die beiden Bauteile die geschlossenen Linienzüge der Außenkanten/-ränder bestimmt. Hat ein Bauteil Bohrungen oder Löcher, so kann es unter Umständen mehrere Außenkanten besitzen. Der Algorithmus entscheidet sich jeweils für die Kanten, welche am nächsten zu den angeklickten Positionen liegen. Denkbar ist auch eine Abwägung der erzielbaren Nahtlängen, falls mehrere Ränder eine gültige Bauteilverbindung zuließen. Dies kann jedoch unter anderem bei zyklischen Schweißnähten zu unerwünschten Ergebnissen führen,

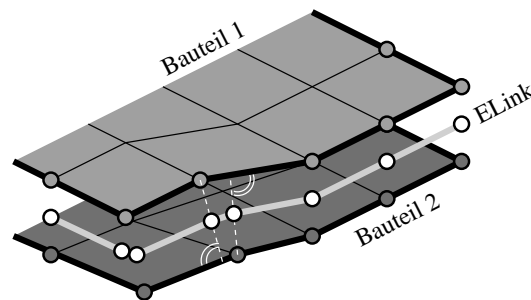


Abbildung 3: Die Stützpunkte einer Schweißnaht werden durch gegenseitige Projektion der Bauteilrandpunkte auf die benachbarte Bauteilkante und anschließende Mittelung gewonnen.

denn in der Regel will man hier die kürzere Nahtlinie. Zudem wird bei obigem Vorgehen implizit die Forderung des Ingenieurs erfüllt, der an genau *dieser* Stelle eine Schweißnaht haben möchte.

Die Randpunkte der zu verschweißenden Komponenten werden jeweils auf den entsprechenden Randkantenabschnitt des Gegenstücks projiziert, wie in Abbildung 3 zu sehen ist. Der Originalpunkt und sein projiziertes Abbild werden gemittelt und ergeben die Position eines Schweißnahtpunktes. Jeweils zwei topologisch benachbarte Punkte definieren dann zusammen ein neues *Edge Link* Element. Da sich die Netze der beiden Bauteile im allgemeinen nicht decken, kann dieses unter Umständen zu entarteten, also sehr kurzen Elementen führen. Es sind zwei Verfahren möglich, dies zu unterbinden. Entweder werden diese kurzen Elemente verworfen und die entstandene Lücke durch Verschmelzung/Mittelung der angrenzenden *Edge Link* Knoten geschlossen; oder aber die Schweißlinie wird abschließend neu abgetastet („Resampling“), so dass die Stützpunkte gleichmäßig verteilt sind. Dies kann gegebenenfalls zu einer Abrundung der Naht an Ecken führen, ein Effekt, der jedoch auch bei einer realen Schweißnaht auftritt.

Anschließend werden die erzeugten Elemente anhand verschiedener Kriterien wie Distanz, Winkelabweichung oder gegenseitige Durchdringung auf ihre Gültigkeit überprüft und gegebenenfalls sofort wieder verworfen. Im Gegensatz zu einem vorgeschalteten Abbruchkriterium erlaubt es diese *Validierung im Nachhinein*, Stellen, die nicht verschweißt werden können, zu überspringen. Dies kann im Bereich von Sicken häufig vorkommen. Hier wird dann automatisch eine dementsprechend gestrichelte Naht erzeugt. Extern definierte, das heißt aus Dateien eingelesene Verbindungen werden ebenfalls diesen Fehlertests unterzogen und können dann entsprechend modifiziert oder neu gesetzt werden.

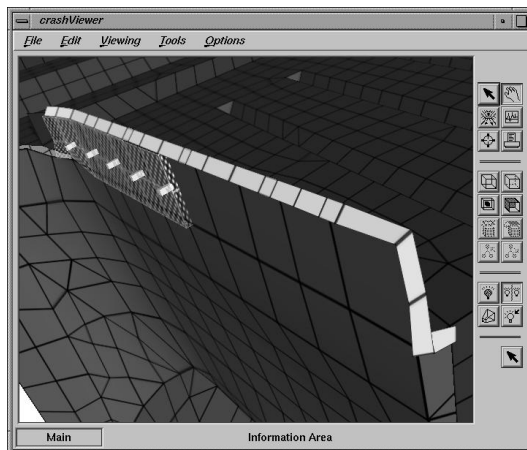


Abbildung 4: Schweißnaht (hellgrau dargestellt); im Hintergrund links die weiteren Verbindungsmöglichkeiten Schweißpunktnaht und Klebung.

nicht betrachtet. Bei dieser verbindet die Schweißnaht zwei Flächen und wird daher auch nicht als *Edge Link*, sondern als *Line Link* bezeichnet. Sie ist einer Schweißpunktnaht

Wie in Abbildung 3 zu sehen, liegen die *Edge Link* Elemente immer zwischen den verschweißten Bauteilen. Sie sind also – insbesondere von entfernt betrachtet – nur schwer zu entdecken. Aus diesem Grund werden sie nicht als Linienzug dargestellt, sondern mittels eines Prismenzuges mit rechtwinklig dreieckiger Grundfläche visualisiert. Die Hypotenuse verdeutlicht zudem den maximal erlaubten Abstand zwischen den zwei Schweißkomponenten, die rechtwinklige Spitze weist von den Bauteilen weg, so dass die Schweißnaht, wie in Abbildung 4, nach außen gut sichtbar ist.

Die letzte Verbindungs-Konstellation Fläche ↔ Fläche wurde bisher noch

[FRSE01, FRSE02] sehr ähnlich, die eingesetzten Algorithmen unterscheiden sich nur wenig, und daher soll an dieser Stelle nicht nochmals auf die Erzeugung eingegangen werden. Die verschiedenen Klassen unterscheiden sich zwar in ihrer numerischen Behandlung, in der Handhabung beim Erzeugen und Editieren sollten sie jedoch gleich behandelt werden. Daher ist die Interaktion auch nicht unterschiedlich gehandhabt, das Programm erkennt automatisch anhand von Distanzbetrachtungen zu den in Frage kommenden Randkanten, welchen Elementtyp der Ingenieur zu erzeugen beabsichtigt. Lediglich in der Visualisierung findet eine Differenzierung statt, da *Line Link* Elemente mittels eines Prismenzugs mit rechteckigem statt dreieckigem Querschnitt symbolisiert werden. Somit bleibt eine Unterscheidung weiterhin möglich, auch wenn dies in der Regel keine Notwendigkeit ist.

4 Sensorpunkte

Während eines Crashversuchs treten neben den sichtbaren Verformungen auch Kräfte und Beschleunigungen auf, die an definierten Stellen mittels Sensoren gemessen werden. Diese werden in der Regel an einem Bauteil festgeklebt oder -geschraubt. Das Zentrum der Mess-Sensorik liegt konstruktionsbedingt nicht auf der Bauteiloberfläche, sondern – wie in Abbildung 5a skizziert – um einen sensortypischen Abstand D verschoben.

Um die Rechenergebnisse mit der Realität vergleichen zu können, bietet nun auch das Simulationsprogramm die Möglichkeit, solche Sensoren nachzubilden. Zur Visualisierung eines solchen Sensors wurde ein Oktaeder gewählt, dessen eine Spitze auf dem Fußpunkt sitzt, an dem der Sensor auf dem Bauteil befestigt ist. An der gegenüberliegenden Spitze liegt das Zentrum der Messaufnehmer innerhalb des Sensorkörpers. Hier wird zusätzlich das orthonormale Achsensystem dieser Messaufnehmer angezeigt. Die Bezeichnung der Achsen ist farblich kodiert: der positiven x-, y- und z-Achse werden entsprechend die Farben rot, grün und blau zugeordnet, wie der Beschriftung in Abbildung 5a und 5b zu entnehmen ist. Auf eine aufwändige Darstellung des Sensorkoordinatensystems, zum Beispiel durch Pfeile, wurde bewusst verzichtet, da dies sowohl die Darstellungsperformanz, als auch die Übersichtlichkeit verschlechtern würde und zudem dadurch keine weitere Information vermittelt wird.

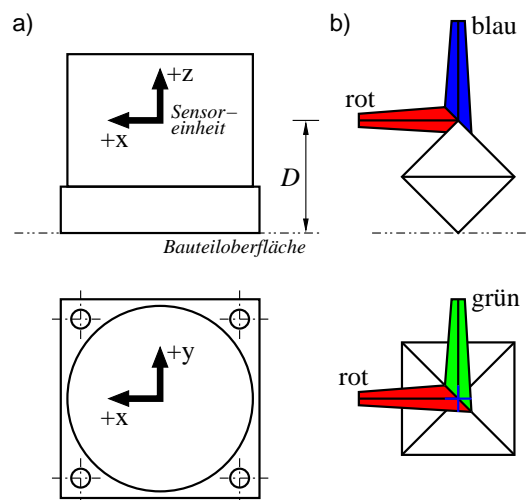


Abbildung 5: a) Realer Beschleunigungsaufnehmer und b) virtueller Sensorpunkt, jeweils in der Seitenansicht und in der Draufsicht.

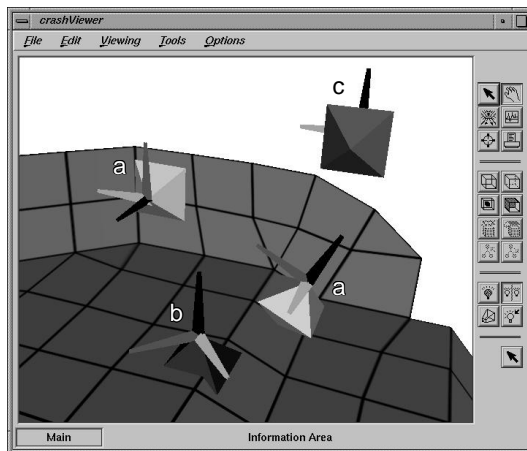


Abbildung 6: Sensorpunkte: a) gelber Grundkörper $\hat{=}$ gültig, b) rot $\hat{=}$ falsche Distanz, c) violett $\hat{=}$ ungültige Bauteilreferenz.

Ausrichtungen hergestellt, sondern stehen lediglich im gleichen eingeschränkten Winkelraster zur Verfügung. Daher ist die schnelle Einstellmöglichkeit mittels 45° -Drehungen der Angabe beliebiger Winkel mittels eines Eingabefelds vorzuziehen.

In Abbildung 6 sieht man des weiteren, dass eingelesene Sensorpunkte validiert und im Fehlerfall entsprechend eingefärbt werden. Diese Überprüfungsmöglichkeit hat sich bereits bei den Verbindungselementen bewährt und der Ingenieur hat anschließend die Möglichkeit, fehlerhafte Elemente automatisch – soweit sinnvoll – zu korrigieren.

Beim Einlesen ergibt sich hier zunächst das Problem, dass die Ausrichtung des Sensorkörpers nicht explizit gespeichert wird, sondern lediglich die Ausrichtung und die Ursprungsordinate der Sensorik-Einheit. Unter Zuhilfenahme der Information, auf welchem Bauteil der Sensor montiert ist und des ebenfalls angegebenen Abstands des Messaufnehmerzentrums zur Oberfläche dieses Bauteils, läßt sich der entsprechende Fußpunkt des Sensors auf dem Bauteil bestimmen. Durch eine hierarchische Suche [SE00, GLM96] nach dem nächstgelegenen finiten Element läßt sich diese Bestimmung beschleunigen. Die Verbindungslinie des Sensorfußpunkts zum Ursprung des Sensorik-Koordinatensystems gibt nun die Ausrichtung beziehungsweise die Normale des Sensorkörpers an. Dadurch lassen sich leicht Aussagen über die Einhaltung bestimmter Kriterien – wie zum Beispiel korrekter Abstand zum Bauteil oder planare, orthogonale Montagefläche – treffen. Ein Distanzfehler, wie in Abbildung 6b, läßt sich daraufhin automatisch durch eine entsprechende Verschiebung entlang der Sensorkörperrnormalen beheben.

Der Berechnungsingenieur hat die Möglichkeit, Sensorpunkte zu platzieren, zu löschen oder sie zu modifizieren. Dies schließt auch Änderungen der Eigenschaften – wie zum Beispiel die Masse oder Achsenausrichtung – ein. Die Ausrichtung der Mess-Sensorik wird in der Regel in globalen Koordinaten angegeben, die relative Orientierung einer Achse orthogonal zur Befestigungsoberfläche ist jedoch ebenfalls möglich. Das Koordinatensystem kann um jede Achse in 45° -Schritten rotiert werden. Eine prinzipiell beliebige Ausrichtung ist zwar mittels einer einfachen Quaternionen-Rotation [Sho85] realisierbar, in der Realität werden Sensoren im allgemeinen jedoch nicht in beliebigen

5 Zusammenfassung und Ausblick

Im vorliegenden Artikel wurden Verfahren zur Visualisierung, Erzeugung und Handhabung der seit kurzem verfügbaren Simulationselemente „Sensorpunkte“, „Schweißnähte“ und „strukturlose Massen“ vorgestellt. Im Prototyp *crashViewer* wurden diese bereits in die Praxis umgesetzt und werden bei Partnern des *AutoBench-Projekts* und zunehmend auch bei deren Zulieferern mit Erfolg eingesetzt. Zur Zeit wird an der Umsetzung des Prototyps zum kommerziellen Tool *FEmod* gearbeitet. Im Rahmen des Nachfolgeprojekts *AutoOpt* wird auch in Zukunft eine konsequente Fortführung der intuitiven, einfachen und konsistenten Bedienung und Handhabung zur Erzeugung und Manipulation neuartiger Elemente angestrebt. Vor allem zur erweiterten Editierung von Netzen sollen neue Verfahren mit einer intuitiven Vorgehensweise verknüpft werden.

Literatur

- [FRSE01] N. Frisch, D. Rose, O. Sommer und T. Ertl. *Pre-processing of Car Geometry Data for Crash Simulation and Visualization*. In: Vaclav Skala, Hrsg., Proceedings of WSCG'01, S. 25–32, Pilsen, 2001.
- [FRSE02] N. Frisch, D. Rose, O. Sommer und T. Ertl. *Visualization and Pre-processing of Independent Finite Element Meshes for Car Crash Simulations*. to appear in Journal: The Visual Computer, 2002.
- [GHSW99] D. Gross, W. Hauger, W. Schnell und P. Wriggers. *Technische Mechanik, Band 3: Kinetik*. Springer Verlag, 6. Auflage, 1999.
- [GLM96] S. Gottschalk, M. Lin und D. Manocha. *OBB-Tree: A hierarchical structure for rapid interference detection*. In: Holly Rushmeier, Hrsg., SIGGRAPH'96 Conference Proceedings, S. 171–180. Addison Wesley, August 1996.
- [KHSE98] S. Kuschfeldt, M. Holzner, O. Sommer und T. Ertl. *Efficient Visualization of Crash-Worthiness Simulations*. IEEE Computer Graphics and Applications, 18:60–55, 1998.
- [SE00] O. Sommer und T. Ertl. *Geometry and Rendering Optimization for the Interactive Visualization of Crash-Worthiness Simulations*. In: Proceedings of the Visual Data Exploration and Analysis Conference in IT&T/SPIE Electronic Imaging, S. 124–134, January 2000.
- [SE01] O. Sommer und T. Ertl. *Comparative Visualization of Instabilities in Crash-Worthiness Simulations*. In: Proceedings of EG/IEEE TCVG Symposium on Visualization VisSym'01, S. 319–328,364, 2001.
- [Sho85] K. Shoemake. *Animating Rotation with Quaternion Curves*. In: SIGGRAPH'85 Conference Proceedings, S. 245–254, July 1985.