

CONSISTENT POSTPROCESSING OF ACOUSTIC SIMULATIONS

Dipl.-Ing. Steffen Liefold, Dipl.-Ing. Fabian Duvigneau, Prof. Dr.-Ing. Ulrich Gabbert

1 Einleitung

Für akustische Berechnungen, Analysen und Darstellungen steht dem Ingenieur eine breite Auswahl an kommerzieller Software zur Verfügung. Diese bietet standardisierte Verfahren für ein allgemeines Postprocessing. Für spezielle akustische Problemstellungen sind sie allerdings nicht flexibel genug, bzw. bieten sie keinen Einblick in die Algorithmen der Datenaufbereitung. Hinsichtlich der Entwicklung von der klassischen akustischen Analyse [1] (Schalldruck, dBA) hin zu psychoakustischen Kenngrößen [2] (Lautheit, Schärfe, etc.), muss gerade dieser Datenverarbeitungsprozesses sorgsam behandelt werden.

2 Problemstellung

Psychoakustische Kenngrößen können sehr sensibel auf Schwankungen im untersuchten Signal reagieren [3]. Deshalb steht zum einen die Signalgenerierung aus Simulationsergebnissen [4-5] im Vordergrund und zum anderen ist die Interpolation der Daten beispielsweise für ein Datenmapping genauer zu betrachten [6]. Die zugrundeliegenden akustischen Berechnungen wurden mit Hilfe der Finiten-Elemente-Methode [7-8] durchgeführt. Die notwendigen Schwingungs- und Akustikanalysen finden hierbei aus Gründen der Rechenzeit ausschließlich im Frequenzbereich statt [9-10]. Strukturschwingungen versetzen ein umgebendes Luftfluid in Schwingung, so dass sich ein komplexer

Schalldruckverlauf im gesamten untersuchten Luftvolumen ergibt [12]. Diese komplexen Größen gilt es nun genauer zu untersuchen. Dadurch können Aussagen über lokale Änderungen getroffen werden, die wiederum wertvolle Hinweise und Anhaltspunkte für erfolgreiche oder verfehlte Simulationsziele liefern.

Eine ähnliche Herausforderung bilden räumliche Modelle unterschiedlicher Topologie bzw. veränderter Geometrie. Üblicherweise müssen für einen direkten Vergleich geometrisch optimierter Modelle die Ergebnisse in den Knotenpositionen aufeinander abgebildet werden, um koinzidente Resultate zu erhalten. Diese Konvertierung beinhaltet zum einen Potential für Fehler in der Abbildung, zum anderen kann die Transformation bei Modellen mit nichtlinearen Ansatzfunktionen entsprechend komplex werden. Die Ergebnisse, die sich innerhalb geometrischer Optimierungsprozesse ergeben, werden daher oftmals nur visuell anhand von Schnittbildern miteinander verglichen. Die Erstellung dieser Bilddaten ist zeitaufwändig und wird üblicherweise vom Berechnungsingenieur manuell durchgeführt. Die dazu erforderlichen Schritte sind: Auswahl des Simulationsmodells im ausgewählten Postprozessor, Darstellung der Ergebnisse in der definierten Schnittfläche mit festgelegter Farbskala (globale Skala muss zuvor aus allen zu betrachtenden Varianten bestimmt werden), Screenshots der einzelnen Varianten und abspeichern der Kameransicht für alle folgende Varianten (ist dies nicht verfügbar, muss für jeden Screenshot die Sicht auf die Szene per Hand eingestellt werden) und abschließend die gemeinsame Darstellung für

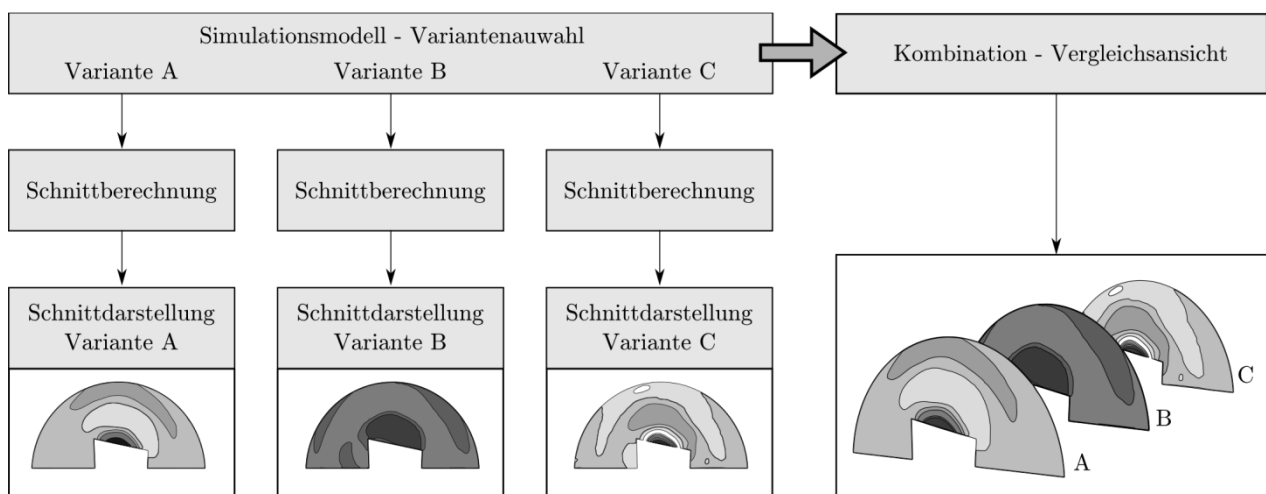


Abbildung 1: Konzept des Variantenvergleichs in einer zusammenhängenden Darstellung mit einheitlicher Skalierung. Die Schnittflächeneigenschaften, Farbskala, sowie die Anzahl der Farbwerte können für alle Varianten identisch definiert werden. Eine Auswahl an verschiedenen Ansichtsoptionen (Reihe / Spalte, Tiefe, Explosionsgrafik [11]) ist optional.

visuellen Vergleich (zusätzliche Software wird benötigt). Diese arbeitsintensive Vorgehensweise erspart zwar ein mathematisch aufwendiges Datenmapping, lässt allerdings Unterschiede in den Ergebnissen häufig nur erahnen. Die klassische manuelle Methodik lässt sich für den Vergleich von Simulationsmodellen mit variierender Geometrie (siehe Abb. 1) erweitern und für einen objektiven Vergleich automatisiert umsetzen. Das mit Hilfe der in Abb. 1 dargestellten Prozedur erzeugte Schnittbild liefert bei übereinstimmenden Schnittebeneigenschaften (Position, Normale) für alle betrachteten Daten die gleiche Ebene. Dies erfüllt die Grundvoraussetzung für einen Vergleich. Die globalen Minima und Maxima lassen sich ebenfalls direkt extrahieren. Für Elemente mit linearen Ansatzfunktionen müssen keinerlei Besonderheiten berücksichtigt werden. Die Simulationsdaten werden entsprechend der Schnittgeometrie an den Eckpunkten der einzelnen Elementschnittflächen linear interpoliert und repräsentieren damit den ermittelten Wert exakt, hinsichtlich der numerischen Lösung. Bei Modellen, die hingegen höherwertige Ansatzfunktionen in den Elementen nutzen, müssen für eine korrekte Abbildung der Ergebnisse mittels Interpolation die genutzten Ansatzfunktionen hinterlegt und in die Schnittwertberechnung einbezogen werden. Die hierfür benötigten Informationen lassen sich aus einer hinterlegten Datenbank dank einer element-spezifischen ID extrahieren, so dass ein genaues Verfahren für ein verbessertes Datenmapping eingesetzt werden kann.

3 Datenmapping

Ergebnisdaten aus Simulationen weisen nicht immer identische geometrische Eigenschaften auf [13]. Um Veränderungen z.B. in geometrischen Optimierungsprozessen oder alternativ vernetzten Modellen darzustellen, ist es daher erforderlich Simulationsergebnisse von einem Modell auf ein anderes abzubilden. Für zwei zu vergleichende Modelle A und B bedeutet dies eine Abbildung der Ergebnisse des Modells A auf die geometrischen Knoten des Modells B oder umgekehrt. Hierfür wird zunächst für jeden Zielknoten N_B im Modell B das Element E_A im Modell A bestimmt. Aus den globalen Koordinaten (Knoten des Elements E_A , Zielknoten N_B) werden anschließend die Koordinaten bezüglich des lokalen Bezugssystems im betreffenden Element E_A berechnet. Nun lassen sich die elementtypischen Ansatzfunktionen nutzen, um für die ermittelte Koordinate den neuen Wert aus deren Summe exakt zu bestimmen. Quadratische Tetraederelemente dienen im gesamten Beitrag zur Veranschaulichung dieser Herangehensweise.

Für kartesische Koordinaten lässt sich die Berechnungsvorschrift für den quadratischen Elementansatz durch Polynome aus dem sogenannten Pascalschen Dreieck (2d) bzw. dem Pascalschen Tetraeder (3d) ableiten [14]. In Abb. 2 ist das Pascalsche Tetraeder für den 3d-Fall dargestellt. Es beinhaltet alle beteiligten Terme, deren Polynomgrad mit jeder zunehmenden Stufe steigt. Der Polynomgrad 0 erzeugt die 1 an der Spitze des Konstrukts. Für ein 2d-Problem

werden die Terme einer Seitenfläche betrachtet – für ein 1d-Problem nur die Terme entlang einer Kante.

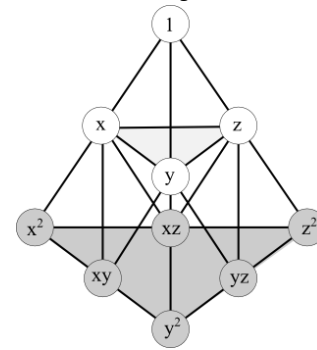


Abb. 2: Pascalsches Tetraeder für quadratischen Polynomansatz eines 3d-Problems.

Aus den globalen Koordinaten (xyz_i -Knoten des Elements E_A) und den zugehörigen Simulationsergebnissen u_i werden anschließend die Koeffizienten C für den Verschiebungsansatz berechnet.

$$u_i(x, y, z, 1) = C_1 + C_2x_i^2 + C_3x_i + C_4y_i^2 + C_5y_i + C_6z_i^2 + C_7z_i + C_8x_iy_i + C_9x_iz_i + C_{10}y_iz_i \quad (1)$$

Diese Koeffizienten C lassen sich mit Hilfe des aufgestellten Gleichungssystems für das hier gezeigte 10-Knoten Tetraeder bestimmen. Die xyz -Koordinaten des Zielknotens N_B führen anschließend zum gesuchten Wert u_B . Diese Vorgehensweise lässt sich auf andere Elementtypen übertragen und ermöglicht für jegliche Knotenpositionen die Berechnung der zugehörigen exakten Werte. Bei Elementen mit linearen Ansatzfunktionen genügt eine lineare Interpolation aufgrund der übereinstimmenden Verteilung im globalen Koordinatensystem.

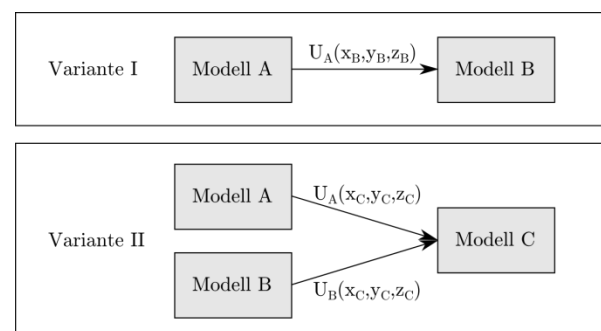


Abb. 3: Mögliche Varianten des Datenmappings für numerische Daten. Die Daten lassen sich entweder ineinander überführen oder sie werden auf ein unabhängiges Modell, bspw. eine Schnittfläche interpoliert.

Um mehrere Modelle miteinander zu vergleichen, bieten sich zwei Möglichkeiten (siehe Abb. 3) an. Zum einen kann ein Referenzmodell ausgewählt werden, das mit seiner Geometrie das Zielmodell darstellt. Alle Resultate der unterschiedlichen Modelle werden auf dieses Modell abgebildet. Wenn an einer spezifischen Modellgeometrie die Ergebnisunterschiede der jeweiligen Simulationsmodelle gezeigt werden sollen, ist dieser Weg sinnvoll. Wird dage-

gen eine andere räumliche Auflösung gefordert oder sind konkrete Werte auf einer definierten Ebene gewünscht, sollte die neue Geometrie diese Form bspw. als Schnittfläche berücksichtigen. Die Daten der Modelle A und B werden dann auf ein neues Modell C projiziert. Dies kann allerdings nur dann ausreichend übereinanderliegende Daten liefern, wenn die Schnittmenge (z.B. Schnittvolumen) der beiden Modelle groß genug ist. Denn für jeden zu interpolierenden Datenpunkt, der außerhalb der Elemente A und B liegt, bedeutet dies eine Extrapolation, die zu enormen Sprüngen in der Werteverteilung führen kann. Diese gilt es hinsichtlich einer möglichst korrekten Darstellung zu vermeiden. Aus diesem Grund werden keine Berechnungen an diesen Stellen durchgeführt.

Im Folgenden dient Variante I aus Abb. 3 einer detaillierteren Beschreibung dieser Herangehensweise. Als Beispiel dient eine Schnittfläche durch ein Volumen vernetzter, quadratischer Tetraeder. Ein Schnitt durch ein Tetraederelement liefert entweder einen Schnittpunkt, eine Schnittkante oder eine Schnittfläche. Die Fläche kann wiederum die Form eines Dreiecks oder Vierecks haben. Für eine Darstellung des Schnitts in 3d gilt es, die Flächen mit ihren Eckknoten zu berechnen - Punkte und Kanten teilen sich den Schnittbereich mit anderen angrenzenden Elementen. Nachdem die Schnittflächen berechnet wurden, ergibt sich meist ein äußerst inhomogenes Schnittbild, wie es beispielsweise in Abb. 4 zu sehen ist.

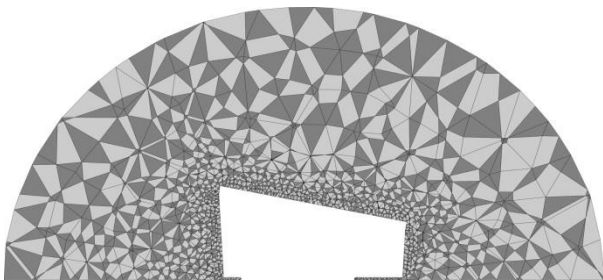


Abb. 4: Frontalansicht eines Schnitts durch ein mit Tetraederelementen vernetztes Volumen in 3d. Die Fläche besteht aus Dreiecksflächen (dunkelgrau) und Vierecksflächen (hellgrau). Unterschiedliche Kantenlängen sorgen für einen, inhomogenen Eindruck.

Die Werte an den Eckknoten lassen sich entweder linear oder quadratisch interpolieren. Für die lineare Interpolation können für eine angenäherte Lösung zwei lineare Funktionen jeweils zum Seitenmittenknoten bestimmt werden. Diese Ergebnisse sind allerdings nur hinsichtlich eines Geschwindigkeitsvorteils empfehlenswert, schließlich weisen die beschriebenen Tetraederelemente einen quadratischen Ansatz auf. Die daraus resultierende Abweichung im Verlauf der Lösungsvariable zu den Knotenwerten wird in Abb. 5 gezeigt. Die analytische Umsetzung bedeutet bezgl. einer geschnittenen Elementkante, einen reduzierten quadratischen Ansatz auf ein 1d-Problem, dessen Terme einer Kante des Pascalschen Tetraeders in Abb. 2. entsprechen.

$$u_i(x, 1) = C_1 x_i^2 + C_2 x_i + C_3 \quad (2)$$

Die Berechnung der Lösungsvariable im Element mittels quadratischer Ansatzfunktion liefert für das vorliegende Beispiel deutlich genauere Ergebnisse. Diese können jedoch, je nach Verteilung der Stützstellenwerte, weit über den an den Elementknoten zuvor bekannten Werten liegen. Bei einer quadratischen Interpolation bedeutet dies eine Abweichung von bis zu 12,5% (siehe Abb. 5) über den bisher bekannten Stützstellenwerten. Da diese maximalen Werte nur an zwei Positionen der Kante (viertel und dreiviertel der Elementkantenlänge) auftreten können, ist es für eine globale Betrachtungsweise empfehlenswert, diese Stellen zu überprüfen und zusätzlich für jedes Element zu bestimmen. Dies stellt nicht nur eine Neuberechnung der Obergrenzen im Modell dar, es sorgt zudem für eine zusätzliche Absicherung der Simulationsergebnisse.

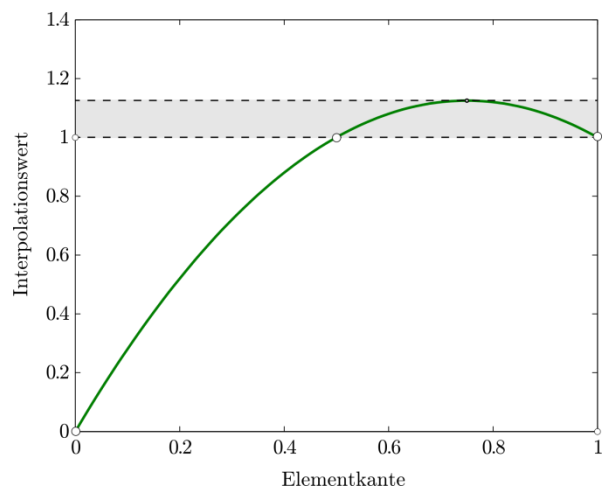


Abb. 5: Eine quadratische Interpolationsfunktion reduziert auf ein 1d-Problem als Verlauf entlang einer Kante mit den Stützstellen $[0,1]$. Die Funktion liefert für diese Stützstellenvariante einen neuen maximalen Werte (grau), der bis zu 12,5% über den vor der Interpolation bekannten Knotenwerten der Kante liegen.

Mit Hilfe dieser Herangehensweise wird allerdings immer noch kein homogener Gesamteindruck der Schnittfläche erzeugt. Hierfür sind maßgeblich die teils sehr unterschiedlichen Kantenlängen verantwortlich. Um die Genauigkeit der Schnittflächendarstellung zu erhöhen, wird daher eine äquidistante Rasterung (siehe Abb. 6) innerhalb der definierten Schnittflächenkoordinaten vorgenommen. Diese werden in ihrer Ausdehnung durch die Geometrie des Tetraedervolumens begrenzt.

Für jeden dieser neu definierten Rasterpunkte wird nun das Tetraederelement bestimmt, in dem er liegt. Dessen Koeffizienten C lassen sich für den vollständigen dreidimensionalen Verschiebungsansatz bestimmen und führen somit zu einer lokalen Berechnungsvorschrift für die innenliegenden Rasterpunkte. Die Koeffizienten werden hierbei aus dem linearen Gleichungssystem (siehe Gleichung 3) gewonnen. Dieses lässt sich aus den Formfunktionen des dreidimensionalen Verschiebungsansatzes aufstellen.

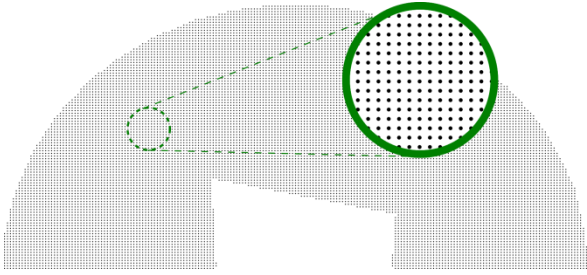


Abb.6: Äquidistantes Raster der Datenpunkte in einer definierten Schnittfläche des 3d-Tetraedervolumens. Außerhalb des Volumens liegende Rasterpunkte wurden mit einem boolschen Wert markiert und für die Darstellung vernachlässigt.

Die zu den Koeffizienten zugehörigen Polynome der quadratischen Ansatzfunktionen der jeweiligen Elementknoten werden hierfür in einer quadratischen Matrix A zusammengefasst.

$$AC_i = u_i. \quad (3)$$

Zusammen mit dem Vektor der Knotenwerte u_i lässt sich das System anschließend nach den Koeffizienten C_i auflösen. Dies lässt sich mittels Gaußschem Eliminationsverfahren

[15] effektiv durchführen. Für jedes Tetraederelement wird anschließend ein spezielles Polynom formuliert, welches für jeden innenliegenden Rasterpunkt die bestmögliche Lösung interpoliert. Im Vergleich zu einer linearen, aber auch quadratischen Interpolation über die Elementkanten, ist die Interpolation innerhalb der einzelnen Elementvolumen hinsichtlich der Ergebnisqualität deutlich im Vorteil (siehe Abb. 8). Die lineare Interpolation liefert die schnellsten Ergebnisse und genügt für eine grobe Darstellung. Erwartungsgemäß liefert die quadratische Interpolation an den Schnittpunkten der Elementkanten mit der Schnittebene exakte Werte, allerdings werden die dazwischenliegenden Werte wieder linear im Shader interpoliert. Folglich liefert nur die Interpolation im Elementvolumen eine genaue Lösung.

Um die für die Interpolation (siehe Gleichung 1) erforderliche Suche der zugehörigen Elemente aller Rasterpunkte zu beschleunigen, wird die Anzahl der zu untersuchenden Elemente im Vorfeld reduziert. Hierfür werden zunächst alle Elemente verworfen, die nicht von der betrachteten Schnittebene geschnitten werden. Dies lässt sich mit einer vereinfachten Fallunterscheidung bzgl. der auf einer Seite liegenden Elementknotenanzahl prüfen. Anschließend wird das durch die Rasterung entstandene reguläre Gitter genutzt, um in diesem neuen 2d-Koordinatensystem Zeilen j

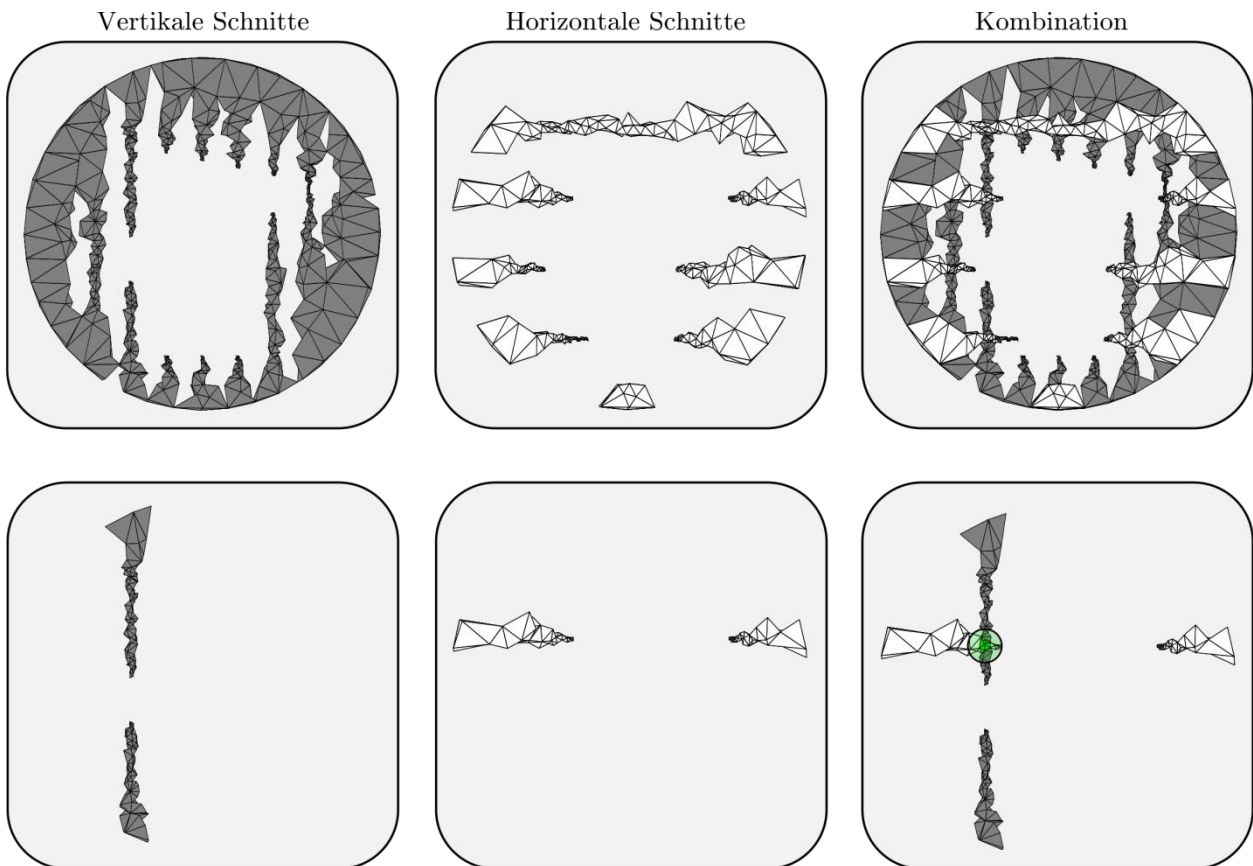


Abbildung 7: Beispiel für die Aufteilung der Elemente an einer Schnittebene entlang einzelner Rasterlinien. Diese können irregulär verteilt sein. Zu jedem Rasterindex (vertikal und horizontal) ergibt dies eine Elementreihe entlang der zugehörigen Rasterlinie. Die Abbildung rechts oben zeigt den gesamten Bereich der zu betrachtenden Elemente. In der unteren Reihe sieht man die Elemente entlang einer abgefragten Rasterlinie jeweils in vertikaler (unten links) und horizontaler Richtung (unten Mitte). Die Schnittmenge der Elemente liefert, wie in der Abbildung rechts unten zu sehen ist, eine deutlich reduzierte Elementanzahl für die Rasterpunktberechnung.

und Spalten i zu definieren. Deren Koordinaten definieren jeweils eine achsenparallele Linie in den z -Rasterkoordinaten. Für jede Linie wird anschließend ein eigener Test durchgeführt und die getroffenen Elemente bestimmt (siehe Abb. 7), die potentiell geschnitten werden. Dieser Test liefert mit Hilfe elementspezifischer Bounding-Boxen ein schnelles Ergebnis – erzeugt allerdings nur eine Abschätzung der weiterhin zu betrachtenden Elemente.

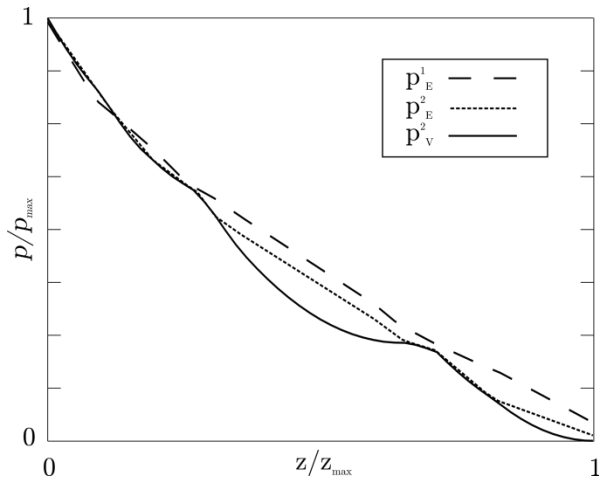


Abb. 8: Vergleich der Ergebnisqualität der Interpolationsvarianten entlang eines Pfades von der Schallquelle in das Fernfeld: $p1E$ – lineare Interpolation entlang der Elementkanten; $p2E$ – quadratische Interpolation entlang der Elementkanten; $p2V$ – quadratische Interpolation innerhalb der Elementvolumen in der Schnittfläche.

Alle zugehörigen Elemente entlang der jeweiligen Linienverläufe werden nun über eine Verknüpfung mit den Linienindizes gespeichert. Diese Indizes funktionieren wie klassische Zeiger und ermöglichen somit einen direkten Zugriff auf deren potentiell geschnittene Elementreihe. Über die Schnittmenge der nummerierten horizontalen E_H und vertikalen Linienelemente E_V lässt sich anschließend die zu untersuchende Menge an Elementen pro Rasterpunkt E_R sehr schnell auf einige Wenige reduzieren.

$$E_R = E_H \cup E_V \quad (4)$$

Dies erfordert in erster Konsequenz einen erhöhten Speicherbedarf, da zusätzlich für jede horizontale und vertikale Rasterkoordinate ein Vektor von Elementindizes gespeichert wird. Da allerdings nach dem Aufbau dieser Datenstruktur alle zuvor betrachteten Elemente nicht mehr benötigt werden, können sie nun aus dem Speicher entfernt werden und reduzieren somit nicht nur den endgültig benötigten Speicheraufwand, sondern auch den Rechenaufwand für die Bestimmung der elementeigenen Rasterpunkte. Über einen Knicktest (Skalarprodukt) der nach außen zeigenden Elementflächennormalen mit den Vektoren der Elementknoten zum Rasterpunkt, lässt sich dies effizient bestimmen. Alle außerhalb des Simulationsmodells liegenden Rasterpunkte werden nach deren Elementzuordnung

mit einem booleschen Wert markiert, so dass sie für die Interpolation berücksichtigt oder ausgeschlossen werden. Dieser Marker lässt sich zudem für eine nachfolgende Visualisierung nutzen, wobei hiermit klar definiert wird, welche Werte gerendert werden sollen und welche nicht.

Das gleichmäßig verteilte Ergebnis liefert nicht nur eine homogene Werteverteilung im Schnittbild (siehe Abb. 6), es ermöglicht zudem die Verwendung der äquidistanten Daten als Textur für eine effiziente, polygonreduzierte Darstellung der Ebene im Postprozessor. Hierfür wird die rechteckige Ergebnismatrix als Textur auf eine 3d-Ebene mit den Eckkoordinaten der Schnittfläche aufgebracht. Dies führt dazu, dass alle zu rendernden Flächen auf zwei resultierende Dreiecke reduziert werden. Diese Methodik lässt sich nun beliebig erweitern, so dass beispielsweise auch 3d-Texturen (siehe Abb. 9) des gesamten Elementraums berechnet werden können.

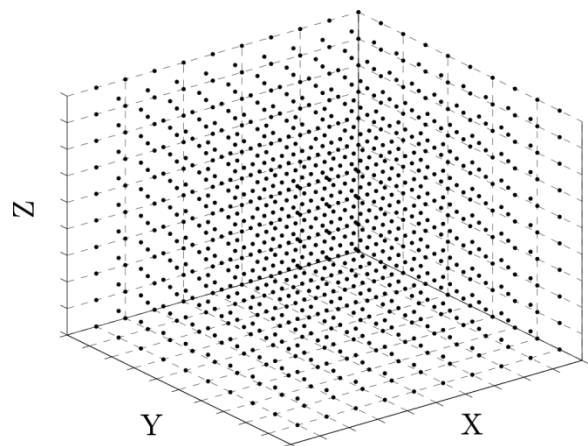


Abb. 9: Äquidistantes Raster der Datenpunkte für eine 3d-Textur. Die entstandenen Daten lassen sich für spezielle 3d-Volumendarstellungen weiterverarbeiten und dienen damit einem verbesserten räumlichen Eindruck der Datenverteilung.

Diese lassen sich anschließend z.B. für spezielle Volumendatenvisualisierungen [16], interaktive Clippingverfahren [17] und Isoflächenberechnungen [18] für Schwellwertdarstellungen verwenden. Hierbei ist zu beachten, dass die 3d-Texturdaten auch im Shader wieder interpoliert werden. Hinsichtlich der übergebenen 3d-Datentextur sind z.B. bei orthogonalen Schnittebenen nur die Werte an genau diesen Texturkoordinaten wieder exakt. Daher ist festzuhalten, dass die Rastereinteilung die finale Qualität der Darstellung maßgeblich beeinflusst – diesen Parameter kann der Berechnungsingenieur je nach Bedarf anpassen. Das beschriebene Datenmapping unter Verwendung des elementspezifischen Verschiebungsansatzes liefert dabei eine maximale Genauigkeit der Ergebnisse auf Basis des zugrundeliegenden numerischen Modells. Dies ermöglicht zum einen genauere Vergleiche unterschiedlicher Simulationsmodelle miteinander, aber zum anderen lassen sich auch Messergebnisse aus realen Versuchen auf diese Art mit einem numerischen Modell wesentlich besser vergleichen, da für jeden beliebigen Raumpunkt (innerhalb des berechneten Modellraums) der genaue Wert aus dem Simulationsmodell bestimmt werden kann.

4 Zusammenfassung

Die vorgestellte Herangehensweise zeigt ein durchgängiges Verfahren auf, mit welchem ein finales Postprocessing unter Berücksichtigung des zugrundeliegenden Simulationsmodells gelingt. Die erreichte Genauigkeit kann dem numerischen Modell bis auf minimale Abweichungen, bedingt durch die grafische Interpolation, optimal angepasst werden. Für die Auswertung psychoakustischer Größen eignet sich diese Methodik besonders [2], da schon geringe Interpolationsfehler die auf Basis der Ergebnisdaten getroffene Prognose hinsichtlich der akustischen Wahrnehmung verändern können. Zusätzlich lassen sich experimentell ermittelte psychoakustische Größen sehr gut mit Simulationsergebnissen vergleichen. Auch hier lässt sich der klassische Fehler durch Abweichungen zwischen realer Mikrofonposition und virtuellen Modellknotenkoordinaten auf 0% reduzieren.

5 Danksagung

Die vorgestellten Ergebnisse entstanden im Rahmen des Verbundprojektes „COMpetence in MOBility – Automotive“, das sowohl mit Geldern des Europäischen Strukturfonds als auch mit Geldern des Landes Sachsen-Anhalt gefördert wird. Die Autoren bedanken sich für die gewährte finanzielle Förderung.

6 Literatur

- [1] F. Duvigneau; S. Nitzschke; J. Strackeljan; U. Gabbert: Ganzheitlicher Virtual Engineering Ansatz zur Schwingungs- und Akustiksimulation eines Verbrennungsmotors, 11th International Conference on Vibrations in Rotating Machines - SIRM, Magdeburg, Germany, 2015.
- [2] S. Liefold; F. Duvigneau; M. Höchstetter: Sound quality of engine encapsulations, ATZ worldwide, Volume 117, Issue 6, 2015.
- [3] F. Duvigneau; S. Liefold; M. Höchstetter; J.L. Verhey; U. Gabbert: Engine Sound Weighting using a Psychoacoustic Criterion based on Auralized Numerical Simulations, 10th European Congress and Exposition on Noise Control Engineering - EuroNoise, Maastricht, Netherlands, 2015.
- [4] S. Liefold; F. Duvigneau; U. Gabbert: Advanced methods for auralization of virtual models in engineering processes, 11th Conference: Digital Engineering for the Planning, Testing and Operation of Technical Systems, Magdeburg, Germany, 2014.
- [5] F. Duvigneau; T. Luft; H. Rottengruber; U. Gabbert: Simulation of a thermo-acoustically encapsulated engine, 21st International Congress on Sound and Vibration - ICSV 21, Beijing, China, 2014.

- [6] S. Liefold; F. Duvigneau; U. Gabbert: Interaktive 3-D Visualisierung zur Unterstützung des Produktentwicklungsprozesses in der virtuellen Realität, 10th Conference: Digital Engineering for the Planning, Testing and Operation of Technical Systems, Magdeburg, Germany, pp. 243-250, 2013.
- [7] O.C. Zienkiewicz; R.L. Taylor: The Finite Element Method: Volume 1 The Basis, Butterworth Heinemann, London, 2000.
- [8] K.-J. Bathe: Finite-Elemente-Methoden, 2. Auflage. Springer-Verlag Berlin, ISBN 3-540-66806-3, 2002.
- [9] F. Duvigneau; S. Nitzschke; J. Strackeljan; U. Gabbert: Virtual engineering approach for the analysis of the acoustic behavior of an engine, 22nd International Congress on Sound and Vibration - ICSV 22, Florence, Italy, 2015.
- [10] U. Gabbert; F. Duvigneau; J. Shan: Active and passive measures to reduce the noise pollution of combustion engines, 11th IEEE International Conference on Information and Automation - ICIA 2014, Hailar, China, 2014.
- [11] O. Karpenko; W. Li, N. Mitra; M. Agrawala: Exploded View Diagrams of Mathematical Surfaces, IEEE Trans. Vis. Comput. Graph., Bd. 16, Nr. 6, pp. 1311–1318, 2010.
- [12] F. Duvigneau; U. Gabbert: Numerical study of a thermo-acoustically encapsulation, 11th World Congress on Computational Mechanics - WCCM XI, Barcelona, Spain, pp. 6825-6832, 2014.
- [13] P. Schrader; F. Duvigneau; T. Luft; H. Rottengruber; U. Gabbert: Development, simulation and experimental investigation of a function-integrated and foam damped oil pan for a two cylinder diesel engine, 44th International Congress and Exposition on Noise Control Engineering - InterNoise, San Francisco, USA, 2015.
- [14] B. Klein: FEM: Grundlagen und Anwendungen der Finite-Element-Methoden im Maschinen- und Fahrzeugbau, 9., Vieweg+Teubner Verlag, 2012.
- [15] G. H. Golub; C. F. V. Loan: Matrix Computations, Third Edition, Baltimore, Johns Hopkins University Press, 1996.
- [16] H. Pfister: Modern Volume Visualization, It - Inf. Technol., Bd. 46, Nr. 3–2004, 2004.
- [17] D. Weiskopf; K. Engel; T. Ertl: Interactive clipping techniques for texture-based volume visualization and volume shading, IEEE Trans. Vis. Comput. Graph., Bd. 9, Nr. 3, 2003.
- [18] J. Kruger; R. Westermann: Acceleration Techniques for GPU-based Volume Rendering, in Proceedings of the 14th IEEE Visualization 2003 (VIS'03), Washington, DC, USA, 2003.