

Finite Elemente Methoden auf konvexen Polyedern für physikalisch-basierte Schnittsimulationen

Mario Botsch^{1,2}, *Martin Wicke*^{1,3} und *Markus Gross*¹

¹ ETH Zürich, ² Universität Bielefeld, ³ Stanford University

Einleitung

Finite Elemente Methoden (FEM) haben sich sowohl in den Ingenieurwissenschaften, der angewandten Mathematik, als auch in der Computergrafik zu einem unverzichtbaren Werkzeug für die Simulation physikalischer Phänomene entwickelt, wie z.B. für die Simulation elastisch deformierbarer Flächen und Körper [1,2]. Im Rahmen des Computational Medicine (Co-Me) Projektes [3] wurden finite Elemente Methoden erfolgreich zur benutzergesteuerten, interaktiven Gewebedeformation und Schnittsimulation in einem Simulator für Hysteroskopie-Operationen eingesetzt [4].

Aufgrund ihrer Einfachheit und der daraus resultierenden Effizienz haben sich in den meisten Anwendungsgebieten Tetraeder oder Hexaeder für die räumliche Diskretisierung der zu simulierenden Objekte durchgesetzt. Obwohl *geometrische Deformationen* von Objekten so effizient implementiert werden können, verkompliziert die Restriktion auf einen festen Elementtyp jegliche *topologische Änderungen* des Simulationsnetzes, wie sie z.B. zur Simulationen von Brüchen oder Schnitten benötigt werden. In derartigen Anwendungen müssen nach jedem Schnitt oder Bruch alle betroffenen Elemente zerteilt und die resultierenden Stücke neu in ausschließlich Tetraeder oder Hexaeder tesselliert werden. Um im Rahmen dieses sogenannten Remeshings degenerierte Elemente zu vermeiden, müssen zudem häufig neue Knoten bzw. neue Elemente eingefügt werden.

Die vorliegende Arbeit schlägt ein Ansatz vor, um Finite Elemente Methoden auf allgemeinere und flexiblere Elementtypen zu erweitern. Basierend auf Mean Value Koordinaten [5,6], welche eine Verallgemeinerung linearer baryzentrischer Koordinaten darstellen, wird eine Finite Elemente Methode für allgemeinen konvexen Polyeder-Elementen entwickelt. Durch die direkte Unterstützung beliebiger konvexer Elemente wird das Re-Tessellieren nach topologischen Änderungen effektiv vermieden, was in der Folge einfachere und robustere Schnitt- und Bruchsimulationen ermöglicht.

Methode

Im folgenden wird die verwendete interne elastischen Energie des deformierten Körpers und deren Diskretisierung beschrieben, bevor Mean Value Koordinaten eingeführt werden und die daraus resultierende FEM Formulierung auf konvexen Polyedern diskutiert wird. Für eine detailliertere Ausführung wird der interessierte Leser auf den Artikel [7] verwiesen.

Die Deformation eines dreidimensionalen Objektes mit Materialkoordinaten $\mathbf{x}=[x,y,z]$ wird durch ein Verschiebungsfeld $\mathbf{u}(\mathbf{x})=[u(\mathbf{x}),v(\mathbf{x}),w(\mathbf{x})]$ beschrieben. Die Materialdeformation wird durch einen Dehnungstensor (*strain*) gemessen, wobei in der vorliegenden Arbeit der lineare Cauchy-Tensor

$$\epsilon(\mathbf{u}) = \nabla \mathbf{u} + \nabla \mathbf{u}^T$$

verwendet wird. Linearisierungsartefakte werden mit Hilfe einer ko-rotierten Formulierung vermieden [8], welche die Rotationskomponente vor der Spannungsberechnung aus dem Verschiebungsfeld $\mathbf{u}(\mathbf{x})$ eliminiert. Die unten vorgeschlagenen Mean Value Koordinaten können allerdings ebenso im Zusammenhang mit nicht-linearen Dehnungstensen, wie z.B. dem Green-Tensor, verwendet werden. Eine Dehnung des Materials führt zu internen Spannungen, welche durch einen Spannungstensor σ (*stress*) gemessen werden. Die Annahme eines Hook'schen Materials liefert einen linearen Zusammenhang zwischen Dehnung und Spannung

$$\sigma(\mathbf{u}) = \mathbf{C} : \epsilon(\mathbf{u}) ,$$

ausgedrückt durch einen 4-Tensor \mathbf{C} bestehend aus Materialparametern. Die interne elastische Energie, welche in dem deformierten Körper gespeichert ist, wird durch

$$E_{\text{int}}(\mathbf{u}) = \int_{\Omega} \sigma(\mathbf{u}) : \epsilon(\mathbf{u}) = \int_{\Omega} \epsilon(\mathbf{u}) : \mathbf{C} : \epsilon(\mathbf{u})$$

beschrieben. Die internen elastischen Rückstellkräfte entsprechen der Variationsableitung

$$\mathbf{f}_{\text{int}}(\mathbf{x}) = (\partial E(\mathbf{u}) / \partial \mathbf{u})(\mathbf{x}) .$$

Zum Zwecke der räumlichen Diskretisierung wird das Objektvolumen Ω in finite Elemente e zerlegt, wobei in typischen Anwendungen meist Tetraeder mit linearen baryzentrischen Basisfunktionen eingesetzt werden. Die Knotenverschiebungen \mathbf{u}_i der Knoten \mathbf{x}_i werden somit durch Basisfunktionen $N_i(\mathbf{x})$ innerhalb der Elemente interpoliert. Daraus resultiert die diskrete Approximation der Verschiebung

$$\mathbf{u}(\mathbf{x}) = \sum_i \mathbf{u}_i N_i(\mathbf{x}) .$$

Um Finite Elemente Methoden auf allgemeinere Polyeder-Elemente zu erweitern, verwenden wir *verallgemeinerte* baryzentrischen Koordinaten, die sogenannten Mean Value Koordinaten. Für deren exakte Definition verweisen wir auf die Originalarbeiten [5,6], weisen aber darauf hin, daß Mean Value Koordinaten analog zu linearen baryzentrischen Koordinaten analytisch in Form von Volumenverhältnissen berechnet werden können. Um die Konvergenz der approximativen Lösung gegen die exakte Lösung unter Verfeinerung des Rechengitters, und damit auch die mathematische Aussagekraft der FEM Formulierung, sicherzustellen, müssen die verwendeten Basisfunktionen im Sobolev-Raum H^1 liegen, was intuitiv bedeutet, daß sie C^1 stetig im Inneren der Elemente und C^0 stetig über Elementgrenzen sind. Zusätzlich müssen die Basisfunktionen konstante und lineare Funktionen exakt reproduzieren, nicht-negativ sein und eine Partition der Eins bilden. Wie in [7] diskutiert, erfüllen Mean Value Koordinaten all diese Voraussetzungen und können somit ohne Bedenken in FEM Simulationen eingesetzt werden. Im folgenden seien diese also mit $N_i(\mathbf{x})$ bezeichnet.

Mit diesen Basisfunktionen folgt die weitere FEM Formulierung dem bekannten Schema. Die diskrete Approximation $\mathbf{u}(\mathbf{x})$ wird in die Energie $E(\mathbf{u})$ eingesetzt, was schliesslich zu einem linearen Gleichungssystem $\mathbf{K}\mathbf{U}=\mathbf{F}$ führt, bei dem $\mathbf{U}=[\mathbf{u}_1, \dots, \mathbf{u}_n]$ und $\mathbf{F}=[\mathbf{f}_1, \dots, \mathbf{f}_n]$ die unbekanntenen Verschiebungen bzw. die bekannten externen Kräfte an den Knoten \mathbf{x}_i bezeichnen. Diese globale Stiffness-Matrix \mathbf{K} wird aus Element-weisen Stiffness-Matrizen \mathbf{K}_e assembliert, welche wie folgt berechnet werden:

$$\mathbf{K}_e = \int_e \mathbf{B}_e^T(\mathbf{x}) \mathbf{C} \mathbf{B}_e(\mathbf{x}) d\mathbf{x} ,$$

wobei die Matrizen \mathbf{B}_e erste partielle Ableitungen der $N_i(\mathbf{x})$ enthalten und eine Approximation des Dehnungstensors beschreiben. Der Unterschied zu linearen baryzentrischen Basisfunktionen besteht nun darin, dass der obige Integrand nicht länger konstant ist und daher numerische Integration für die (Vor-)Berechnung der Elementmatrizen eingesetzt werden muss. Sobald die Stiffness Matrix \mathbf{K} ermittelt ist, folgt die dynamische Deformationssimulation durch Zeit-Integration der Differentialgleichung

$$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{D}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F} .$$

Für weitere Details und Implementierungshinweise verweisen wir auf [7].

Resultate & Diskussion

Die vorgestellte Finite Elemente Methode erlaubt mit Hilfe von Mean Value Koordinaten das Verwenden beliebiger konvexer Polyeder-Elemente. Dies vereinfacht die Implementierung von Schnitten, da ein Tetraeder oder Hexaeder im initialen Simulationsnetz nach einem Schnitt in zwei Elemente (i.a. keine Tetraeder/Hexaeder) zerfällt. Diese Elemente müssen nun nicht länger in Tetraeder zerlegt werden, sondern können direkt simuliert werden. Dies ist in Bild 1 am einfachen Beispiel eines Würfels dargestellt, dessen initiale 27 Hexaeder nach dem Schnitt zu nur 60 Elementen zerfallen.

Ein direkter Vergleich mit einer Tetraeder-basierten Schnittsimulation wird in Bild 2 gezeigt. Hier wurden 17 komplexe Schnitte auf ein aus 8278 Tetraedern bestehendes Bunny-Modell angewandt. Dies führt mit unserer Methode zu 44k Elementen, von denen 51% Tetraeder und 49% allgemeine Polyeder sind. Im Gegensatz dazu ergibt die Methode von [4] aufgrund der steten Neu-Tessellierung letztendlich ca. 300k Tetraeder-Elemente. Die vorgestellte Methode vermeidet das komplizierte Remeshing völlig, erleichtert dadurch die Implementierung und ermöglicht zudem Schnittsimulationen mit deutlich reduzierter Elementzahl.

Der Nachteil der Methode ist die komplexere numerische Integration der Elementmatrizen \mathbf{K}_e . Da aber Mean Value Koordinaten eine echte Verallgemeinerung linearer baryzentrischer Koordinaten darstellen, und somit im Falle von Tetraedern eben diese exakt reproduzieren, muß die aufwendige Integration ausschließlich für nicht-Tetraeder verwendet werden. Es kann also für alle initialen Tetraeder die existierende effiziente analytische Integration verwendet werden, so dass nur für neue, aus Schnitten resultierende Elemente, auf die aufwendige numerische Integration zurückgegriffen werden muß.

Literatur

1. Bathe KJ. Finite Element Procedures. Prentice Hall, 1995.
2. Nealen A, Müller M, Keiser R, Boxerman E, Carlson M. Physically Based Deformable Modeling in Computer Graphics. Computer Graphics Forum 2006; 25(4):809-36.
3. Computer-Aided and Image-Guided Medical Interventions. www.co-me.ch.
4. Steinemann D, Harders M, Gross M, Szekely G. Hybrid Cutting of Deformable Solids. In Proc. of IEEE Virtual Reality 2006, 35-42.
5. Ju T, Schaefer S, Warren J. Mean Value Coordinates for Closed Triangular Meshes. ACM Trans. on Graphics 2005; 24(3):561-6.
6. Floater M, Kos G, Reimers M. Mean Value Coordinates in 3D. Computer Aided Geometric Design 2005; 22:623-31.
7. Wicke M, Botsch M, Gross. A finite element method on convex polyhedra. Computer Graphics Forum 2007; 26(3):355-64.
8. Müller M, Gross M. Interactive Virtual Materials. In Proc. of Graphics Interface 2004, 239-46.

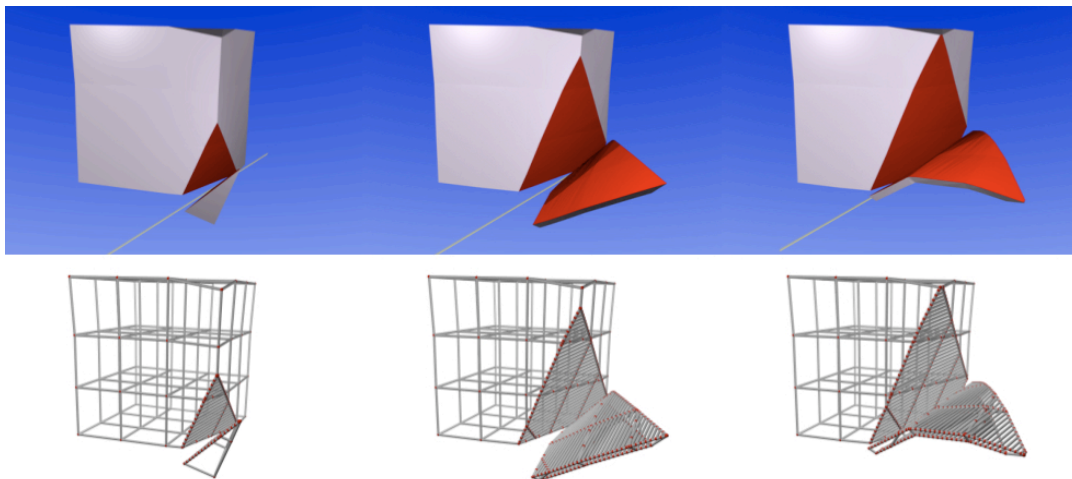


Bild 1: Oben: Mehrere planare Schnitte durch ein einfaches Block-Modell (27 Hexaeder). Unten: Gittermodell.

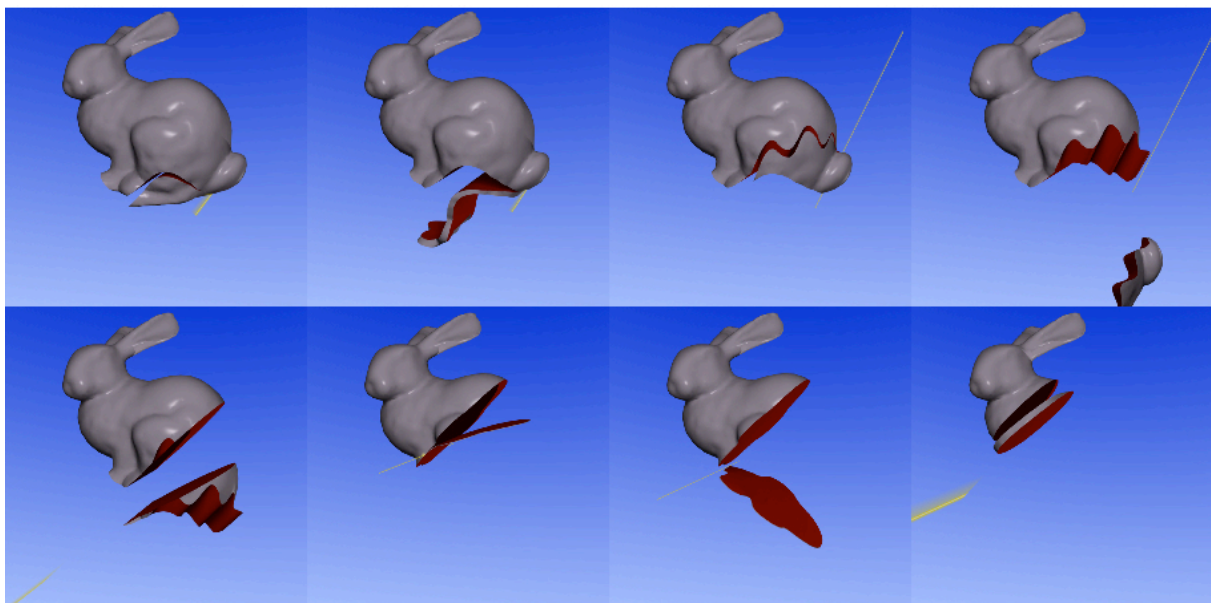


Bild 2: Mehrere komplexe Schnitte durch ein mit 8278 Tetraedern diskretisiertes Modell.