FAST FINITE ELEMENTS FOR SURGERY SIMULATION

MORTEN BRO-NIELSEN, Ph.D.

HT Medical, Inc.
Rockville, Maryland, USA

and

Department of Mathematical Modelling
Technical University of Denmark, DENMARK

e-mail: bro@ht.com WWW: http://www.imm.dtu.dk/~bro/

Abstract

This paper discusses volumetric deformable models for modeling human body parts and organs in surgery simulation systems. These models are built using finite element models for linear elastic materials. To achieve real-time response condensation has been applied to the system stiffness matrix, and selective matrix vector multiplication has been used to minimize the computational cost.

1 Introduction

Surgery simulation technology differs from traditional computer graphics by the fact that the user touch and interact with objects, ie. organs or other human tissue, in the rendered scene. Because the nature of the interaction rarely can be completely anticipated, computer graphics animations are insufficient for creating a realistic response. Instead the physical behavior of objects needs to be modeled in real-time during the user-interaction.

Developing real-time algorithms for physically-based modeling of volumetric 3D objects is, therefore, currently the central problem in surgery simulation research.

Modeling deformation of the patient anatomy has traditionally been accomplished by using surface models (eg. [8,9]), but recent work on volumetric mass-spring models [6] and Finite Element (FE) models [1,3,7] have shown that 3D volumetric deformable patient organs can be modeled in real-time.

In [1] Fast Finite Element (FFE) models for linear elastic deformation were proposed for use in surgery simulation. These models use condensation techniques to reduce the complexity of the system equations and thereby achieve a considerable speed-up compared to the volumetric models in [3,6,7].

In principle a FFE models with N surface nodes and M interior nodes have the same complexity as the FE or mass-spring models in [3,6,7] with a total of N interior and surface nodes. The number of interior nodes in a FFE model do not play any role in the computation, since they are removed using condensation in a pre-computation step.

This paper discusses real-time simulation of volumetric deformable objects using the 3D solid volumetric Fast Finite Element models and demonstrate the behavior for a human leg..

2 Theory

The deformation algorithm have been developed using mesh-based 3D Finite Element (FE) models of linear elastic materials. By using linear elasticity as the basic model we implicitly make a number of assumptions regarding the physical material that is being modeled. Most importantly linear elastic models are only valid for very small deformations and strains. They are typically correct for such rigid structures as metal beams, buildings etc. Although they are used extensively in modeling, the visual result of large deformation modeling using linear elasticity is seldom satisfactory But when used with FE these models lead to linear matrix systems $\mathbf{Ku} = \mathbf{f}$ which are easy to solve and fast.

2.1 Condensation

The linear matrix system $\mathbf{Ku} = \mathbf{f}$ models the behavior of the *solid* object. This includes both surface nodes as well as the internal nodes of the model. But for simulation purposes we are usually only interested in the behavior of the surface nodes since these are the only *visible* nodes. We, therefore, use condensation [5] to remove the internal nodes from the matrix equation.

The matrix equation for the condensed problem has the same size as would result from a FE *surface* model. But, it is important to understand that it shows *exactly* the same behavior for the surface nodes as the original *solid volumetric* system.

Without loss of generality, let us assume that the nodes of the FE model have been ordered with the surface nodes first, followed by the internal nodes. Using this ordering we can rewrite the linear system as a block matrix system (*s*urface / *i*nternal):

$$\begin{bmatrix} \mathbf{K}_{ss} & \mathbf{K}_{si} \\ \mathbf{K}_{is} & \mathbf{K}_{ii} \end{bmatrix} \begin{bmatrix} \mathbf{u}_s \\ \mathbf{u}_i \end{bmatrix} = \begin{bmatrix} \mathbf{f}_s \\ \mathbf{f}_i \end{bmatrix}$$

From this block matrix system we create a new linear matrix system

$$\hat{\mathbf{K}}_{ss}\mathbf{u}_s = \hat{\mathbf{f}}_s$$

which only involves the variables of the *surface* nodes:

$$\hat{\mathbf{K}}_{ss} = \mathbf{K}_{ss} - \mathbf{K}_{si}\mathbf{K}_{ii}^{-1}\mathbf{K}_{is}$$
$$\hat{\mathbf{f}}_s = \mathbf{f}_s - \mathbf{K}_{si}\mathbf{K}_{ii}^{-1}\mathbf{f}_i$$

The displacement of the internal nodes can still be calculated using

$$\mathbf{u}_i = \mathbf{K}_{ii}^{-1}(\mathbf{f}_i - \mathbf{K}_{is}\mathbf{u}_s)$$

Note, that if no forces are applied to internal nodes, $\hat{\mathbf{f}}_s = \mathbf{f}_s$.

Generally the new stiffness matrix will be dense rather than sparse as the original system. But, since we intend to solve the system by inverting the stiffness matrix in the pre-calculation stage, this is not important. Without loss of generality, we will understand that both the original system and the condensed system $\hat{\mathbf{K}}_{ss}\mathbf{u}_s = \hat{\mathbf{f}}_s$ can be used when the following text refers to the original system $\mathbf{Ku} = \mathbf{f}$.

2.2 Solving $\mathbf{Ku} = \mathbf{f}$ using Selective Matrix Vector Multiplication

Solving the linear matrix system using the inverted stiffness matrix is performed using $\mathbf{u} = \mathbf{K}^{-1}\mathbf{f}$. If only a few positions of the force vector are non-zero, clearly standard matrix vector multiplication would involve a large number of superfluous multiplications. We note that

$$\mathbf{u} = \mathbf{K}^{-1}\mathbf{f} = \sum_i \mathbf{K}_{*\mathbf{i}}^{-1}\mathbf{f}_i$$

where $\mathbf{K}_{*i}^{-1}$ is the *i*'th column vector of $\mathbf{K}^{-1}$ and $\mathbf{f}_i$ the *i*'th element of $\mathbf{f}$. Since the majority of the $\mathbf{f}_i$ are zero, we restrict *i* to run through only the positions of $\mathbf{f}$ for which $\mathbf{f}_i \neq 0$ [1]. If *n* of the *N* positions in $\mathbf{f}$ are non-zero this will reduce the complexity to *O(n/N)* times the time of a normal matrix vector multiplication. We call this approach Selective Matrix Vector Multiplication (SMVM).

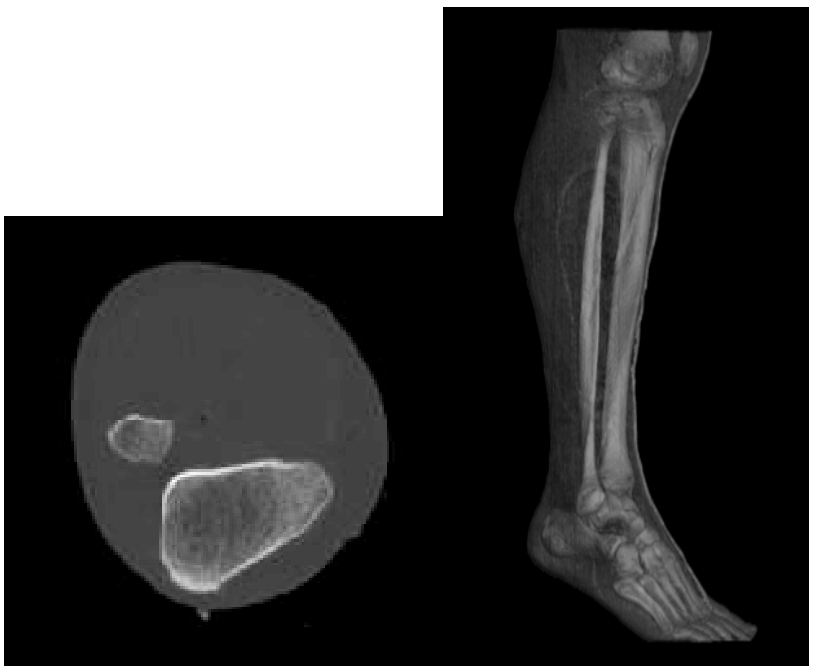

Figure 1. Voxel data from the visible human data set

3 Simulation system

In this section we describe how we generate the FE mesh model of the physical organ, limb etc. is generated, and show the simulation system which has been implemented.

In addition to a range of simple box-like structures, data from the Visible Human project has been used to test the algorithms described above.

Since the Visible Human data set is voxel-based (see figure 1) it was necessary to convert it into a mesh model. To do this, the Mvox software package [2] were first used to manually draw contours on the boundary of the skin and bone in the voxel data. The Nuages software [4] was then used to create a 3D tetrahedral mesh model of the leg. The result was the FE mesh model shown in figure 3.

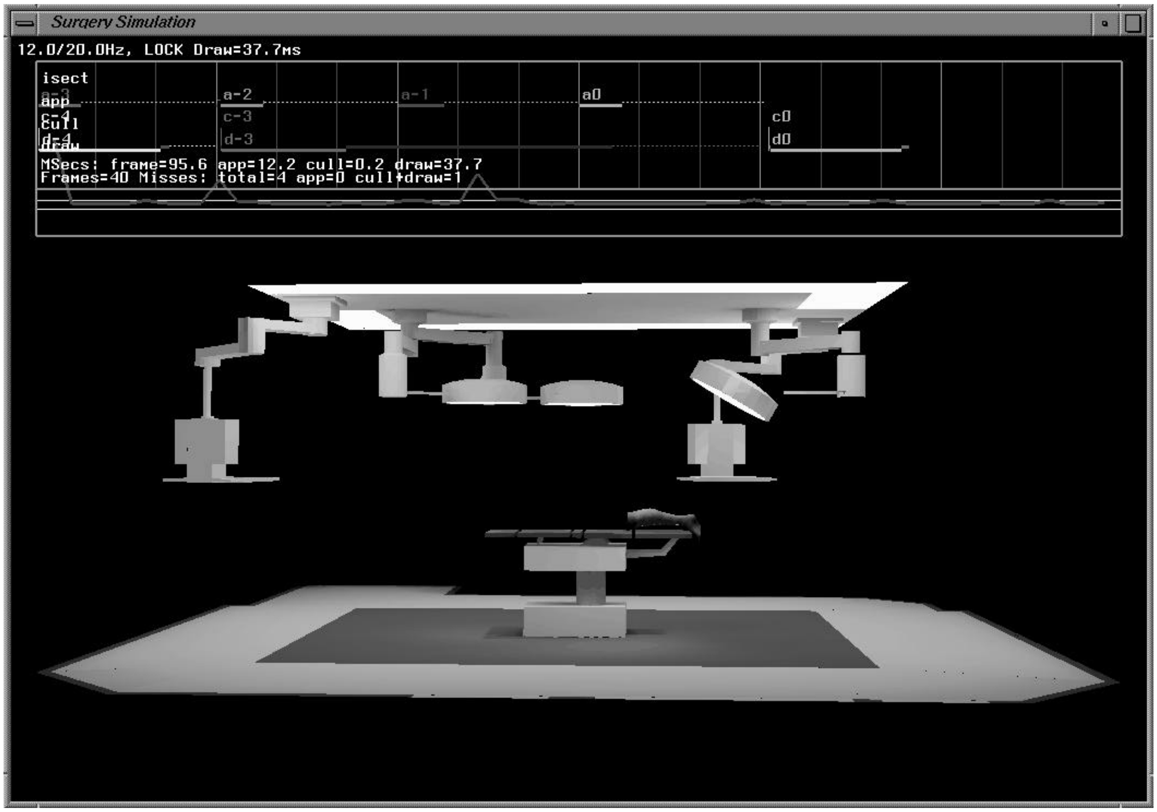

Figure 2. Simulation system implemented using SGI Performer

The simulation system has been implemented on an Silicon Graphics ONYX with four Mips R4400 processors using the SGI Performer graphics library. SGI Performer helps the programmer create parallel pipe-lining software by providing the basic tools for communication, shared memory etc.

Figure 2 shows a screen dump with the Virtual Operating room environment and the leg lying on the operating table. Figure 3 shows the surface of the FE mesh shown in the simulator.

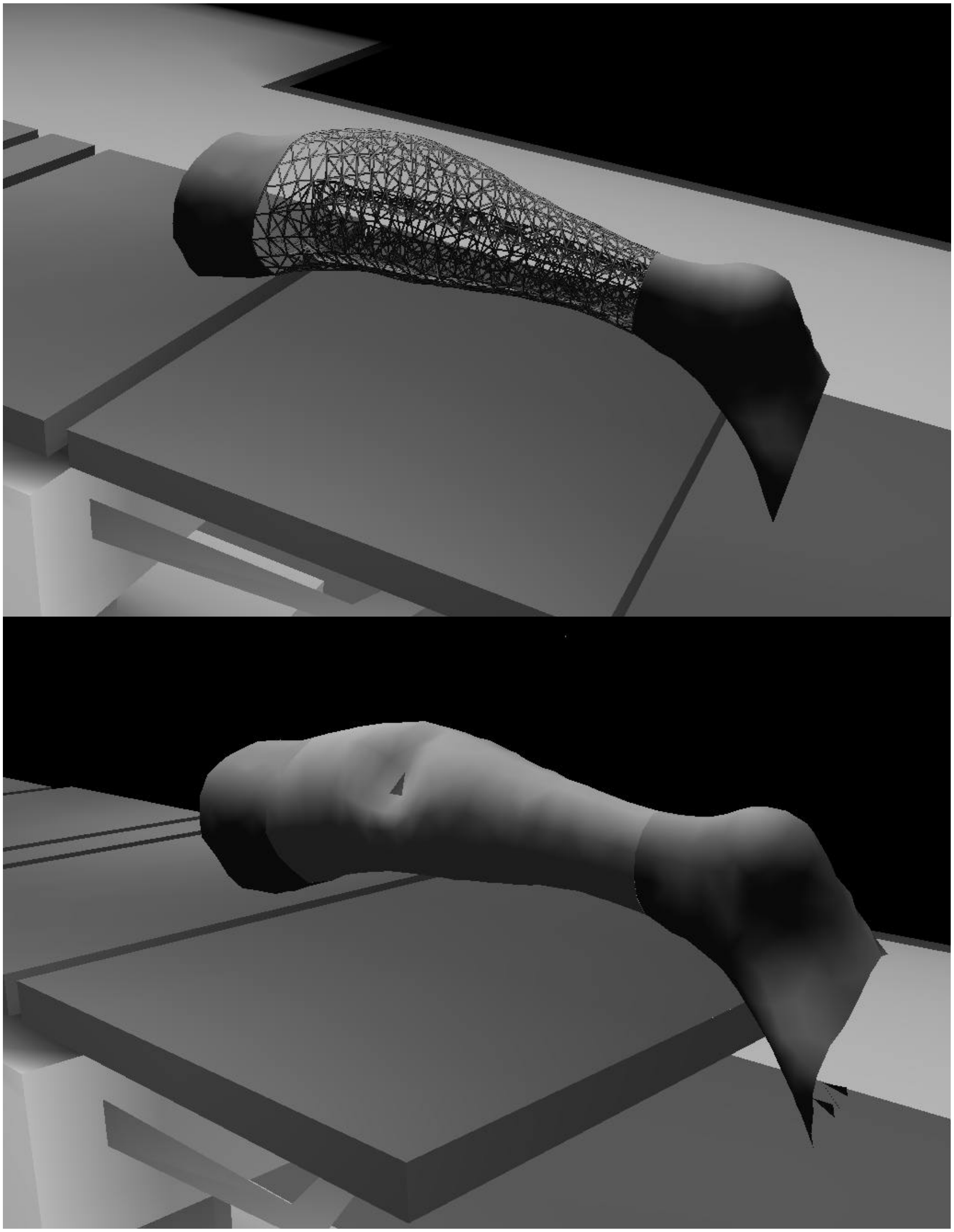

Figure 3. Top: Wireframe model of lower leg in simulator. Bottom: Simulation of pushing on a lower leg.

5 Conclusion

In this paper we have described a method for real-time simulation of elastic deformation of a volumetric solid based on linear elastic Fast Finite Elements (FFE).

An example using a leg from the Visible Human data set with 700 system nodes (condensed system with only surface nodes) ran comfortably with a single processor using only 1/3 of a frame (20 frames/second) when forces were applied to 3 nodes. This included calculation of the deformation and also basic processing.

We are now in the process of implementing these Fast Finite Element models for use in the DARPA Abdominal Trauma simulator.

4 Related papers

[1] M. Bro-Nielsen and S. Cotin, Real-time volumetric deformable models for surgery simulation using finite elements and condensation, Computer Graphics Forum, 15(3):57-66 (Eurographics'96), 1996

[2] M. Bro-Nielsen, Mvox: Interactive 2-4D medical image and graphics visualization software, Proc. Computer Assisted Radiology (CAR'96), 1996
See also: http://www.imm.dtu.dk/~mvox/

[3] S. Cotin, H. Delingette, J.M. Clement, V. Tassetti, J. Marescaux, and N. Ayache, Volumetric deformable models for simulation of laparascopic surgery, Proc. Computer Assisted Radiology (CAR'96), pp. 793-798, 1996

[4] B. Geiger, Three-dimensional modeling of human organs and its application to diagnosis and surgical planning, INRIA Tech. Rep. 2105, Dec. 1993

[5] H. Kardestuncer, Finite element handbook, McGraw-Hill, ISBN 0-07-033305-X, 1987

[6] Ch. Kuhn, U. Kuhnapfel, H-G. Krumm, and B. Neisius, A virtual reality based training system for minimally invasive surgery, Proc. Computer Assisted Radiology (CAR'96), pp. 764-769, 1996

[7] S. Cotin, H. Delingette, J.M. Clément, M. Bro-Nielsen, N. Ayache , J. Marescaux, Geometrical and Physical Representations for a Simulator of Hepatic Surgery, Proc. MMVR-4'96, 1996

[8] S.A. Cover, N.F. Ezquerra and J.F. O'Brien, R. Rowe, T. Gadacz and E. Palm, Interactively deformable models for surgery simulation, IEEE Computer Graphics & Applications, pp. 68-75, Nov. 1993

[9] J.R. Merril, G.M. Preminger , R. Babayan, R.T. Roy, and G.L. Merril, Surgical simulation using virtual reality technology: Design, implementation, and implications, Surgical Technology III, pp. 53-60, 1994