


Toward high-speed 3D nonlinear soft tissue deformation simulations using Abaqus software

Ashraf Idkaidek¹ · Iwona Jasiuk² 

Received: 7 July 2015 / Accepted: 8 September 2015 / Published online: 26 September 2015
© Springer-Verlag London 2015

Abstract We aim to achieve a fast and accurate three-dimensional (3D) simulation of a porcine liver deformation under a surgical tool pressure using the commercial finite element software Abaqus. The liver geometry is obtained using magnetic resonance imaging, and a nonlinear constitutive law is employed to capture large deformations of the tissue. Effects of implicit versus explicit analysis schemes, element type, and mesh density on computation time are studied. We find that Abaqus explicit and implicit solvers are capable of simulating nonlinear soft tissue deformations accurately using first-order tetrahedral elements in a relatively short time by optimizing the element size. This study provides new insights and guidance on accurate and relatively fast nonlinear soft tissue simulations. Such simulations can provide force feedback during robotic surgery and allow visualization of tissue deformations for surgery planning and training of surgical residents.

Keywords Computational surgery · Nonlinear constitutive model · Numerical simulations · Mathematical models · Robotics

Electronic supplementary material The online version of this article (doi:[10.1007/s11701-015-0531-2](https://doi.org/10.1007/s11701-015-0531-2)) contains supplementary material, which is available to authorized users.

✉ Iwona Jasiuk
ijasiuk@illinois.edu

¹ Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, 205 North Mathews Ave, Urbana, IL 61801, USA

² Departments of Mechanical Science and Engineering and Bioengineering, University of Illinois at Urbana-Champaign, 1206 West Green Street, Urbana, IL 61801, USA

Introduction

Robotic surgery allows surgeons to perform complex surgical procedures using robotic arms. Advantages include small incisions, which lead to faster patient recovery. However, since surgeons have no direct contact with the tissue, soft tissue resistance feedback is not directly available. Modeling of soft tissue deformations under surgical tools interaction can provide surgeons with valuable insights into deformations of tissues during surgery. These include information on the amount of force needed to perform a given surgical task and visualization of deformations. Such knowledge can also be used for surgery planning and training of surgical residents.

Accurate soft tissue simulations must incorporate realistic material properties. Numerous experimental studies have been done to characterize mechanical properties of biological materials and organs, including liver. For example, Kemper et al. [1] performed tension tests on a human liver parenchyma at various loading rates to characterize its viscoelastic and failure properties. This study showed that the liver parenchyma is rate dependent, with higher rate tests giving higher failure stresses and lower failure strains. Also, Costin et al. [2] performed tensile tests on fresh human samples of the liver parenchyma at several loading rates.

Simulating soft tissue response due to surgical tools' interaction using linear versus nonlinear properties leads to large differences in force–displacement responses [3, 4]. Several studies used linear elastic constitutive models, but those generated results only for small deformations. For instance, Chanthasopeephan et al. [5] simulated the porcine liver cutting to enable fast haptics display using linear properties. Delingette et al. [6] described the basic components of a surgery simulator prototype using the linear

elasticity theory and finite elements method (FEM). Bro-Nielsen [7] presented the application of 3D solid volumetric finite element (FE) models to surgery simulation using the linear elastic theory.

Soft tissue simulations have also accounted for nonlinear material properties. For example, Grand et al. [8] used average nodal pressure tetrahedral elements for better handling of a volumetric locking numerical problem to simulate soft tissue deformations. This method requires a higher computational time compared to traditional FEM. Kevin et al. [9] developed a real-time haptics-enabled simulator for probing soft tissue using the FEM with a nonlinear experimentally based constitutive law. This study accounted for the soft tissue material nonlinearity but it did not focus on generating fast simulations using 3D nonlinear FE models. Ahn et al. [10] did a 3D simulation of indentation of porcine liver and correlated it with experimental results. The liver tissue properties were assumed to be incompressible and nonlinear. Again, this study focused on generating accurate simulation results without considering a simulation time. Picinbono et al. [11] developed a simulator for laparoscopic liver surgery to enable fast haptics display of cutting. He accounted for nonlinear elastic and anisotropic material behavior using a simple hyperelastic model. Wu et al. [12] proposed a real-time soft tissue deformation analysis by using nonlinear FEM and adaptive meshing techniques. The analysis included material nonlinearity, but no details were provided regarding a material constitutive model used in their simulations.

Thus, the modeling of soft tissue deformations due to interaction with surgical tools is a challenging and still open research topic. Prior simulations idealized mechanical properties and/or required long simulation times, as discussed above, which make them not fully suitable for robotic surgery and other medical implementations.

In this paper, we address this problem by simulating the deformation of a porcine liver under a surgical tool while accounting for problem nonlinearities: contacts, large deformations, and nonlinear material properties. More specifically, we investigate the effects of the implicit versus explicit analysis schemes, mesh size, and element type on the computational time and accuracy of results. Results obtained from this study provide guidance on accurate and efficient algorithms for soft tissue simulations.

Methods

Porcine liver MRI scanning

The porcine liver was scanned using magnetic resonance imaging (MRI) with 0.9 mm^3 resolution. The MRI scan was performed at the Beckman Institute at University of

Illinois at Urbana-Champaign. It generated multiple IMA type files, and Simpleware software was used to generate the 3D volume geometry and the FE models. The scanned liver dimensions were $277 \text{ mm} \times 290 \text{ mm} \times 53 \text{ mm}$. (Fig. 1a, b).

Soft tissue nonlinear constitutive model

To model a nonlinear behavior of the liver tissue, a hyperelastic model involving the Ogden strain energy potential [13, 14], available in Abaqus, was used, as shown in Eq. 1:

$$U = \sum_{i=1}^N \frac{2\mu}{\alpha_i^2} (\bar{\lambda}_1^{\alpha_i} + \bar{\lambda}_2^{\alpha_i} + \bar{\lambda}_3^{\alpha_i} - 3) + \sum_{i=1}^N \frac{1}{D_i} (J^{\text{del}} - 1)^{2i}, \quad (1)$$

where $\bar{\lambda}_i$ represent the deviatoric principal stretches, J^{el} is the elastic volume ratio, N is the order of the polynomial, and μ_i , α_i , and D_i are material constants. In this study, a third-order polynomial form of the Ogden model was used to represent the liver material properties based on tensile loading tests reported by Kemper et al. [1]. These test results presented Second Piola–Kirchhoff stress versus Green–Lagrange strain. Engineering stress versus engineering strain are needed as inputs for the Abaqus software. Therefore, these test results were converted to the appropriate form based on solid mechanics principles. Since soft tissues are considered roughly incompressible materials with Poisson's ratio in the range between 0.45 and 0.49 [15], the Poisson ratio of 0.48 was assumed in this study. Finally, the Abaqus software used these inputs to calculate the Ogden model material coefficients.

Finite element analysis: preprocessing

FE analysis problem was defined by applying translational boundary condition constraints on the liver bottom nodes (Abaqus has no rotational DOF for C3D4, C3D8, and C3D8R elements) as shown in Fig. 1d. A surgical knife, tapered toward the bottom at 0.54 degrees with a rounded 0.1 mm radius tip (Fig. 1c), was modeled using first-order hexahedral elements and elastic properties of steel. Note that representing the surgical tool as a rigid surface did not give noticeable difference in a simulation speed. This is because the simulation time was mostly taken by contact and soft tissue deformation calculations. The analysis involved applying the 10-mm vertical displacement to the knife as shown in Fig. 1e.

Six different FE analysis models were developed: two models were built using hexahedral elements while the other four were built using tetrahedral elements (Fig. 2). Each FE model was developed with the relatively constant

Fig. 1 Liver 3D volume geometry and 3D finite element model. **a** Liver geometry top view. **b** Liver geometry side view. **c** Surgical tool (knife) cross section. **d** Finite element model boundary conditions; all highlighted nodes (in red) are restrained in all translational directions (Abaqus has no rotational DOF for C3D4, C3D8 and C3D8R elements). **e** Liver finite element model and the surgical knife

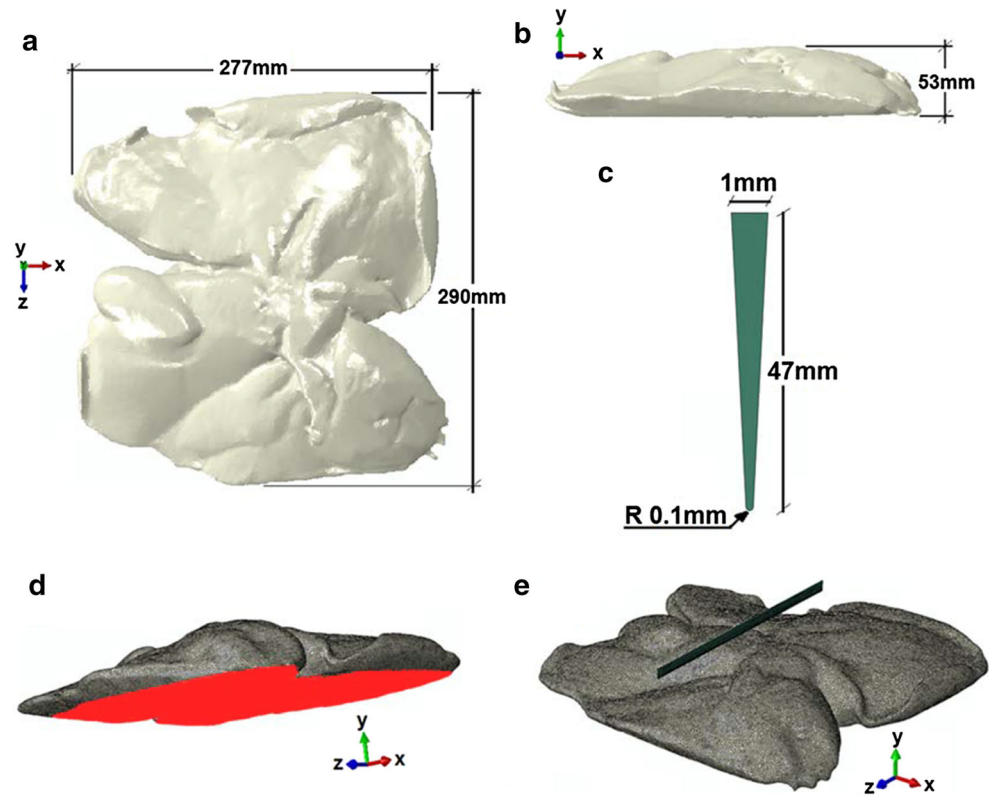
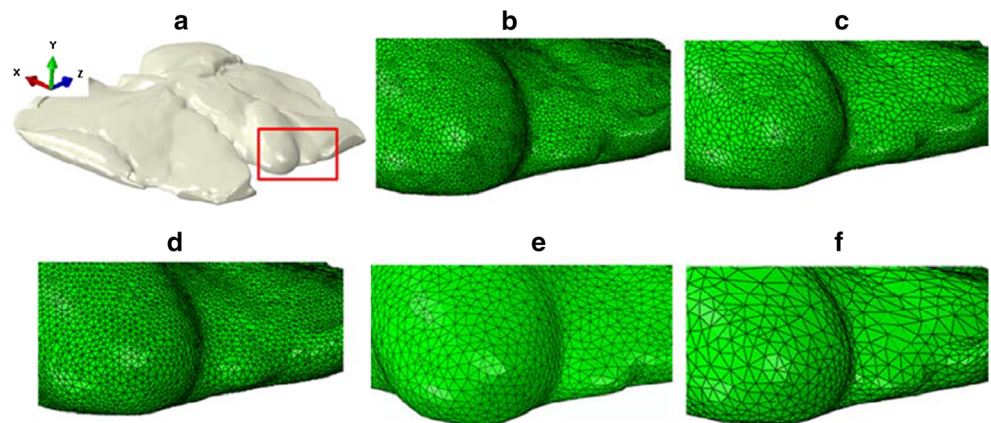


Fig. 2 Liver finite element (FE) models. **a** ISO view of the liver geometry, **b** FE model built using 841,146 first-order hexahedral elements. **c** FE model built using 358,390 first-order hexahedral elements. **d** FE model built using 899,153 first-order tetrahedral elements. **e** FE model built using 237,060 first-order tetrahedral elements. **f** FE model built using 116,371 first/second-order tetrahedral elements



average element size (no local refinement or adaptive meshing) to generate a fair simulation time comparison.

Abaqus solvers and simulation time

As background information, Abaqus offers implicit and explicit solvers. The implicit algorithm provides accurate results when solving quasi-static problems [16]. On the other hand, achieving equilibrium is a challenge due to problem complexity (knife-tissue contact, large deformations, and soft tissue nonlinear properties) [16].

The explicit solver was developed to model high-speed events. No energy dissipation is expected when solving a

soft tissue deformation problem due to its quasi-static nature [16]. Therefore, performing this type of analysis using an explicit solver should be acceptable as long as the model internal and external energies are comparable. The major advantage of using an explicit solver over an implicit solver is that the simulation will always converge. This is because the explicit solver depends on time steps without the need to keep checking if an equilibrium is achieved. On the other hand, the explicit solver requires a high computational time when compared to implicit solver. To overcome these issues two approaches were used:

- *Increase load rate* This artificially increases the material strain rate by the same load rate factor. To preserve

Table 1 Simulation speed comparison using Abaqus implicit solver ($X = 2:03:33 = \text{hh:mm:ss}$)

Iteration number	Iteration description	Element type	Number of nodes	Number of elements	Max. vertical displacement Running time
1	Quasi-static	Hex. first-order elements	997,796	841,146	6.21 mm, 54.80X
2	Quasi-static	Hex. first-order elements	433,092	358,390	6.82 mm, 4.69X
3	Static	Tet. first-order elements	226,875	899,153	1.47 mm, 0.58X
4	Quasi-static	Tet. first-order elements	226,875	899,153	9.03 mm, 4.71X
5	Static	Tet. first-order elements	67,893	116,371	2.58 mm, 0.31X
6	Quasi-static	Tet. first-order elements	67,893	116,371	2.58 mm, 0.28X
7	Quasi-static	Tet. second-order elements	179,686	116,371	7.49 mm 3.07X
8	Quasi-static	Tet. first-order elements	95,307	237,060	8.01 mm, X

Bold values indicate faster simulation iteration using implicit solver

a quasi-static response, it was noticed that the impact velocity should be less than 1 % of the material wave speed.

- *Apply mass scaling* Here, the stable time increment increases by a factor of f when the material density is artificially increased by a factor of f^2 as shown in Eqs. 2 and 3 below.

The increase in the load rate and/or mass scaling will reduce the Abaqus explicit simulation time significantly but inertia forces need to be insignificant to insure accurate results.

There are two ways to perform mass scaling when using the explicit solver, fixed mass scaling and variable mass scaling [16]. In this study, the variable mass scaling was used, where scaling was adjusted based on simulation behavior during the step to control Abaqus explicit simulation time.

The Abaqus explicit algorithm requires the following time increment condition to insure a stable and accurate solution [16]:

$$\Delta t \leq \frac{1}{\pi v_{\max}}, \quad (2)$$

where v_{\max} is the FE model highest natural frequency.

The highest natural frequency depends on the time taken by a dilatational stress wave to cross the smallest element in the FE model. Therefore, the element stable time increment is equivalent to [16]:

$$\Delta t \leq \frac{L_e}{C_d}, \quad (3)$$

where L_e is the characteristic element length, C_d is the dilatational wave speed $= \sqrt{\frac{M}{\rho}}$, M is the P-wave modulus $= \frac{E(1-\nu)}{(1+\nu)(1-2\nu)}$, E is the Young's modulus, ν is the Poisson's ratio, ρ is the material density.

Results and discussion

FE simulation speed results, using an implicit solver, are summarized in Table 1. All iterations were performed using the Abaqus version 6.13 and 32 cores (2.9 GHz and 64 GB RAM each). Direct solver was used for all the implicit analyses to preserve the results accuracy.

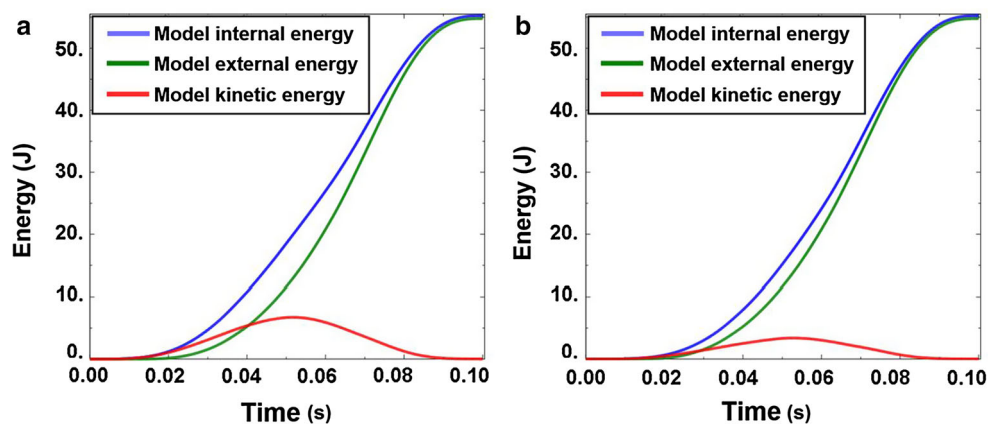
Reduced integration was used for models with hexahedral elements because traditional hyperelastic hexahedral elements were not able to achieve equilibrium with acceptable tolerance. Even though reduced integration hexahedral elements converged better compared to traditional hexahedral elements, the static (implicit) nonlinear FE analysis did not achieve a full convergence due to large deformations and soft tissue material nonlinearity. Therefore, a quasi-static implicit FE analysis was considered. It was noticed that achieving equilibrium using a quasi-static implicit scheme is better than a static implicit scheme for this kind of analysis. On the other hand, it requires more simulation time and yet it is challenging to fully converge. Resolving an implicit simulation convergence problem was not considered in this study to be able to make a fair comparison between different solvers' ability to complete such simulations.

Table 2 Simulation speed comparison using Abaqus explicit solver ($X = 02:31:32 = \text{hh:mm:ss}$)

Iteration number	Iteration description	Element type	Number of nodes	Number of elements	Explicit analysis time	Mass scaling	Max. vertical displacement Running time
1	Double precision	Hex. first-order elements	997,796	841,146	0.1	No	5 mm 42.29X
2	Double precision	Tet. first-order elements	226,875	899,153	0.1	No	10 mm 33.4X
3	Double precision	Tet. first-order elements	95,307	237,060	0.1	No	10 mm 14.14X
4	Double precision	Tet. first-order elements	67,893	116,371	0.1	No	10 mm 10X
5	Double precision	Tet. first-order elements	67,893	116,371	0.05	No	10 mm 3.88X
6	Double precision	Tet. first-order elements	67,893	116,371	0.1	$dt = 1.5 \times 10^{-7}$	10 mm 0.52X
7	Double precision	Tet. first-order elements	67,893	116,371	0.1	$dt = 1.0 \times 10^{-7}$	10 mm 0.73X
8	Double precision	Tet. first-order elements	67,893	116,371	0.1	$dt = 0.9 \times 10^{-7}$	10 mm 1.4X
9	Single precision	Tet. first-order elements	67,893	116,371	0.1	$dt = 0.9 \times 10^{-7}$	10 mm X
10	Double precision	Tet. first-order elements	67,893	116,371	0.1	$dt = 0.5 \times 10^{-7}$	10 mm 1.28X

Bold values indicate faster simulation iteration using explicit solver

Fig. 3 Abaqus explicit model energy response (iteration 6 and iteration 7). **a** Iteration 6 model energy response using Abaqus explicit and mass scaling with minimum $dt = 1.5 \times 10^{-7}$. **b** Iteration 7 model energy response using Abaqus explicit and mass scaling with minimum $dt = 1.0 \times 10^{-7}$



As shown in Table 1, the use of the Abaqus implicit solver did not lead to the 100 % completion of any of the iterations, even when employing a quasi-static algorithm and regardless of the element type. The Iteration 1 used a FE model with 841,146 first-order hexahedral elements, the simulation only completed 62 % of the analysis and it was extremely slow. Therefore, a coarser FE model was considered (iteration 2), but it was able to complete only 68 % of the analysis in about 10 h. Due to a long simulation time, tetrahedral elements were used. Static and quasi-

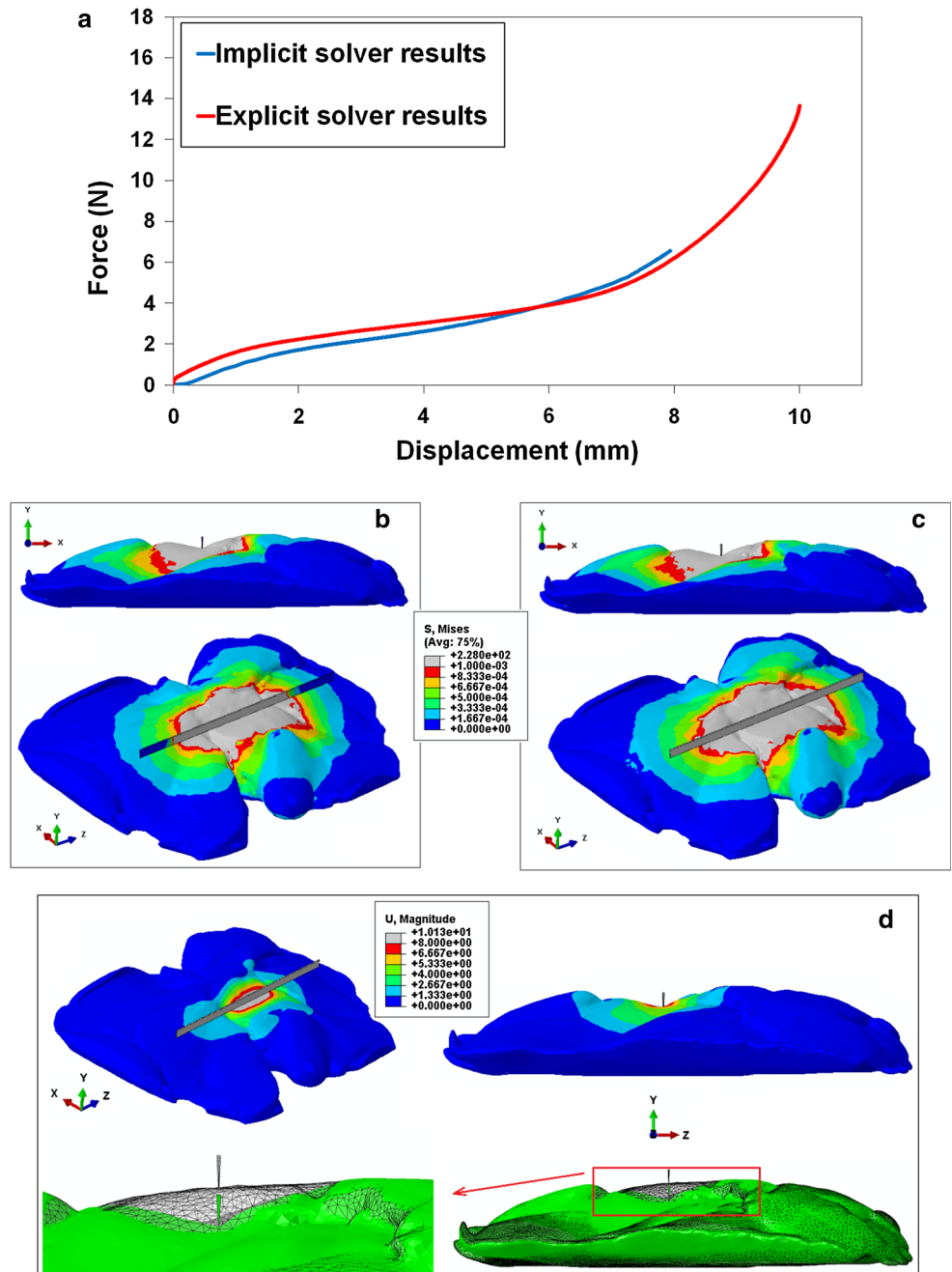
static simulations were performed, respectively (Iteration 3 and iteration 4, respectively); both iterations used FE model with 899,153 first-order tetrahedral elements. The static simulation was able to complete close to 15 % of the analysis. On the other hand the quasi-static analysis completed 90 % of the analysis but simulation time was relatively high. To further reduce the simulation time, a coarser mesh was considered (116,371 tetrahedral elements) and three iterations were performed: static analysis using first-order elements (iteration 5), quasi-static analysis using first

and second-order elements (iteration 6 and iteration 7, respectively). Due to the coarse FE model, using first-order elements was not enough to achieve convergence even when using a quasi-static algorithm. On the other hand, using second-order elements allowed to complete close to 75 % of the simulation, but simulation time was still relatively high. The finer FE model with 237,060 first-order tetrahedral elements was able to complete 80 % of the analysis in 2 h and 3 min (iteration 8). Therefore, this iteration was considered best among all eight iterations

performed using the implicit solver. This iteration is marked in bold in Table 1.

FE simulation speed results, using the explicit solver, are summarized in Table 2. All iterations were performed using the Abaqus version 6.13, double precision (except iteration 9), and 32 cores (2.9 GHz and 64 GB RAM each). Explicit solver fully completed all iterations simulations. Based on these results, using hexahedral hyperelastic elements and explicit scheme requires extremely long time to complete a simulation. On the other hand, using tetrahedral

Fig. 4 Implicit and explicit solvers simulations results. **a** Implicit versus explicit solvers reaction force results—FE model with 237,060 first-order tetrahedral elements and 95,307 nodes. **b** Implicit solver results—8 mm vertical knife deformation and Mises stress distribution. **c** Explicit solver results—8 mm vertical knife deformation and Mises stress distribution. **d** Model deformation under 10 mm vertical knife displacement



hyperelastic elements, one can solve the problem relatively fast while preserving the results accuracy.

Simulation times of iterations 1–5 were relatively high (no mass scaling was used) while the simulation times of Iterations 6 and 7 were relatively low. On the other hand, both iterations results were considered inaccurate because both models experienced high dynamic response due to a high element stable time increment (dt), where a model's kinetic energy was relatively high compared to a model's internal energy (Fig. 3). The iterations 8 and 9 were similar except that the iteration 8 was completed using a double precision while the iteration 9 was completed using a single precision. Both iterations provided similar results in terms of accuracy, but the iteration 9 was 40 % faster compared to the iteration 8. Using the mass scaling with element stable time increment less than 0.9×10^{-7} did not improve results accuracy. On the other hand, it increased the simulation time (iteration 10). Iteration 9 was completed in 2 h and 31 min using the Abaqus explicit solver. This was a relatively short time compared to other iteration analysis times. Therefore, the iteration 9 was considered best among all ten iterations performed using the explicit solver. This iteration is marked in bold in Table 2.

The force versus displacement results using implicit and explicit solvers are close as shown in Fig. 4. When the explicit solver was used, a reaction force oscillation was noticed due to a dynamic behavior. Therefore, the Butterworth filter was used to eliminate such oscillation. The inertial reaction response showed a slight difference between the implicit solver result and the explicit solver result due to an explicit solver dynamic effect. This difference is considered acceptable because it is not affecting the overall liver response or von Mises stress distribution as shown in Fig. 4.

Abaqus implicit solver was able to accurately simulate the nonlinear liver deformation under surgical tool vertical displacement in a relatively short time. However, the analysis convergence was always a challenge. Therefore, using a fine mesh is essential for the simulation to complete. Abaqus explicit solver was also able to complete the simulation with similar accuracy compared to the implicit solver and without going through the simulation convergence problem. On the other hand, it required higher simulation time compared to the implicit solver, which was compensated by increasing the load rate and using mass scaling.

Table 3 in Appendix provides a summary of the analysis scheme effects on the simulation speeds and convergence. In addition, Figs. 5, 6, 7, 8, 9, 10, 11, and 12 in the Appendix show the liver deformation and Mises stress contours due to surgical knife pressure at various

displacements. Also an animation video is available as Supplementary Material.

Conclusions

This study provides guidance on how to simulate soft tissue deformations under surgical tools displacement (and resulting pressure), while taking into account problem's nonlinearity and soft tissue constitutive nonlinear model, in a relatively short time, using the Abaqus implicit and explicit solvers. Accurate results were obtained using first-order tetrahedral elements with relatively fine mesh in a relatively short time. Therefore, using first or second-order hexahedral elements or second-order tetrahedral elements would not necessarily improve results accuracy but would increase the simulation time.

Both implicit and explicit analysis schemes are capable of solving the problem in comparable analysis times while preserving results accuracy. On the other hand, solving the problem using implicit static or quasi-static algorithms is very challenging to converge.

In this paper, we simulated soft tissue deformation under a surgical knife in a relatively short time. Because this study depends on iterations simulation time comparison, 32 Central Processing Units (CPUs: 2.9 GHz and 64 GB RAM each) were used for all iterations even though the current computing power is capable of using many more CPUs. Therefore, the shortest simulation time reported in this study is expected to be many times faster when using a supercomputer and/or introducing graphics processing unit (GPU) capabilities.

Acknowledgments The authors would like to acknowledge help of research scientist Ryan Larsen for performing liver MRI scanning at the Beckman Institute at University of Illinois at Urbana-Champaign. We would also like to thank Dr. Richard H. Pearl from OSF Saint Francis Medical Center in Peoria, IL, and Dr. T. “Kesh” Kesavadas from University of Illinois at Urbana-Champaign for helpful discussions.

Compliance with ethical standards

Disclosure of potential Conflicts of Interest Authors AI and IJ declare that they have no conflict of interest.

Research Involving Human Participants/Animals All applicable international, national, and/or institutional guidelines for the care and use of animals were followed.

Informed consent Not applicable.

Appendix

See Table 3 and Figs. 5, 6, 7, 8, 9, 10, 11, and 12.

Table 3 Simulation time and convergence comparison (For mesh density information, please see Tables 1 and 2)

Element types Algorithms	First-order Hex element	First-order reduced integration Hex element	Second-order Tet element	First-order Tet element ^a	First-order Tet element with mass scaling
Static implicit algorithm	Simulation time is relatively slow Simulation convergence is extremely challenging	Simulation time is relatively slow Simulation convergence is extremely challenging	Simulation time is relatively acceptable Simulation convergence is extremely challenging	Simulation time is relatively acceptable Simulation convergence is extremely challenging	
Quasi-static implicit algorithm	Simulation time is extremely slow Simulation convergence is extremely challenging	Simulation time is extremely slow but better than using fully integrated first-order Hex element Simulation convergence is extremely challenging but better than using fully integrated first-order Hex element	Simulation time is relatively slow Simulation convergence is acceptable	Simulation time is relatively acceptable Simulation convergence is acceptable	
Dynamic explicit algorithm	Simulation time is extremely slow May encounter convergence challenges due to high loading rate ^b	Simulation time is extremely slow May encounter convergence challenges due to high loading rate ^b	Simulation time is extremely slow but relatively faster than using Hex elements Simulation convergence is not an issue	Simulation time is slow Simulation convergence is not an issue	Simulation time is relatively acceptable Simulation convergence is not an issue

^a A model built with first-order Tet elements is expected to be relatively finer than a model built with second-order Tet elements

^b When using the explicit solver, a model built using fully integrated hyperelastic Hex elements has a better chance to converge than a model built with reduced integration hyperelastic Hex elements

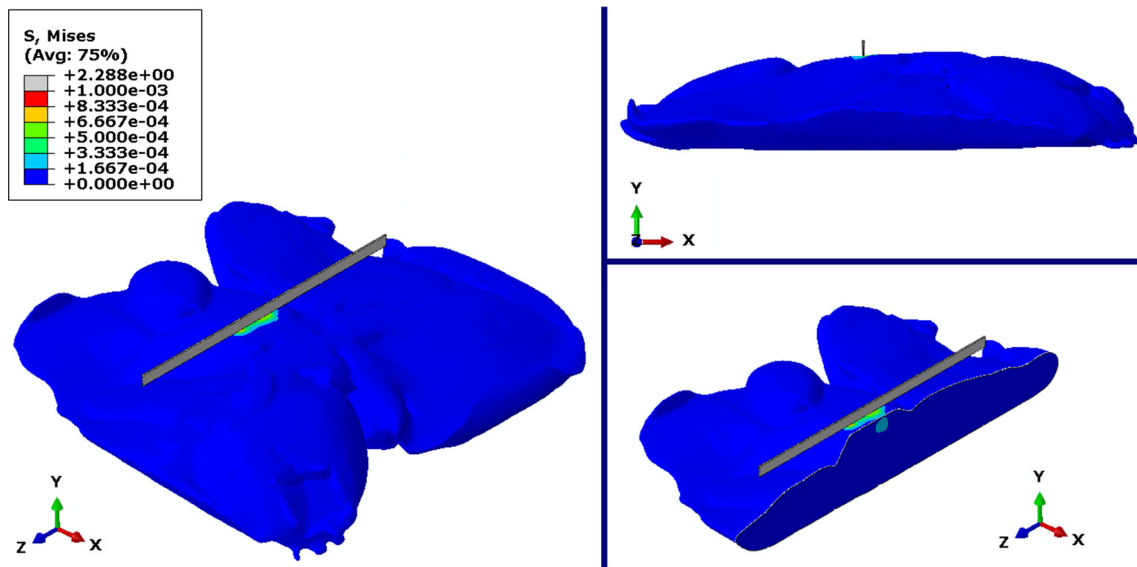


Fig. 5 Liver deformation and Mises stress contours at 1.5 mm vertical displacement due to surgical tool (knife) pressure

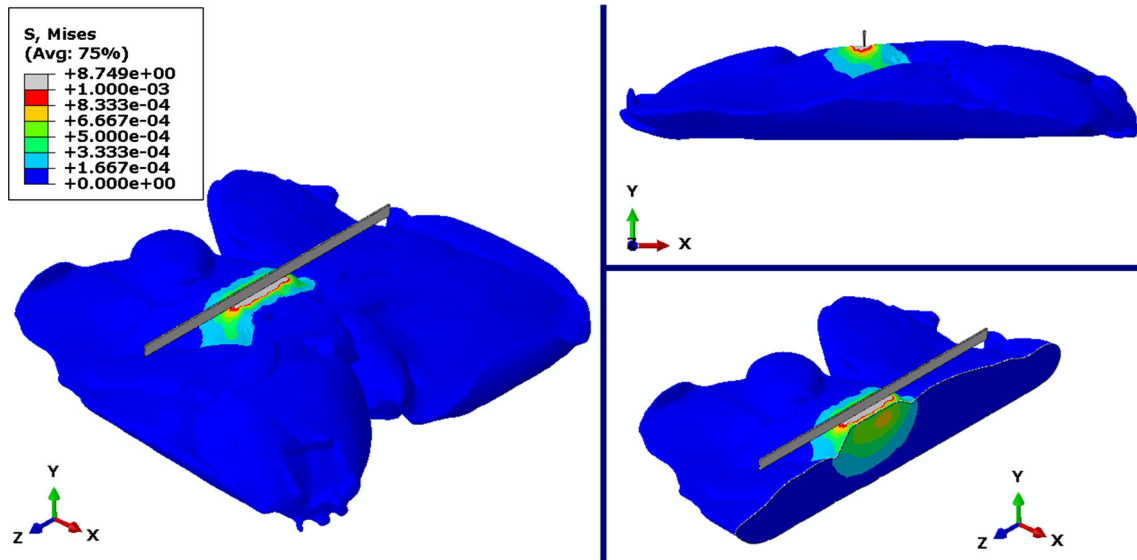


Fig. 6 Liver deformation and Mises stress contours at 2.5 mm vertical displacement due to surgical tool pressure

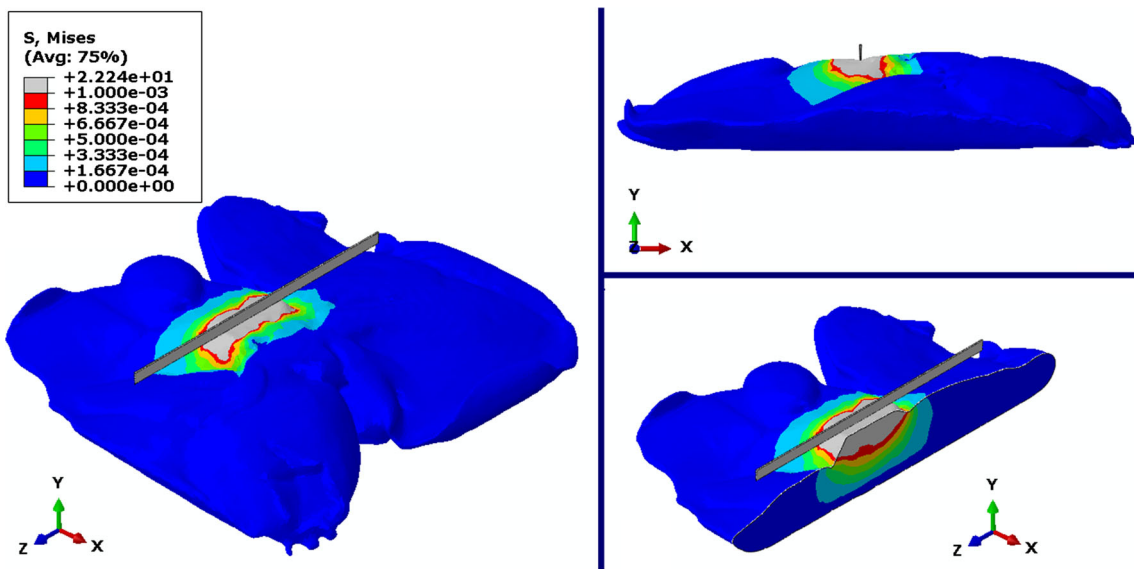


Fig. 7 Liver deformation and Mises stress contours at 3.5 mm vertical displacement due to surgical tool pressure

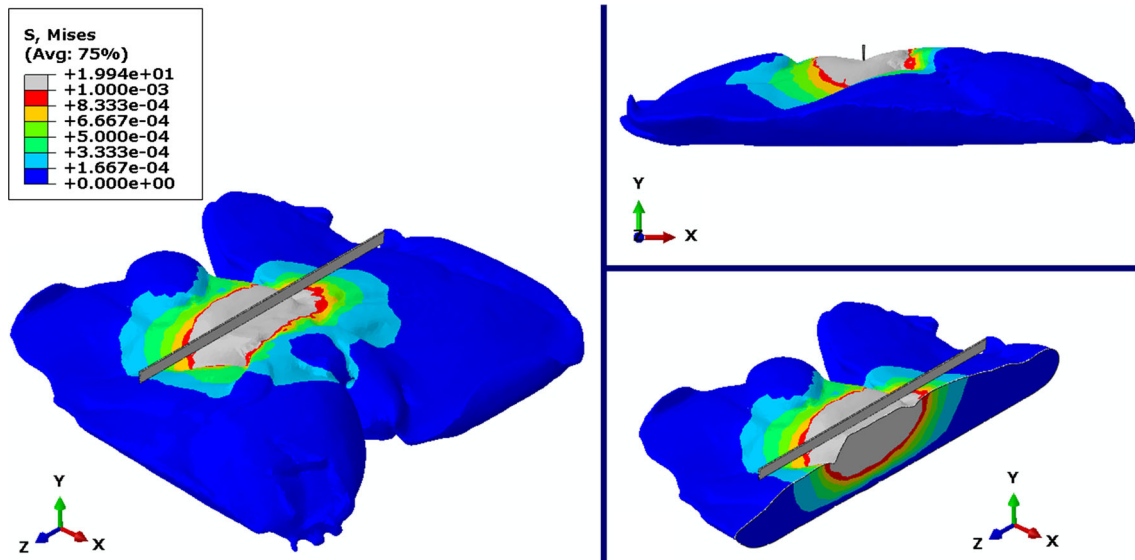


Fig. 8 Liver deformation and Mises stress contours at 5.0 mm vertical displacement due to surgical tool pressure

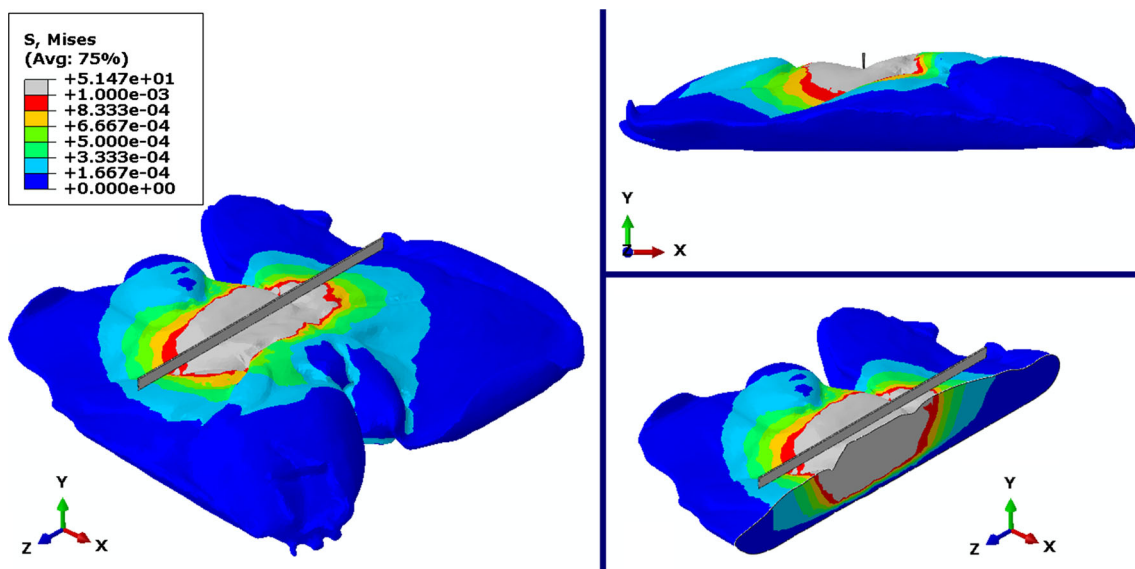


Fig. 9 Liver deformation and Mises stress contours at 6.0 mm vertical displacement due to surgical tool pressure

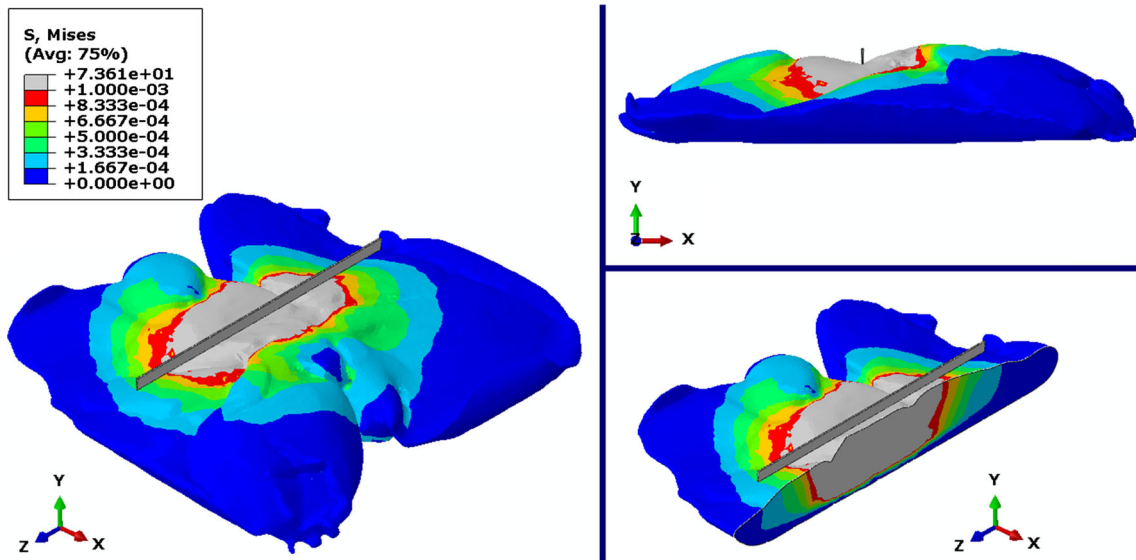


Fig. 10 Liver deformation and Mises stress contours at 7.0 mm vertical displacement due to surgical tool pressure

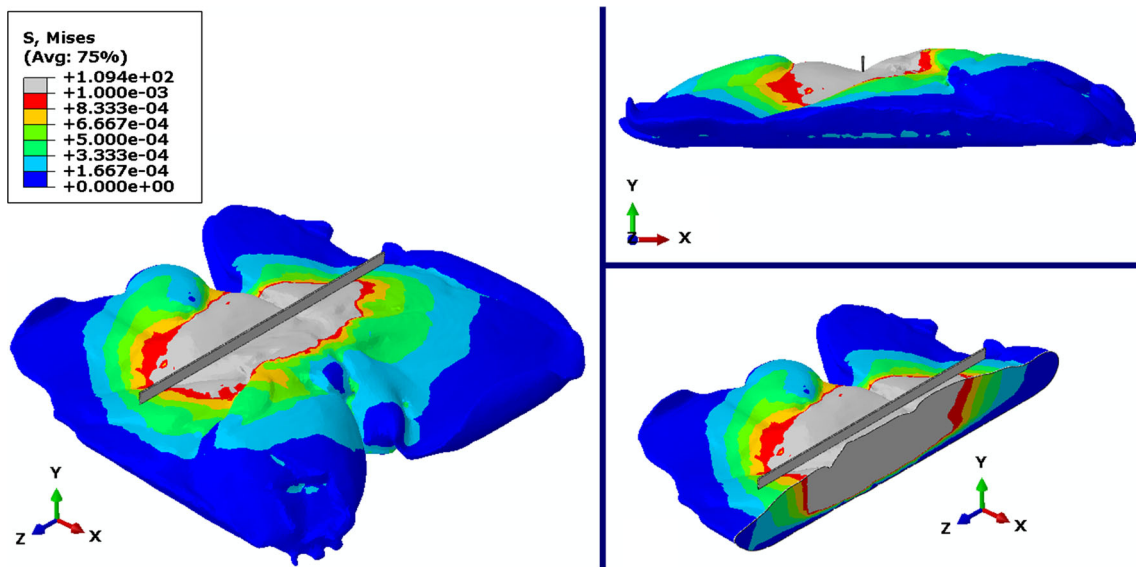


Fig. 11 Liver deformation and Mises stress contours at 9.0 mm vertical displacement due to surgical tool pressure

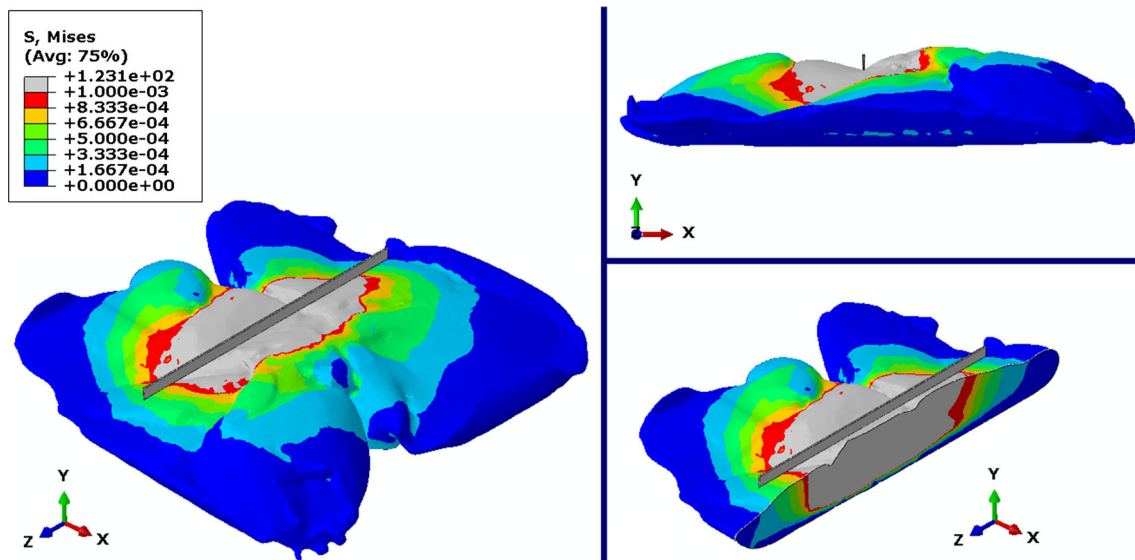


Fig. 12 Liver deformation and Mises stress contours at 10.0 mm vertical displacement due to surgical tool pressure

Liver response under surgical tool (knife) pressure animation video is available as supplementary material.

References

- Kemper AR, Santago AC, Stitzel JD, Sparks JL, Duma SM (2010) Biomechanical response of human liver in tensile loading. *Ann Adv Automot Med* 54:15–26
- Untaroiu CD, Lu Y-C, Kemper AR (2013) Modeling the biomechanical and injury response of human liver parenchyma under tensile loading. In: IRCOBI conference
- Misra S, Okamura MA, Ramesh KT (2007) Force feedback is noticeably different for linear versus nonlinear elastic tissue models. In: 2nd Joint EuroHaptics conference and symposium on haptic interfaces for virtual environments and teleoperator systems, vol 6
- Shi Hongjian, Farag Aly A, Fahmi Rachid, Chen Dongqing (2008) Validation of finite element models of liver tissue using micro-CT. *IEEE Trans Biomed Eng* 55(3):978–984
- Chanthasopsephan T, Desai JP, Lau ACW (2005) 3D and 2D finite element analysis in soft tissue cutting for haptic display. In: 12th International conference on advanced robotics (IEEE Cat. No. 05TH8813), pp 360–367
- Delingette H, Cotin S, Ayache N (1999) A hybrid elastic model allowing real-time cutting, deformations and force-feedback for surgery training and simulation. In: Proceedings computer animation, pp 70–81
- Morten B-N, Stephane C (1996) Real-time volumetric deformable models for surgery simulation using finite elements and condensation. *Comput Graph Forum* 15:57–66
- Joldes GR, Wittek A, Miller K (2009) Non-locking tetrahedral finite element for surgical simulation. *Commun Numer Methods Eng* 25(7):827–836
- Lister K, Gao Z, Desai JP (2009) Real-time, haptics-enabled simulator for probing ex vivo liver tissue. In: 31st Annual international conference of the IEEE engineering in medicine and biology society. EMBC, Minneapolis, pp 1196–1199
- Ahn B, Kim J (2007) An efficient soft tissue characterization method for haptic rendering of soft tissue deformation in medical simulation. In: *Frontiers in the Convergence of Bioscience and Information Technologies (FBIT '07)*, pp 549–553
- Picibono Guillaume, Delingette Herve, Ayache Nicholas (2003) Non-linear anisotropic elasticity for real-time surgery simulation. *Graph Models* 65(5):305–321
- Xunlei Wu, Downes Michael S, Goktekin Tolga, Tendick Frank (2001) Adaptive nonlinear finite elements for deformable body simulation using dynamic progressive meshes. *Comput Graph Forum* 20(3):349–358
- Umale Sagar, Deck Caroline, Bourdet Nicolas, Dhumane Parag, Soler Luc, Marescaux Jacques, Willinger Remy (2013) Experimental mechanical characterization of abdominal organs: liver, kidney & spleen. *J Mech Behav Biomed Mater* 17:22–33
- Gao Zhan, Lister Kevin, Desai Jaydev P (2009) Constitutive modeling of liver tissue: experiment and theory. *Ann Biomed Eng* 38(2):505–516
- Fung YC (1981) *Mechanical properties of living tissues*. Springer-Verlag, New York
- Commercial software Abaqus V6.13 theory guide. <http://www.3ds.com/products-services/simulia/support/documentation/>