Medical Image Analysis 13 (2009) 234–244

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

On modelling of anisotropic viscoelasticity for soft tissue simulation: Numerical solution and GPU execution

Z.A. Taylor [a,*], O. Comas [b], M. Cheng [b], J. Passenger [b], D.J. Hawkes [a], D. Atkinson [a], S. Ourselin [a]

[a] *Centre for Medical Image Computing, University College London, Gower St, London, WC1E 6BT, UK*

[b] *BioMedIA Lab, e-Health Research Centre, CSIRO ICT Centre, Level 20, 300 Adelaide St, Brisbane, QLD 4000, Australia*

ARTICLE INFO

Article history:
Received 31 January 2008
Received in revised form 3 October 2008
Accepted 6 October 2008
Available online 17 October 2008

Keywords:
Surgical simulation
Nonlinear analysis
Finite element method
Viscoelasticity
Anisotropy
Graphics processing units

ABSTRACT

Efficient and accurate techniques for simulation of soft tissue deformation are an increasingly valuable tool in many areas of medical image computing, such as biomechanically-driven image registration and interactive surgical simulation. For reasons of efficiency most analyses are based on simplified linear formulations, and previously almost all have ignored well established features of tissue mechanical response such as anisotropy and time-dependence. We address these latter issues by firstly presenting a generalised anisotropic viscoelastic constitutive framework for soft tissues, particular cases of which have previously been used to model a wide range of tissues. We then develop an efficient solution procedure for the accompanying viscoelastic hereditary integrals which allows use of such models in explicit dynamic finite element algorithms. We show that the procedure allows incorporation of both anisotropy and viscoelasticity for as little as 5.1% additional cost compared with the usual isotropic elastic models. Finally we describe the implementation of a new GPU-based finite element scheme for soft tissue simulation using the CUDA API. Even with the inclusion of more elaborate constitutive models as described the new implementation affords speed improvements compared with our recent graphics API-based implementation, and compared with CPU execution a speed up of 56.3× is achieved. The validity of the viscoelastic solution procedure and performance of the GPU implementation are demonstrated with a series of numerical examples.

1. Introduction

Computational biomechanics has emerged as a topic of considerable interest to the medical image analysis community. Of particular utility are methods for modelling soft tissue deformation. The most physically consistent models are those based on equations of nonlinear continuum mechanics (Holzapfel, 2000), which are generally solved using a numerical technique such as the finite element (FE) method (Bathe, 1996). Specific applications of such models are interactive simulation environments (Cotin et al., 1999; Picinbono et al., 2003; Szekely et al., 2000; Wu and Heng, 2004; Wu and Heng, 2005) and biomechanically driven image registration (Archip et al., 2007; Carter et al., 2005; Clatz et al., 2005; Edwards et al., 1998; Ferrant et al., 2000; Miga et al., 2000; Skrinjar et al., 2002; Warfield et al., 2002; Wittek et al., 2007). Both such applications may be subject to stringent solution time constraints; interactive simulation requires solutions to be obtained at visual, or even haptic feedback rates (>500 Hz), while *intraoperative* image registration must be fast enough that the work flow of the surgical procedure is not interrupted. Therefore viable simulation procedures for these applications are necessarily those that yield rapid solutions.

In response to these constraints, most groups have used simplified linear models which allow much faster solution (Cotin et al., 1999; Wu and Heng, 2004; Wu and Heng, 2005; Archip et al., 2007; Clatz et al., 2005; Edwards et al., 1998; Ferrant et al., 2000; Skrinjar et al., 2002; Warfield et al., 2002). That is, their mathematical formulations assume the tissue stress–strain response is linear, and that the organs undergo infinitesimally small deformations. However the validity of these assumptions for soft tissue simulation is limited since virtually all tissues exhibit a nonlinear constitutive response (Fung, 1993), and most may undergo large deformations. The requirement for rapid *nonlinear* analysis procedures for soft tissues was addressed recently by Miller et al. (2007) who presented a total Lagrangian explicit dynamic (TLED) algorithm for this purpose. Use of a total Lagrangian kinematic framework produces constant shape function spatial derivatives, which may therefore be precomputed, and a simplified treatment of incremental strains. The focus on an explicit dynamic framework was motivated by the fact that such procedures obviate the requirement for assembly of large system matrices, and the use of iterative nonlinear solution schemes such as Newton's method.

* Corresponding author. Tel.: +44 0 20 7679 0209; fax: +44 0 20 7679 0255.
E-mail address: z.taylor@cs.ucl.ac.uk (Z.A. Taylor).

1361-8415/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.media.2008.10.001

Nodal forces are instead computed and compiled in an element-wise fashion. This latter fact also endowed the algorithm with inherent data-parallelism and was the basis for the subsequent graphics processing unit (GPU) implementation of Taylor et al. (2007a, 2008), which allowed real-time solution of systems with up to 10000 degrees of freedom (DOF) (approx. 16000 linear tetrahedral elements), and up to 16.8× speed up compared with a single CPU implementation. An important point is that these implementations incorporated both kinematic and constitutive nonlinearities, thus making them suitable for analysis of soft tissues under finite deformations.

It was emphasised in the papers by Taylor et al. (2007a, 2008) that an attractive feature of explicit analyses is the relative ease with which arbitrarily complicated constitutive models may be incorporated. This arises from the fact that element stresses are computed directly from strains in the course of the procedure. Additionally, there is no requirement for computation of tangent matrices as in implicit or quasi-static procedures, since there is no involvement of Newtonian iterations. While the developments above incorporated a hyperelastic constitutive formulation (neo-Hookean, specifically), thus accommodating nonlinearity of the strain-related tissue stress response, the formulation was intentionally simple nonetheless since the focus was on validation of the computational framework. Two further key features of the response of most biological tissues are time- (and rate-) dependence and anisotropy (Fung, 1993).

Time-dependence manifests itself in many aspects of the mechanical response. Soft tissues under constant load generally exhibit creep, while those under constant deformation exhibit stress relaxation (Fung, 1993). Additionally, most tissues appear stiffer at higher loading velocities (Fung, 1993). Thus viscoelastic constitutive models have been widely employed in analysis of such materials (Miller and Chinzei, 1997; Pioletti and Rakotomanana, 2000; Miller, 2000; Miller and Chinzei, 2002; Limbert and Middleton, 2004; Taylor et al., 2007b; Nava et al., 2008). In particular, so-called visco-hyperelastic models based on strain energy functions with time-dependent parameters have been shown to reproduce both the time-dependent and large strain aspects of the response (Miller and Chinzei, 1997; Miller, 2000; Miller and Chinzei, 2002; Nava et al., 2008).

Anisotropic mechanical response may arise, for example, from the presence of a highly organised microstructure such as those of connective tissues. These are predominantly composed of collagen or elastin fibres embedded in an amorphous matrix (Fung, 1993), and may be considered as fibre reinforced composites in some cases. Alternatively the presence of vasculature and other functional components means even non-load bearing organs may exhibit directional dependence (Prange and Margulies, 2002; Picinbono et al., 2003; Chui et al., 2007). Whereas isotropic constitutive models may be formulated in terms of the usual principal strain invariants, directional dependence requires inclusion of so-called pseudo-invariants of strain *and* material direction (see Section 3). Formulations of this kind allow kinematically consistent analysis of materials under large deformations. They have formed the basis of models (both elastic and viscoelastic) of many biological tissues, including tendons and ligaments (Weiss et al., 1996; Puso and Weiss, 1998; Limbert and Middleton, 2004; Limbert and Middleton, 2006; Vena et al., 2006; Pena et al., 2007, arterial walls Holzapfel et al., 2000; Holzapfel et al., 2002), leg tissues (Bonet and Burton, 1998), cardiac tissue (Humphrey and Yin, 1987; Humphrey et al., 1990), liver (Picinbono et al., 2003; Chui et al., 2007) and cornea (Kaliske, 2000; Pinsky et al., 2005).

In order to model such phenomena with the established explicit FE scheme above without significant performance penalties an efficient constitutive update procedure involving time integration of the relevant hereditary equations is required. We address this problem by presenting a procedure similar to that developed by Poon and Ahmad (1998) for analysis of anisotropic linear viscoelastic models. This was also adapted by Taylor et al. (2007b) for solution of their fibre composite-based microstructural model. In these cases the rate independent responses were based on linear elasticity, rather than a hyperelastic formulation suitable for large deformations. In the present work we begin with a class of anisotropic visco-hyperelastic models, and develop a constitutive update procedure for explicit analyses based on these. We propose that the use of an explicit framework is particularly efficient since there is no requirement for computation of tangent matrices as in implicit or quasi-static procedures (as there is no involvement of Newtonian iterations). Indeed we show that the present formulation allows incorporation of both tissue anisotropy and viscoelasticity at very little additional cost over a comparable isotropic elastic model, and is fully compatible with rapid GPU execution.

Our previous GPU implementation was developed using the OpenGL graphics API (Shreiner et al., 2006) and the Cg programming language (Fernando and Kilgard, 2003). While significant speed improvements were achieved and the suitability of the TLED algorithm for parallel execution was demonstrated, the use of a graphics API led to an inelegant and inflexible code heavy with irrelevant graphics-related functionality. In the present work we used the new CUDA API from NVIDIA Corporation (CUDA, 2007), which was developed to simplify the implementation of general purpose GPU applications. While this obviates the mentioned peripheral code, the structuring of computations remains essentially unchanged from that described in Taylor et al. (2007a, 2008), in particular with regard to the *gathering* of nodal forces. Nonetheless use of a new memory arrangement plus the increased number of processors on CUDA-enabled GPUs lead to further improvements in computational speed, as will be shown.

Hence, the contributions of the paper are twofold: (1) we introduce an efficient constitutive update scheme for anisotropic viscoelastic models for use in explicit analyses, and (2) we present a revised GPU implementation of the TLED FE algorithm, incorporating the developed scheme. The remainder of the paper is structured as follows. A brief overview of the TLED algorithm is presented in Section 2. In Section 3 we present a constitutive framework for modelling anisotropic viscoelastic materials at finite strains, which is used as the basis for the constitutive update scheme described in Section 4. The implementation of the TLED algorithm using CUDA is presented in Section 5, and the performance of this implementation with the developed constitutive framework is assessed in Section 6. The paper concludes with a discussion of the developed scheme.

2. TLED finite element algorithm

A complete description of the TLED algorithm for soft tissue simulation is available in previous publications (Miller et al., 2007; Taylor et al., 2008), and more general details may be obtained from Bathe (1996). Briefly, the algorithm consists of a precomputation phase in which element shape function derivatives $\partial \mathbf{h}$ (and other quantities) and the system mass matrix $\mathbf{M}$ are calculated, followed by a time-loop in which incremental solutions for the node displacements $\mathbf{u}$ are found. During each step of the time-loop we

(1) Apply loads (displacements and/or forces) and boundary conditions to relevant nodal degrees of freedom.
(2) For each element compute
  (a) deformation gradient $\mathbf{F}$
  (b) linear strain–displacement matrix $\mathbf{B}_L$
  (c) second Piola–Kirchhoff stress $\mathbf{S}$

(d) element nodal forces $\tilde{\mathbf{f}}$, and add these forces to the total nodal forces $\mathbf{f}$.

(3) For each node compute new displacements $\mathbf{u}$ using the central difference method.

The central difference method is an explicit method, hence the procedure is stable only for time increments lower than a critical value Δt_{cr}.

Here we focus on the computation of element stresses and nodal forces, with which the main innovation of the present work is concerned, and refer interested readers to the above mentioned publications for other details. The nodal force contributions $\tilde{\mathbf{f}}$ from each element are obtained from

$$\tilde{\mathbf{f}} = \int_V \mathbf{B}_L^T \check{\mathbf{S}} \, \mathrm{d}V, \tag{1}$$

where V is the initial volume of the element, and $\check{\mathbf{S}}$ is the vector form of the stress tensor $\mathbf{S}$. This integral is generally evaluated numerically, for example using Gaussian quadrature. For reduced integration 8-node hexahedral elements we obtain

$$\tilde{\mathbf{f}} = 8\mathbf{B}_L^T \check{\mathbf{S}} \det \mathbf{J}, \tag{2}$$

where $\mathbf{J}$ is the precomputed element Jacobian matrix. For 4-node tetrahedral elements we obtain

$$\tilde{\mathbf{f}} = V\mathbf{B}_L^T \check{\mathbf{S}}. \tag{3}$$

The above equations make no assumption concerning the constitutive model employed. The deformation state $\mathbf{F}$ in each element is known, allowing stresses $\mathbf{S}$ to be computed from any valid constitutive equation.

3. Anisotropic viscoelastic constitutive equations for soft tissues

3.1. Visco-hyperelasticity

In presenting generic constitutive equations for soft tissues we focus on two particular aspects of their mechanical response: (a) their ability to undergo large recoverable deformations, and (b) the time- and rate-dependence of their response. These combined features have led to the formulation of visco-hyperelastic constitutive models (Miller and Chinzei, 1997; Miller and Chinzei, 2002; Nava et al., 2008), in which an underlying hyperelastic formulation is augmented by time dependent (viscoelastic) material parameters. Models of this type are well known in the continuum mechanics community, and provide a kinematically consistent basis for modelling nonlinear materials at large deformations. Particular cases have been shown to be excellent models of the mechanical response of many soft tissues (see references cited in Section 1).

For such materials the constitutive response is defined in terms of a time-dependent Helmholtz free energy (strain energy) function $\hat{\Psi}$, expressed in the form of a convolution integral:

$$\hat{\Psi}(\Psi, t) = \int_0^t \alpha(t-s) \frac{\partial \Psi}{\partial s} \mathrm{d}s, \tag{4}$$

where t is time and Ψ is the underlying hyperelastic strain energy function. The relaxation functions $\alpha(t)$ commonly assume the form of a Prony series:

$$\alpha(t) = \alpha_\infty + \sum_{i=1}^{N} \alpha_i \mathrm{e}^{-t/\tau_i}, \tag{5}$$

where α_∞, α_i, and τ_i are positive real constants. Such forms for the relaxation functions have a physical interpretation, namely that of a generalised Maxwell model (Holzapfel, 1996). If we impose the condition

$$\left(\alpha_\infty + \sum_{i=1}^{N} \alpha_i\right) = 1 \tag{6}$$

we may rewrite Eq. (5) as

$$\alpha(t) = 1 - \sum_{i=1}^{N} \alpha_i (1 - \mathrm{e}^{-t/\tau_i}), \tag{7}$$

which will be of use in subsequent sections.

The required stress $\mathbf{S}$ may be obtained via differentiation with respect to strain:

$$\begin{aligned} \mathbf{S} = 2\frac{\partial \hat{\Psi}(\Psi, t)}{\partial \mathbf{C}} &= \int_0^t \alpha(t-s) \left(2\frac{\partial}{\partial s} \frac{\partial \Psi}{\partial \mathbf{C}}\right) \mathrm{d}s \\ &= \int_0^t \alpha(t-s) \frac{\partial \mathbf{\Phi}}{\partial s} \mathrm{d}s, \end{aligned} \tag{8}$$

where $\mathbf{C} = \mathbf{F}^T\mathbf{F}$ is the right Cauchy–Green deformation tensor, and we have introduced $\mathbf{\Phi} := 2\partial\Psi/\partial\mathbf{C}$ as the instantaneous hyperelastic stress response.

Models of this form have been presented by Miller and coworkers for analysis of brain tissue (Miller and Chinzei, 1997, 2002) and of liver and kidney (Miller, 2000). They were shown to model the tissue responses to compressive (and in the case of brain, tensile) loading at strain rates varying over two orders of magnitude very well. We next consider evaluation of the hyperelastic stress response $\mathbf{\Phi}$.

3.2. Hyperelastic response

We firstly consider the case of isotropic materials, from which anisotropic formulations follow. We then consider the standard cases of transverse isotropy and orthotropy, which may be viewed as arising from, for example, the presence of mutually orthogonal reinforcing fibre phases. In considering these cases we encompass constitutive equations which have been proposed for a wide variety of biological tissues.

3.2.1. Isotropic materials

For isotropic materials the strain energy is a function of strain only, hence $\Psi = \Psi(\mathbf{C})$. In this work we consider strain energy functions with separated isochoric (volume preserving) and volumetric components (Holzapfel, 2000):

$$\Psi(\mathbf{C}) = \Psi^{\mathrm{iso}}(\bar{\mathbf{C}}) + \Psi^{\mathrm{vol}}(J) = \Psi^{\mathrm{iso}}(\bar{I}_1, \bar{I}_2) + \Psi^{\mathrm{vol}}(J), \tag{9}$$

where $J := \det\mathbf{F}$ is the Jacobian determinant, $\bar{\mathbf{C}} = J^{-2/3}\mathbf{C}$ is the modified right Cauchy–Green deformation, and $\bar{I}_1 = \mathrm{tr}\bar{\mathbf{C}}$ and $\bar{I}_2 = [(\mathrm{tr}\bar{\mathbf{C}})^2 - \mathrm{tr}(\bar{\mathbf{C}}^2)]/2$ are invariants of $\bar{\mathbf{C}}$. The hyperelastic stress $\mathbf{\Phi}$ then also consists of isochoric and volumetric components:

$$\mathbf{\Phi} = \mathbf{\Phi}^{\mathrm{iso}} + \mathbf{\Phi}^{\mathrm{vol}}, \tag{10}$$

where

$$\mathbf{\Phi}^{\mathrm{vol}} = 2\frac{\partial \Psi^{\mathrm{vol}}(J)}{\partial \mathbf{C}} = J\frac{\mathrm{d}\Psi^{\mathrm{vol}}(J)}{\mathrm{d}J}\mathbf{C}^{-1}, \tag{11}$$

and

$$\mathbf{\Phi}^{\mathrm{iso}} = 2\frac{\partial \Psi^{\mathrm{iso}}(\bar{\mathbf{C}})}{\partial \mathbf{C}} = J^{-2/3}\mathrm{Dev}\bar{\mathbf{\Phi}}, \tag{12}$$

where $\mathrm{Dev}(\bullet) = (\bullet) - (1/3)[(\bullet){:}\mathbf{C}]\mathbf{C}^{-1}$ is the referential configuration deviatoric operator for a second order tensor (Holzapfel, 2000), and

$$\bar{\mathbf{\Phi}} = 2\frac{\partial \Psi^{\mathrm{iso}}(\bar{\mathbf{C}})}{\partial \bar{\mathbf{C}}} = \bar{\gamma}_1\mathbf{I} + \bar{\gamma}_2\bar{\mathbf{C}}, \tag{13}$$

with

$$\bar{\gamma}_1 = 2\left(\frac{\partial \Psi^{\mathrm{iso}}}{\partial \bar{I}_1} + \bar{I}_1 \frac{\partial \Psi^{\mathrm{iso}}}{\partial \bar{I}_2}\right), \quad \bar{\gamma}_2 = -2\frac{\partial \Psi^{\mathrm{iso}}}{\partial \bar{I}_2}. \tag{14}$$

3.2.2. Transversely isotropic materials

Transversely isotropic (TI) materials are characterised by a single preferred direction $\mathbf{a_0}$ in the reference configuration; the mechanical response is isotropic in the plane orthogonal to this direction. The strain energy is then a function of both $\mathbf{C}$ and a structure tensor $\mathbf{A_0} := \mathbf{a_0} \otimes \mathbf{a_0}$, where $\otimes$ denotes a tensor product. Analogous to the isotropic case, we consider strain energy functions of the form

$$\Psi(\mathbf{C}, \mathbf{A_0}) = \Psi^{\mathrm{iso}}(\bar{I}_1, \bar{I}_2, \bar{I}_4, \bar{I}_5) + \Psi^{\mathrm{vol}}(J), \tag{15}$$

where $\bar{I}_4 = \mathbf{a_0} \cdot \bar{\mathbf{C}}\mathbf{a_0}$ and $\bar{I}_5 = \mathbf{a_0} \cdot \bar{\mathbf{C}}^2\mathbf{a_0}$ are pseudo-invariants of $\bar{\mathbf{C}}$ and $\mathbf{A_0}$.

As can be seen TI strain energy functions Eq. (15) differ from isotropic ones Eq. (9) only in the form of the isochoric term. Therefore the volumetric and isochoric stresses $\mathbf{\Phi}^{\mathrm{vol}}$ and $\mathbf{\Phi}^{\mathrm{iso}}$ remain as in Eqs. (11) and (12), but with $\bar{\mathbf{\Phi}}$ given by

$$\bar{\mathbf{\Phi}} = \bar{\gamma}_1\mathbf{I} + \bar{\gamma}_2\bar{\mathbf{C}} + \bar{\gamma}_4\mathbf{A_0} + \bar{\gamma}_5(\mathbf{a_0} \otimes \bar{\mathbf{C}}\mathbf{a_0} + \bar{\mathbf{C}}\mathbf{a_0} \otimes \mathbf{a_0}), \tag{16}$$

with

$$\bar{\gamma}_a = 2\frac{\partial \Psi^{\mathrm{iso}}}{\partial \bar{I}_a}, \quad a = 4, 5. \tag{17}$$

3.2.3. Orthotropic materials

Orthotropic materials are characterised by three mutually orthogonal preferred directions, which we identify with unit vectors $\mathbf{a_0}$ and $\mathbf{b_0}$ (and corresponding structure tensors $\mathbf{A_0}$ and $\mathbf{B_0}$), in the reference configuration. We need specify only two vectors, since the direction orthogonal to these naturally emerges as a preferred direction also. Orthotropic strain energy functions are then of the form

$$\Psi(\mathbf{C}, \mathbf{A_0}, \mathbf{B_0}) = \Psi^{\mathrm{iso}}(\bar{I}_1, \bar{I}_2, \bar{I}_4, \bar{I}_5, \bar{I}_6, \bar{I}_7) + \Psi^{\mathrm{vol}}(J), \tag{18}$$

where $\bar{I}_6 = \mathbf{b_0} \cdot \bar{\mathbf{C}}\mathbf{b_0}$ and $\bar{I}_7 = \mathbf{b_0} \cdot \bar{\mathbf{C}}^2\mathbf{b_0}$ are pseudo-invariants of $\bar{\mathbf{C}}$ and $\mathbf{B_0}$. In a similar manner to the TI case we obtain $\mathbf{\Phi}^{\mathrm{vol}}$ and $\mathbf{\Phi}^{\mathrm{iso}}$ from Eqs. (11) and (12), with $\bar{\mathbf{\Phi}}$ now given by

$$\begin{aligned}\bar{\mathbf{\Phi}} = {} & \bar{\gamma}_1\mathbf{I} + \bar{\gamma}_2\bar{\mathbf{C}} + \bar{\gamma}_4\mathbf{A_0} + \bar{\gamma}_5(\mathbf{a_0} \otimes \bar{\mathbf{C}}\mathbf{a_0} + \mathbf{a_0}\bar{\mathbf{C}} \otimes \mathbf{a_0}) + \bar{\gamma}_6\mathbf{B_0} \\ & + \bar{\gamma}_7(\mathbf{b_0} \otimes \bar{\mathbf{C}}\mathbf{b_0} + \mathbf{b_0}\bar{\mathbf{C}} \otimes \mathbf{b_0}),\end{aligned} \tag{19}$$

where

$$\bar{\gamma}_a = 2\frac{\partial \Psi^{\mathrm{iso}}}{\partial \bar{I}_a}, \quad a = 6, 7. \tag{20}$$

3.3. Recapitulation

For a visco-hyperelastic material stress may be obtained from Eq. (8). This form is general in the sense that any underlying hyperelastic response may be used, including the anisotropic formulations described. Eq. (10), supplemented with Eqs. (11)–(13) is the general form of an isotropic hyperelastic stress response, defined in terms of invariants. TI or orthotropic models may be produced by substituting Eq. (13) for Eqs. (16) or (19), respectively. The specification of particular forms of Ψ^{vol} and Ψ^{iso} would be motivated by the particular tissue/material under analysis, and may stem from phenomenological or microstructural considerations. Many examples are available in the references cited in Section 1. Finally we note that for separated isochoric and volumetric hyperelastic functions as used here, viscoelastic terms may be applied to either or both independently.

4. Constitutive update procedure for explicit analyses

Use of the above visco-hyperelastic models within the TLED algorithm (or any other explicit dynamic FE procedure) requires a constitutive update scheme involving time integration of Eq. (8).

4.1. Stress update equations

We proceed by restating Eq. (8) and using Eq. (7):

$$\mathbf{S} = \int_0^t \left[1 - \sum_{i=1}^{N} \alpha_i\left(1 - \mathrm{e}^{(s-t)/\tau_i}\right)\right] \frac{\partial \mathbf{\Phi}}{\partial s}\mathrm{d}s. \tag{21}$$

This may be separated into rate-dependent and -independent terms as

$$\mathbf{S} = \mathbf{\Phi} - \sum_{i=1}^{N} \mathbf{\Upsilon}_i, \tag{22}$$

where

$$\mathbf{\Upsilon}_i = \int_0^t \alpha_i\left(1 - \mathrm{e}^{(s-t)/\tau_i}\right) \frac{\partial \mathbf{\Phi}}{\partial s}\mathrm{d}s, \quad i \in [1, N] \tag{23}$$

are rate-dependent terms associated with each term in the Prony series.

In an incremental analysis we require the stress at the current increment given the deformation state and history of the material. Adding superscripts to indicate time increments the stress may be updated using

$$\mathbf{S}^n = \mathbf{\Phi}^n - \sum_{i=1}^{N} \mathbf{\Upsilon}_i^n. \tag{24}$$

The instantaneous terms $\mathbf{\Phi}^n$ may be computed directly from the (known) current deformation $\mathbf{C}^n$. The main difficulty is then computation of the incremental rate-dependent terms $\mathbf{\Upsilon}_i^n$. Following Poon and Ahmad (1998) our strategy is to maintain each $\mathbf{\Upsilon}_i$ as a separate state variable to be updated at each increment also.

4.2. State variable update equations

Our approach is to convert the integral Eq. (23) into a rate form which may then be numerically integrated to produce an incremental update formula for $\mathbf{\Upsilon}_i$. We note that Eq. (23) is of the form

$$y(t) = \int_0^t f(t, s)\mathrm{d}s. \tag{25}$$

Poon and Ahmad (1998) provide the following formula for differentiating such equations with respect to t:

$$\dot{y} = f(t, t) + \int_0^t \dot{f}(t, s)\mathrm{d}s. \tag{26}$$

Applying this formula to Eq. (23) we obtain

$$f(t, t) = \alpha_i(1 - \mathrm{e}^{(t-t)/\tau_i})\frac{\partial \mathbf{\Phi}}{\partial s} = 0 \tag{27}$$

and

$$\dot{f}(t, s) = \frac{\mathrm{d}}{\mathrm{d}t}\left[\alpha_i(1 - \mathrm{e}^{(s-t)/\tau_i})\frac{\partial \mathbf{\Phi}}{\partial s}\right] = \alpha_i\mathrm{e}^{(s-t)/\tau_i}\frac{\partial \mathbf{\Phi}}{\partial s}\frac{1}{\tau_i}, \tag{28}$$

leading to the required rate form for $\mathbf{\Upsilon}_i$:

$$\dot{\mathbf{\Upsilon}}_i = \frac{1}{\tau_i}\int_0^t \alpha_i\mathrm{e}^{(s-t)/\tau_i}\frac{\partial \mathbf{\Phi}}{\partial s}\mathrm{d}s = \frac{1}{\tau_i}(\alpha_i\mathbf{\Phi} - \mathbf{\Upsilon}_i). \tag{29}$$

Eq. (29) may be integrated using a convenient numerical method. In particular the unconditionally stable backward Euler method suggested by Poon and Ahmad (1998) yields the following formula for $\boldsymbol{\Upsilon}_i^n$

$$\boldsymbol{\Upsilon}_i^n = \left(\frac{\alpha_i \boldsymbol{\Phi}^n}{\tau_i} + \frac{\boldsymbol{\Upsilon}_i^{n-1}}{\Delta t}\right) \bigg/ \left(\frac{1}{\Delta t} + \frac{1}{\tau_i}\right) = A\boldsymbol{\Phi}^n + B\boldsymbol{\Upsilon}_i^{n-1}, \tag{30}$$

where Δt is the time step size, and $A = \Delta t \alpha_i / (\Delta t + \tau_i)$ and $B = \tau_i / (\Delta t + \tau_i)$ are constant coefficients.

Since as mentioned there is no requirement for computation of tangent matrices in explicit analyses, this concludes the constitutive update procedure.

4.3. Summary

The constitutive update procedure consists of

(1) Updating state variables (one for each Prony term) via Eq. (30).
(2) Updating stresses via Eq. (24).

Assuming a constant time step size Δt, the coefficients A and B may be precomputed. When used with, for example, reduced integration hexahedra (with a single integration point per element) or linear tetrahedra the present scheme introduces only $12N$ (N = total number of Prony terms) extra multiplications per element per time step compared with an equivalent hyperelastic formulation. Additional storage requirements are also minimal since only state variables (6-vectors) are retained between increments.

5. GPU execution

Having established a comprehensive constitutive framework for soft tissues and an efficient computational scheme for its use in explicit dynamic FE procedures, we now describe the implementation of such a procedure for GPU execution. The solver consists of a host (CPU) and a device (GPU) component. The host component comprises C++ classes (Stroustrup, 1997) for performing precomputations and managing simulation parameters, and a suite of directly callable CUDA functions. These latter facilitate allocation of device memory, binding of textures, data exchange between host and device, and launch of computational kernels on the device (the device component) during each step of the time-loop. CUDA version 1.1 was used.

5.1. Kernel arrangement

Time-loop computations were implemented as two kernels, K_e and K_n, corresponding to element and node calculations, respectively (see Section 2). A simulation time-step was executed by launching K_e across N_e threads and subsequently K_n across N_n threads, where N_e and N_n are the number of elements and nodes in the system, respectively. The major task of K_e was computation of element stresses based on the current model configuration, and conversion of these into equivalent nodal force contributions which were then written to memory. Within K_n nodal forces were summed on each node and used to compute new nodal displacements, thus completing the time step. The arrangement of force arrays in memory required to perform the retrieval and summation in K_n is described in detail in Taylor et al. (2008). The use of the developed viscoelastic constitutive update scheme necessitates storage of an additional array of state variables $\boldsymbol{\Upsilon}_i$.

The kernel arrangement described here is essentially the same as that used in our earlier OpenGL/Cg-based implementation (Taylor et al., 2007a; Taylor et al., 2008). The major difference between this parallel configuration and a serial one is the writing of element nodal forces to memory and their subsequent retrieval and summation during the second kernel. In a serial implementation nodal force contributions would most likely be added to a global sum as they are computed, rather than stored and summed in a second loop. As was the case in our earlier implementation, this is not feasible on the present GPU architecture due to the impracticality of scattered writes. Although CUDA allows scattered writes in theory, it offers no write conflict management between threads, and potential measures to address this have proved extremely detrimental to performance.

5.2. Memory structure

All solution variables were stored in linear global memory (CUDA, 2007), and accessed via texture fetching. For the present application in which access patterns are essentially random (since unstructured meshes are used in general) our experiments showed this method to be fastest.

A key feature of the CUDA API is the exposure of a small pool of fast memory accessible by all threads within a given block – so-called shared memory. CUDA developers suggest (CUDA, 2007) that this could be useful in reducing the number of expensive texture fetches or global memory reads. For example a data element used by several threads could be fetched by one thread and written to shared memory from where it could be accessed very efficiently by the other threads. Examples are included in the CUDA programming guide (CUDA, 2007). However, as noted in our previous publication (Taylor et al., 2008) such a mechanism is only useful in cases where neighbouring threads use common data, but the present application employs unstructured meshes in which this is generally not the case. Therefore we make no use of this resource in the present implementation.

5.3. Host–device interaction

Host–device interaction is generally cited as a significant bottleneck in general purpose GPU applications due to the relatively low interface bandwidth (CUDA, 2007), and therefore it is desirable to minimise such interaction, as discussed previously Taylor et al., 2008. CUDA alleviates the problem somewhat by allowing allocation of areas of page-locked host memory which are directly accessible by the device, and therefore offer much higher bandwidth (CUDA, 2007). All transfers between host and device are made via this mechanism.

5.4. Element technology

We used both reduced integration 8-node hexahedral and 4-node (linear) tetrahedral elements. Hexahedra are preferable both in terms of solution accuracy and computational efficiency. Unlike linear tetrahedra they are not susceptible to volumetric locking (Hughes, 1987), and they allow construction of meshes with far fewer elements for a given number of degrees of freedom. Since the majority of the calculations in explicit dynamic analyses are performed per element, this results in reduced overall computation time.

However a decided disadvantage of hexahedra is the difficulty in automatically generating hexahedral meshes over arbitrary domains. Construction of hexahedral meshes is time consuming and laborious, which fact is of even greater significance when patient-specific simulations are considered. For this reason tetrahedral meshes (which are easily generated) are widely used in simulations despite their shortcomings (Cotin et al., 1999; Picinbono et al., 2003; Clatz et al., 2005; Ferrant et al., 2000). A further disadvantage of (reduced integration) hexahedra is the existence of spurious zero-energy modes (so-called hourglass modes) which unless addressed cause rapid deterioration of the solution (Flanagan and

Belytschko, 1981). Measures are available for suppressing these modes (Flanagan and Belytschko, 1981; Hutter et al., 2000; Joldes et al., 2008), but inevitably they detract somewhat from the computational efficiency of the elements.

Compared with linear tetrahedral elements, hexahedra impose a substantially heavier burden on GPU memory resources. Element matrices such as shape function derivatives and nodal displacements are twice as large for hexahedra, which necessitates twice as many texture fetches per element, and use of twice as many registers per thread. Similarly, twice as many nodal forces per element are written to global memory. Additional variables associated with hourglass control are required also. Therefore, on a per element basis hexahedra are significantly less efficient than tetrahedra – especially for GPU execution, where memory efficiency is paramount. However, from the point of view of an entire model the lower number of hexahedra required for a given number of degrees of freedom still outweighs this element-wise inefficiency.

In the present work we demonstrate the efficiency of the hexahedral formulation using geometrically simple models for which meshing is straight-forward. We then use a tetrahedral mesh for simulation of an organ model derived from segmented image data. It should be noted that the developed constitutive update procedure is independent of the element formulation.

5.5. SOFA

The new GPU-based solver and constitutive update procedure have been further implemented within the SOFA open source framework (Allard et al., 2007)[1], and will be included in a future release. The process is described in Comas et al. (2008).

6. Numerical examples

We present a series of examples based on pure shear and compression of cube models to demonstrate the validity of the constitutive update procedure and the performance of the GPU-based implementation. We conclude with an example of simulation of liver deformation, since as mentioned there is experimental evidence that liver exhibits an anisotropic response (Chui et al., 2007). In each case we used a TI visco-hyperelastic model with elastic strain energy components defined by

$$\Psi^{\text{iso}} = \Psi^{\text{I}}(\bar{I}_1) + \Psi^{\text{TI}}(\bar{I}_4) = \frac{\mu}{2}(\bar{I}_1 - 3) + \frac{\eta}{2}(\bar{I}_4 - 1)^2, \tag{31}$$

$$\Psi^{\text{vol}} = \frac{\kappa}{2}(J - 1)^2, \tag{32}$$

where Ψ^{I} and Ψ^{TI} are isotropic and TI components, respectively, μ is the small strain shear modulus, κ is the bulk modulus, and η is a material parameter with units of Pa. The isotropic component Ψ^{I} represents the well known neo-Hookean model, while the anisotropic component Ψ^{TI} is the same form used by Picinbono et al. (2003). Here the dependence of Ψ^{TI} on $\bar{I}_5$ is omitted and we include only $\bar{I}_4$ terms. This was done foremost for simplicity, but is also a common practical feature of anisotropic models for soft tissues – e.g. see Puso and Weiss (1998), Limbert and Middleton (2004, 2006), Vena et al. (2006), Pena et al. (2007), Holzapfel et al. (2002), Kaliske (2000), Picinbono et al. (2003). The main reason is the clear physical interpretation of $\bar{I}_4$ as the square of the stretch λ along the preferred direction $\mathbf{a_0}$ (Holzapfel, 2000)[2], whereas $\bar{I}_5$ has a less clear physical meaning. $\bar{I}_4$ terms are therefore more amenable to experimental investigation, and of more use in modelling structural elements (e.g. fibres) responsible for the anisotropy. It must be emphasised that this model is used as an example only, and we make no claim concerning its appropriateness for any particular tissue.

The instantaneous isochoric and volumetric responses are then given by

$$\mathbf{\Phi}^{\text{iso}} = J^{-2/3}\left[\mu\left(\mathbf{I} + \frac{I_1}{3}\mathbf{C}^{-1}\right) + \eta(\bar{I}_4 - 1)\left(\mathbf{A_0} + \frac{I_4}{3}\mathbf{C}^{-1}\right)\right], \tag{33}$$

$$\mathbf{\Phi}^{\text{vol}} = \kappa J(J - 1)\mathbf{C}^{-1}. \tag{34}$$

We used viscoelastic isochoric terms only, with a single Prony series term for simplicity. The complete stress response is then

$$\begin{aligned}\mathbf{S} &= \mathbf{\Phi}^{\text{vol}} + \int_0^t \left[1 - \alpha_1(1 - e^{(s-t)/\tau_1})\right]\frac{\partial \mathbf{\Phi}^{\text{iso}}}{\partial \tau}\,ds \\ &= \mathbf{\Phi}^{\text{vol}} + \mathbf{\Phi}^{\text{iso}}(1 - \alpha_1) + \alpha_1 \int_0^t e^{(s-t)/\tau_1}\frac{\partial \mathbf{\Phi}^{\text{iso}}}{\partial \tau}\,ds.\end{aligned} \tag{35}$$

Note that in this case the developed constitutive update procedure need only be employed for isochoric terms.

Material parameters were chosen based on recent results for the viscoelastic response of human liver in vivo. Nava et al. (2008) used an aspiration device to obtain an estimate of $\mu = 19700$ Pa for the combined liver/liver capsule.[3] However, they noted that an earlier study suggested the presence of the liver capsule meant this value could overestimate liver *parenchyma* stiffness by up to three times. Since it is the parenchyma we are concerned with, we employed a value of $\mu = 19700/3 = 6567$ Pa. Assuming near incompressibility we specified $\kappa = 326210$ Pa (corresponding to a Poisson ratio of 0.49). Nava et al. employed two Prony terms, and we use parameters from the first of these: $\alpha_1 = 0.5$, $\tau_1 = 0.58$. The short time constant τ_1 means this Prony term most affects faster loadings, which are of interest in surgical simulations. Finally, since Nava et al. assumed an isotropic response their data provide no basis for estimating η. Therefore we chose a conservative value of $\eta = 2\mu$ – conservative since Picinbono et al. (2003) used a value closer to $\eta = 16\mu$.

6.1. Pure shear of a cube

The first example involved shearing of a unit cube model. We imposed a ramped loading such that the displacements u of the loaded nodes at time t were given by $u = rt$, where r is the loading speed, and we chose the x-direction to be the preferred material direction ($\mathbf{a_0} = [1\,0\,0]$). Since the deformation is homogeneous only a single element was required to reproduce the stresses exactly. Moreover an analytical solution is available (see Appendix A) with which numerical results may be compared. Four independent non-zero stress terms are produced.

In Fig. 1 we demonstrate the effects of material anisotropy independent of viscoelastic effects by plotting the limiting instantaneous (hyperelastic) stress curves, produced by setting the relaxation parameter $\alpha_1 = 0$. Fig. 1a shows curves for the anisotropic model, while Fig. 1b shows curves for an equivalent isotropic model (i.e. with $\eta = 0$).

Firstly we confirm that the FE solutions (data marks) match the analytical solutions (solid curves) very accurately. Secondly we observe that the anisotropic term significantly affects both the magnitude and shape of the stress curves. In particular the S_{11} curve which is coincident with the S_{33} curve in the isotropic model actually changes sign when anisotropy is introduced. The other curves exhibit a more pronounced stiffening for the anisotropic case.

[1] URL: http://www.sofa-framework.org.

[2] Strictly, $\bar{I}_4 = J^{-2/3}\lambda^2$.

[3] Hyperelastic coefficients of $C_{10} = 9.85$ kPa were reported, from which $\mu = 2C_{10}$ may be obtained (Holzapfel, 2000). In this case μ is the small strain modulus, which is appropriate for use with the neo-Hookean type model employed presently.

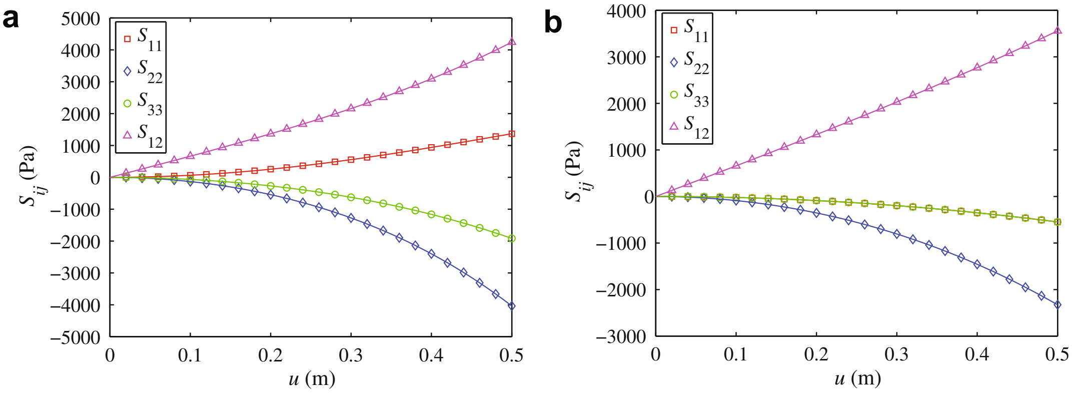

Fig. 1. Instantaneous stress curves for pure shear deformation of (a) the anisotropic model compared with those of (b) an isotropic model. Solid lines correspond to the analytical solution given in Appendix A, while markers indicate the FE solution.

In Fig. 2 we demonstrate the effect of loading rate on the complete viscoelastic model by plotting the S_{11} component for strain rates of 2.5 s^{-1}, 0.25 s^{-1}, and 0.025 s^{-1}, plus the bounding instantaneous and equilibrium responses. The strain rate-dependence introduced by the model (and commonly observed in biological tissues) is clearly shown. Additionally the close match between the analytic and numerical solutions demonstrates the validity of the developed constitutive update scheme.

6.2. Compression of a cube

The second example involved compression of a cube of edge length 0.1 m (comparable to some human organs). The cube was imagined to be fixed to opposing load platens and free on the remaining four faces. Compression was applied along the x-axis. A stress-relaxation type test protocol was simulated, in which the cube was compressed by 30% in 0.5 s and the compression was held for a further 4.5 s. We demonstrate the effects of anisotropy on the deformation response by comparing the deformed shapes of anisotropic models with $\mathbf{a_0} = [0\,1\,0] \equiv \mathbf{a}_0^{\mathbf{y}}$ and $\mathbf{a_0} = [0\ \frac{1}{\sqrt{2}}\ \frac{1}{\sqrt{2}}] \equiv \mathbf{a}_0^{\mathbf{yz}}$ with that of an isotropic model (with $\eta = 0$).

Fig. 3a shows the undeformed cube, while Fig. 3b shows the deformed isotropic model. With no preferred direction the lateral expansion was uniform. In Fig. 3c the increased y-direction stiffness of the first anisotropic model lead to a much reduced expansion in this direction (vertical in the image), and an accompanying increase in the orthogonal z-direction. For the second anisotropic

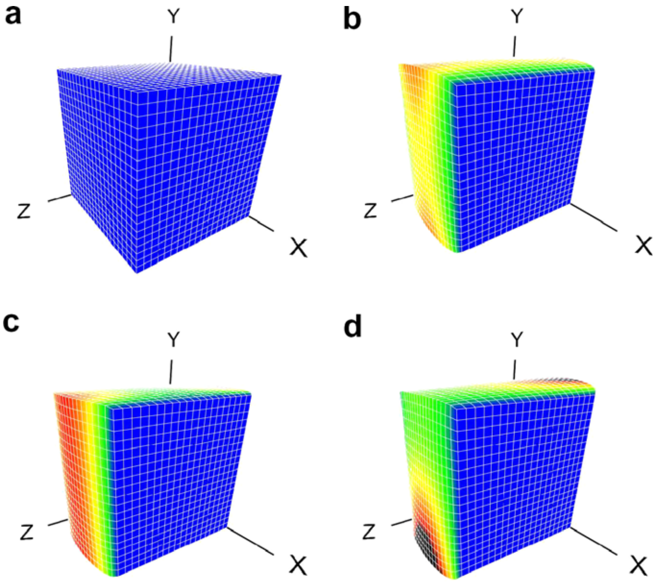

Fig. 3. Deformation patterns of transversely isotropic models compared with that of an isotropic model: (a) the undeformed cube, (b) the deformed isotropic model, (c) the deformed anisotropic model with $\mathbf{a_0} = \mathbf{a}_0^{\mathbf{y}}$, (d) the deformed anisotropic model with $\mathbf{a_0} = \mathbf{a}_0^{\mathbf{yz}}$. Colour maps indicate relative magnitude of lateral displacement ($(u_y^2 + u_z^2)^{1/2}$). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

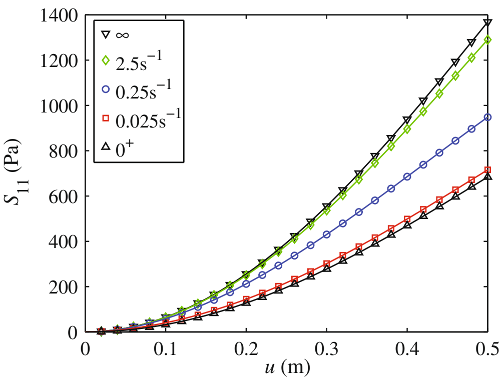

Fig. 2. S_{11} curves for pure shear deformation of the anisotropic viscoelastic model at varying strain rates. Curves for strain rates of 2.5 s^{-1}, 0.25 s^{-1}, and 0.025 s^{-1} are given (as labelled), along with the bounding instantaneous and equilibrium responses. The latter two are labelled ∞ and 0^+, respectively, indicating that they correspond to strain rates approaching these values. Solid lines correspond to the analytical solution given in Appendix A, while markers indicate the FE solution.

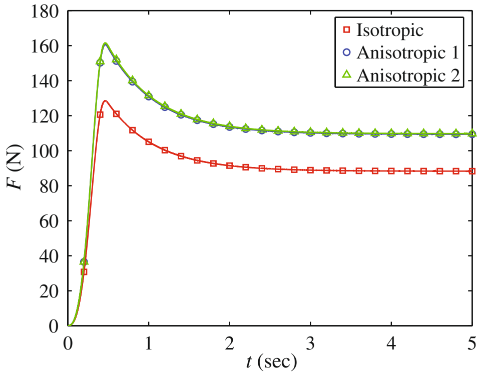

Fig. 4. FE computed reaction forces on the end faces of the cube models over time. Curves are labelled according to the constitutive model used, where *Anisotropic 1* refers to the TI viscoelastic model with $\mathbf{a_0} = \mathbf{a}_0^{\mathbf{y}}$ and it Anisotropic 2 refers to the case of $\mathbf{a_0} = \mathbf{a}_0^{\mathbf{yz}}$. Note: in this figure both markers and solid lines correspond to FE solutions.

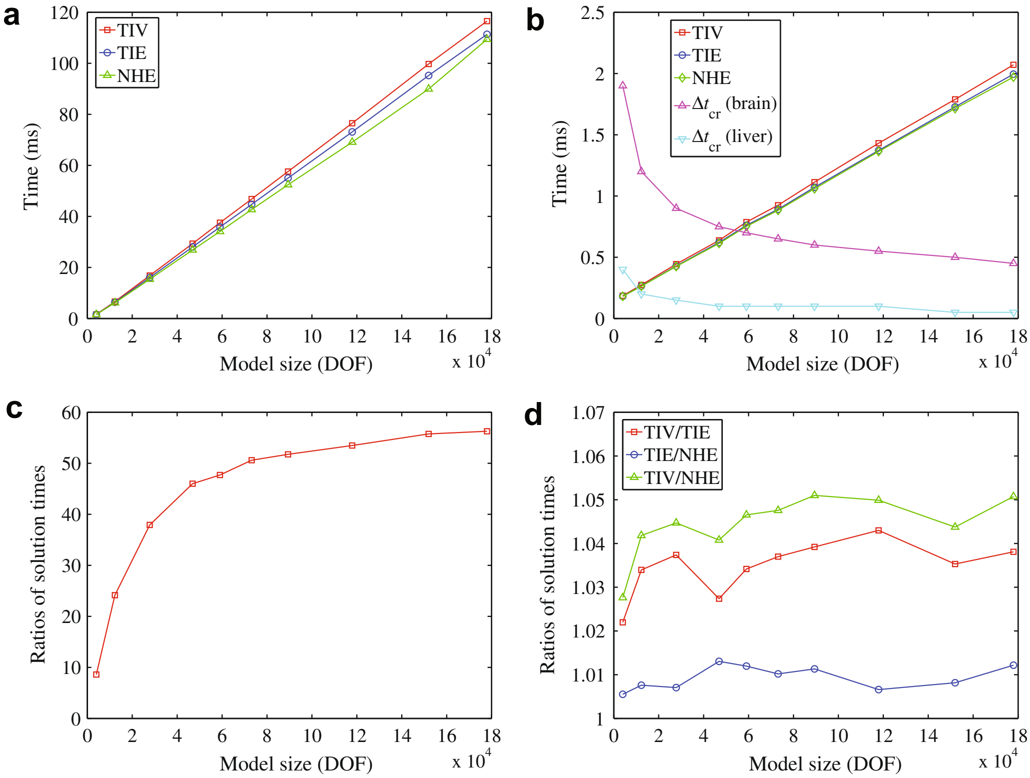

Fig. 5. (a) CPU solution times for a single time step for each constitutive model as a function of model size, (b) GPU solution times for a single time step for each constitutive model as a function of model size, plus critical time steps Δt_{cr} for each model size using liver-like and brain-like properties, (c) ratios of CPU to GPU solution times for model TIV, and (d) ratios of GPU solution times for the constitutive models: TIV to TIE, TIE to NHE, and TIV to NHE.

model the direction defined by $y = z$ was stiffened, and Fig. 3d shows the resulting reduced expansion along this axis and the increased orthogonal expansion.

The end face reaction force history for each model is shown in Fig. 4. A distinct decay curve resulting from stress relaxation, and similar to that commonly noted in biological tissues (Fung, 1993), was observed. The increased stiffness afforded by the anisotropic models (albeit in an orthogonal direction to the loading) result in greater reaction forces than the isotropic model.

6.3. GPU performance

Next, we assessed the computational performance of the GPU implementation and efficiency of the constitutive update scheme by measuring computation times for a range of mesh densities. Using the cube geometry as above we generated models with mesh sizes ranging from 3993 DOF (1331 nodes, 1000 hexahedral elements) to 177957 DOF (59319 nodes, 54872 hexahedral elements). We compared the GPU solution times T for a single time step with those of a CPU-based implementation developed using C++. The test machine included an Intel Core2Duo 2.4 GHz CPU, 2GB RAM, and an NVIDIA GeForce 8800GTX GPU. Three constitutive models were considered: the TI viscoelastic model (TIV), TIV minus the viscoelastic terms (TIE), and TIE minus the anisotropy terms (NHE). For TIV and TIE the preferred direction was defined by $\mathbf{a_0} = [0\,1\,0]$ in each case.[4] Comparison of times for TIV and TIE indicates the computational load introduced by the developed viscoelastic constitutive update scheme. Comparison of times for TIE and NHE (which is similar to the neo-Hookean model used previously (Taylor et al., 2007a, 2008) indicates the computational load introduced by material anisotropy. Fig. 5 shows the results of the experiments.

In Figs. 5a and b it can be seen that GPU solution affords significant speed improvements over CPU solution for all constitutive models. Additionally it appears that solution times are little affected by the introduction of the more complex constitutive models, and importantly by the use of the developed constitutive update scheme. This is borne out in Fig. 5d, where we observe that the maximum solution time ratio for model TIE to model NHE (anisotropy vs isotropy) was 1.013, and that of model TIV to model TIE (viscoelastic vs elastic) was 1.043. The largest *total* solution time increase for an anisotropic viscoelastic model compared with an isotropic hyperelastic one was 5.1%. We conclude that the key features of anisotropy and viscoelasticity may be included in simulations at very little additional computational cost.

Referring to Fig. 5c we observe a maximum speed improvement for GPU solution over CPU solution of 56.3 × . In our previous publication (Taylor et al., 2008) we reported a maximum speed up of 16.8× using an NVIDIA GeForce 7900GT GPU and an OpenGL/Cg-based implementation. The present results imply that despite improvements in CPU technology in the interim, current GPUs combined with the CUDA API allow even greater speed ups for the TLED algorithm.

Fig. 5b also shows the decreasing critical time steps Δt_{cr} with increasing model sizes (which results from the decreasing element sizes), using *liver*-like and*brain*-like materials. We define liver-like materials to be those described by model TIV with material parameters quoted at the beginning of the section, whereas brain-like materials are those described by model NHE with material parameters $\mu = 1006.71$Pa and Poisson's ratio $\nu = 0.45$ (see discussion in

[4] The choice of preferred direction has no effect on the solution times.

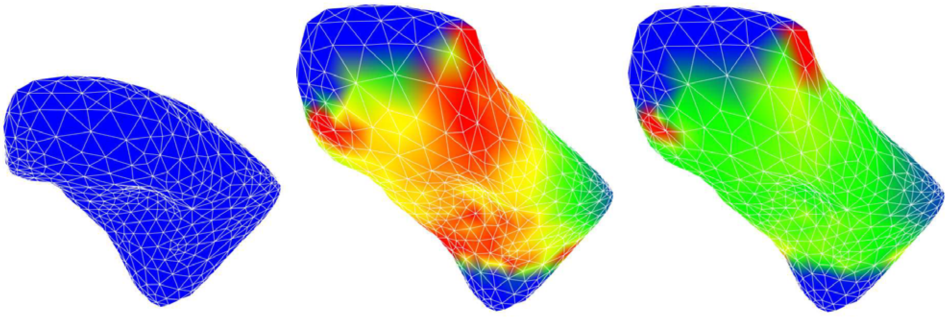

Fig. 6. Deformation of a liver model using the TI model (middle) and the TIV model (right). The undeformed mesh is shown on the left. Colour maps indicate the relative Von Mises equivalent stress magnitude. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Taylor et al., 2008). The intersections of these curves with the solution time curves provide an estimate of the real-time capacity of the implementation. This is discussed further in Section 7.

6.4. Simulation of liver deformation

Finally, we performed simulations of manipulation of a liver model in order to demonstrate the feasibility of the procedure with more realistic organ geometries. The model geometry was obtained from segmented CT data. An initial mesh was generated using *TetGen* (available at http://tetgen.berlios.de), and subsequently smoothed to improve uniformity of element sizes and node valencies. A mesh of 748 nodes and 2428 tetrahedral elements was solvable in real-time. For anisotropic models a preferred material direction $\mathbf{a_0} = [0\,1\,0]$ was used, corresponding to the vertical direction in Fig. 6. We note that in a real organ the preferred direction likely varies depending on the local vasculature and orientation of lobules (Chui et al., 2007), but in the absence of experimental data for the present specimen the mentioned direction was assumed throughout. Moreover, on this point we note that different preferred directions may be specified for each element in the mesh, if such data are available.

The Von Mises equivalent stress distributions resulting from stretching the model with and without inclusion of viscoelasticity are compared in Fig. 6. For the viscoelastic case (Fig. 6, right image) the model is depicted several seconds after the completion of the stretch, so that stresses had reached an approximately steady state. The different stress time-evolutions for the two models are shown in Video Clip 1, also (Supplementary material). The constitutive model employed in each case is indicated in the right hand panel of the Video Clip. The stress dissipation introduced by the viscoelastic model is clearly evident, and for example would result in significantly altered virtual tool reaction forces. The deformed model shape is largely controlled by the volumetric response, and since only isochoric viscoelastic terms were used minimal shape change results in this case.

Additional animations (Video Clips 2 and 3 – online) show the effects of different models subject to indentation. Video Clip 2 shows the Von Mises stresses around an indented region on the top of the mesh using isotropic elastic and viscoelastic models. Again, introduction of viscoelasticity leads to distinct stress dissipation. Video Clip 3 shows the under side of the mesh subject to the same indentation. In this case isotropic and anisotropic elastic models are compared, and the different resulting stress distributions may be noted.

7. Discussion and conclusions

We have made two main contributions in this work: (1) establishment of an efficient procedure which allows material anisotropy and viscoelasticity to be included in surgical simulations at little additional computational cost, and (2) implementation of the procedure (within the TLED algorithm) for GPU execution using the CUDA API. Our methods are based on kinematically consistent deformation measures and nonlinear constitutive equations, which allow analysis of soft tissues under large deformations. Compared with use of isotropic hyperelastic constitutive models our procedure allows incorporation of viscoelasticity for as little as 4.3% additional computational cost, and both anisotropy and viscoelasticity for as little as 5.1%. Additionally our new GPU-based solution scheme for the TLED algorithm provides a speed improvement of up to 56.3× over CPU solution.

A constitutive framework for soft tissues was presented in Section 3. The elastic component of this framework (Section 3.2) was treated in a general way in terms of strain energy functions, so that any hyperelastic formulation may be incorporated. A simple anisotropic strain energy function was used in the numerical examples, however as noted specification of particular forms would be motivated by the particular tissues under analysis. A number of relevant articles were cited in Section 1.

Constitutive models were presented in terms of separated isochoric and volumetric terms, which allows relaxation mechanisms to be included in either or both. Many treatments of visco-hyperelasticity e.g. Holzapfel (1996); Kaliske and Rothert (1997); Bonet (2001) neglect volumetric relaxation terms, citing experimental data for polymeric materials. However it is well known that soft tissues exhibit volumetric relaxation also (e.g. see (Taylor and Miller, 2006) for discussion), for example due to interstitial fluid flow. Inclusion of volumetric relaxation terms could therefore enhance the completeness of the constitutive modelling of such materials.

The numerical examples included isotropic and TI materials, however the constitutive framework presented in Section 3 included equations for orthotropic materials also. It may be seen that an orthotropic version of Eq. (31) may be produced by adding an $\bar{I}_6$-dependent term similar to the $\bar{I}_4$ term. While we have not obtained results for such a model, based on the cost of introducing the $\bar{I}_4$ term it may be assumed that the additional cost of an $\bar{I}_6$ term would be small. Since the viscoelastic constitutive update procedure (Section 4) operates on the final stress values only its computational cost is independent of the complexity of the underlying hyperelastic function.

Our CUDA-based GPU implementation affords significant speed improvements over our recent graphics-based implementation (Taylor et al., 2007a, 2008), and further demonstrates the viability of GPU computing. The gains are partially due to improved hardware, yet our experiments (not presented) have indicated that the present implementation is around 2.5× faster than the previous implementation on current hardware. As the arrangement of the computational kernels is essentially unchanged it would seem that among other things CUDA affords large improvements in memory access performance – a significant bottleneck in graphics-based implementations.

Of interest in interactive surgical simulation is the size of the model which may be solved in real-time. For explicit FE analyses real-time solutions are those for which $T \leqslant \Delta t_{cr}$. Referring to Fig. 5b we estimate that for liver-like material properties and model dimensions of around 0.1 m, models of up to 10000 DOF may be solved in real-time using hexahedral elements. This capacity is reduced for equivalently sized tetrahedral meshes since these entail higher node valencies and possibly lower Δt_{cr}.[5] Additionally, unstructured meshes may reduce real-time capacity due to the presence of smaller elements and higher node valencies – hence use of the smaller liver model in Section (6.4). Since element nodal force contributions are gathered and summed in the second kernel, higher valencies will reduce performance (see Taylor et al., 2008). Therefore we emphasise that real-time capacity is ambiguous for explicit solvers since Δt_{cr} depends on element dimensions and material stiffness (refer to Taylor et al., 2008). In particular it is inappropriate to compare it with the value of 10000 DOF quoted in Taylor et al. (2008), since this was for significantly more compliant brain tissue (μ = 1006.71 Pa) and assumed slightly higher compressibility (Poisson's ratio ν = 0.45). Moreover tetrahedral elements were used in the earlier study. The time steps for models using such a material are plotted in Fig. 5 b also. From these we estimate that for brain-like materials the real-time capacity of the present implementation increases to around 55000 DOF. For stiffer materials implicit integration procedures may be more effective.

Another issue which can impact in this area is the modelling of contact between organs and other virtual objects. While this is beyond the scope of the present work, we note that a range of contact formulations are available in industry standard commercial FE codes, e.g. LS-Dyna (Hallquist, 2006) and ABAQUS/Explicit (ABAQUS, 2003). Two main types are in common use: penalty methods and kinematic constraint methods. Penalty methods involve use of fictitious contact springs at interfaces, effectively to penalise interpenetration of opposing surfaces. If excessively stiff springs are used a reduced Δt_{cr} may result, meaning that implicit methods may be more effective in such cases. According to Hallquist (2006), except in extreme conditions (e.g. from explosive or high speed impacts, or imposition of very large forces on highly constrained objects) spring stiffnesses of the same order of magnitude as the material stiffness are suitable, resulting in minimal effect on Δt_{cr}. However experiences from surgical simulation have suggested a more detrimental effect for explicit analyses (Zhuang and Canny, 1999; Picinbono et al., 2002). Kinematic constraint methods involve (loosely) direct relocation of penetrating nodes back to the outside surface of the penetrated body, and appear to be more stable than penalty methods in explicit analyses (Picinbono et al., 2002). These may be especially useful for modelling interaction between soft tissues and (effectively) rigid surgical tools. The computational cost of either variety has not been investigated here.

By introducing an efficient solution method for anisotropic viscoelastic materials and an improved GPU-based solution scheme we have expanded the utility of the TLED algorithm for soft tissue simulation. The contributions of this paper should facilitate greater physical fidelity in modelling of tissue deformations, and be of particular interest in time-critical applications. Specific examples include biomechanically driven intraoperative image registration and interactive surgical simulation.

Acknowledgement

The financial support of the EPSRC (Grant reference: EP/F01144X/1) and the CSIRO Preventative Health Flagship is gratefully acknowledged. The authors also thank Mr E. Bonner and Mr D. Conlan for assistance with software development, and Dr L. Soler for providing liver CT data.

Appendix A. Stresses under pure shear of a unit cube

Imposing a ramped shear loading according to $r = ut$ will result in the following deformation gradient:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & 0 \\ rt & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{A.1}$$

Using this in Eq. (35) with a preferred direction defined by $\mathbf{a_0} = [1\,0\,0]$ yields four independent nonzero stress terms:

$$\begin{aligned} S_{11} = {} & \frac{1}{3}r^2(t^2(\alpha_1 - 1)(\eta(r^2t^2 - 2) + \mu) - 2\alpha_1\tau_1(2r^2t^3\eta \\ & - 6r^2t^2\eta\tau_1 + (t + (e^{-\frac{t}{\tau_1}} - 1)\tau_1)(\mu + 2\eta(6r^2\tau_1^2 - 1)))), \end{aligned} \tag{A.2}$$

$$\begin{aligned} S_{22} = {} & \frac{1}{3}r^2\Big(-t^2((1 + r^2t^2)^2\eta + (4 + r^2t^2)\mu) + \\ & \alpha_1(t^2(\chi + r^2t^2((2 + r^2t^2)\eta + \mu)) + \\ & 2\tau_1\Big(-t(\chi + r^2t^2((4 + 3r^2t^2)\eta + 2\mu)) + \tau_1\Big(\chi\Big(1 - e^{-\frac{t}{\tau_1}}\Big) \\ & + 15r^4t^4\eta - 60r^4t^3\eta\tau_1)))), \end{aligned} \tag{A.3}$$

$$\begin{aligned} S_{33} = {} & \frac{1}{3}r^2(t^2(\alpha_1 - 1)(\eta(r^2t^2 + 1) + \mu) - 2\alpha_1\tau_1(2r^2t^3\eta \\ & - 6r^2t^2\eta\tau_1 + (t + (e^{-\frac{t}{\tau_1}} - 1)\tau_1)(\mu + \eta(12r^2\tau_1^2 + 1)))), \end{aligned} \tag{A.4}$$

$$\begin{aligned} S_{12} = {} & \frac{1}{3}(-rt(\alpha_1 - 1)(r^4t^4\eta + 3\mu + r^2t^2(\eta + \mu)) \\ & + e^{-\frac{t}{\tau_1}}r\alpha_1\tau_1(-3\mu - 6r^2\tau_1^2(\eta + \mu + 20r^2\eta\tau_1^2) + e^{\frac{t}{\tau_1}}(3\mu \\ & + 3r^2(\eta + \mu)(t^2 - 2t\tau_1 + 2\tau_1^2) + 5r^4\eta(t^4 - 4t^3\tau_1 \\ & + 12t^2\tau_1^2 - 24t\tau_1^3 + 24\tau_1^4)))) \\ = {} & S_{21}, \end{aligned} \tag{A.5}$$

where

$$\chi = \eta + 4\mu + 12r^2\tau_1^2(2\eta + \mu + 30r^2\eta\tau_1^2). \tag{A.6}$$

It may be noted that the bulk modulus κ does not appear in the equations above, since the deformation involves no volumetric components.

Appendix B. Supplementary material

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.media.2008.10.001.

References

ABAQUS, 2003. ABAQUS Analysis User's Manual Version 6.4. ABAQUS, Inc., USA.

Allard, J., Cotin, S., Faure, F., Bensoussan, P.J., Poyer, F., Duriez, C., Delingette, H., Grisoni, L., 2007. SOFA – an open source framework for medical simulation. In: Medicine Meets Virtual Reality. pp. 13–18.

Archip, N., Clatz, O., Whalen, S., Kacher, D., Fedorov, A., Kot, A., Chrisochoides, N., Jolesz, F., Golby, A., Black, P.M., Warfield, S.K., 2007. Non-rigid alignment of pre-operative MRI, fMRI, and DT-MRI with intra-operative MRI for enhanced visualization and navigation in image-guided neurosurgery. NeuroImage 35 (2), 609–624.

Bathe, K.-J., 1996. Finite Element Procedures. Prentice Hall, Upper Saddle River, NJ.

[5] For hexahedra and tetrahedra with comparable edge lengths the so-called *characteristic element length* L_e is smaller for the latter. $\Delta t_{cr} \propto L_e$ is then similarly reduced.

Bonet, J., 2001. Large strain viscoelastic constitutive models. International Journal of Solids and Structures 38 (17), 2953–2968.

Bonet, J., Burton, A.J., 1998. A simple orthotropic, transversely isotropic hyperelastic constitutive equation for large strain computations. Computer Methods in Applied Mechanics and Engineering 162, 151–164.

Carter, T.J., Sermesant, M., Cash, D.M., Barratt, D.C., Tanner, C., Hawkes, D.J., 2005. Application of soft tissue modelling to image-guided surgery. Medical Engineering and Physics 27 (10), 893–909.

Chui, C., Kobayashi, E., Chen, X., Hisada, T., Sakuma, I., 2007. Transversely isotropic properties of porcine liver tissue: experiments and constitutive modelling. Medical and Biological Engineering and Computing 45 (1), 99–106.

Clatz, O., Delingette, H., Talos, I.F., Golby, A.J., Kikinis, R., Jolesz, F.A., Ayache, N., Warfield, S.K., 2005. Robust nonrigid registration to capture brain shift from intraoperative MRI. IEEE Transactions on Medical Imaging 24 (11), 1417–1427.

Comas, O., Taylor, Z.A., Allard, J., Ourselin, S., Cotin, S., Passenger, J., 2008. Efficient nonlinear FEM for soft tissue modelling and its GPU implementation within the open source framework SOFA. In: International Symposium on Computational Models for Biomedical Simulation. London, UK.

Cotin, S., Delingette, H., Ayache, N., 1999. Real-time elastic deformations of soft tissues for surgery simulation. IEEE Transactions On Visualization and Computer Graphics 5 (1), 62–73.

CUDA, 2007. NVIDIA CUDA Programming Guide Version 1.0. NVIDIA Corporation.

Edwards, P.J., Hill, D.L.G., Little, J.A., Hawkes, D.J., 1998. A three-component deformation model for image-guided surgery. Medical Image Analysis 2 (4), 355–367.

Fernando, R., Kilgard, M.J., 2003. The Cg Tutorial: The Definitive Guide to Programmable Real-Time Graphics. Addison-Wesley Professional, Castleton, New York.

Ferrant, M., Warfield, S.K., Nabavi, A., Macq, B., Kikinis, R., 2000. Registration of 3D intraoperative MR images of the brain using a finite element biomechanical model. In: Third International Conference on Medical Image Computing and Computer Assisted Intervention. Pittsburgh, USA, October 11–14, pp. 19–28.

Flanagan, D.P., Belytschko, T., 1981. A uniform strain hexahedron and quadrilateral with orthogonal hourglass control. International Journal for Numerical Methods in Engineering 17, 679–706.

Fung, Y.C., 1993. Biomechanics: mechanical properties of living tissues, second ed. Springer-Verlag, New York.

Hallquist, J.O., 2006. LS-DYNA Theory Manual. Livermore Software Technology Corporation, Livermore, California.

Holzapfel, G.A., 1996. On large strain viscoelasticity: continuum formulation and finite element applications to elastomeric structures. International Journal for Numerical Methods in Engineering 39, 3903–3926.

Holzapfel, G.A., 2000. Nonlinear Solid Mechanics: A Continuum Approach for Engineering. John Wiley & Sons, Chichester.

Holzapfel, G.A., Gasser, T.C., Ogden, R.W., 2000. A new constitutive framework for arterial wall mechanics and a comparative study of material models. Journal of Elasticity 61 (1–3), 1–48.

Holzapfel, G.A., Gasser, T.C., Stadler, M., 2002. A structural model for the viscoelastic behavior of arterial walls: continuum formulation and finite element analysis. European Journal of Mechanics – A/Solids 21 (3), 441–463.

Hughes, T.J.R., 1987. The Finite Element Method: Linear Static and Dynamic Finite Element Analyses. Prentice-Hall, Inc., Englewood Cliffs, NJ.

Humphrey, J.D., Strumpf, R.K., Yin, F.C., 1990. Determination of a constitutive relation for passive myocardium: I. A new functional form. Journal of Biomechanical Engineering 112 (3), 333–339.

Humphrey, J.D., Yin, F.C., 1987. On constitutive relations and finite deformations of passive cardiac tissue: I. A pseudostrain-energy function. Journal of Biomechanical Engineering 109 (4), 298–304.

Hutter, R., Hora, P., Niederer, P., 2000. Total hourglass control for hyperelastic materials. Computer Methods in Applied Mechanics and Engineering 189, 991–1010.

Joldes, G.R., Wittek, A., Miller, K., 2008. An efficient hourglass control implementation for the uniform strain hexahedron using the total Lagrangian formulation. Communications in Numerical Methods in Engineering 24 (11), 1315–1323.

Kaliske, M., 2000. A formulation of elasticity and viscoelasticity for fibre reinforced material at small and finite strains. Computer Methods in Applied Mechanics and Engineering 185, 225–243.

Kaliske, M., Rothert, H., 1997. Formulation and implementation of three-dimensional viscoelasticity at small and finite strains. Computational Mechanics 19 (3), 228–239.

Limbert, G., Middleton, J., 2004. A transversely isotropic viscohyperelastic material: application to the modeling of biological soft connective tissues. International Journal of Solids and Structures 41 (15), 4237–4260.

Limbert, G., Middleton, J., 2006. A constitutive model of the posterior cruciate ligament. Medical Engineering and Physics 28 (2), 99–113.

Miga, M.I., Paulsen, K.D., Hoopes, P.J., Kennedy, F.E.J., Hartov, A., Roberts, D.W., 2000. In vivo quantification of a homogeneous brain deformation model for updating preoperative images during surgery. IEEE Transactions on Biomedical Engineering 47 (2), 266–273.

Miller, K., 2000. Constitutive modelling of abdominal organs. Journal of Biomechanics 33 (3), 367–373.

Miller, K., Chinzei, K., 1997. Constitutive modelling of brain tissue: experiment and theory. Journal of Biomechanics 30 (11–12), 1115–1121.

Miller, K., Chinzei, K., 2002. Mechanical properties of brain tissue in tension. Journal of Biomechanics 35 (4), 483–490.

Miller, K., Joldes, G., Lance, D., Wittek, A., 2007. Total Lagrangian explicit dynamics finite element algorithm for computing soft tissue deformation. Communications in Numerical Methods in Engineering 23 (2), 121–134.

Nava, A., Mazza, E., Furrer, M., Villiger, P., Reinhart, W.H., 2008. In vivo mechanical characterization of human liver. Medical Image Analysis 12 (2), 203–216.

Pena, E., Calvo, B., Martinez, M.A., Doblare, M., 2007. An anisotropic visco-hyperelastic model for ligaments at finite strains. Formulation and computational aspects. International Journal of Solids and Structures 44 (3–4), 760–778.

Picinbono, G., Delingette, H., Ayache, N., 2003. Non-linear anisotropic elasticity for real-time surgery simulation. Graphical Models 65, 305–321.

Picinbono, G., Lombardo, J.-C., Delingette, H., Ayache, N., 2002. realism of a surgery simulator: linear anisotropic elasticity, complex interactions and force extrapolation. Journal of Visualization and Computer Animation 13 (3), 147–167.

Pinsky, P.M., van der Heide, D., Chernyak, D., 2005. Computational modeling of mechanical anisotropy in the cornea and sclera. Journal of Cataract and Refractive Surgery 31, 136–145.

Pioletti, D.P., Rakotomanana, L.R., 2000. Non-linear viscoelastic laws for soft biological tissues. European Journal of Mechanics – A/Solids 19 (5), 749–759.

Poon, H., Ahmad, M.F., 1998. A material point time integration procedure for anisotropic, thermo rheologically simple, viscoelastic solids. Computational Mechanics 21 (3), 236–242.

Prange, M.T., Margulies, S.S., 2002. Regional, directional, and age-dependent properties of the brain undergoing large deformation. Journal of Biomechanical Engineering 124 (2), 244–252.

Puso, M.A., Weiss, J.A., 1998. Finite element implementation of anisotropic quasi-linear viscoelasticity using a discrete spectrum approximation. Journal of Biomechanical Engineering 120, 62–70.

Shreiner, D., Woo, M., Neider, J., Davis, T., 2006. OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 2, fifth ed. Addison-Wesley, Upper Saddle River, NJ.

Skrinjar, O., Nabavi, A., Duncan, J., 2002. Model-driven brain shift compensation. Medical Image Analysis 6 (4), 361–373.

Stroustrup, B., 1997. The C++ Programming Language, third ed. Addison-Wesley, Reading, Mass.

Szekely, G., Brechbühler, C., Hutter, R., Rhomberg, A., Ironmonger, N., Schmid, P., 2000. Modelling of soft tissue simulation for laparscopic surgery simulation. Medical Image Analysis 4, 57–66.

Taylor, Z.A., Cheng, M., Ourselin, S., 2007a. Real-time nonlinear finite element analysis for surgical simulation using graphics processing units. In: 10th International Conference on Medical Image Computing and Computer Assisted Intervention. Brisbane, Australia, October, pp. 701–708.

Taylor, Z.A., Cheng, M., Ourselin, S., 2008. High-speed nonlinear finite element analysis for surgical simulation using graphics processing units. IEEE Transactions on Medical Imaging 27 (5), 650–663.

Taylor, Z.A., Kirk, T.B., Miller, K., 2007b. Confocal arthroscopy-based patient-specific constitutive models of cartilaginous tissues – I: Development of a microstructural model. Computer Methods in Biomechanics and Biomedical Engineering 10 (4), 307–316.

Taylor, Z.A., Miller, K., 2006. Constitutive modelling of cartilaginous tissues: A review. Journal of Applied Biomechanics 22 (3), 212–229.

Vena, P., Gastaldi, D., Contro, R., 2006. A constituent-based model for the nonlinear viscoelastic behavior of ligaments. Journal of Biomechanical Engineering 128 (3), 449–457.

Warfield, S.K., Talos, F., Tei, A., Bharatha, A., Nabavi, A., Ferrant, M., Black, P.M., Jolesz, F.A., Kikinis, R., 2002. Real-time registration of volumetric brain MRI by biomechanical simulation of deformation during image guided neurosurgery. Computing and Visualization in Science 5 (1), 3–11.

Weiss, J.A., Maker, B.N., Govindjee, S., 1996. Finite element implementation of incompressible, transversely isotropic hyperelasticity. Computer Methods in Applied Mechanics and Engineering 135 (1–2), 107–128.

Wittek, A., Miller, K., Kikinis, R., Warfield, S.K., 2007. Patient-specific model of brain deformation: Application to medical image registration. Journal of Biomechanics 40 (4), 919–929.

Wu, W., Heng, P.A., 2004. A hybrid condensed finite element model with GPU acceleration for interactive 3D soft tissue cutting. Computer Animation and Virtual Worlds 15, 219–227.

Wu, W., Heng, P.A., 2005. An improved scheme of an interactive finite element model for 3D soft-tissue cutting and deformation. Visual Comput 21 (8–10), 707–716.

Zhuang, Y., Canny, J., 1999. Real-time simulation of physically realistic global deformation. In: Proceedings of the IEEE Visualization Conference. San Francisco.