

Bhautik Joshi · Andriy Fedorov · Nikos Chrisochoides · Simon K.
Warfield · Sébastien Ourselin

A Quantitative Assessment of Approaches to Mesh Generation for Surgical Simulation

Abstract In surgical simulation, it is common practice to use tetrahedral meshes as models for anatomy. These meshes are versatile, and can be used with a number of different physically based modelling schemes. A variety of *mesh generators* are available that can automatically create tetrahedral meshes from segmented anatomical volumes. Each mesh generation scheme offers its own set of unique attributes. However, few are readily available. When choosing a mesh generator for simulation, it is critical for it to output good-quality, patient-specific meshes that provide a good approximation of the shape or volume to be modelled. To keep computation time within the bounds required for real-time interaction, there is also a limit imposed on the number of elements in the mesh generated. To the authors knowledge, there has been little work directly assessing the suitability of mesh generators for surgical simulation. This paper seeks to address this issue by assessing the use of six mesh generators in a surgical simulation scenario, and examining how they affect simulation precision. This paper aims to perform these comparisons against high-resolution reference meshes, where we examine the precision of meshes from the same mesh generator at different levels of complexity.

Keywords Mesh Generation · FEA · Soft-body deformation · Validation

B. Joshi & S. Ourselin
BioMedIA Lab, eHealth Research Centre, CSIRO, Brisbane
E-mail: bjoshi@gmail.com, Sebastien.Ourselin@csiro.au

A. Fedorov & N. Chrisochoides
Department of Computer Science, College of William and
Mary, Virginia
E-mail: {nikos, fedorov}@cs.wm.edu

S. K. Warfield
Computational Radiology Laboratory
Children's Hospital
300 Longwood Avenue
Boston MA 02115
E-mail: simon.warfield@childrens.harvard.edu

1 Introduction

Surgical Simulation is a current area of research for many groups, both commercial and academic. The promise of being able to simulate part or all of the human body holds enormous potential for improving patient safety through application in clinical training and in surgical planning. As with all simulation scenarios, high-fidelity models of the objects are required to make the simulation accurate and reliable. A surgical simulation typically needs to run at real-time rates, meaning that the visual update has to run at a minimum of 30Hz, and, if needed, the haptic (force) feedback needs to typically run at least at 500Hz.

The real-time constraint restricts the simulation model complexity, meaning that appropriate choices of modelling scheme are necessary. These simulations need to include a method for modelling the biomechanics of the tissue, and a scheme for generation of a model of the anatomy being simulated. For the biomechanical, physically-based modelling (PBM) scheme, we require a physical model that can faithfully reproduce the behaviour of tissue at real-time rates. To match this, the model generation scheme must generate a model that, when matched with PBM, both provides a suitable level of accuracy and can be computed at the needed rates.

PBM of tissue has been approached in many fashions. Finite Element Analysis (FEA), whilst the most computationally expensive approach, has proven to be the most effective, both in terms of precision and faithful modelling of tissue properties. Despite the computational cost, FEA is possible in near or actual real-time. Initial work by Terzopoulous *et al.* [47] proved the feasibility of this approach, with later work by Cotin *et al.* bringing real-time FEA to surgical simulation [13].

FEA is now used to simulate parts of surgical procedures for training and pre-operative planning. The area in which it has seen most activity is in that of laparoscopic surgery, where the area to be simulated is in a closed, controllable environment. Outside of time-critical applications, FEA is widely used as a clinical tool in a

variety of areas, such as nonrigid registration [19, 48, 12], planning and prediction in craniofacial surgical procedures [22], articulation in joints [50], tumour growth [10], prediction of tissue deformation from needle insertion for biopsies [3], robotic surgery validation [49] and a large variety of other biomechanically-driven simulations.

Biomechanical FEA methods solve a partial differential equation (PDE) over a continuous problem region. To be able to solve this on a computer, the problem region must be decomposed into a series of discrete subdomains, each with their own material and mechanical properties. This discretisation is often referred to as meshing, and in 3D usually comes in the form of the generation of a series of tessellated volumetric elements, which together are the finite elements to be solved over. This tessellation is often referred to as a volumetric grid or mesh.

In a clinical environment, it is desirable to obtain a solution to FEA problems as quickly as possible. In surgical simulation, this constraint is even tighter, requiring a solution in fractions of a second. A common approach is to reduce the number of finite elements that are to be solved over. However, if the number of elements decreases, the precision of the simulation potentially also decreases. The shape and *quality*¹ of the elements themselves affect the solution [42].

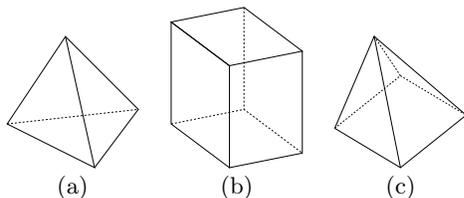


Fig. 1 Three Finite Elements - (a) tetrahedral, (b) hexahedral, (c) pyramidal

3D Finite elements come in many types, some examples of which are shown in Figure 1. The two most commonly used are hexahedral and tetrahedral. Automatic generation of hexahedral meshes from anatomical data remains an open problem, as they can be difficult to generate automatically. A more common choice of FE element in medical simulations are tetrahedra. There are a large number of automatic methods to generate tetrahedral meshes with a wide variety of geometric properties.

In surgical simulation, it is common practice to use tetrahedral meshes as models for anatomy, and can be used with a number of different PBM schemes. A number of available *mesh generators* can create tetrahedral meshes from segmented anatomical volumes. Each mesh generation scheme offers its own set of unique attributes. However, few are readily available. When choosing a mesh generator for simulation, it is critical for it to out-

put good-quality² meshes that provide a good approximation of the shape or volume to be modelled. Adding a further constraint is that the meshes do not generate more elements than are needed, to keep computational time low.

To the authors knowledge, there has been little work directly assessing the suitability of mesh generators for surgical simulation [20]. We present an in-depth extension to our recent previous study [26] by assessing the use of six mesh generation techniques in a surgical simulation scenario, and examining the effects this choice has on simulation precision. We perform these comparisons against high-resolution reference meshes, where we examine the deformation precision of meshes from the same mesh generator at different levels of complexity (resolution).

To this date, it remains a challenging problem to model and simulate tissue. Actual tissue characteristics are difficult to obtain in a consistent fashion, and there is still open speculation as to which biomechanical approximations are most suitable for particular tissue types. In addition, surgical simulation at real-time rates - graphical or haptic - places severe restrictions on the size and complexity of the mesh that can be used. Experiments that compare deformed physical phantoms against simulated material do exist. In particular, we believe that the truth cube experiment by Kerdok *et al.* [27] is a sufficiently thorough test of mesh generation techniques as the geometry of a cube is relatively simple. It has been proven that mesh quality does have a significant impact on the convergence rate and precision of Finite Element simulations [42]. The interaction of this with the number of nodes in the mesh - which is most often a constant in surgical simulation - has not been examined.

Given the difficulty of accurately modelling the biomechanics of real tissue, a ground truth for the deformation of a geometrically complex, Finite Element object is at present difficult to obtain. We approach this problem by using a simpler analytical model - in this case, uniaxial compression of primitives - to obtain an understanding of how the number of nodes in a mesh affects simulation precision. Based on this analysis, we select a cutoff mesh resolution for which the potential error in deformation is minimised. This cutoff is minimum resolution criteria for generating *reference* meshes against which all others are compared to in measuring deformation accuracy.

In Section 1.1, we detail some of the previous work that has been undertaken in this area along with a broad motivation for the study. Section 2 details the methodology of the experiments, detailing the mesh generation strategies used, a description of the physical models, an evaluation of the error in the experimental apparatus and the experimental setup itself. Section 3 examines the results of the deformation experiments, and is concluded in Section 4 with a discussion and some potential directions for future work.

¹ This is usually measured by a geometric criteria, such as the aspect ratio of the elements.

² *i.e.* meshes with good element quality

1.1 Motivation

Biomechanical simulations of soft tissue are commonly modelled via FE techniques. Validation of these simulations has focused on measuring the error between a digital approximation and a physical experiment. Kedrok *et al.* [27] and Miller *et al.* [35] are both recent examples of such experiments, where the deformation of a physical object with embedded markers is imaged. These markers indicate known deformations, and are replicated and measured against in a FE approximation. These models require a good approximation of mechanical tissue parameters, such as elastic modulus or stiffness coefficients [36,51]. FE-assisted non-rigid registration is somewhat more practical to evaluate, where the accuracy of the technique can be compared directly against non-FE-based registration algorithms using real clinical data [12, 8].

For a given FEA to run, a corresponding mesh of the problem domain needs to be generated. The mesh is coupled with a set of initial boundary conditions and the simulation can be run using an FE solver³. Subsequently, FEA accuracy is affected by both the choice of mesh and the FE solver used. In the mesh, good quality (e.g. good aspect ratio) elements are required for the FE solver to converge to an acceptable solution [4, 42, 24]. It is possible that extremely poor quality elements will prevent the simulation converging to a solution.

Poor quality approximations can be improved via h-refinement (mesh) and p-refinement (polynomial degree) of the solution. Few software packages are available that perform p-refinement automatically, and in parallel, but are largely too complex to implement and analyse [28]. Individual mesh generation schemes may also introduce artifacts into the FE simulation, especially in terms of boundary shape and structure. Alliez *et al.* [1] assert that structured mesh generation schemes may cause bias in FE schemes because edges in the mesh tend to be aligned to a finite set of axes, but this point does not appear to be qualified.

Increasing the number of degrees of freedom (number of nodes) in our study improves the precision of the FE solution [32]. This comes at the cost of computation time, however. In time-critical applications, such as surgical simulation, this imposes an upper limit on the mesh size that can be simulated. Different mesh generators offer a variety of approaches to generating quality meshes from a limited number of nodes. However, there has been little work on how FE simulation precision is affected by use of low- to medium-resolution (2,000 – 15,000 nodes) meshes.

To address this, we present a study which compares the deformation fields generated by deforming low- to medium- resolution FEA meshes generated by six different mesh generators. We assess the observed preci-

sion of six tetrahedral mesh generation approaches - red-green crystalline [17], Tetgen [43], SolidMesh [33,34] and GHS3D [46], CHEW [6] (an alternative approach also using GHS3D) and the in-house developed adaptive octree, or OT. For our physical modelling, we use two biomechanical models - linear elasticity, and compressible Neo-hookean elasticity.

2 Method

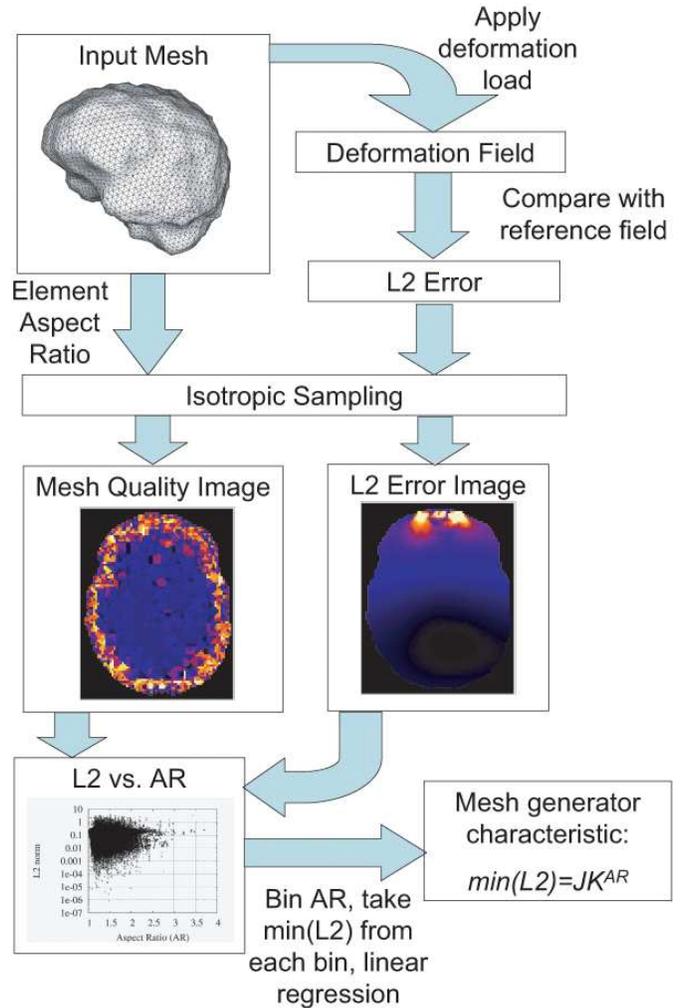


Fig. 2 The proposed scheme for estimating the error characteristic of a particular mesh when it is deformed and compared against a reference deformation field. The technique initially rasterises the deformation field error (L2 norm) and the element quality as images. These images are then compared using the MQL2 metric to establish an error characteristic.

This paper aims to measure the differences in deformation fields generated by different meshing strategies in an FE simulation environment. The inputs, which are

³ An *FE solver* is the specific algorithm in FEA that is used to solve the PDE that the system represents.

a series of segmentations, from the study described by Clatz *et al.* in [11], are fixed. For each case (i.e. each segmentation), a known boundary displacement is applied.

The variables in our experiment are the mesh generation technique and the mesh resolution used, along with the two chosen biomechanical models - linear elasticity and Neo-Hookean elasticity. A high-resolution, reference mesh is used as a gold-standard reference for each clinical case examined, and it provides the standard deformation field against which the other meshes are compared.

We propose a novel scheme for estimating the best-case error performance of a particular mesh generator for generating meshes that simulate biomechanical deformation. Figure 2 provides a schematic summary of the implemented technique.

To present a meaningful discourse on the method, both the mesh generation strategies (Section 2.1) and the corresponding physical models (Section 2.2) are discussed, alongside an evaluation of the error inherent in the experimental apparatus and how this affects selection of a reference resolution for the *gold-standard* meshes (Section 2.3). This is followed by a description of the error metrics used in Section 2.4, and concluded with the experimental setup itself (Section 2.5).

2.1 Mesh Generation Strategies

A number of excellent survey papers [39, 9, 21] have been published in recent years that cover the wide variety of available tetrahedral mesh generators. We will examine, in closer detail, three of the main approaches [18]:

1. *Constrained Delaunay (CDT)* The mesh is initialised with a surface Delaunay triangulation of the boundary points, with subsequent Delaunay refinement and recovery of the domain boundary [44, 1]. CDT can give certain guarantees on mesh quality in 3D.
2. *Advancing Front (AFT)* The mesh is built layer by layer starting from the surface triangulation and progressing towards the object centre [31, 33]. While heuristic, in practical terms the resulting mesh quality is good.
3. *Adaptive Space-tree (AST)* The approaches of this group start from a coarse structured discretisation of the bounding box enclosing the object of interest with subsequent refinement and boundary recovery [41, 19, 25, 18]. Some of these approaches also are based on heuristic algorithms.

The choice of mesh generation software is limited, with the best mesh generators traditionally implemented in the commercial domain. We have selected representative mesh generation techniques from each of the three categories described above:

- *Red-green crystalline meshing* (AST category, further referred as **RGM**) [17]. The mesh size control is

Table 1 Comparison of the evaluated mesh generation methods

method	initial object	adaptivity	max mesh size control
RGM	binary mask	yes	yes
OT	binary mask	yes	yes
TG	tri. surf.	limited	no
SM	tri. surf.	limited	limited
GHS3D	tri. surf.	limited	no
CHEW	tri. surf.	limited	no

achieved by the appropriate choice of the initial lattice spacing. Mesh adaptivity is controlled by user-defined refinement criteria.

- *Octree* (AST category, **OT**) For in-house use, we have developed a parallel, adaptive octree based mesh generator [25]. This scheme uses an octree of the binary mask to be meshed to generate an initial mesh, which is refined at the object boundary to better fit the original surface.
- *Tetgen* (CDT category, **TG**), version 1.4.1, is an implementation of the constrained Delaunay triangulation and refinement [43]. The available implementation does not allow for mesh size to be controlled, but user-controlled adaptive meshing (mesh density variation) is possible.
- *GHS3D* is a commercial implementation of Delaunay method (CDT category, **GHS3D**) [46]. **CHEW** [6] uses **GHS3D** to generate the meshes but uses an alternative scheme for surface mesh generation. Mesh size can only be controlled easily by the input surface size. The input has to be defined by a *piecewise linear complex* (PLC). The implementation does not allow to limit the mesh size. Mesh adaptivity can be achieved by specifying the maximum element volume and prescribed list of points to be inserted. [6]
- *AFLR* is a component of the *SolidMesh* commercial package (AFT category, **SM**) [33, 34]. The input has to be a triangular mesh of good quality. Mesh adaptivity is limited by having smaller elements near the mesh boundary only.

As shown in Table 1, most of the methods generate meshes based on an initial triangular surface. When the triangular mesh needed re-sampling, we used the method described in [14] to find a triangular mesh of a prescribed size. Except **RGM**, and **OT** the methods require a triangular surface in order to generate a mesh. Surface reconstruction from a segmented volume image is a separate research issue; however, in order to avoid another dimension of complexity in this study, surface mesh generation was standardised. We used the triangular surface mesh extracted from **RGM** volume meshes as input for **SM**, **TG** and **GHS3D** methods, and used the surface mesh generation method detailed in [6] in **CHEW**. We ob-

served that for the same surfaces, quality volume meshing with **TG** produced very large meshes unacceptable for our study. Smaller size meshes were possible to generate by instructing **TG** not to insert new points on the surface. However, this led to the meshes of significantly lower element shape quality⁴. The **GHS3D** implementation did not introduce any new points on the input surface and produced high quality meshes, but again, created meshes which were too large to be considered in the study. We used an implementation of a technique described in [14] to experimentally find a triangular surface of such a size that the volume mesh meets the size constraints imposed by the application of interest.

Techniques like **RGM** and **OT** work with the voxel data directly, avoiding the in-between triangulation stage necessary for the other methods. This also allows for determination of mesh adaptivity in the voxel domain using *octrees*, which is cheaper to calculate than adding adaptivity in an existing mesh.

2.2 Physical Models

Early volumetric models of tissue were based on mass-spring systems. However, mass-spring systems are very sensitive to the damping parameters and time-step selected [7]. Poor choice of parameters lead to an unstable simulation where error can quickly accumulate. A more precise approximation is possible by solving continuum mechanics equations with FEA [14, 15, 40], which, in some configurations, offers the same computational speed as a mass-spring system, but with more precise results [23].

Using FEA, several continuum mechanics models are available, providing a spectrum of techniques that balance simulation size and fidelity against simulation speed. In our study, we have chosen two mechanical models: linear elasticity and Neohookean elasticity. Linear elasticity provides a good approximation of tissue for relatively small deformations, whereas a nonlinear model such as Neohookean elasticity is accurate under a wider range of situations.

This is illustrated in the classic bar-bending example, shown in Figure 3. Figure 3(a) shows the bar at rest, without gravity pulling it down. Under large strains, linear elasticity does not preserve volume, as shown in the artifact in Figure 3(b). A nonlinear model, shown in Figure 3(c) is stable under large strains.

We have used the *salmon* framework developed by Nienhyus *et al.* [37] to simulate both mechanical models for tetrahedral meshes. We drive the system by providing an initial set of displacements and anchored nodes - commonly referred to in FE literature as boundary conditions. Using an explicit time-integration scheme, the system then iteratively converges to a solution, giving a final set of global displacements.

⁴ This is expected behaviour described in the accompanying documentation of *Tetgen*, as the possibilities to improve mesh by inserting new points are limited.

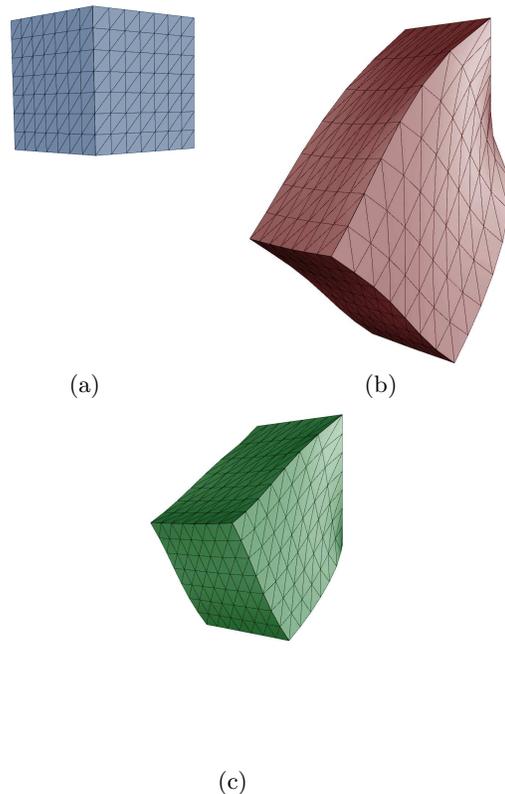


Fig. 3 Classic bar bending example. (a) is the bar at rest state. Under gravity, (b) is the bar modelled using linear elasticity, and (c) is the bar modelled with Neohookean elasticity. Note the volume change in (b).

The system is only conditionally stable, as opposed to our first, quasi-static, scheme [40], but can handle much larger meshes and run in a much smaller computation time. Analysis of the system with regards to its error can be found in Section 2.3.

The material mechanical properties can be encoded as two Lamé coefficients, λ and μ . The elastic modulus, E , and a Poisson ratio ν can be expressed in terms of λ and μ :

$$E = \mu \frac{3\lambda + 2\mu}{\lambda + \mu} \quad (1)$$

$$\nu = \frac{\lambda}{2\lambda + 2\mu} \quad (2)$$

2.2.1 Linear Elastic Material

In the linear elastic material model, the simulated system can be represented by the equation

$$\sigma = C \epsilon \quad (3)$$

where σ is the forces at the nodes, ϵ are the displacements at the nodes and C is the assembled stiffness matrix. Initially, the surgical simulation environment used

in our study was based on a quasi-static linear elastic model of deformation, which allows for both haptic and visual feedback from interacting with a FE mesh [40, 29]. Unfortunately, this does not scale well in terms of size, and has a high cost in the pre-computation of the inverse of the stiffness matrix, C^{-1} . We instead use *salmon* [37], which, while offering a lower degree of accuracy (see Section 2.3), allowed for much larger meshes to be used, with the simulations running at a much faster rate.

2.2.2 Compressible Neo-Hookean Elastic Material

The compressible Neo-Hookean constitutive model implemented by Nienhuys is an alternative version of the St. Venant-Kirchhoff tissue model. It is primarily used to describe rubbery material, which has been found to be a better approximation than linear elasticity. The model is nonlinear, and copes better under bending stresses, as illustrated in Figure 3. The material is able to undergo a more realistic, nonlinear response.

The system can be described as

$$\mathbf{S} = \mu(\mathbf{I} - \mathbf{C}^{-1}) + \lambda T(T - 1)\mathbf{C}^{-1} \quad (4)$$

Supplementary to this, we define F as the deformation gradient tensor, where

$$\mathbf{F} = \begin{bmatrix} \frac{\partial X_1}{\partial x_1} & \frac{\partial X_2}{\partial x_1} & \frac{\partial X_3}{\partial x_1} \\ \frac{\partial X_1}{\partial x_2} & \frac{\partial X_2}{\partial x_2} & \frac{\partial X_3}{\partial x_2} \\ \frac{\partial X_1}{\partial x_3} & \frac{\partial X_2}{\partial x_3} & \frac{\partial X_3}{\partial x_3} \end{bmatrix} \quad (5)$$

where x is the position at rest, and X is the position at a given time. S is the second Piola-Kirchhoff stress tensor, $C = F^T F$ is the Right Cauchy-Green deformation tensor, and $T = \det(\mathbf{F})$.

2.3 Deformation Error Estimation

To measure the error in the simulation, we simulate an artificial scenario and compare against an analytical result, the error is zero. For this paper, we simulate the compression of a block of material along one axis - uniaxial compression.

In uniaxial compression, a volume is anchored on one face such that it is free to move in the yz -plane, but fixed in movement along the x -axis. A load is applied to the opposing face, so that the volume is compressed or pulled, causing it to move laterally in the y and z directions. Two primitives are used - a cubic block, and a cylinder (Figure 4). The analytical solution for deformation in the nodes volume is the same for both primitives in uniaxial compression.

As illustrated in Figure 5, at rest, the block has height d_0 and, from its centre line, width l_0 . When it is compressed, we obtain new height d and width l . In this state, we define $\Delta d = d - d_0$ and $\Delta l = l - l_0$

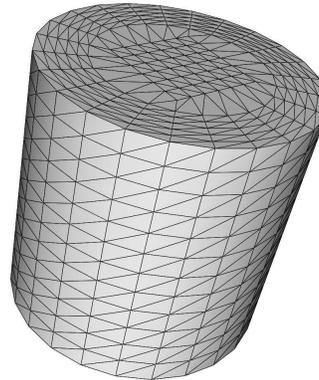


Fig. 4 Unit cylinder primitive used in deformation validation

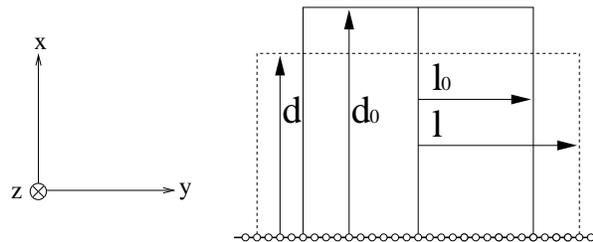


Fig. 5 Uniaxial compression of a volume. At rest, volume has a height of d_0 and, from the centre line, a length of l_0 . After applying a compression, the length changes to l and the height to d . The box is anchored at its base so that it is free to move in the yz -plane but fixed on the x -axis.

2.3.1 Linear Elastic Case

In the linear case, we have a strain tensor, ε . There are no shear components, only movement along the Cartesian axes. Along the x -axis, where the compression occurs

$$\varepsilon_{11} = \frac{\Delta d}{d_0} \quad (6)$$

In the yz -plane, we assume that the strain is equal in both the y and z -axes. Hence

$$\varepsilon_{22} = \varepsilon_{33} = -\nu \varepsilon_{11} = -\nu \frac{\Delta d}{d_0} \quad (7)$$

Since l and l_0 are also known, we can also state that

$$\varepsilon_{22} = \frac{\Delta l}{l_0} \quad (8)$$

These then can be rewritten as

$$\Delta l = -\nu l_0 \frac{\Delta d}{d_0} \quad (9)$$

2.3.2 Neo-hookean Elastic Case

In the Neo-hookean case, we begin with equation 4. Since the stretches are only along the axes, we can state:

$$\mathbf{F} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \quad (10)$$

where:

$$\frac{\partial X_0}{\partial x_0} = \lambda_1 = \frac{d}{d_0} \quad (11)$$

Since the lateral displacements in y and z are equal, we can also assume that

$$\lambda_2 = \lambda_3 = \frac{l}{l_0} \quad (12)$$

And thus

$$\Delta l = l_0(\lambda_2 - 1) \quad (13)$$

Given d and d_0 , we would like to determine Δl . The stresses in the y and z directions are equal to zero. Substituting equations 5 and 12 into equation 4

$$S_{22} = \mu - \frac{\mu}{\lambda_2} + \lambda \lambda_1^2 \lambda_2^2 - \lambda \lambda_1 = 0 \quad (14)$$

Solving for λ_2

$$\lambda_2 = \sqrt{\frac{-\mu + \lambda \lambda_1 + \sqrt{(\mu - \lambda \lambda_1)^2 + 4 \lambda \lambda_1^2 \mu}}{2 \lambda \lambda_1^2}} \quad (15)$$

2.3.3 Reference Resolution Selection

There are two available solvers in *salmon* - static, and dynamic. In the static solver, the selection of a timestep is not necessary. Solution precision is then dependent on the tolerances of the optimiser and the error function chosen. In the dynamic solver, an implicit integration scheme is used. This solver requires careful choice of timestep and damping function. For this reason, and for simplicity, the static solver was chosen.

In FE analysis, simulation precision drops proportional to the number of nodes used in the system. The uniaxial extension of a box is carried out using the *salmon* solver using two constitutive models - linear and neo-hookean. The material parameters are $E = 3000$ and $\mu = 0.45$.

The volumes are extended 5% in the x -axis. After a solution is reached, the distance between the calculated position of a node is compared against the analytical location is recorded (i.e. $dist = p_{observed} - p_{theoretical}$). This is performed for all the nodes in the mesh, and the RMS of all these error values is recorded.

Meshes of unit cubes and cylinders were generated at various resolutions, generating an RMS error for each resolution for both the linear and neo-hookean models.

$\log(RMS)$ is plotted in Figure 6. For the neo-hookean models, a gradual drop-off of error can be clearly seen as mesh resolution is increased. The linear model error is two orders of magnitude smaller than the neo-hookean error, but shows less consistent behaviour, where the error rises again at 30k nodes. This inconsistency may be related to bias caused by the bulk of edges being aligned in a small set of directions, but cannot be substantiated at this stage. Given the order of the RMS error (10^{-6} - 10^{-7}), this may not be a significant issue.

From inspection of Figure 6, the error in displacement appears to converge to a stable value at around 25k nodes. This then serves as a baseline for the minimum required resolution for *gold-standard* meshes to be compared against for mesh deformation validation, and is used in the selection of reference meshes in Section 3.

2.3.4 Maximum Extension Estimation

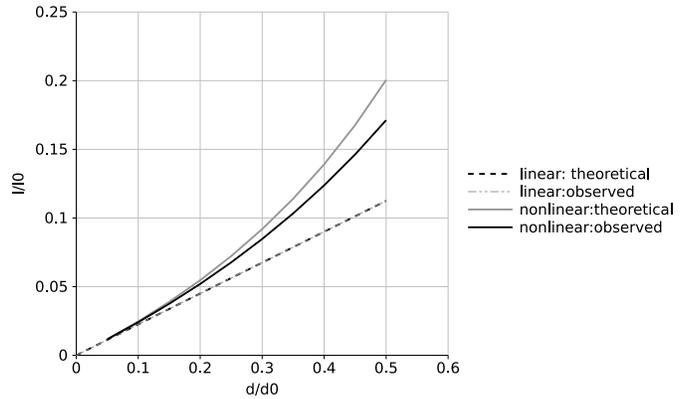


Fig. 7 Uniaxial compression tests on a tetrahedralised unit cylinder, using a linear and Neo-hookean material model in *salmon*. The upper curve shows that in the linear case, the error is small - well outside of 5 significant figures. The uniaxial case demonstrates error increasing as the loading does. However, for small deformations ($< 10\%$ of the cube length), the error remains outside of 3 significant figures.

We set up an experiment within *salmon* to test the error in compression of a unit cylinder, comprising of 20000 nodes. This mesh was subject to a deformation load along the x -axis, and its deformation response is recorded in Figure 7 for a linear and Neo-hookean material model. For the linear model, we were unable to detect any error greater than 0.001% of the theoretical value, even at extreme deformations (i.e. > 0.5 of object length). However, for the Neo-hookean case, the error increased as the compression did. For small deformation (less than 10% of object length), the error remained less than 0.1%. Using the uniaxial compression as a benchmark for deformation experiments to follow, this defines the upper limit of precision within the *salmon* FE solver for deformation experiments. The Lamé parameters used were $\lambda = 9310.3$ and $\mu = 1034.5$ [35].

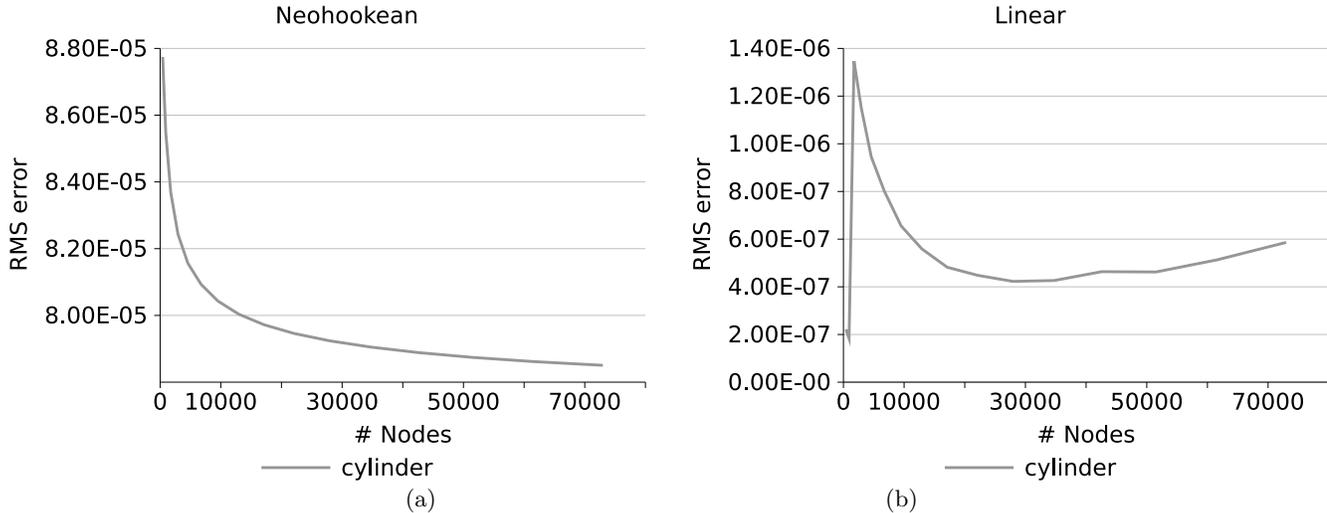


Fig. 6 Uniaxial extension with linear and neohookean constitutive models using both unit cylinder and unit cube models.

2.4 Error Estimation Metrics

Alteroviz *et al.* have examined the problem of trying to benchmark in [2] soft tissue deformation algorithms. They present a number of metrics, such as the relative error norm (REN), the interior error metric (IEM, or L2 norm) and exterior error metric. The REN and L2 norm are commonly used in evaluating the precision and accuracy of FE analysis, and are often used in determining the convergence of the FE problem being solved [30, 16, 28].

A wealth of literature is available detailing *a posteriori* methods for determining the possible error in a simulation, which can take into account model geometry and topology, and the value and local gradient of an initial FE solution [30, 52, 38, 45, 5]. A global metric such as the L2 norm or REN effectively reduces the entire mesh down to a single datapoint, which, in a complex, non-analytical case may not properly take into account local changes in mesh quality and the range of boundary conditions that may be applied in the simulation.

An effective error estimation metric should be able to assess a particular mesh generator by establishing the local relationship between the mesh quality in a particular area of the mesh, and the measured L2 (or REN) error in that particular region. A particular mesh generator may be labelled as more effective if, for a given number of nodes, the error rate predicted by the estimation metric is lower than that of other mesh generators it is compared to.

2.4.1 Re-sampling Deformation Fields

To remove bias in comparing one deformation field against another, we need to re-sample both fields using the same referential. In our case, we use a uniformly sampled point distribution (i.e. on an isotropically sampled grid). Given

a deformation field, D , and sample point q , to obtain the vector corresponding to the deformation field at q , D_q , we use:

$$\text{ResampleField}(D, q) = D_q \quad (16)$$

This is performed by first identifying the corresponding tetrahedron in which point q lies. Using the barycentric coordinates of q relative to the corners of the tetrahedron, D_q is the weighted sum of the deformation field vectors at the corners of the tetrahedron.

We assume we have two meshes, ma and mb . When a deformation fields DA and DB is applied to ma and mb respectively, a given point, q , may be transformed. In ma , $q_a = q + DA_q$ and in mb , $q_b = q + DB_q$. Using this re-sampling scheme, the fields then can be re-sampled as DA^{resamp} and DB^{resamp} , which are no longer dependent on the nodes of the original mesh to act as sampling points.

2.4.2 MQL2 Metric

A direct, point-to-point relationship between mesh quality and deformation field error in any given mesh is difficult, if not impossible to establish without making an initial attempt at solving the FE analysis. This process (commonly a precursor to further h- or p-refinement) is not practical in a clinical scenario. At the same time, reducing the mesh quality and error norms down to a single data point by averaging over the whole mesh, whilst fast, obliterates subtle information about local changes in mesh quality that are characteristic to each mesh generator.

We take an approach that is a compromise between these two extreme ends of analysis. It is difficult to establish a direct relationship between a given, local mesh quality value and the corresponding deformation error at that point. Because the mesh is made up of a discrete set

of neighbouring elements, each with their own mesh quality, the error caused by one element may spill over and influence the error of neighbouring elements. For all the points in a mesh that correspond to a particular quality value, there will be a range of deformation error values in at the corresponding deformation field error points.

We have found that this range is not bound at its upper point - a given mesh quality element is heavily affected by its neighbours - but it is bound at its lowest point. We have found that the \log_{10} of the *best* (i.e. lowest) possible error value at a given point in the mesh is linearly proportional to the mesh quality at that point.

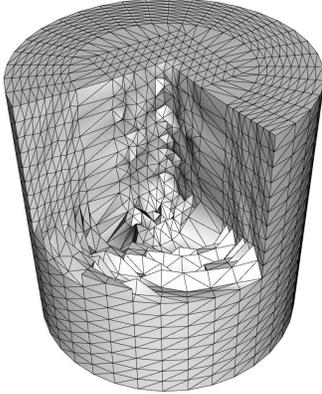


Fig. 8 Cutaway view of cylinder mesh with perturbed nodes. Mesh has 9514 nodes and 52488 tetrahedra.

To test this metric, we selected one of the cylinder meshes used in uniaxial compression and randomly perturbed its internal points to create a range of elements in the mesh with varying quality. A cutaway view of this is shown in Figure 8.

The deformation field error and mesh quality are sampled using the resampling algorithm described in Section 2.4.1. At each sample point, the quality metric (L2) and the mesh quality metric (AR) are recorded as a tuple. This data is plotted as AR vs. L2 in Figure 9(a). It is difficult to establish a direct relationship between AR and L2 because, as previously mentioned, the maximum L2 value for a particular AR value can be unbounded. We divide the AR axis into a series of bins, and plot the minimum value of L2 in each bin, as shown in Figure 9(b). There is a linear relationship between AR and $\log(\min(L2))$. In Figure 9(b), this was verified using linear regression, with a regression coefficient of 0.901.

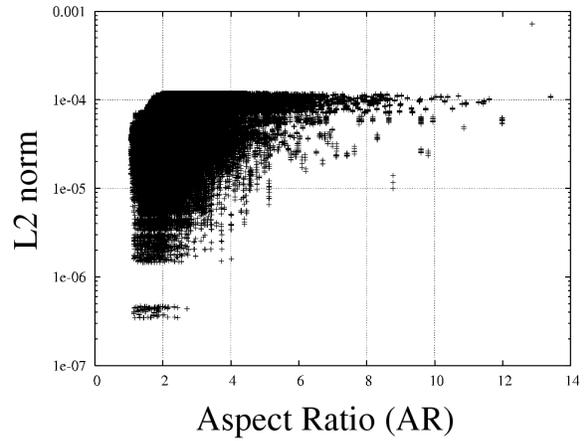
The equation of the fitted line is characteristic for each mesh generation scheme. Less effective mesh generation schemes will have a line that has a greater gradient - that is, worse schemes will degrade in terms of deformation error at a faster rate than a better scheme. The metric does not completely describe the efficacy of a given mesh generation scheme, but the relationship it describes can locally define the best-case scenario in

terms of deformation error based on the mesh quality at a given point. Since every mesh generated has a different distribution in terms of mesh quality, the efficacy of a particular mesh will change from instance to instance.

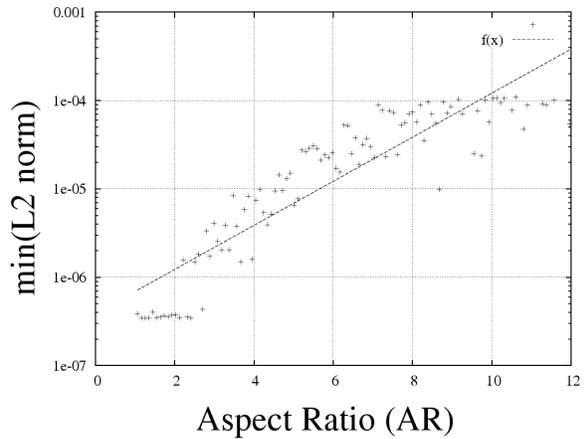
2.5 Experimental Setup

The error metrics detailed above are dependent on a pairwise comparison of two deformation fields. We have six mesh generation techniques, with which we have created four resolution meshes for each case examined:

20k nodes	<	<i>xlg</i>		
10k nodes	<	<i>lrg</i>	<	20k nodes
5k nodes	<	<i>med</i>	<	10k nodes
		<i>sml</i>	<	5k nodes



(a)



(b)

Fig. 9 Quality Metric (AR) vs. $\log(L2 \text{ norm})$ in the uniaxial compression of a cylinder. Plot of all values is in (a). Dataset was divided into bins based on AR. Min of each bin is plotted in (b); linear regression is able to find a line fit to this data with a regression coefficient of 0.901.

If the same deformation is applied to each of these meshes, each mesh can be compared directly against another. The *xlg* classification is for the reference meshes, which each must have $> 25k$ nodes.

From this set of comparisons, we seek to assess two goals - firstly, how mesh size affects the precision of the deformation field (compared against all other meshes of the same size). This gives us an indicator as to how different the deformation fields are for a particular mesh generator, compared to the rest. Secondly, given a particular mesh generator, we examine how the error in the deformation changes as the mesh resolution is dropped, relative to the highest resolution mesh.

Five brain volume segmentations from the study described by Clatz *et al.* in [11] have been selected to test different mesh generation approaches. Six different mesh generation approaches described in Section 2.1 were applied to generate meshes for each of the segmented masks.

We simulate an in-vitro brain deformation by anchoring the nodes of the mesh at the medulla and applying a deformation to the surface, similar to the approach taken by Wittek *et al.* [49]. The anchor nodes are chosen on a per-mesh basis by selecting the nodes that lie within a sphere of a fixed centre and radius.

A contact field is applied to the mesh in order to generate an initial set of displaced nodes. To ensure consistency, this is generated by simulating a plane compressing the frontal lobes in a caudal direction. This deformation is used to drive the simulation. We simulate brain tissue using the in-vivo stiffness parameters $\lambda = 9310.3$ and $\mu = 1034.5$ [35]. Since our mechanical models are only valid for small deformations (see Section 2.3.4), the compression is set to be roughly 5% of rostral-caudal length of the brain.

3 Results

3.1 Mesh sizes

Table 2 lists the average number of nodes for each mesh type generated and used in the experiment. **CHEW** and **GHS3D** have significantly larger number of degrees of freedom. The available mesh generators were not all able to create meshes within the size bounds as specified in Section 2.5, so there are some missing data points. *xlg-GHS3D* meshes were available for each of the five clinical cases, with an average resolution of $28k$ nodes. These meshes were used to generate the required reference deformation fields.

Table 3 records the average and worst-case mesh quality metrics for the different mesh generation techniques at different resolutions. The aspect ratio (AR) of the tetrahedral elements in the mesh is evaluated, with a value closer to 1 indicating better overall element quality in the mesh. Also recorded is the minimum dihedral angle (angle between faces) in the tetrahedra of the meshes; a value close to 60 degrees is ideal.

Table 2 Average mesh sizes across the 5 cases

Technique	<i>xlg</i>	<i>lrg</i>	<i>med</i>	<i>sml</i>
OT	-	14440	-	2782
RGM	-	10329	5948	2684
TG	-	-	5364	3222
SM	-	10293	5879	3451
GHS3D	28572	14168	5228	-
CHEW	33070	13978	6487	-

Table 3 Mesh Quality table

Mesh Generator-size	AR (avg)	AR (max)	MDA (avg)	MDA (min)
ot-lrg	2.01	16.35	41.275	4.535
ot-sml	2.118	24.3	40.12	2.808
ghs3d-xlg	1.508	5.828	49.74	11.616
ghs3d-lrg	1.51	5.198	49.78	13.348
ghs3d-med	1.518	5.17	49.62	13.428
chew-xlg	1.508	4.128	49.84	14.94
chew-lrg	1.514	4.108	49.7	15.82
chew-med	1.518	4.092	49.6	15.46
sm-lrg	1.402	3.956	53.48	15.92
sm-med	1.408	3.586	53.38	18.52
sm-sml	1.364	3.61	54.26	18.92
rgm-lrg	1.446	4.492	52.78	24.74
rgm-med	1.458	3.664	52.56	28.66
rgm-sml	1.468	3.252	52.22	29.18
tg-med	2.312	391.6	45.04	0.28148
tg-sml	2.308	168.217	45.35	0.533

3.2 Deformation Field Evaluation

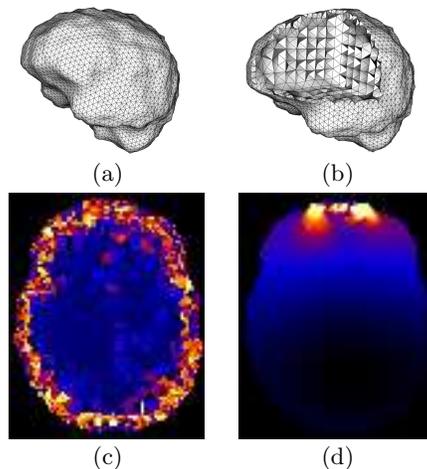


Fig. 10 Case2-RGM-lrg: (a) - original mesh, (b) - cutaway of mesh, (c) quality image, (d) L2 image

An example mesh from the dataset, case2-**RGM-lrg** is shown in Figure 10(a) and (b). A slice through its quality image is in (c), the corresponding L2 image slice in (d). As shown in Figure 11(a), it is difficult or impossible to extract a trend from the totality of the data presented.

However, by taking the minimum value of L2 per bucketed AR value, as shown in (b), the MQL2 gradient is extracted successfully ($L2 = 1.54 \cdot 10^{-15} \cdot 10480^{AR}$), and with a good degree of certainty - a regression coefficient of 0.789. This strong correlation is kept for all the meshes tested, with an average regression coefficient of 0.773.

Using the metric described in Section 2.4.2, we establish that:

$$\min(L2) = J \cdot K^{AR} \quad (17)$$

where J and K are the parameters extracted from linear regression. A table of these values, averaged across the six cases for each combination of mesh generator and resolution, is in Table 4. These values are plotted graphically in Figures 12 and 13.

OT and **TG** generate meshes with poor average and worst-case element quality (Table 3). For this particular experimental set-up, both these schemes appear to perform better compared to their peers in terms of the MQL2 metric. When reading the minimum L2 error for a particular mesh generator at its average element quality value in Figures 12 and 13, the best-case error is still less than that of the other mesh generators.

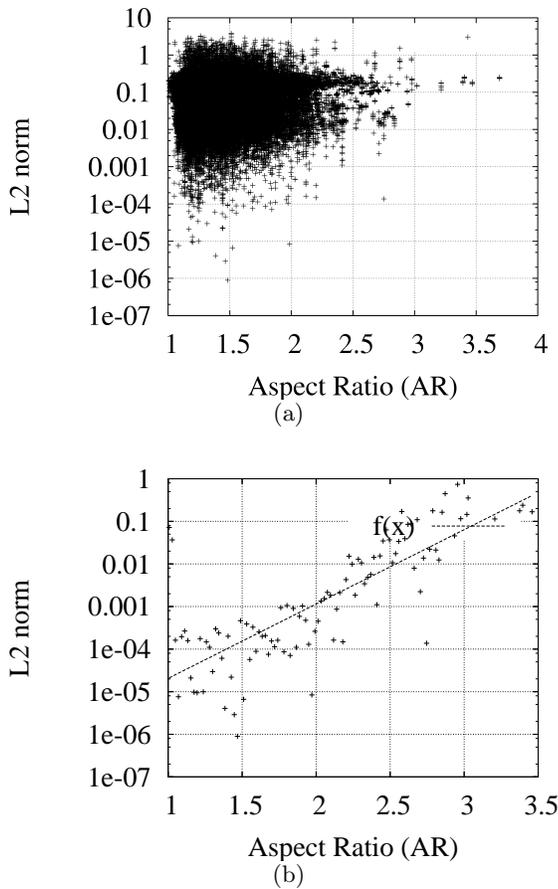


Fig. 11 Case2-RGM-lrg: (a) Unbinned plot of AR vs. L2, (b) Binned plot of AR vs. $\min(L2)$ with MQL2 line fit

Table 4 MQL2 values

MeshGen-Size	#nodes (average)	Neohookean		Linear	
		J	K	J	K
ot-lrg	14440	0.122	0.032	0.158	0.055
ot-sml	2783	0.170	0.112	0.166	0.088
rgm-lrg	10329	0.087	0.044	0.038	0.013
rgm-med	5948	0.199	0.165	0.116	0.104
rgm-sml	2684	0.274	0.147	0.138	0.063
tg-med	5365	0.131	0.070	0.077	0.033
tg-sml	3223	0.183	0.109	0.117	0.032
sm-lrg	10294	0.067	0.030	0.034	0.011
sm-med	5880	0.244	0.193	0.110	0.118
sm-sml	3452	0.365	0.098	0.129	0.022
ghs3d-lrg	14167	0.152	0.146	0.108	0.118
ghs3d-med	5228	0.248	0.063	0.109	0.046
chew-lrg	13978	0.145	0.095	0.195	0.080
chew-med	6487	0.333	0.129	0.227	0.080

In general, the Neohookean material model appears to run with double the magnitude of error of the linear material model. As expected, the average error as the number of nodes drops is higher than the higher resolution cases. The variance of the solutions at lower resolutions is also higher. This is also to be expected as the solution is less deterministic at low resolutions. At large and small resolutions, **OT** appears to consistently be the best mesh generator. In terms of the other schemes available, however, at large and medium resolutions, there is little difference between the schemes.

The best performing schemes are quite different in their approach - **OT** is primarily driven by a structured mesh generation scheme, and **TG** is a Delaunay-based approach. Surface fidelity in this case does not appear to affect the result, as both schemes generate different boundary triangulations. The resulting suitability of a mesh for a biomechanical simulation is a complex combination of mesh generator used, individual element quality and mesh complexity, rather than any one of these factors on their own. The MQL2 metric provides a framework in which these factors may be collectively assessed, allowing an objective comparison of mesh generators for a particular simulation configuration.

4 Conclusions and Future Directions

We have presented an evaluation of several mesh generation techniques, using a novel error metric, the MQL2 gradient. Using a series of analytical cases, we set up the allowable bounds and parameters for the experiments. We ran deformation experiments using FE analysis, comparing against a gold standard, using both linear and Neohookean constitutive models. From these experiments, we were able to draw out the characteristic performance

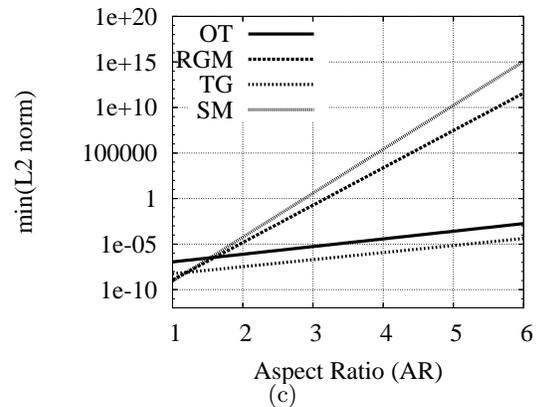
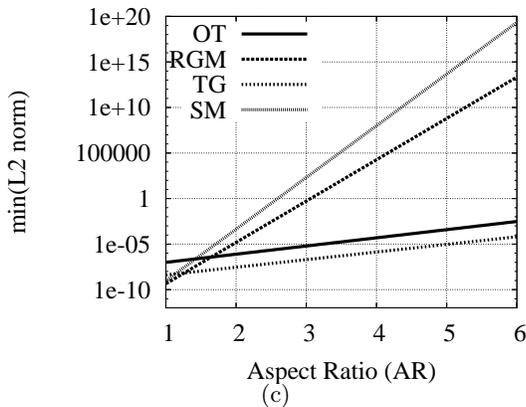
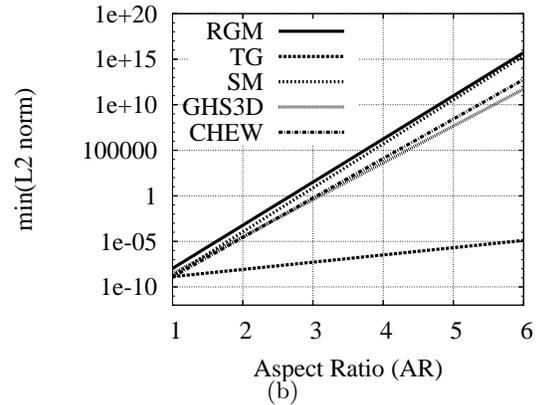
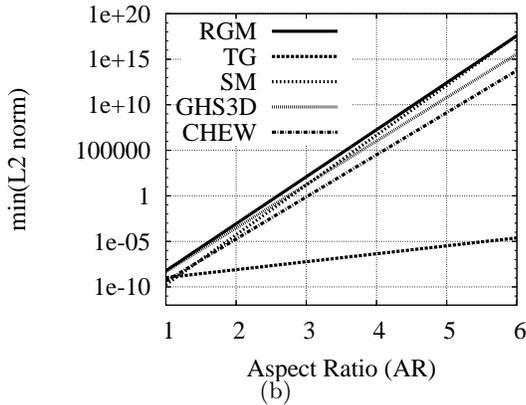
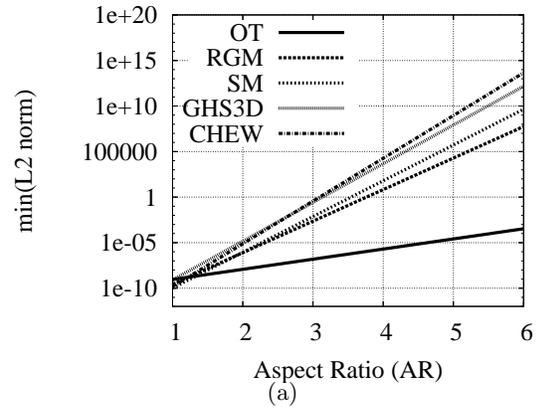
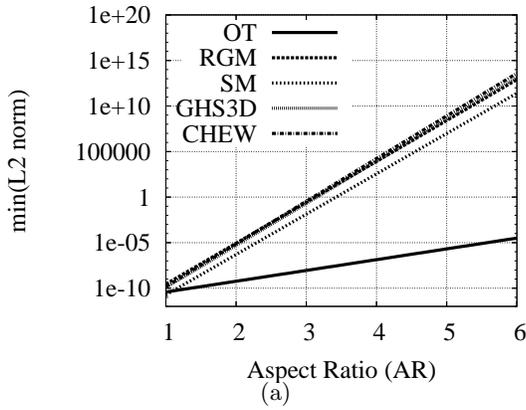


Fig. 12 Plot of the Neohookean MQL2 values from Table 4. (a) large, (b) medium and (c) small

Fig. 13 Plot of the Linear MQL2 values from Table 4. (a) large, (b) medium and (c) small

of the tested mesh generators in terms of error, and were able to determine desirable characteristics of a generated mesh.

Whilst the upper limit on deformation error is unbounded, we have established that quality of an element will locally affect the error in the solution. Locally, the log of the minimum L2 error is proportional to the element quality. The relationship between these two values is characteristic for each mesh generator. The MQL2 er-

ror metric cannot globally predict the overall error in a mesh deformation - but it can be used to establish a global estimate of the *best-case error* in a deformation.

An important factor for the precision of deformed meshes in surgical simulation is the number of nodes used to construct the mesh. In terms of choice of mesh generation technique, this translates to the requirement of having a good degree of control over the number of nodes or tetrahedra that are generated.

In general, the Neo-Hookean material model appears to run with two orders of magnitude greater the error of the linear material model. As expected, the average error as the number of nodes drops is higher than the higher resolution cases. The variance of the solutions at lower resolutions is also higher. This is also to be expected as the solution is less deterministic at low resolutions. An ideal mesh generator would then have a lower MQL2 value as resolution increases, and the variance of solutions these values at this resolution would be expected to drop. At medium and large resolutions, the choice of mesh generator is less significant - the driving factor is that there are as many nodes as can be practically allowed, and that the mesh is of reasonable quality. Another conclusion that can be drawn is that nonlinear physical models, such as Neo-Hookean elasticity, are more sensitive mesh quality than their linear counterparts.

The choice of mesh generation scheme is largely driven by the resolution of the mesh that is required. Individual mesh generation schemes performed characteristically better at a given resolution compared to its peers, rather than performing better at all resolutions. For this experiment, at high resolutions **OT** appears to perform consistently well, keeping a relatively low deformation error for even badly shaped elements. At medium to small resolutions, **TG** outperforms its peers.

This conclusion is drawn from a particular experimental setup, and it is yet to be established how widely this can be applied. What we have been able to demonstrate, however, is that the suitability of a particular mesh for a biomechanical simulation is a complex combination of mesh quality, mesh complexity, and mesh generator used, instead of any one of these factors individually. The presented MQL2 metric combines these factors in a novel way, allowing for the direct comparison of mesh generation schemes for individual simulation configurations.

This still leaves the question of whether structured (such as octree-based) or unstructured (such as Delaunay) are much different from each other in terms of generated error at simulation resolution. Testing this broader hypothesis will require further work outside of the scope of this study.

Acknowledgements We would like to thank Hervé Delingette, Dan Popescu and Zieke Taylor for their help in editing and reviewing this article prior to submission. This investigation was supported in part by a research grant from CIMIT, and by NIH grants R21 MH067054, R03 CA126466 and R01 RR021885.

References

- Alliez, P., Cohen-Steiner, D., Yvinec, M., Desbrun, M.: Variational tetrahedral meshing. *ACM Trans. Graph.* **24**(3), 617–625 (2005). DOI <http://doi.acm.org/10.1145/1073204.1073238>
- Alterovitz, R., Goldberg, K.: Comparing Algorithms for Soft Tissue Deformation: Accuracy Metrics and Benchmarks. Tech. Rep. Draft, ALPHA Lab, UC Berkeley (2002)
- Alterovitz, R., Pouliot, J., Taschereau, R., Hsu, I.C.J., Goldberg, K.: Needle Insertion and Radioactive Seed Implantation in Human Tissues: Simulation and Sensitivity Analysis. In: Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA 2003), pp. 1793–1799 (2003)
- Babuska, I., Aziz, A.: On the Angle Condition in the Finite Element Method. *SIAM J. Num. Anal.*, **13**, 214–226 (1976)
- Becker, R., Rannacher, R.: An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica* **10**(00), 1–102 (2001)
- Boissonnat, J.D., Oudot, S.: Provably good sampling and meshing of surfaces. *Graph. Models* **67**(5), 405–451 (2005). DOI <http://dx.doi.org/10.1016/j.gmod.2005.01.004>
- Bourguignon, D., Cani, M.P.: Controlling anisotropy in mass-spring systems. In: Proceedings of the 11th Eurographics Workshop, Interlaken, Switzerland, Springer Computer Science, pp. 113–123. Springer-Verlag (2000)
- Chi, Y., Liang, J., Yan, D.: A material sensitivity study on the accuracy of deformable organ registration using linear biomechanical models. *Med. Phys.* **33**(2), 421–433 (2006)
- Chrisochoides, N.: A Survey of Parallel Mesh Generation Methods, *Numerical Solution of Partial Differential Equations on Parallel Computers*, vol. 51, pp. 1–487. Springer Verlag, n/a (2006)
- Clatz, O., Bondiau, P., Delingette, H., Malandain, G., Sermesant, M., Warfield, S.K., Ayache, N.: In silico tumor growth: Application to glioblastomas. In: C. Barillot, D. Haynor, P. Hellier (eds.) Proc. of the 7th Int. Conf on Medical Image Computing and Computer-Assisted Intervention - MICCAI 2004 (2), *LNCS*, vol. 3217, pp. 337–345. Springer Verlag, Saint-Malo, France (2004)
- Clatz, O., Delingette, H., Bardinet, E., Dormont, D., Ayache, N.: Patient specific biomechanical model of the brain: Application to parkinson's disease procedure. In: N. Ayache, H. Delingette (eds.) IS4TM'03, *LNCS*, vol. 2673, pp. 321–331. Springer-Verlag (2003)
- Clatz, O., Delingette, H., Talos, I., Golby, A., Kikinis, R., Jolesz, F., Ayache, N., Warfield, S.: Robust nonrigid registration to capture brain shift from intraoperative MRI. *IEEE Trans Med Imaging* **24**(11), 1417–1427 (2005)
- Cotin, S., Delingette, H., Ayache, N.: Real-time elastic deformations of soft tissues for surgery simulation. *IEEE Transactions On Visualization and Computer Graphics* **5**(1), 62–73 (1999)
- Delingette, H.: Initialization of deformable models from 3D data. In: Proceedings of the Sixth Int. Conf. on Computer Vision (ICCV'98), pp. 311–316 (1998)
- Delingette, H., Cotin, S., Ayache, N.: A hybrid elastic model allowing real-time cutting, deformations and force-feedback for surgery training and simulation. In: CA '99: Proceedings of the Computer Animation, p. 70. IEEE Computer Society, Washington, DC, USA (1999)
- Dey, S.: Geometry-based three-dimensional hp-finite element modelling and computations. Ph.D. thesis, Rensselaer Polytechnic Institute, Troy, NY (1997)
- Fedorov, A., Chrisochoides, N., Kikinis, R., Warfield, S.: Tetrahedral mesh generation for medical imaging. In: IJ - 2005 MICCAI Open-Source Workshop (2005)
- Fedorov, A., Chrisochoides, N., Kikinis, R., Warfield, S.K.: An evaluation of three approaches to tetrahedral mesh generation for deformable registration of mr images. In: ISBI2006. IEEE (2006)
- Ferrant, M.: Physics-based Deformable Modeling of Volumes and Surfaces for Medical Image Registration, Segmentation and Visualization. Ph.D. thesis, Laboratoire

- de Télécommunications et Télédétection de l'Université catholique de Louvain, Belgium (2001)
20. Freitag, L.A., Knupp, P.M.: Tetrahedral element shape optimization via the jacobian determinant and condition number. In: International Meshing Round Table 1999, pp. 247–258 (1999)
 21. Frey, J.P., George, P.L.: Mesh Generation, vol. n/a. HERMES Science Publishing, Paris, France (2000)
 22. Gladilin, E., Zachow, S., Deuffhard, P., Hege, H.: Anatomy and physics based facial animation for craniofacial surgery simulations. *Medical & Biological Engineering & Computing* **42**(2), 42 (2004)
 23. Harders, M., Hutter, R., Rutz, A., Niederer, P., Székely, G.: Comparing a simplified FEM approach with the mass-spring model for surgery simulation. *Stud Health Technol Inform* **94**, 103–109 (2003)
 24. Hu, T., Desai, J.P.: Characterization of soft-tissue material properties: Large deformation analysis. In: S. Cotin, D.N. Metaxas (eds.) ISMS, *Lecture Notes in Computer Science*, vol. 3078, pp. 28–37. Springer (2004)
 25. Joshi, B.: Model Generation and Interaction in Surgical Simulation. Ph.D. thesis, BioMedIA Lab, CSIRO and Graduate School of Biomedical Engineering, UNSW (2007). Note: Submitted
 26. Joshi, B., Fedorov, A., Chrisochoides, N., Warfield, S., Ourselin, S.: Application-driven quantitative assessment of approaches to mesh generation. In: 2007 IEEE International Symposium on Biomedical Imaging: from Nano to Macro. IEEE (2007)
 27. Kerdok, A., Cotin, S., Ottensmeyer, M., Galea, A., Howe, R., Dawson, S.: Truth cube: Establishing physical standards for soft tissue simulation. *Medical Image Analysis* **7**, 283–291 (2003)
 28. Kirby, R.M., Warburton, T.C., Lomtev, I., Karniadakis, G.E.: A discontinuous Galerkin spectral/ HP method on hybrid grids. *J Appl. Num. Math* **33**(1–4), 393–405 (2000)
 29. Lee, B., Popescu, D., Ourselin, S.: Contact modelling based on displacement field redistribution for surgical simulation. In: Proceedings of The Second International Workshop on Medical Imaging and Augmented Reality (MIAR 2004), *LNCS*, vol. 3150, pp. 337–345. Springer Verlag, Beijing, China (2004)
 30. Lee, N., Bathe, K.: Error indicators and adaptive remeshing in large deformation finite element analysis. *Finite Elements in Analysis and Design* **16**, 99–139 (1993)
 31. Lohner, R., Parikh, P.: Three-dimensional grid generation by the advancing front method. *International Journal for Numerical Methods in Engineering* **8**, 1135–1149 (1988)
 32. Luo, X., Shephard, M.S., Remacle, J.F.: The Influence of Geometric Approximation on the Accuracy of High Order Methods. In: 8th International Conference on Numerical Grid Generation in Computational Field Simulations (2002)
 33. Marcum, D.: Efficient generation of high-quality unstructured surface and volume grids. *Engineering with Computers* **17**(3), 211–233 (2001)
 34. Marcum, D.: Solidmesh (2005). URL <http://www.erc.msstate.edu/simcenter/docs/solidmesh/>
 35. Miller, K., Chinzai, K.: Simple Validation of Biomechanical Models of Brain Tissue. In: European Society of Biomechanics Conference, UK, vol. 1, p. 104. Elsevier Science (1998)
 36. Miller, K., Chinzai, K.: Mechanical properties of brain tissue in tension. *J Biomech* **35**(4), 483–490 (2002)
 37. Nienhuys, H.W., van der Stappen, A.F.: A surgery simulation supporting cuts and finite element deformation. In: MICCAI '01: Proceedings of the 4th International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 145–152. Springer-Verlag, London, UK (2001)
 38. Oden, J., Ainsworth, M.: A posteriori error estimation in finite element analysis. *Computer Methods in Applied Mechanics and Engineering* **142**, 1–88 (1997)
 39. Owen, S.: A survey of unstructured mesh generation technology. In: Proceedings of 7th International Meshing Roundtable, pp. 239–267 (1998)
 40. Popescu, D.C., Compton, M.: A Model for Efficient and Accurate Interaction with Elastic Objects in Haptic Virtual Environments. In: Proc. 1st International Conference on Computer Graphics and Interactive Techniques in Australia and South East Asia (GRAPHITE'03'), pp. 245–250. ACM Press, Melbourne, Australia (2003)
 41. Shephard, M.S., de Cougny, H.L., O'Bara, R.M., Beall, M.W.: Automatic grid generation using spatially based trees. In: J.F. Thompson, B.K. Soni, N.P. Weatherill (eds.) *Handbook of Grid Generation*, chap. 15. CRC Press, Boca Raton, FL (1999)
 42. Shewchuk, J.R.: What is a Good Linear Element? Interpolation, Conditioning, and Quality Measures. In: Proceedings of the 11th International Meshing Roundtable, Ithaca, New York, USA, pp. 115–126. ACM Press (2002)
 43. Si, H.: Tetgen: A quality tetrahedral mesh generator and three-dimensional delaunay triangulator (2005). URL <http://tetgen.berlios.de/>
 44. Si, H., Gartner, K.: Meshing piecewise linear complexes by constrained delaunay tetrahedralizations. In: Proceedings of the 14th International Meshing Roundtable, pp. 147–163 (2005)
 45. Stewart, J., Hughes, T.: A tutorial in elementary finite element error analysis: A systematic presentation of a priori and a posteriori error estimates. *Computer Methods in Applied Mechanics and Engineering* **158**, 1–22 (1998)
 46. Technologies, S.: Tetmesh-ghs3d (2005). URL <http://www.simulog.fr/mesh/gener2.htm>
 47. Terzopoulos, D., Platt, J., Barr, A., Fleischer, K.: Elastically deformable models. In: SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques, pp. 205–214. ACM Press, New York, NY, USA (1987). DOI <http://doi.acm.org/10.1145/37401.37427>
 48. Timoner, S.: Compact Representations for Fast Non-rigid Registration of Medical Images. Tech. Rep. MIT-CSAIL-TR-2003-001, Massachusetts Institute of Technology Computer Science and Artificial Intelligence Laboratory (2003)
 49. Wittek, A., Laporte, J., Miller, K., Kikinis, R., Warfield, S.: Computing Reaction Forces on Surgical Tools for Robotic Neurosurgery and Surgical Simulation. In: Australasian Conference on Robotics and Automation 2004, Australia, Australian Robotics & Automation Association, vol. 1 (2004)
 50. Yao, J., Funkenbusch, P.D., Snibbe, J., Maloney, M., Lerner, A.L.: Sensitivities of Medial Meniscal Motion and Deformation to Material Properties of Articular Cartilage, Meniscus and Meniscal Attachments Using Design of Experiments Methods. *Journal of Biomechanical Engineering* **128**(3), 339–408 (2006)
 51. Zauel, R., Yeni, Y.N., Bay, B.K., Dong, X.N., Fyhrie, D.P.: Compression of the Linear Finite Element Prediction of Deformation and Strain of Human Cancellous Bone to 3D Digital Volume Correlation Measurements. *Journal of Biomechanical Engineering* **128**, 1–6 (2006)
 52. Zhu, J.: A posteriori error estimation—the relationship between different procedures. *Computational Methods in Applied Mechanics and Engineering* **150**, 411–422 (1997)