Creating Animatable Non-Conforming Hexahedral Finite Element Facial Soft-Tissue Models for GPU Simulation

Mark Warburton Steve Maddock
Department of Computer Science
The University of Sheffield
211 Portobello Street
Sheffield, S1 4DP, United Kingdom
{ M.Warburton | S.Maddock }@dcs.shef.ac.uk

ABSTRACT

Physically-based animation techniques enable more realistic and accurate animation to be created. Such approaches require the creation of a complex volumetric model that can be realistically and efficiently simulated, particularly for interactive computer graphics applications. We present an approach to automatically construct animatable non-conforming hexahedral finite element (FE) facial soft-tissue simulation models, including automatic determination of element material types, boundary conditions and muscle properties, making them immediately ready for simulation.

Keywords

physically-based modelling, soft-tissue modelling, facial modelling, physically-based animation, finite element method.

1 INTRODUCTION

Facial modelling and animation is one of the most challenging areas of computer graphics. While various techniques have been proposed to create and animate facial models, by using a physically-based model, the effects of muscle contractions can be propagated through the facial soft tissue to deform the model in a more realistic and anatomical manner.

Physics-based soft-tissue simulation systems often focus on either efficiently producing realistic-looking animations for computer graphics applications [TW90, KHS01], or simulating models with high physical accuracy for studying soft-tissue behaviour [BJTM08, HMSH09] or surgical simulation [KRG+02, ZHD06]. Popular simulation techniques include the efficient mass-spring (MS) method [TW90, KHS01], the accurate but computationally complex finite element (FE) method [SNF05, HMSH09], and the FE-based but precomputation-heavy mass-tensor (MT) method [MSNS05, XLZH11]. Physics engines, which focus on performance and stability, can also be used [MHHR06]. Indeed, increases in computational power, and the use of GPU computing architectures mean that complex FE simulations are now possible in real time [TCO08].

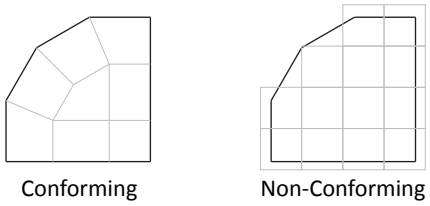

Figure 1: A conforming and non-conforming simulation model.

Physics-based simulations require an appropriate simulation model to be created, for example, using surface meshes of the object to be simulated. As illustrated by Figure 1, such models can either conform to a surface mesh [MBTF03, BJTM08], or a non-conforming model with a bound surface mesh can be used [Coo98, DGW11]. High-quality conforming models that can be efficiently simulated are often difficult and time-consuming to create, although such models are usually required for high-accuracy applications. In contrast, non-conforming models can enable more efficient production of stable, realistic-looking animations for computer graphics applications.

The main aim of this work is to develop an automatic process to easily construct animatable non-conforming hexahedral FE simulation models, including automatic determination of element material types, boundary conditions and muscle properties. While the focus is on creating facial soft-tissue models (the soft tissue be-

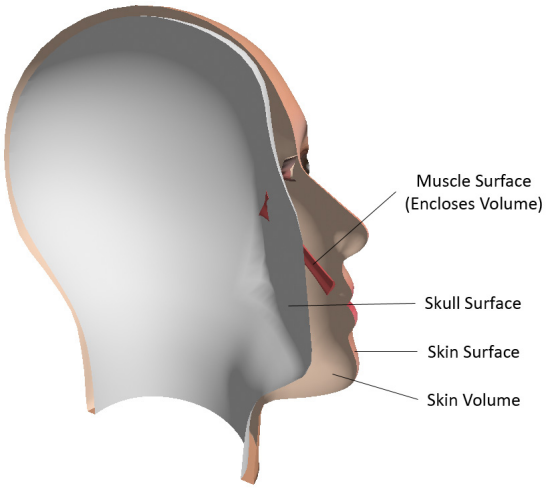

Figure 2: Surfaces and volumes of a facial soft-tissue model. The whole volume between the skull and skin surfaces (i.e. the skin and muscle volumes) is discretised to create an FE facial soft-tissue model.

tween the skull and outer skin surface, as shown by Figure 2), the process can be used to create any multi-layered model from any surface meshes. Such facial models can be used, for example, to efficiently produce realistic-looking facial animations for computer graphics applications. The following sections detail relevant related work, followed by a description and examples of the model creation process, including model simulation examples using our GPU FE system, finishing with a comparison between conforming and non-conforming simulation meshes, particularly for GPU FE simulations.

2 RELATED WORK

2.1 Physically-Based Facial and Soft-Tissue Models

Physically-based facial animation systems for computer graphics applications normally consist of muscle and skin models, sometimes along with a skull model and wrinkle models. For increased realism, a skull model can include a rotatable mandible, which can be geometrically controlled [KHS01] or physically-based [Cou05]. Muscles have been modelled using vectors [Wat87], and more anatomically accurate geometric [KHS01, Cou05] and physics-based volumes [BWL+10, RP07]. Many muscle contraction models are based on a Hill-type model [RP07, LST09], some of which are biologically inspired [HMSH09], and the direction of contraction can be approximated as parallel to the central action curve [TZT09], or, more anatomically, by using a fibre field [SNF05].

Various facial soft-tissue models have been proposed, ranging from simple but efficient physics-engine-based [CP06] and MS models [TW90, KHS01], to more anatomical and realistic FE models [SNF05, BWL+10]. Detailed models of blocks of skin and soft tissue have also been created [KSY08], along with complex soft-tissue constitutive models [Bis06]. Due to its efficiency, the TLED algorithm has been used for various non-linear FE soft-tissue simulations [TCO08], resulting in large speed-ups. The FE-based MT method has also been used to produce such simulations and also for facial surgical applications [MSNS05, XLZH11], showing similar accuracy to the FE method when simulating small displacements.

2.2 Model Creation Approaches

The model creation process is normally difficult and time-consuming, and it is also dependent on the required model structure. To create a model for FE simulation, a suitable mesh must be created, and FE simulation properties, such as boundary conditions, must be set. Regarding model element types for simulation, we only consider linear elements with a single integration point for optimal computational performance.

Simple automatic model creation approaches have been used that just create a layered MS model and skull from a surface mesh [TW90]. On the other hand, CT and MRI scans, or anthropometric data can be used to manually or automatically create an anatomical reference head model [MSNS05, KHS01]. Such data from the Visible Human Dataset[1] has previously been used for reference model creation [SNF05]. Various techniques have been proposed to deform reference skull, muscle or full physically-based head models using manually defined landmarks [KHS01, AZ10], although these often rely on good landmark placement. Kähler et al. also developed an interactive editor to enable easy muscle creation by processing user-specified grid points [KHS01].

Numerous algorithms exist for fast automatic generation of high-quality tetrahedral models that conform to surface meshes [MBTF03, SG05]; however, 4-node tetrahedra are susceptible to volume locking, particularly when simulating incompressible materials like soft tissue. In contrast, reduced-integration 8-node hexahedral elements (with hourglass control) have increased stability and accuracy [WJC+10], particularly when modelling non-linear anisotropic materials [fLhLfT11], and can be used to create meshes using fewer elements, normally outweighing the efficiency of tetrahedra. Hexahedra are therefore often preferred for FE simulations.

Although various algorithms for producing conforming hexahedral meshes have been proposed [SKO+10,

[1] http://www.nlm.nih.gov/research/visible/visible_human.html

ZHB10, NRP11], hexahedral mesh generation is often difficult and time consuming, and, without heavy manual work, many such algorithms suffer problems regarding element quality and robustness, particularly with complex geometries like soft tissue. Techniques have been proposed to improve the quality of hexahedral meshes [ISS09, SZM12], although these can produce models with an increased number of elements.

Simple hexahedral meshes can also be merged to produce a complex mesh [SSLS10, Lo12], although these approaches would require a manual decomposition of the complex model such that high quality elements are able to be produced during the merging process. Similarly, conforming and non-conforming domain decomposition FE methods can be used [Lam09], which involve performing an FE analysis on a model decomposed into several independent subdomains. Some other techniques involve deforming a reference hexahedral mesh [CPL00, fLhLfT11], although such approaches require a high-quality reference mesh, and often also require manual work or modifications to the final mesh.

Alternatively, non-conforming hexahedral meshes are easier to create, for example, using voxelisation techniques, and a surface mesh can also be bound to the volume mesh for visual purposes. Such meshes can be used to create models for more stable and computationally efficient FE simulations [DGW11]. Kumar et al. performed linear elastic FE simulations using structured non-conforming hexahedral grids, and compared these with conforming hexahedral simulation meshes [KPB08], which produced similar stresses, although only relatively simple models were examined. Non-conforming tetrahedral facial and soft-tissue FE models have also been used for stability and performance reasons [Coo98, SNF05], although linear tetrahedral elements can cause problems such as volume locking.

Once an FE simulation mesh has been created, model properties, such as element constitutive properties and boundary conditions, must be specified. Developing on current techniques for producing non-conforming hexahedral meshes with bound surface meshes, our model creation process automatically produces complete simulation-ready multi-layered FE models (i.e. with FE model properties computed) from any surface meshes. Focussing on facial soft-tissue models, this includes automatically assigning material properties to different regions of soft tissue, such as muscle and skin, and determining boundary conditions and muscle properties. With sensible model data organisation, the models can be efficiently simulated on the GPU.

3 NON-CONFORMING HEXAHEDRAL FINITE ELEMENT SOFT-TISSUE MODEL CREATION

A process has been developed to create simulation meshes with hexahedral elements that approximate but don't conform to surface meshes. Such meshes have several advantages over simulation meshes that conform to the shape of the object they are simulating, including advantages relating to mesh creation, stability and computational performance. A comparison between using conforming and non-conforming hexahedral simulation meshes with a CUDA FE system is presented in the discussion in Section 4.3.

A program has been implemented to voxelise a closed surface mesh, that can include closed internal boundaries, described by an OBJ file. Ray-polygon intersection tests for a ray fired from voxel centres are used to determine whether voxels are enclosed by a set of polygons. The enclosed voxels are used as hexahedral elements to produce a hexahedral FE mesh for use with our simulation system.

The surface mesh can contain various surfaces, and elements are separated depending on the user-defined collection of surfaces (volumes) by which they are enclosed, enabling different sets of elements to be assigned different material laws and properties. For example, with a facial mesh, there may be surfaces for the skull, skin and each muscle. The skull and skin surfaces might together define a single 'skin' volume (i.e. the volume of soft tissue between the skull and skin surfaces), whereas the volume-enclosing muscle surfaces could define their own volumes, as illustrated by Figure 2.

Element properties are determined by firstly assigning each volume a *level* and a *priority* within that level, where level 0 is the lowest level, and priority 0 is the highest priority within a level. Semantically, a volume in a higher level is contained within, and bound by, either a single or several volumes in the level immediately below (see Figure 3). Continuing with the facial mesh example, level 0 might consist solely of the skin volume, whereas level 1 might consist of the muscle volumes, each assigned a different priority. Lower-level volumes are voxelised first. Properties of elements are then overwritten if they are also contained within an immediately higher-level volume; hence, properties of elements within the skin volume would be overwritten by those that are also enclosed by a muscle volume. Within a level, higher-priority volumes are voxelised first, although properties of elements within lower-priority volumes are not overwritten.

This level-based voxelisation process is described by Algorithm 1, and illustrated by Figure 3. It can be seen that only lenient requirements are imposed on the

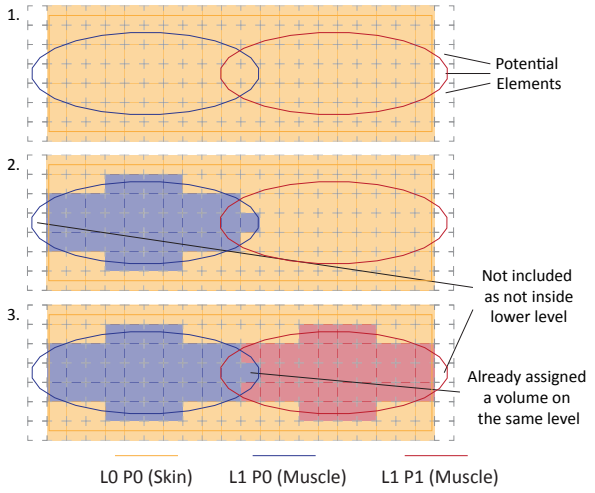

Figure 3: An example of the level-based voxelisation process for a skin block containing two muscles, showing the model state after each level priority has been considered in turn. Note this is a 2D illustration of a 3D process.

```
1:  volLevs ← volume levels
        {array (levels) of arrays (priorities)}
2:  for all voxels, v do
3:      {from lowest to highest level}
4:      while levPris ← volLevs.next and
        levChanged(v) do
5:          lev ← level
6:          {from highest to lowest priority}
7:          while pri ← levPris.next and not
            priChanged(v) do
8:              if insideVolume(v, lev, pri) then
9:                  setVolume(v, lev, pri)
```

Algorithm 1: The voxelisation process, which includes determining the volume by which each element is enclosed.

creation of the surface mesh; for example, as muscles should be contained within the skin volume, parts of muscle surfaces that cross the bounds of this volume are appropriately ignored. Muscle surfaces can therefore simply penetrate the skull (like the muscle in Figure 2), rather than having to 'attach' and conform to the skull surface, therefore simplifying the modelling of surface meshes.

User-specified sections of a surface mesh within a particular bounding box can also be voxelised, enabling easy simulation of only relevant sections of meshes with increased simulation speed, as opposed to simulating the whole mesh. Although distorting element shape, and therefore possibly impacting simulation stability, the hexahedral length, width and height can be independently set, which could be useful, for example, to

```
1:  for all rigid vertices, lines and polygons, r do
2:      for all voxels, v do
3:          c ← centre(v)
4:          p ← getClosestPosition(r, c)
                {p ← r for vertices}
5:          if insideBoundingBox(v, p) then
6:              l ← p − c
7:              i ← getIntersection(v, l)
8:              for all v nodes, n do
9:                  if withinRange(n, i) then
10:                     setRigid(n)
```

Algorithm 2: The process to determine rigid nodes.

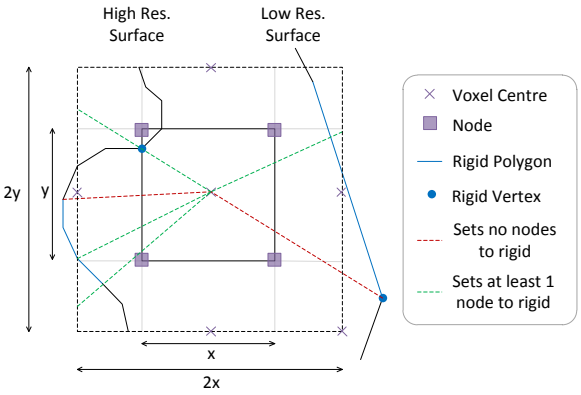

Figure 4: Examples of arbitrarily selected rigid surfaces and nodes inside and outside a voxel bounding box (the dashed box surrounding the voxel). Note this is a 2D illustration of a 3D process.

help improve performance when modelling large, thin materials.

When voxelising a soft-tissue mesh, if surfaces are given names that correctly identify them (e.g. if the names of surfaces enclosing muscles start with the word 'muscle'), an animatable soft-tissue mesh can be automatically created with fixed skull nodes and animatable muscles. Using a collection of user-defined rigid vertices, lines and polygons on the surface mesh, an attachment process, described by Algorithm 2, has been developed to determine rigid simulation nodes, such as fixed skull nodes on a facial model.

Figure 4 illustrates the bounding box test in line 5 of the attachment algorithm, which is successful if position p (a rigid vertex position, or the closest point on a rigid line or polygon to the voxel centre) lies within a box, centred at voxel centre c, that has sides twice the length of voxel v in each direction. The test in line 9, to determine which nodes to set as rigid depending on the location of intersection point i with respect to the nodes of the voxel face on which it lies, is illustrated by Figure 5. For each node n, if i lies within a box starting at n that has sides 0.75x the length of v in each direction, n is set to rigid. The box side lengths could be changed

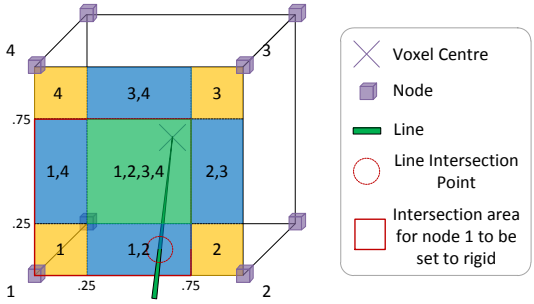

Figure 5: An indication of which nodes are set to rigid depending on the location of the line intersection point on the voxel face, indicated by the node indices inside the various sections. An example line is shown, which would set nodes 1 and 2 to rigid.

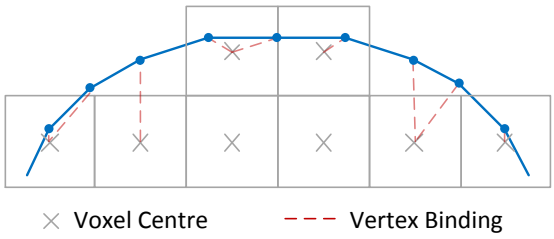

Figure 6: Vertices of a surface mesh bound to the closest elements. Note this is a 2D illustration of a 3D process.

to make the algorithm more or less 'strict'; for example, with side lengths of 0.5x, only one node would be set to rigid for each rigid polygon, line and node. However, 0.75x seems to work well and produce the desired results with the current examples.

The attachment algorithm can also handle cases when the simulation mesh is much higher or lower resolution than the surface mesh, as shown by Figure 4. After attaching skull nodes, each muscle can then be assigned a contraction centre, which is calculated as the average of the muscle nodal positions that lie on the simulation skull.

After simulation model creation, as with Dick et al.'s simulation approach [DGW11], the vertices of the surface mesh can be simply bound to and animated with elements of the FE mesh using trilinear interpolation and extrapolation (see Figure 6). A position, $\mathbf{p}$, is bound to the closest element (determined by a distance test from the element centres) using three weights, w_i - one along each local axis, i, from the first node of the element, $\mathbf{x}$. For a simple voxel element at its rest position and aligned with the global axes, these can be easily calculated:

$$w_i = 1 - \frac{p_i - x_i}{v_i} \quad (1)$$

where v_i is the voxel dimension in the i^{th} direction.

Using these weights, trilinear interpolation or extrapolation (if any weight is negative) can be easily applied before rendering the surface mesh.

4 EXAMPLES AND RESULTS

4.1 Model Creation Examples

Various simulation models of varying complexity have been automatically created using the described technique, which include a sphere, cylindrical muscle, skin block with an inclined relatively flat cylindrical muscle, and simple facial model with a simple left zygomatic major muscle. The facial surface, including an inner surface used as an approximation of the skull, was generated using FaceGen[2]. As the face model includes only a single muscle, only the relevant section of the face containing this muscle was voxelised. Figure 7 shows the simulation meshes, and also their rigid nodes by which they were constrained during simulations, and the muscle contraction centres (except for the sphere model which contains no muscles). Figure 8 better shows the surface models used to create the skin block and facial models, and Figure 9 shows the voxelisations of the skin and muscles within these models.

The model creation process generally produced desirable results, for example, with the desired facial model nodes set to rigid on a complex surface, and muscle elements correctly determined. The muscle part of the skin block simulation model, however, shows the importance of using a high enough resolution to capture the surface model shape well. With a too low resolution, surfaces that cross voxel boundaries at small angles may not be captured by the simulation model.

4.2 Model Simulations

We have implemented a non-linear total Lagrangian explicit dynamic (TLED) FE solver on the GPU using the Fermi CUDA architecture to efficiently perform accurate FE simulations. The details of the system, and the formulation and presentation of the TLED algorithm are beyond the scope of this paper, although the TLED algorithm has been presented by Miller et al. [MJLW07].

Various simulations have been performed using the TLED FE system with the models presented in Section 4.1 - a sphere of soft-tissue material, cylindrical muscle, skin block with a muscle, and simple facial model with a single muscle. Each model is composed of reduced-integration 8-node hexahedral elements, and uses a single Neo-Hookean material (for both passive tissue and muscle where applicable) with a Young's modulus of $3000Pa$ and a Poisson ratio of 0.49. A time-step of $0.1ms$ was used for each simulation. Such a small time-step was necessary for stability due

[2] http://facegen.com/

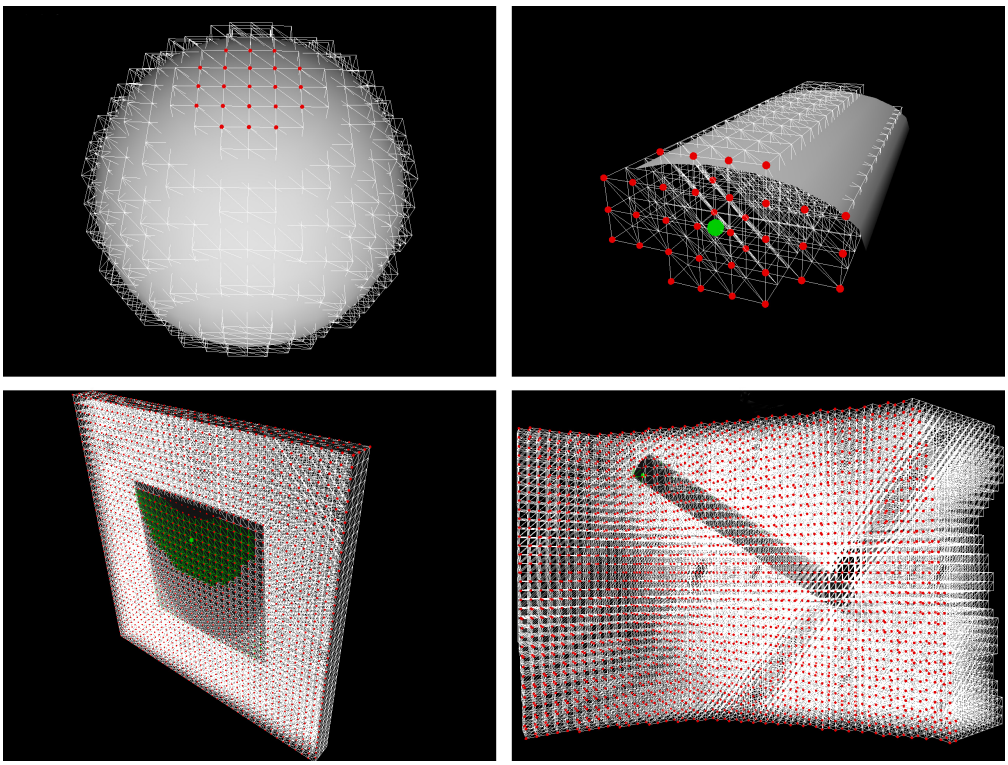

Figure 7: Model rigid node attachments for, from top left to bottom right, a sphere, cylindrical muscle, skin block with muscle, and face, represented by red spheres. The green spheres are muscle contraction centres (where applicable), and muscle elements are coloured green for the skin block and face models.

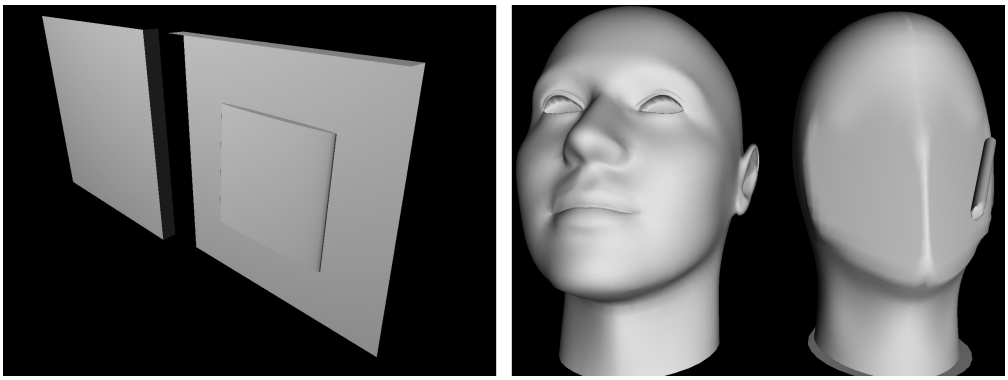

Figure 8: Surface models for the skin block model (left) and facial model (right). Within each image, the left-hand side shows the skin surface, while the right-hand side shows the rigid surfaces to which the muscles are attached.

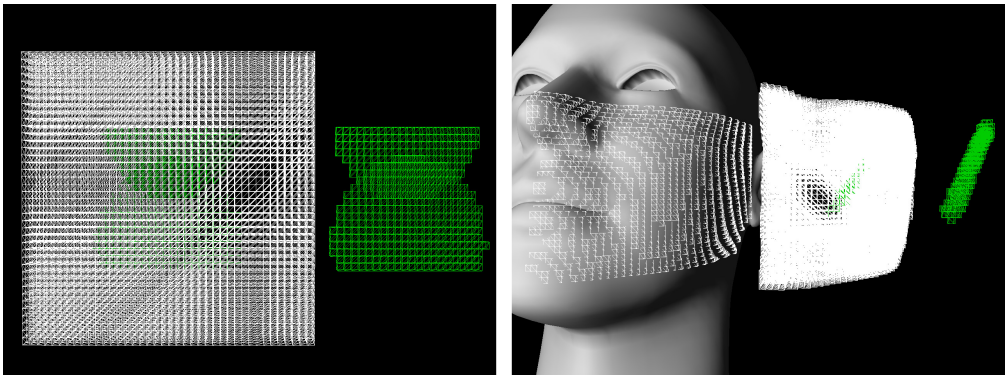

Figure 9: Model voxelisations for the skin block model (left) and facial model (right), where muscle elements are coloured green.

Detail	Sphere	Muscle	Block	Face
Nodes	2849	480	6724	15516
Elements	2176	280	4800	12267
Element Length (*mm*)	12.5	1.5	1.5	2.5
Muscle Contraction	N/A	0.25	0.6	0.4
Time-Step Solution Time (*ms*)	0.187	0.205	0.342	0.542
FPS	34	38	15	7

Table 1: Some model and simulation statistics of simulations performed on a NVIDIA GeForce GTX 460 1GB GPU.

to the small, highly incompressible elements used. The sphere model was influenced by external forces, such as gravity or from user interaction, whereas the other models were influenced purely by internal active muscle stresses. Table 1 shows data on the structure of each model, as well as some simulation statistics, and Figure 10 shows the equilibrium positions for the sphere and face simulations.

Simulations perform well under gravity and user interaction, although it is likely that the face model will be simulated a lot better with more accurate surface models and constitutive laws. For example, it has been shown that it is necessary to model the inhomogeneity and anisotropy of skin to simulate any decent wrinkling effects [HMSH09], rather than as a single isotropic material, as with this work. With the independent muscle simulation, it was necessary to use a small contraction value as there is no surrounding soft-tissue to offer resistance against the muscle active stresses.

4.3 Comparison of Conforming and Non-Conforming Simulation Meshes

For our work, non-conforming hexahedral FE simulation meshes have been used. However, simulation meshes that conform to surface meshes can also be used for FE simulations. Table 2 summarises the creation and use of conforming and non-conforming hexahedral meshes with a CUDA TLED FE system.

Experiments using a conforming (created using IA-FEMesh[3]) and a non-conforming hexahedral simulation mesh for a sphere have demonstrated the performance and stability advantages of non-conforming simulation meshes. As shown by Table 3, using a non-conforming simulation mesh with a bound higher resolution surface mesh has led to performance increases of almost 2x compared to using a conforming simulation mesh with roughly the same number of nodes and elements. Also, stable simulations were able to be performed using a considerably higher Young's modulus and Poisson ratio with a non-conforming mesh, which is necessary for simulating, for example, the stiff properties of the epidermal skin layer, and incompressible soft-tissue material. On the downside, depending on element size, accuracy is likely to be reduced using a non-conforming simulation mesh.

[3] `https://simtk.org/home/ia-femesh`

5 CONCLUSIONS

This work has presented a process for creating animatable non-conforming hexahedral FE simulation models to which the object surface meshes are bound. Using regular or irregular voxel sizes, the process involves discretising the volume enclosed by surface meshes, possibly with internal surfaces, and separating elements accordingly. Rigid nodes on the simulation mesh can also be automatically determined, as can muscle parameters. Some soft-tissue models, including a simple facial model with a muscle, have been created to demonstrate the process, which could also be used to produce models for more complex surface meshes. A GPU TLED FE system has been used to produce some example real-time and interactive simulations with the created models, showing that a surface mesh can be easily transformed into a complex animatable physically-based model. Finally, the various advantages provided by using a non-conforming simulation mesh have been outlined, including model creation, speed, memory and stability advantages.

Various improvements could currently be made to the model creation process. For example, the voxelisation process could be based on the the proportion of a voxel that is enclosed by a volume, which could be used to determine whether a voxel is inside the volume, and to calculate the element material properties, similar to the approach by Lee et al. [LST09]. Using such an approach would also require the rigid node attachment algorithm to be modified. Future work will also focus on creating and animating a more accurate full facial model with inhomogeneous anisotropic viscoelsatic materials.

6 REFERENCES

[AZ10] Olusola O. Aina and Jian Jun Zhang. Automatic Muscle Generation for Physically-Based Facial Animation. In *SIGGRAPH 2010 Posters*, pages 105:1–105:1, Los Angeles, California, USA, 2010. ACM.

[Bis06] J. E. Bischoff. Reduced Parameter Formulation for Incorporating Fiber Level Viscoelasticity into Tissue Level Biomechanical Models. *Ann. Biomed. Eng.*, 34(7):1164–1172, 2006.

[BJTM08] Giuseppe Barbarino, Mahmood Jabareen, Juergen Trzewik, and Edoardo Mazza. Physically Based Finite Element Model of the Face. In *Proc. ISBMS 2008*, pages 1–10, London, UK, 2008. Springer-Verlag.

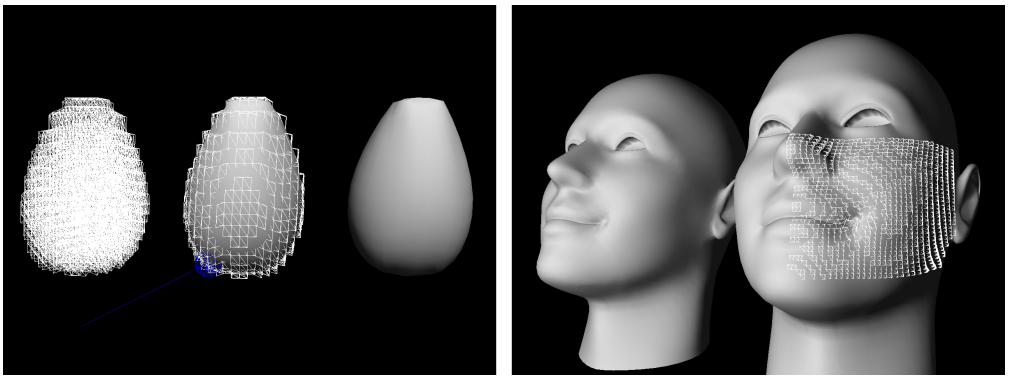

Figure 10: Simulation equilibrium states for sphere under gravity and interaction (where the blue line represents the interactive force), and a face with a left zygomatic major muscle.

Criterion	Conforming	Non-Conforming
Volumetric Model Generation	Complex, involving manual input	Simple and automatic
Simulation Speed	Difficult to achieve speed-ups	More efficient memory accesses and global memory coalescing can be achieved for increased speed
Memory Requirements	High, especially with large models	Low as some values are the same for each element
Stability	Badly-shaped or tiny elements can greatly reduce stability	All elements are regular for better stability
Accuracy	Mesh more closely matches the object being simulated	Less accurate approximation of the object being simulated may lead to less accurate results

Table 2: A summary of the differences between using a conforming and non-conforming regular hexahedral simulation mesh with a CUDA TLED FE system.

Criterion	Conforming	Non-Conforming
Nodes	2736	2849
Elements	2197	2176
Minimum Element Length (mm)	$<$12.5	12.5
Time-Step Solution Time (ms)	0.48	0.246
Stability with a 0.1ms timestep:		
Maximum E (kPa)	3	10
Maximum ν	0.4	0.47

Table 3: Results of a comparison between using a conforming and a non-conforming regular hexahedral simulation mesh of a spherical object, constrained by its upper nodes and acting under gravity, with a Neo-Hookean material model (with a default Young's modulus, E, of 3kPa and Poisson ratio, ν, of 0.4). Simulations were performed on a NVIDIA GeForce GTX 460 1GB GPU.

[BWL+10] Liliana Beldie, Brian Walker, Yongtao Lu, Stephen Richmond, and John Middleton. Finite element modelling of maxillofacial surgery and facial expressions – a preliminary study. *Int. J. Med. Robot. Comput. Assist. Surg.*, 6(4):422–430, 2010.

[Coo98] Lee Cooper. *Physically Based Modelling of Human Limbs*. PhD thesis, University of Sheffield, 1998.

[Cou05] Alasdair D. Coull. *A Physically-Based Muscle and Skin Model for Facial Animation*. PhD thesis, University of Glasgow, UK, 2005.

[CP06] C. Chen and E. C. Prakash. Physically based facial expression synthesizer with performance analysis and GPU-aided simulation. In *Proc. CyberGames 2006*, pages 171–176, Perth, Australia, 2006. Murdoch University.

[CPL00] Béatrice Couteau, Yohan Payan, and Stéphane Lavallée. The mesh-matching algorithm: an automatic 3D mesh generator for finite element structures. *J. Biomech.*, 33(8):1005–1009, 2000.

[DGW11] Christian Dick, Joachim Georgii, and Rudiger Westermann. Simulation Modelling Practice and Theory. *Simul. Model. Pract. Theory*, 19(2):801–816, 2011.

[fLhLfT11] Meng fei Li, Sheng hui Liao, and Ruo feng Tong. Facial hexahedral mesh transferring by volumetric mapping based on harmonic fields. *Comput. Graph.*, 35(1):92–98, 2011.

[HMSH09] A. Hung, K. Mithraratne, M. Sagar, and P. Hunter. Multilayer Soft Tissue Continuum Model: Towards Realistic Simulation of Facial Expressions. volume 54, pages 134–138, Paris, France, 2009.

[ISS09] Yasushi Ito, Alan M. Shih, and Bharat K. Soni. Octree-based reasonable-quality hexahedral mesh generation using a new set of refinement templates. *Int. J. Numer. Methods Eng.*, 77(13):1809–1833, 2009.

[KHS01] Kolja Kähler, Jörg Haber, and Hans-Peter Seidel. Geometry-based Muscle Modeling for Facial Anima-

tion. In *Proc. GI 2001*, pages 37–46, Ottawa, Ontario, Canada, 2001. Canadian Information Processing Society.

[KPB08] Ashok V. Kumar, Sanjeev Padmanabhan, and Ravi Burla. Implicit boundary method for finite element analysis using non-conforming mesh or grid. *Int. J. Numer. Methods Eng.*, 74(9):1421–1447, 2008.

[KRG+02] R. M. Koch, S. H. M. Roth, M. H. Gross, A. P. Zimmermann, and H. F. Sailer. A Framework for Facial Surgery Simulation. In *Proc. SCCG 2002*, pages 33–42, Budmerice, Slovakia, 2002. ACM.

[KSY08] O. Kuwazuru, J. Saothong, and N. Yoshikawa. Mechanical approach to aging and wrinkling of human facial skin based on the multistage buckling theory. *Med. Eng. & Phys.*, 30(4):516–522, 2008.

[Lam09] B. P. Lamichhane. Mortar Finite Elements for Coupling Compressible and Nearly Incompressible Materials in Elasticity. *Int. J. Num. Anal. Model.*, 6(2):177–192, 2009.

[Lo12] S. H. Lo. Automatic merging of hexahedral meshes. *Finite Elem. Anal. Des.*, 55:7–22, 2012.

[LST09] Sung-Hee Lee, Eftychios Sifakis, and Demetri Terzopoulos. Comprehensive Biomechanical Modeling and Simulation of the Upper Body. *ACM Trans. Graph.*, 28(4):99:1–99:17, 2009.

[MBTF03] Neil Molino, Robert Bridson, Joseph Teran, and Ronald Fedkiw. A Crystalline, Red Green Strategy for Meshing Highly Deformable Objects with Tetrahedra. In *Proc. IMR12*, pages 103–114, Santa Fe, New Mexico, USA, 2003. Sandia National Laboratories.

[MHHR06] Matthias Müller, Bruno Heidelberger, Marcus Hennix, and John Ratcliff. Position Based Dynamics. In *Proc. VRIPHYS 2006*, pages 71–80, Madrid, Spain, 2006.

[MJLW07] Karol Miller, Grand Joldes, Dane Lance, and Adam Wittek. Total Lagrangian explicit dynamics finite element algorithm for computing soft tissue deformation. *Commun. Numer. Methods Eng.*, 23(2):121–134, 2007.

[MSNS05] Wouter Mollemans, Filip Schutyser, Nasser Nadjmi, and Paul Suetens. Very fast soft tissue predictions with mass tensor model for maxillofacial surgery planning systems. In *Proc. CARS 2005*, pages 491–496, Berlin, Germany, 2005. Elsevier.

[NRP11] M. Nieser, U. Reitebuch, and K. Polthier. CUBECOVER – Parameterization of 3D Volumes. *Comp. Graph. Forum*, 30(5):1397–1406, 2011.

[RP07] Oliver Röhrle and Andrew J. Pullan. Three-dimensional finite element modelling of muscle forces during mastication. *J. Biomech.*, 40(15):3363–3372, 2007.

[SG05] Hang Si and Klaus Gärtner. Meshing Piecewise Linear Complexes by Constrained Delaunay Tetrahedralizations. In *Proc. 14th Int. Meshing Roundtable*, pages 147–163, San Diego, California, USA, 2005. Sandia National Laboratories.

[SKO+10] Matthew L. Staten, Robert A. Kerr, Steven J. Owen, Ted D. Blacker, Marco Stupazzini, and Kenji Shimada. Unconstrained plastering – Hexahedral mesh generation via advancing-front geometry decomposition. *Int. J. Numer. Methods Eng.*, 81(2):135–171, 2010.

[SNF05] Eftychios Sifakis, Igor Neverov, and Ronald Fedkiw. Automatic Determination of Facial Muscle Activations from Sparse Motion Capture Marker Data. *ACM Trans. Graph.*, 24(3):417–425, 2005.

[SSLS10] Matthew L. Staten, Jason F. Shepherd, Franck Ledoux, and Kenji Shimada. Hexahedral Mesh Matching: Converting non-conforming hexahedral-to-hexahedral interfaces into conforming interfaces. *Int. J. Numer. Methods Eng.*, 82(12):1475–1509, 2010.

[SZM12] Lu Sun, Guoqun Zhao, and Xinwu Ma. Quality improvement methods for hexahedral element meshes adaptively generated using grid-based algorithm. *Int. J. Numer. Methods Eng.*, 89(6):726–761, 2012.

[TCO08] Zeike A. Taylor, Mario Cheng, , and Sébastien Ourselin. High-Speed Nonlinear Finite Element Analysis for Surgical Simulation Using Graphics Processing Units. *IEEE Trans. Med. Imaging*, 27(5):650–663, 2008.

[TW90] Demetri Terzopoulos and Keith Waters. Physically-Based Facial Modeling, Analysis, and Animation. *J. Vis. Comput. Animat.*, 1(2):73–80, 1990.

[TZT09] C. Y. Tang, G. Zhang, and C. P. Tsui. A 3D skeletal muscle model coupled with active contraction of muscle fibres and hyperelastic behaviour. *J. Biomech.*, 42:865–872, 2009.

[Wat87] Keith Waters. A muscle model for animation three-dimensional facial expression. *SIGGRAPH Comput. Graph.*, 21(4):17–24, 1987.

[WJC+10] Adam Wittek, Grand Joldes, Mathieu Couton, Simon K. Warfield, and Karol Miller. Patient-specific non-linear finite element modelling for predicting soft organ deformation in real-time; Application to non-rigid neuroimage registration. *Prog. Biophys. Mol. Biol.*, 103(2–3):292–303, 2010.

[XLZH11] Shaoping Xu, Xiaoping P. Liu, Hua Zhang, and Linyan Hu. A Nonlinear Viscoelastic Tensor-Mass Visual Model for Surgery Simulation. *IEEE Trans. Instrum. Meas.*, 60(1):14–20, 2011.

[ZHB10] Yongjie Zhang, Thomas J. R. Hughes, and Chandrajit L. Bajaj. An Automatic 3D Mesh Generation Method for Domains with Multiple Materials. *Comput. Methods Appl. Mech. Eng.*, 199(5–8):405–415, 2010.

[ZHD06] Stefan Zachow, Hans-Christian Hege, and Peter Deuflhard. Computer-Assisted Planning in Cranio-Maxillofacial Surgery. *J. Comp. Inf. Technol.*, 14(1):53–64, 2006.