

NURBS-Based Galerkin Method and Application to Skeletal Muscle Modeling

Xianlian Zhou¹
Department of Mechanical and
Industrial Engineering
Center for Computer Aided Design
University of Iowa

Jia Lu²
Department of Mechanical and
Industrial Engineering
Center for Computer Aided Design
University of Iowa

Abstract

Non-Uniform Rational B-spline (NURBS) is often used to construct the free-form boundary representation of three-dimensional objects. In this paper, we propose a method for mechanical analysis for deformable bodies by combining NURBS geometric representation and the Galerkin method. The NURBS surface bounding a 3D body is extended to a trivariate NURBS solid by adding another parametric domain represented by additional control points. The displacement field of the body is constructed using the NURBS shape representation with the control point being the generalized coordinates. The interpolated displacement field is directly used to facilitate finite element formulation. In this manner, traditional FEM meshing is not required. In this work, the NURBS-FEM is applied to skeletal muscle modeling. Muscle is modeled as anisotropic, active hyperelastic solids. The directions of the contractile fibers can be uniform or along the tangent direction of NURBS curves. Typical contractive motions of isolated muscle are simulated.

Categories and Subject Descriptors: I.3.5 [Computer Graphics] Computational Geometry and Object Modeling

Keywords: NURBS solid, finite element method, physically based deformable body modeling.

1 Introduction

The aim of this work is to propose a method suitable for biomechanical modeling of human skeletal muscles in an interactive simulation environment. The direct application is digital human modeling, where the muscle analysis interacts with other modules such as skeletal motion prediction, comfort assessment, etc. The nature of the problem requires that the model to be physically realistic, and efficient to allow fast computation. By “physically realistic”, we mean that the shape and mechanical property of the muscle are accurately represented, and stress and deformation are predicted using the laws of mechanics. The standard finite element method could be used for biomechanical

¹ email: xianzhou@engineering.uiowa.edu

² Corresponding author. email: jia-lu@uiowa.edu

analysis of muscles; however, a full-scale FEM muscle model could result in large numbers of degree-of-freedom and could be in some sense unnecessarily complicated. The challenge therefore is to balance the needs of physical realism and computational efficiency. Indeed, the same issue is frequently encountered in many other applications involving physical modeling and visualization of deformable bodies. In this paper, we propose a NURBS-based Galerkin method in which the muscle geometry is parameterized by NURBS primitives with reasonable small numbers of degree-of-freedom, and the free-form representation is directly used to facilitate mechanical analysis. The method by itself is sufficiently general, and applicable to general deformable bodies. Being essentially a finite element method, the formulation retains most the desirable properties of the latter. In particular, the treatment of material constitutive model is well separated from other formulation aspect, thus allowing for realistic material models being easily used in the analysis.

Over the past decades, physically based deformable modeling has been an interest for researchers in computer graphics and other technical fields. Several major methods have been proposed [Gibson and Mirtich 1997]: mass-spring models, finite element methods, finite volume models [Teran et al. 2003] and other low degree approximated continuum models. As an example of deformable objects, muscle is modeled with various approaches. Porcher-Nedel and Thalmann [1998] used an action line to represent the force produced by the muscle, and a surface mesh of muscle as a mass-spring network that can be deformed under applied force. Ng-Thow-Hing [2000; 2002] used the trivariate B-spline solid to model individual muscle in animals and humans. Muscular deformation is also modeled by the embedding a mass-spring-damper network defined in the B-spline solid. The mass-spring model is simple and fast, but less accurate. A recent trend is to use FEM to simulate muscle behavior based on the realistic muscle shape and pointwise stress-strain relations. Chen and Zeltzer [1992] developed a finite element model of skeletal muscle to simulate muscle forces with Zajac’s [1986; 1989] dimensionless biomechanical model of muscle. The low DOF prismatic bounding box of muscle shape is used as the mesh for finite element simulation. The resulted muscle deformations were visualized by free-form deformation [Sederberg and Parry 1986] defined by the mesh. Teran et al. [2003] used finite volume method to perform rigorous large deformation analysis, where tetrahedral meshes of the biceps and triceps are generated from Visible Human Data Set [U.S. National Library of Medicine 1994].

There are essentially two issues in deformable body modeling: how to parameterize a body and how to bring in physical behavior to the system. B-Splines and NURBS have been widely used to parameterize geometric objects [de Boor 2001; Pieggl and Tiller 1997]. Many researchers have explored ways to couple the geometric representation to physical modeling. One of the

seminal works is the Dynamic NURBS (D-NURBS) method developed by Terzopoulos and Qin [1994]. In D-NURBS, the NURBS control points and the weights are used as generalized coordinates, and the dynamical equations are derived from the Hamiltonian principle. The deformation of a NURBS body is described by displacing the control points and alerting the weights. Hollig [2003] developed finite element formulations using weighted extended B-splines as basis functions to solve boundary value problems. The FEM formulation utilizes a regular grid embedding the domain. To handle essential boundary condition, the B-spline basis is multiplied with weight function that vanishes on the boundary. de Boor [2001] used B-splines to solve boundary value problems by collocation. From the standpoint of computational mechanics, the B-splines finite element falls into the category of the meshfree methods [Belytschko et al. 1996; Li and Liu 2002; Liu et al. 1996], in which globally smooth basis functions constructed from piecewise least square fitting or other local smoothing schemes are used in place of the classical finite element interpolation functions.

The present contribution is built upon the following existing ideas: representing a 3D body by trivariate NURBS solids [Hoschek et al. 1993; Ma et al. 2001; Ng-Thow-Hing and Fiume 2002] and in particular, describing deformation of the body by moving the control points [Sederberg and Parry 1986]; deriving the discrete governing equation directly in terms of NURBS control points. Specifically, the NURBS solid is defined by augmenting the surface representations that typically comes with solid modeling with a third parametric direction that extends to the interior of the body [Ng-Thow-Hing and Fiume 2002]. The resulting representation is directly used in the weak form of the equilibrium equation, resulting in a set of nonlinear algebraic equations for the position of the control points. The method differs from the B-spline finite element method [Hollig 2003] in that both the deformed and undeformed bodies are represented by NURBS, and hence, the formulation is essentially an isoparametric finite element with the NURBS function being the basis functions. The formulation bears some similarities with the meshless method [Belytschko et al. 1996; Li and Liu 2002; Liu et al. 1996] in that the shape functions do not have the δ -property and hence, the “nodal” parameters, namely the control positions, are not the nodal values of the displacements. The NURBS control points are *generalized nodes* not necessarily lying in the domain. Nevertheless, they carry a clear geometric connotation since they directly determine the geometry of the body. Unlike meshfree method, there is an underlying mesh in the parametric space which can be readily used for numerical integration.

The method is ideally suited to fast simulation of muscle response. The major muscles in human limbs are bodies bounded by smooth surfaces, and the surface shapes can be obtained from medical imaging data, e.g. the Visible Human Data Set [U.S. National Library of Medicine 1994]. The bounding surface of muscle can be readily converted into NURBS form, and hence, the whole body can be parameterized by reasonably small numbers of degree-of-freedom. On the other hand, muscle’s mechanical response is extremely complicated. Over simplification of biomechanical aspect in a muscle model could lead to unacceptable error. The proposed formulation, like any Galerkin based method, allows one to implement highly realistic constitutive equations for muscles with minimal additional efforts. In this work, muscles are modeled as hyperelastic, anisotropic material embedded with active contractile fibers.

The paper is organized as follows. Section 2 contains a brief review of trivariate NURBS solids and how a solid model can be extended from NURBS surface. In section 3, the NURBS-FEM formulation is discussed in the context of 3D nonlinear elasticity. Section 4 provides an account of the muscle model used in the paper. Examples of numerical simulation of muscle motion are presented in Section 5.

2 Trivariate NURBS Solid

2.1 Trivariate NURBS Solids

A NURBS solid representation is the generalization of NURBS representation of curves and surfaces. It defines not only the surface boundary of an object but also its interior. Within this approach the position of a generic point in the solid is defined by [Hoschek et al. 1993; Ma et al. 2001]

$$\mathbf{P}(u, v, w) = \frac{\sum_{i=0}^n \sum_{j=0}^m \sum_{k=0}^l \mathbf{P}_{i,j,k} W_{i,j,k} N_{i,p}(u) N_{j,q}(v) N_{k,r}(w)}{\sum_{i=0}^n \sum_{j=0}^m \sum_{k=0}^l W_{i,j,k} N_{i,p}(u) N_{j,q}(v) N_{k,r}(w)} \quad (2.1)$$

where $\mathbf{P}_{i,j,k}$ are the position of the control points, $W_{i,j,k}$ are the weights associated with the control points, and $N_{i,p}, N_{j,q}, N_{k,r}$ are the B-spline basis functions. The parametric variables $\{u, v, w\}$ are defined on three non-decreasing sets of knots

$$\{u_0, u_1, \dots, u_{n+p+1}\}, \{v_0, v_1, \dots, v_{m+q+1}\}, \{w_0, w_1, \dots, w_{l+r+1}\}$$

The basis functions for parameter u are defined recursively as

$$N_{i,k}(u) = \frac{u - u_i}{u_{i+k-1} - u_i} N_{i,k-1}(u) + \frac{u_{i+k} - u}{u_{i+k} - u_{i+1}} N_{i+1,k-1}(u)$$

where

$$N_{i,0}(u) = \begin{cases} 1 & \text{for } u_i \leq u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

The same applies to the basis functions for parameter v and w . The products $N_{i,p}(u)N_{j,q}(v)N_{k,r}(w)$ are the trivariate tensor-product b-spline basis functions defined by the knot sequences, and p, q, r are the orders of the b-spline in each of the parametric variables.

Introducing the piecewise rational *shape functions*

$$R_{i,j,k}(u, v, w) = \frac{W_{i,j,k} N_{i,p}(u) N_{j,q}(v) N_{k,r}(w)}{\sum_{i=0}^n \sum_{j=0}^m \sum_{k=0}^l W_{i,j,k} N_{i,p}(u) N_{j,q}(v) N_{k,r}(w)}$$

the NURBS solid representation (2.1) can also be written as

$$\mathbf{P}(u, v, w) = \sum_{i=0}^n \sum_{j=0}^m \sum_{k=0}^l \mathbf{P}_{i,j,k} R_{i,j,k}(u, v, w) \quad (2.2)$$

Some important properties of the functions $R_{i,j,k}(u, v, w)$, which are the cornerstone for the success of Galerkin formulation, are summarized below.

1. (Partition of unity): The shape functions $R_{i,j,k}(u, v, w)$ satisfy

$$\sum_{i=0}^n \sum_{j=0}^m \sum_{k=0}^l R_{i,j,k}(u, v, w) = 1$$

for all $u \in [u_p, u_{n+1}]$, $v \in [v_q, v_{m+1}]$ and $w \in [w_r, w_{l+1}]$.

2. (Local support): $R_{i,j,k}(u, v, w) = 0$ if

$$(u, v, w) \notin [u_i, u_{i+p+1}] \otimes [v_j, v_{j+q+1}] \otimes [w_k, w_{k+r+1}].$$

3. In any given domain

$$[u_{i_0}, u_{i_0+1}] \otimes [v_{j_0}, v_{j_0+1}] \otimes [w_{k_0}, w_{k_0+1}],$$

there are at most $(p+1)(q+1)(r+1)$ shape functions which are nonzero. In particular, the shape function $R_{i,j,k}(u, v, w)$ is nonzero for $i_0 - p \leq i \leq i_0$, $j_0 - q \leq j \leq j_0$ and $k_0 - r \leq k \leq k_0$.

Property 1 ensures that the isoparametric representation used in this work is affine invariant, namely, an affine transformation is applied to the volume by applying to the control points. In other words, a linear displacement field can be exactly reproduced. This is a necessary requirement for the convergence of the numerical solution. Property 2 implies that the ensuing finite element equations are sparse.

2.2 Constructing NURBS Solid from Surface

In computer graphics, a 3D solid body is typically represented by its bounding surfaces, given for example by tensor-product NURBS surface

$$\mathbf{P}(u, v) = \sum_{i=0}^n \sum_{j=0}^m \mathbf{P}_{i,j} R_{i,j}(u, v)$$

However, the boundary surface does not contain information about the interior the body. A NURBS solid represented by equation (2.2) can be obtained by adding another parametric domain W and corresponding new control points. Some of commonly used method for solids construction are described in [Hoschek et al. 1993; Ma et al. 2001]. A particular parameterization by Ng-Thow-Hing and Fiume [2002] for a bounded body, where the third parameter represents the radius direction in the intuitive sense, is used in this paper. In general, the methods used in surface construction, such as ruling, sweeping and swing, can be applied to solid model construction. In addition, a method called shrinking [Ma et al. 2001] has been used frequently when dealing with a close or periodic NURBS surface. A NURBS solid can be constructed by shrinking a surface to a point or a curve. For example, a spheroid is derived by shrinking a sphere to its center point, and a cylinder solid is obtained by shrinking a cylinder surface to its centerline. The concept of shrinking naturally corresponds to the parameterization in [Ng-Thow-Hing and Fiume 2002].

As an example, the 3D muscle belly shape shown in figure 1 was obtained by skinning the contour stack extracted from the Visible Human Data Set. The surface is then represented by NURBS. In general, denser contour stack will be used to reconstruct the muscle surface in order to capture better anatomical features.

The NURBS surface, with 56 control points and cubic order, of muscle shape was then extended to NURBS solid by shrinking it into the longitudinal center line for the additional parameter domain. The NURBS solid, which can be seen in figure 6, with 63 control points and 189 DOF, was used as initial muscle geometry in relaxed state.

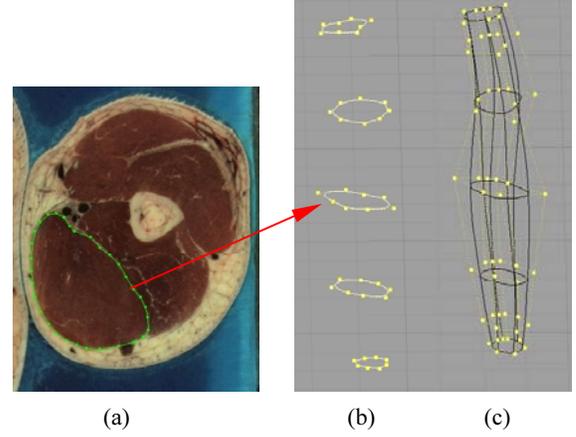


Figure 1. 3D reconstruction of muscle shape with NURBS: (a) Contour of muscle boundary in one slice; (b) Stacks of contours represented by NURBS curves; (c) NURBS surface by skinning the NURBS contours.

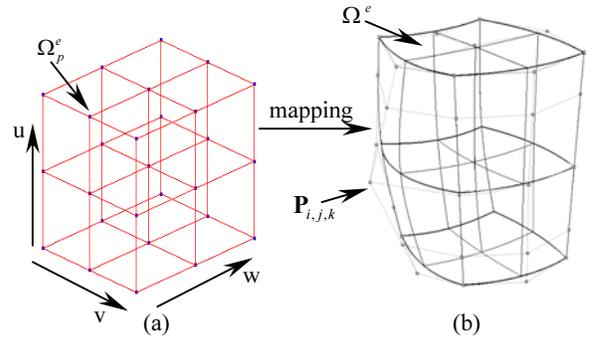


Figure 2. Schematics of NURBS mapping: (a) Parametric domain (Ω_p) and elements (Ω^e); (b) Spatial domain (Ω) and elements (Ω^e).

3 NURBS Based GALERKIN Method

3.1 Isoparametric Mapping

Equation (2.2) is a geometric mapping that maps the parametric domain

$$\{\Omega_p : u \otimes v \otimes w \mid u \in [u_p, u_{n+1}], v \in [v_q, v_{m+1}], w \in [w_r, w_{l+1}]\}$$

to the real geometry. For any point $(u, v, w) \in \Omega_p$, there is a spatial material point in the NURBS solid corresponding to it. In light of this, (u, v, w) is called the parametric coordinates of the spatial material point. If the Jacobian of (2.2) has the same sign throughout the domain, then the mapping is one-to-one, namely for every parametric point there is one and only one corresponding Cartesian point.

In figure 2(a), the cubic represents the entire parametric domain, denoted by Ω_p . The domain is partitioned into a patch of small volumes called parametric elements. The partition of the whole parametric domain into elements is constructed by the tensor

products of knot intervals of u, v and w . In other words, any small domain in the form of the trivariate tensor product $[u_{i_0}, u_{i_0+1}] \otimes [v_{j_0}, v_{j_0+1}] \otimes [w_{k_0}, w_{k_0+1}]$, denoted by Ω_p^e , is an element in the parametric domain Ω_p . In the case where $u_{i_0} \neq u_{i_0+1}, v_{j_0} \neq v_{j_0+1}$ and $w_{k_0} \neq w_{k_0+1}$, the parametric element is non-degenerated. In the present formulation, the parametric elements define the underlying mesh where numerical integration is carried out.

From the property 3 (in Section 2.1), it is easy to show that in any parametric element

$$[u_{i_0}, u_{i_0+1}] \otimes [v_{j_0}, v_{j_0+1}] \otimes [w_{k_0}, w_{k_0+1}],$$

there are at most $nc = (p+1)(q+1)(r+1)$ numbers of control points $\mathbf{P}_{i,j,k}$ ($i_0 - p \leq i \leq i_0, j_0 - q \leq j \leq j_0, k_0 - r \leq k \leq k_0$) influencing on it, namely the corresponding shape functions $R_{i,j,k}(u, v, w)$ are nonzero in that element. Similar to standard finite element method, here $\mathbf{P}_{i,j,k}$ and $R_{i,j,k}(u, v, w)$ associated with the parametric element are treated as ‘nodes’ and ‘shape functions’ of this parametric element. In addition, from the local support property of shape functions, any control point $\mathbf{P}_{i,j,k}$ has limited influence domain

$$(u, v, w) \in [u_i, u_{i+p+1}] \otimes [v_q, v_{q+1}] \otimes [w_k, w_{k+r+1}]$$

which may include at most $(p+1)(q+1)(r+1)$ parametric elements. In other words, the elements in a subdomain $[u_i, u_{i+p+1}] \otimes [v_q, v_{q+1}] \otimes [w_k, w_{k+r+1}]$ will share the control point $\mathbf{P}_{i,j,k}$ as their common node.

For notational simplicity, the triple subscripts (i, j, k) are written into one single index $\alpha = (i \times (m+1) + j) \times (l+1) + k$, so that $\mathbf{P}_\alpha \equiv \mathbf{P}_{i,j,k}$. In a typical patch $[u_{i_0}, u_{i_0+1}] \otimes [v_{j_0}, v_{j_0+1}] \otimes [w_{k_0}, w_{k_0+1}]$, we represent the coordinates of a generic material point in the reference (undeformed) configuration as

$$\mathbf{X}(u, v, w) = \sum_{\alpha} R_{\alpha}(u, v, w) \mathbf{X}_{\alpha} \quad (3.1)$$

where \mathbf{X}_{α} is the coordinates of control point \mathbf{P}_{α} , and the summation is over the controls points which has nonzero contribution. At the same time, the coordinate of the same material point in the current (deformed) configuration is constructed by

$$\mathbf{x}(u, v, w) = \sum_{\alpha} R_{\alpha}(u, v, w) \mathbf{x}_{\alpha} \quad (3.2)$$

where \mathbf{x}_{α} are the coordinates of \mathbf{P}_{α} after deformation. Thus, the displacement field, defined as the difference of the current to the reference position, is given by

$$\mathbf{d}(u, v, w) = \sum_{\alpha} R_{\alpha}(u, v, w) \mathbf{d}_{\alpha} \quad (3.3)$$

where $\mathbf{d}_{\alpha} = \mathbf{x}_{\alpha} - \mathbf{X}_{\alpha}$ is the displacement of the control point \mathbf{P}_{α} , and $\mathbf{d}(u, v, w)$ is the displacement of a spatial point whose parametric coordinates is (u, v, w) .

The central idea of this work is to use (3.1) and (3.2) as isoparametric mapping. With the kinematic mapping in hand, the basic procedure of isoparametric finite element formulation [Belytschko et al. 2002; Zienkiewicz and Taylor 2000] can be applied to NURBS solid. In what follows we describe the formulation in the context of large deformation analysis.

3.2 Large Deformation Formulation

Let Ω_0 be the region that a material body occupies in its reference configuration. Let $\mathbf{x} = \mathbf{x}(\mathbf{X}, t)$ be the motion of the body, which maps Ω_0 into the region Ω the body occupies in the current configuration. The governing equation for a static boundary value problem is expressed as

$$\nabla \boldsymbol{\sigma} + \rho \mathbf{b} = 0 \text{ in domain } \Omega$$

$$\mathbf{d} = \bar{\mathbf{d}}, \text{ on essential (i.e. displacement) boundary } \partial\Omega_{\varphi}$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \bar{\mathbf{t}}, \text{ on natural (i.e. force) boundary } \partial\Omega_f$$

where $\boldsymbol{\sigma}$ is stress tensor, ρ is the current density of the material, \mathbf{b} is the body force, $\bar{\mathbf{t}}$ is the boundary traction, and \mathbf{n} is the normal vector of the natural boundary. In large deformation, the fundamental measure of deformation is the deformation gradient

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} \quad (3.4)$$

The strain at a material point can be described by the right Cauchy-Green deformation tensor $\mathbf{C} = \mathbf{F}^T \mathbf{F}$. The constitutive equation for a nonlinear elastic solid is typically given by an equation $\mathbf{S} = \mathbf{S}(\mathbf{C})$, where \mathbf{S} is the 2nd Piola Kirchhoff stress. If the material is hyperelastic, the constitutive equation is specified with a strain energy density $W = W(\mathbf{C})$, such that

$$\mathbf{S} = 2 \frac{dW}{d\mathbf{C}}.$$

The true (Cauchy) stress $\boldsymbol{\sigma}$ can be obtained as

$$\boldsymbol{\sigma} = \frac{1}{J} \mathbf{F} \mathbf{S} \mathbf{F}^T,$$

where J is the determinant of the deformation gradient.

As a standard procedure in FEM, the governing equation is cast into the weak form

$$\int_{\Omega} \boldsymbol{\sigma} \cdot \mathbf{grad} \boldsymbol{\eta} dv - \int_{\Omega} \rho \mathbf{b} \cdot \boldsymbol{\eta} dv - \int_{\partial\Omega_f} \bar{\mathbf{t}} \cdot \boldsymbol{\eta} da = 0 \quad (3.5)$$

where $\boldsymbol{\eta}$ is any admissible variation of deformation, and ‘grad’ stands for the Eulerian gradient operator. Using the standard finite element notation [Belytschko et al. 2002], the discrete balance equations emanating from (3.5) is written in the form

$$\boldsymbol{\Psi}_{\alpha} = \mathbf{f}_{\alpha} - \int_{\Omega} \mathbf{B}_{\alpha}^T \boldsymbol{\sigma} dv = 0 \quad (3.6)$$

where

$$\mathbf{f}_{\alpha} = \int_{\Omega} R_{\alpha} \rho \mathbf{b} dv + \int_{\partial\Omega_f} R_{\alpha} \bar{\mathbf{t}} da$$

and \mathbf{B}_{α} is the nodal strain-displacement matrix defined by

$$\mathbf{B}_{\alpha} = \begin{bmatrix} \partial_x R_{\alpha} & 0 & 0 \\ 0 & \partial_y R_{\alpha} & 0 \\ 0 & 0 & \partial_z R_{\alpha} \\ \partial_y R_{\alpha} & \partial_x R_{\alpha} & 0 \\ 0 & \partial_z R_{\alpha} & \partial_y R_{\alpha} \\ \partial_z R_{\alpha} & 0 & \partial_x R_{\alpha} \end{bmatrix}$$

Here, $x, y,$ and z are the coordinates in the current configuration. The derivatives, i.e. Eulerian gradient of the shape functions, are computed with the aid of the chain rule

$$\begin{Bmatrix} \partial_x R_\alpha \\ \partial_y R_\alpha \\ \partial_z R_\alpha \end{Bmatrix} = \mathbf{J}^{-T} \begin{Bmatrix} \partial_u R_\alpha \\ \partial_v R_\alpha \\ \partial_w R_\alpha \end{Bmatrix}$$

The Matrix \mathbf{J} is the Jacobian of the geometric mapping (3.2). Using the geometric mapping (3.1) and (3.2), the deformation gradient in (3.4) can be computed according to

$$\mathbf{F} = \sum_\alpha \mathbf{x}_\alpha \otimes \text{Grad } R_\alpha$$

where $\text{Grad } R_\alpha$ is the referential gradient of the shape function, computed again using the chain rule, and “ \otimes ” stands for the standard tensor product. The equation (3.6) is typically solved iteratively, for example using the Newton-Raphson method in which a series of linearized equations are solved. The displacement increment at i^{th} iteration is computed from

$$\Delta \mathbf{d}^{i+1} = - \left[\frac{\partial \Psi}{\partial \mathbf{d}} \right]_{\mathbf{d}=\mathbf{d}^i} \Psi(\mathbf{d}^i) \quad (3.7)$$

The derivative in the bracket gives the so-called tangent stiffness matrix \mathbf{K} . The tangent stiffness consists of two terms,

$$\mathbf{K}_{\alpha\beta} = \int_\Omega \mathbf{B}_\alpha^T \mathbf{D} \mathbf{B}_\beta \, dv + \int_\Omega R_{\alpha,i} \sigma_{ij} B_{\beta,j} \mathbf{I} \, dv = \mathbf{K}_M + \mathbf{K}_G \quad (3.8)$$

where the first term is the material tangent \mathbf{K}_M arising from material non-linearity, in which \mathbf{D} is the matrix form of the material elasticity tensor tangent in the current configuration. The second term \mathbf{K}_G defines a tangent term arising from the non-linear strain-displacement relation and is often called the geometric stiffness matrix. A detailed description of the stress, strain, and tangent moduli will be given in Section 4 with a constitutive model of muscle. An in-depth description of the nonlinear FEM formulation can be found in standard textbooks [Belytschko et al. 2002; Zienkiewicz and Taylor 2000].

3.3 Examples of Deformable Body Modeling

To demonstrate the qualitative behavior of NURBS-FEM, some basic NURBS primitives are deformed under various loading conditions, as shown in figures 3-4. The neo-Hookean material, which will be presented in Section 4, is used for the simulation. In all the examples, we use only first order (linear) shape function for extended parameter (W). This is equivalent to shrinking the boundary surface to its centerline or center point. In general, the order of shape function and number of extended control points can be increased so that the interior interpolation is smoother. Numerically, this will increase the accuracy of finite element computation due to increases “mesh” resolution.

Figure 3 shows a sphere under applied stretch and pinching at its poles. The surface, in quadratic order, is described by 26 non-repeated control points originally. By adding only one center point, a solid sphere obtained with totally 27 control points, and 81 DOF.

Figure 4 shows a torus under out-of-plane shear. Initially, there are 100 non-repeated control points for describing the torus surface with cubic order. And ten control points are duplicated out in order to separate the end surfaces. Another 11 inside control points are added by extending the torus surface to solid. Thus, there are totally 121 control points for the torus solid and 363 DOF.

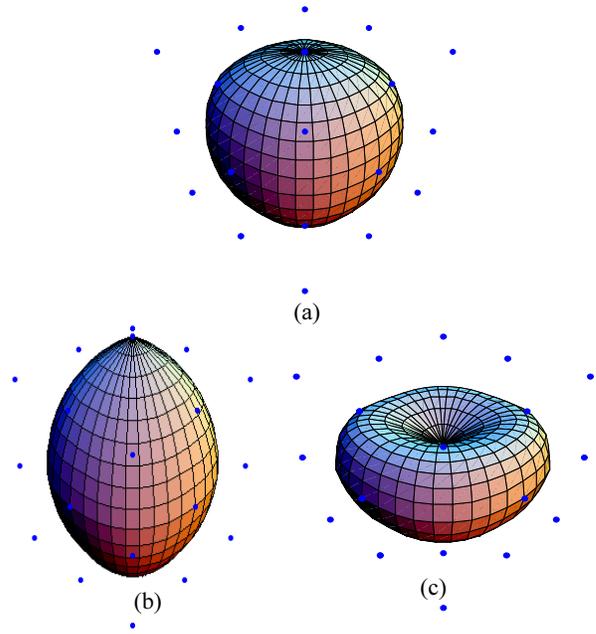


Figure 3. A sphere under tension and pinching:
(a) Initial shape; (b) Tension; (c) Pinching

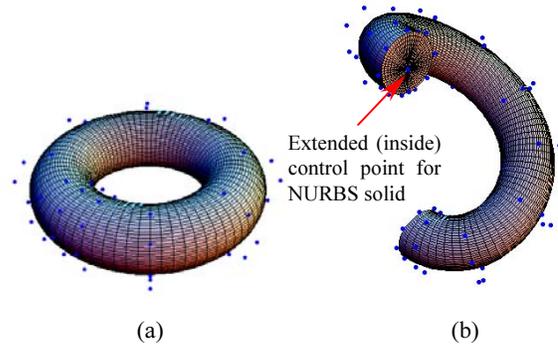


Figure 4. Out-of-plane shear of a torus:
(a) Initial shape; (b) Deformation

4 Skeletal Muscle Modeling

4.1 Constitutive Model

Muscle tissue has a highly complex material behavior: it is active, nonlinear, incompressible, anisotropic, and hyperelastic [Herzog 2000; Lemors et al. 2001; Oomens et al. 2003; Teran et al. 2003]. The microstructure of muscle is determined by the arrangement of the contractile elements (muscle fibers) and passive background tissue within a muscle. Following [Oomens et al. 2003], muscle is modeled as a hyperelastic solid. The strain energy function is assumed to be the sum of two parts

$$W(\mathbf{C}) = W_{matrix}(\mathbf{C}) + W_{fiber}(\lambda, a)$$

where the first part $W_{matrix}(\mathbf{C})$ is the strain energy associated with the passive ground substance, and the second part $W_{fiber}(\lambda, a)$

represents the active and passive muscle fiber strain energy. The ground substance consists of connective tissue, water, etc, and is typically isotropic. In this paper, it is modeled as a neo-Hookean material with an energy form

$$W_{matrix}(\mathbf{C}) = \frac{\mu}{2}(I_1 - 2 \ln J - 3) + \frac{\kappa}{2}(\ln J)^2$$

where $I_1 = tr(\mathbf{C})$ is the first principal invariant of \mathbf{C} , μ, κ are the material constants, and $J = det \mathbf{F}$ is the determinant of the deformation gradient. The constant κ may be best understood as a penalty parameter for incompressibility: nearly incompressibility can be approximately modeled with a large value of κ . A set of assumed parameters $\kappa = 500 \text{ N/cm}^2$ and $\mu = 10 \text{ N/cm}^2$ are used in the simulation.

The muscle fiber strain energy is assumed a form

$$W_{fiber}(\lambda, a) = W_{act}(\lambda, a) + W_{pass}(\lambda)$$

where $W_{act}(\lambda, a)$ is the active strain energy of the muscle fiber and $W_{pass}(\lambda)$ is the passive strain energy of the muscle fiber due to stretch. The active strain energy is a function of the muscle fiber stretch λ and the muscle activation level a , where $\lambda = \sqrt{\mathbf{N} \cdot \mathbf{C} \mathbf{N}}$ and \mathbf{N} is the fiber direction in the reference configuration. The first derivative of $W_{fiber}(\lambda, a)$ with respect to λ is defined as

$$\frac{\partial W_{act}(\lambda, a)}{\partial \lambda} = a \sigma_{max} [1 - 4(\lambda - 1)^2]$$

and

$$\frac{\partial W_{pass}(\lambda)}{\partial \lambda} = \begin{cases} m_1 \lambda^2 (e^{m_2(\lambda^2-1)} - 1), & \lambda > 1 \\ 0, & \lambda \leq 1 \end{cases}$$

where m_1 and m_2 are the coefficients for the passive property of muscle fiber [Oomens et al. 2003]. With these terms, the well-known nonlinear stress-strain relationship of muscle fibers [Zajac

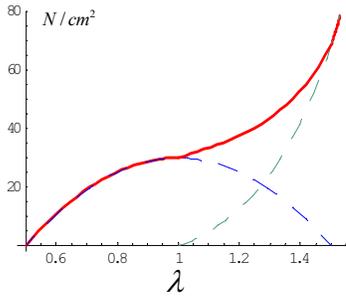


Figure 5. Stress-strain relationship of muscle fiber with $\sigma_{max} = 30 \text{ N/cm}^2$, $m_1 = 35$ and $m_2 = 0.5$

et al. 1986; Zajac 1989], shown in figure 5, can be modeled in the continuum level.

The second Piola Kirchhoff stress in a hyperelastic material is related to the strain by

$$\mathbf{S} = 2 \frac{\partial W}{\partial \mathbf{C}} = 2 \left(\frac{\partial W_{matrix}}{\partial \mathbf{C}} + \frac{\partial W_{fiber}}{\partial \mathbf{C}} \right) = \mathbf{S}_{matrix} + \mathbf{S}_{fiber}$$

where

$$\mathbf{S}_{matrix} = 2 \frac{\partial W_{matrix}}{\partial \mathbf{C}} = \mu [\mathbf{I} - \mathbf{C}^{-1}] + \ln J \mathbf{C}^{-1}$$

Here \mathbf{I} is the second order identity tensor, and

$$\begin{aligned} \mathbf{S}_{fiber} &= 2 \frac{\partial W_{fiber}}{\partial \mathbf{C}} = 2 \left(\frac{\partial W_{act}}{\partial \mathbf{C}} + \frac{\partial W_{pass}}{\partial \mathbf{C}} \right) \\ &= 2 \left(\frac{\partial W_{act}}{\partial \lambda} + \frac{\partial W_{pass}}{\partial \lambda} \right) \frac{\partial \lambda}{\partial \mathbf{C}} = \frac{1}{\lambda} \left(\frac{\partial W_{act}}{\partial \lambda} + \frac{\partial W_{pass}}{\partial \lambda} \right) \mathbf{N} \otimes \mathbf{N} \end{aligned}$$

The Cauchy (true) stress $\boldsymbol{\sigma}$ takes the form

$$\begin{aligned} \boldsymbol{\sigma} &= \frac{1}{J} \mathbf{F} \mathbf{S} \mathbf{F}^T = \frac{1}{J} \mathbf{F} (\mathbf{S}_{matrix} + \mathbf{S}_{fiber}) \mathbf{F}^T \\ &= \frac{1}{J} (\mu (\mathbf{B} - \mathbf{I}) + \kappa \log J \mathbf{I}) + \frac{1}{J} \left(\frac{\partial W_{act}}{\partial \lambda} + \frac{\partial W_{pass}}{\partial \lambda} \right) \mathbf{n} \otimes \mathbf{n} \end{aligned}$$

where $\mathbf{B} = \mathbf{F} \mathbf{F}^T$ is the left Cauchy-Green tensor and \mathbf{n} is the fiber direction in the current configuration. The material elasticity tensor is defined as

$$\mathbb{D} := \frac{\partial \mathbf{S}}{\partial \mathbf{E}}$$

For hyperelastic material specified by a strain energy function W ,

$$\mathbb{D} := 4 \frac{\partial^2 W}{\partial \mathbf{C} \partial \mathbf{C}}, \text{ in component form, } \mathbb{D}_{ABCD} = 4 \frac{\partial^2 W}{\partial C_{AB} \partial C_{CD}}$$

The spatial form of the material tensor, needed for computing the material stiffness matrix, is related to the elasticity tensor \mathbb{D} through the relation

$$\mathbb{C} = \frac{1}{J} \mathbf{F} \mathbb{D} \mathbf{F}^T$$

In components,

$$\mathbb{C}_{abcd} = \frac{1}{J} F_{aA} F_{bB} \mathbb{D}_{ABCD} F_{cC} F_{dD}$$

The spatial tangent matrix \mathbf{D} for the linearized tangent stiffness matrix at equation (3.8) is obtained by transforming the fourth order tensor \mathbb{C} to a second order matrix using Voigt notation [Belytschko et al. 2002]. The details are omitted here.

4.2 Fiber Representation

The fiber arrangement of skeletal muscles can be classified into several categories [Lemors et al. 2001; Ng-Thow-Hing 2000]: parallel-fibered muscles (muscle fibers oriented in parallel to the muscle line-of-action), pennate-fibered muscles (all muscle fibers oriented at the same angle relative to the muscle line-of-action), fusiform muscles (long, fusiform like muscle with fibers attach to the two ends), triangular muscles (fibers radiate from a narrow attachment at one end to a broad attachment at the other end) and others.

In our work, we use constant direction vector \mathbf{N} in the undeformed configuration to model parallel-fibered muscles and unipennate-fibered muscles which have only one distinct fiber direction. In the work of [Ng-Thow-Hing 2000; 2002], B-spline solid models were built to capture muscle architectural details of internal fiber arrangements by carefully fitting the isocurves (two parameter of B-spline solid are held constant while the third varies) of B-spline solid to the real fiber directions in actual muscle specimens. This idea is also used in our work. For non-uniform fiber distribution, the fiber direction at any point is acquired by computing the tangent of an appropriate isocurve through that point.

For example, if the fibers are parallel to the parametric curve $(v, w) = const.$, namely

$$\mathbf{P}(u, v_0, w_0) = \sum_{\alpha} \mathbf{P}_{\alpha} R_{\alpha}(u, v_0, w_0)$$

Taking the derivative of the above equation with respect to u , the normalized tangent vector of this isocurve is expressed as

$$\mathbf{N}(u, v_0, w_0) = \frac{\frac{\partial \mathbf{P}(u, v_0, w_0)}{\partial u}}{\left\| \frac{\partial \mathbf{P}(u, v_0, w_0)}{\partial u} \right\|} = \frac{\sum_{\alpha} \frac{\partial R_{\alpha}(u, v_0, w_0)}{\partial u} \mathbf{P}_{\alpha}}{\left\| \sum_{\alpha} \frac{\partial R_{\alpha}(u, v_0, w_0)}{\partial u} \mathbf{P}_{\alpha} \right\|}$$

At each Gauss integration point, this vector is computed and is used to represent the fiber direction.

5 Simulation of Muscle Stretch And Contraction

Muscle response can be either active or passive. Active muscle generates forces inside the muscle by fiber contraction. The degree of fiber contraction is control by neural input which is mathematically represented by a scalar parameter a in our constitutive model. If this scalar parameter equals to zero, the muscle is fully passive and do not generate any contractive force. There are two distinct types of muscle contraction. The first is isometric contraction, where the muscle contracts or tenses without changing its length; the other is isotonic contraction, where the length of a muscle changes while keeping the contractive force constant. In this Section, the NURBS-FEM muscle model is employed to simulate these typical contractions for an isolated muscle.

Passive Stretch The muscle is treated as totally passive material by setting the activation level $a = 0$. The results for passive stretch are showed in Figure 6.

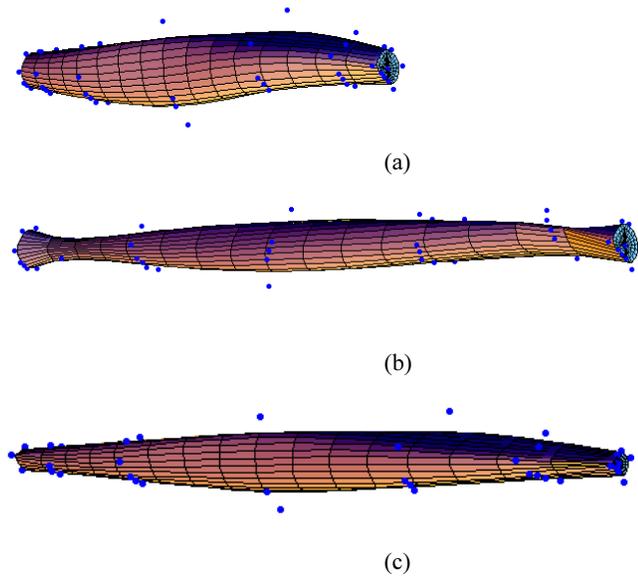


Figure 6. Passive stretch

(a) Initial shape; (b) Deformation with end surfaces shape fixed; (c) Deformation without fixing end surfaces shape.

Isometric Contraction To simulate isometric contraction, the two end surfaces of the muscle solid are fixed to keep the length of muscle unchanged. The initial shape at rest, shown in Figure 7(a), is applied the maximum activation level. The muscle fiber contracts and reaches the new balanced shape of muscle and stress level, as shown in Figure 7(b). Note the stress was calculated iteration by iteration at every gauss point for solving the balance equation (3.6), and the stress elsewhere can be obtained by an extrapolation scheme or calculated exactly by the constitutive model in section 4.

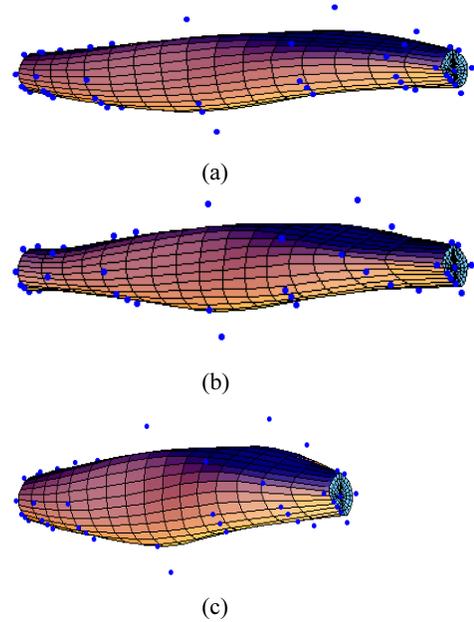


Figure 7. Active contraction (a) Initial shape; (b) Isometric contraction with total length unchanged; (c) Active shortening contraction.

Isotonic Contraction Under the constraint of fixing shape of end surface, the muscle model was loaded for different cases. The static analysis of muscle response is performed with passive

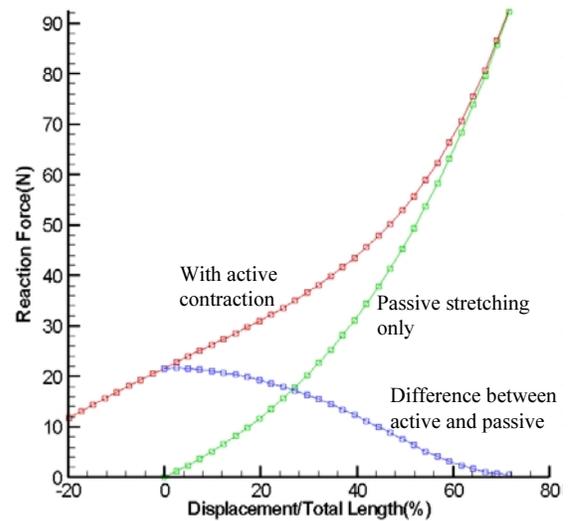


Figure 8. Force-extension curve

stretching, active lengthening contraction and active shortening contraction (Figure 7(c)). For all the cases, the muscle force-extension curves are shown in figure 8 with different length changes. The curves show a behavior that qualitatively agrees what reported in the literature. It can be seen that, with the same reaction force, the length of muscle is different between active and passive case, so the path of muscle shortening from the passive to active case is indeed an isotonic contraction.

In these simulations, the end surfaces of muscles are subject to essential boundary conditions. Like meshfree methods, imposing essential boundary condition is not straightforward since the nodal variables are not the nodal values of displacements. In meshfree computations, a number of methods have been developed to treat the essential boundary conditions. In this work, the transformation method proposed by Chen and Wang [2000] is implemented.

6 Concluding Remarks

This work is the first step toward fast muscle modeling and simulation in an interactive environment [Yang et al. 2004]. We have proposed a NURBS based Galerkin method that combines the NURBS geometric representation with the finite element method. Salient features of the method include: (1) no additional meshing required; (2) better, smoother geometric description with less degree-of-freedom compared to the standard finite element method. Although the method is still at the early stage of development, it has demonstrated some promising features and a great potential for application.

The major limitation of the method at present lies in its inability to handle arbitrary geometry. For instance, muscles composing of multiple branches can not be represented by one single tensor product NURBS solid. Nonetheless, the authors believe that the idea can also be extend to more advanced geometric modeling methods such as subdivision or multiresolution representations where arbitrary topology can be readily handled. Also, at present we only consider isolated muscles. In future, we will develop contact algorithm to handle mutually contacted muscle bundles.

7 Acknowledgements

The research is a part of the Virtual Soldier Research Program funded by the U.S. Army under contract DAAE07-03-D-1003/0001 to the University of Iowa.

References

Belytschko, T., Krongauz, Y., Organ, D., Fleming, M., and Krysl, P. 1996. Meshless methods: an overview and recent developments, *Computer Methods in Applied Mechanics and Engineering*, special issue on Meshless Methods, 139: 3-47.

Belytschko, T., Liu, W.K., and Moran, B. 2002. *Nonlinear finite elements for continua and structure*, J. Wiley & Sons, New York.

Chen, D.T., Zeltzer, D. 1992. Pump it up: computer animation of a biomechanically based model of muscle using the finite element method, *ACM SIGGRAPH Computer Graphics*, 26:89-98.

Chen, J. S. and Wang, H. P. 2000. Some Recent Improvements in Meshfree Methods for Incompressible Finite Elasticity Boundary Value Problems with Friction, *Computational Mechanics*, 25: 137-156.

de Boor, C. 2001. *A practical Guide to Splines*. Springer, New York.

Gibson, S., and Mirtich, B. 1997. A survey of deformable modeling in computer graphics, Tech. Report No. TR-97-19, Mitsubishi Electric Research Lab., Cambridge, MA.

Herzog, W. (ed.) 2000. *Skeletal muscle mechanics: from mechanisms to function*. Wiley & Sons, Chichester, UK.

Hollig, K. 2003. *Finite Element Methods with B-Splines*. SIAM, Philadelphia.

Hoschek, J., Lasser, D., and Schumaker, L.L. 1993. *Fundamentals of Computer Aided Geometric Design*, AK Peters, Ltd.

Lemors, R., Epstein, M., Herzog, W., Wyvill, B. 2001. Realistic skeletal muscle deformation using finite element analysis. *Proceedings of the 14th Brazilian Symposium on Computer Graphics and Image Processing*. IEEE Computer Society Press. 192-199.

Li, S. and Liu, W.K. 2002. Meshfree and particle methods and their applications. *Applied Mechanics Review*, 55: 1-34.

Liu, W.K., Chen, Y., Chang, C.T. and Belytschko, T. 1996. Advances in multiscale kernel particle methods. *Computational Mechanics*, 18:73-111.

Ma, D., Lin, F., and Chua, C.K. 2001. Rapid prototyping applications in medicine. Part 1: NURBS-based volume modeling, *International Journal of Advanced Manufacturing Technology*. 18(2):103-117.

Ng-Thow-Hing, V. 2000. Anatomically-based models for physical and geometric reconstruction of humans and other animals. Ph.D. Thesis, Department of Computer Science, University of Toronto.

Ng-Thow-Hing, V. and Fiume, E. 2002. Application-specific muscle representations. *Proc. Graphics and Interface*, 107-115. Eds. Sturzlinger and McCool M.

Oomens, C.W.J., Maenhout, M., van Oijen, C.H.G.A., Drost, M.R., and Baaijens, F.P.T. 2003. Finite element modelling of contracting skeletal muscle, *Phil Trans R Soc Lon B*, 358(1437):1453-1460.

Piegl, L. and Tiller, W. 1997. *The NURBS Book*. Springer-Verlag, New York.

Porcher-Nedel, L., and Thalmann, D. 1998. Real time muscle deformation using mass-spring systems. *Proc. CGI'98*, IEEE Computer Society Press.

Sederberg, T., and Parry, S. 1986. Free-Form deformation of solid geometric models. *ACM Computer Graphics (SIGGRAPH '86 Proc.)*, 20:151-160.

Teran, J., Blemker, S., Ng-Thow-Hing, V., and Fedkiw, R. 2003. Finite volume method for the simulation of skeletal muscle. In *Proc. of the 2003 SIGGRAPH/Eurographics Symp. on Computer Animation*, 68-74.

Terzopoulos D., and Qin, H. 1994. Dynamic nurbs with geometric constraints for interactive sculpting. *ACM Transaction on Graphics*, 13(3):103-136.

U.S. National Library of Medicine. The visible human project, 1994. <http://www.nlm.nih.gov/research/visible/>.

Yang, J., Abdel-Malek, K., Farrell, K., and Nebel, K. 2004. The IOWA Interactive Digital-Human Virtual Environment, 2004 ASME International Engineering Congress, The 3rd Symposium on Virtual Manufacturing and Application, November 13-19, Anaheim, California.

Zajac, F.E., Topp, E.L., and Stevenson, P.J. 1986. A dimensionless musculotendon model. *Proceedings IEEE Engineering in Medicine and Biology*.

Zajac, F.E. 1989. Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. *Critical Reviews in Biomedical Engineering*, 17:359-411.

Zienkiewicz, O.C., and Taylor, R.L. 2000. *The Finite Element Method* (5th. Ed.), Butterworth-Heinemann, Oxford.