Spring 2012

Analysis-ready isogeometric model of skeletal muscles

Matthew John Boss
University of Iowa

ANALYSIS-READY ISOGEOMETRIC MODEL OF SKELETAL MUSCLES

by

Matthew John Boss

A thesis submitted in partial fulfillment
of the requirements for the Master of
Science degree in Mechanical Engineering
in the Graduate College of
The University of Iowa

May 2012

Thesis Supervisor: Associate Professor Jia Lu

Graduate College
The University of Iowa
Iowa City, Iowa

CERTIFICATE OF APPROVAL

MASTER'S THESIS

This is to certify that the Master's thesis of

Matthew John Boss

has been approved by the Examining Committee
for the thesis requirement for the Master of Science
degree in Mechanical Engineering at the May 2012 graduation.

Thesis Committee: ___________________________________
Jia Lu, Thesis Supervisor

____
Timothy Marler

____
Salam Rahmatalla

____
Ralph Stephens

ABSTRACT

New methods are employed to develop an anatomically accurate, analysis-ready isogeometric model of skeletal muscles. Current modeling techniques for the analysis of skeletal muscles include the utilization of finite element meshing, which inherently poses a few well-known problems that provide motivation for isogeometric analysis. In addition to those issues, standard FEA meshing cannot preserve smooth geometries, therefore the accuracy of the foregoing model and analysis is reduced. Moreover, there is no easy means to characterize fiber direction in the FEA framework due to discontinuities at element boundaries. Additionally, material property distributions such as the transition of the muscle-tendon complex along the longitudinal axis through FEA are prescribed on an element by element basis, leading to abrupt, unrealistic property changes at element boundaries.

The current research builds on the idea of an isogeometric tensor-product rod using harmonic coordinates and NURBS [1]. Through a direct comparison between a meshed, FEA model and the tensor-product rod model, it can be shown that the tensor-product rod model preserves smoothness, enhancing the geometric representation passed through to analysis while reducing the total DOF of the model. Muscle fibers can be easily implemented as parametric lines with muscle-specific orientations along the muscles' longitudinal axis that match distinct fiber orientations existent within common skeletal muscles. This technique not only allows for the representation of perfectly parallel-fibered structures, but also those that do not directly follow the longitudinal axis such as a helical twist. Utilizing this geometric method also provides the framework for implementing material properties using an interpolative-style scheme. Varying properties at specific longitudinal control point cross-sections near muscle termination areas can be designated to more accurately represent the muscle-tendon complex. These new techniques allow for the creation of an analysis-ready, realistic skeletal muscle model of the male human arm. The model contains 28 muscles complete with muscle-specific geometric, fiber, and heterogeneous property characterizations all compiled into a complete "digital muscle library."

TABLE OF CONTENTS

LIST OF TABLES vi

LIST OF FIGURES vii

INTRODUCTION 1

CHAPTER

I. ISOGEOMETRIC ANALYSIS BACKGROUND 10

What is Isogeometric Analysis? 10
Origin of NURBS 11
Fundamental NURBS Tools 16
NURBS Surfaces 19
NURBS Solid Representation 20
NURBS for Analysis 21
Development of Tensor-product Rod Method 23
Rational Bezier Circles 23
Harmonic Mapping on a Unit Disk 27
Tensor-Product Rod 32
Cylindrical Element 32
Geometric Accuracy 34

II. APPLICATION TO SKELETAL MUSCLE MODELING 37

Skeletal Muscle Background 37
Muscle Matter 37
Classifications of Contractions 38
Muscle Fiber Orientation 39
Geometric Muscle Model of Male Human Arm 40
Parametric Modeling of Fiber Orientation 44
Parallel Fibers 45
Helical Fibers 47
Convergent Fibers 49
Pennate Fibers 51
Issues with Modeling Fiber Orientation 54
Defining Heterogeneous Material Properties 56
Utilizing the Rod Structure 56
Constitutive Model 59
Digital Muscle Library 61
A Taste of Analysis 76
Kinematics 76
Fiber Stretch 77
Stress Functions 78
Example of Kinematic Analysis 79

III. FUTURE WORK 83

CONCLUSION 86

APPENDIX NURBS CURVE BASICS...89

REFERENCES 99

LIST OF TABLES

Table

1. Elemental summary of NURBS example 22

2. Cross-sectional property distribution for abductor pollicis longus tensor-product model 64

3. Summary of muscles present within the previously pictured muscle model 66

LIST OF FIGURES

Figure

1. de Boor's recursive triangle 14

2. Knot insertion NURBS example..........19

3. A circular Bezier that sweep a value of $2\emptyset$ 24

4. 4-arc Bezier circle..........25

5. A star-shaped polygon 28

6. Isoparametric element mapping example 31

7. Master element definition example..........33

8. A geometric comparison of a baseball bat model..........36

9. Cross-sectional picture of the upper male arm from the Visible Human Data Set 41

10. Rhinoceros rendered images of the complete NURBS-based arm muscle model..........43

11. Mapping of fibers from the master rod to the physical rod 45

12. Parallel-fibered extensor carpi ulnaris 47

13. Helical-fibered brachioradialis..........49

14. Convergent-fibered pronator teres 51

15. Pennate-fibered anconeus 53

16. The deltoid muscle where mapping is not one-to-one..........55

17. Visually assigned tendon/muscle fiber properties and their cross-sectional breakdown..........59

18. Analysis-ready isogeometric model of the abductor pollicis longus..........62

19. Model of a king cobra snake created in Rhinoceros..........80

20. A contour plot produced from fiber stretches of gauss points at deformed state 81

A1. 2D NURBS curve showing the effect of varying weights..........90

A2. 2D NURBS curve basis functions..........90

A3. Example of de Boor's recursive triangle for..........91

A4. Displacement of control point $\boldsymbol{P1}$ 92

A5. Effect of changing the degree of a NURBS curve 93

A6. Effect of varying degree on basis functions..94

A7. Effect of multiple control points in a single location on the NURBS curve.............95

A8. Basis functions for the knot vector $U = 0, 0, 0, 1, 2, 3, 3, 3$ 96

A9. Non-uniform knot vector NURBS example 97

A10. Periodic knot vector NURBS example 98

INTRODUCTION

Skeletal muscle modeling has become an extremely active field of research in the last few years and has been typically driven by two distinct motivations, computer graphics and biomechanical analysis. The motivation for computer graphics is to be able to accurately replicate the shape of muscles and simulate realistic looking muscle deformation. Three-dimensional models can be created using a number of techniques but the most common include polygonal modeling, NURBS, and subdivision surfaces [2]. In polygonal modeling meshes are formed from points in 3D space, called vertices, connected by line segments. Polygonal modeling one of the oldest modeling techniques and is employed in many applications because the models are flexible and can be rendered extremely quickly [2]. A drawback here is the inability to model curved surfaces accurately as they must be approximated by a series of lines segments, which can look unrealistic and limit accuracy. Moreover, attempting to refine the polygon structure to make curved surfaces look as smooth as possible will result in larger processing times. The second modeling type is NURBS (Non-Uniform Rational B-Splines). NURBS curves are generated by utilizing control points which lie off the curve and influence the curve’s shape. Curves can be combined to form NURBS surfaces and solids. Conceptually NURBS modeling is a little less straightforward than polygonal modeling but its benefit lies in its ability to accurately model curved shapes. NURBS can offer smoother surfaces with a smaller footprint in memory storage and faster processing [2]. The last modeling technique is subdivision surfaces. This technique essentially combines polygonal modeling and NURBS. Subdivision surfaces are rendered from a polygon mesh and then subdivided so the surfaces appear smoother and more rounded.

The original polygon mesh still acts as a control cage that can be manipulated similar to NURBS surfaces [2]. This is the most common modeling type used today.

Most 3D skeletal muscle models available today are constructed from polygonal meshes and subdivision surfaces. There have been numerous detailed 3D anatomical models developed with this type of geometry that available for purchase. A couple examples include polygonal models of “3D Male Muscular System Model 5.0” by Zygote Anatomy and “Human Male Anatomy Complete 3D Model Pack V04” developed by Plasticboy [3][4]. These models can be viewed and manipulated using common 3D CAD modeling software such as Maya, 3D Studio Max, or Softimage, all geared toward CG animation.

Other computer graphics modeling techniques such as NURBS have been used for skeletal muscles but not near the magnitude of polygonal geometry. Most applications utilizing 3D NURBS muscle objects strive for the goal of creating realistic-looking muscle deformation from underneath a model’s skin. Christopher Evans and Peter Shipkov are technical artists that have spent extensive time modeling in this manner [5][6]. In the case of CG artists, realistic looking muscle deformation takes precedence as the subsequent anatomical accuracy becomes less of a priority.

Thus regardless of the modeling geometry utilized in generating 3D CG models, the primary goal is to create a realistic looking muscle model. Thus no emphasis is placed on analysis or real-time muscle capabilities, which leads to the second distinct motivation in muscle modeling, biomechanical analysis. Here there is a need to create a model from which efficient analysis techniques can be performed and overall muscle force or stress results can be determined. Here there are really two major techniques for modeling skeletal muscles, line-of-action and finite element analysis (FEA) models. For gross muscle model behavior involving multiple muscles in unison, a line-of-action modeling technique must be used [7]. In this model type muscles are modeled by connecting straight lines from each muscle's origination point to its insertion point. The data characterizing the model is minimal, which makes the computational side of analysis for determining force production and direction trivial, allowing for real-time capabilities. However, being a simplistic model with no associated volumes, this model type is only useful for determining overall force data and not stress-related characteristics. Furthermore modeling with this technique, complex muscle geometries and fiber orientations cannot be accurately represented, which limits the realistic nature of the overall model. An example of this a similar type of analysis was conducted by Patrick and Abdel-Malek in 2007 [8].

To obtain a more detailed and accurate depiction of muscle behavior, FEA models are commonly utilized and accepted as the standard analysis technique. Through FEA, volumetric data is utilized and useful stress results can be obtained. There has been extensive research conducted utilizing FEA to model stand-alone skeletal muscles and develop muscle stress distributions. An example of such research was conducted by Spyrou and Avaras in 2011. The objective here was to introduce a 3D FEA model of the human foot and lower leg to simulate human ankle flexion and estimate internal stresses and strains as well as shape changes of the deformed tissues during human movement [9].

Although FEA is the commonly accepted practice for stress determination in CAD models, it does however have a number of limitations associated with its use. First, model generation for analysis is done through the use of a meshed approximation. To obtain accurate results for smooth domains, this meshed approximation consisting of nodes connected by straight lines must be significantly refined to approximate smooth, curved surfaces. If this level of refinement is neglected, the ensuing analytical results will suffer. This fine approximation also drastically increases the computational cost due to the large number of degrees of freedom (DOF) necessary to construct the model. The robust nature of the FEA meshed, geometric model is the main reason only individual muscles, not multiple muscles working in unison, are typically analyzed in most of the studies conducted today.

Modeling fiber orientation in skeletal muscles has also historically been a difficult task to model within the FEA framework. Fiber direction is one of the main determinants of overall muscle force generation [10]. Therefore a proper parameterization scheme is essential to an accurate model. The drawback here is that each element of the FEA model is parameterized separately by a piecewise polynomial approximation. Therefore C^0 exists at element boundaries and it follows that there is no easy way to trace fibers across elements. Most current methods involve creating a mesh whose fiber direction is associated with the structure of the ensuing meshed element boundaries and/or shape. Such is the case in the work of Lemos [11] in which eight-node brick-like elements were assembled one on top of another to represent parallel-fibered and pennate-fibered muscles. Thus fiber orientation was determined directly from the structural representation. Tang [12] used a method to model fiber orientation in which the distribution of fibers take on the outline of the fusiform muscle geometry. This was done for each element by determining the line that connected the two central points on the two end surfaces. Models of this nature are muscle-specific, meaning a unique mesh structure must be created for each muscle to accurately represent the geometry and fiber direction. Also, to alter the mesh structure one must refer back to the original geometry, which lacks the versatility needed for a general-use model. In yet another study, Blemker [13] created a fiber map of interpolated cubic splines based on fascicle arrangement measurements of ultrasound images. For each element the fiber length was determined from the vector tangent to the spline curve that runs the element's center. Oomens [14] measured fiber directions using tensor imaging and then described them with polynomial fits. In both of these cases the fiber maps utilized remained detached from the actual geometric model itself.

The previously discussed discontinuity between FEA elements is also limiting for the assignment of material properties within the model. Elements must be assigned on an element-by-element basis, which translates to an unrealistic, abrupt property change at element boundaries of the meshed model. In realistic skeletal muscle the muscle-tendon complex exhibits a gradual transitional period, thus it is imperative to model this region as accurately as possible to ensure the highest possible accuracy in analytical results. In many current FEA muscle studies, this transitional period is predominantly neglected and studies that include properties assignment schemes generally do so element-by-element [11][14][12].

All of the afore mentioned limitations culminate in an FEA analysis-ready model that lacks accuracy due to a misrepresentation of volumetric results and the inability to model muscle fiber and material properties in a smooth and efficient manner. In addition, modern FEA models of this nature utilize a high number of DOF in implementation, which limits the possibilities for multi-muscle analysis. To alleviate many of the issues common with FEA, an alternative method termed “Isogeometric Analysis” was developed by Hughes in 2005 which can be utilized [15]. This method satisfies both of the two previously discussed motivations of skeletal muscle modeling, computer graphics and biomechanical analysis. In this manner the physical realism can be combined with a useful stress analysis technique, i.e. merging a CAD (Computer Aided Design) geometric representation with FEA capabilities. There are a few different geometric modeling techniques utilized in isogeometric methods, but of the most relevance in this application of smooth-bodied skeletal muscles is NURBS. More specifically, a structural variation of NURBS called the tensor-product rod formulation pioneered by Lu and Zhou in 2020 will be utilized [1].

The tensor-product method is ideal modeling skeletal muscles because of their intrinsic rod-like shape. This method merges cross-sectional and axial representations to form this rod-like structure. The cross-sectional representation uses rational Bezier arcs as a base and combines multiple Bezier arcs to form the unit circle, which can be reduced to ensure continuity across the cross-sectional boundaries. Extending that boundary description to the interior is done through harmonic mapping. The result of which is an interior that is completely parameterized by the boundary control points only. Using this harmonic mapping, cross-sectional isoparametric master elements are developed. The combination of the cross-sectional and axial descriptions represented by a simple NURBS curve is done through the utilization of the tensor-product, for which the domain is described by a unit cylinder. Finally, the master rod structure can be mapped to that of the physical domain as determined by the overall muscle shape.

The tensor–product rod formulation has numerous benefits which allow us to exceed the current capabilities of FEA for skeletal muscle modeling. NURBS models such as this can preserve smooth boundaries and exactly represent conic cross-sections. This allows for a much lower total DOF model, decreasing the overall computational burden. However, perhaps the most important feature that accompanies the tensor-product model is the globally smooth representation, meaning that elements have levels of continuity across their boundaries that are nonexistent in FEA. The global parametric setup and continuity across element boundaries of tensor-product rod allows for easy implementation of the various fiber arrangement schemes that are common within skeletal muscles, leading to more accurate results. Also exploiting that element continuity and globally smooth representation, the structure of the tensor-product rod can be utilized for easy assignment of material property variations along the muscle's longitudinal axis, which again allows for a more accurate depiction of realistic material property distributions associated with the muscle-tendon complex.

Knowing the limitations of FEA and the benefits of the tensor-product isogeometric method, the objectives of this work can clearly be defined. The overall goal of this work is to create an analysis-ready isogeometric model of skeletal muscles within the human arm that utilize much lower total DOF and is more accurate in the parameterization of muscle fibers and material property distribution than currently possible through modern FEA techniques. To completely characterize an analysis-ready model in this manner, one needs to accurately satisfy three main criteria. The first is creating an anatomically accurate geometric model that is fully 3D and capable of handling the complexities of muscle geometry with a lower number of DOF. Secondly, a model needs to accurately reflect the fiber architecture presented within skeletal muscles as fiber direction is the major determinant of overall force generation. And lastly a model needs to accurately represent heterogeneous property distributions, such as the muscle-tendon complex. Utilizing the isogeometric tensor-product method, an anatomically accurate geometric model of the male human arm will be developed and new techniques for muscle-specific fiber parameterization and material property assignment will be employed to complete all steps in the characterization process and satisfy this hypothesis.

The need for an accurate as possible and low DOF analysis-ready model has become more and more explicit as of late due to computational analysis being an exponentially expanding field. Here the end goal is a full scale muscle model capable of analyzing muscle stresses through real-time interaction. The application possibilities with a model such as this are numerous. Ergonomists in industry could utilize a model to predict muscle stress and overall force capabilities in a number of virtual environments before actual product prototypes are constructed. This would lead to immediate cost savings and a reduction in the overall design process. Here injury prevention also becomes a real possibility through the prediction of muscle overexertion in a digital environment.

This thesis begins with a brief description of isogeometric analysis. It then proceeds to discuss the background and basics of the most prominent geometric tool utilized in isogeometric analysis, NURBS. The NURBS elemental structure and fundamental manipulation tools are described in detail. Next the tensor-product rod method is derived through the utilization of NURBS as a base geometry. The mesh-like structure and elemental breakdown of the tensor-product rod geometry is derived. The geometric accuracy of this method is compared to that of standard FEA techniques. The construction of a complete NURBS model of the human arm is presented along with the tensor-product rod method to model realistic skeletal muscles. Actual muscle specific parametric fiber orientations are presented and employed. Derived fiber parameterizations are explained including parallel, helical, convergent, and pennate orientation cases. The material property implementation scheme through the tensor-product rod structure for the muscle-tendon complex and respective constitutive relations are described with emphasis on fiber stretch determination. The complete characterization of this muscle model is then summarized with the concept of a “Digital Muscle Library” along with its usefulness discussed in further detail. Lastly, limitations of the model and future work are examined.

CHAPTER I

ISOGEOMETRIC ANALYSIS BACKGROUND

What is Isogeometric Analysis?

The idea of isogeometric analysis was first introduced by Thomas Hughes, a professor at the University of Texas, in 2005 [15]. This approach is a new method for analyzing partial differential equations and is similar to existing techniques of finite element analysis (FEA) and mesh-less methods. NURBS are the most commonly used geometry type when implementing isogeometric analysis. Other types of geometric representations do exist and are used in isogeometric analysis but are not as prevalent. One major generalization of NURBS is called T-splines. T-splines work exceptionally well for local refinement and coarsening, but are very robust and computationally extensive when sewing together adjacent patches [16]. Subdivision surfaces, Gordon patches, Gregory patches, S-patches, and A-patches have also been utilized and could play future roles in isogeometric research.

The current engineering analysis procedure is to develop a CAD model and use that model's geometric shape to develop a mesh structure to conduct analysis. The mesh generated in these cases is an approximation of the original geometry, implying there are inherent errors in any analytical results arising from the geometric error. When creating a mesh of a geometric model in this manner, the link to the original geometry is completely severed. Any refinement strategies thereafter must utilize the original CAD model rather than the most recent meshed approximation. A benefit of the geometric structure of a NURBS model is that it allows for direct analysis, eliminating the need to create that external mesh. Using this approach, there is no longer a need for the mesh generation process in isogeometric methods, which can be an extremely time consuming. Currently, the mesh generation process, which includes meshing and mesh manipulation, accounts for about 20% of overall analysis time [16]. Therefore it is the primary goal of isogeometric analysis is to be geometrically exact no matter how coarse the discretization. Another goal is to simplify mesh refinement by eliminating the need for communication with the CAD geometry once the initial mesh is created [15].

Origin of NURBS

As NURBS are the primary basis for the current analysis technique, a brief background on the subject is necessary. NURBS are a parametric formulation. Simple parametric or mapped curves can be represented by

$$\boldsymbol{C}(u) = \left(x(u), y(u), z(u)\right), \qquad \mathrm{a} \leq u \leq \mathrm{b} \tag{1}$$

where the x, y, and z components trace out a curve as *u* varies from *a* to *b*. Parametric curves are extremely flexible because they are axis independent, easily represent multi-valued functions, and have additional degrees of freedom when compared to explicit or implicit formulations [17].

NURBS are derived from the Bezier curve discovered by Pierre Bezier in the late 1960’s and early 1970’s [17]. The method of shape description developed was suitable to model free-form curves and surfaces. Originally the method was used to model automobiles, aircraft wings, and yacht hulls, but has since expanded into a much broader field [17]. A Bezier curve follows a parametric representation and can be described by

$$\boldsymbol{C}(u) = \sum_{i=0}^{n} B_{n,i}(u)\boldsymbol{P}_i \qquad 0 \leq u \leq 1 \tag{2}$$

where P_i is the *i*th polygon vertex location, or control point as it will be here on referred to as, in the form of a position vector. $B_{n,i}(u)$ is the *i*th *n*th-order Bernstein polynomial which is described by

$$B_{n,i}(u) = \binom{\mathrm{n}}{\mathrm{i}} u^i(1-u)^{n-i}, \qquad 0 \le u \le 1 \tag{3}$$

where

$$\binom{\mathrm{n}}{\mathrm{i}} = \frac{\mathrm{n}!}{\mathrm{i}!\,(\mathrm{n}-\mathrm{i})!} \tag{4}$$

is the binomial coefficient. In these equations *n* is the degree of the Bezier basis function or the polynomial curve segment. Therefore an *n*th degree curve would require *n* + 1 control points.

Bezier curves are very useful in their own sense, yet they have a few limitations inherent in their formulation. One limitation is that the number of specified control points, as mentioned, fixes the order of the polynomial curve. Hence, the only way to reduce the order is to reduce the number of control points. This also means that a high degree is required to fit complex shapes. Secondly, each basis function of the curve is nonzero for all parameter values. This is limiting in the sense that a change in any one of the control points would affect the shape of the entire curve, so local changes cannot be made.

Shortly following Bezier's discovery, methods applying a B-spline basis were developed [17]. B-splines are piecewise Bezier curves that all have the same polynomial degree. B-spline curves are non-global in nature because each control point has a local influence region meaning perturbing a control point will only affect a certain range of parameter values. B-splines also allow for the order of a curve to be altered without changing the total number of control points. A B-spline curve is described by

$$\boldsymbol{C}(u) = \sum_{i=0}^{n} N_{\mathrm{i,p}}(u)\boldsymbol{P}_i \qquad \mathrm{a} \leq u \leq \mathrm{b} \tag{5}$$

with P_i again being control point locations. Here $N_{\mathrm{i,p}}(u)$ are the normalized B-spline basis functions described by the Cox-de Boor recursion formula given as

$$N_{\mathrm{i,0}}(u) = \begin{cases} 1, & u_i \leq u \leq u_{i+1} \\ 0, & otherwise \end{cases} \tag{6}$$

and

$$N_{\mathrm{i,p}}(u) = \frac{(\mathrm{u} - \mathrm{u_i})}{(\mathrm{u_{i+p}} - \mathrm{u_i})} N_{\mathrm{i,p-1}}(u) + \frac{(\mathrm{u_{i+p+1}} - \mathrm{u})}{(\mathrm{u_{i+p+1}} - \mathrm{u_{i+1}})} N_{\mathrm{i+1,p-1}}(u) \tag{7}$$

These basis function are of degree *p* (order *p* + 1) and controlled by *n* +1 control points. The values of $\mathrm{u_i}$ are elements of a knot vector, called knots. A knot vector is a non decreasing sequence of real numbers of the form $U = \{\underbrace{a, \dots, a}_{p+1}, u_{p+1}, \dots, u_{m-p-2}, \underbrace{b, \dots, b}_{p+1}\}$ for the non-periodic and non-uniform case. Each knot at the beginning and ending of the knot vector is repeated *p* + 1 times. The end conditions ensure that the curve initiates and terminates at the initial and final control point locations. Interior knots can be repeated up to *p* times, but each time a knot is repeated, the continuity at that location will decrease by one. Each knot interval $[u_k, u_{k+1}]$ represents an individual Bezier segment. The number of control points required to describe a B-spline curve is $l = m - p - 1$, where *m* is the length of the knot vector and *p* is the degree of the B-spline curve. The number of basis functions required is $n + 1 = l$, corresponding to the number of control points forming the curve.

The basis functions can be better understood by looking at de Boor's recursive triangle as show in Figure 1.

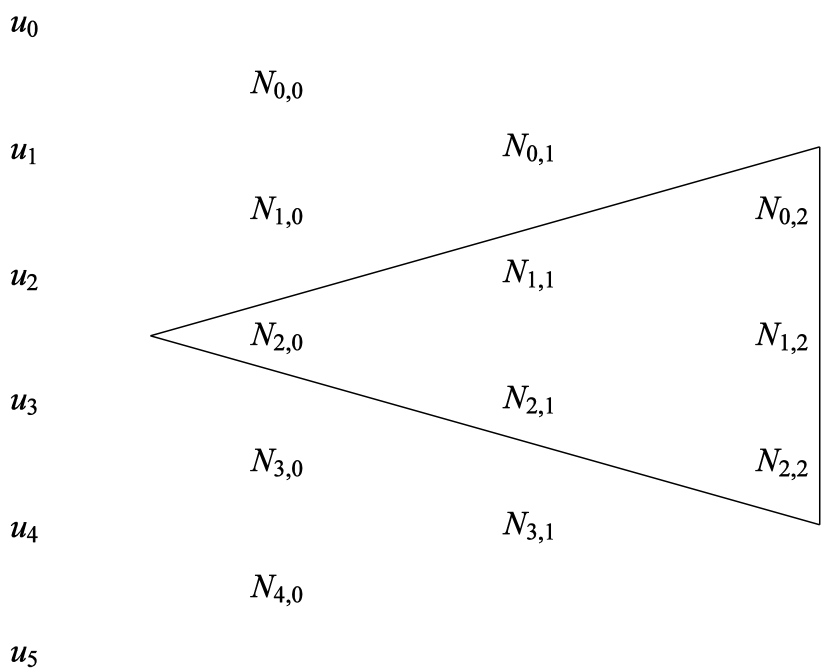


Figure 1. de Boor's recursive triangle

Consider a knot vector described by six knots as given by $U = \{u_0, u_1, u_2, u_3, u_4, u_5\}$. These values are shown off to the left-hand side. The second, third, and fourth columns represent the degree p = 0, p = 1, and p = 2 basis functions, respectively. Considering a degree p = 0 curve over the knot span $\{u_2, u_3\}$, $N_{2,0}$ is the only basis function that is non-zero over the knot span. Considering a degree $p = 1$ curve over the knot span $\{u_2, u_3\}$, the basis functions $N_{1,1}$ and $N_{1,2}$ are both non-zero over the knot span. Likewise for a degree p = 2 curve over that same knot span, $N_{0,2}$, $N_{1,2}$, and $N_{2,2}$ are all non-zero basis functions. Thus a triangle of non-zero basis functions is formed for this particular knot span.

A rational B-spline curve is the projection of a non-rational B-spline curve defined in (d+1)-D homogeneous coordinate space back to d-D physical space, where

$\mathrm{d} = 2{,}3$ is the dimension of the physical space [17]. Dividing through by the homogeneous coordinate yields the formal definition for a NURBS curve

$$\boldsymbol{C}(u) = \frac{\sum_{i=0}^{n} N_{\mathrm{i,p}}(u)\mathrm{w_i}\boldsymbol{P}_i}{\sum_{i=0}^{n} N_{\mathrm{i,p}}(u)\mathrm{w_i}} = \sum_{i=0}^{n} R_{\mathrm{i,p}}(u)\boldsymbol{P}_i \qquad \mathrm{a} \leq u \leq \mathrm{b} \tag{8}$$

The transformation of the equation for a B-spline curve to this representation includes the addition of a weighting term, $\mathrm{w_i}$, for each control point. Here, $\mathrm{w_i} > 0$ for all values of *i*. The term $R_{\mathrm{i,p}}(u)$ represents the rational basis functions. All previous symbols for the B-spline definition retain similar meanings. The weighting terms in the NURBS representation allows for exact representation of conics as discussed earlier. B-splines do not have that capability and cannot exactly represent these basic geometric shapes. The NURBS curve has the following analytic and geometric characteristics [17]:

$R_{\mathrm{i,p}}(u) \geq 0$ for all parameter values.

The basis functions exhibit partition of unity, i.e., $\sum_{i=0}^{n} R_{\mathrm{i,p}}(u) = 1$

Except for first-order basis functions, each rational basis function has only one maximum.

A NURBS curve of degree p is $\mathrm{C^{p-1}}$ continuous everywhere, providing that no internal knots are repeated.

A NURBS curve exhibits the variation-diminishing property, meaning no plane has more intersections with the curve than with the control polygon [18].

A NURBS curve generally follows the shape of the control polygon and $C(u)$ lies within the convex hull of control points $\boldsymbol{P}_{i-p}, \dots, \boldsymbol{P}_i$.

Any projective transformation is applied to a NURBS curve by applying it to the control points, i.e., the curve is invariant with respect to affine and projective transformations.

Some brief examples of NURBS curves and common variations are explored further in Appendix A.

Fundamental NURBS Tools

There a few useful manipulation techniques of NURBS curves which prove to be very useful in isogeometric analysis. These include degree elevation, degree reduction, knot insertion, knot refinement, and knot removal. Degree elevation is the process of increasing the degree of a NURBS curve, which may always be done if desired and will not change the geometry or parameterization. Degree reduction is the exactly opposite process of degree elevation. This is useful when combining NURBS curves and surfaces because before doing so it is necessary for them to have a common degree. Degree elevation can be thought of as p-refinement in the FEA framework. Degree elevation will not visually yield a different result.

Following the notation in [18], consider the basic representation for a NURBS curve as shown in (8), $\boldsymbol{C_p}(u) = \sum_{i=0}^{n} R_{\mathrm{i,p}}(u)\boldsymbol{P}_i$. This is a pth-degree NURBS curve parameterized by the knot vector U. By elevating the degree of this polynomial curve to $p + 1$, the resulting curve has the notation

$$\boldsymbol{C_{p+1}}(u) = \sum_{i=0}^{\hat{n}} R_{\mathrm{i,p+1}}(u)\boldsymbol{Q}_i \qquad \mathrm{a} \leq u \leq \mathrm{b} \tag{9}$$

where $\boldsymbol{C_p}(u) = \boldsymbol{C_{p+1}}(u)$ and they represent the same curve both geometrically and parametrically. The new control point locations $\boldsymbol{Q}_i$ and knot vector $\widehat{U}$ are the unknowns solved for in the degree elevation process.

Knot insertion is the process of inserting a knot into a knot vector. This is just a change of vector space; the curve again is not changed geometrically or parametrically [18]. Knot insertion is used for the evaluation of points and derivatives on curves, subdividing curves, and adding control points in order to increase flexibility in shape control. Knot refinement is similar to knot insertion, but instead multiple knots are inserted at one time. Knot removal is the reverse process of knot insertion. This can be useful in generating a more compact geometric representation. For example, when combining curves the first step is to make the curves have a common degree. Next, composition is accomplished by using interior knots of multiplicity equal to the common degree. Thus knot removal can be invoked afterward to remove unnecessary knots. In general, as many knots as possible can be removed as long as the continuity of the curve permits.

Knot insertion is similar to FEA's h-refinement. Again following the notation found in [18], consider the basic representation for a NURBS curve as shown in (8), $\boldsymbol{C}(u) = \sum_{i=0}^{n} R_{\mathrm{i,p}}(u)\boldsymbol{P}_i$, which is defined on the knot vector $U = \{u_0, \dots, u_m\}$. Inserting a knot span $\bar{u} \in [u_k, u_{k+1})$ into this knot vector results in a new knot vector $\bar{U} = \{\bar{u}_0 = u_0, \dots, \bar{u}_k = u_k, \bar{u}_{k+1} = \bar{u}, \bar{u}_{k+2} = u_{k+1}, \dots, \bar{u}_{m+1} = u_m\}$. Therefore the new curve takes the form

$$\boldsymbol{C}(u) = \sum_{i=0}^{n+1} \bar{R}_{\mathrm{i,p}}(u)\boldsymbol{Q}_i \qquad \mathrm{a} \le u \le \mathrm{b} \tag{10}$$

with $\bar{R}_{\mathrm{i,p}}(u)$ as the pth-degree basis functions on $\bar{U}$. The new control point locations $\boldsymbol{Q}_i$ are determined by solving the following set of linear equations

$$\sum_{i=0}^{n} R_{\mathrm{i,p}}(u)\boldsymbol{P}_i = \sum_{i=0}^{n+1} \bar{R}_{\mathrm{i,p}}(u)\boldsymbol{Q}_i \tag{11}$$

The proof in its entirety is shown in [18]. The solution of which is determined to be

$$\boldsymbol{Q}_i = \propto_i \boldsymbol{P}_i + (1 - \propto_i)\boldsymbol{P}_{i-1} \tag{12}$$

where

$$\propto_i = \frac{\bar{u} - u_i}{u_{i+p} - u_i} \ and \ 1 - \propto_i = \frac{u_{i+p} - \bar{u}}{u_{i+p} - u_i} \tag{13}$$

A visual example showing the effect of knot insertion is shown in Figure 2. The NURBS information for a curve prior to knot insertion as depicted in Figure 2a is:

$$\{\boldsymbol{P}_0, \boldsymbol{P}_1, \boldsymbol{P}_2\} = \{(0,0), (1,2), (2,0)\}$$

$$u = \{0,0,0,1,1,1\}$$

$$w = \{1,1,1\}$$

$$p = 2$$

Figure 2b shows the same curve after a knot has been inserted at $u = 0.5$. Here the number of knots increases but the curve geometry remains exactly the same as it was before the knot was inserted. The NURBS information for the curve after knot insertion as depicted in Figure 2b is:

$$\{\boldsymbol{Q}_0, \boldsymbol{Q}_1, \boldsymbol{Q}_2, \boldsymbol{Q}_3\} = \{(0,0), (0.5,1), (1.5,1), (2,0)\}$$

$$u = \{0,0,0,0.5,1,1,1\}$$

$$w = \{1,1,1,1\}$$

$$p = 2$$

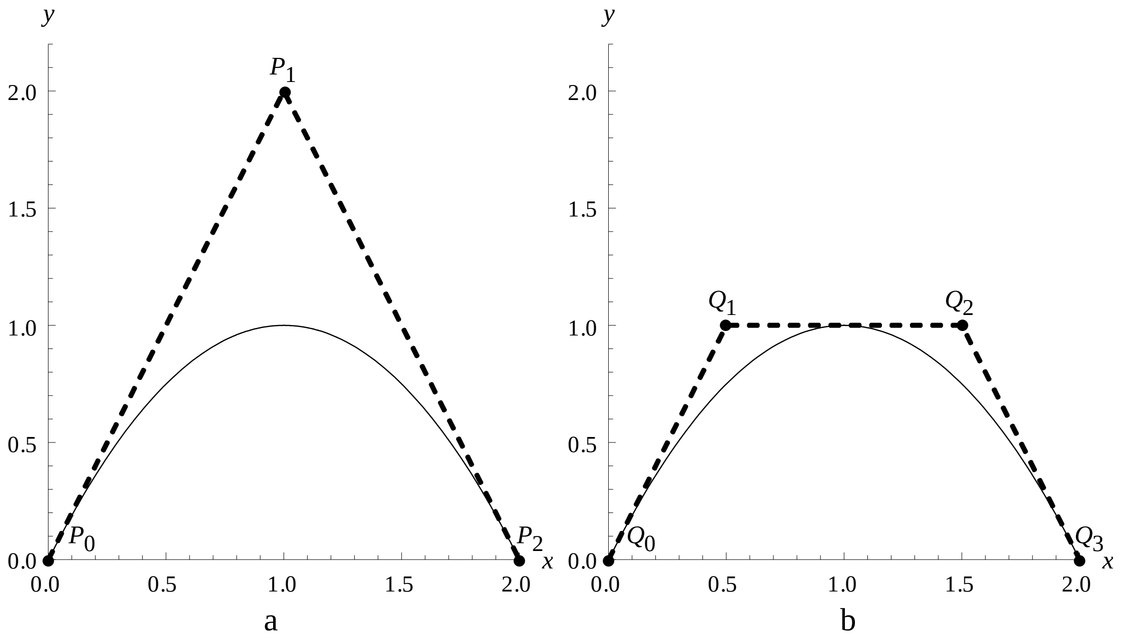


Figure 2. Knot insertion NURBS example
(a) Before knot insertion
(b) After knot insertion

In this work, the algorithms of each of these processes will not be covered in detail. Instead it is simply adequate for the reader to just be aware of these manipulation techniques. Details regarding the various computational algorithms for degree elevation, knot removal, etc. can be found in [17] or [18].

NURBS Surfaces

A NURBS surface as one could imagine is developed by varying two different parametric quantities in two different directions. In general a parametric surface can be described by

$$\boldsymbol{S}[u,v] = \left(\boldsymbol{x}(u,v), \boldsymbol{y}(u,v), \boldsymbol{z}(u,v)\right), \qquad \mathrm{a} \le u \le \mathrm{b}, \mathrm{c} \le v \le \mathrm{d} \tag{14}$$

A NURBS surface is then represented by

$$\boldsymbol{S}(u,v) = \frac{\sum_{i=0}^{n}\sum_{\mathrm{j=0}}^{\mathrm{m}} N_{\mathrm{i,p}}(u)N_{\mathrm{j,q}}(v)\mathrm{w_{i,j}}\boldsymbol{P}_{i,j}}{\sum_{i=0}^{n}\sum_{\mathrm{j=0}}^{\mathrm{m}} N_{\mathrm{i,p}}(u)N_{\mathrm{j,q}}(v)\mathrm{w_{i,j}}} = \sum_{i=0}^{n}\sum_{\mathrm{j=0}}^{\mathrm{m}} R_{\mathrm{i,j}}(u,v)\boldsymbol{P}_{i,j} \tag{15}$$

$$\mathrm{a} \le u \le \mathrm{b}, \mathrm{c} \le v \le \mathrm{d}$$

where *p* is the degree of the surface in the *u* direction and *q* is the degree in the *v* direction. In contrast to the curve representation described by control points, the surface representation is described by a control net $\boldsymbol{P}_{i,j}$. $N_{\mathrm{i,p}}(u)$ and $N_{\mathrm{j,q}}(v)$ are the basis functions in the two directions with distinct knot vectors

$U = \{\underbrace{a, \dots, a}_{p+1}, u_{p+1}, \dots, u_{n-p-1}, \underbrace{b, \dots, b}_{p+1}\}$ and $V = \{\underbrace{c, \dots, c}_{q+1}, v_{q+1}, \dots, v_{m-q-1}, \underbrace{d, \dots, d}_{q+1}\}$.

The properties of NURBS surfaces are the same as discussed earlier for NURBS curves.

NURBS Solid Representation

NURBS tensor solids are defined in a similar manner to NURBS surfaces. The general parametric solid is defined by

$$\boldsymbol{D}[u,v,w] = \big(\boldsymbol{x}(u,v,w), \boldsymbol{y}(u,v,w), \boldsymbol{z}(u,v,w)\big), \qquad \mathrm{a} \le u \le \mathrm{b}, \mathrm{c} \le v \le \mathrm{d}, \mathrm{e} \le w \le \mathrm{f}. \tag{16}$$

It then follows that the NURBS solid representation

$$\boldsymbol{D}(u,v,w) = \frac{\sum_{i=0}^{n}\sum_{\mathrm{j=0}}^{\mathrm{m}}\sum_{\mathrm{k=0}}^{\mathrm{l}} N_{\mathrm{i,p}}(u)N_{\mathrm{j,q}}(v)N_{\mathrm{k,r}}(w)\mathrm{w_{i,j,k}}\,\boldsymbol{P}_{i,j,k}}{\sum_{i=0}^{n}\sum_{\mathrm{j=0}}^{\mathrm{m}}\sum_{\mathrm{k=0}}^{\mathrm{l}} N_{\mathrm{i,p}}(u)N_{\mathrm{j,q}}(v)N_{\mathrm{k,r}}(w)\mathrm{w_{i,j,k}}} = \sum_{i=0}^{n}\sum_{\mathrm{j=0}}^{\mathrm{m}}\sum_{\mathrm{k=0}}^{\mathrm{l}} R_{\mathrm{i,j,k}}(u,v,w)\boldsymbol{P}_{i,j,k} \tag{17}$$

$$\mathrm{a} \le u \le \mathrm{b}, \mathrm{c} \le v \le \mathrm{d}, \mathrm{e} \le w \le \mathrm{f}$$

where *p* is the degree of the solid in the *u* direction, q is the degree in the *v* direction, and *r* is the degree in the *w* direction. The solid representation is controlled by a tri-variate control lattice $\boldsymbol{P}_{i,j,k}$. $N_{\mathrm{i,p}}(u)$, $N_{\mathrm{j,q}}(v)$, and $N_{\mathrm{k,r}}(w)$ are the basis functions in the three directions with distinct knot vectors $U = \{\underbrace{a, \dots, a}_{p+1}, u_{p+1}, \dots, u_{n-p-1}, \underbrace{b, \dots, b}_{p+1}\}$, $V = \{\underbrace{c, \dots, c}_{q+1}, v_{q+1}, \dots, v_{m-q-1}, \underbrace{d, \dots, d}_{q+1}\}$, and $W = \{\underbrace{e, \dots, e}_{r+1}, w_{r+1}, \dots, w_{l-r-1}, \underbrace{f, \dots, f}_{r+1}\}$. Here again the properties of a NURBS curve also pass through to the solid representation.

It should be noted that the NURBS tensor solid representation as defined here is an adequate geometric and parametric representation for muscle analysis. However, the focus in the present work is for tubular skeletal muscles. Therefore a more suitable method simplifying the geometric description is available for this case. This parameterization method is termed the tensor-product rod method and will be described further in future sections.

NURBS for Analysis

Analysis of a NURBS based geometric model is very similar to that of traditional FEA (Finite Element Analysis) techniques. In FEA the assumed solution is an interpolation of nodal values via basis functions which are defined by polynomials over each specific element. The assumed solution is infinitely smooth within an element, but C^0 continuous at element boundaries, meaning the values of the primary unknowns in neighboring elements must be equal where the derivatives do not. Displacing nodes generates a displacement field, which can then be linked to stress with the use of constitutive relations.

Unlike FEA, in the NURBS case control points are the “element nodes.” Therefore the degrees of freedom in NURBS analysis are associated with these control points located off the physical curve/surface/solid. These control points in general do not lie on the physical surface itself, thus two different meshes ensue. The control mesh is formed by the connecting the actual control points off the body, which determine overall surface orientation. Analysis of a NURBS model is formulated using the control mesh. The physical mesh lies on the actual surface of the model. The physical mesh is really just used for visualization purposes. A physical mesh is possible because NURBS curves consist of piecewise segments combined together and the locations where those segments are joined together can be thought of as the division lines between elements. For each non-empty knot span $[u_i, u_{i+1}]$, the curve can be viewed as infinitely smoothly and thus each span is an individual element. Again, a major difference between FEA and NURBS is that at the NURBS element boundary, the curve is C^{p-1} continuous if no multiplicity occurs. Perturbing the control points in a NURBS analysis technique leads to similar results as displacing nodes in FEA.

The elemental breakdown is understood by referring to the example from Figure A21 and Figure A22 of Appendix A. This is a curve of degree *p* = 2 and a knot vector of $U = \{0, 0, 0, 1, 2, 3, 3, 3\}$. There are three non-empty knot spans therefore the curve must contain three elements. Each element is determined by three control points or "element nodes." The results are summarized in Table 1.

Table 1. Elemental summary of NURBS example

Element	Knot Interval	Element Nodes (Control Points)
1	$[u_2, u_3]$	$\boldsymbol{P_0}, \boldsymbol{P_1}, \boldsymbol{P_2}$
2	$[u_3, u_4]$	$\boldsymbol{P_1}, \boldsymbol{P_2}, \boldsymbol{P_3}$
3	$[u_4, u_5]$	$\boldsymbol{P_2}, \boldsymbol{P_3}, \boldsymbol{P_4}$

Development of Tensor-product Rod Method

With the formulation behind NURBS known, one may begin to see the numerous benefits of using such a geometric representation. One particular benefit of NURBS is the fact that it can exactly represent conic curves and surfaces [18]. This is ideal for the development of the tensor-product rod method because rod-like structures can be constructed from circular cross-sections. Representing conical and circular cross-sections is done through utilizing a special case of NURBS, the rational Bezier circle.

Rational Bezier Circles

A quadratic rational Bezier arc has the form

$$\boldsymbol{C}(u) = \frac{(1-u)^2 w_0 \boldsymbol{P}_0 + 2u(1-u)w_1\boldsymbol{P}_1 + u^2 w_2 \boldsymbol{P}_2}{(1-u)^2 w_0 + 2u(1-u)w_1 + u^2 w_2}$$

$$= R_{0,2}(u)P_0 + R_{1,2}(u)\boldsymbol{P}_1 + R_{2,2}(u)\boldsymbol{P_2} \tag{18}$$

where $$R_{i,2}(u) = \frac{B_{i,2}(u)w_i}{\sum_{j=0}^{2} B_{j,2}(u)w_j} \qquad 0 \le u \le 1$$

Here $B_{i,2}(u)$ are the Bezier basis functions described previously. If $w_0 = w_2 = 1$ and $w_1 = w$, the arc shaped with weighting values of $w > 1, w = 1, and\ w < 1$ will lead to a hyperbolic, parabolic, and elliptical representation, respectively [19].

For the present case, circular arcs are of the most importance. A circle is a special case of an ellipse, therefore with the right value of w a circular arc can be formed from the above equations. In this case three control points form an isosceles triangle because of symmetry and from there geometrically it can be determined that $w = \cos \emptyset$ [19]. Here $\emptyset$ is the angle between the line segments $\boldsymbol{P_0P_1}$ and $\boldsymbol{P_0P_2}$ as shown in Figure 3. Thus, a circular arc can be represented with the control point locations

$$\boldsymbol{P} = \{(1,0), (1, \tan \emptyset), (\cos 2\emptyset , \sin 2\emptyset)\} \tag{19}$$

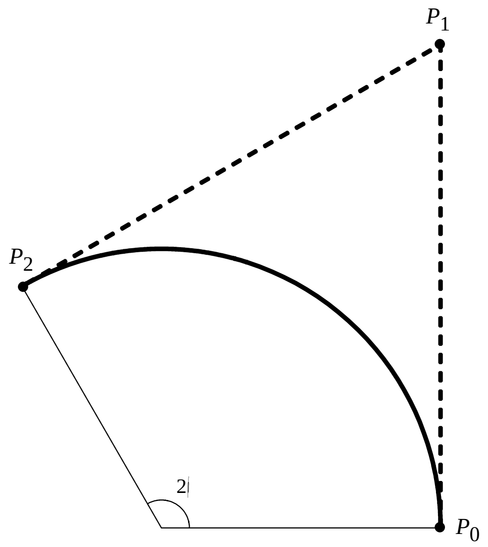


Figure 3. A circular Bezier that sweep a value of $2\emptyset$

The previous description of a circular arc allows for a sweep of up to 180°. Therefore at least three circular arc segments of this nature are needed to piece together a complete circle. When piecing arcs together, the termination point for one arc becomes the starting point for the next arc. These specific points are referred to as junction points. Thus, a circle can be defined by n equal arcs and $2n + 1$ control points [19]. Keeping the arcs of equal lengths provides a cleaner parameterization. The resulting circle can be thought of as a piecewise rational Bezier curve or a NURBS curve.

A circle parameterized by four arcs is shown in Figure 4a. The combination of arcs can be represented by a single parametric domain $[0, n]$, where each arc interval is represented on the parametric interval $[u_i, u_{i+1}) = [i, i + 1)$ for $i = 0, \dots, n - 1$. This representation also can be thought of as a degree p = 2 NURBS curve. For the 4-arc circle in Figure 4a with similar control point locations, the NURBS representation would contain the following weighting values and knot vector:

$$w = \{1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1\}$$

$$u = \{0, 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 4\}$$

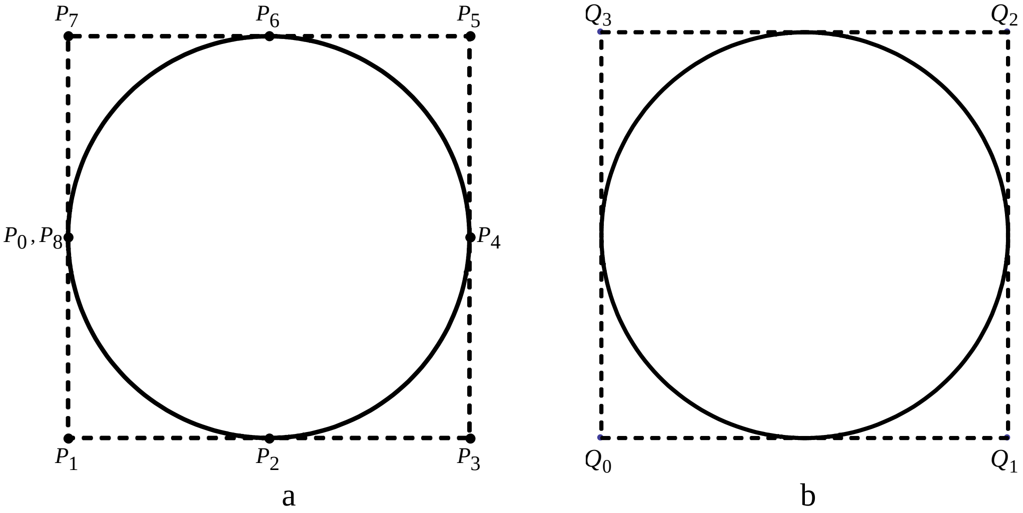


Figure 4. 4-arc Bezier circle
(a) Before reduction
(b) After reduction

In this representation the circle may be deformed by displacing the control points. If deformed in the present configuration and for example $\boldsymbol{P}_2$ did not remain collinear with its neighboring control points, a cusp may form and smoothness may be lost. Following the procedure described in [20] to maintain C^1 continuity, junction points can be eliminated and the piecewise Bezier curve depicted in Figure 4a can be reduced to Figure 4b. Considering the arc represented by $\boldsymbol{P}_0\, \boldsymbol{P}_1\, \boldsymbol{P}_2$, the following substitutions of $\boldsymbol{P}_0 = \frac{1}{2}(\boldsymbol{P}_7 + \boldsymbol{P}_1)$ and $P_2 = \frac{1}{2}(\boldsymbol{P}_1 + \boldsymbol{P}_3)$ are made resulting in a curve given by

$$\boldsymbol{c}(u) = \frac{N_{0,2}(u)}{2}\,\boldsymbol{P}_7 + \frac{N_{0,2}(u) + 2N_{1,2}(u) + N_{2,2}(u)}{2}\,\boldsymbol{P}_1 + \frac{N_{2,2}(u)}{2}\,\boldsymbol{P}_3 \tag{20}$$

The process is repeated for all arcs and only n control points are left, which will be referred to as $\boldsymbol{Q}_i$. The resulting formulation is

$$\boldsymbol{c}(u) = \sum_{i=1}^{n} \phi_i(u)\boldsymbol{Q}_i \tag{21}$$

with

$$\phi_i = \begin{cases} \frac{1}{2}N_{2,2}\left(\frac{u-u_{i-2}}{u_{i-1}-u_{i-2}}\right), & u_{i-2} \le u < u_{i-1} \\ \frac{1}{2}\left(N_{0,2}\left(\frac{u-u_{i-1}}{u_i-u_{i-1}}\right) + 2N_{1,2}\left(\frac{u-u_{i-1}}{u_i-u_{i-1}}\right) + N2,2\;\left(\frac{u-u_{i-1}}{u_i-u_{i-1}}\right)\right), & u_{i-1} \le u < u_i \\ \frac{1}{2}N_{0,2}\left(\frac{u-u_i}{u_{i+1}-u_i}\right), & u_i \le u < u_{i+1} \\ 0, & otherwise \end{cases} \tag{22}$$

Replacing u with the normalized value is needed because the original rational, quadratic Bezier arc is defined on the domain $u \in [0,1]$ where now multiple arcs are used. As the rational Bezier circle is constructed from Bezier arcs, the basis functions for the circle satisfy many of the properties that were applicable to the Bezier arc basis functions. Of the most importance are:

Partition of Unity: $\sum_{i=1}^{n} \phi_i\,(t) = 1$

Affine Invariance: affine transformation applied to circle is done by displacing control points

Smoothness: curve is C^1 over the entire domain

Convex Hull Property: the curve generated by control points is contained within the hull formed by those control points

Locality: basis functions are nonzero over three consecutive know spans

Tangent: tangent of curve knot values is parallel to vector connecting adjacent control points

Harmonic Mapping on a Unit Disk

With the boundary represented by the reduced Bezier circle in the manner above, any point on the circle can be represented as a convex combination of control points. The wish is then to extend that boundary representation to the interior of the circular element. The results in a circular domain that is completely characterized by boundary control points only. The computer graphics field utilizes a number of techniques similar to this quite often for boundary data interpolation or graphic morphing. Some common techniques include barycentric coordinates, mean-value coordinates, and harmonic coordinates.

Barycentric and mean-value coordinates are used for the parameterization polygonal elements. When using Barycentric coordinates, points within the interior of a convex polygonal element are determined from point masses placed at each of the vertices. With $v_0, v_1, \ldots, v_k$ as points in a plane forming a star-shaped polygon,

$$\sum_{i=1}^{k} \lambda_i v_i = v_0 \tag{23}$$

$$\sum_{i=1}^{k} \lambda_i = 1 \tag{24}$$

where λ_i are the barycentric coordinates of v_0 [21]. Barycentric coordinates are only C^1 and only well-defined and positive when the geometric mesh is convex in nature. Mean-value coordinates eliminate these drawbacks. They are a generalization of barycentric coordinates and are derived from the Mean Value Theorem. With α_i as the angle at the vertex v_0 in the triangle as shown in Figure 5, the weights

$$\lambda_i = \frac{w_i}{\sum_{j=1}^{k} w_j}, \qquad w_i = \frac{\tan(\alpha_{i-1}/2) + \tan(\alpha_i/2)}{\|\mathrm{v_i} - \mathrm{v_0}\|} \tag{25}$$

are coordinates for v_0 with respect to $v_1, \ldots, v_k$. Further derivations can be found in [21].

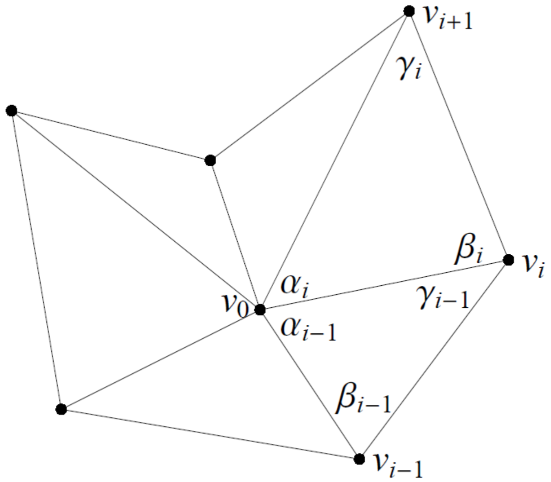


Figure 5. A star-shaped polygon

Mean value coordinates are based on straight line distance and can be negative, leading to less desirable graphic results. A better geometric representation of the interior can be generated by using harmonic coordinates, which also allows for easy interpolation of smooth domains. With solutions to Laplace's equation being termed as harmonic functions, coordinates produced as solutions to Laplace's equation are termed harmonic coordinates. These coordinates have advantages in the computer graphics application of [22]:

Increased topological flexibility

Interior locality: influences fall off with distances within the convex hull

No counter motion: ability to retain positiveness even in strongly concave situations

Interior control: option for additional vertices, edges, and faces to be added

To parameterize the interior using harmonic mapping, the wish is to be able to express any interior point ($\boldsymbol{\xi}$) as a function of the boundary control points with the expression

$$\boldsymbol{\xi} = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{T}_{\mathrm{i}}(\boldsymbol{\xi})\mathbf{Q}_{\mathrm{i}} \tag{26}$$

In the foregoing discussion about techniques used in the computer graphics field, $\mathbf{Q}_{\mathrm{i}}$ is typically a physical point located on the boundary whereas in the present case coordinates reside off of the physical boundary. Here the basis $\mathrm{T}_{\mathrm{i}}(\boldsymbol{\xi})$ is the solution of the Laplace equation and these bases are constructed using harmonic coordinates.

In this work, harmonic coordinates were selected because it is most useful for the interpolation of smooth domains. The coordinate functions are constructed in a circular domain and are then applied to other convex domains isoparametrically. With the displacement boundary data given, the solution of the Dirichlet problem provides an interpolation of boundary data. The Dirichlet problem can be explicitly solved for on the unit disk with the implementation of Poisson's integral formula. Following the notation in [20], with the boundary data described by $u = \bar{u}(\boldsymbol{\eta})$ on the unit circle $C\colon |\boldsymbol{\eta}| = 1$, the solution is given by

$$u(\boldsymbol{\xi}) = \int_{\eta \in C} H(\boldsymbol{\xi}, \boldsymbol{\eta})\, \bar{u}(\boldsymbol{\eta}) ds_{\eta} \tag{27}$$

here

$$H(\boldsymbol{\xi}, \boldsymbol{\eta}) = \frac{1}{2\pi} \frac{1 - |\boldsymbol{\xi}|^2}{|\boldsymbol{\eta} - \boldsymbol{\xi}|^2} \tag{28}$$

Here $H(\boldsymbol{\xi}, \boldsymbol{\eta})$ is known as Poisson's kernel which has the following three important properties [20]:

Partition of unity: sum of all function values is 1 for all values of ξ in the domain, i.e. $\int_{\eta\in C} H(\boldsymbol{\xi},\boldsymbol{\eta})\, ds_{\eta} = 1$.

Linear consistency: linear distribution prescribed on the boundary uniquely results in a linear distribution in the domain, that is $\boldsymbol{\xi} = \int_{\eta\in C} H(\boldsymbol{\xi},\boldsymbol{\eta})\boldsymbol{\eta}\, ds_{\eta}$.

Kronecker delta property: boundary representation is preserved for the interior, i.e. $\int_{\eta\in C} H(\boldsymbol{\xi},\boldsymbol{\eta})\, \bar{u}(\boldsymbol{\eta}) ds_{\eta} = \bar{u}(\boldsymbol{\xi})$ when $\boldsymbol{\xi} \in C$.

Applying Poisson's kernel and the corresponding properties to our particular case, the harmonic basis functions can be derived. The boundary values are described by

$$\boldsymbol{\eta} = \boldsymbol{c}(t) = \sum_{\mathrm{i}=1}^{\mathrm{n}} \phi_{\mathrm{i}}(\mathrm{t})\mathbf{Q}_{\mathrm{i}} \tag{29}$$

Thus from the linear consistency property,

$$\boldsymbol{\xi} = \int_{\eta\in C} H(\boldsymbol{\xi},\boldsymbol{\eta}) \sum_{\mathrm{i}=1}^{\mathrm{n}} \phi_{\mathrm{i}}\mathbf{Q}_{\mathrm{i}}\ ds_{\eta} = \mathrm{T}_{\mathrm{i}}(\boldsymbol{\xi})\mathbf{Q}_{\mathrm{i}} \tag{30}$$

with

$$\mathrm{T}_{\mathrm{i}}(\boldsymbol{\xi}) = \int_{\eta\in C} H(\boldsymbol{\xi},\boldsymbol{\eta})\, \phi_{\mathrm{i}}\, ds_{\eta}. \tag{31}$$

Therefore utilizing the derived bases above, a point inside the circle can be described as a linear combination of the boundary control points. Since the bases $\mathrm{T}_{\mathrm{i}}(\boldsymbol{\xi})$, is the solution to Laplace's equation, it follows that they may also be termed harmonic basis. These derived bases also have the following properties:

Smoothness: bases are infinitely smooth for all values within the domain, i.e. $\mathrm{T}_{\mathrm{i}}(\boldsymbol{\xi}) \subset \mathrm{C}^{\infty}$

Positiveness: no negative coordinates, i.e. $\mathrm{T}_{\mathrm{i}}(\boldsymbol{\xi}) \geq 0$

Partition of Unity: sum of all function values at each interior point is 1, i.e. $\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{T}_{\mathrm{i}}(\boldsymbol{\xi}) = 1$

Affine Invariance: any affine transformation to interior points is applied by displacing the boundary control points

In the same manner physical isoparametric elements can be mapped from the circular harmonic element to the physical domain. If the boundary of the physical element is described by the Bezier form

$$\boldsymbol{x}(u) = \sum_{i=1}^{n} \phi_i(u)\boldsymbol{P}_i \tag{32}$$

where P_i are the physical control point coordinates. The harmonic basis functions are used to achieve

$$\boldsymbol{x} = \sum_{i=1}^{n} \mathrm{T}_i(\boldsymbol{\xi})\boldsymbol{P}_i \tag{33}$$

where the interior of the circular element is mapped smoothly to the physical element. A visual description can be seen in Figure 6. It is important to note that this mapping in its current state is only valid for convex-shaped physical elements.

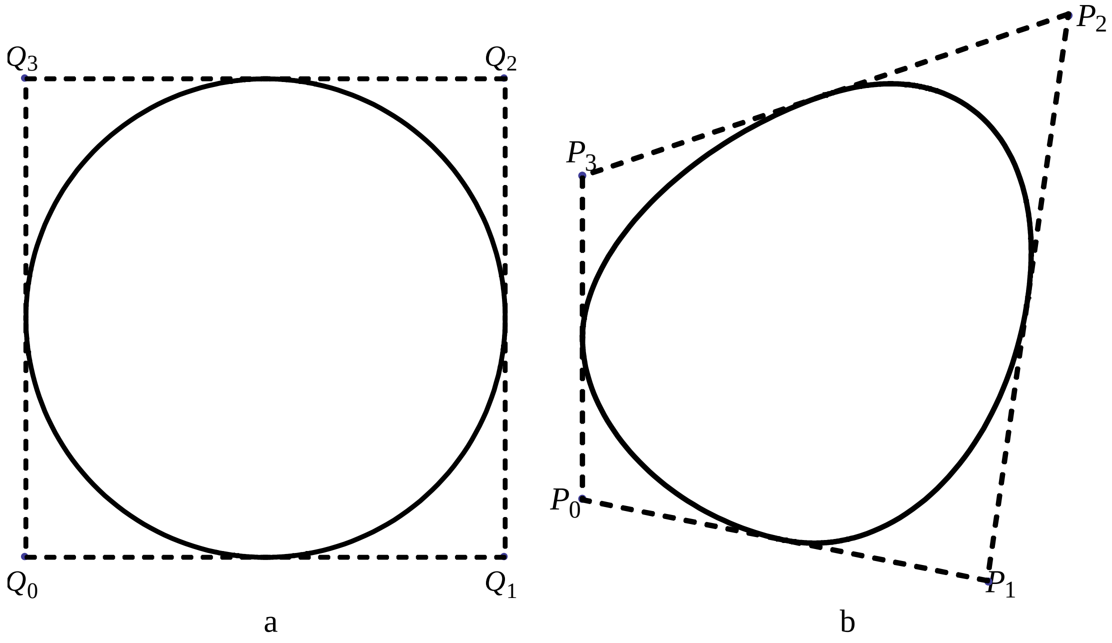


Figure 6. Isoparametric element mapping example
(a) Master element
(b) Physical element

Tensor-Product Rod

After utilizing Bezier circles for the cross-sectional representation, a NURBS centerline is used to add the longitudinal component into the 3D rod representation, formally known as the tensor-product rod. As derived in the previous section, the method assumes that the cross-sections of the rod are parameterized using harmonic mapping of a Bezier circle with n control points. The centerline of the rod-like structure can be described by a NURBS curve with m control points. In this method each longitudinal control point is replaced by a harmonically mapped cross-section. Combining in this manner gives the tensor-product rod formulation

$$\boldsymbol{x} = \sum_{i=1}^{n} \sum_{j=1}^{m} T_i(\xi^1, \xi^2) R_j(u) \boldsymbol{P}_{ij} \tag{34}$$

with $T_i(\xi^1, \xi^2)$ as the cross-sectional representation and $R_j(u)$ as the axial representation. The domain in this case is defined by

$$\Omega = D \otimes \left[u_1, u_{m+p+1}\right] = \{\xi^1 \otimes \xi^2 \otimes u : (\xi^1)^2 + (\xi^1)^2 \leq 1,\ \ u_1 \leq u \\ \leq u_{m+p+1}\} \tag{35}$$

which is that of a cylindrical element with a NURBS knot span of $[u_1, u_{m+p+1}]$.

Cylindrical Element

The tensor-product rod drives the geometric basis for the analysis of a cylindrical element. The reference (undeformed) geometry and current (deformed) geometry can both be described by the same basis functions. The current geometry is determined by arbitrarily displacing the control points of the reference configuration. Using the tensor-product rod formulation, a 3D solid is naturally subdivided into multiple elements much like that in FEA. In the present case the cylinder is parameterized longitudinally by a

NURBS centerline of degree p; therefore each segment of the centerline can be thought of as an individual element. This divides the cylinder into sub-cylinders where each sub-cylinder corresponds to a non-empty knot span of $[u_i, u_{i+1}]$. With n control points per reduced Bezier cross-section and a centerline of degree p, each element in this case contains $n \times (p + 1)$ control points. An example of a portion of an undeformed tensor-product rod is shown in Figure 7a. The NURBS centerline is of degree p = 2. Therefore this cylinder contains two elements as shown in the physical mesh of Figure 7b. With the model having four control points per cross-section, this leads to 12 control points being utilized for each of the two elements.

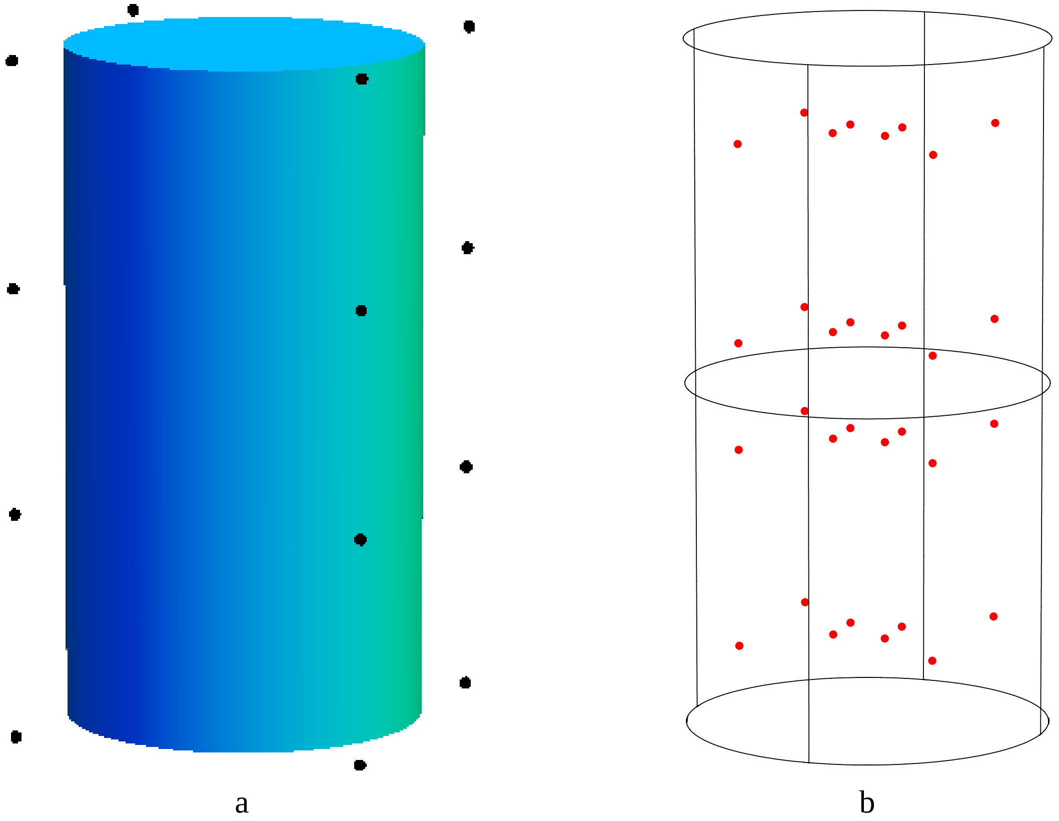


Figure 7. Master element definition example
(a) Cylindrical rod with control point locations
(b) Gauss point locations for 4 x 2 x 2 quadrature

Following common FEA practice, integration over the rod-element is required to compute the weak form. To do so the following quadrature rule was implemented as described in [1]. First the cylindrical domain is divided into n angular slices of $\frac{2\pi}{n}$, with n representing the number of control points of the reduced Bezier circle. Each slice will then be assigned Gauss quadratures utilizing cylindrical-polar coordinates. Integration of a function F is computed by

$$\int_{\Omega} Fdv \approx \sum_{i=1}^{n} \left[\sum_{j=1}^{r} \sum_{k=1}^{s} \sum_{l=1}^{t} F(\xi_{ijk}, u_l) r_{ij} w_{rj} w_{\theta k} w_{zl} \right] \tag{36}$$

Here r, s and t represent quadrature points in the radial, angular, and longitudinal direction, respectively. Also (ξ_{ijk}, u_l) is the position of a gauss point and r_{ij} is the corresponding radius. Weights for the r, θ, and z directions are w_{rj}, $w_{\theta k}$, and w_{zl}, respectively. This quadrature rule is denoted as $n \times r \times s \times t$. As an example, the following configuration was utilized for the cylinder shown in Figure 7a: $r = 2$, $s = 1$, and $t = 2$. The gauss point locations are shown along with the physical mesh in Figure 7b.

Geometric Accuracy

As mentioned before, one of the major benefits of using isogeometric analysis is the fact that the geometric representation used for analysis is exactly the same as the CAD model it was developed from. In fact this is where the term isogeometric analysis originated from. With traditional FEA techniques, approximations or meshed models are used to conduct analysis. It can be shown that the tensor-product rod method does indeed follow the isogeometric trend by providing a much more accurate model for analysis.

The following is a comparison between an FEA meshed object compared to a tensor-product rod isogeometric object. A simple baseball bat was modeled using Rhinoceros, a NURBS CAD modeling software, with 32 control points per cross-section. An IGES file of the model was exported for analysis. This file was then uploaded in ABAQUS for FEA mesh construction and Mathematica for tensor-product rod pre-analysis. The baseball bat was meshed in ABAQUS using 2,677 tetrahedral elements and 742 nodes. The baseball bat in Mathematica code was broken down into 20 elements by definition as determined from the 736 total control points. Gaussian quadrature integration was conducted using the configuration $r = 2$, $s = 1$, and $t = 2$, for a total of 128 quadrature points per element. The corresponding volume from the meshed FEA object was 114.4 in^3 while that from the tensor-product rod was 122.2 in^3. The actual volume of the baseball bat is 121.8 in^3 as determined from Rhinoceros, leading for an error of 6.08% in FEA and 0.328% in the tensor-product method. Therefore it is easily shown that the FEA meshed object provides low accuracy volumetric results when compared to the tensor-product rod. This means that the ensuing analysis will also be affected by this approximation. Furthermore, the tensor-product rod was constructed using a significantly less number of elements and control points compared to the elements and nodes utilized in the FEA meshed model. A geometric comparison of the two methods is shown in Figure 8.

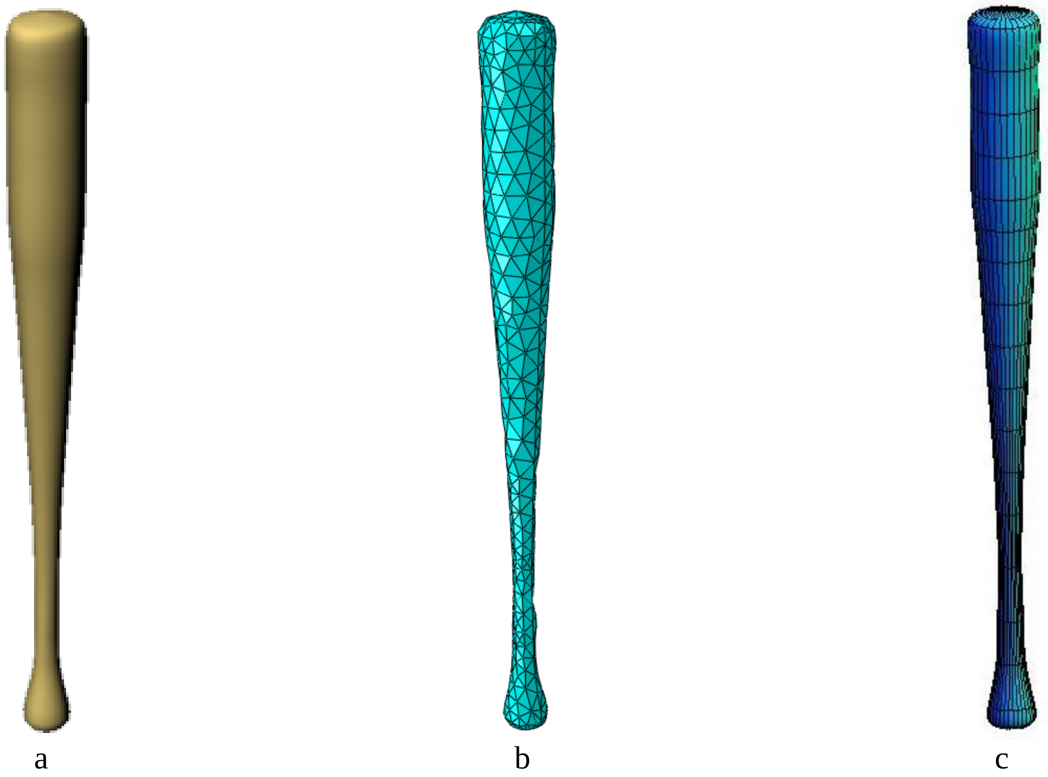


Figure 8. A geometric comparison of a baseball bat model
(a) Rhinoceros CAD software
(b) ABAQUS FEA analysis program
(c) Mathematica tensor-product analysis code

CHAPTER II

APPLICATION TO SKELETAL MUSCLE MODELING

The tensor-product method as previously mentioned is ideally suited for the skeletal muscle modeling application. There are a few characteristics of the geometric representation that make it a suitable candidate. First, skeletal muscle bodies are generally referred to as long, slender rod-like bodies which provide an easy mapping to the physical domain. Secondly, the globally smooth nature of the rod in the axial direction allows for easy, smooth parameterization of muscle fibers and assignment of a smooth material property distribution. Before describing these unique contributions, a brief background on pertinent skeletal muscle information is given.

Skeletal Muscle Background

Muscle Matter

Skeletal muscle is the basic machinery that generates contractions and produces driving forces in the human body. Approximately 80% of muscle matter consists of two types of contractile proteins, actin and myosin. The arrangement of these two proteins in a highly ordered array allows for the generation of force and which produces movement. The thick myosin filaments are arranged so that thin actin filaments can slide between them. This is termed as the sliding filament theory of muscle movement [10]. Groups of about 100 to 400 of these filaments are combined to form a myofibril. Each myofibril is enveloped in a membranous bag, called the sarcoplasmic reticulum and theses membranous bags are bound transversely by a network of branching T-tubules. Bundles of these fibers then combine to form the overall skeletal muscle shape that is commonly known. Other elements within a skeletal muscle are fibroblasts, which secrete collagen fibers to form a thin connective tissue matrix, the endomysium. A thicker layer of connective tissue called the perimysium also surrounds groups of 10 to 100 muscle fibers to form fascicles. Small blood vessels and motor axons also traverse the space to make connections with muscle fibers which add to the overall contribution of passive ground materials present within skeletal muscles [10].

Tendons serve as the joining material connecting bones to skeletal muscles. They transmit the forced produced by the muscles to the respective bone it is attached. Tendons are tough parallel bands containing arrays of collagen fibers and their stiffness is much greater than that of muscle fibers. Functionally, they act somewhat similar to a spring, storing and recovering energy while also protecting fibers against high velocities [10].

Classifications of Contractions

There are few different types of voluntary muscle contractions. If muscle tension is developed and a load is not moved, the contraction is called isometric [23]. This occurs when a muscle is attempting to move a load that it is not capable of and no external work is done. In this case, the muscle length does not change. Isometric contractions are a prerequisite of isotonic contractions. Isotonic contractions occur when a load is moved and muscle shortening takes place [23]. In isotonic contractions, the tension remains fairly constant after the initial tension is developed to move the load. Shortening then stops when the active tension drops to the point where it equals the load and the contraction again becomes isometric. A concentric contraction is when the force generated overcomes the applied resistance and muscle shortening occurs while generating a force. Eccentric contraction is elongation of muscle because the force generated is insufficient to overcome the resistance. Another type of contraction is the isokinetic contraction in which the velocity remains constant while the force is allowed to vary [23].

Muscle Fiber Orientation

As mentioned, fiber orientation is one of the primary determinants of the force capabilities of individual skeletal muscles. There are three general fiber architecture types common in skeletal muscles: (1) parallel-fibered, (2) Convergent-fibered, and (3) pennate-fibered muscles. Parallel-fibered muscles have orientations which are directed along the longitudinal direction or line-of-action of the muscle. They generally have one origination point and one termination point. Convergent fibers generally follow the longitudinal axis of the muscle but have multiple attachment points on one end which converge to a much smaller tendon attachment location at the other end. Pennate-fibered muscles have one or more tendons running through the bulk of the muscle body and fibers are attached to these tendons(s) at a distinct angle relative to the line-of-action of the muscle. Pennate-fibered muscles can be classified further into unipennate-, bipennate-, or multipennate-fibered muscles, which have one, two, and multiple distinct fiber directions, respectively.

The arrangement of muscle fibers in skeletal muscle is extremely critical for the determination of a muscle's range of motion and power. In general, the closer the fiber direction is to the muscle's longitudinal axis, the more a muscle can shorten. It follows that the more a muscle can shorten, the lower the force generating potential. Therefore muscles with parallel arrangements have the greatest shortening potential and least force generating potential. Muscle power rather is directly proportional to the total number of fibers in the muscle. Therefore the fascicles in bipennate and multipennate types of muscle pulls on the tendons at an angle therefore they do not shorten as much as their parallel muscle counterparts [24]. Despite this they generate greater tension because they possess a greater amount of muscle fibers than similarly sized parallel muscles. Convergent arrangements allows for a more versatile movement but the fibers cannot produce as much force as paralleled-fibered muscles because the fibers are not all pulling in the same direction [24].

Geometric Muscle Model of Male Human Arm

The aim of the current research again is to create an analysis-ready muscle model of the entire male human arm. There are currently a large number of muscle models available but most are polygonal in nature and were produced only for visual purposes. In this work an entire muscle and bone model of a male human arm was constructed entirely from NURBS geometry. This was done using the NURBS-based 3D CAD software, Rhinoceros, and data from the Visible Human Data Set collected by the National Library of Medicine in 1994 [25]. To generate the data set, a male cadaver was encased and frozen in a gelatin and water mixture and then cut in the axial plane at 1 millimeter intervals. This resulted in 1,871 slices that were sequentially photographed.

Utilizing these muscle cross-sections, each specific muscle’s boundary was traced out. An excerpt from the data set is shown in Figure 9 with a cross-sectional breakdown of the different muscles.

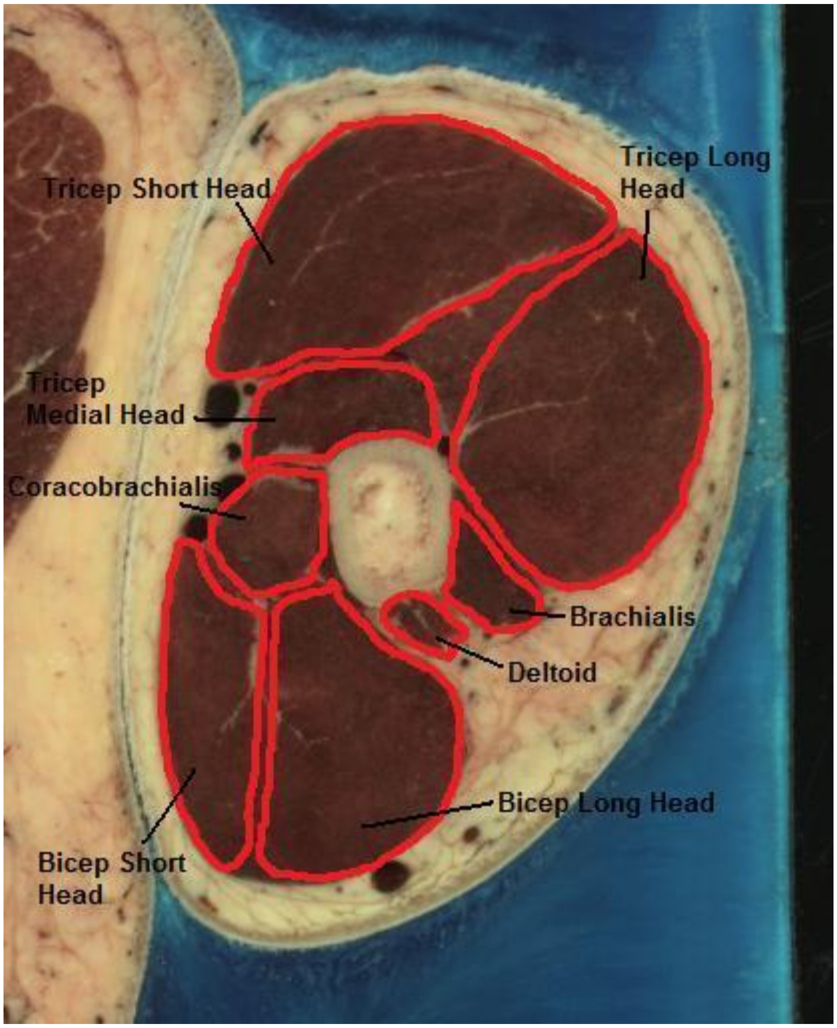


Figure 9. Cross-section of the upper male arm from the Visible Human Data Set with labeled muscles [25]

In this work, cross-sections were traced out at approximately 10-15 millimeter intervals. Solid models were then generated by compiling all cross-sections for each specific muscle and lofting them into a NURBS surface model. The cross-sections were each rebuilt with 8 control points and skeletal muscle shapes were terminated near their tendon/bone origination/termination locations. At these endpoint locations, 8 point objects are positioned in a single location. This was a necessary feature to create a complete-looking surface model and may need to be altered or neglected to avoid singularity errors in future analysis. Each model contains a varied number of cross-sections and total number of control points, which depends on the overall length of each individual muscle. The arm model can be seen as a rendered Rhinoceros object in Figure 10. All 28 muscles from the deltoid down to the wrist were modeled.

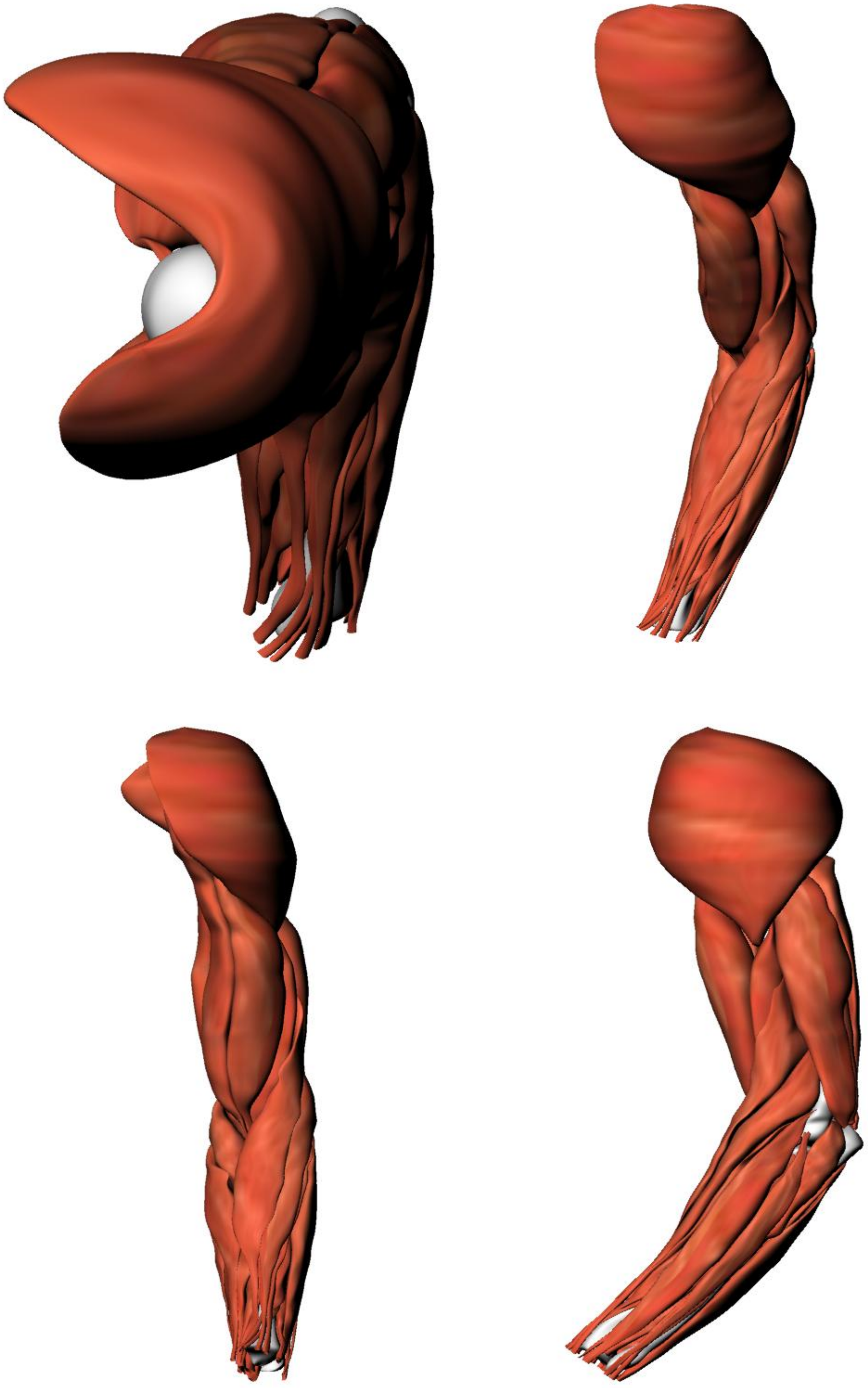


Figure 10. Rhinoceros rendered images of the complete NURBS-based arm muscle model

Each individual muscle was exported as an IGES file and imported to a Fortran IGES reader. The reader did not directly utilize the parameters from the Rhinoceros surface model. Rather, it extracted the model's control point coordinates. This geometric data was then introduced into Mathematica code for further analysis. Utilizing the control point information, the NURBS model was then converted to fully defined tensor-product model with cross-sectional representations, control point locations, and NURBS centerline described by a $p = 2$ degree NURBS curve with respective knot vector and weighting values. The geometric solid model was then fully defined and ready for fiber parameterization and material property assignment.

Parametric Modeling of Fiber Orientation

Fiber parameterization in this work through the tensor-product rod framework is much more straightforward than previous methods which have been utilized through FEA. The fiber direction can be implemented directly within the rod structure utilizing the parametric nature of the axial knot vector. The idea is to define the fiber lines in the master rod, and then pass them through to the physical geometry using the mapping technique discussed earlier. This method is illustrated in Figure 11.

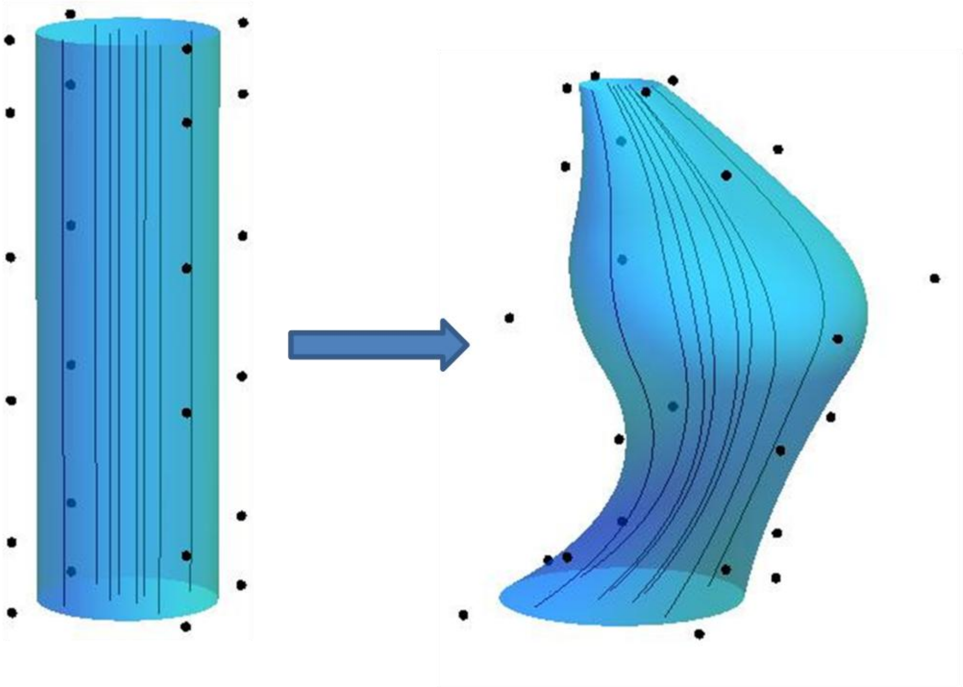


Figure 11. Mapping of fibers from the master rod to the physical rod

In this way the parameterization is universal in nature and can be applied to any muscle shape that has a similar fiber structure. There is also no need for a specific ordered arrangement of elements or a completely separate fiber mapping approximation like many of the FEA techniques in use today. The following are a few example fiber parameterizations developed that can be applied to the master tensor-product rod to accurately represent realistic fiber directions as determined visually from cadaver dissections [26].

Parallel Fibers

Within the tensor-product rod framework, parallel fibered architecture can be easily implemented. In the master rod, parallel fibers are modeled as straight lines parallel to the longitudinal axis. The two coordinate directions associated with the cross-sectional representation of the tensor-product rod method (ξ^1, ξ^2) are fixed. Utilizing the parametric nature of the centerline represented by a NURBS curve, the variable u can be traced out in the longitudinal direction. Thus, in the physical domain, the fiber curve is represented by

$$x(u) = T(\xi^1, \xi^2)R(u)P \qquad u_1 \leq u \leq u_{m+n+1} \tag{37}$$

where again the two cross-sectional coordinates are fixed. The tangent of this curve at any point defines the fiber direction. In particular, fiber directions for points of interest within the tensor-product rod, i.e. the Gaussian integration points, can be determined. An example of a parallel fibered muscle is the extensor carpi ulnaris, a strap-like muscle located on the back of the forearm which assists in adduction and extension of the wrist. The muscle fiber orientation can be seen visually in the pictorial representation shown in Figure 12a. The tensor-product muscle model is shown in Figure 12b plotted utilizing the parallel fiber parameterization.

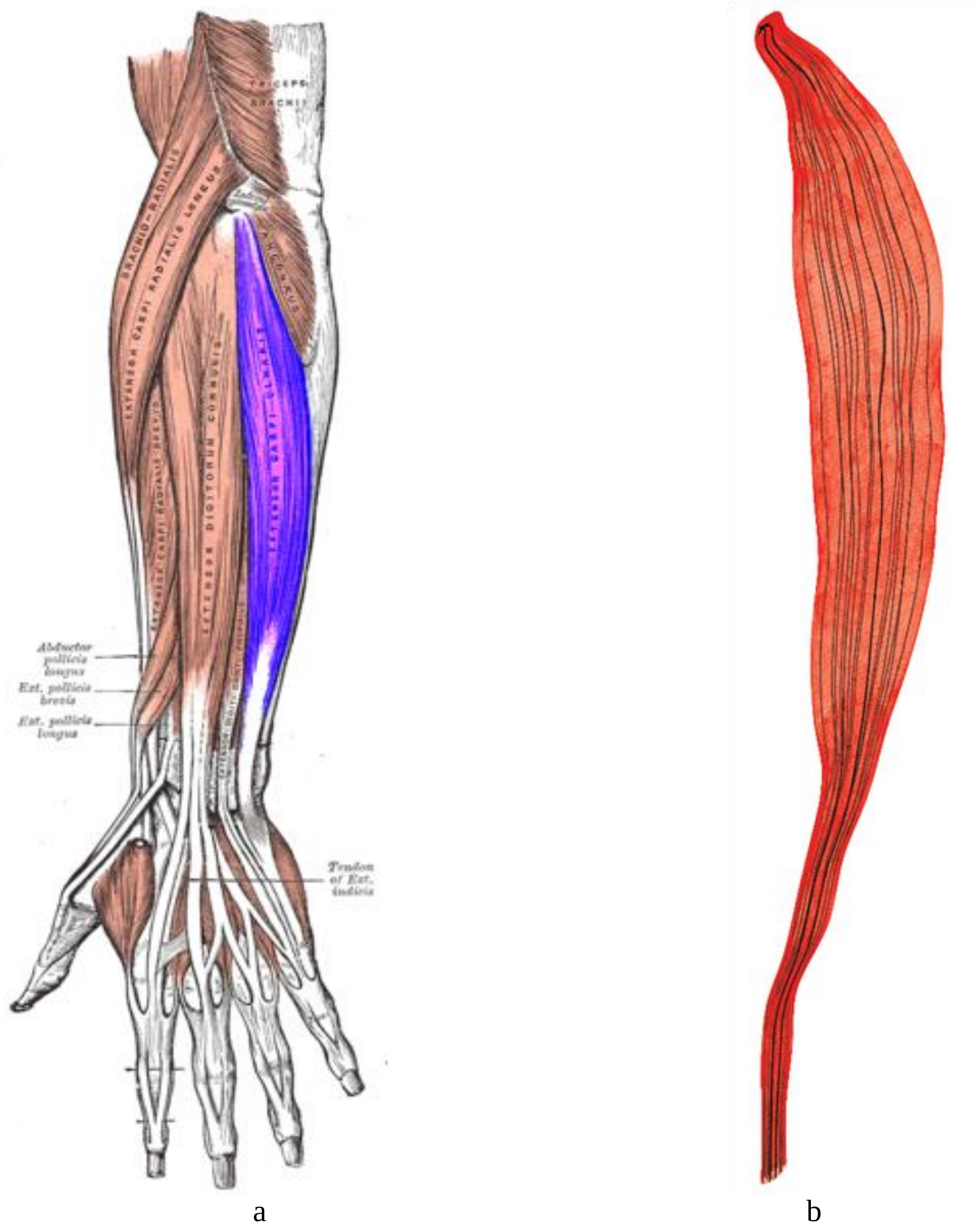


Figure 12. Parallel-fibered extensor carpi ulnaris
(a) Actual muscle location and fiber direction [27]
(b) Tensor-product rod model and fibers

Helical Fibers

The human body is a very complex system and as one could imagine, not all muscles have fiber orientations that behave in a perfectly parallel-fibered manner. As it is extremely important to model fiber orientation as closely as possible to mimic actual muscle fiber behavior, these small differences should be accounted for to ensure the best possible accuracy. One minor deviation is the twisting of parallel-fibered muscles along their longitudinal axis. Again the parametric nature of the tensor-product rod easily allows for this type of behavior to be implemented. A fiber orientation that is helical in nature can be represented by

$$x(u) = T(r\cos(\alpha u + \emptyset)\,, r\sin(\alpha u + \emptyset))R(u)P, \qquad u_1 \leq u \leq u_{m+n+1} \qquad (38)$$

with $r = \sqrt{(\xi^1)^2 + \ (\xi^2)^2}$ and $\emptyset = \tan^{-1}(\xi^2/\xi^1)$. Here α is a constant that can modify the severity of the helical twist and twist direction. Increasing α will produce more coils per unit length while decreasing will produce fewer. The helical direction is reversed by the use of a negative constant. Due to the longitudinal division into knot spans of the tensor-product rod, this type of representation can be implemented for an entire skeletal muscle or in conjunction with another type of fiber parameterization such as the parallel-fibered case. An example of a muscle with fibers that behave in this manner is the brachioradialis shown in Figure 13a, which is located mainly in the forearm wraps around the outside of the elbow to help with flexion as well as pronation and supination of the forearm. The brachioradialis model is shown in Figure 13b with a parallel fiber parameterization for the majority of muscle length and a helical-fiber parameterization applied where the muscle wraps around the brachialis muscle just above the elbow.

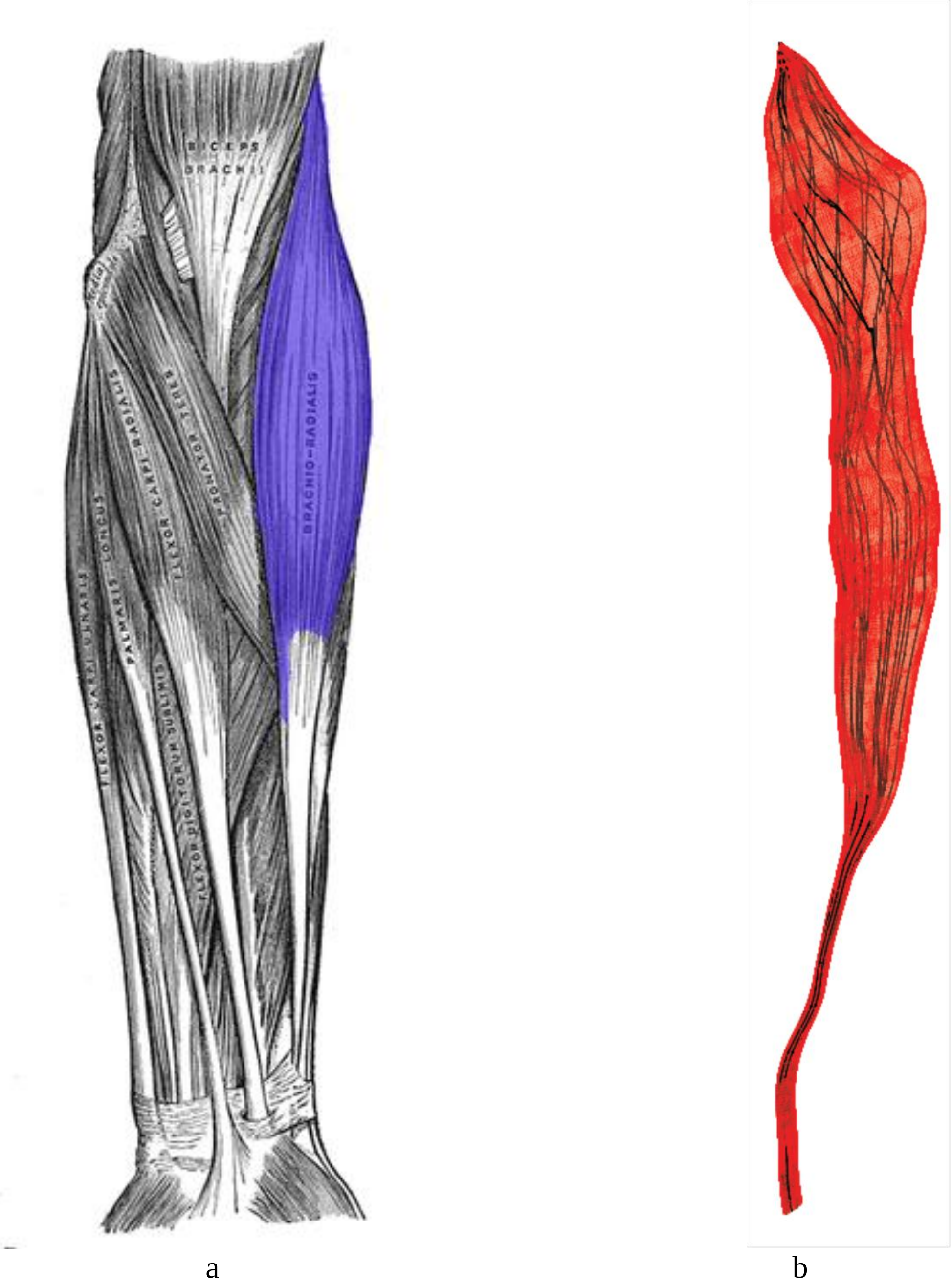


Figure 13. Helical-fibered brachioradialis
(a) Actual muscle location and fiber direction [27]
(b) Tensor-product rod model and fibers

Convergent Fibers

Assigning a fiber direction for convergent muscles requires yet another variation. Convergent muscles do not necessarily terminate at a common tendon location as in the parallel case, but rather they have multiple attachment points. In the muscle model created, the tensor-product solid has a single origination and termination point. This means if fiber direction were assigned along the isocurves as described in the parallel fiber case, fiber direction would converge rapidly on the origination/termination point regardless of geometric shape. To remedy this, a parameterization was used in which the fiber direction can converge or diverge from the isocurves, leading to fiber termination points at the walls of the tensor-product muscle model. The formulation for this procedure is depicted by the following representation

$$x(u) = T(\xi^1(1+au), \xi^2(1+au))R(u)P \qquad u_1 \leq u \leq u_{m+n+1} \tag{39}$$

Here the variable a is the convergence/divergence intensity of the particular skeletal muscle. Increasing a will cause the fibers to diverge radially outward from the parallel case. A negative value for a will cause fibers to converge to the NURBS centerline of the tensor-product rod shape. A boundary condition was utilized that disallows fibers from passing through the skeletal muscle's wall. An example of a muscle that is convergent in nature is the pronator teres shown in Figure 14a. The fiber convergent parameterization of this muscle model, which drives pronation of the forearm, can be seen in Figure 14b.

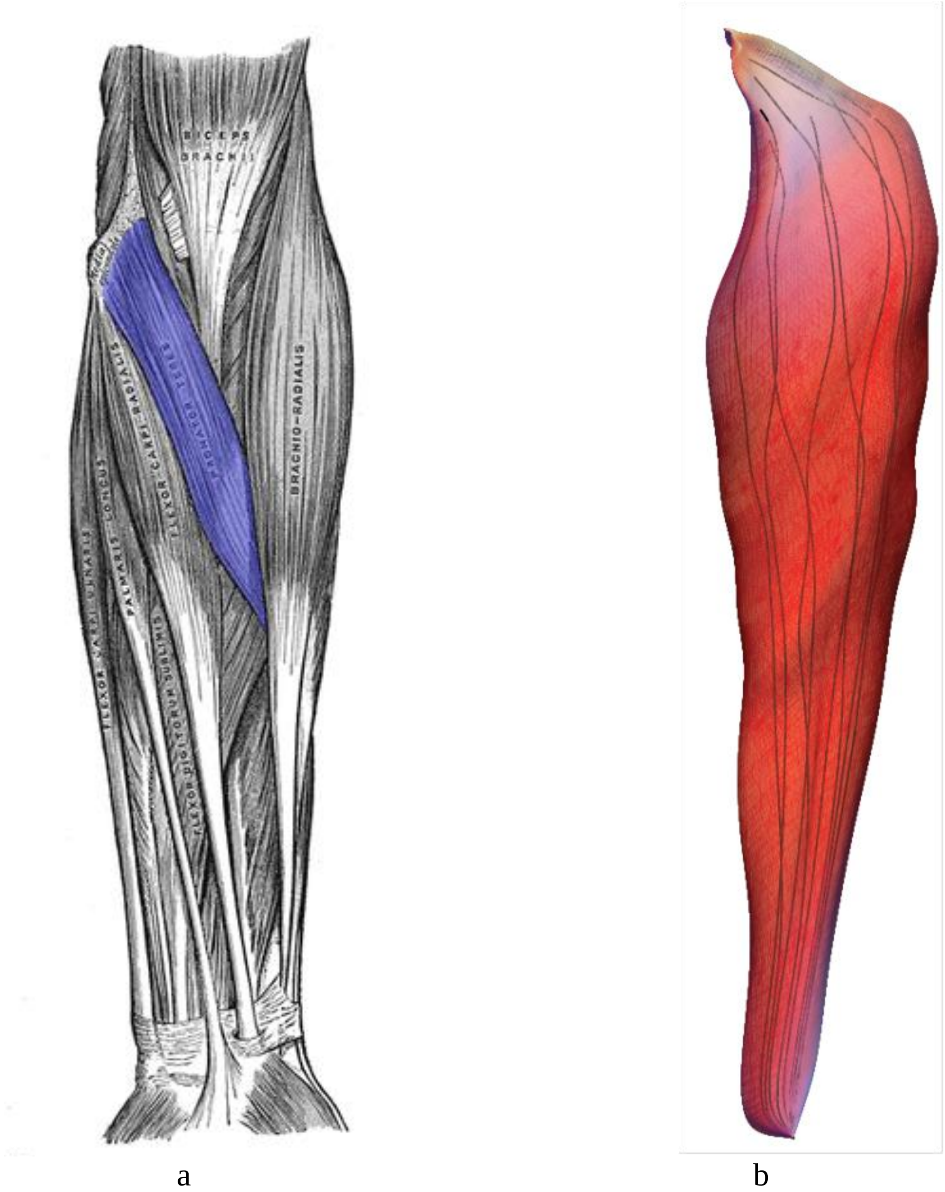


Figure 14. Convergent-fibered pronator teres
(a) Actual muscle location and fiber direction [27]
(b) Tensor-product rod model and fibers

Pennate Fibers

Pennate fibers are yet another common orientation type for skeletal muscles. Pennate-fibered muscles have one or more tendons running the through the bulk of the muscle body and the fibers are attached to the tendons at a distinct angle relative to the line-of-action of the muscle. A pennate fibered muscle represented in the tensor-product rod domain can be defined by

$$x(u) = T(b(u - u_o) + c, \xi^2)R(u)P, \qquad u_1 \le u_i \le u_{m+n+1} \tag{40}$$

with $c = \xi^1 - bz$. The degree of pennation is determined by the slope factor b and z is the longitudinal location of the point which fibers are running through in each element. Switching signs of the slope factor will switch the pennate direction. Of course pennation can also be implemented in the ξ^2 direction, as well as any combination of the two. An example of a pennate fibered muscle is the anconeus muscle shown in Figure 15a. This smaller muscle located on the posterior of the elbow joint which supports elbow extension. The anconeus tensor-product muscle model and pennate fiber parameterization is shown in Figure 15b.

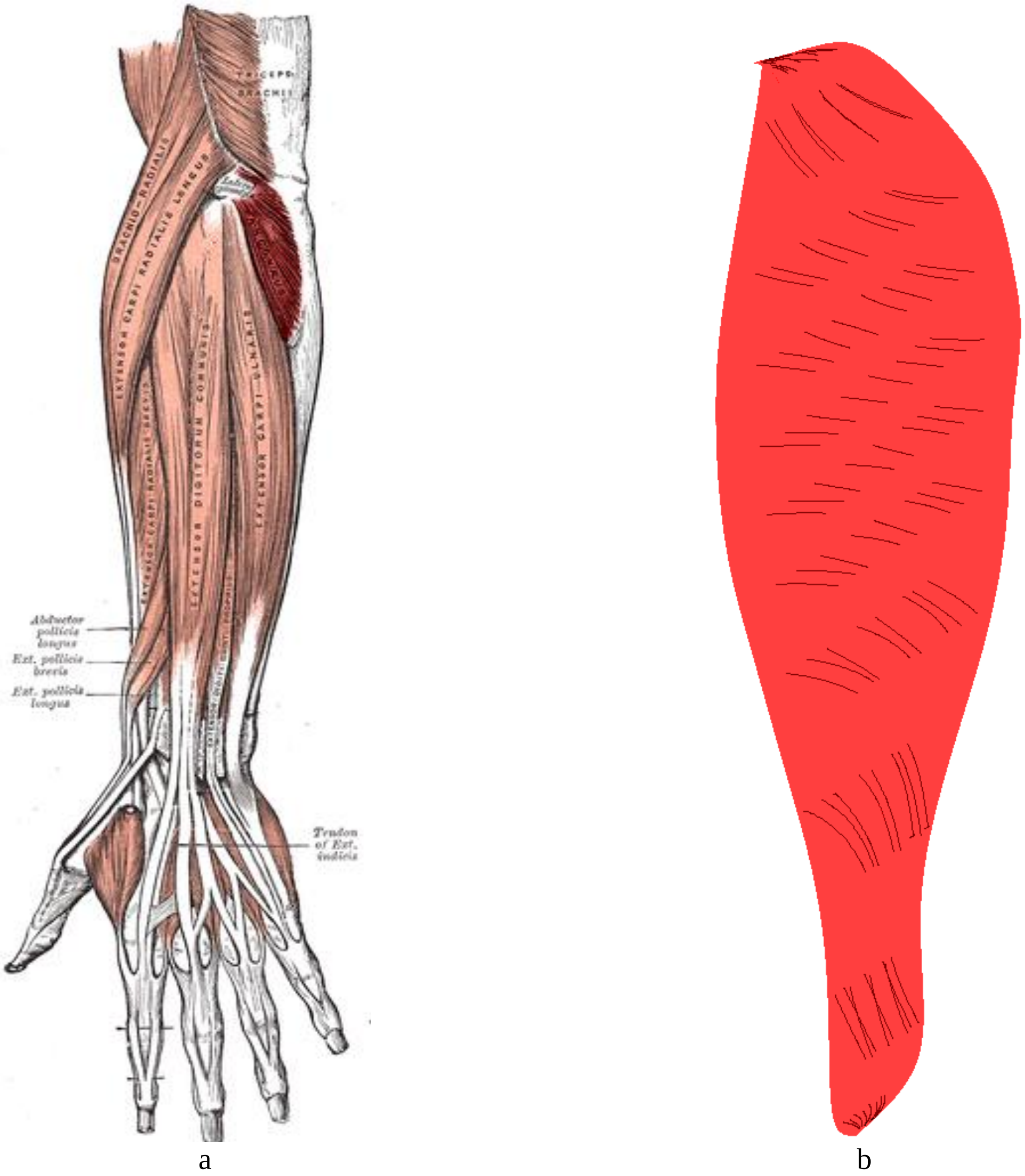


Figure 15. Pennate-fibered anconeus
(a) Actual muscle location and fiber direction [27]
(b) Tensor-product rod model and fibers

The fibers shown in the actual model do not originate and terminate on the muscle walls one would expect because the formulation does not visually pass well through to adjacent knot spans. So fibers look disconnected but the actual fiber direction at any particular point in each knot span still remains valid. If allowed the fibers would terminate at the side walls of the muscle as opposed to the ends shown in the parallel and helical cases. Due to this instability, analysis is limited to points away from the muscle boundaries. This particular formulation is a rough derivation but it shows that pennate fibered arrangements can indeed be implemented within the tensor-product framework.

Issues with Modeling Fiber Orientation

A few notable drawbacks of the previously mentioned techniques should be noted. Within previous research dedicated to the tensor-product rod, it was discovered that arbitrary elements with largely concave-shaped domains present a few issues with the current method. If a shape's cross-section is too concave, mapping from the master element to the physical element will not be accurate. This is because the mapping is not invertible over the entire domain. The determinant of the Jacobian must not be zero anywhere in the geometric domain, meaning a sign change cannot occur. This is a known drawback to the tensor-product rod method. For example, the deltoid's concavity proved to be too large to achieve one-to-one mapping from the master element to the physical element. This error in mapping of the deltoid is plotted with gauss point locations in Figure 16, with a Gaussian quadrature of $1 \ x \ 1 \ x \ 1$. Here one can easily see the gauss points lay outside the deltoid's boundary. The supinator is another muscle in the current scope of research that has a largely concave geometric domain. Thus fiber parameterizations cannot be implemented directly for these muscles using the current technique.

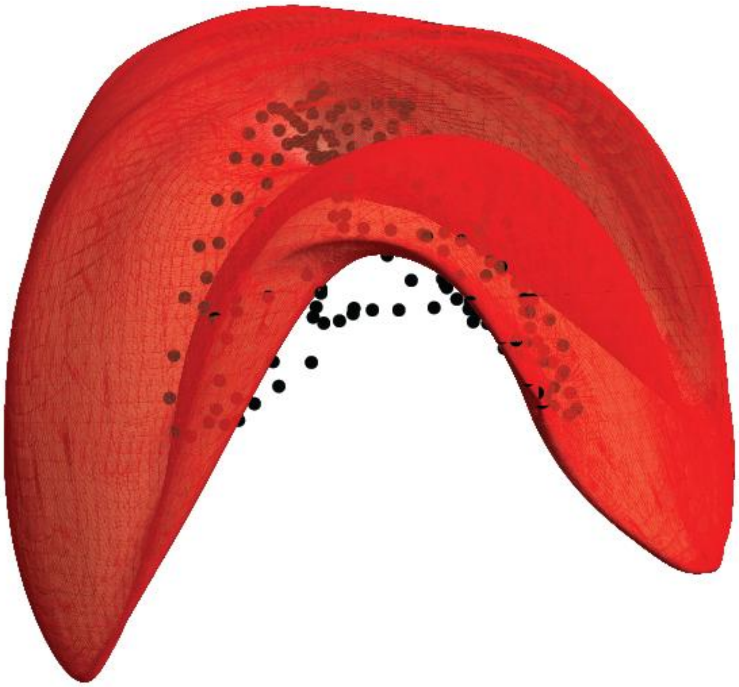


Figure 16. The deltoid muscle where mapping is not one-to-one

Another useful note is that the computational time to derive the fiber orientations can be quite high depending upon the level of accuracy one is seeking. Increasing the number of gauss points in any direction will cause the computational time to increase significantly. However, these calculations are only necessary during initial fiber parameterization. Therefore in the analysis thereafter those fiber arrangements are already stored and need not be derived again.

In addition to the computational time issues, there are also problems with determining actual fiber orientation with human muscle tissue. Ideally the best method to determine actual fiber orientation is to dissect human muscle tissue and determine pennate angles and muscle fiber types through visual inspection and measurement. Currently, most methods for determining actual fiber orientations are based on measurements from cadavers. This is limiting in a sense that it does not encapsulate a good representative sample of the entire population as most cadavers used are of an older age. MRI, ultrasonic, other innovative techniques have been utilized more and more in recent years, but they remain expensive to conduct and require a substantial amount of experimental time. These also require separate fitting techniques to approximate the fiber orientation. To date there is not a comprehensive database with explicitly listed fiber orientations for all muscles within the human body.

Many current FEA models also assume that the pennate angle is constant throughout all fibers of a skeletal muscle, which is also not an accurate assumption. Pennate angles may change during muscle contraction say from a 30 degree pennation angle from the axis at its resting state to a 15 degree pennation angle at an active state, further complicating the fiber parameterization process. This continues to be an ongoing challenge in muscle modeling. Only a select few research studies on fiber orientation have been conducted to determine fiber orientation and pennate angles of actual human skeletal muscles and have done so with varying levels of success.

Defining Heterogeneous Material Properties

Utilizing the Rod Structure

The final step in developing a completely characterized analysis-ready model is to accurately assign material properties, i.e. parameters for constitutive relations. Current FEA techniques of muscle analysis inaccurately assign muscle properties to specific elements. In most FEA analysis conducted to date on skeletal muscles, entire elements for muscle tissue and tendon properties were assigned specific property values. In actuality the muscle-tendon complex is a gradual change of properties, not an abrupt, unrealistic property change at element boundaries as depicted by most researchers. A close look at a dissection of the human arm will reveal that this is in fact the case.

Therefore muscle tissue properties can also assumed to vary longitudinally along skeletal muscle body models.

This longitudinal distribution of varying properties can be broken down in a number of ways. In this work the NURBS centerline description was utilized, more specifically the globally smooth basis functions. Recalling the description of a basic 1D NURBS curve,

$$C(u) = \sum_{i=0}^{n} N_{\mathrm{i,p}}(u) P_i$$

it is shown that a curve can be mapped out from the control point locations P_i. Using the exact same basis function distribution, an overall material property distribution of

$$a(u) = \sum_{i=0}^{n} N_{\mathrm{i,p}}(u) a_i \tag{41}$$

can be assumed. Here a_i are material components that can be arbitrarily determined to match any muscle's specific muscle-tendon transitional period in the longitudinal direction. Remembering the tensor-product formulation, each NURBS centerline control point can be thought of as being replaced by a cross-sectional representation. Thus cross-section specific material property parameters are assigned. By assigning properties in this manner with the globally smooth representation of the tensor-product model's NURBS centerline, a smooth transition at the property level is also assumed. Therefore as long as the basis functions, $N_{\mathrm{i,p}}$, are smooth the resulting property distribution will also be smooth. Application of the property distribution is done in a similar manner to the fiber parameterization technique. The properties are applied to the master rod which then mapped through to the physical rod. In this way the formulation is again easily applied and universal in nature, allowing for greater flexibility and accuracy in the ensuing model description. Of course similar techniques of applying the variations of material properties could be done radially or circumferentially at the cross-sectional level as well as the longitudinal direction, but this was deemed unnecessary for the current work.

The material property distribution is not easily represented graphically. An attempt to more clearly define the distribution is shown in Figure 17 with a visual representation of the muscle-tendon color scheme for the bicep short head. A specific color scheme was assigned to each cross-section along the muscle body using the formulation in equation (41). Here the parameters assigned to each cross-section of the master rod are the control values, a_i, of the material property distribution. In between the individual cross-sections, the NURBS basis functions essentially blend the property control values of each cross-section to ensure a smooth transition from one control value to the next. The process is repeated until the color scheme has completely transitioned from white (tendon) to red (muscle fiber) or vice versa. After assigning the material properties, the tendon-to-muscle and muscle-to-tendon transitional periods are defined from $[-66.5, -16.7]$ on the left hand side (distal end) and $[77.4, 123]$ on the right hand side (proximal end) in the physical domain. This type of implementation is possible because a NURBS element is not confined to nodes on the element boundary like an FEA element. Instead the boundaries for NURBS elements are C^{p-1} continuous, meaning control points within the NURBS framework are shared among neighboring elements, exhibiting the globally smooth nature of the tensor-product rod and delivering the heterogeneous property distribution that is desired.

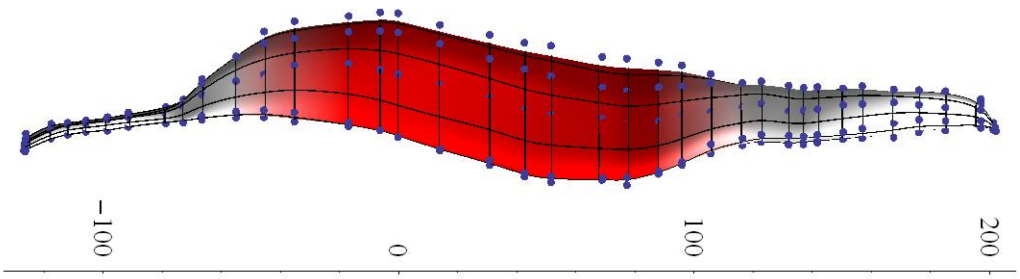


Figure 17. Visually assigned tendon/muscle fiber properties and their cross-sectional breakdown

Further refinement of the original geometry through the use of knot insertion allows for the material property distribution to be more accurately defined. Adding knots in between existing knot values of the NURBS centerline provides an additional interpolation data point that can be assigned to make the overall distribution of the muscle-tendon complex more accurately reflect the realistic nature present within cadaver studies after model creation has already been completed. Here again it is important to note that the local changes made to the axial NURBS centerline through refinement techniques do not affect the overall geometric shape. This refinement is done easily and gives great flexibility in the refinement of models regardless of the original longitudinal element breakdown.

Constitutive Model

Similar results can be expected for the assignment of muscle specific constitutive parameters. As mentioned earlier, muscle tissue is a complex material in the sense that it is active, nonlinear, incompressible, anisotropic, and hyperelasic. Thus all these material properties must be accounted for in an accurate muscle model. There are numerous mathematical models employed today but for use in this work the energy density functions developed by Oomens and colleagues in 2003 were utilized [14]. Following his formulation, muscle makeup contains of both an isotropic and anisotropic portion given by

$$W(\boldsymbol{C}) = W_{matrix}(\boldsymbol{C}) + W_{fiber}(\boldsymbol{C}). \tag{42}$$

Here $\boldsymbol{C}$ is the Cauchy stress tensor. The isotropic portion represents the ground material present such as the membranous sarcoplasmic reticulum and branching T tubules that encase muscle fibers and ensure their connectivity to one another. The ground material is assumed to take the form of a neo-Hookean material which is described by

$$W_{matrix}(\boldsymbol{C}) = \frac{\mu}{2}(I_1 - 2\ \ln J - 3) + \frac{\kappa}{2}(\ln J)^{\ 2} \tag{43}$$

where $I_1 = tr(\boldsymbol{C})$ is the first principal invariant and $J = \det \boldsymbol{F}$ is the determinant of the deformation gradient. Here μ is the effective shear modulus and κ is the effective bulk modulus. It has been shown that for nearly incompressible behavior, the bulk modulus must be approximately 1000 times that of the shear modulus [28]. In this work these material constants are assumed to be $\mu = 0.5\ kN/m^2$ and $\kappa = 10\ MN/m^2$ for the ground material in muscle tissue [13]. Tendon properties can also be modeled using the neo-Hookean formulation but with much stiffer with material constants of $\mu = 50\ kN/m^2$ and $\kappa = 100\ MN/m^2$ [13].

The actual fiber portion of the muscle tissue is anisotropic and contains both an active and passive components. This is represented by the following energy density function

$$W_{fiber}(\boldsymbol{C}) = W_{active}(\lambda, \alpha) + W_{passive}(\lambda). \tag{44}$$

The active portion is controlled by neural excitation and is dependent on the fiber stretch λ and level of activation α. It represents the interactions of the contractile proteins that drive muscle movement and produces stresses that occurs in the fiber direction. The passive portion exists because of collagen and other cytoskeletal materials present within skeletal muscles that are not affected by any type of muscle activation level [14].

Digital Muscle Library

The objective of this research is to completely characterize an isogeometric muscle model that is analysis-ready. As defined earlier, an analysis-ready model requires three essential characterizations to be fully defined. The first characterization is a geometric model that accurately reflects actual muscle anatomy and is capable of handling the structural complexities. The second characterization is an accurate representation of fiber architecture and the last characterization is an accurate representation of the heterogeneous property distribution, such as the muscle-tendon complex. It has been shown by employing new techniques through the utilization of the tensor-product rod method that these three characterizations are satisfied and implemented in a manner that exceeds the capabilities of current FEA techniques.

By implementing the analysis-ready characteristics for each of the 28 muscles in the entire model of the male human arm, a working “Digital Muscle Library” is formed. Here each specific muscle is complete with its respective geometric, fiber, and material property characterization and has that data locally stored. More specifically, the geometric description will contain all control point locations with weighting values and the knot vector that characterizes the NURBS centerline of degree of $p = 2$. Cross-sectionally the models all contain 8 control points per cross-section. Each muscle also has stored a specific fiber parameterization tailored to meet each muscle’s specific fiber architecture, i.e. any of the previously described parallel, helical, pennate, and convergent fiber descriptions or combination. All fiber parameterization constants such as the helical direction and intensity, pennate angle and direction, and level of convergence are also stored. Lastly, the muscles also have the material property values as determined by the previously described constitutive relations stored for each cross-section. More specifically, the parameters that vary in the axial direction are the two assumed material constants μ and κ. Their distribution can arbitrarily be assigned to each muscle to match its specific muscle-tendon distribution.

An example of the information stored within a muscle of the digital library can be exhibited with the abductor pollicis longus. The muscle can be seen in its complete, analysis-ready form in Figure 18.

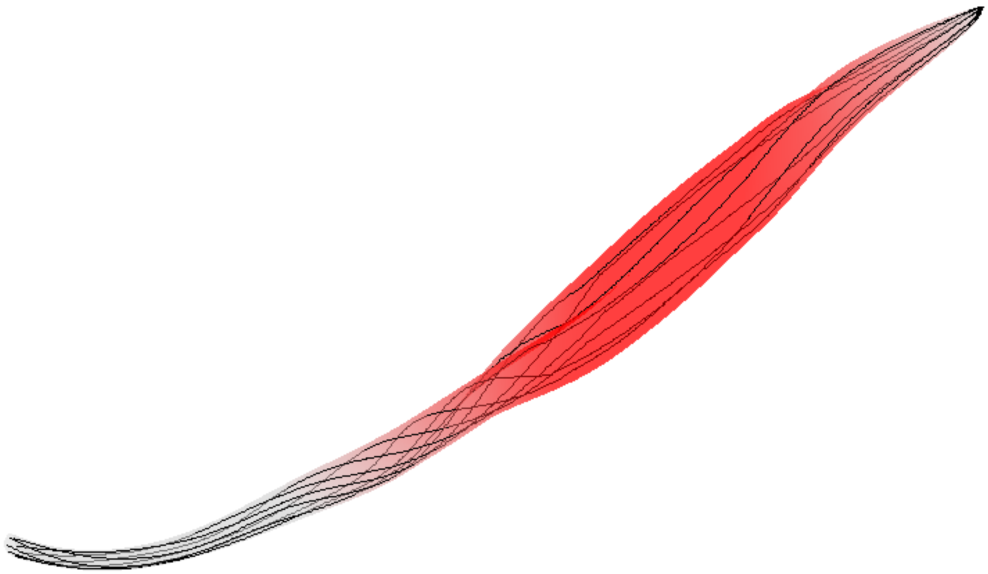


Figure 18. Analysis-ready isogeometric model of the abductor pollicis longus

The abductor pollicis longus has the following control point locations:

$$Q = \begin{pmatrix} 590.0 & 253.0 & -134.0 \\ 590.0 & 253.0 & -134.0 \\ 590.0 & 253.0 & -134.0 \\ 590.0 & 253.0 & -134.0 \\ 590.0 & 253.0 & -134.0 \\ 590.0 & 253.0 & -134.0 \\ 590.0 & 253.0 & -134.0 \\ 590.0 & 253.0 & -134.0 \\ 590.4 & 252.5 & -134.6 \\ 590.3 & 253.1 & -134.6 \\ 589.8 & 252.9 & -134.6 \\ 589.4 & 252.7 & -134.6 \\ 588.9 & 252.4 & -134.6 \\ 589.3 & 252.0 & -134.6 \\ 589.8 & 252.0 & -134.6 \\ 590.3 & 252.1 & -134.6 \\ 592.8 & 251.9 & -135.7 \\ 592.1 & 254.2 & -135.7 \\ 589.6 & 253.6 & -135.7 \\ 588.2 & 252.5 & -135.7 \\ 586.0 & 250.9 & -135.7 \\ 587.9 & 249.3 & -135.7 \\ 589.9 & 249.4 & -135.7 \\ 592.1 & 249.8 & -135.7 \\ 581.7 & 232.5 & -157.2 \\ 582.4 & 240.9 & -157.2 \\ 577.4 & 233.7 & -157.2 \\ 572.8 & 233.8 & -157.2 \\ 562.9 & 233.5 & -157.2 \\ 570.6 & 225.0 & -157.2 \\ 574.5 & 225.6 & -157.2 \\ 580.6 & 227.9 & -157.2 \\ 584.0 & 226.0 & -164.8 \\ 578.0 & 235.0 & -164.8 \\ 566.3 & 230.7 & -164.8 \\ 561.4 & 219.3 & -164.8 \\ 557.9 & 209.4 & -164.8 \\ 555.0 & 190.0 & -164.8 \\ 567.0 & 197.0 & -164.8 \\ 572.5 & 210.5 & -164.8 \\ 569.0 & 202.6 & -177.3 \\ 563.8 & 220.0 & -177.3 \\ 554.6 & 207.7 & -177.3 \\ 550.8 & 197.3 & -177.3 \\ 548.2 & 188.4 & -177.3 \\ 552.0 & 172.0 & -177.3 \\ 556.5 & 186.7 & -177.3 \\ 563.7 & 192.9 & -177.3 \\ 561.2 & 191.0 & -188.5 \\ 557.1 & 206.5 & -188.5 \\ 548.5 & 195.8 & -188.5 \\ 550.3 & 184.6 & -188.5 \\ 539.0 & 173.0 & -188.5 \\ 544.0 & 162.0 & -188.5 \\ 554.6 & 171.8 & -188.5 \\ 554.8 & 181.0 & -188.5 \\ 555.9 & 178.3 & -197.0 \\ 551.2 & 192.5 & -197.0 \\ 542.5 & 183.5 & -197.0 \\ 545.1 & 171.3 & -197.0 \\ 531.0 & 166.0 & -197.0 \\ 536.0 & 153.8 & -197.0 \\ 546.2 & 160.3 & -197.0 \\ 546.9 & 171.2 & -197.0 \end{pmatrix} \quad \ldots. \quad \begin{pmatrix} 541.9 & 160.4 & -214.3 \\ 537.5 & 170.1 & -214.3 \\ 535.3 & 160.0 & -214.3 \\ 53181 & 155.5 & -214.3 \\ 523.5 & 152.0 & -214.3 \\ 525.0 & 146.7 & -214.3 \\ 532.0 & 148.0 & -214.3 \\ 539.0 & 151.0 & -214.3 \\ 530.0 & 138.0 & -223.9 \\ 534.0 & 143.0 & -223.9 \\ 529.3 & 139.9 & -223.9 \\ 525.3 & 140.0 & -223.9 \\ 518.0 & 142.7 & -223.9 \\ 516.0 & 140.0 & -223.9 \\ 521.0 & 137.0 & -223.9 \\ 525.1 & 136.3 & -223.9 \\ 516.0 & 128.0 & -233.2 \\ 516.0 & 130.0 & -233.2 \\ 518.0 & 132.3 & -233.2 \\ 512.0 & 131.0 & -233.2 \\ 505.8 & 132.0 & -233.2 \\ 505.4 & 128.5 & -233.2 \\ 506.9 & 128.9 & -233.2 \\ 507.9 & 129.5 & -233.2 \\ 499.0 & 114.6 & -243.3 \\ 501.0 & 117.1 & -243.3 \\ 499.3 & 117.3 & -243.3 \\ 497.5 & 116.6 & -243.3 \\ 494.8 & 119.0 & -243.3 \\ 493.7 & 115.1 & -243.3 \\ 496.4 & 113.8 & -243.3 \\ 498.5 & 113.9 & -243.3 \\ 486.7 & 109.4 & -256.9 \\ 488.7 & 111.2 & -256.9 \\ 486.5 & 112.1 & -256.9 \\ 484.6 & 112.6 & -256.9 \\ 482.6 & 111.3 & -256.9 \\ 482.3 & 109.0 & -256.9 \\ 483.0 & 107.0 & -256.9 \\ 485.3 & 108.5 & -256.9 \\ 476.1 & 107.9 & -261.9 \\ 478.5 & 108.9 & -261.9 \\ 476.2 & 110.0 & -261.9 \\ 474.2 & 110.6 & -261.9 \\ 473.1 & 111.5 & -261.9 \\ 472.2 & 109.5 & -261.9 \\ 472.6 & 107.5 & -261.9 \\ 474.6 & 107.5 & -261.9 \\ 470.1 & 107.8 & -264.0 \\ 472.8 & 108.3 & -264.0 \\ 470.4 & 109.6 & -264.0 \\ 469.1 & 110.3 & -264.0 \\ 467.9 & 112.5 & -264.0 \\ 466.7 & 110.8 & -264.0 \\ 466.8 & 108.8 & -264.0 \\ 468.5 & 107.9 & -264.0 \end{pmatrix} mm$$

The knot vector for the NURBS centerline of the abductor pollicis longus is

$u = \{0, 0, 0, 0, 2.1, 7.7, 14.3, 19.1, 23.2, 27.3, 32.9, 38.2, 43.5, 47.8, 52.2, 56.4, 56.4, 56.4, 56.4\}$

with weighting values of $w = \{1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1\}$. This knot vector begins at the top of the muscle shown in Figure 18. The abductor pollicis longus is primarily a parallel fibered muscle in the forearm which runs from the back of the forearm and wraps around the brachioradialis muscle and the radius bone near the wrist to control abduction and extension of the thumb. Therefore a parallel fiber representation is applied until the knot $u_{10} = 32.9$, where the fiber parameterization switches to a helical representation with a helical intensity and direction of $\alpha = -0.2$. Then the material properties μ and κ that vary axially are applied to each cross-section. For these heterogeneous properties of the constitutive equations, the longitudinal assignments are shown in Table 2.

Table 2. Cross-sectional property distribution for abductor pollicis longus tensor-product model

Longitudinal Property	$\mu\ (kN/m^2)$	$\kappa\ (MN/m^2)$
a_0	50.0	100.
a_1	49.0	98.2
a_2	47.3	95.0
a_3	12.7	32.2
a_4	0.500	10.0
a_5	0.500	10.0
a_6	0.500	10.0
a_7	12.4	31.6
a_8	36.6	75.6
a_9	50.0	100.
a_{10}	50.0	100.
a_{11}	50.0	100.
a_{12}	50.0	100.
a_{13}	50.0	100.
a_{14}	50.0	100.

Graphically this distribution can be seen in Figure 18 with the tendon properties modeled in white and muscle properties modeled in red. The transition periods for the muscle-tendon complex are again shown by the color blending scheme. The key here is the utilization of the smooth basis functions and the fact that cross-sectional control point values are utilized by neighboring elements in the longitudinal direction.

Similar data are stored for all 28 muscles within the male human arm. A complete listing of all the muscles from this research with muscle location data, fiber type, and geometric model is given in Table 3.

These muscles are completely characterized with the previously mentioned criteria for an analysis-ready model. Therefore individual muscles and/or specific muscle groups can be pulled at will to be analyzed for any type of isometric or isotonic simulation. With a completely characterized digital muscle library, there is no longer a need to determine any of these characteristics and the focus can then be shifted to the analysis itself, striving for the end goal of a realistic stress analysis for muscles working in collaboration with a minimized computational burden.

Table 3. Summary of muscles present within the previously pictured muscle model

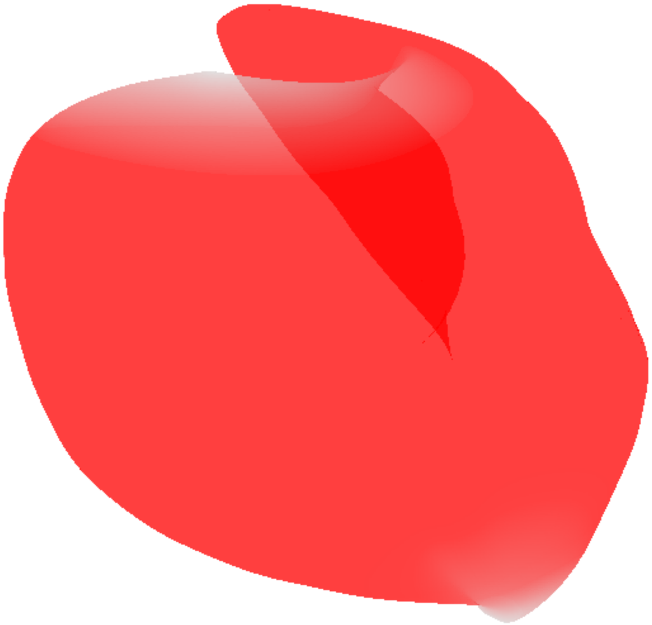
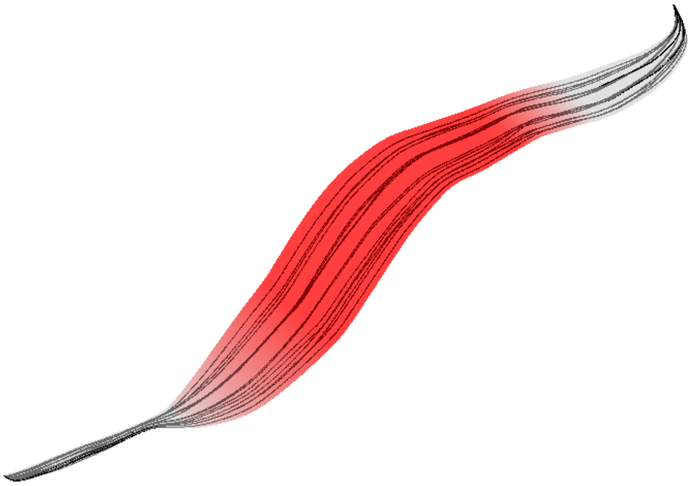
Muscle Description with Origination (O), Insertion (I), Primary Actions (A), and Fiber Type (F)	**Model with Fiber Parameterization and Material Property Distribution**
Deltoid **O:** anterior border and upper surface of the lateral 3rd of the clavical acromion, line of the scapula **I:** deltoid tuberosity of humerus **A:** shoulder abduction, flexion, and extension **F:** pennate (no fiber direction shown due to excess concavity)	
Bicep Short Head **O:** oricoracoid process of the scapula **I:** radial tuberosity and medial part of forearm **A:** supination of the forearm **F:** parallel, fusiform (modeled as separate muscle bellies)	

Table 3. Continued

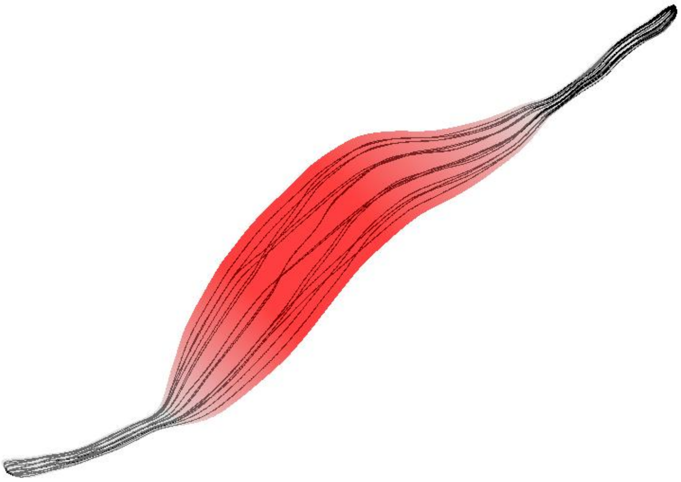
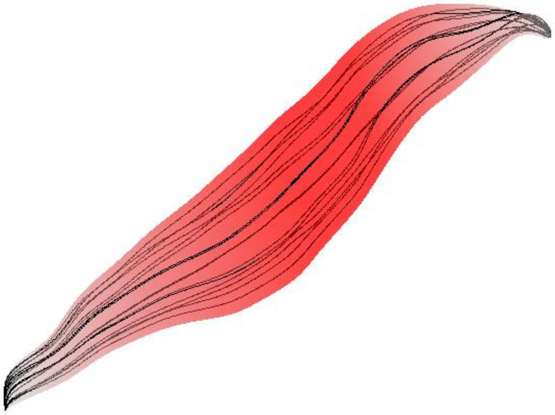
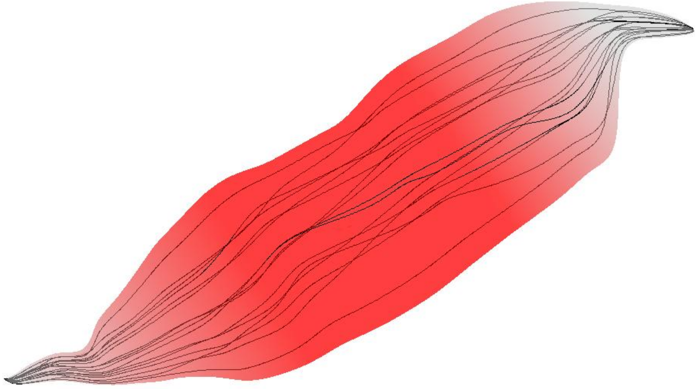
Bicep Long Head **O:** supraglenoid tubercle **I:** radial tuberosity and medial part of forearm **A:** flexion of the forearm **F:** parallel, fusiform (modeled as separate muscle bellies)	
Coracobrachialis **O:** coracoid process of scapula **I:** medial humerus **A:** adducts humerus and flexes arm **F:** parallel	
Brachialis **O:** anterior of distal humerus **I:** coronoid process and tuberosity of ulna **A:** flexion at elbow **F:** parallel	

Table 3. Continued

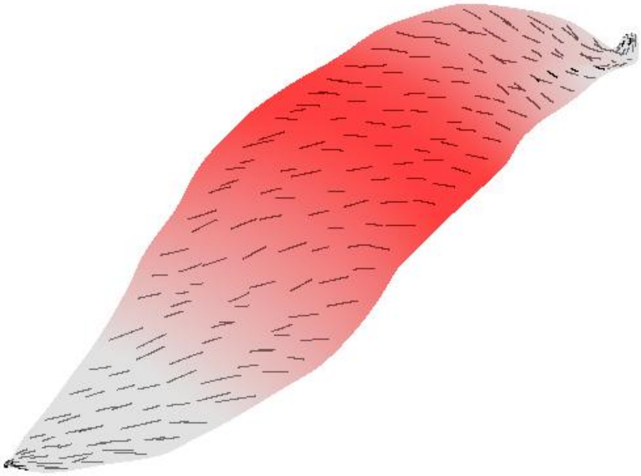
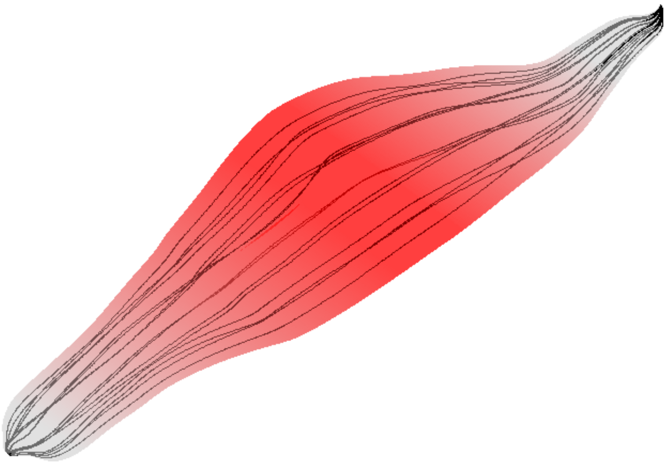
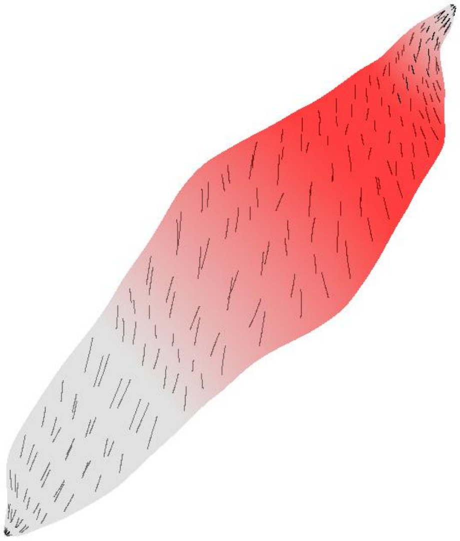
Tricep Long Head **O:** infraglenoid tubercle of scapula **I:** olecranon process of ulna **A:** adduction of shoulder **F:** pennate	
Tricep Medial Head **O:** posterior humerus **I:** olecranon process of ulna **A:** extension of forearm **F:** parallel	
Tricep Short Head **O:** posterior humerus **I:** olecranon process of ulna **A:** extension of forearm **F:** pennate	

Table 3. Continued

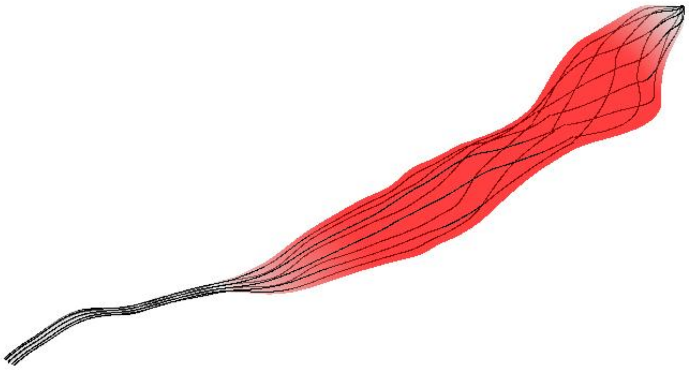
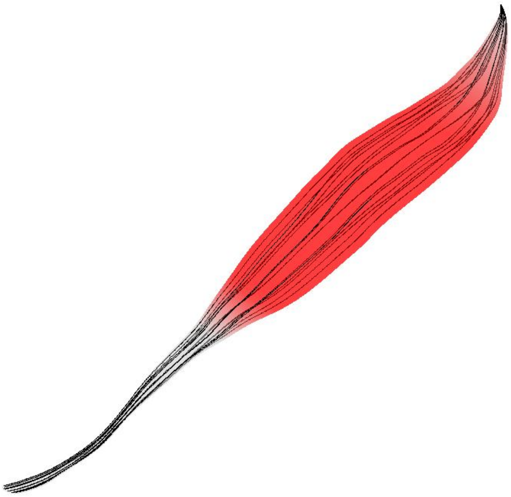
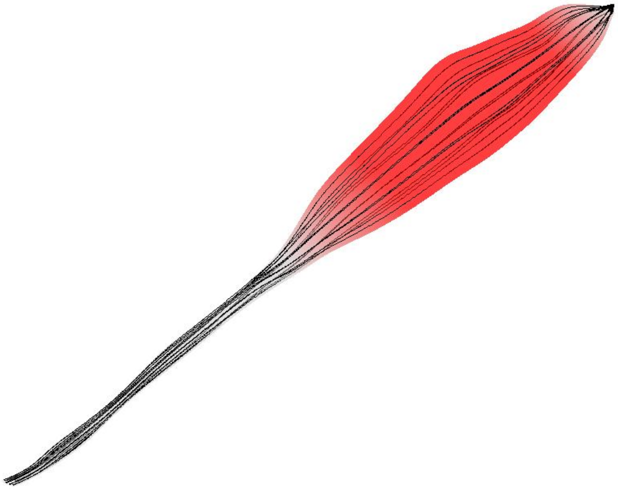
Brachioradialis **O:** lateral supracondylar ridge of humerus **I:** distal radius (radial styloid process) **A:** flexion of forearm **F:** parallel, helical	
Extensor Carpi Radialis Longus **O:** lateral supracondylar ridge of humerus **I:** 2nd metacarpal **A:** extension at wrist and adduction of the hand **F:** parallel	
Extensor Carpi Radialis Brevis **O:** humerus at anterior of lateral epicondyle **I:** base of the 3rd metacarpal **A:** extension and abduction of the hand at wrist joint **F:** parallel	

Table 3. Continued

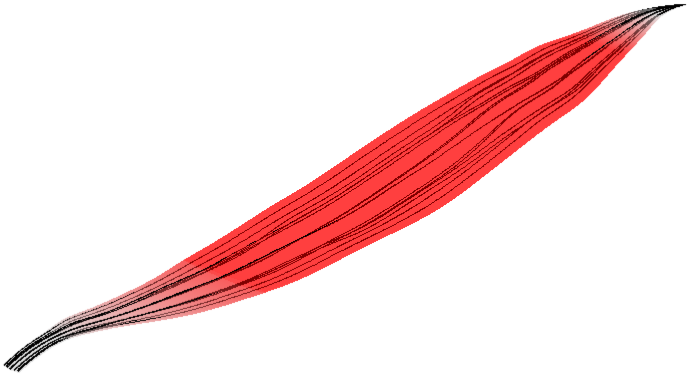
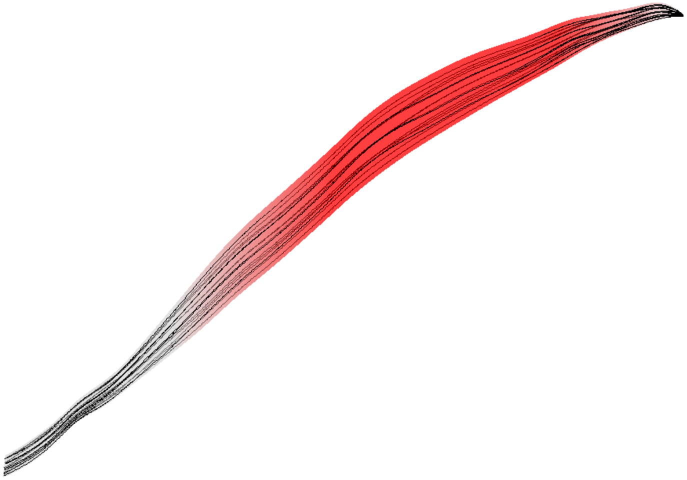
Extensor Digitorum **O:** lateral epicondyle **I:** extensor expansion of middle and distal phalanges of the 2nd, 3rd, 4th, and 5th fingers **A:** extension of hand, wrist, and fingers **F:** parallel	
Extensor Digiti Minimi **O:** anterior of the lateral epicondyle of humerous **I:** the extensor expansion at the base of the proximal phalanx on dorsal side **A:** extends little finger at all joints **F:** parallel	
Extensor Carpi Ulnaris **O:** lateral epicondyle on the ulna **I:** 5th metacarpal **A:** extension/adduction of carpus/wrist **F:** parallel	

Table 3. Continued

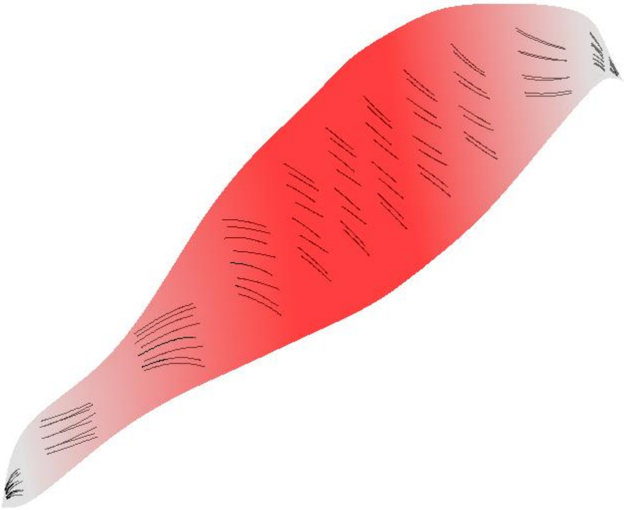
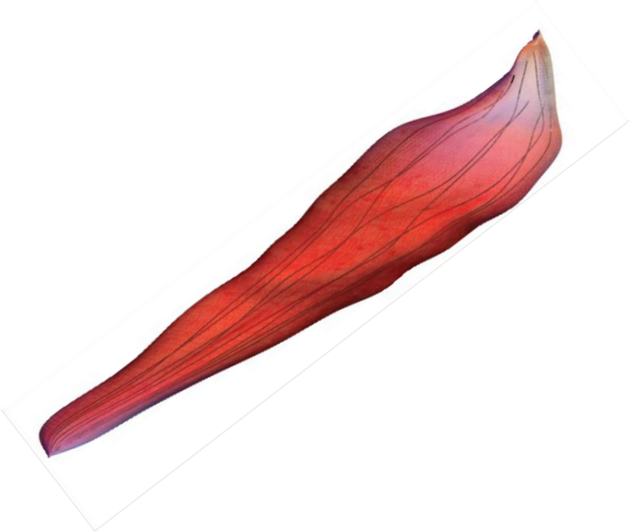
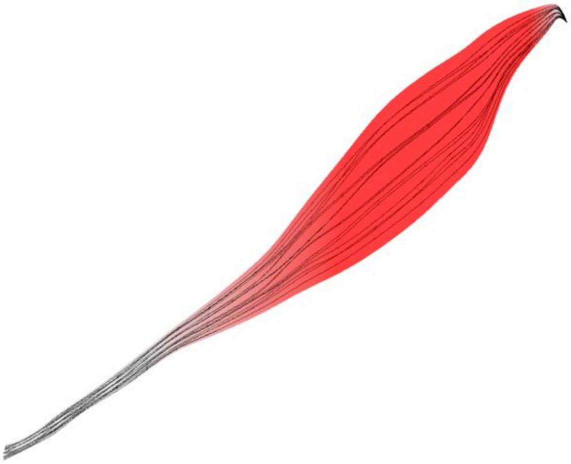
Anconeus **O:** lateral epicondyle of the proximal humerous **I:** lateral surface of the olecranon process and the posterior/distal ulna **A:** extension of and stability for pronation/supination of forearm **F:** pennate	
Pronator Teres **O:** medial epicondyle of humerus and coronoid process of ulna **I:** middle of lateral body of radius **A:** pronation of the forearm and flexion at elbow **F:** convergent	
Flexor Carpi Radialis **O:** medial epicondyle of humerous **I:** bases of 2nd and 3rd metacarpals **A:** flexion and abduction at wrist **F:** parallel	

Table 3. Continued

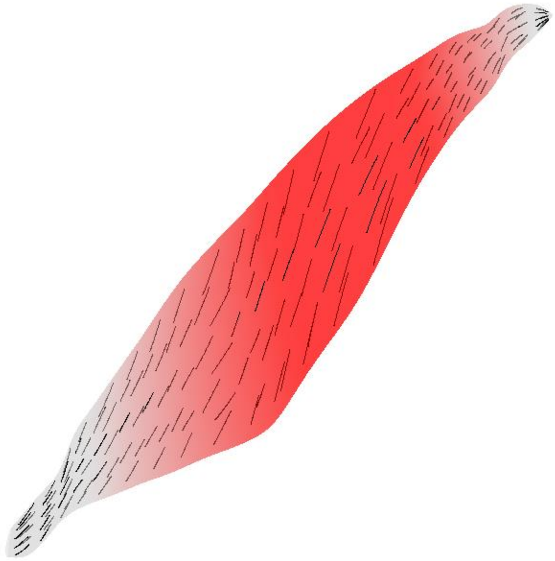
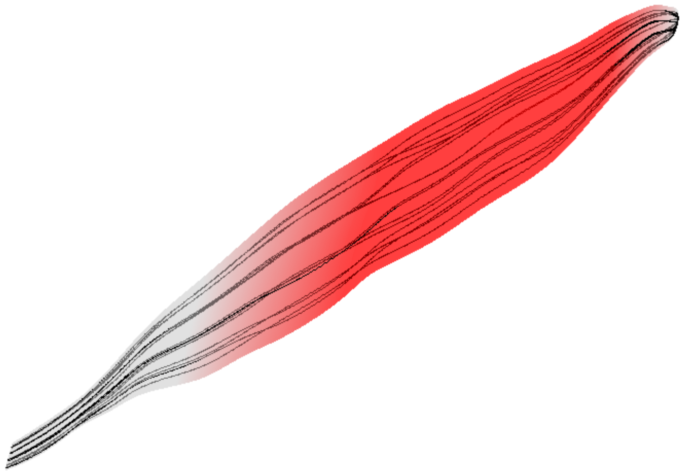
Palmaris Longus **O:** medial epicondyle of humerus **I:** palmar aponeurosis **A:** flexion of wrist **F:** parallel	
Flexor Digitorum Superficialis **O:** medial epicondyle of the humerus as well as radius and ulna **I:** anterior bases of the middle phalanges of four fingers **A:** flexion of fingers primarily at the proximal interphalangeal joints **F:** pennate	
Flexor Digitorum Profundus **O:** upper ¾ of volar and medial surfaces of the ulna and interosseous membrane **I:** base of the distal phalanges of the fingers **A:** flexion of hand at the interphalangeal joints **F:** parallel	

Table 3. Continued

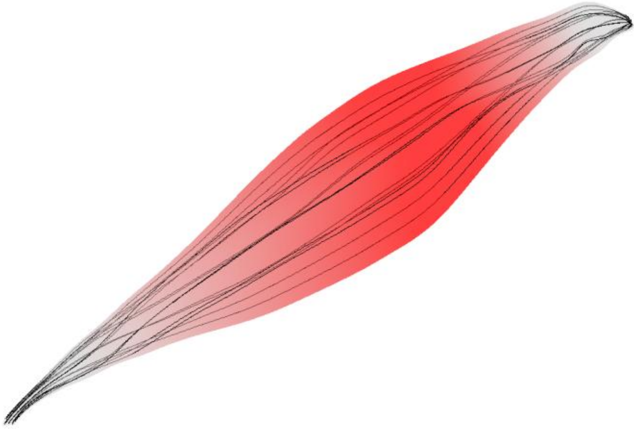
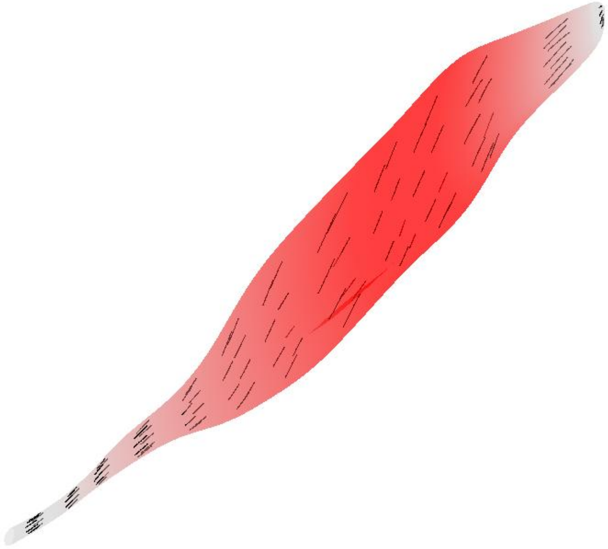
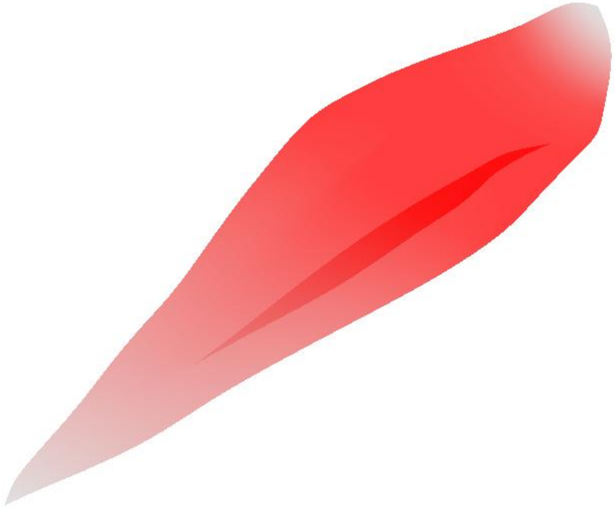
Flexor Carpi Ulnaris **O:** medial epicondyle **I:** pisiform **A:** flexion and adduction of wrist **F:** parallel	
Flexor Pollicis Longus **O:** middle volar surface of radius and interosseous membrane **I:** base of the distal phalanx of the thumb **A:** flexion of the thumb **F:** pennate	
Supinator **O:** lateral epicondyle of humerus, supinator crest of ulna, radial collateral ligament, annular ligament **I:** lateral proximal radial shaft **A:** supination of forearm **F:** pennate (no fiber direction shown due to excess concavity)	

Table 3. Continued

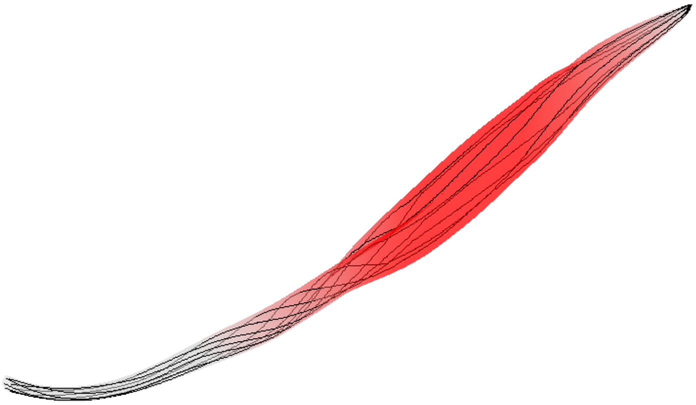
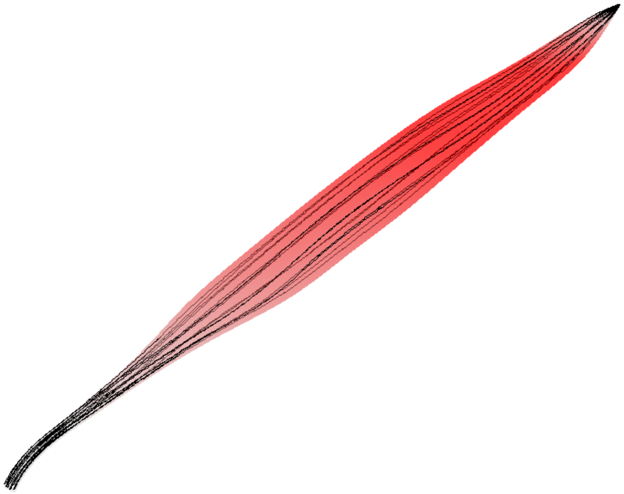
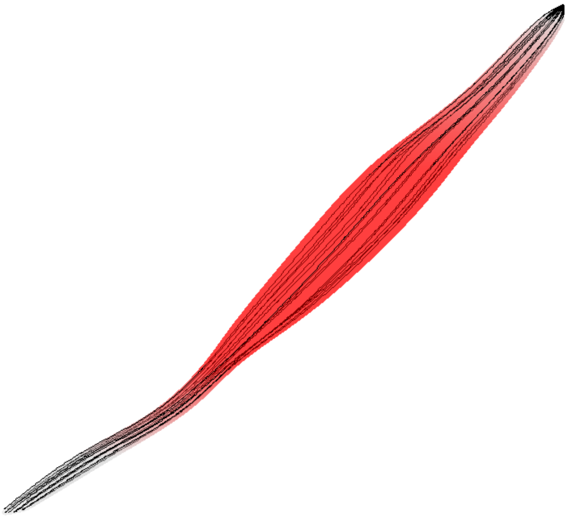
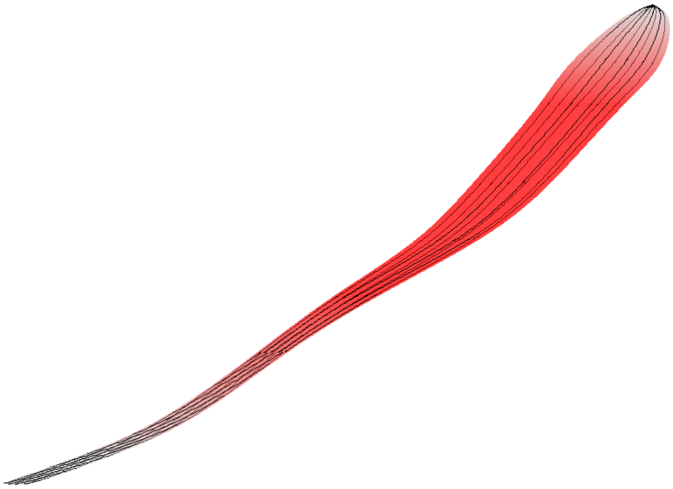
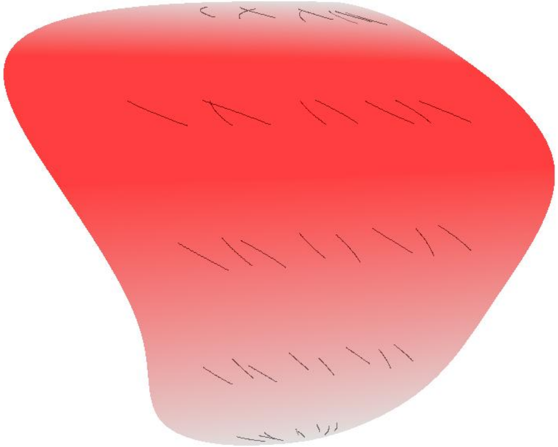
Abductor Pollicis Longus **O:** ulna, radius, interosseous membrane **I:** first metacarpal **A:** abduction and extension of thumb **F:** parallel, helical	
Extensor Pollicis Longus **O:** middle of posterior surface of the ulna **I:** thumb, distal phalanx **A:** extension of thumb **F:** parallel	
Extensor Indicis **O:** ulna, interosseous membrane **I:** index finger **A:** extension of the index finger and wrist **F:** parallel	

Table 3. Continued

Extensor Pollicis Brevis **O:** radius and the interosseous membrane **I:** thumb, proximal phalanx **A:** extension of thumb at metacapophangeal joint **F:** parallel, helical	
Pronator Quadratus **O:** medial, anterior surface of the ulna **I:** lateral, anterior surface of radius **A:** pronation of forearm **F:** pennate	

Source: Origination points, insertion points, and muscle actions taken from [27].

A Taste of Analysis

Kinematics

Analysis of an isogeometric model using NURBS has many similarities to FEA with a few key differences. Specifically, the nodal locations in FEA lie on the geometric boundaries where nodes in NURBS the control points reside off the actual geometric boundary. Therefore displacement of a NURBS model is applied by manipulating the control points instead. This control point displacement is then transferred through the basis functions to depict the actual displacement of the model. As in FEA, displacement is determined by the difference between the reference and current positions of the nodal values, or control points. In the case of the NURBS rod representation, the displacement field is given by the equation

$$\boldsymbol{u} = \sum_{i=1}^{n} \sum_{j=1}^{m} T_i(\xi^1, \xi^2) R_j(u) \boldsymbol{d}_{ij}, \tag{45}$$

where d_{ij} is the control point displacement,

$$\boldsymbol{d}_{ij} = \boldsymbol{P}_{ij} - \boldsymbol{P}_{ij}^{0}. \tag{46}$$

The deformation gradient of the form

$$\boldsymbol{F} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}} = \frac{\partial(\boldsymbol{u} + \boldsymbol{X})}{\partial \boldsymbol{X}} = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}} + \boldsymbol{I} \tag{47}$$

is then determined for each Gauss point with $\boldsymbol{x}$ as the current position and $\boldsymbol{X}$ as the initial position. The identity matrix is added to the ratio of change in displacement with respect to the reference configuration. In implementation, the chain rule of the following form is utilized:

$$\boldsymbol{F} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{u}} \cdot \left(\frac{\partial \boldsymbol{X}}{\partial \boldsymbol{u}}\right)^{\boldsymbol{-1}} \tag{48}$$

With the deformation gradient known, the Cauchy-Green deformation tensor given by

$$\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F} \tag{49}$$

is also be determined for each Gauss point location. The Cauchy-Green strain tensor is the primary determinant of the overall strain map within a deformed object.

Fiber Stretch

The fiber parameterization, as previously mentioned, is extremely important for the analysis of an anisotropic solid or skeletal muscle in this case. Strength properties along the longitudinal axis of a skeletal muscle are much greater than those in the transverse direction. Thus using the fiber parameterizations previously derived, the fiber direction at each Gauss point in the undeformed configuration of the model can be determined. With undeformed fiber direction $\boldsymbol{N}$ and the Cauchy-Green deformation tensor $\boldsymbol{C}$, the fiber stretch λ is determined through the following relation,

$$\lambda = \sqrt{\boldsymbol{NCN}}. \tag{50}$$

The undeformed fiber direction is of course taken from one of the preceding fiber parameterization techniques. With those curves, for example in the parallel fibered case,

$$\boldsymbol{x}(u) = T(\xi^1, \xi^2)R'(u)\boldsymbol{P} \qquad u_1 \leq u \leq u_{m+n+1},$$

the fiber direction at a specific Gauss integration point is determined by taking the derivative of the basis functions with respect to the axial parameter u. Then normalizing the result leads to the following fiber direction formula

$$\boldsymbol{N}(u) = \frac{\frac{\partial \boldsymbol{x}(u)}{\partial u}}{\left\|\frac{\partial \boldsymbol{x}(u)}{\partial u}\right\|} = \frac{\sum \frac{\partial(T(\xi^1,\xi^2)R(u))}{\partial u}\boldsymbol{P}}{\left\|\sum \frac{\partial(T(\xi^1,\xi^2)R(u))}{\partial u}\boldsymbol{P}\right\|}. \tag{51}$$

The fiber stretch values determined can be thought of as resembling the actual muscle contractions necessary to attain a particular model shape, i.e. the deformed state. By logging the specific fiber stretches at each Gauss point, a contour plot can be developed to produce a visual depiction of fiber contraction and fiber lengthening for a deformed body.

Stress Functions

Utilizing the strain map, one of the many constitutive equations available can be applied to determine the overall stress results. In this example, the relations discussed earlier as developed by Oomens are applied [14]. Stress terms are derived by taking the derivatives of the energy density functions with respect to the fiber stretch which leads to

$$\sigma_{active} = \lambda \frac{\partial W_{active}(\lambda, \alpha)}{\partial \lambda} = \alpha \lambda \sigma_{max}[1 - 4(1 - \lambda)^2] \tag{52}$$

and

$$\sigma_{passive} = \lambda \frac{\partial W_{passive}(\lambda)}{\partial \lambda} = \begin{cases} m_1 \lambda^2 \left(e^{m_2(\lambda^2 - 1)} - 1\right), & \lambda > 1 \\ 0, & \lambda \leq 1 \end{cases}. \tag{53}$$

Here $\sigma_{max} = 30 \frac{N}{cm^2}$ is the assumed maximum stress that can be generated by the muscle fiber while $m_1 = 35$ and $m_2 = 0.5$ are material coefficients for the passive properties of muscle fiber [1].

The second Piola-Kirchhoff stress takes the form

$$\boldsymbol{S} = 2 \frac{\partial W}{\partial \boldsymbol{C}} = 2\left(\frac{\partial W_{matrix}}{\partial \boldsymbol{C}} + \frac{\partial W_{fiber}}{\partial \boldsymbol{C}}\right). \tag{54}$$

Here the partial of the matrix material with respect to the Cauchy-Green deformation tensor is

$$\frac{\partial W_{matrix}}{\partial \boldsymbol{C}} = \frac{\mu}{2}(\boldsymbol{I} - \boldsymbol{C}^{-1}) + \frac{\kappa}{2} \ln J \boldsymbol{C}^{-1} \tag{55}$$

and the partial of the fiber material can be separated into the two previously mentioned components of

$$\frac{\partial W_{fiber}}{\partial \boldsymbol{C}} = \frac{\partial W_{active}}{\partial \boldsymbol{C}} + \frac{\partial W_{passive}}{\partial \boldsymbol{C}} = \left(\frac{\partial W_{active}}{\partial \lambda} + \frac{\partial W_{passive}}{\partial \lambda}\right)\frac{\partial \lambda}{\partial \boldsymbol{C}}$$
$$= \frac{1}{2\lambda}\left(\frac{\partial W_{active}}{\partial \lambda} + \frac{\partial W_{passive}}{\partial \lambda}\right)\boldsymbol{N} \otimes \boldsymbol{N}. \tag{56}$$

Therefore combining the results leads to an overall second Piola-Kirchhoff stress of

$$\boldsymbol{S} = \mu(\boldsymbol{I} - \boldsymbol{C}^{-1}) + \kappa \ln J\boldsymbol{C}^{-1} + \frac{1}{\lambda}\left(\frac{\partial W_{active}}{\partial \lambda} + \frac{\partial W_{passive}}{\partial \lambda}\right)\boldsymbol{N} \otimes \boldsymbol{N} \tag{57}$$

and a Cauchy stress of

$$\boldsymbol{\sigma} = \frac{1}{J}\boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T$$
$$= \frac{1}{J}(\mu(\boldsymbol{B} - \boldsymbol{I}) + \kappa \, log \, J\boldsymbol{I}) + \frac{\lambda}{J}\left(\frac{\partial W_{active}}{\partial \lambda} + \frac{\partial W_{passive}}{\partial \lambda}\right)\boldsymbol{n} \otimes \boldsymbol{n} \tag{58}$$

where $\boldsymbol{B} = \boldsymbol{F}\boldsymbol{F}^T$ and $\boldsymbol{n} = 1/\lambda \, \boldsymbol{F}\boldsymbol{N}$ is the fiber direction in the current configuration. When simulating contractions, the stress derived here using the previously assumed constitutive relations can be calculated exactly at any point.

Example of Kinematic Analysis

An example of a snake is used to demonstrate possible kinematic analysis that can be facilitated using the muscle model descriptions. Initial preliminary analysis was conducted utilizing a king cobra snake model created using the NURBS modeling software, Rhinoceros. Incremental elliptical and circular cross-sections of various sizes were swept about a centerline, creating a straight NURBS surface model of the snake. The straight snake model serves as the reference or undeformed configuration in the present work. Using this model, a current configuration or deformed snake was created by wrapping the reference snake model around a NURBS curve of an identical length. In this process the exact shape, cross-sections sizes, and number of control points were preserved. Each model contains 61 cross-sections with 4 nodes per cross-section, leading to a total of 244 control points. It should also be noted that the termination point on either end of the snake models consists of 4 point objects, all located in the same position. Again this is necessary to create a complete-looking surface model. In the present analysis this is irrelevant but future research may need to be conducted in order to attain useful analytical results. The snake in the undeformed and deformed configurations are depicted in Figure 19 as rendered Rhinoceros objects.

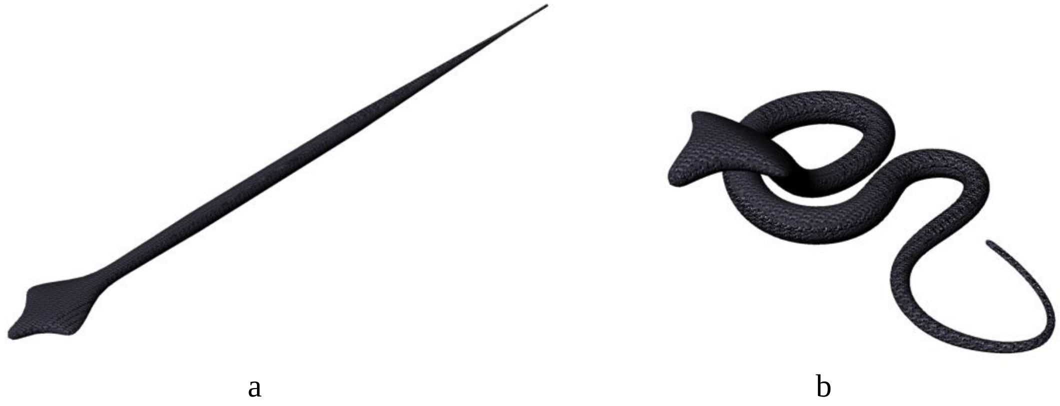


Figure 19. Model of a king cobra snake created in Rhinoceros
(a) Undeformed state
(b) Deformed state

The two surface models were exported as IGES files and imported to a Fortran IGES reader. The reader extracted information from the NURBS model and converted it into tensor-product form with control point locations, knot vectors, knot weights, and the degree of the NURBS centerline necessary to exactly reconstruct the surface model. This data was introduced into Mathematica analysis code for some preliminary testing.

The contour plot for the fiber stretches within the snake model is shown in Figure 20 on the deformed configuration. Here the red portion of the snake depicts muscle fibers contractions and blue portions represent passive fiber lengthening. Referencing the notation used earlier, $r = 1$, $s = 8$ and $t = 8$, which indicates the number of quadrature points in the radial, angular, and longitudinal direction, respectively. Therefore each meshed patch has 8 angular divisions and 8 longitudinal divisions. Each division in this case represents the fiber stretch value for the respective Gaussian quadrature point in that region using a parallel fiber orientation.

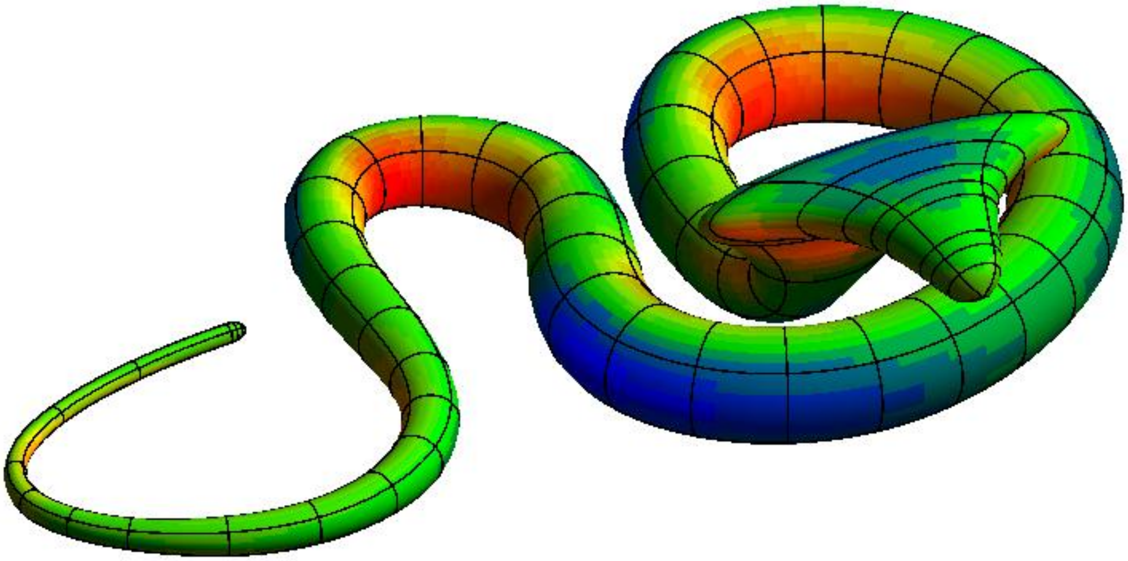


Figure 20. A contour plot produced from fiber stretches of gauss points at deformed state

The fiber stretch information, or muscle contraction drivers, determined from the deformed tensor-product rod solid opens numerous possibilities for future research. For example, a force balance can be conducted on the snake to determine the muscle contractions necessary to raise the snake's head from its original resting position. Furthermore, utilizing the information and constitutive relations discussed, one could then attempt to apply those relations to the undeformed configuration in an attempt to derive the stress results and movement depicted in the previous figures showing the deformed configuration. This inverse analysis is just one possibility that the analysis-ready model can support.

CHAPTER III

FUTURE WORK

With a fully defined analysis-ready model of the human arm, there are numerous other possibilities to expand on the current work. The models for all muscles need to be verified quantitatively. Muscle simulations for isometric and isotonic contractions can be performed to do so. To simulate isometric behavior within a muscle model, both ends of the muscle must be fixed ensuring the overall muscle length remains constant. Utilizing one of the fully defined muscle models with fiber parameterization, material property distribution, and suggested constitutive relations, the activation level could be increased to the maximum level and ensuing stress results analyzed. Comparable simulations were performed by Lu and Zhou in 2010 on a generic model of the biceps brachii [1]. This model consisted of 56 control points per bicep head. A total of 5 elements were used with the two end elements of the model strictly modeled with tendon properties and the three interior elements modeled with muscle properties. Resulting von Mises stress results for the tensor-product model and FEA models were compared and shown to be in qualitative agreement [1].

In the previously mentioned model, an isogeometric contact algorithm developed by Lu was utilized [29]. With the end goal of a full-scale 3D model of the arm working in unison, more work needs to be completed to verify the applicability of this contact algorithm to each specific muscle's case. It would need to be robust enough to handle the large amounts and varieties of muscle contact between the various muscle bodies and bones. In general, further efforts should be devoted to analysis of multiple muscles working together. Similarly, end conditions for the muscles need to be re-evaluated to avoid singularities.

As mentioned earlier one of the limitations of this model includes the utilization of only tubular shapes. Therefore further work needs to be explored to determine a suitable method for handling arbitrary geometry in which the mapping is not invertible over the entire domain. One proposed possibility is to bisect a master rod formulation into multiple tensor-product rods and patch them together to form one complete bi-concave muscular shape in which the preceding fiber parameterizations and materials property assignments could be applied. This may decrease the overall continuity along these patched locations on the cross-sectional boundary but it would allow for the preceding characterizations to be employed. Alternatively one could use the normal tri-variate NURBS solid representation to effectively model arbitrary muscle shapes with a generic B-spline mapping technique used for fiber parameterization and property distribution.

A number of simulations could be performed to test and compare the different derived concepts of the current work. One comparison could include choosing one specific model and running the same analysis for each of the different fiber parameterization schemes. By doing so, one could compare the overall impact of utilizing the correct fiber parameterization. It should be verified that a muscle is capable of producing a larger force/stress for the pennate-fibered arrangement as opposed to the parallel case, following previous research. The largest difference in stress results should be noticed for the pennate and helical results, which should have higher overall stress magnitudes than the parallel and convergent cases. Another comparison would exonerate the beneficial value of the smooth combination scheme of assigning material properties. The accuracy of this method should be shown to exceed the capabilities and accuracy of traditional FEA assignment schemes. The expectation here is that employing the linear combination scheme for the NURBS rod model should provide more accurate results near the musculo-tendon regions when directly compared to traditional approaches of property assignment on an element-by-element basis as in FEA.

With a model constructed in this manner, there is also the potential for integration with a higher level model that predicts muscle activation. Through research conducted by Patrick and Abdel-Malek in 2007, a real-time simulation of the musculoskeletal system has been developed which includes realistic constraints on joints and real-time wrapping algorithms for muscle action lines. They have coupled this model with known optimization algorithms where muscle activation levels for prescribed joint torques can be calculated in real time. Furthermore, integrating the analysis-ready model of the human arm created in the present work with a model such as this allows for the possibility to venture into the field of multi-scale modeling.

CONCLUSION

Isogeometric analysis is an extremely new field which has opened numerous possibilities for future research. Isogeometric analysis has benefits of being able to retain the exact geometry all the way through to analysis. Unlike FEA, isogeometric analysis techniques do not require a separate meshing step, which can be extremely time consuming for large models. Instead the elemental dissection and intrinsic mesh structure is inherent within the geometric representation itself. In comparison to FEA, the total number of so called “nodes” and “elements” can also be reduced with the implementation of an isogeometric method, leading to a reduced overall computational burden. The tensor-product rod is an exceptional isogeometric analysis tool for modeling and analyzing skeletal muscles due to the rod-like nature of the geometric structure. Exploiting the globally smooth nature of the NURBS centerline basis functions along elements within this method allows for universal, flexible, and more accurate fiber parameterizations and property assignments than previously capable through common FEA techniques.

Utilizing the tensor-product method, an anatomically accurate, analysis-ready isogeometric muscle model has been produced. This muscle model satisfies the three main criteria that fully characterize an analysis-ready model: geometric description, fiber parameterization, and heterogeneous property assignment. Contributions of the current work include creating an anatomically accurate geometric NURBS muscle model based on realistic images of the National Library of Medicine’s Visible Human Data Set. Also, new techniques were employed for muscle-specific fiber parameterization and material property assignment of individual muscles within this model to more accurately represent depict these characterizations. Specifically, muscle fiber parameterizations were proposed for parallel, helical, convergent, and pennate arrangements and were shown to be easily implemented. These parameterizations are universal and can be tailored to any skeletal muscle with that fiber orientation through implementation on the master rod. Parameterizing in this way eliminates the need for any completely separate fiber mapping technique or specifically ordered elemental arrangement. They also allow for adjustments to be made such as fiber direction, pennation angle, and divergence intensity. Along the same line, heterogeneous material properties of the muscle/tendon complex can be implemented as a smooth combination of cross-sectional property control values. In this case the elements are C^{p-1} continuous at the element boundaries leading to cross-sectional property values that are shared among elements. Assigning properties in this way enables a more accurate depiction of the muscle-tendon complex previous FEA modeling techniques. Furthermore, refinement techniques are easily implemented, do not alter the geometry and allow for local refinement of the muscle-tendon region. Fiber parameterization and property assignment are again made possible primarily because of the globally smooth nature of the basis functions corresponding to the NURBS centerline of the tensor-product rod.

A total of 28 muscles within the male human arm have been modeled. Each individual muscle is completely characterized with all muscle-specific criteria and stored within a “digital muscle library.” Muscles from this library can be readily utilized for various types of analysis. Examples of potential analysis applications and overall usefulness of a library of this nature were shown through the fiber stretch of a NURBS snake model and previous research results depicting analysis of the human bicep [1]. The work conducted here constitutes as the first steps toward the end goal of creating an anatomically accurate, analysis-capable, multi-scale model of the human arm with optimized integration of muscle activation.

The current formulation for the tensor-product rod is limited to convex shapes. Further research needs to be conducted in order to determine a suitable method for handling arbitrary, convex-shaped muscles. The integration of contact formulations and suitable attachment locations or end-point conditions within the current model are additional challenges that will require further research before future in vivo muscle simulations and applications can be conducted.

APPENDIX

NURBS CURVE BASICS

NURBS geometry is better understood by exploring a few examples. As it is the simplest to understand, the 2D case will be explored with a single curve. First, an arbitrary NURBS curve is shown in Figure A21. The corresponding knot vector of this curve is $U = \{0, 0, 0, 1, 2, 3, 3, 3\}$, which is categorized as non-periodic and uniform. It is also of degree p = 2, therefore the number of control points needed is $l = m - p - 1 = 8 - 2 - 1 = 5$. The weight vector for these control points is given by $w = \{1, 1, \mathrm{h}_3, 1, 1\}$. Therefore the effect of altering weighting values for the control point P_2 by setting $h_3 = 0.25, 1, 5$ can be realized. The dashed line connecting each control point represents the boundary for the convex hull. The corresponding basis functions for this curve are depicted in Figure A22. There are five basis functions corresponding to the five control points shaping the given curve. Note that these are the basis functions for when all weighting terms are equal to one, meaning the basis functions are identical to the non-rational basis functions. If the weights were to change as in Figure A21, the basis functions would also be affected.

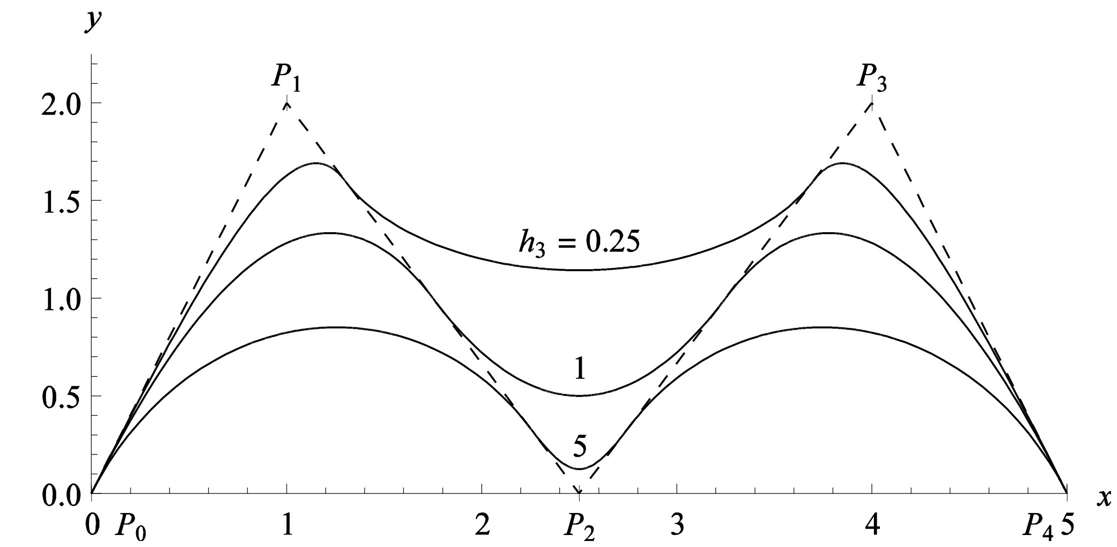


Figure A21. 2D NURBS curve showing the effect of varying weights

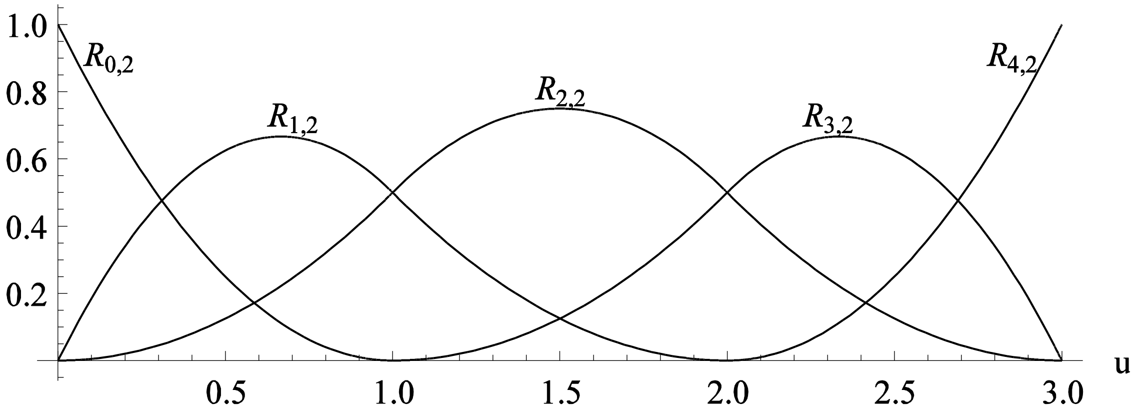


Figure A22. 2D NURBS curve basis functions

An example of de Boor’s triangle for this particular case is shown in Figure A23.

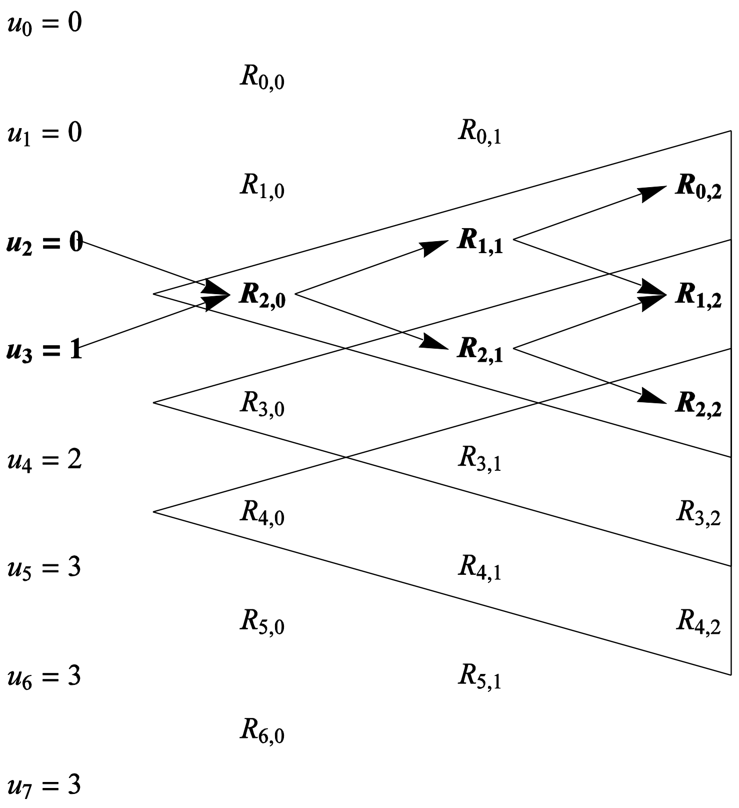


Figure A23. Example of de Boor's recursive triangle for

In this figure, the 8 knots are shown on the left hand side with their respective values. The second, third, and fourth columns represent the degree $p = 0$, $p = 1$, and $p = 2$ basis functions. Considering the knot span from 0 to 1, it is shown that the basis functions $R_{0,2}$, $R_{1,2}$, and $R_{2,2}$ affect the shape of the resulting curve in that span for $p = 2$. Likewise it follows that the knot span 1 to 2 is dependent on basis functions $R_{1,2}$, $R_{2,2}$, and $R_{3,2}$. Thus a perturbation of the last control point would not affect either of these segments of the curve. For this curve, there are a total of 3 non-empty knot spans. Each of these spans is shown in Figure A23 with their respective triangle and non-zero basis functions.

In the previous example, the curve was altered by adjusting the weighting value associated with $\boldsymbol{P}_2$. Of course the curve can also be modified by displacing individual control points as shown in Figure A24. Notice the right side of the curve remains identical to the curve before displacing $\boldsymbol{P}_1$. The reason being that this is curve is composed of three parabolic segments joined with C^1 continuity where the curve intersects the dashed line. The three parabolic segments are determined by control points $\{\boldsymbol{P}_0, \boldsymbol{P}_1, \boldsymbol{P}_2\}$, $\{\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3\}$, and $\{\boldsymbol{P}_2, \boldsymbol{P}_3, \boldsymbol{P}_4\}$, respectively. Therefore the last parabolic segment remains unaffected in this case. This exhibits one of the major benefits of using NURBS because it allows for local change.

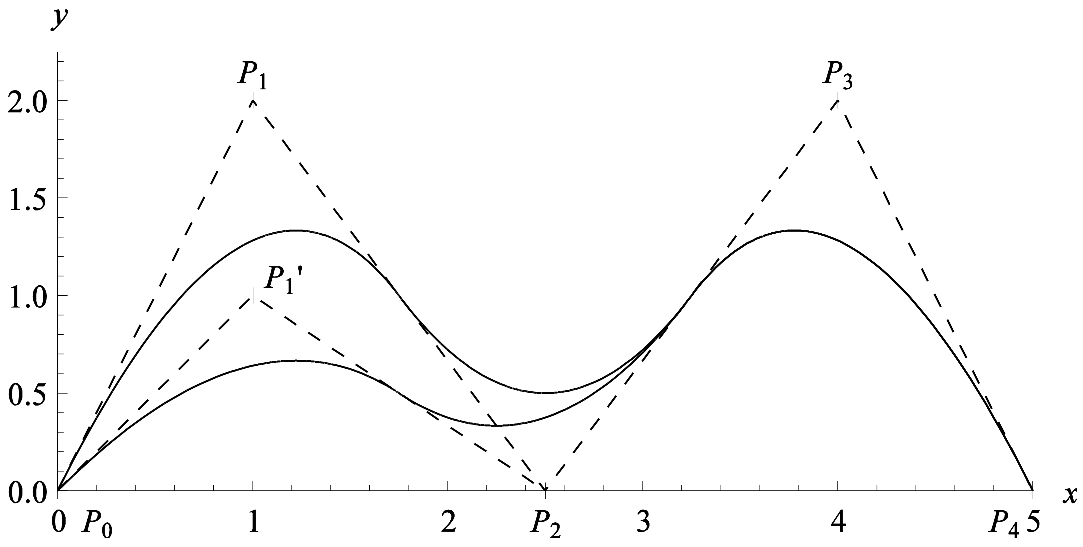


Figure A24. Displacement of control point $\boldsymbol{P}_1$

Representations of curves with degrees p = 3 and p = 4 can also be shown using the same exact same control point locations. Of course here the knot vectors will change due to the change of order. The degree p = 2 knot vector, $U = \{0, 0, 0, 1, 2, 3, 3, 3\}$, will be changed to $U = \{0, 0, 0, 0,1, 2, 2, 2, 2\}$ for a curve of degree p = 3 and

$U = \{0, 0, 0, 0, 0, 1, 1, 1, 1, 1\}$ for a curve of degree *p* = 4. This difference is shown in Figure A25 and the corresponding basis functions are shown in Figure A26. The curve of degree *p* = 3 is represented by 2 cubic polynomials and the curve of degree *p* = 4 is represented by a single, quartic curve. Notice that as the order increases, the curve begins to flatten out and becomes less influenced by each control point, exhibiting a smoothing effect. As can be seen, basis functions are dependent of the degree of the curve, knot vector values, and weighting values.

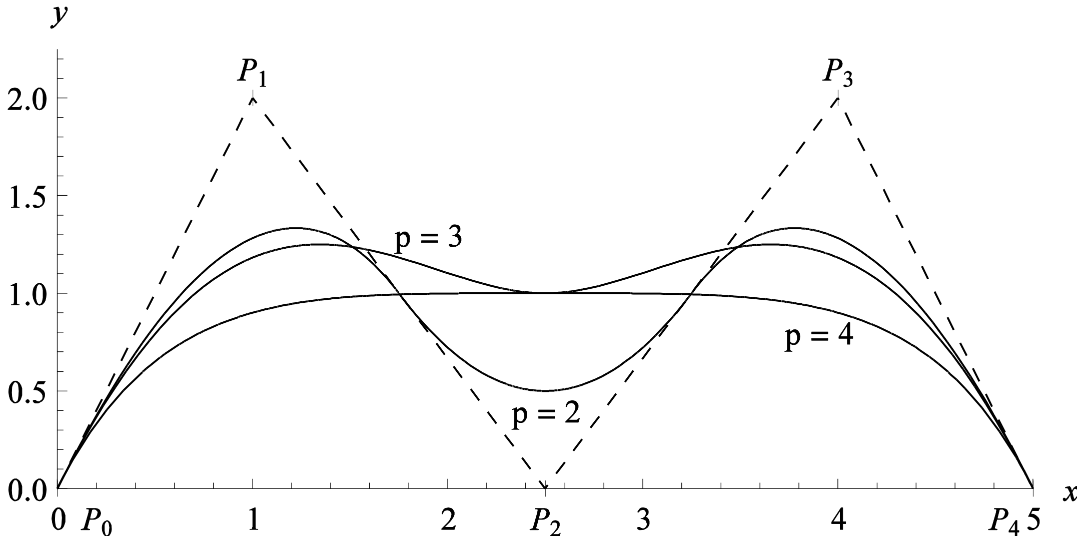


Figure A25. Effect of changing the degree of a NURBS curve

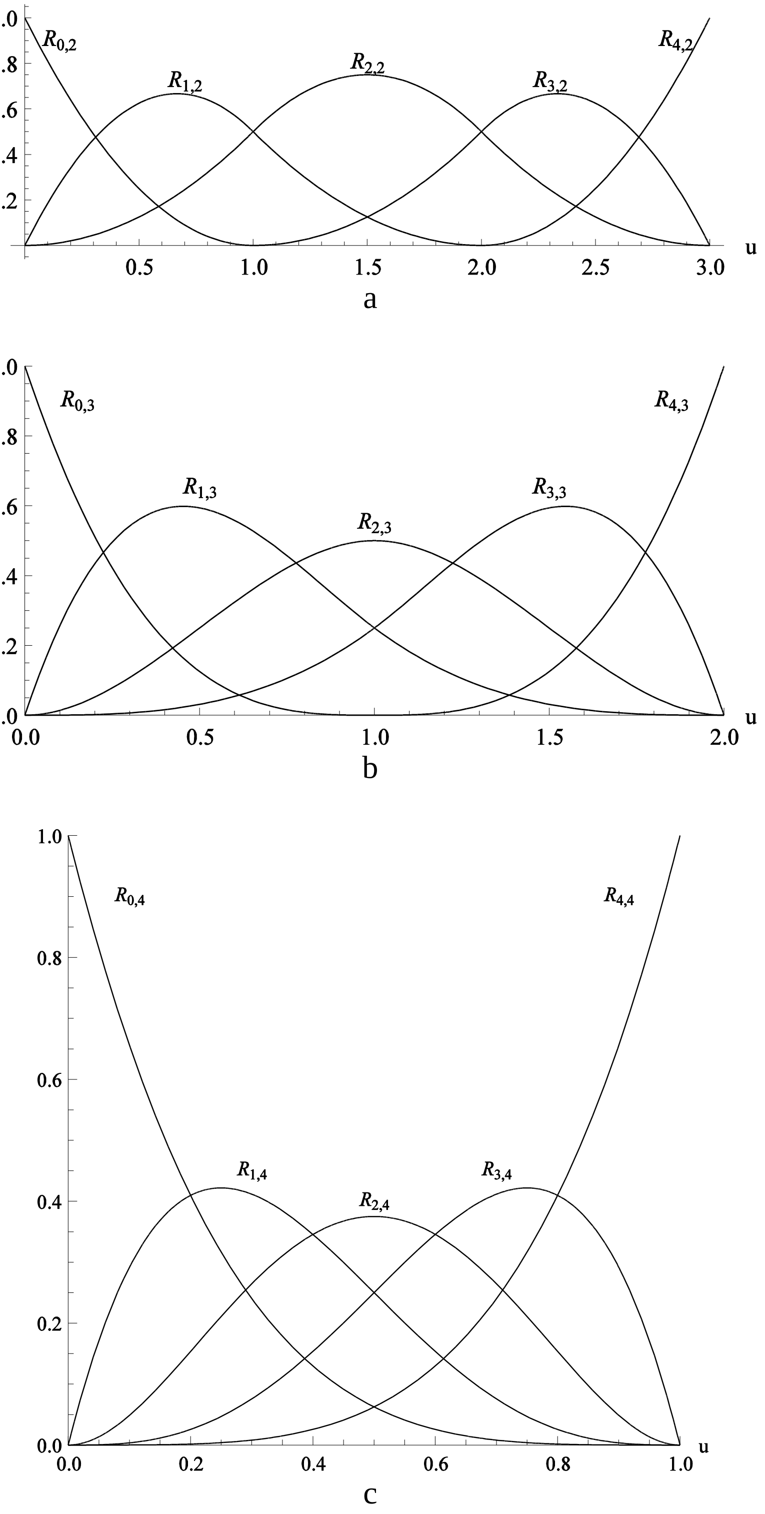


Figure A26. Effect of varying degree on basis functions
(a) $p = 2$
(b) $p = 3$
(c) $p = 4$

Adding a duplicate control point at an identical location will also modify a NURBS curve. In the following case a control point was repeated at the $\boldsymbol{P_1}$ location. Therefore the new knot vector will be $U = \{0, 0, 0, 1, 1, 2, 3, 3, 3\}$. This result is plotted against the same curve with no repeating points and is shown in Figure A27. For a degree $p = 2$ curve, repeating a control point once will make that particular segment exactly follow the convex hull or line segments connecting the control points. If it were a degree $p = 3$ curve, then three repeated control points would be needed to get the same effect. If there were only two repeating knots for a degree $p = 3$ curve, then that segment would be pulled closer to that particular control point, much like that which occurs when increasing the weighting value for a control point. In the current case, the curve is C^0 continuous at the $\boldsymbol{P_2}$ location.

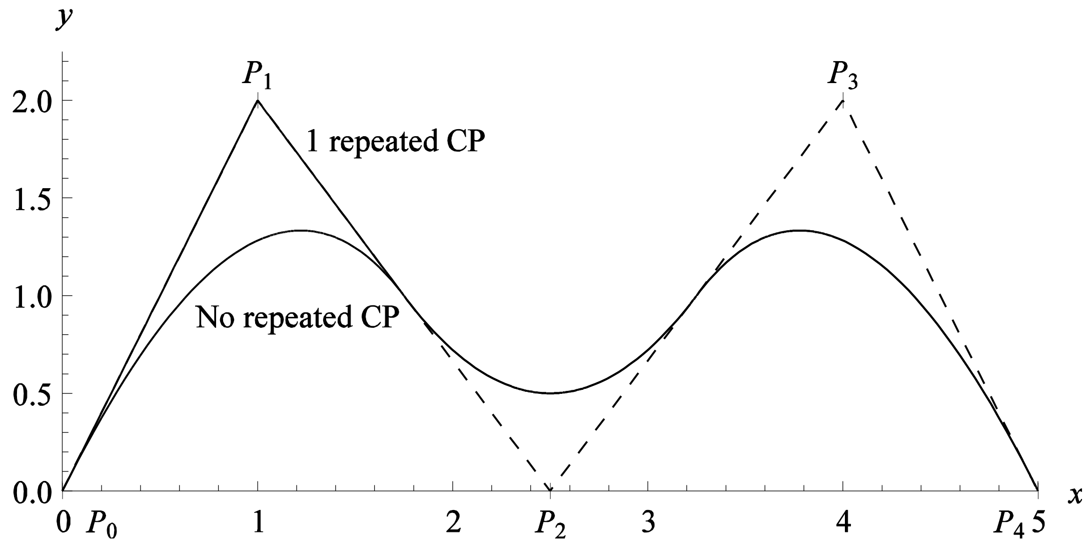


Figure A27. Effect of multiple control points in a single location on the NURBS curve

The basis functions for this particular knot vector are shown in Figure A28. Here again it is clearly shown that the curve will have a cusp at the location where the duplicate control point is placed. In general a curve is C^{p-1} continuous everywhere and each repeated knot value drops the continuity by one.

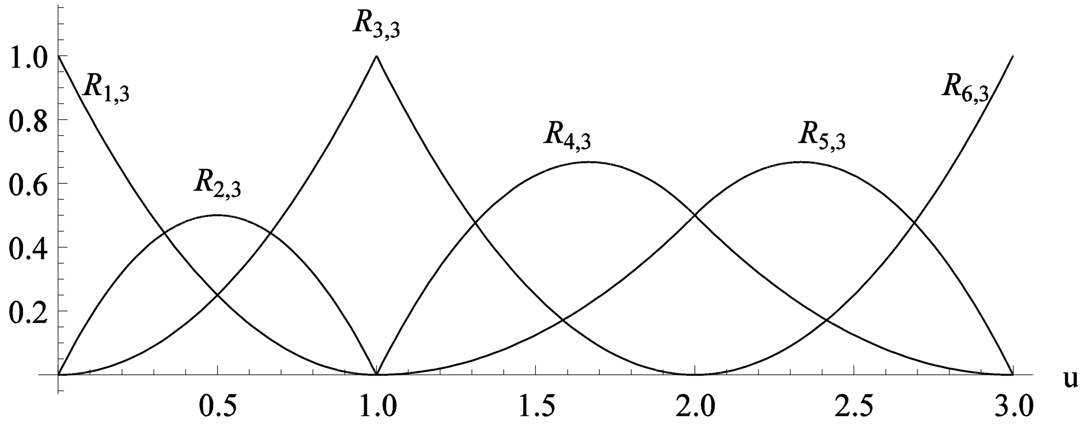


Figure A28. Basis functions for the knot vector $U = \{0, 0, 0, 1, 2, 3, 3, 3\}$

All examples up until now have used uniform knot vectors, meaning that knot values have been incremented uniformly. If these knot values are non-uniform, the respective curve and basis functions will also appear in a non-uniform nature. For example, altering the previous knot vector of $U = \{0, 0, 0, 1, 2, 3, 3, 3\}$ to $U = \{0, 0, 0, 0.25, 1.75, 3, 3, 3\}$ gives the curve and basis functions shown in Figure A29.

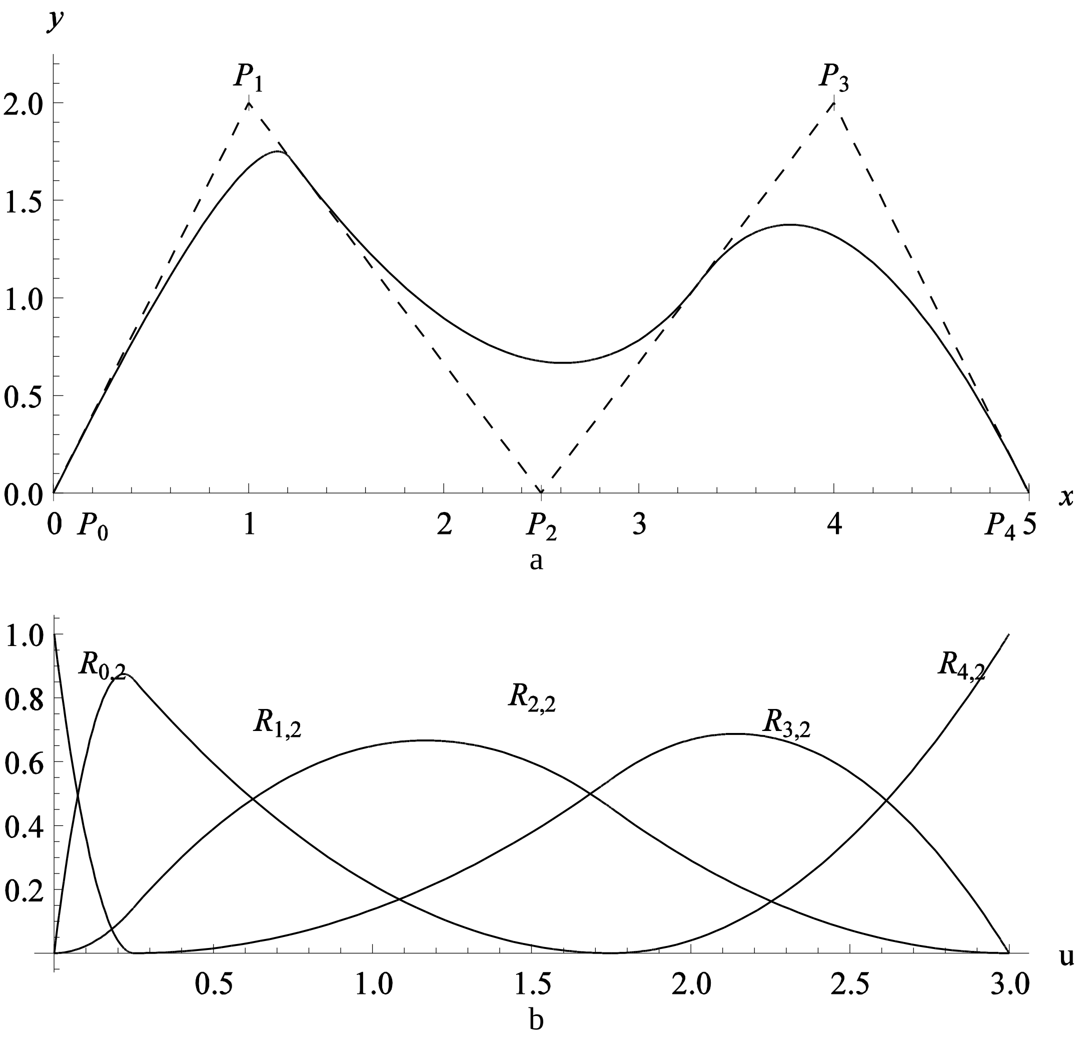


Figure A29. Non-uniform knot vector NURBS example
(a) Curve
(b) Basis functions

All previously shown curves have also been non-periodic, meaning that the beginning and ending elements of a knot vector have exhibited a multiplicity equal to the degree of the curve plus one. A knot vector for a periodic NURBS curve does not exhibit this characteristic. For example, the curve shown in Figure A30 shows the same curve in the previous examples but with a periodic knot vector of the form

$U = \{0, 1, 2, 3, 4, 5, 6, 7\}$. Notice that because of this change, the curve does not fully extend and initiate/terminate at the initial/final control point locations.

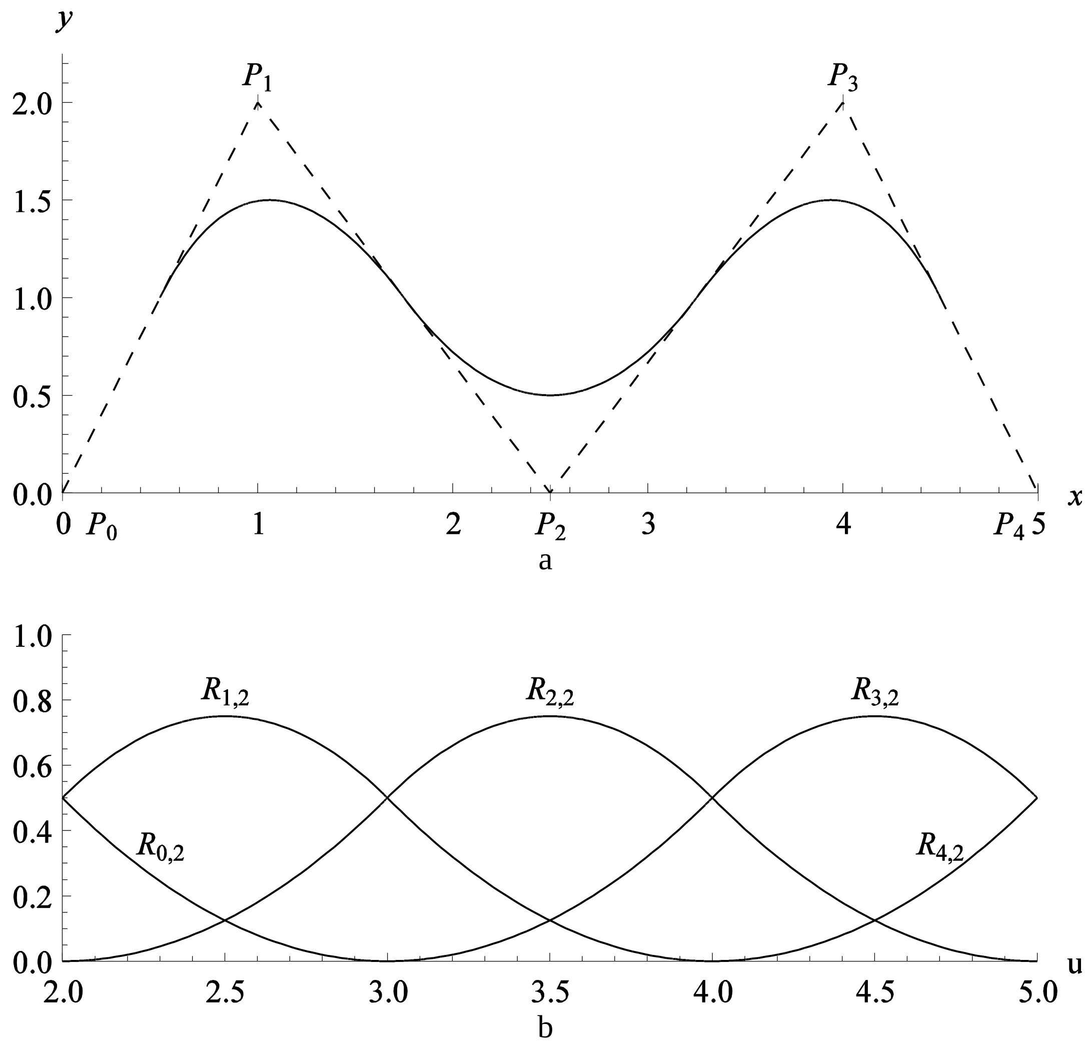


Figure A30. Periodic knot vector NURBS example
(a) Curve
(b) Basis functions

REFERENCES

1. Lu, Jia and Zhou, Xianlian. Cylindrical element: Isogeometric model of continuum rod. *Computational Methods in Applied Mechanics and Engineering.* January 2011, Vol. 200, 1-4, pp. 233-241.

2. Chopine, Ami. *3D Art Essentials: The Fundamentals of 3D Modeling, Texturing, and Animation.* Oxford : Focal Press, 2011.

3. Zygote Media Groupe, Inc. 3D Male Muscular System Model. *3Dscience.com.* [Online] 2011. [Cited: August 13, 2011.] http://www.3dscience.com/3D_Models/Human_Anatomy/Male_Systems/Male_Muscles.php.

4. Plasticboy Pictures, CC. Human Male and Female Anatomy Complete 3D Pack V04 (Fully Textured). *plasticboy.co.uk.* [Online] 2009. [Cited: August 13, 2011.] http://www.plasticboy.co.uk/store/Human_Male_Female_Anatomy_Complete_V04.html?gclid=CPLA387ej60CFWkDQAoduDOplw.

5. Evans, Christopher. Simulating Musculature in Maya. *chrisevans3d.com.* [Online] 2003. [Cited: August 16, 2011.] http://chrisevans3d.com/tutorials/maya_muscle.htm.

6. Shipkov, Peter. Muscles 4.1. *petershipkov.com.* [Online] [Cited: August 13, 2011.] http://petershipkov.com/development/muscles/muscles.htm#usage.

7. Delp, Scott P. and Loan, J. Peter. A Computational Framework for Simulation and Analysis of Human and Animal Movement. *Computing in Science and Engineering.* 2000, Vol. 2, pp. 46-55.

8. Patrick, Amos and Abdel-Malek, Karim A. A Musculoskeletal Model of the Upper Limb for Real Time Interaction. *2007 Digital Human Modeling Conference.* 2007, pp. 1209-1216.

9. Spyrou, L.A. and Aravas, N. Muscle-driven finite element simulation of human foot movements. *Computer Methods in Biomechanics and Biomedical Engineering.* June 29, 2011.

10. Jones, D., Round, J. and de Hann, A. *Skeletal Muscles from Molecules to Movement.* London : Churchill Livingstone, 2004. 9780443074271.

11. Lemos, Robson, et al. Realistic Skeletal Muscle Deformation using Finite Element Analysis. *Proceedings of the XIV Brazilian Symposium.* October 2001, pp. 192-199.

12. Tang, C.Y., Zhang, G. and Tsui, C.P. A 3D skeletal muscle model coupled with active contraction of muscle fibers and hyperelastic behavior. *Computer Methods in Biomechanics and Biomedical Engineering.* May 2009, Vol. 42, 7, pp. 865-872.

13. Blemker, Silvia S., Pinsky, Peter M. and Delp, Scott L. A 3D model of muscle reveals causes of nonuniform strains in the biceps brachii. *Jounal of Biomechanics.* April 2005, Vol. 38, 4, pp. 657-665.

14. Oomens, C.W.J., et al. Finite element modelling of contracting skeletal muscle. *Philosohpical Transactions of the Royal Society of London.* Sept. 29, 2003, pp. 1453-60.

15. Hughes, T.J.R., Cottrell, J.A. and Brazilevs, Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry, and mesh refinement. *Computer Methods in Applied Mechanics and Engineering.* October 2005, Vol. 194, 39-41, pp. 4135-4195.

16. Cottrell, J. Austin, Hughes, Thomas J.R. and Bazilevs, Yuri. *Isogemetric Analysis: Toward Integration of CAD and FEA.* Chichester : John Wiley & Sons, Ltd., 2009. 9780470749081.

17. Rogers, David F. *An Introduction to NURBS With Historical Perspective.* San Diego : Academic Press, 2001.

18. Piegl, Les and Tiller, Wayne. *The NURBS Book.* Berlin : Springer, 1995.

19. Blanc, Carole and Schlick, Christophe. Accurate Parametrization of Conics by NURBS. *Computer Graphics and Applications.* 1996, Vol. 16, 6, pp. 64-71.

20. Lu, Jia. Circular element: Isogeometric elements of a smooth boundary. *Computer Methods in Mechanics and Engineering.* June 2009, Vol. 198, 30-32, pp. 2391-2402.

21. Floater, Michael S. Mean Value Coordinates. *Computer Aided Geometric Design.* 2003, Vol. 20, pp. 19-27.

22. Derose, Tony and Meyer, Mark. Harmonic Coordinates. *Pixar Technical Memo #06-02.* 2006.

23. Marieb, E. *Human Anatomy & Physiology.* 6th. San Francisco : Pearson Benjamin Cummings, 2004. pp. 279-319. 0805354638.

24. Martini, Frederic H, Timmons, Michael J and Tallitsch, Robert B. *Human Anatomy.* 6th. s.l. : Benjamin Cummings, 2008. pp. 251-252.

25. *Visible Human Data Set.* s.l. : National Library of Medicine, 1994.

26. *McMinn's Color Atlas of Human Anatomy.* Abrahams, Peter H. s.l. : Mosby, 2003, p. 378. ISBN 0723432120.

27. Gray, Henry. *Anatomy of the Human Body.* [ed.] Warren H. Lewis. 20th. Philadelphia : Lea & Febiger, 1918.

28. Gardiner, John C. and Weiss, Jeffrey. Simple Shear Testing of Parallel-Fibered Soft Tissues. *Journal of Biomechanics.* April 2001, Vol. 123, 2, pp. 170-175.

29. Lu, Jia. Isogeometric contact analysis: Geometric basis and formulation for frictionless contact. *Computer Methods in Applied Mechanics and Engineering.* 2011, Vol. 200, 5-8, pp. 726-741.