

**MUSCULOSKELETAL MODELING OF *SMILODON FATALIS*
FOR VIRTUAL FUNCTIONAL PERFORMANCE TESTING**

by

KIRAN S KONAKANCHI

AUGUST 19, 2005

A project submitted to the Faculty of the Graduate School of the State
University of New York at Buffalo in partial fulfillment of the requirements for
the degree of

MASTER OF SCIENCE

Department of Mechanical and Aerospace Engineering
State University of New York at Buffalo
Buffalo, New York 14260

To my parents

Abstract

There is a considerable interest among paleobiologists in studying the behavioral patterns of extinct animals. In the absence of living specimens, typically two approaches can be adopted. One approach is to study the behavior of a living animal from a closely related species under the assumption that it resembles the extinct animal. Any observations and conclusions to the extinct animal can then be extrapolated. The alternate approach, one that we will explore, is to create a Virtual Prototype of the extinct animal.

Virtual Prototyping is a concurrent engineering practice that uses computer based numerical simulations to aid the functional performance evaluation of various engineering problems. In our project we wish to adopt a similar strategy to aid the functional performance evaluation of the hunting habits of such extinct animals.

The main goal of this project is to be able to construct accurate biomechanically constrained computational virtual prototypes of living/extinct animals and then to be able to perform a range of virtual/ computational experiments under known and estimated conditions in order to study the functional performance. This is explored in a case study of Bite force analysis of Tiger and Smilodon.

In terms of implementation we anticipate the following steps. Since the fossil remaining of the extinct animals is available, we can only obtain point data (CT/MRI data) that is of little significance for engineering analysis. This data is then converted first into an intermediate geometric model. Then by progressively adding physiological constraints (such as muscles, tendons) as well as physical constraints (such as impenetrability, motion limits and actuator capacities) we obtain the constrained virtual prototype.

Creating a virtual prototype in this manner allows us to retain and preserve various geometric features that play a critical role in the analysis. Subsequently other limitations and constraints pertaining to material, strength, rigidity etc. can be applied to this Virtual Prototype allow it to mimic the actual animal in form and function. Once the final model is developed a range of virtual experiments/ analysis can now be systematically and iteratively performed on the computational virtual prototype in order to study the functional performance. A proof of concept example will be used to present and showcase the various steps in this multistage process.

Acknowledgment

I would like to acknowledge the invaluable contributions of the following persons in the successful completion of the project.

First, I would like to express my sincerest gratitude to my advisor, Dr. Venkat Krovi, for giving me an opportunity to work under him. He was not only my mentor but also as a friend he always provided me with valuable suggestions whenever I needed the most.

I would like to express my gratitude to the committee member, Dr. Frank Mendel for reading through my project and providing me with valuable suggestions. I would like to thank all the members of the 'Sabertooth Cat Research Group', Dr. Abani Patra, Dr. Kevin Hulme, Scott Woodward, Vamsi Patel, Michael Del Signore, Talib, and Anu. This group has always provided me with valuable information and lively discussions though we have to wake up early in the morning for the meetings, especially in winter.

I would like to specially thank my lab members Rajan Bhatt, Chin Pei Tang for always lending me a helping hand in clarifying my doubts. Not to forget Glenn White, Chetan, Tao, and Mike for providing such a lively environment to work.

I would like to thank some of my friends Kranthi, Vamsi, Ravi, Raghu, Preetam, Schubert, Anand etc. for their cheerful talk and support through out my masters work.

And especially I would to thank my parents, Prasad Konaknachi, Annapurna Konakanchi, and my family members Kishore, Shantala, Rama Krishna , Sudha for each and every thing they did for me. Their support and encouragement have always proven to be useful and made me reach until this stage.

Contents

Abstract.....	i
Acknowledgment	iii
Contents	v
List of Figures	viii
List of Tables	xiii
1 Introduction.....	1
1.1 Overview.....	1
1.2 Background.....	2
1.3 Virtual Prototyping	3
1.4 Biomechanical Modeling	5
1.5 Musculoskeletal Modeling.....	7
1.6 Project Goals.....	7
1.7 Challenges.....	10
1.7.1 Geometric Complexity.....	10
1.7.2 Individual Muscle model	11
1.7.3 Group Muscle Modeling.....	12
1.8 Project Organization	14
2 Background.....	15
2.1 Musculoskeletal Modeling Applications	15
2.1.1 Bite Force Analysis.....	16

2.2	Musculoskeletal Analysis Tools	16
2.2.1	Software for Interactive Musculoskeletal Modeling (SIMM)	17
2.2.2	SIM Mechanics	18
2.2.3	Virtual Muscle	19
2.2.4	LifeMod	20
2.2.5	Visual Nastran.....	21
2.2.6	Vertebrate Analyzer (VA).....	21
2.2.7	AnyBody	22
2.3	Applications	23
3	Musculoskeletal Modeling With AnyBody Software.....	24
3.1	What is AnyBody?.....	24
3.2	Modeling Procedure.....	25
3.3	Our Model in AnyBody	30
3.3.1	Creating a CAD Model from Point Data (CT/MRI Data)	31
3.4	Building Musculoskeletal Model in AnyBody	35
3.5	MATLAB Interface	40
4	Optimization And Bite Force Calculation	48
4.1	Background.....	48
4.1.1	Forward Dynamics.....	48
4.1.2	Inverse Dynamics.....	48
4.1.3	Redundancy Resolution	49
4.2	Problem A: Determining Muscle Forces	51
4.2.1	Our Model:.....	53
4.2.2	Case 1: Tiger – VPMuscle – Simple.....	56

4.2.3	Case 3: Tiger – SPMuscle - Simple	57
4.2.4	Case 5: Sabertooth tiger - VPMuscle - Simple	58
4.2.5	Case 7 : Sabertooth tiger - SPMuscle - Simple.....	59
4.2.6	Case 2: Tiger – VPMuscle - Complex	60
4.2.7	Case 4: Tiger – SPMuscle - Complex.....	61
4.2.8	Case 6: Sabertooth tiger – VPMuscle – Complex	62
4.2.9	Case 8: Sabertooth tiger – SPMuscle - Complex.....	63
4.3	Problem B: Bite Force Calculation.....	65
4.4	Problem C: Finding Optimal Muscle Attachment Location	67
4.4.1	Parametric Sweep Studies.....	70
4.4.2	Full Optimization:.....	77
5	Conclusion And Future Work.....	83
5.1	Conclusion	83
5.2	Future Work.....	85
	Bibliography.....	86
	Appendix A.....	90

List of Figures

Figure 1-1 Example of evolutionary change that couldn't be answered may obtain a possible solution by the application of engineering concepts.....	1
Figure 1-2 (a) Virtual Model of backhoe (b) Virtual Prototype of Sabertooth tiger (courtesy Mike).....	4
Figure 1-3 Different modeling procedures (a) Conceptual model (b) Analytical model (c) CAD model (d) Musculoskeletal model	6
Figure 1-4 (a) Coarse model of skull and mandible of Sabertooth tiger (b) Fine model with all the geometry captured.....	11
Figure 1-5 Black box model of muscle with two inputs and an output	12
Figure 1-6 (a) Individual muscle model of tiger (b) Group muscle model of tiger	13
Figure 1-7 Overall structure of the case study	14
Figure 2-1 Musculoskeletal model of human in SIMM.....	18
Figure 2-2 Musculoskeletal model of human in LifeMod.	20
Figure 2-3 Sample Visual Nastran model of cam and piston	21
Figure 2-4 (a) Shaded model of Smilodon in Vertebrate Analyzer, (b) "Vertex-based" Smilodon with muscle and corresponding muscle characteristics in VA.....	22
Figure 3-1 Simplified modeling procedure in AnyBody	25
Figure 3-2 AnyBody Software Interface.....	26
Figure 3-3 Importing DICOM files into MIMICS software	31
Figure 3-4 Scan image of tiger.....	32
Figure 3-5 (a) 3D model of tiger skull, (b) 3D model of tiger mandible.....	34

Figure 3-6 (a) Window showing the STL file selection, (b) Window showing the properties of STL file.....	34
Figure 3-7 Global reference frame.....	35
Figure 3-8 (a) Skull of tiger, (b) Skull of Sabertooth tiger	36
Figure 3-9 (a) Skull and Mandible of tiger, (b) Skull and Mandible of Sabertooth tiger .	37
Figure 3-10 (a) Musculoskeletal model of tiger, (b) Musculoskeletal model of Sabertooth tiger	39
Figure 3-11 (a) Global frame selection window, (b) Node properties in Global frame ...	40
Figure 3-12 (a) Segment properties, (b) Segment Node properties	41
Figure 3-13 Joint selection window with its corresponding properties	42
Figure 3-14 Kinematic Measure properties for measuring distance in 3D.....	43
Figure 3-15 Driver selection window and its corresponding properties.....	44
Figure 3-16 (a) Simple muscle model, (b) Hill muscle model	45
Figure 3-17 (a) Via point muscle class properties, (b) Shortest path muscle class properties.....	46
Figure 3-18 Running the analysis	47
Figure 4-1 Mass being acted by two forces showing the problem of redundancy	50
Figure 4-2 Process flow chart for calculating muscle forces	53
Figure 4-3 3-dimensional representation of case studies	55
Figure 4-4 (a) Plot 1 :Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – VPMuscle – Simple (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – VPMuscle - Simple	56

Figure 4-5 (a) Plot1: Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – SPMuscle - Simple (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – SPMuscle - Simple 57

Figure 4-6 (a) Plot 1: Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left , right hand sides) for Sabertooth tiger – VPMuscle – Simple (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Sabertooth tiger – VPMuscle - Simple..... 58

Figure 4-7 (a) Plot 1: Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Sabertooth tiger – SPMuscle – Simple (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Sabertooth tiger – SPMuscle - Simple 59

Figure 4-8 (a) Plot 1: Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – VPMuscle – Complex (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – VPMuscle - Complex 60

Figure 4-9 (a) Plot 1: Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left, right hands sides) for Tiger – SPMuscle – Complex (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – SPMuscle - Complex 61

Figure 4-10 (a)Plot 1: Muscle Activities of Temporalis, Masseter , Pterygoid muscles (left, right hand sides) for Sabertooth tiger–VPMuscle–Complex (b)Plot2: Individual muscle forces of Temporalis, Masseter, Pterygoid muscles (left, right hand sides) for Sabertooth tiger–VPMuscle-Complex 63

Figure 4-11 (b) Plot 1: Muscle Activities of Temporalis, Masseter, Pterygoid muscles (left,right hand sides) for Sabertooth tiger – VPMuscle – Complex (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Sabertooth tiger – VPMuscle - Complex	64
Figure 4-12 Process flow chart for calculating bite force.....	67
Figure 4-13 Process flow chart for creating musculoskeletal model from point data	68
Figure 4-14 Parametric sweep studies showing the variation of bite force with Temporalis origin for tiger (a) surface plot (b) Line plot	70
Figure 4-15 Parametric sweep studies showing the variation of bite force with Temporalis insertion for tiger (a) surface plot (b) Line plot	71
Figure 4-16 Parametric sweep studies showing the variation of bite force with Masseter origin for tiger (a) surface plot (b) Line plot.....	71
Figure 4-17 Parametric sweep studies showing the variation of bite force with Masseter insertion for tiger (a) surface plot (b) Line plot	72
Figure 4-18 Parametric sweep studies showing the variation of bite force with Pterygoid origin for tiger (a) surface plot (b) Line plot.....	72
Figure 4-19 Parametric sweep studies showing the variation of bite force with Pterygoid insertion for tiger (a) surface plot (b) Line plot	73
Figure 4-20 Parametric sweep studies showing the variation of bite force with Temporalis origin for sabertooth tiger (a) surface plot (b) Line plot.....	74
Figure 4-21 Parametric sweep studies showing the variation of bite force with Temporalis insertion for sabertooth tiger (a) surface plot (b) Line plot	74
Figure 4-22 Parametric sweep studies showing the variation of bite force with Masseter origin for sabertooth tiger (a) surface plot (b) Line plot.....	75

Figure 4-23 Parametric sweep studies showing the variation of bite force with Masseter insertion for sabertooth tiger (a) surface plot (b) Line plot	75
Figure 4-24 Parametric sweep studies showing the variation of bite force with Pterygoid origin for sabertooth tiger (a) surface plot (b) Line plot.....	76
Figure 4-25 Parametric sweep studies showing the variation of bite force with Pterygoid insertion for sabertooth tiger (a) surface plot (b) Line plot	76
Figure 4-26 Bite force graph for Tiger – VPMuscle - Simple.....	78
Figure 4-27 Bite force graph for Tiger – SPMuscle - Simple.....	78
Figure 4-28 Bite force graph for Sabertooth tiger – VPMuscle - Simple.....	79
Figure 4-29 Bite force graph for Tiger – SPMuscle - Simple.....	79
Figure 4-30 Bite force graph for Tiger – VPMuscle - Complex.....	80
Figure 4-31 Bite force graph for Tiger – SPMuscle - Complex	80
Figure 4-32 Bite force graph for Sabertooth tiger – VPMuscle - Complex.....	81
Figure 4-33 Bite force graph for Sabertooth tiger – SPMuscle - Complex	81
Figure A- 1 Black box model of muscle with two inputs and an output	91
Figure A- 2 Experimental set up for determining the Muscle parameters.....	93
Figure A- 3 Two variations of Hill muscle model.....	94
Figure A- 4 (a) Force-length properties of Series and Parallel elastic elements, (b) Force-length properties of Contractile Element.....	94

List of Tables

Table 4-1 : Different case studies resulting from (i) the actual specimen geometry (ii) muscle route from origin to insertion, and (iii) type of muscle	55
Table 4-2 Design variables for calculating optimal muscle locations	69

1 Introduction

1.1 Overview

For many centuries, paleobiologists have been trying to answer many unanswered questions pertaining to evolutionary changes occurring in the species due to changes in the environment. Traditionally, thousands of years of evolutionary observation were the only methods available to such researchers to propose and validate hypotheses pertaining to how such animals lived and died. However, we believe that contemporary tools of biomechanical engineering analysis such as computational musculoskeletal analysis have reached a maturity that can be used to ratify or eliminate such hypotheses.

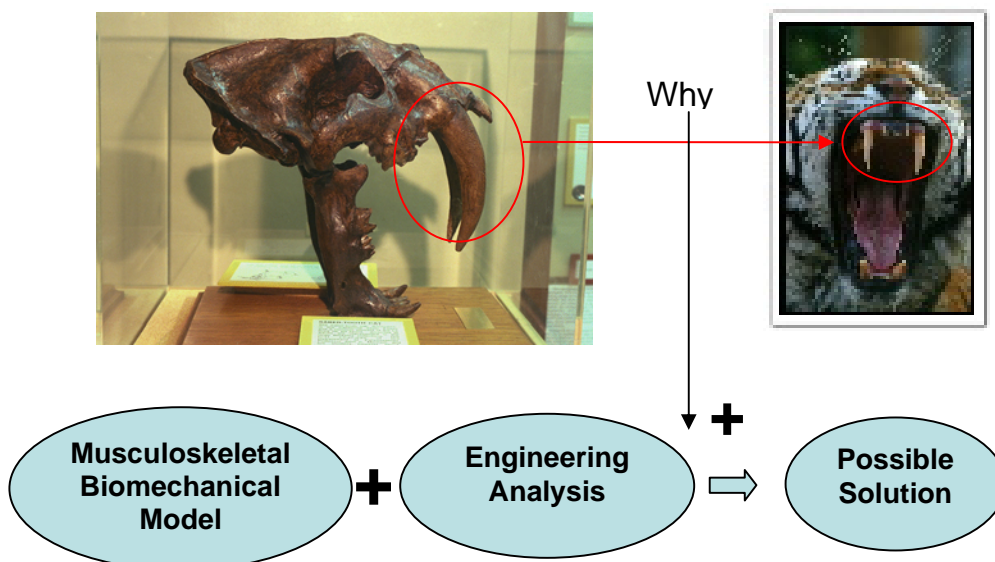


Figure 1-1Hypotheses regarding the functional significance of different dentitions in extinct and modern cats may be tested by the application of engineering concepts

Our efforts are focused on developing such a framework, identifying the critical issues, validating the tools, and finally presenting the biological scientists with a novel validated toolbox for use in computational testing of various hypotheses. Specifically, we highlight many of these aspects using the case study of bite & muscle force analysis in a sabertooth tiger. Paleobiologists have long wondered about the use of saber-like teeth of *Smilodon fatalis*. Many hypotheses have come to the forefront pertaining to the function of the saber teeth:

- *Did the Smilodon use its saberteeth during hunting? Or were these teeth purely ornamental to attract members of the opposite sex?*
- *Why do present members of feline family not have saber teeth?*
- *How did the presence of saber-like teeth offer any evolutionary advantages?*

Thus, the overall goal in this project is to extend the use of musculoskeletal analysis tools to perform bite & muscle force analysis which could then help answer some of the above questions.

1.2 Background

In the recent years there has been tremendous advancement in computer technology. The engineering domain had benefited significantly with the application of computational power in many fields like product design, development, manufacturing, analysis etc. In case of biological domain, factors such as: (i) inadequate biological computational tools;

(ii) inherent complexity of the system; or (iii) lack of exposure to advanced technology; has hindered the ability of the biological scientists to bring to bear the enormous computational power effectively and efficiently.

With the advent of the modeling techniques like Virtual Prototyping (VP) the manufacturing world has saved time and money right from product design to its final stages of development. This success in the engineering domain has served to inspire a similar use of VP for modeling and analysis of biological systems. In order for the analysis to make sense in the biological field, it becomes necessary to carefully model various biological components like bones, muscles, tendons etc. which can then be used to create a musculoskeletal model to serve as the virtual prototype.

1.3 Virtual Prototyping

“Virtual prototype is a computer simulation of a physical product that can be presented, analyzed, and tested from concerned product life-cycle aspects such as design /engineering, manufacturing, service, and recycling as if on a real physical model. The construction and testing of a virtual prototype is called Virtual Prototyping (VP).” [1]

Virtual Prototyping, also known as Simulation-Based design (SBD), has rapidly gained popularity and became a crucial part of most engineering design processes [2]. Virtual prototyping in the form of mechanical simulation computer models have been used by researchers, clinical professionals and commercial companies to study human movement and to develop products used by humans. Computer simulation can now be used to calculate the kinematic, dynamic and FEA based responses of a prototype and the results obtained can be visualized in a 3D interactive virtual environment. This will allow

the designers to realistically, accurately and quantitatively prototype and test multiple intermediate models within a virtual environment.

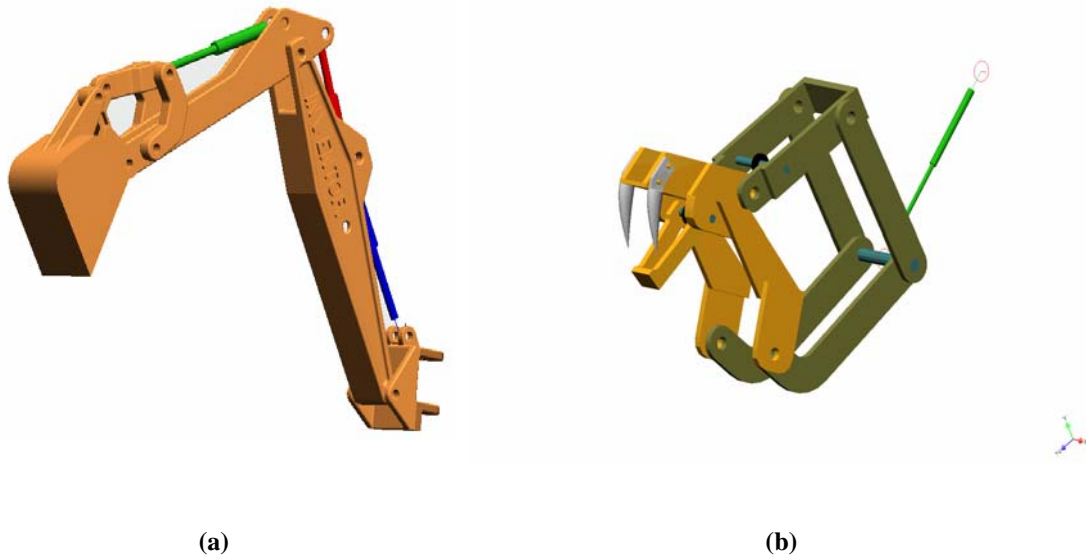


Figure 1-2 (a) Virtual Model of backhoe (b) Virtual Prototype of Sabertooth tiger (courtesy Mike)

Some of the *advantages* of using VP techniques include: (i) the ability to accelerate and improve the product life cycle (ii) use as a tool for testing and analyzing various ‘if then’ scenarios and (iii) ability to develop the final physical prototype without the need for further modifications. VP also has certain *disadvantages or limitations* which need to be carefully considered. These include the fact that the accuracy of the analysis results for a virtual prototype depends on factors like (a) the skill of the designer, (b) availability of computational power, (c) level of detail, etc. Further, as the intricacy of the system increases, the effort, skill and the computational power required for developing a virtual prototype increases exponentially. We will consider some of these aspects carefully within this project presentation.

1.4 Biomechanical Modeling

“Biomechanics is the science concerned with the internal and external forces acting on the human body and the effects produced by these forces”. [3]

From the mechanical point of view, biomechanical analysis of functional performance consists of two basic areas, *kinematics* and *kinetics*. Kinematics, sometimes referred to as the “geometry of motion”, describes the fundamental characteristics of motion such as displacement, velocity and acceleration of an individual body segment as part of the overall animal body. Kinetic analysis refers to those analyses which focus on the forces and energetics (intra and inter segmental energy flows) associated with motion. Most often, in biomechanical analysis, kinetics are obtained indirectly by calculating joint reaction forces and net muscle moments.

Kinematic and kinetic analysis are conducted by modeling the various body segments as rigid links using kinematic and morphometric data in order to satisfy two basic requirements. First, there is great interest in pulling together a collection of components such that the system mathematical model may be easily “composed” based on the physical inter connection of the component parts with their component mathematical models. Second, such system may then be modularly composed and encapsulated to any desired level of detail. These are two requirements that we will look for in all our modeling efforts.

From the biomechanics point of view there are mainly four types of models that can be constructed. *Conceptual model*, used to make a point without having the need for mathematical analysis, is rarely used because of its extremely simple nature and inability to prove hypothesis. *Mathematical/Analytical models* are generally used to represent

simple models. Such models can be analyzed by developing the equations of motions and solving for them analytically. These models can be used as a starting step in the analysis process. However, they cannot accurately model complex geometries like the ones encountered in biomechanical modeling. *CAD based models* are used to represent complex systems where it is difficult to derive the equations of motion analytically. These models are used to represent overly simplified biological models. In order to represent a more complete system we need to develop *musculoskeletal models*.

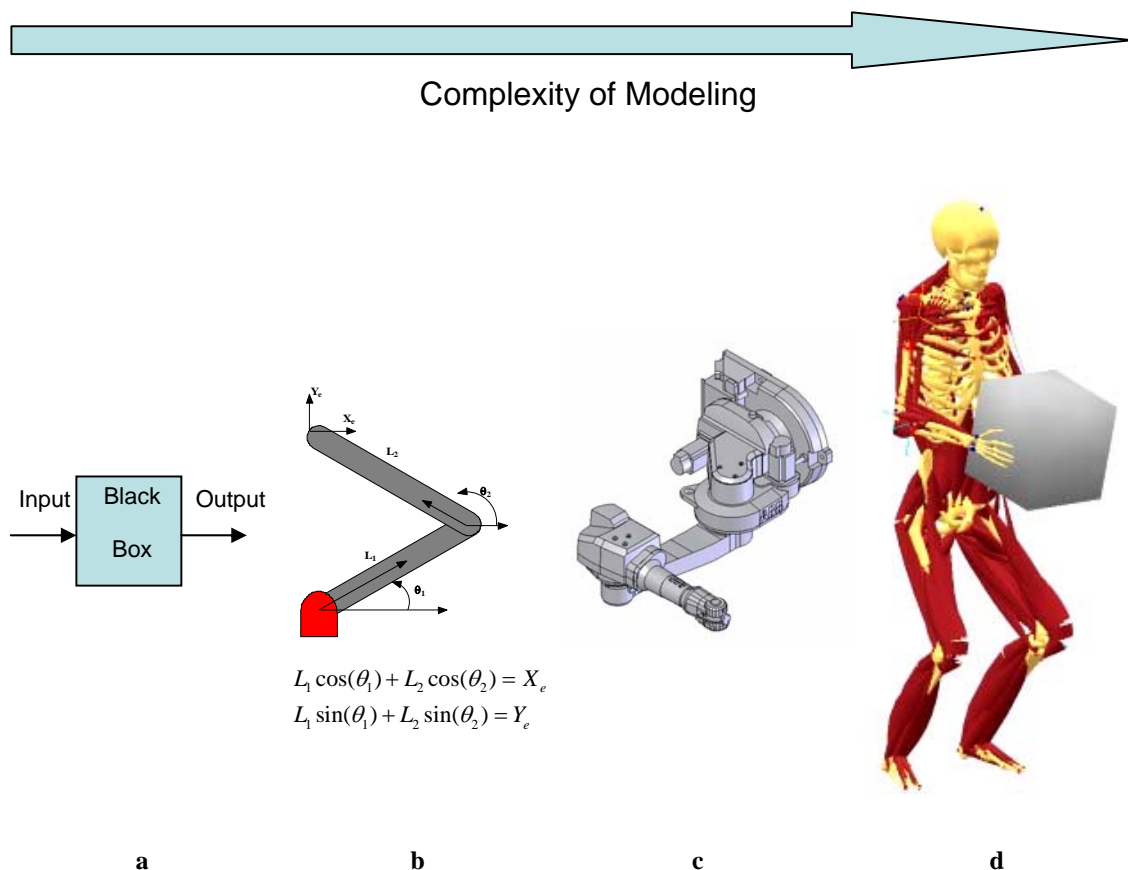


Figure 1-3 Different modeling procedures (a) Conceptual model (b) Analytical model (c) CAD model (d) Musculoskeletal model

1.5 Musculoskeletal Modeling

In order to satisfy the project goal, i.e. to study the functional performance of species, we need to construct an accurate biomechanically constrained musculoskeletal model. A musculoskeletal model can be defined as a “unique representation of skeletal and muscular structure of human or an animal”, utilizing the concepts of biomechanical modeling. The basic elements of a musculoskeletal system are bones, joints, muscles, tendons etc. Musculoskeletal systems can be thought of as articulated mechanical systems composed from these elements by linking together bones (links) using joints(kinematic pairs). The motion constraints induced by these kinematic pairs result in the desired motion of the articulated system (musculoskeletal system).

Such musculoskeletal models provide almost similar benefits to biological community as engineering models do to the engineering community. These advantages can range significantly-for example: (i) In neuroprosthesis, simulation performance of the musculoskeletal model can be helpful to eliminate the human experimentation at least in the initial phases. (ii) Limitations on in-vivo experimentation due to ethical reasons can be overcome with computational testing and helps in the functional and behavioral analysis. (iii) In exercise and sports sciences, musculoskeletal analysis helps in the field of performance biomechanics by improving the skill with which a motor task (such as multi joint reaching, finger coordination etc) is performed or in improving the sports equipment.

1.6 Project Goals

Our main aim is to create a constrained musculoskeletal model by connecting together individual segments with biomechanical constraints to:

- 1. Examine various aspects of systematic musculoskeletal model building with the help detailed examples*
- 2. Explore various issues pertaining to the modeling and analysis of such systems and provide solutions.*

A number of case studies using musculoskeletal jaw models of members of the feline family (tiger, Smilodon) are used to highlight critical aspects. For example, these include (i) muscle force calculation, (ii) bite force analysis, (iii) determination of optimum muscle location points. The initial modeling and analysis will be done with respect to Tiger and the results obtained will be utilized to validate the steps in the analysis process. Once the confidence in the system is developed we will perform a similar procedure for extinct animal, Smilodon. Specifically we would like to identify some of the issues that might arise during the modeling and analysis phase and subsequently propose possible approaches to address them.

How do we effectively model a musculoskeletal virtual prototype with all its features captured?

The central paradigm in creating computational models for biological system concerns drawing an analogy between the biological systems and mechanical systems. This mechanical analogy is then embodied in the equations of motion. The final outcome thus depends on the skill of the designer. We need to incorporate the boundary conditions as well so that the model does not violate the physical constraints. The results obtained from the analysis of such a virtual prototype are useful only when the model is accurate and behaves close to reality.

Biological models are extremely complex in nature and it would be extremely difficult to incorporate all the details, hence the level of complexity depends on the type of problem and the desired accuracy of results. Factors such as skill of designer, fidelity of the model, selection of the simulation phenomena, availability of the computational power etc affect the accuracy and usefulness of the results. The model should represent a fine balance between fidelity and efficacy. It must not be overly complicated so as to have prohibitively long run times with more opportunities for inaccuracies and errors, and not too simple so that it might produce incorrect or undesirable response [4]. All the important factors that have considerable impact on the final outcome need to be identified initially.

Can a particular animal modeled as above bite through a particular prey?

Before running the simulation, the model must be validated against the experimental results or closed form analytical solutions or published results etc. Responses pertaining to position, velocity, forces etc of various joints and segments can also be used for validation process. If experimental data is not possible, then model parameters such as joint motion trajectories obtained from digitized videos can be used. Model validation must also include examining the response trends to any minor changes or perturbations in the input. During this process we need to observe the response of the system with other models or components in the model space or environment. For example, there are many hypotheses on smilodon hunting, such as, the smilodon bites through the neck of the prey or bite through the belly etc. The analysis results should complement or eliminate such hypothesis. One of the main limitations of computational/virtual tools is that it is possible to simulate the impossible scenarios as well (such as the

animal biting through hard bone which may not be possible). So care should be taken so that results do not contradict or violate the physical constraints pertaining to the system and/or the environment.

1.7 Challenges

The successful implementation of a Virtual Prototype faces many challenges which must be resolved (or at least addressed). These arise due to geometric complexity, modeling of muscles individually as well as in groups which we will briefly discuss next.

1.7.1 Geometric Complexity

Unlike in traditional engineering systems, virtual prototype of a biological system generally possesses considerable irregularities that need to be dealt carefully during the modeling phase. Musculoskeletal systems are geometrically complex by nature. The inherent complex nature can be attributed to the presence of large number of muscles being attached to the irregular bony surfaces. Geometric complexity can be observed in two phases 1.Modeling phase 2.Analysis phase. In the modeling phase the geometry of the bone need to be captured. In the analysis phase, unlike in static analysis, the muscle location and its line of action vary continuously throughout the simulation. Hence the entire geometry of the system is changing continuously, which need to be dealt carefully. The point data (CT/MRI data) obtained from the fossil remaining of extinct animals can be used to preserve the bone geometry of the system by converting into CAD model using MIMICS (Materialise's Interactive Medical Image Control System). The details of the process are explained in section 3.3.1

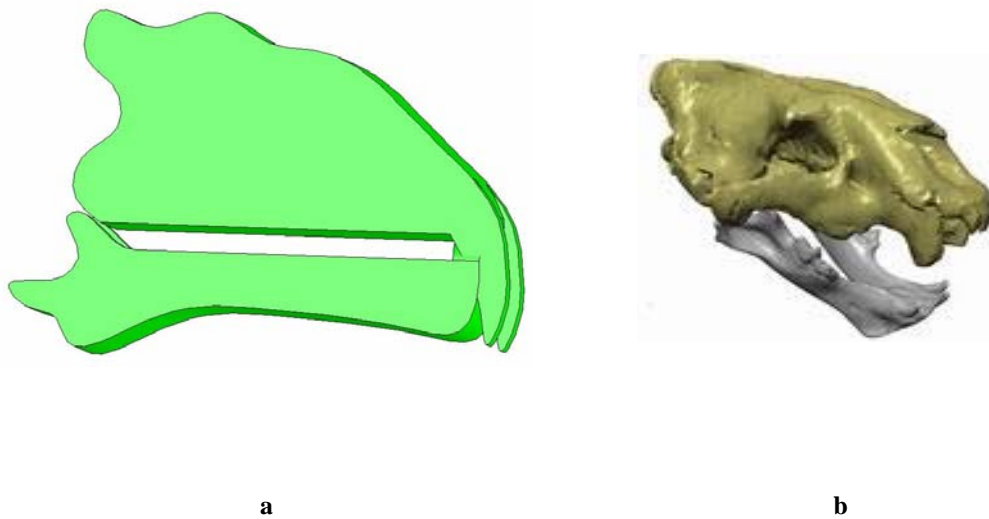


Figure 1-4 (a) Coarse model of skull and mandible of Sabertooth tiger (b) Fine model with all the geometry captured

1.7.2 Individual Muscle model

Bio mechanics researchers are not only interested in constructing musculoskeletal models but they are also interested in muscle forces and how the geometric relationships between muscles and bones transform into moments about the joints. In order to unveil these issues many mathematical models and software's are being developed. One such mathematical model of muscle that is being widely used in biomechanics domain is the Hill's Muscle model.

In studying the muscle properties, Hill used the reductionist approach i.e. obtaining the physiological data without directly attributing the observed behavior to any biophysical mechanisms. Ignoring the thermodynamic effects, Hill suggested that there are two inputs and one output for a muscle as shown in figure.

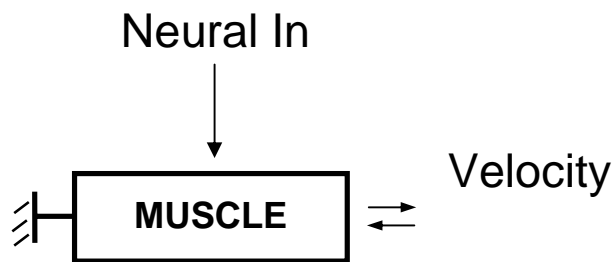


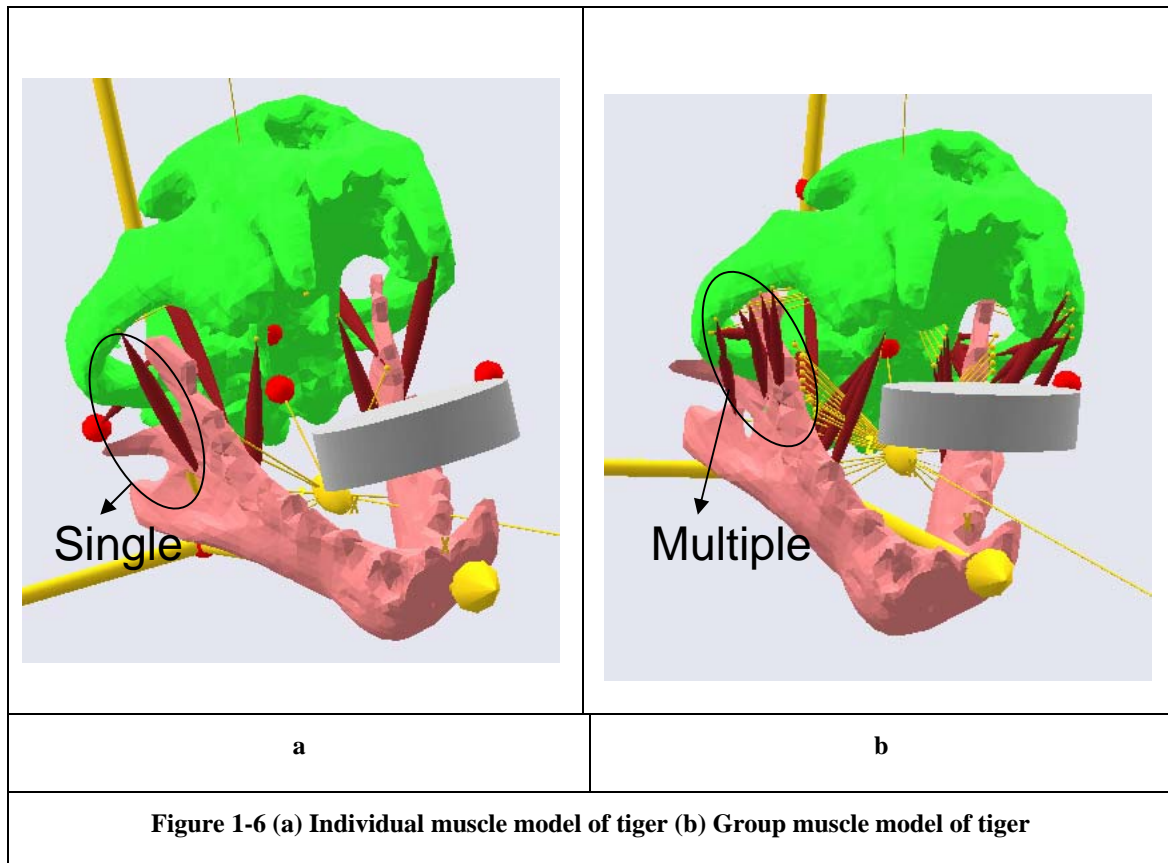
Figure 1-5 Black box model of muscle with two inputs and an output

The “Neural In” is the input obtained from the motor neuron (MN). There are two classical types of motor neuron inputs. 1. Twitch or the Impulse input, a brief electrical pulse causing sudden activation of the muscle. 2. Tetanus or step input which has the ability to generate maximum muscle force. This is a unidirectional input meaning the MN has direct effect on the muscle activation whereas the muscle activation doesn’t have a direct effect on MN. The other type of input to the muscle is the mechanical input from the mechanical interface i.e. where the muscle couples to the environment. It is a bidirectional input. Contact force f , and a kinematic variable v , velocity, have been chosen as the inputs that interface through this port. If force is the input velocity is the output and vice-versa. We will adopt this model despite the fact that this is a relatively simple model and more complex models exist [5]. The use of greater complexity in muscle modeling is one of the possible avenues of future research. For details pertaining to the characteristics of the muscle please refer to Appendix A.

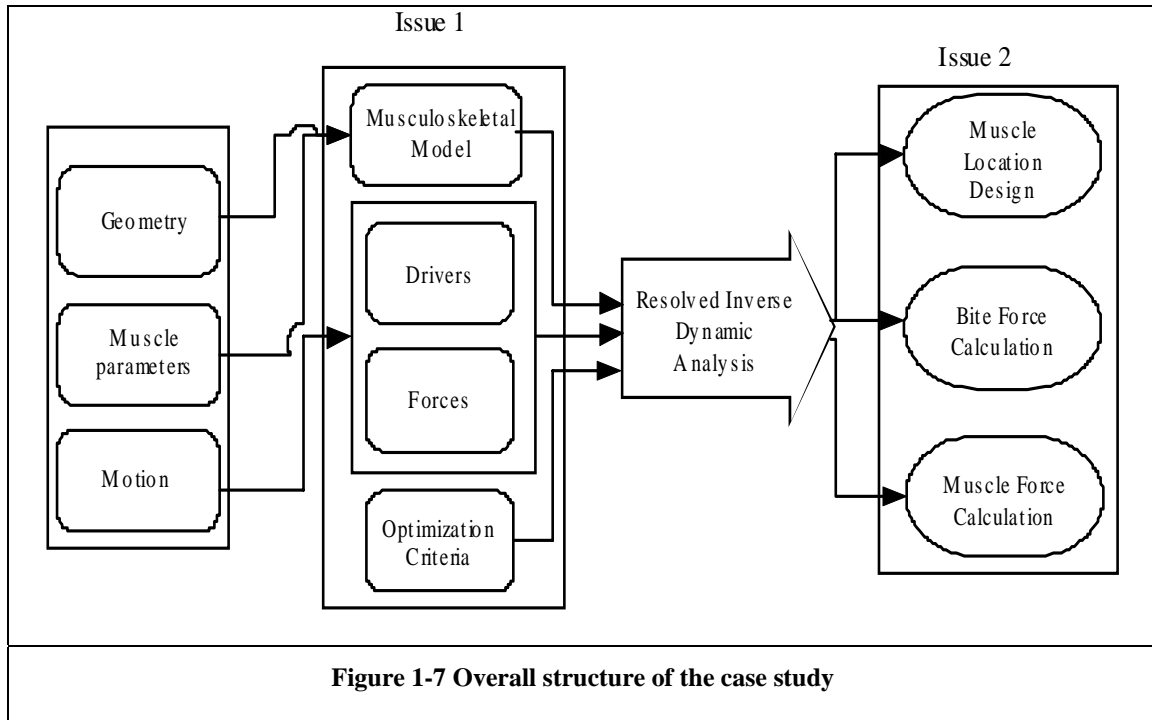
1.7.3 Group Muscle Modeling

The skull and mandible joint (of Tiger/ Smilodon) that form the major components of the bite force analysis can be thought of as a parallel chain manipulator with muscles acting as the links between mandible/lower jaw (base) and the skull/upper jaw (mobile manipulator). Simple jaw motions like opening and closing of jaw can be

achieved with the help of a single muscle or multiple muscles. As the musculoskeletal systems are redundant, and all the muscles do not get activated at the same instant, it is necessary to identify the muscle recruitment pattern for motion analysis.



In general, most of the musculoskeletal modeling packages assume that the muscles are attached to bones as point contacts but in a reality they are attached as surface contacts. This issue needs to be resolved for meaningful analysis results.



1.8 Project Organization

- Chapter 2 deals with literature survey, various musculoskeletal modeling softwares, their limitations and reasons for not being used in the project.
- Chapter 3 deals with AnyBody Software and its features. It gives a detailed explanation of modeling the virtual prototype and various analyses that can be run on the model within the limitations of the software. It also deals with the reasons why a MATLAB interface for AnyBody was developed and details its features.
- Chapter 4 first deals with background for calculating the muscle forces. It also deals with various case studies pertaining to Muscle force calculation, Bite force analysis and optimal muscle point location for both Tiger and Smilodon.
- Chapter 5 concludes the project with list of contributions in the work and provides suggestions for future work.

2 Background

Conventionally, biological studies have been unable to take full advantage of ubiquitous computing due to the lack of both computing tools as well as the technical knowledge on how to use these tools for various biological problems. However, this situation is rapidly changing with the recent advancements in the computational technology, the availability of easy to use tools, and perhaps most importantly, as greater numbers of bio problems are being tackled, the usefulness of the results is becoming more evident. While tremendous benefits can be gained from these tools, there are numerous issues as well. In this chapter we will review some of literature that discusses applications of such technological tools to biological problems. To help the focus narrow we will restrict attention to (i) some of the tools that are currently available and (ii) the computational musculoskeletal modeling application.

2.1 Musculoskeletal Modeling Applications

Musculoskeletal modeling has been widely used in a number of applications areas. Sample application include use for control of neuroprostheses where optimal stimulation patterns are determined to assist in functional electrical stimulation for restoring walking to paraplegia [6]; performance evaluation of physical activities such as lifting of loads [7]; performing parameter identification for modeling the upper extremity [8]; investigating biomechanical aspects of the evolution based on the hypothesis of dynamic cooperative interactions between the locomotion pattern and the body shape in the quadrupedal hopping and bipedal hopping [9]. We will restrict our attention to using bite force type analyses.

2.1.1 Bite Force Analysis

Bite-force is defined as the amount of force that can be exerted by the jaw adductor musculature and realized at the tooth row as a function of jaw geometry. It has been used by a number of researchers to study the functional performance and the behavioral analysis of animals. Bite force is one of the least investigated aspects that has been used to indicate the functional effects or morphological transitions in muscle placement throughout the evolution [10, 11]. Gape angle and bite point also play a significant role in the bite force analysis [12].

Meers [13] combined the bite force data with prey body mass in order to establish predator/prey relationships for *Triceratops horridus*, *Tyrannosaurus Rex* and other dinosaurs. Their relationships support the hypothesis that *Tyrannosaurus Rex* could have been a competent, solitary predator even of large herbivores such as an adult *Triceratops*. Such analysis can be used to prove such hypothesis. Through their analysis Verwaijen and his group [14] concluded that differences in head and body size, that has considerable impact on the bite force magnitude, may affect prey selection either directly or via handling efficiency. Raadsheerl and his group [15] examined the effect of jaw muscles and craniofacial morphology on human bite force magnitude and concluded that the contribution of the masseter muscle to the variation in bite force magnitude was higher than that of the craniofacial factors.

2.2 Musculoskeletal Analysis Tools

With the recent development in computational technology, a number of biomechanical analysis tools have been developed for modeling complex musculoskeletal

systems. In this section we will discuss some of the widely used tools and their limitations with respect to our analysis method and available data.

2.2.1 Software for Interactive Musculoskeletal Modeling (SIMM)

The SIMM Biomechanics Software Suite is a powerful and popular tool kit developed by MusculoGraphics, Inc to facilitate the modeling, animation and analysis of 3D musculoskeletal systems. Unlike traditional animation and CAD packages, the SIMM software suite is specifically designed to work with systems that consist of bones, muscles, ligaments and tendons. In this software, a musculoskeletal model consists of a set of rigid body segments that are connected together by joints. Muscles span around these joints and develop moment about the joints.

To define a musculoskeletal model we need to define three types of input files which define body segments or bones, joints and muscles. In order to develop a model in SIMM we need to create Bone, Joint and the Muscle files. The bone file contains lists of polygons that define the bone surfaces. The joint file specifies the kinematics of the joints and the characteristics of the body segments. The muscle file contains lists of coordinates that describe the muscle lines of action, and the parameters needed to compute isometric muscle forces. Apart from the muscle parameters like the maximum muscle force, fiber length, pennation angle etc we need to know the force-length relations of the muscles and tendons, and the force-velocity relation of the muscles.



Figure 2-1 Musculoskeletal model of human in SIMM

The data available for analysis is point data in the form of STL files that need to be converted into mesh format for creating bone files. Software tools like ‘Rapidform’ can be used to create solid models into surface or mesh files. However, we do not have the data required to create the muscle files. Building dynamic models with SIMM requires substantial C programming which limits its use. SIMM also has substantial limitations on its ability to incorporate runtime changes of muscle excitation and external forces, which handicap its use to study control algorithms [16]. Hence, the above limitations on the software restricted its usage for the present analysis.

2.2.2 SIM Mechanics

SIM Mechanics is used to model and analyze the mechanical systems in a simulation environment. We can model complex systems, simulate them and analyze the results without having the need to derive mathematical equations. Since it forms part of MATLAB we can combine our model with other MATLAB products for further analysis. Some of the key features of the software are:

- Modeling of three dimensional rigid body mechanical systems with block diagrams in Simulink.

- Performing kinematic, forward, and inverse dynamic analysis of the systems.
- Providing visualization tools for animating the models.
- Enabling the implementation of high-fidelity, nonlinear plant models in Simulink to support the development and testing of control systems.

In SIM Mechanics we need to model the rigid bodies along with the mass and inertia properties, define the joints connecting the rigid bodies in order to constrain the motions and finally the drivers to move these joints. In case of musculoskeletal models we need to define the mathematical models of muscles (obtained from physical experiments). It becomes very difficult to develop such models with all the characteristics incorporated. Though many people tried to develop mathematical muscle models [17] they could only capture part of the muscle properties. These difficulties have restricted us from using SIM Mechanics.

2.2.3 Virtual Muscle

Virtual Muscle is a MATLAB tool to create Simulink blocks of the muscles with the help of a graphical user interface. It is composed of two programs, 'BuildFiberTypes' and 'BuildMuscles'. Using these two programs we can create a variety of muscles by defining various muscle parameters. The Simulink muscle block created by BuildMuscles can be incorporated into a users Simulink model of the NMS (Neuromusculoskeletal) system, or can be incorporated with MMS (Musculoskeletal Modeling in Simulink) and SIMM [18]. In order to define a muscle we need to specify a large number of parameters, for which we do have the data. The limitations of the model are its inability to simulate pennated muscle and its intensive computational requirements. Each additional motor unit simulated will increase the computational time required, and users need to trade-off

speed versus accuracy. With multiple muscles in the system, the software doesn't provide any algorithm for solving the muscle recruitment pattern problem.

2.2.4 LifeMod

Brg.LifeMod is musculoskeletal software developed by Biomechanics Research Group Inc. (BRG) to develop human computer models in order to study human motion and other activities. Here the human is viewed as a mechanical system just like a car or bike. The body interacts with its environment by virtue of neuromuscular system creating loads on bones, joints and muscles. Some of the applications include vehicular crash analysis, development of external prosthetics, gain insight into reconstruction methods and surgical protocols, analysis of rehabilitation tools etc.

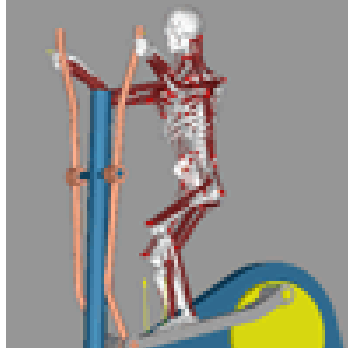


Figure 2-2 Musculoskeletal model of human in LifeMod.

The inability of the software to load STL files or CT/MRI files has restricted its usage. Though we can perform forward and inverse dynamic analysis, the data available with us is not compatible to the requirement of the software.

2.2.5 Visual Nastran

Visual Nastran is a mechanical simulation platform for the integration of motion and stress simulation into a single functional modeling system. It also provides for data transfer in standard formats, including STEP, Parasolid, ACIS, IGES and STL. The software has the capability to integrate with Matlab Simulink but it doesn't have the capability to attach muscles and ligaments directly to the models. Though we can use software like Virtual Muscle to attach muscles and ligaments to the model the problem of multiple muscle recruitment can not be solved.

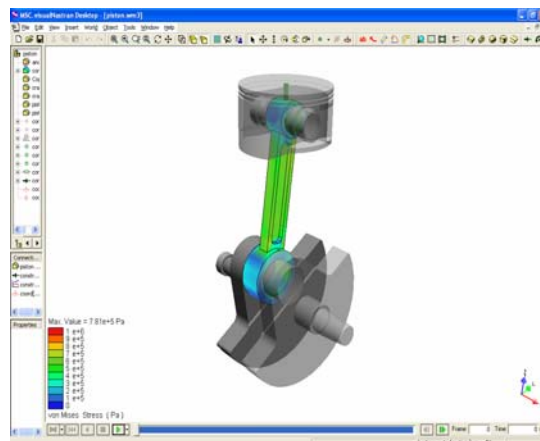


Figure 2-3 Sample Visual Nastran model of cam and piston

2.2.6 Vertebrate Analyzer (VA)

The Vertebrate Analyzer (VA) [19] is aimed at building a toolkit for constructing, articulating and visualizing physically accurate and physiologically constrained biomechanical models that can be programmed to emulate known or hypothesized

behavior. We can visualize and experiment with these accurate biomechanically constrained models in order to test hypothesis relating to form and function. The present version has the capability to assemble skeleton, add muscles, tendons and ligaments with specified characteristics and determine the feasibility or efficiency of a movement. The present VA toolkit contains models of the sabertooth cat, a tiger, a lion and a human skull, and is still under the developmental stage. The proposed future work would be an ideal package for testing hypothesis and study the functional performance of extinct and extant animals.

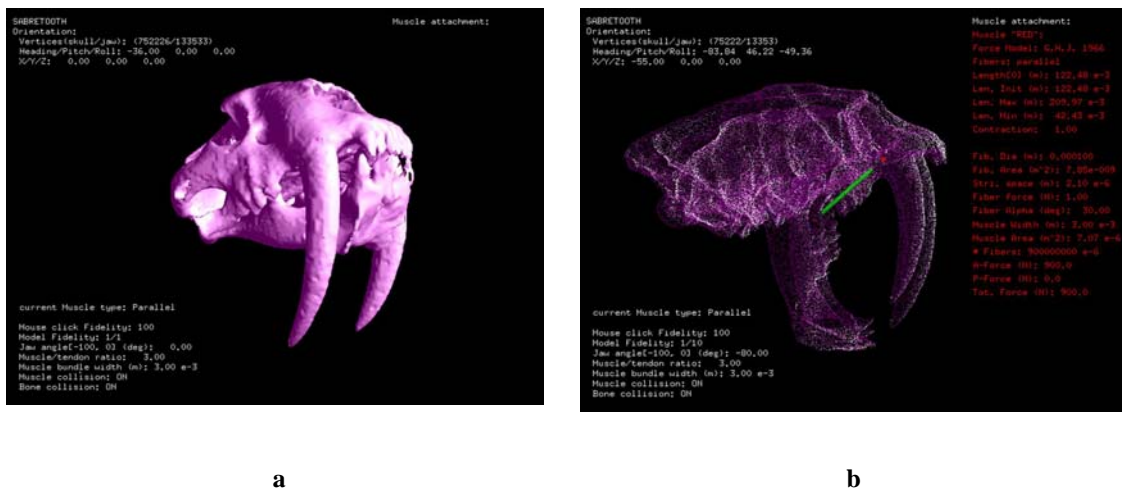


Figure 2-4 (a) Shaded model of Smilodon in Vertebrate Analyzer, (b) "Vertex-based" Smilodon with muscle and corresponding muscle characteristics in VA

2.2.7 AnyBody

AnyBody [20] is musculoskeletal modeling software developed as a part of the 'The AnyBody Project'. It provides numerous facilities like (i) adding real bones geometries to the models, (ii) performing inverse dynamic analysis for calculating muscle

forces, (iii) execute the analysis from an external environment using the command line version, (iv) solving the problems of redundancy, multiple muscle recruitment pattern etc. All the features provide us with an ideal environment to explore our research goals. Hence, in this project we propose to use “AnyBody” to explore our multifold research interest. Further details about the software implementation are given in Chapter 3.

2.3 Applications

A musculoskeletal model can be used in a number of application arenas.

- *Therapy*: Musculoskeletal analysis can be used in the field of therapy for studying the recovery level of the patient. By defining the human model and the environment, we can run the motion analysis through a variety of simulation tasks. By observing the muscle recruitment pattern and the amount of force the muscle can build while performing the task, a therapist can comment on the recovery process of the patient. Such a technique can also be used as a test bed.
- *Ergonomic Design*: All the parameters of the human model and the boundary conditions are parametric. These parameters can be used as design variable in optimization problems for designing products such as vehicles, sports equipment, rehabilitation tools, hand tools etc. A concept of example is provided by group of researchers in Denmark [21].
- It can be used in studying the functional performance of extinct and extant animals through engineering analysis by constructing an accurate biomechanically constrained model (this application arena is explored in our project).
- Musculoskeletal models when combined with the virtual feedback environment can be used for training surgeons and evaluate the quality of surgery as well.

3 Musculoskeletal Modeling With AnyBody Software

This chapter introduces the reader to musculoskeletal modeling using AnyBody software. It presents the modeling procedure in some detail with emphasis on defining terminology for the system components. Additionally, aspects of the types of analyses that are possible are presented and discussed. We then present various steps taken to create models of the sabertooth tiger and tiger jaws in this AnyBody software.

3.1 What is AnyBody?

“AnyBody” is a musculoskeletal modeling software used for developing detailed multi body biomechanical systems. The software was developed as a part of the ‘The AnyBody Project’. It is script based, where writing programs is needed to develop the musculoskeletal models. The scripting language, called ‘Any Script’, is an object oriented language similar to C++ or Java Script.

This software has the capability to perform analysis of the body models that are complex in terms of (i) geometry, (ii) number of muscles, (iii) dynamic changes in muscles line of action, and (iv) the number of degrees of freedom. The system can handle several hundreds of muscles and the whole body models. It also has the capability to import ‘STL’ files which suggest that analysis is not only restricted to human models, but also can be extended to any other animal as well.

3.2 Modeling Procedure

A musculoskeletal model is a combination of bones, muscles, tendons, joints, etc. The software is developed in such a way that all these elements are built in a systematic manner to perform kinematic and dynamic analysis. In this section, we describe the overall procedure of building a musculoskeletal model in AnyBody, as shown in Figure 3-1.

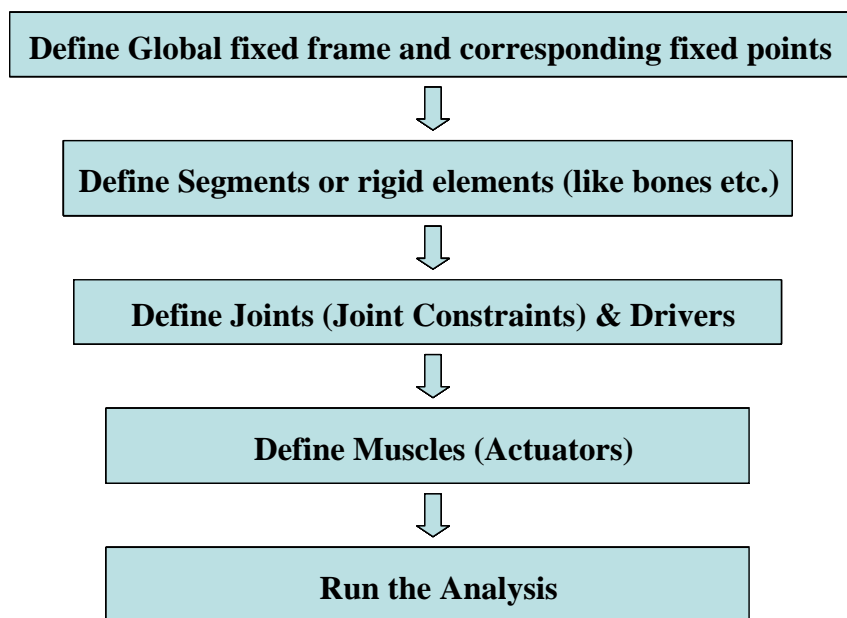


Figure 3-1 Simplified modeling procedure in AnyBody

Step1: Define Global or Inertial fixed frame

In any other software that can analyze a musculoskeletal system, we need to define a reference frame for the entire model. These are predefined in some of the other software. However in AnyBody, we need to explicitly define a world frame (or a Global fixed frame) in the form of class. Everything thing else defined in this model will be

directly or indirectly related to this global frame. All the points that need to be fixed during the simulation are defined in this class. The initial window to write script is as shown in Figure 3-2. The various analyses performed for the model, such as Kinematic, Inverse Dynamic and Muscle Calibration analyses, are referred to as studies. Details of these studies are given in the analyses section. A detailed explanation of the syntactic notation and the software structure is given in [22].

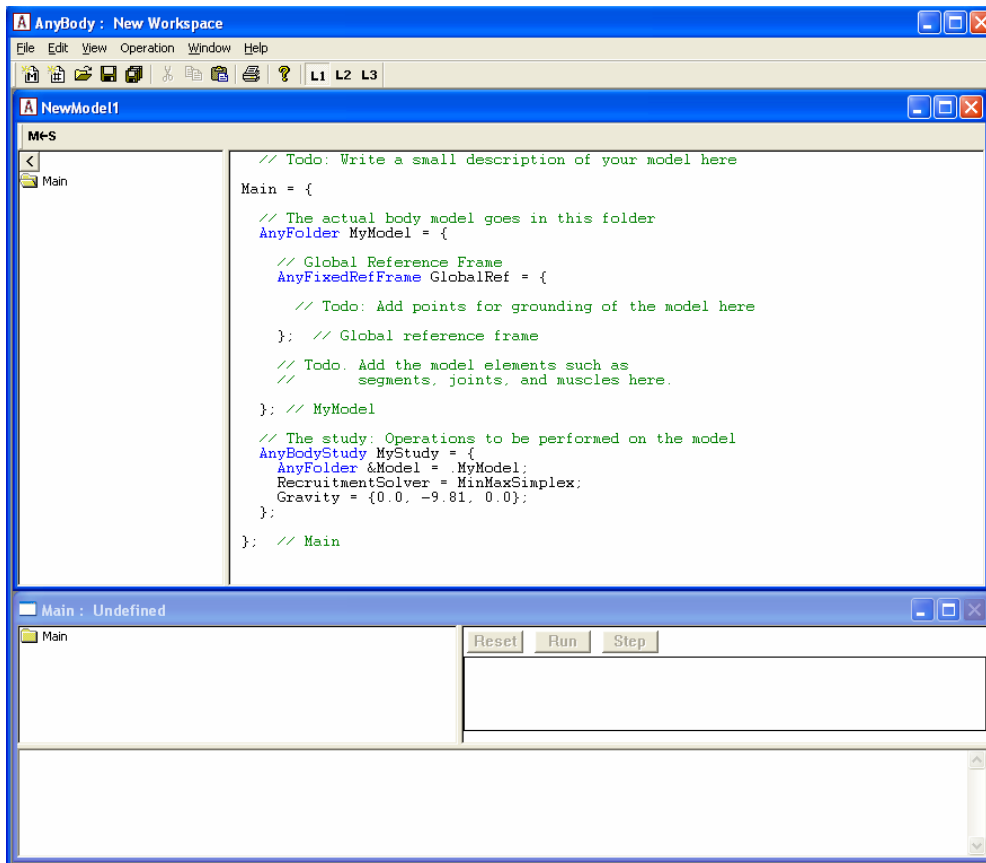


Figure 3-2 AnyBody Software Interface

Step 2: Define Segments

The next step in building the model is to define the body parts or the ‘segments’. Segments constitute the rigid elements of the model that moves around.

Step3: Define Joints and Drivers

In any musculoskeletal model the body movements are constrained, i.e. there are limitations on the body motions. These constrained motions are provided by the joints that attach the segments. So, joints need to be defined in order to constrain the system movements. Once the joints are defined, we define drivers that move the joints. Drivers provide the input for body motions.

Step4: Define Muscles

In any mechanical system drivers are used to move the system but in musculoskeletal models muscles serve this purpose. So we need to define the muscles. Once the entire system is defined, we perform the inverse dynamic analysis in order to calculate the muscle forces. Though the system is actually moved by the drivers the analysis does a back calculation such that muscles act as actuators that move the system and the drivers remain passive. This is the main concept behind analysis.

Step5: Analysis

Once the model is built, the software has the capability to perform a number of studies that will help in the analysis of the results. There are mainly 4 different modes in the AnyBody study. They are:

- Setting Initial conditions
- Kinematic Analysis
- Inverse Dynamic Analysis
- Muscle Calibration Analysis

Setting Initial Conditions: Often times, it is required to position the model in a predefined position to start the analysis. This feature helps in tuning the parameters that define the initial configuration of the model, i.e. the starting position of the simulation. In order to perform any type of analysis, the model needs to be kinematically determinate right from the initial stage.

Kinematic Analysis: In kinematic analysis, the main emphasis is on the system motion without taking into consideration of the forces or reactions that are acting on/by the system. However, the system has to be kinematically determinate, where the number of equations to solve the system is equal to the number of unknowns. If there are 'n' segments in the system the model will have a total of $6n$ degrees of freedom, unless they are constrained. In order to solve these $6n$ unknowns, we need $6n$ equations. The main purpose of this kinematic analysis is to solve for these $6n$ unknowns, i.e. the position and orientation of all the segments at all times through out the simulation process. From the kinematic analysis we can obtain the position, velocity and acceleration information for each and every point of interest.

Inverse Dynamic Analysis: Inverse dynamics analysis (IDA) can be thought as the main process that lead to the success of AnyBody modeling system. Muscle and reaction forces are calculated by setting up the equations of motion. Problems pertaining to static indeterminacy, limitations on the maximum muscle forces need to be resolved before the analysis. Redundancy posed by the system indicates that there is no unique solution to the inverse dynamics problem. While performing a body motion the muscles collaborate according to some rational criteria. These criteria when combined with the fact that muscles can only pull and not push results in an unique recruitments pattern. This suggested that the central nervous system applies some sort of optimality criteria to

determine the muscle activation order. If such an optimal criteria is combined with the equilibrium equations, we can have unique solution for the problem. The basic optimality assumption is that “the body attempts to use its muscles in such a way that fatigue is postponed as far as possible”. Hence the optimization problem for calculating the muscle forces can be mathematically written as:

Minimize (Maximum muscle activity)

+ $e1$ *(sum of activities)

+ $e2$ *(sum of squared activities)

Subject to

$$Cf=d \tag{3.2.1}$$

$$\frac{F_{M,i}}{N_i} \leq \beta, i \in \{1, \dots, n\} \tag{3.2.2}$$

$$F_{M,i} \geq 0, i \in \{1, \dots, n\} \tag{3.2.3}$$

Where,

$F_{M,i}$ is te muscle of ith muscle

N_i represent the normalization factor

$e1 \rightarrow$ RecruitmentLpPenalty or recruitment linear penalty

$e2 \rightarrow$ RecruitmentQpPenalty or recruitment quadratic penalty

Equation (3.2.1) represents the equilibrium equations, the second constraint i.e. equation (3.2.2) takes care of the muscle activity limits and the non negativity constraint i.e. equation (3.2.3) on the muscles makes sure that the muscles can only pull.

These terms play a role to eliminate any unrealistic fluctuations during the simulation of complex systems. The main advantage of using such a min/max optimization algorithm is that it guarantees that there is no other muscle recruitment pattern that will lead to a smaller maximum activity.

Muscle Calibration Analysis: A muscle tendon unit is generally attached to two segments i.e. one at the origin and the other at the insertion. These muscles wrap around the joints and bones. During the range of motion it is assumed that the muscle-tendon unit has its optimal length at some particular position of the joint span. The muscle calibration analysis adjusts the length of the tendons such that the optimal length is obtained in the middle of the simulation. If we run the IDA after the calibration analysis, the simulation makes use of these calibrated tendon lengths for the calculating the muscle forces. AnyBody has three types of muscle models. The simple muscle model is not affected by the calibration analysis while the other two that have force-length relationships (Hill Model) do have an affect.

3.3 Our Model in AnyBody

The data available for building musculoskeletal model is the point data obtained from CT/MRI (Computer Tomography / Magnetic Resonance Imaging) images in the form of DICOM files. DICOM(Digital Imaging and Communications in Medicine) is a standard created by the National Electrical Manufacturers Association (NEMA) to aid the distribution and viewing of medical images, such as CT scans, MRI scans, and ultrasound [23].

3.3.1 Creating a CAD Model from Point Data (CT/MRI Data)

DICOM files of skull and mandible are available for tiger/Smilodon. These DICOM files need to be converted into STL (ASCII) files in order for them to be imported into AnyBody. MIMICS (Materialise's Interactive Medical Image Control System) are an interactive tool used for developing CAD models (STL, IGES etc) from CT scan images or MRI images. MIMICS8.13 and above versions are capable in converting these DICOM files into STL (ASCII) files. This section gives a brief view of converting DICOM files into STL files.

Step1: Importing

After importing the DICOM files into MIMICS software system, as shown in Figure 3-3, we need to indicate the orientation (Anterior or Posterior, Right or Left and Top or Bottom portions) of the CT/MRI image.

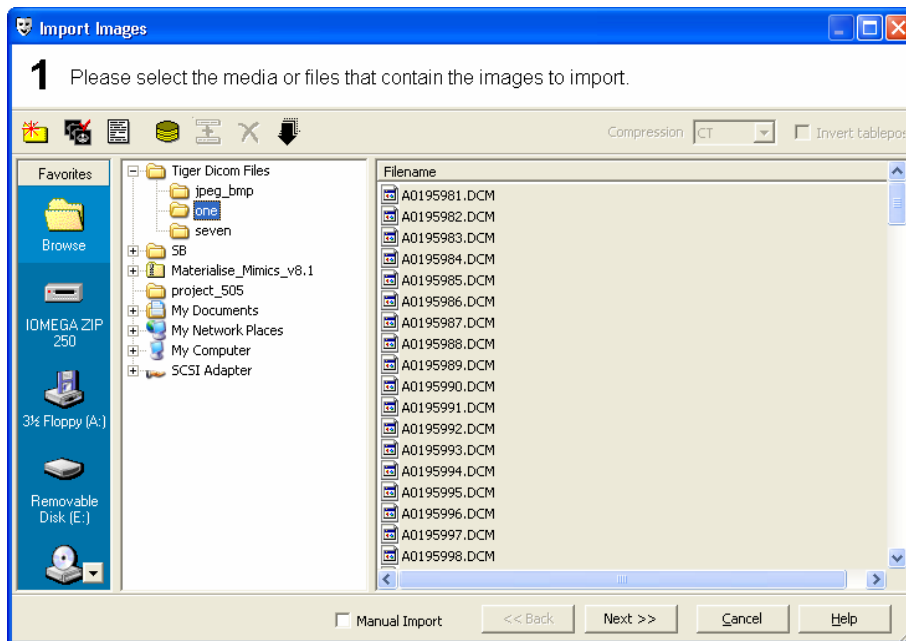


Figure 3-3 Importing DICOM files into MIMICS software

Step2: Windowing:

Once the image is adjusted as shown in Figure 3-4, we can adjust the gray scale in order to distinguish between the bright and dark regions by adjusting the gray scale. This operation is called ‘Windowing’.

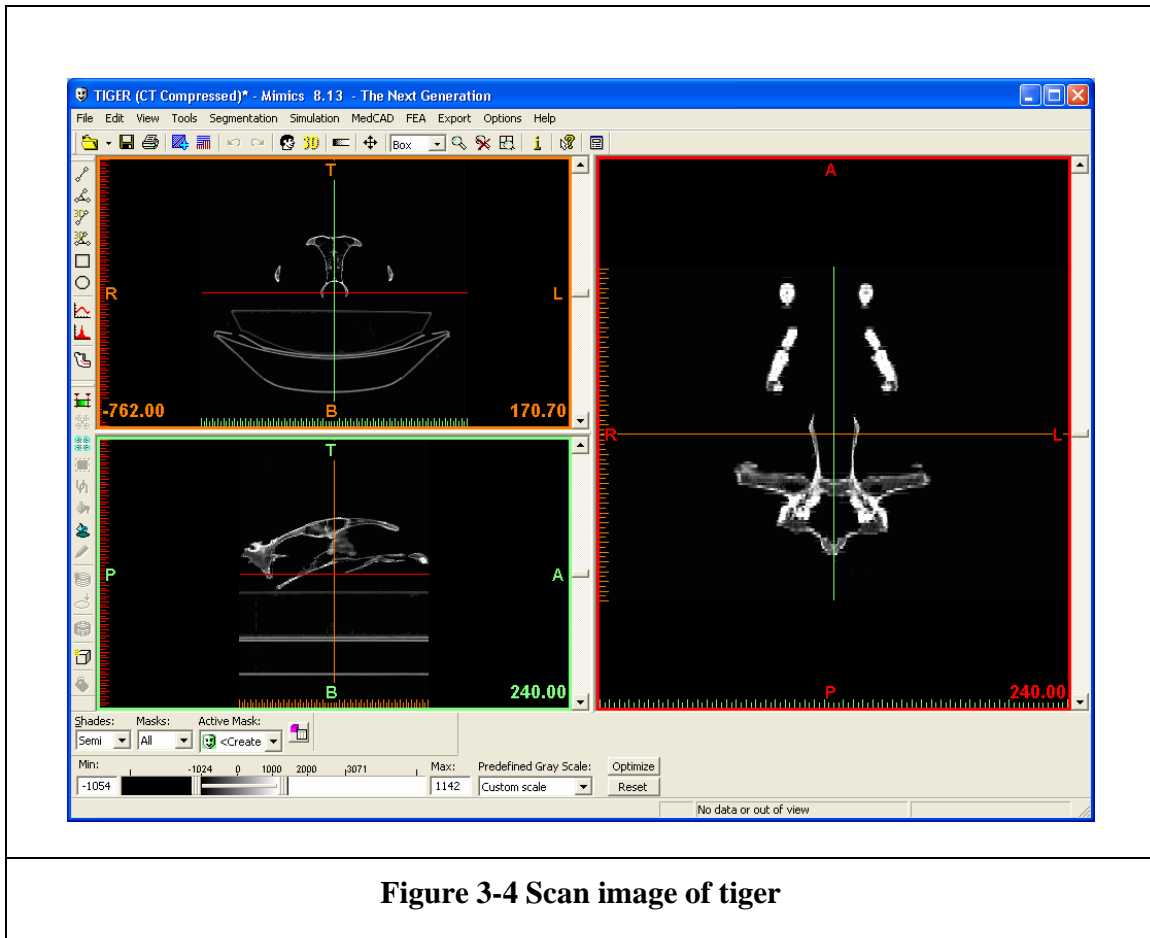


Figure 3-4 Scan image of tiger

Step3: Segmentation

Segmentation is the process of sorting (segmenting) pixels in a CT digital image by gray scale or density. Different materials impede X-rays differently so they are depicted in different shades of gray. Areas with similar density are assumed to be of the same material. We have to use segmentation tools for improving the quality of the final

outcome. In the current work segmentation is used to distinguish soft bone from hard bone. The different techniques used in segmentation are described below

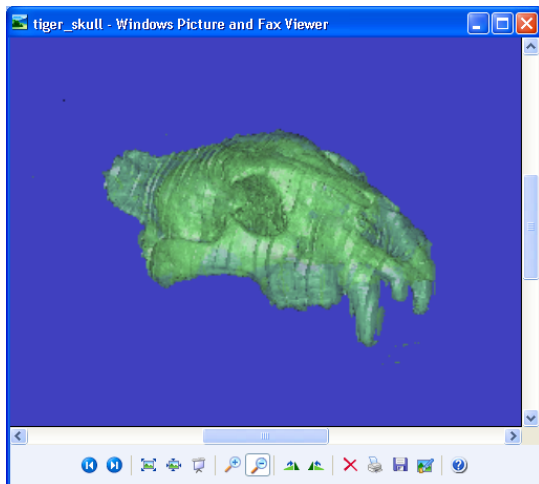
(a) *Thresholding*: In biological terms, thresholding gives the user capability to select between soft tissues and hard bone. Thresholding implies that the segmentation object (visualized by a colored mask) contains only those pixels of the image whose value is higher than or equal to the threshold value. A low threshold value makes it possible to select the soft tissue of the scanned patient. With a high threshold, only very dense parts remain selected. We can select the threshold value using a slider bar or by using a 'Profile Line'. Please refer to [24] for further details.

(b) *Region Growing*: Region growing permits splitting of object into several parts (soft and hard bone). In this technique, one can select a point within a feature and then expand the region.

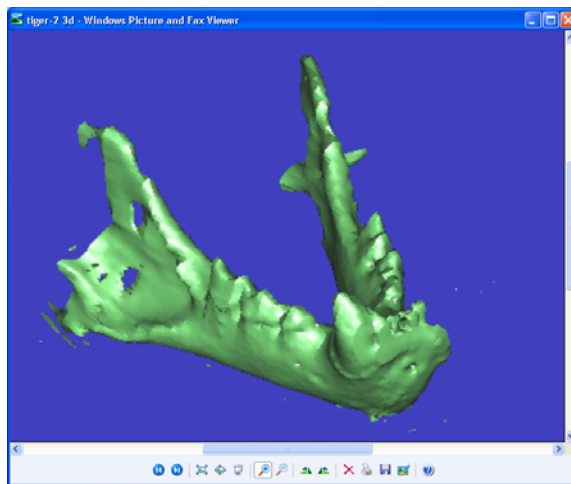
(c) *Editing*: After thresholding and region growing, we can remove any unwanted portions (like table or any other spots present) from all the scan images. This can be done using 'Edit Masks'.

The scanned images is thus free of artifacts and will be used for building a 3D model by selecting 'Calculate 3D' from segmentation menu (result is as shown in Figure 3-5). If any modifications need to be made to the model, we can make them using 'Edit Masks' and again build the 3D model. There are other functionalities like Morphology operations, Boolean operations, etc. that can be used for fine tuning the image.

The final 3D model is converted into STL files for importing into AnyBody. This can be achieved using the STL module in MIMICS. This module has the capability to convert 3D objects into STL (either Binary or ASCII) files (Figure 3-6 a). We can set up the quality of the STL file as well (Figure 3-6 b).

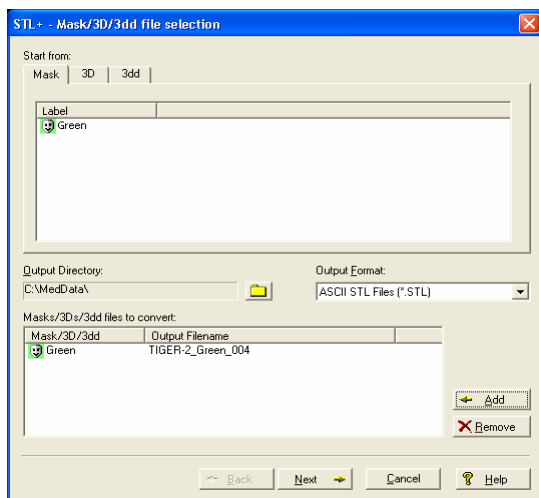


a

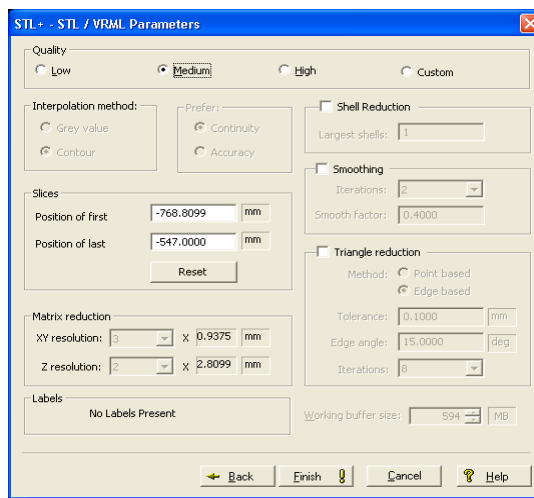


b

Figure 3-5 (a) 3D model of tiger skull, (b) 3D model of tiger mandible



a



b

Figure 3-6 (a) Window showing the STL file selection, (b) Window showing the properties of STL file

3.4 Building Musculoskeletal Model in AnyBody

A musculoskeletal model in AnyBody has 4 basis building blocks namely segments, joints, drivers and muscles. We will highlight the on the details of model creation using the case study of a tiger ajw modeling with in the AnyBody software.

Procedure:

Define the global fixed frame or the inertial frame. As mentioned earlier we define all the fixed points in the global fixed frame. The result of defining the global frame can be seen visually in the “model window” as shown in Figure 3-7.

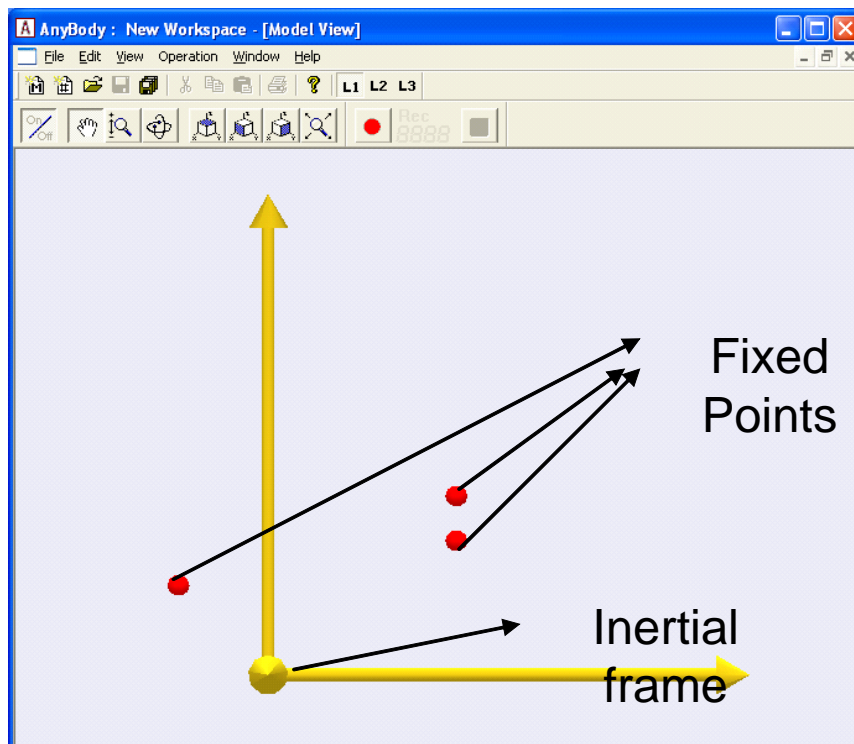


Figure 3-7 Global reference frame

Once the inertial reference frame is defined, segments for our musculoskeletal model are defined. A segment is nothing more than a rigid body that can move around in space with six degrees of freedom. Each segment can be represented by a reference frame with its origin where the center of mass is located and axes coinciding with the principal inertia axis. We can also define the muscle attachment locations on these segments.

A segment is like a free body in space. So it has six degrees of freedom. In our model we have two segments, 'UpperJaw' and 'LowerJaw'. While creating the Upper Jaw we need to define the position of its center of mass with respect to the Global frame. All other nodes we define in this upper jaw are related to the frame attached at the center of mass. Properties such as mass of the segment (Mass), its orientation in space (Axes) and the Inertia tensor (J_{ii}) for the segment are defined. We can also define nodes to which Muscles, Joints, Drivers or STL files will be attached. The result of defining the upper jaw is as shown in Figure 3-8.

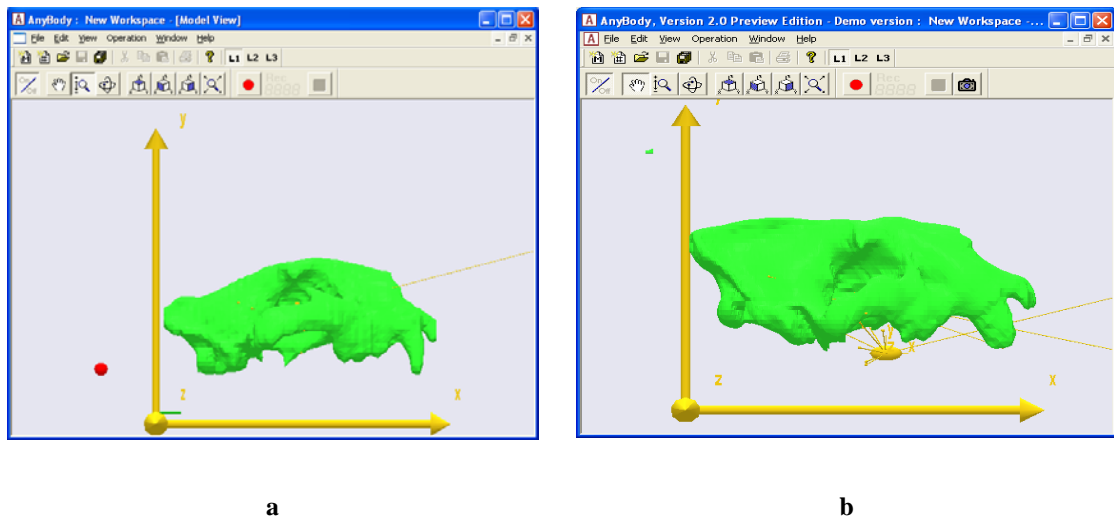


Figure 3-8 (a) Skull of tiger, (b) Skull of Sabertooth tiger

In a similar procedure we define the 'LowerJaw' for the model (Figure 3-9).

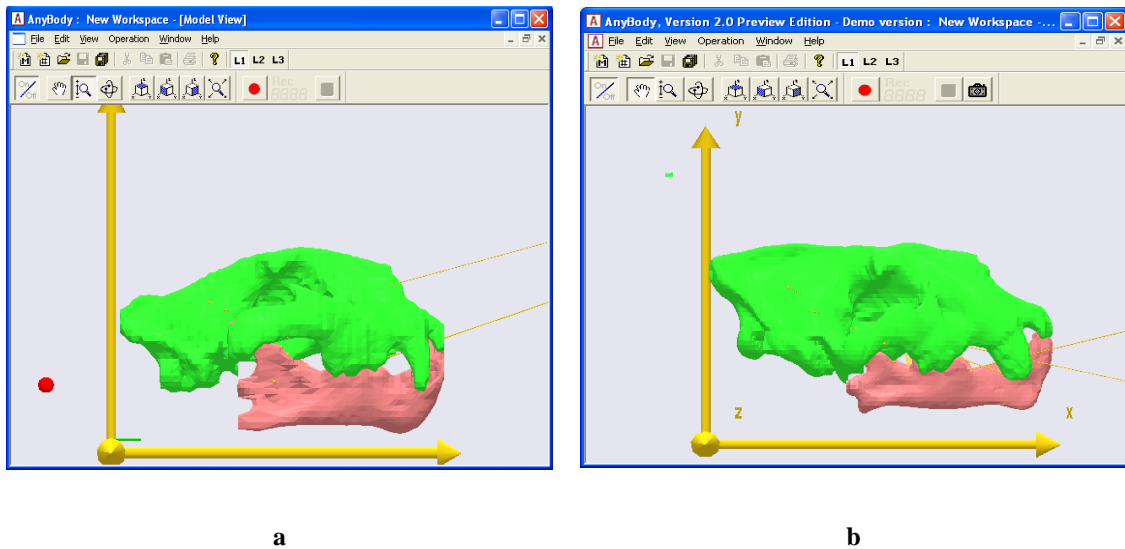


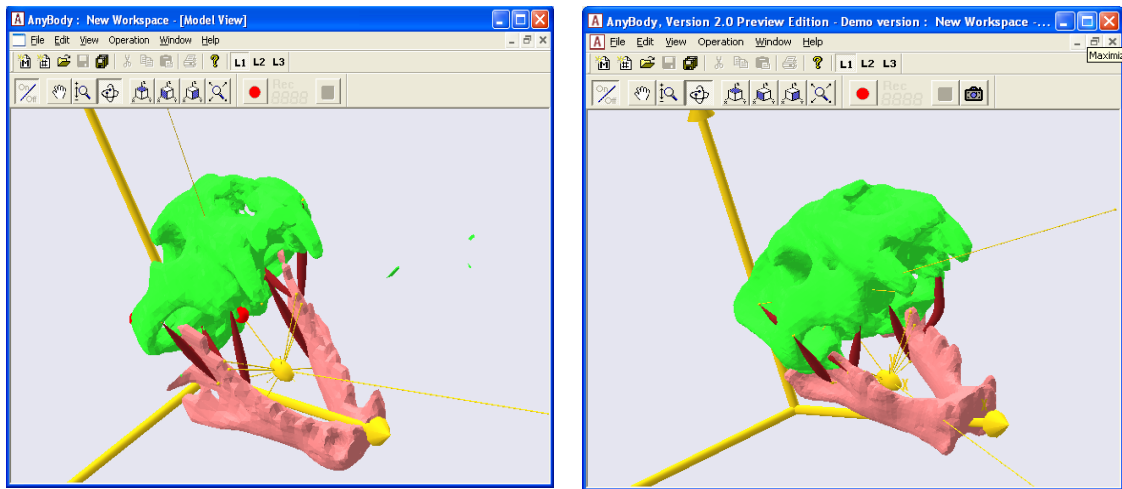
Figure 3-9 (a) Skull and Mandible of tiger, (b) Skull and Mandible of Sabertooth tiger

Once the segments are defined, we need to create joints to constrain the motions of these segments. Two revolute joints are created for both upper and lower jaws so that each can move independent of other while doing the biting action. The joint connecting the skull and mandible can be approximated to the motion of a hinge joint as there is very little lateral movement between the jaws [25]. The position vectors that measure the distance of the muscle attachment points to the tip of the tooth can be obtained from the kinematic measures that are defined in the model. These position vectors are later useful in calculating the bite force.

In order to provide constrained motion(s) for our musculoskeletal model, we need to define drivers that move the joints. We defined two drivers that are attached to the two revolute joints. Using these drivers we can independently control the velocity and acceleration jaw movements. The starting position of simulation can be set by changing the initial conditions of the driver (position, velocity, etc.).

Once the segments, joints and drivers are defined, we need to attach muscles. In real musculoskeletal systems muscles serve the function of actuators. They are activated by the Central Nervous System (CNS) through an electro-chemical process. AnyBody mimics the computing of CNS by back calculating the muscle forces from the movement and load specifications by a process called inverse dynamics. The drivers added in the model will provide the necessary forces and moments to make the body motions but in a real musculoskeletal system muscles are responsible for creating forces and moments. In order to depict reality in to our modeling procedure i.e. to make muscles as the active supplier of motion and make the drivers passive we have to set one of the driver properties (Reaction.Type) to zero. This will ensure that all the forces and moments that are necessary to perform body motions are provided by the muscles and not by drivers.

Before attaching muscles, muscle models are defined with all the necessary characteristics. Once the model is built, we will attach muscles between the nodes defined in upper and lower jaw. A muscle is generally attached between the origin and the insertion. The most proximal point from the mid line is the origin and the most distal point will be the insertion. Depending on the location of the muscle attachment points we define the origin and insertion either on the upper or the lower jaw. The final musculoskeletal model will be as shown in Figure 3-10.



a

b

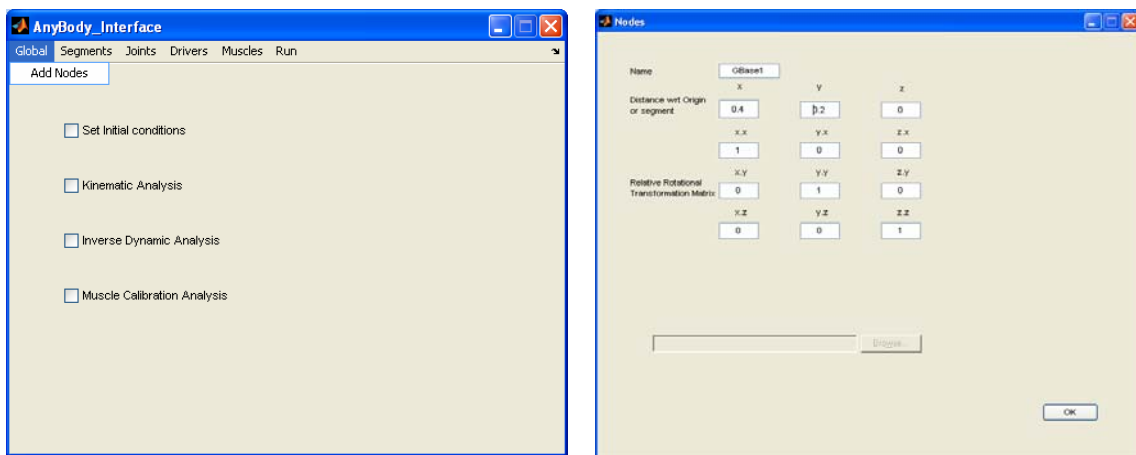
Figure 3-10 (a) Musculoskeletal model of tiger, (b) Musculoskeletal model of Sabertooth tiger

Analysis can be done on the model thus built. Any model build in AnyBody is a script based model. Such a script based method poses a number of restrictions on the user pertaining to the knowledge of ‘AnyScript’ language, its syntax notation etc. Since musculoskeletal models are generally developed by people who come from biological background, whom requires little or no programming knowledge, a Graphical User Interface (GUI) would be more useful. Hence we have developed a GUI in MATLAB (the details of which are described below) using command line interface feature. The command line interface provides the user with very limited options to run the model. We have utilized this feature to its best with the help of MATLAB.

3.5 MATLAB Interface

The MATLAB interface will provide the user with all the basic functionalities that are required to build a musculoskeletal model. Modeling such a system with MATLAB follows the same process as in AnyBody. The greatest advantage is providing the user with friendly windows that will guide through in defining the multibody musculoskeletal model with all the constraints. The steps in building and analyzing the musculoskeletal model are

As stated earlier, building the musculoskeletal model starts with defining the global fixed frame with all the fixed points defined. The user has the flexibility to always move back and forth through out the modeling procedure in defining the fixed points. By selecting Global \rightarrow Add Nodes, as shown in Figure 3-11(a), we get window shown in Figure 3-11(b), where properties such as position and orientation of these fixed points with respect to the global fixed frame are defined.



a b
Figure 3-11 (a) Global frame selection window, (b) Node properties in Global frame

Segments: The segment properties window is shown in Figure 3-12 (a). We define the segment name, the position vector i.e. the position of the center of mass of the segment with respect to the global fixed frame, the mass of the segment and finally the orientation of the segment. Once these properties are defined, the user has the choice of defining the nodes or points of interest as shown in Figure 3-12(a) that can be used later to constrain the system. Apart from the position and orientation information, we can also specify surface information to each segment via STL files as shown in Figure 3-12(b). STL files are helpful in preserving the geometry of the system which is useful for muscle attachment and routing. For example, defining the surface at the point of interest will ensure that the muscles attached later will wrap around the bone instead of penetrating through the bone. The user can define any number of nodes in a segment or any number of segments in the system. One of the limitations to the present version of the interface is that once the segment is defined (along with nodes) and closed, it can not be re-edited and needs to be addressed in future versions.

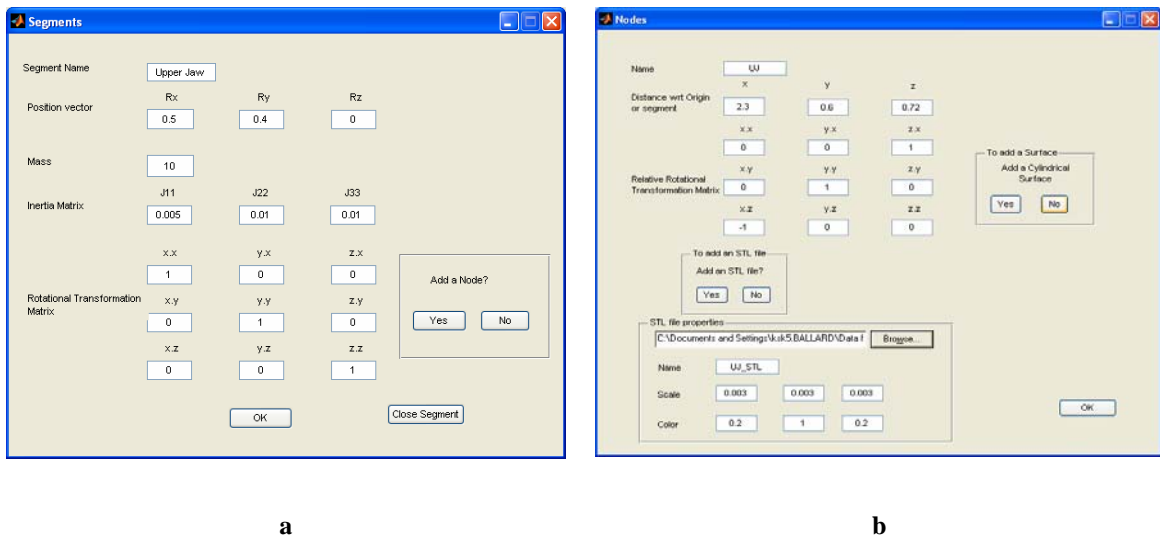


Figure 3-12 (a) Segment properties, (b) Segment Node properties

Joints: Joints are mainly used to constrain the motions of the segments. As shown in Figure 3-13 we can attach 6 different types of joints (Standard, Spherical, Cylindrical, Revolute, Prismatic and Universal Joint) to our system. In our present analysis, there is very little or approximately no lateral movement between the skull and mandible. So we have used revolute joint in our analysis. The other joint types will be useful when we extend our model to incorporate other parts of the body. A joint is generally attached to two reference frames, 'Reference Frame 1' and 'Reference Frame2'. These reference frames are the nodes that are defined in the global fixed frame and the segments.

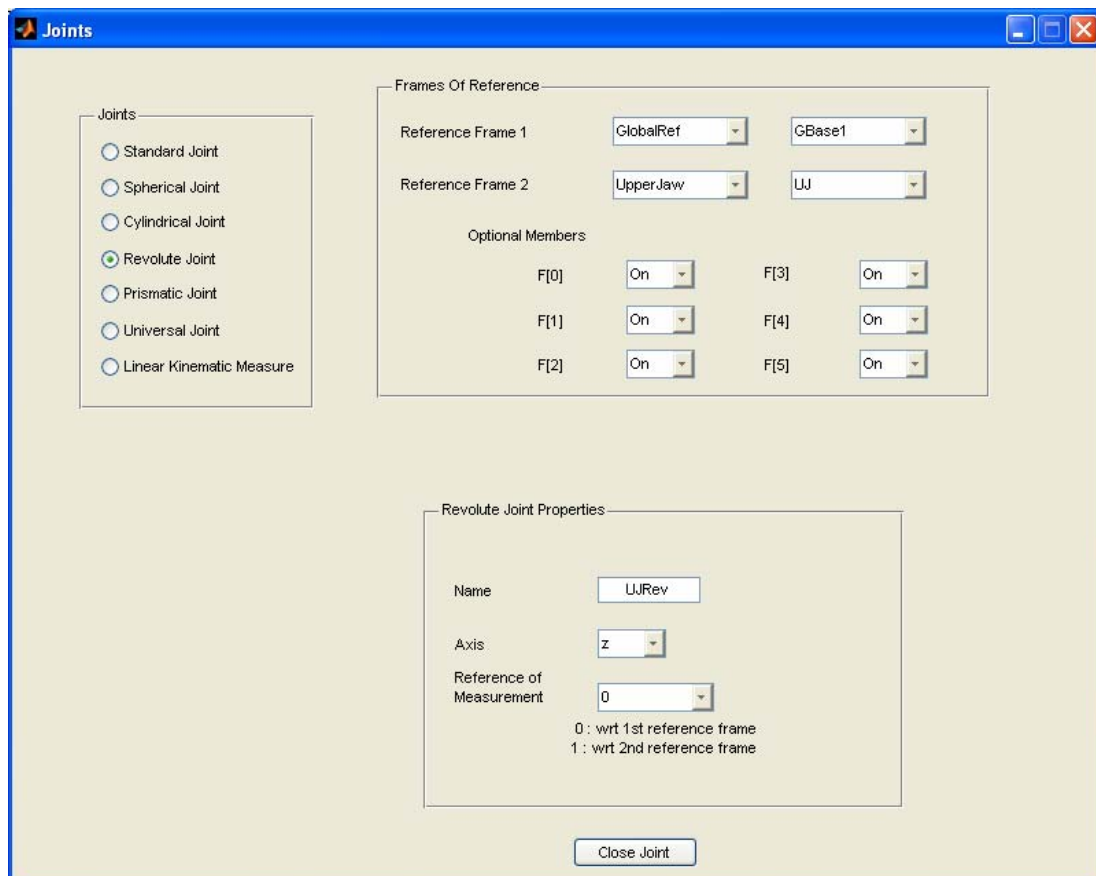


Figure 3-13 Joint selection window with its corresponding properties

Various kinematic measures, canalso be easily created. For example, Figure 3-14 depicts the specification of the linear kinematic measure. In Figure 3-13, ‘Reference Frame 1’ and ‘Reference Frame 2’ represent the two points in space between which the distance is measured. These measured distances will be helpful in the calculating the bite force.

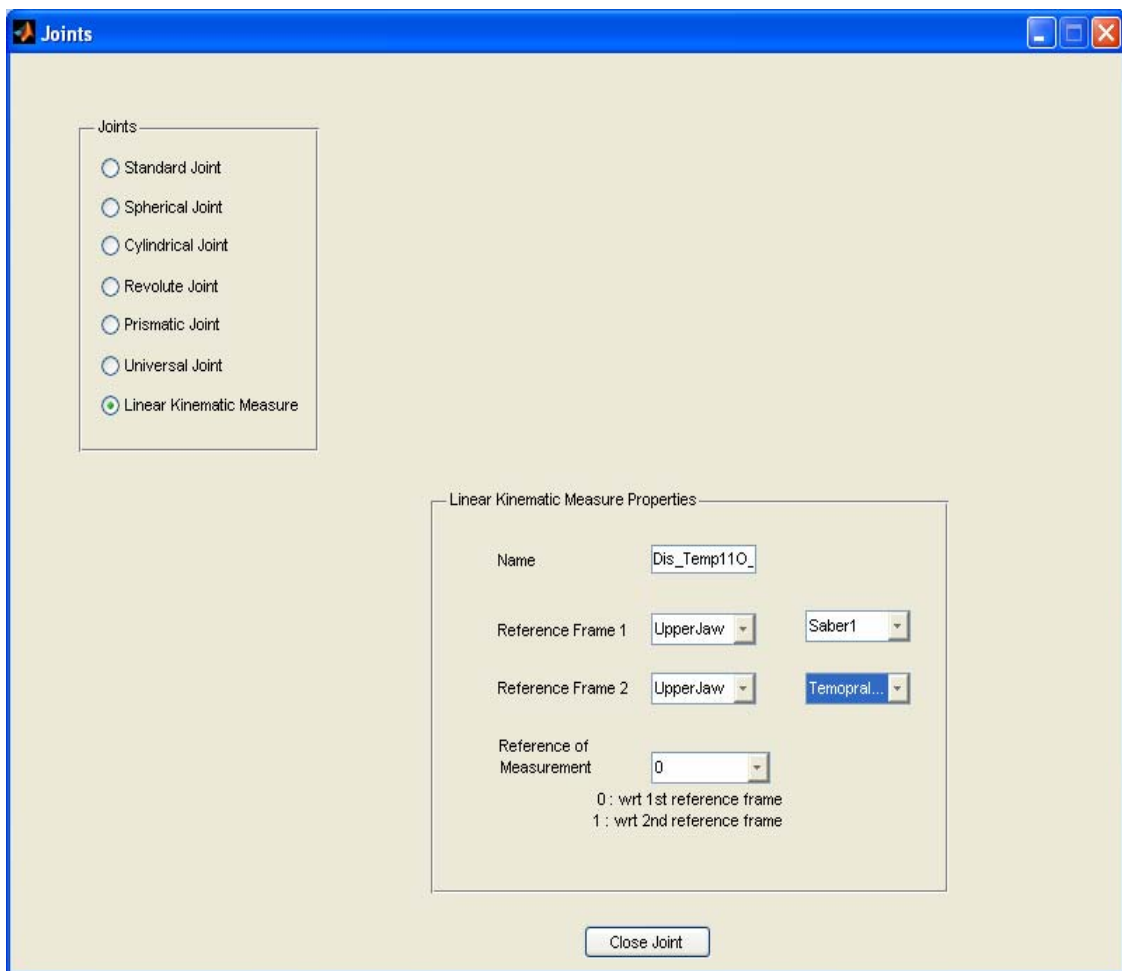


Figure 3-14 Kinematic Measure properties for measuring distance in 3D

Drivers: Drivers are attached to the joints to move the musculoskeletal system. As shown in Figure 3-15, we have two different types of drivers. ‘Simple Kinematic Driver’, a

constant acceleration driver, and 'Interpolation Driver', prescribed motion driver. The prescribed motion can be defined with the help of data points in the form of ASCII file.

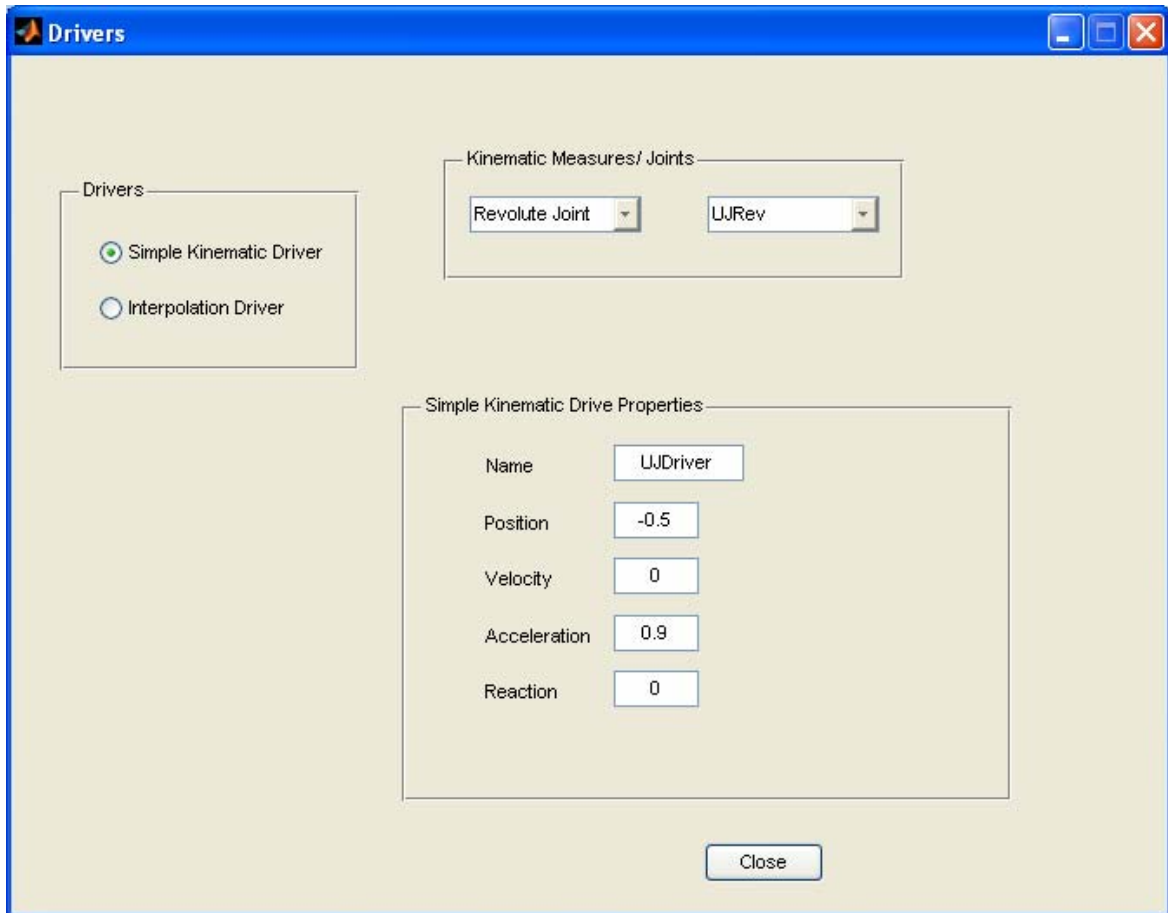
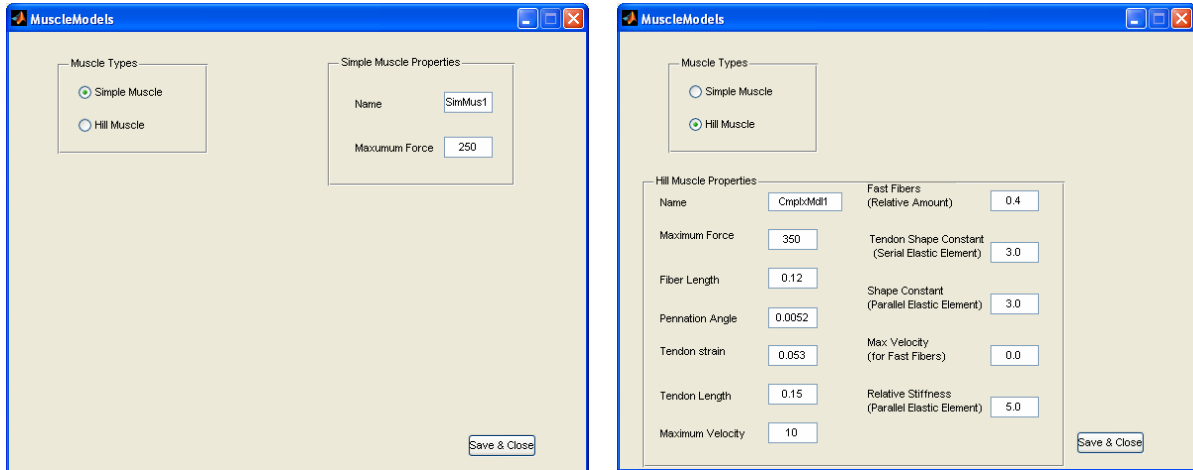


Figure 3-15 Driver selection window and its corresponding properties

Muscle models and Inserting Muscles: In order to attach muscles between two points we need to define the characteristics of the muscle first. We have two different types of muscles. Simple Muscle, as shown in Figure 3-16(a), can be used for initial stages of modeling or in case where the muscle properties are not known exactly. Hill Muscle model, as shown in Figure 3-16(b), a more complex muscle is based on the model

developed by Hill with most of the muscle properties such as pennation angle, fiber length, maximum muscle force, tendon strain, etc. This muscle model is used when more accurate results are required and moreover if we know the behavior of the muscle.

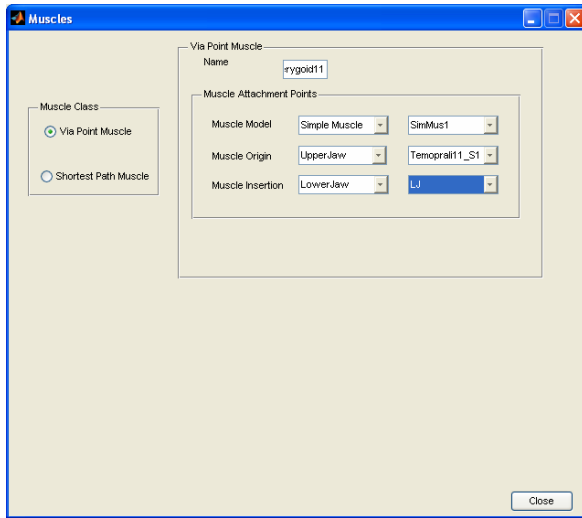


a

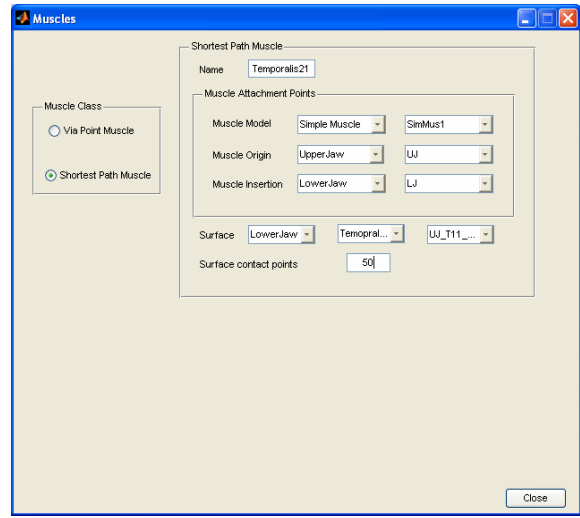
b

Figure 3-16 (a) Simple muscle model, (b) Hill muscle model

While attaching muscles we can use two methods: (i) ‘Via Point Muscle’, Figure 3-17(a), joins the muscle as a string from the origin to insertion; and (ii) ‘Shortest Path Muscle’, Figure 3-17(b) is more sophisticated one that can wrap around the obstacles, which are indicated in the form of surfaces (in the segments)



a



b

Figure 3-17 (a) Via point muscle class properties, (b) Shortest path muscle class properties

Analysis: After completely defining the musculoskeletal system, we can run various analyses, as shown in Figure 3-18. Steps from 1 to 6 can be used to build and perform analysis of any musculoskeletal system.

Using the MATLAB interface we have performed experiments pertaining to three tasks:

- Calculation of Muscle forces
- Calculation of Bite forces
- Performing an optimization routine to approximately determine the muscle origin and insertion.

The theory and application part of these tasks is described in chapter 4.

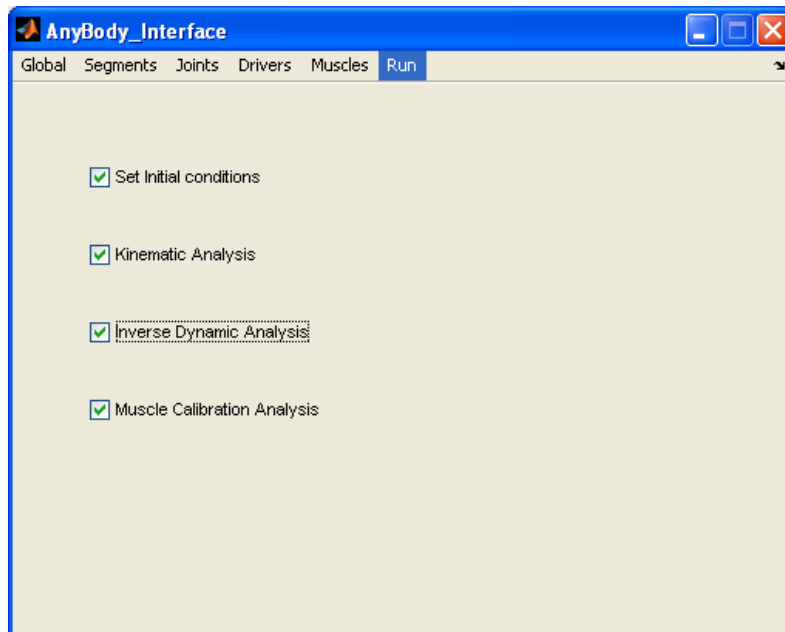


Figure 3-18 Running the analysis

4 Optimization And Bite Force Calculation

In this chapter we will provide the reader with mathematical background on calculating muscle forces, bite forces and the optimization routine used to determine the optimal muscle origin and Insertion locations.. Finally we will present some results pertaining to muscle forces and bite forces which will comprehend the validity of our systematic analysis framework as described in the previous chapters.

4.1 Background

4.1.1 Forward Dynamics

In a forward dynamic problem we obtain the motion of the multi-body system over a given time interval given the forces and the initial condition of the problem. The Forward dynamics problem is also called the “simulation problem” since it allows for the simulation of the system and thus prediction of system behavior.

4.1.2 Inverse Dynamics

The inverse dynamics aims at solving for the joint variables, motor or driving forces as well as the reaction forces that appear at each of the multi-body system that result in the specified motion of the system. The inverse dynamics problem can also be considered as a control problem, since it permits the solution for muscle forces and joint reactions and this will be the focus of our work.

For the inverse dynamic analysis we assume that the bone or segments to be rigid bodies connected together by joints. Assuming a uniform mass distribution and a

corresponding density, a preliminary estimate of mass and inertias of these bones is obtained from the geometry. Additionally, we need to take care of various forces in the system (such as muscle forces, joint constraint forces, gravitational and inertia forces and external forces, if any) while performing the analysis. To this end, position, velocity and acceleration of various points of interest (such as the tip of the tooth, a joint etc.) are required and can be calculated given the position time history of the system together with the kinematic structure of the skeleton.

In the inverse dynamics problem, one seeks to resolve the forces $F_1(t)$ and $F_2(t)$ that are necessary in order to realize a given motion $x(t)$. As we see clearly, we have indeterminacy in the form of fewer equations than the unknowns.

4.1.3 Redundancy Resolution

The equations of motion obtained in the dynamics stage can in general be represented in matrix form as

$$C_{m \times n} f_{n \times 1} = d_{m \times 1} \quad (4.1.1)$$

In the equation(4.1.1), C represents a coefficient matrix that depends on the current position of the body segments, f are the unknown forces or muscle forces. In musculoskeletal systems, generally, the number of muscles in the system will be greater than the degrees of freedom of the model, $m < n$, which implies that the number of equations is less than the number of unknowns. This is best illustrated by this example of a mass that is constrained to translate only in the x-direction under the action of two forces F_1 and F_2 . The governing equations of motion can be written in terms of the unknown F_1 and F_2 as:

$$mg\ddot{x} = F_1 + F_2 \quad (4.1.2)$$

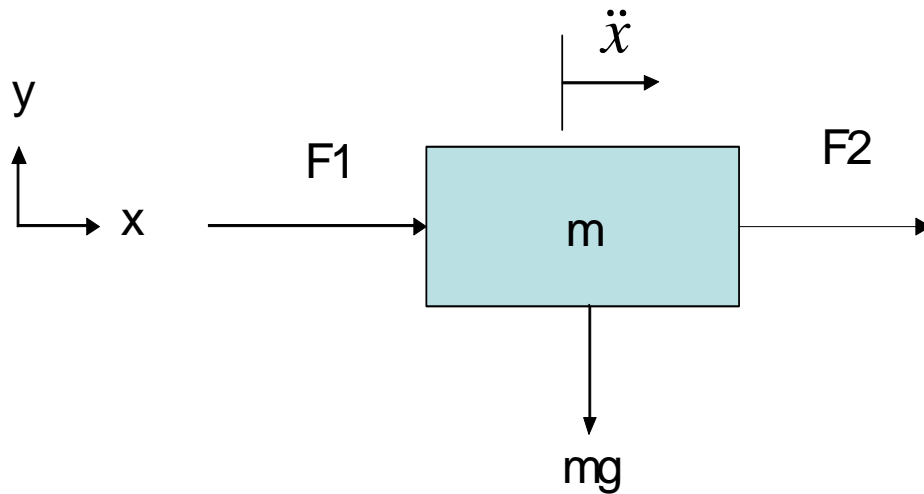


Figure 4-1 Mass being acted by two forces showing the problem of redundancy

A number of solutions have been suggested in past in order to solve this mathematical indeterminacy. In the *reduction method*, the number of unknowns is reduced by grouping them together until they match the number of degrees of freedom. In the *addition method*, additional number of constraints is introduced into the system so the system becomes mathematically determinate. In both the methods, the results depend on the designer's choice of grouping of unknowns or addition of constraints which are highly variable.

Collins [26] and Lu [27] used *Dynamically Determinate One –Sided Constrained* (DDOSC) method for solving redundancy. The indeterminate problem is divided in to a series of dynamically determinate problems. The solutions thus obtained are accepted or rejected based on the system and biological constraints. However, the effort involved in solving such systems increases exponentially as the complexity of the system increases.

It is in this arena that optimization is an indispensable tool in biomechanics to solve the mathematical indeterminacy. Different optimization methods such as gradient-

based, parameter optimization algorithm, variational approach [28], modified Polak-Mayne algorithm [29], stochastic optimal feedback control [30], and Fourier-based methods [31] were used in the literature depending on the static-dynamic or linear-nonlinear nature of the problem [32-34].

4.2 Problem A: Determining Muscle Forces

Various hypotheses have been proposed pertaining to how the CNS performs muscle recruitment for motor control. One of the more popular hypotheses gathering increasingly experimental evidence is Weber's hypothesis, which states that "muscle recruitment is performed in such a way that the *muscular activity* is minimized during routine activities". The "muscular activity" may be modeled as the ratio of the individual muscle force, $F_{M,i}$, and the normalization factor (for peak muscle force or PCSA), N_i . Hence in an effort to implement this mathematically the optimization objective may be stated as:

$$\text{Minimize } V = \sum_{i=1}^n \left(\frac{F_{M,i}}{N_i} \right)^p$$

Subject to:

$C_f = d$ and

$$F_{M,i} \geq 0 \text{ for } i = 1, \dots, n$$

In words, the above problem can be stated as minimizing the summed muscle activity (V) subject to constraints on the muscle forces placed by the structure as well as the actuator limits. Various forms of this optimization problem may be created raising the power of the individual muscle activity to a polynomial power, 'p'. With increasing value of 'p', the criteria tends to distribute the relative load evenly between the muscles [35].

The min/max objective function is non-differentiable and therefore appears to complicate the practical solution of the optimization problem. However, by using bound formulation which is widely used and well-tested in the field of optimum engineering design we can easily solve the min/max problems. By introducing a new artificial variable, β , and an artificial criterion function, $B(\beta)$ the new criterion can be a monotonic function of β . By choosing the $B(\beta) = \beta$, we can reformulate the above problem as :

$$\begin{aligned}
 & \text{Minimize } \beta \\
 & \text{Subject to:} \\
 & \underline{C} \underline{f} = \underline{d} \\
 & \frac{F_{M,i}}{N_i} \leq \beta, i \in \{1, \dots, n\} \\
 & F_{M,i} \geq 0, i \in \{1, \dots, n\}
 \end{aligned} \tag{4.1.3}$$

In the min/max optimization problem we are looking at the muscle recruitment that balances the exterior loads and minimizes the largest relative load on any muscle in the system, thereby postponing fatigue of the muscle as far as possible [20]. This is the approach used with in the AnyBody software which we will also employ.

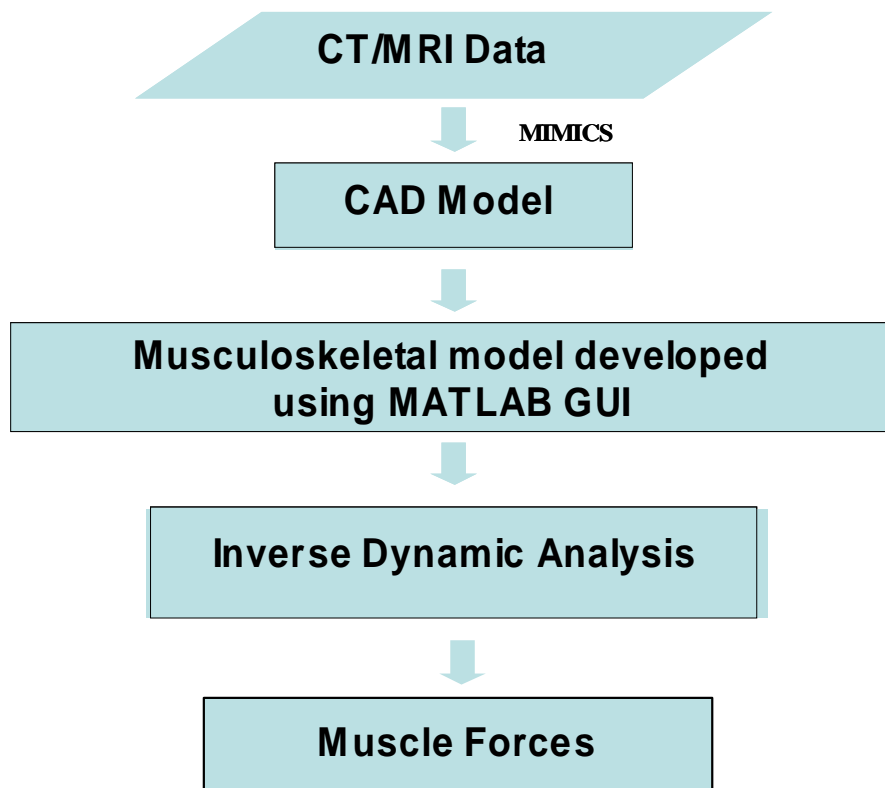


Figure 4-2 Process flow chart for calculating muscle forces

4.2.1 Our Model:

In the inverse dynamic analysis the jaw is allowed to bite through the prey at a specified rate. Currently, this value is chosen based on qualitative empirical observation. However, in the future, qualitative motion capture studies can be used to provide this information. The muscle forces obtained through IDA are very much dependent on (i) the actual specimen geometry as well as the designer's choice of certain variables which include (ii) muscle route from origin to insertion, and (iii) type of muscle. We will

perform a series of studies to examine the effects of these variables on muscle forces in the next few subsections. In each of these cases, we will present two sets of plots.

- Plot1 compares muscle activity of each muscle through out the simulation. Traditionally, since the muscles need not get activated at the same instant, there can be an ambiguity in determining the activation pattern. However, the min/max optimization criteria make sure that only the muscle recruitment pattern that can minimize the maximum muscle activity is computed. Thus it results in a unique solution that is also able to give us information about the firing order of muscles.
- Plot 2 shows the individual muscle forces for the Temporalis, Masseter & Pterygoid for the left and right hand sides. This was done since the muscle activity plots in plot1 represent normalized values and may not readily give a clear idea of the actual muscle forces involved. Both sides are shown so that cases of asymmetric activation can also be detected.

The studies are intended to span the various choices that are available to a biologist. First we have a choice among the feline family members where we employed Tiger & Smilodon. Second, we have a choice in terms of selection of muscle routing algorithm between ‘ViaPointMuscle’ and ‘ShortestPathMuscle’. Third, a biologist would also have a choice for use of individual muscle models between simple & complex.

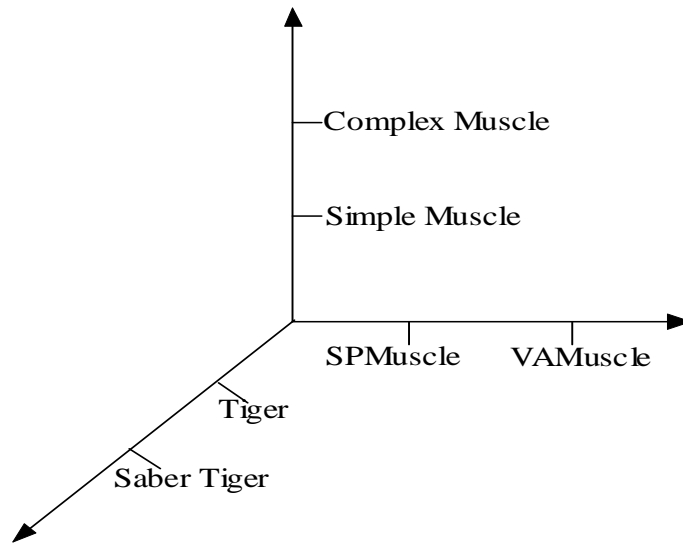


Figure 4-3 3-dimensional representation of case studies

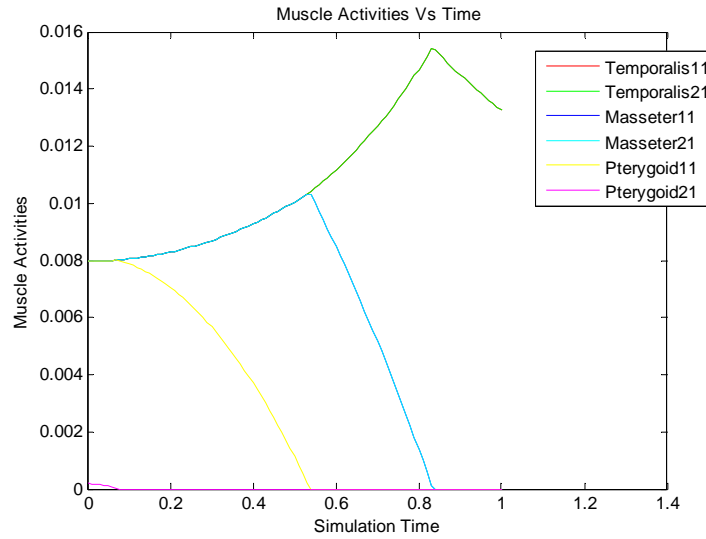
These choices are reasonably independent of each other and can be represented as the three coordinate axes as shown in the Figure 4-3. Thus by taking various combinations of these choices, eight case studies may be generated as shown in table4-1.

Case	Feline member	Routing algorithm	Muscle Type
Case1	Tiger	VPMuscle	Simple
Case2	Tiger	VPMuscle	Complex
Case3	Tiger	SPMuscle	Simple
Case4	Tiger	SPMuscle	Complex
Case5	Saber Tiger	VPMuscle	Simple
Case6	Saber Tiger	VPMuscle	Complex
Case7	Saber Tiger	SPMuscle	Simple
Case8	Saber Tiger	SPMuscle	Complex

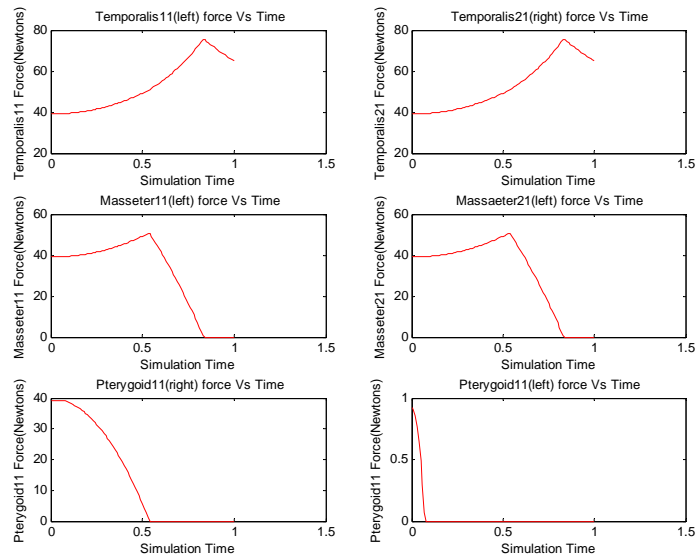
Table 4-1 : Different case studies resulting from (i) the actual specimen geometry (ii) muscle route from origin to insertion, and (iii) type of muscle

The two plots for each case are shown in the following pages

4.2.2 Case 1: Tiger – VPMuscle – Simple



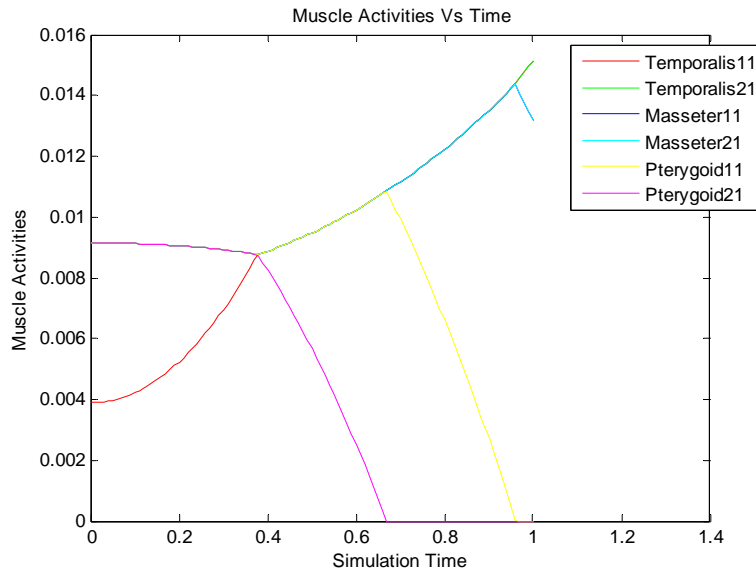
a



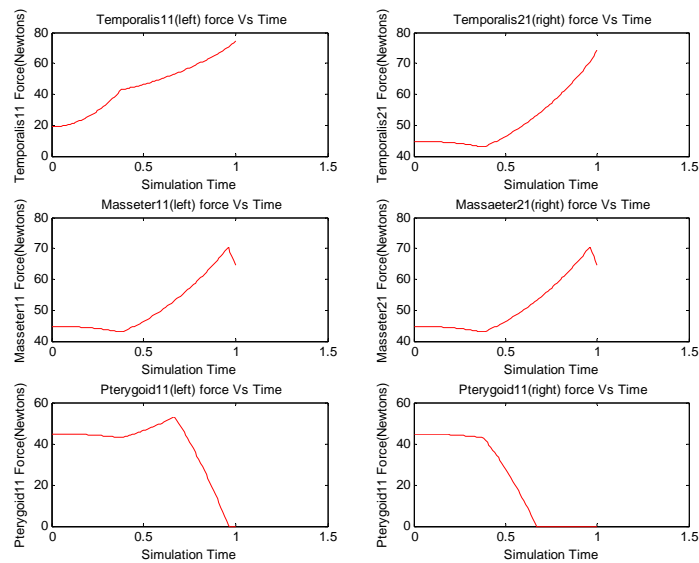
b

Figure 4-4 (a) Plot 1 :Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – VPMuscle – Simple (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – VPMuscle - Simple

4.2.3 Case 3: Tiger – SPMuscle - Simple



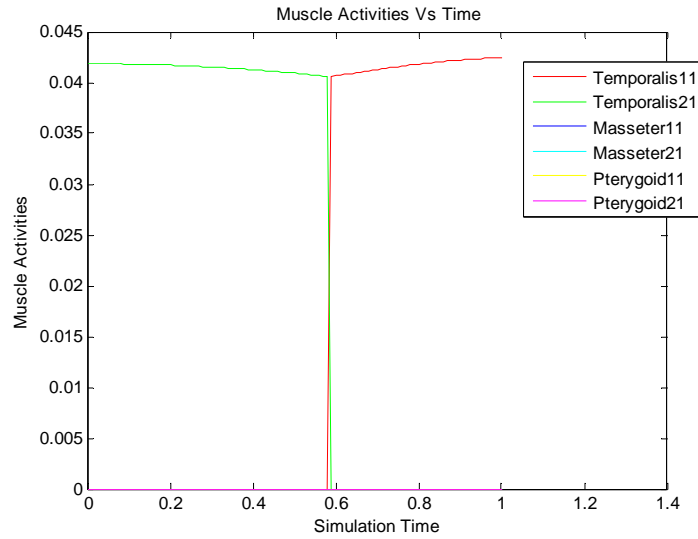
a



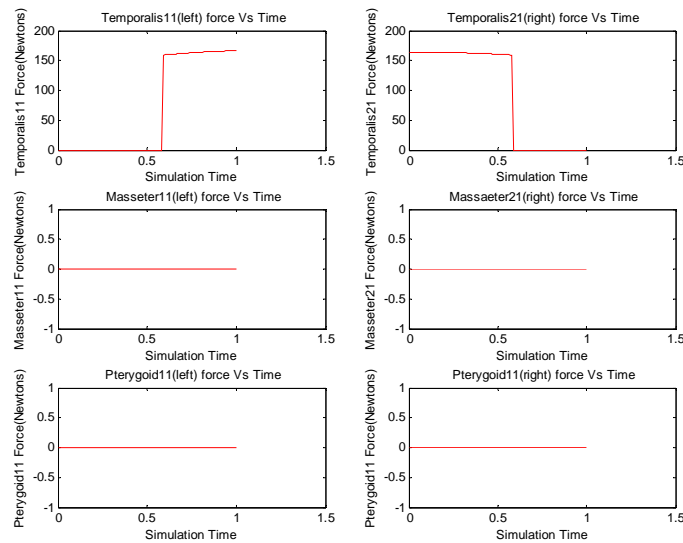
b

Figure 4-5 (a) Plot1: Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – SPMuscle - Simple (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – SPMuscle - Simple

4.2.4 Case 5: Sabertooth tiger - VPMuscle - Simple



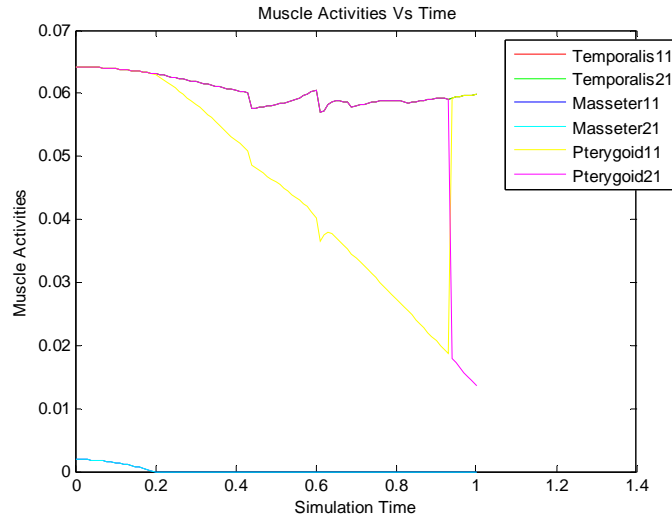
a



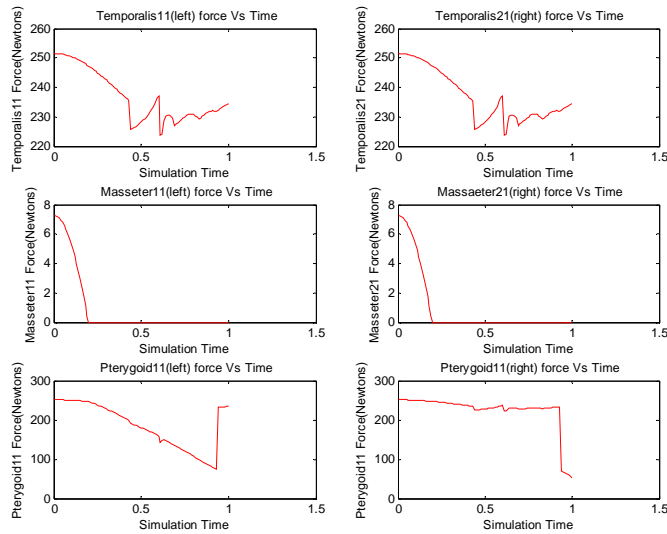
b

Figure 4-6 (a) Plot 1: Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left , right hand sides) for Sabertooth tiger – VPMuscle – Simple (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Sabertooth tiger – VPMuscle - Simple

4.2.5 Case 7 : Sabertooth tiger - SPMuscle - Simple



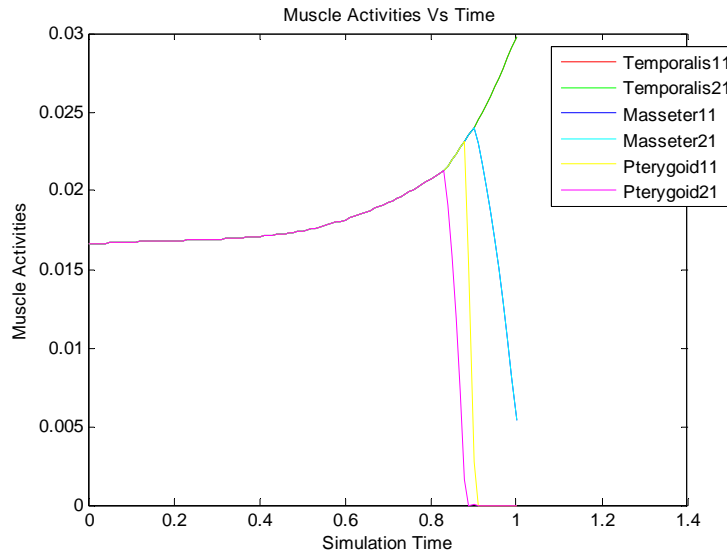
a



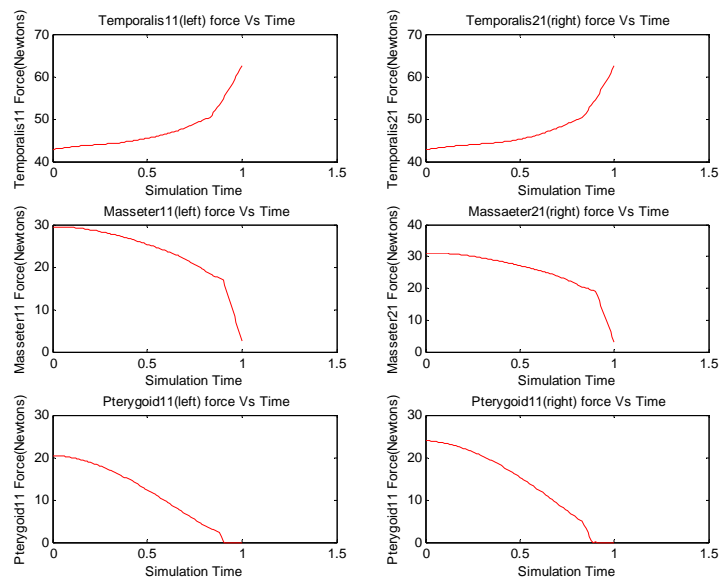
b

Figure 4-7 (a) Plot 1: Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Sabertooth tiger – SPMuscle – Simple (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Sabertooth tiger – SPMuscle - Simple

4.2.6 Case 2: Tiger – VPMuscle - Complex



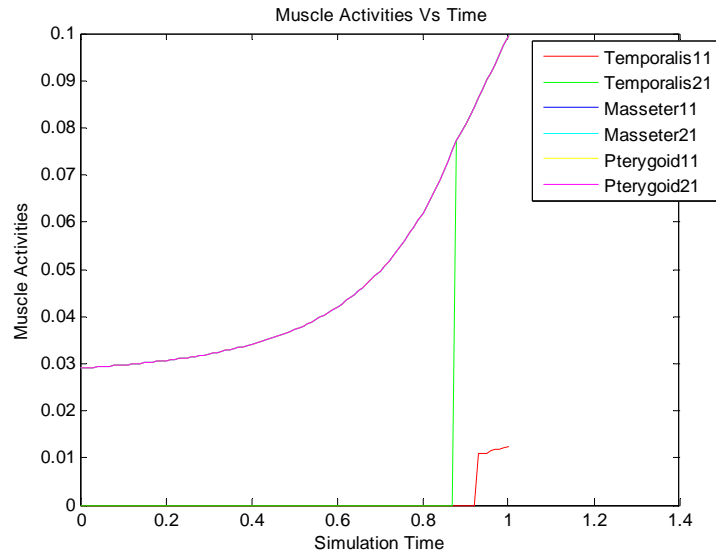
a



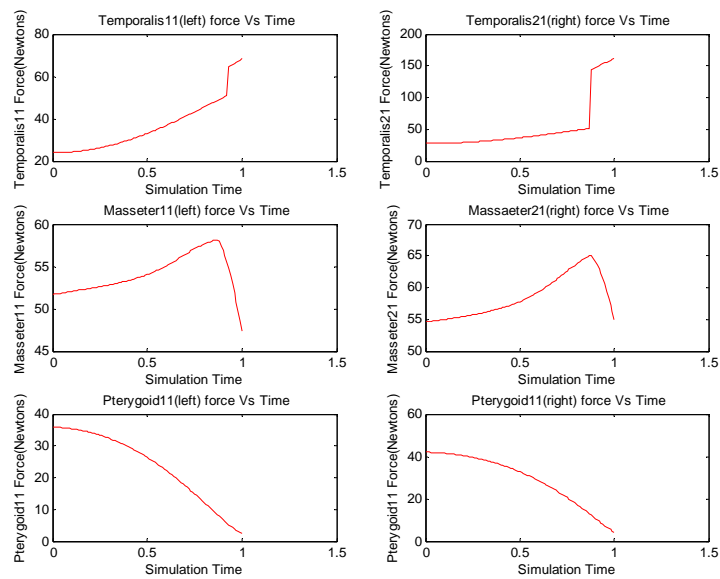
b

Figure 4-8 (a) Plot 1: Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – VPMuscle – Complex (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – VPMuscle - Complex

4.2.7 Case 4: Tiger – SPMuscle - Complex



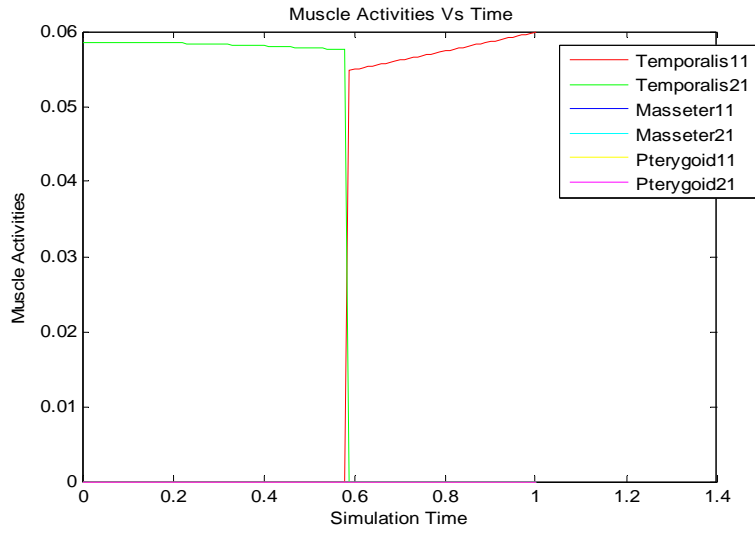
a



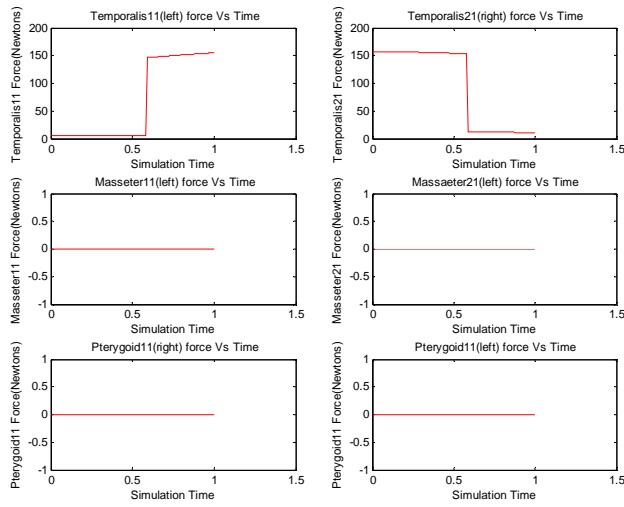
b

Figure 4-9 (a) Plot 1: Muscle Activities of Temporalis, Masseter and Pterygoid muscles (left, right hands sides) for Tiger – SPMuscle – Complex (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Tiger – SPMuscle - Complex

4.2.8 Case 6: Sabertooth tiger – VPMuscle – Complex



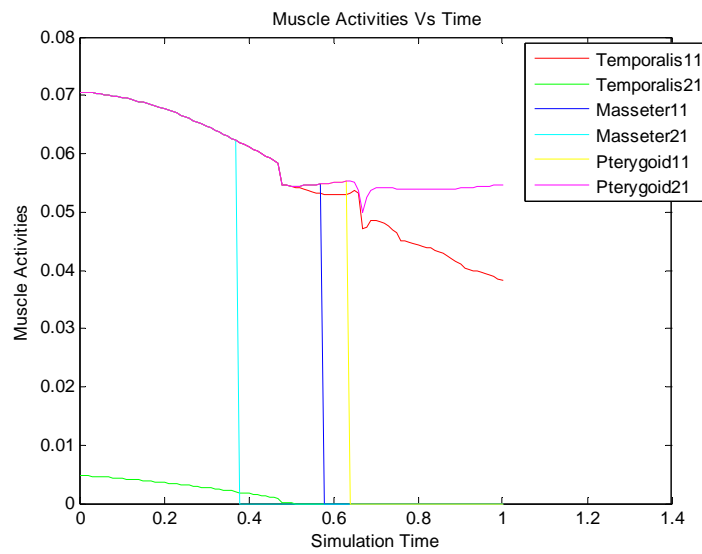
a



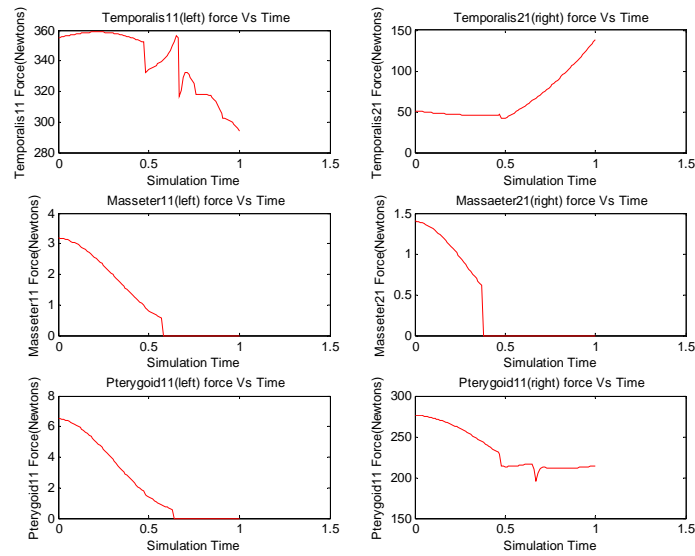
b

Figure 4-10 (a)Plot 1: Muscle Activities of Temporalis, Masseter , Pterygoid muscles (left, right hand sides) for Sabertooth tiger–VPMuscle–Complex (b)Plot2: Individual muscle forces of Temporalis, Masseter, Pterygoid muscles (left, right hand sides) for Sabertooth tiger–VPMuscle-Complex

4.2.9 Case 8: Sabertooth tiger – SPMuscle - Complex



a



b

Figure 4-11 (b) Plot 1: Muscle Activities of Temporalis, Masseter, Pterygoid muscles (left,right hand sides) for Sabertooth tiger – VPMuscle – Complex (b) Plot2: Individual muscle forces of Temporalis, Masseter and Pterygoid muscles (left, right hand sides) for Sabertooth tiger – VPMuscle - Complex

From Cases 1 and 3 , we observe that SPMuscle and VPMuscle, the muscle routing methods, have considerable effect on the forces muscles can produce. Hence the muscle route has a considerable effect on the bite force.

Complex muscle models are based on the Hill muscle model, which is more sophisticated than the simple muscle model. It captures more realistic muscle properties. Hence the muscle forces in these models are more close to reality

4.3 Problem B: Bite Force Calculation

As mentioned in Chapter 2, Bite force has been used by many people in order to study the functional performance and behavioral analysis of animals. Bite force can be calculated from the following equilibrium equation.

$$[F_e] = [\hat{\$}][F_M] \quad (4.1.4)$$

Where,

$\hat{\$}$ is the line vector matrix that depends on the muscles line of action

F_M is the column vector representing the muscle force

F_e is the external force or Bite force

Calculating $\hat{\$}$:

In 3-D each muscle can be modeled as a bound vector with known origin and insertion points. The line of action of force for a muscle can be obtained from the homogeneous coordinates developed by Plucker or generally referred as Plucker coordinates. If $O_1 = (x_1, y_1, z_1, w_1)$ and $I_1 = (x_2, y_2, z_2, w_2)$ represent the origin and insertion of a muscle respectively, then Plucker coordinates represented by $[L, M, N, P, Q, R]^T$, which is a [6x1]vector, can be obtained from the determinants of the 2x2 submatrices of the so called Grassmanian matrix shown in equation (4.1.5)

$$\begin{bmatrix} w_1 & x_1 & y_1 & z_1 \\ w_2 & x_2 & y_2 & z_2 \end{bmatrix} \quad (4.1.5)$$

Hence,

$$L = \begin{bmatrix} w_1 & x_1 \\ w_2 & x_2 \end{bmatrix} \quad M = \begin{bmatrix} w_1 & y_1 \\ w_2 & y_2 \end{bmatrix} \quad N = \begin{bmatrix} w_1 & z_1 \\ w_2 & z_2 \end{bmatrix}$$

$$P = \begin{bmatrix} y_1 & z_1 \\ y_2 & z_2 \end{bmatrix} \quad Q = \begin{bmatrix} z_1 & x_1 \\ z_2 & x_2 \end{bmatrix} \quad R = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \end{bmatrix}$$

$[L, M, N, P, Q, R]^T$ form the column vector of $\hat{\$}$

With 'n' muscles, $\hat{\$}$ is a [6xn] vector

Given F_M and $\hat{\$}$, the bite force at the tip of the tooth can be calculated from equation (4.1.4).

In a musculoskeletal system the location of the muscle that affects the line of action of force has a significant effect on the amount of force it can produce during the motion. Hence, in calculating the bite force of any animal it becomes necessary to accurately determine the location of muscle origin/insertion.

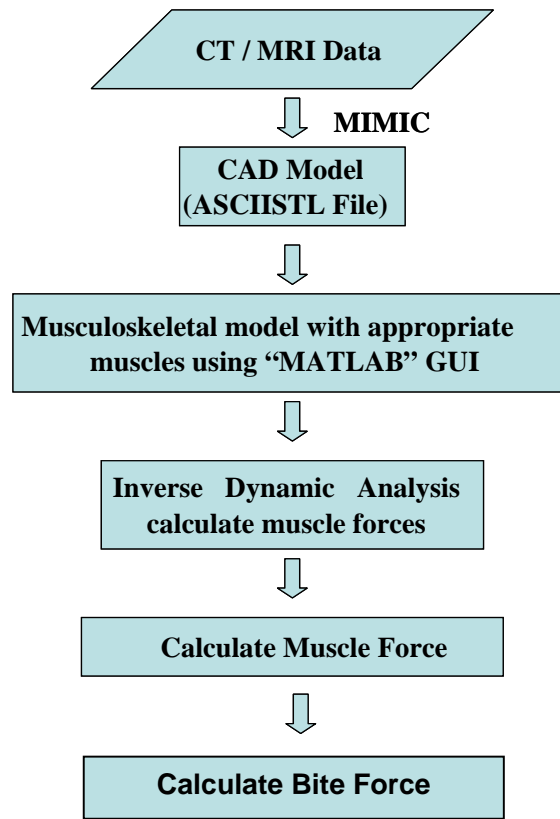


Figure 4-12 Process flow chart for calculating bite force

4.4 Problem C: Finding Optimal Muscle Attachment Location

As mentioned above, in a musculoskeletal system the location of the muscle, which in turn affects the line of action of force, has a significant effect on the amount of force it can produce during the body motion. From the data available for saber tooth tiger, we can only determine a relatively large region for muscle origin/insertion based upon morphometry. Hence a tool that can help a biologist to optimally search within these boundaries and narrow the possibility of muscle location would be extremely helpful in better evaluation of functional performance. For example, by choosing bite force as the

objective function (as it has been used earlier for functional performance), we can now determine the muscle origin and insertion points within the limits that can produce maximum bite force during the biting action.

The complete representation of project can be viewed in flowchart form as shown in Figure 4-13

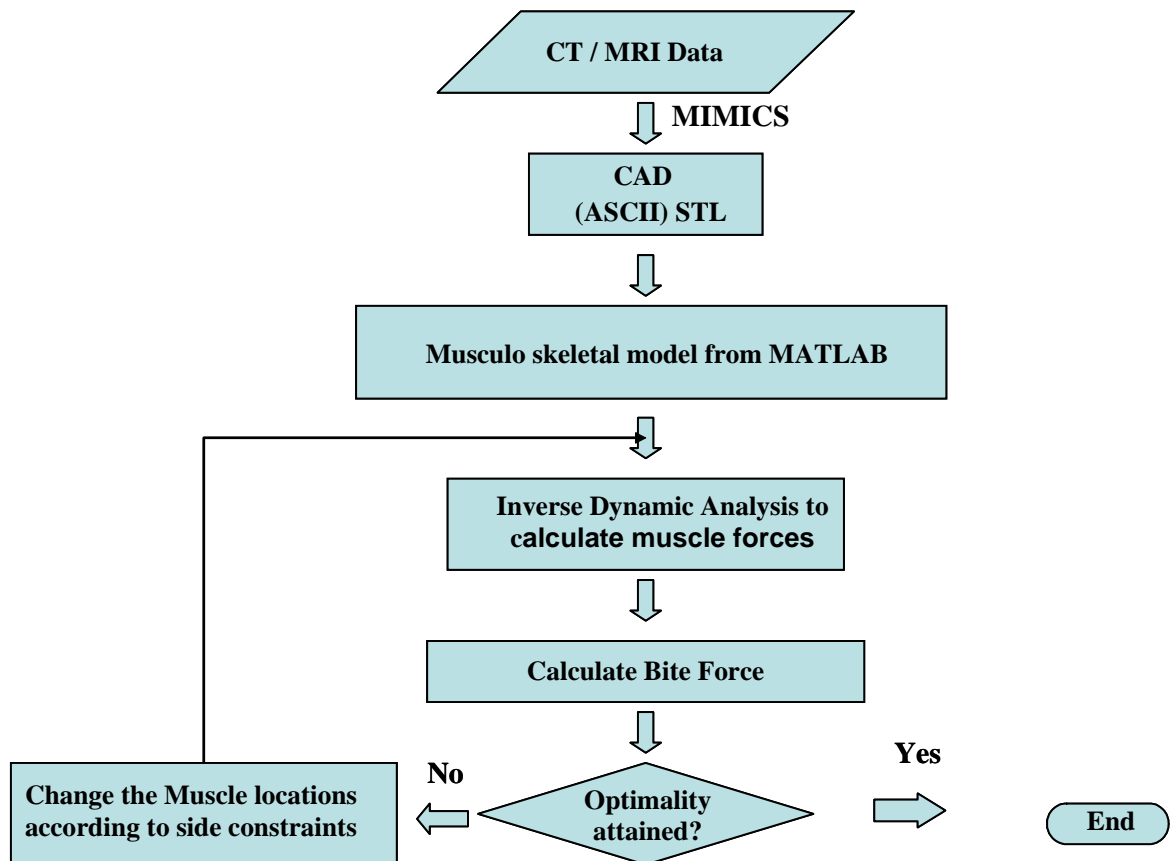


Figure 4-13 Process flow chart for creating musculoskeletal model from point data

The optimality criteria for determining the optimal muscle location can be represented as

Minimize: –Bite force (X)

Subject to:

$Cf=d$

$$\frac{F_{M,i}}{N_i} \leq \beta, i \in \{1, \dots, n\} \quad (4.1.6)$$

$$F_{M,i} \geq 0, i \in \{1, \dots, n\}$$

$$X_L \leq X \leq X_U$$

Where,

X -> Vector representing design variables i.e. muscle origin and insertion coordinates

X_L and X_U represent the lower and upper limits on the design variable vector.

Before examining the results of the full optimization it is instructive to create various such problems using subsets of the design variables. These smaller optimization problems are often easier to solve and additionally allow a systematic study of the design variable space through parametric sweeps. The results of such parametric sweeps performed on origin/insertion of the three muscles are shown in next few figures.

Design Variables	Number
X,Z positions of Temporalis Origin	2
X,Y positions of Temporalis Insertion	2
X,Z positions of Masseter Origin	2
X,Y positions of Masseter Insertion	2
X,Y positions of Pterygoid Origin	2
X,Z positions of Pterygoid Insertion	2
Table 4-2 Design variables for calculating optimal muscle locations	

Table 4-2 shows the six subsets of design variables used in the parametric study while the rest of the design variables are held constant.

4.4.1 Parametric Sweep Studies

Figure 4-14 to Figure 4-19 indicate the variation of bite force with changing muscle origin and insertion for the case 1 i.e. Tiger – VPMuscle- Simple. Brute force method is employed to span the entire probable area of muscle location(s)

4.4.1.1 Tiger VPMuscle Simple

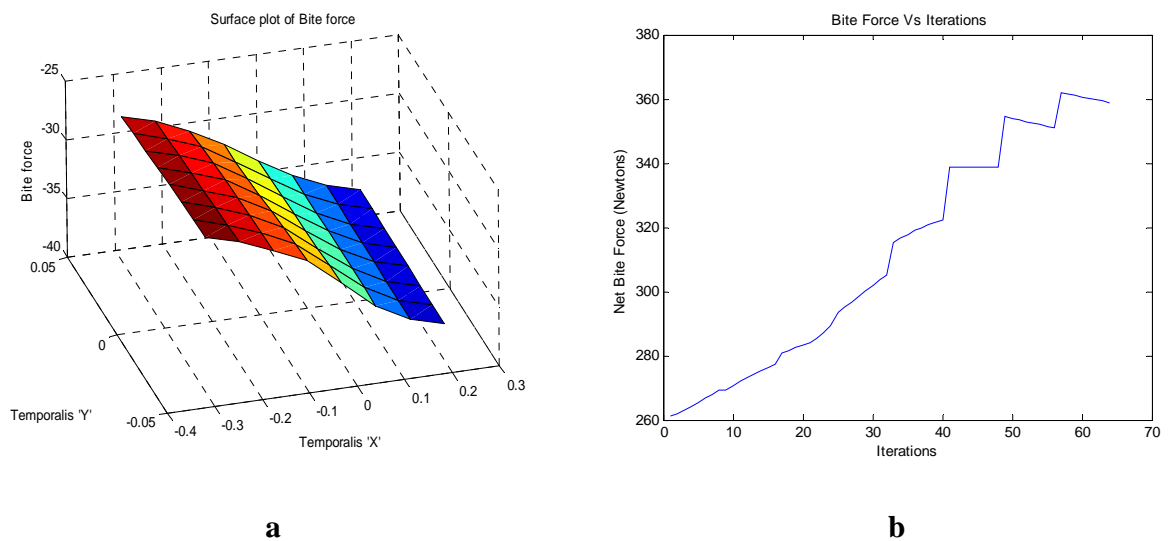
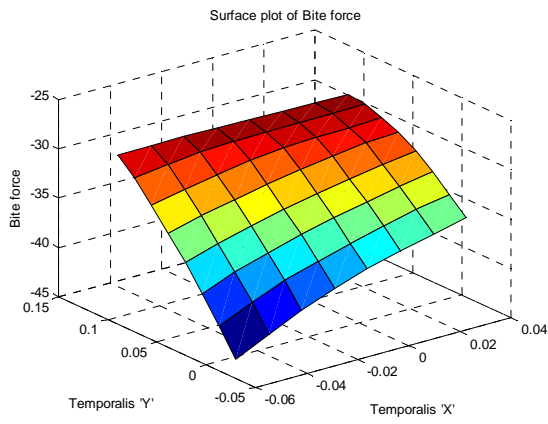
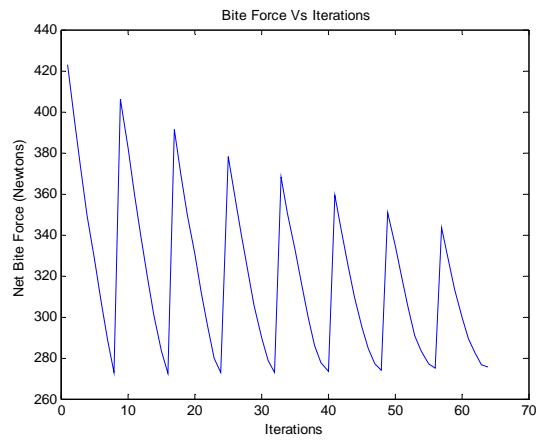


Figure 4-14 Parametric sweep studies showing the variation of bite force with Temporalis origin for tiger (a) surface plot (b) Line plot

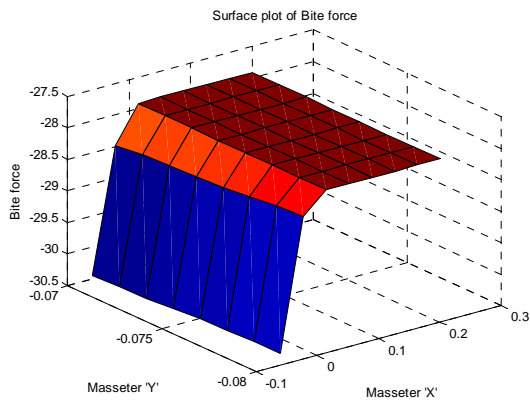


a

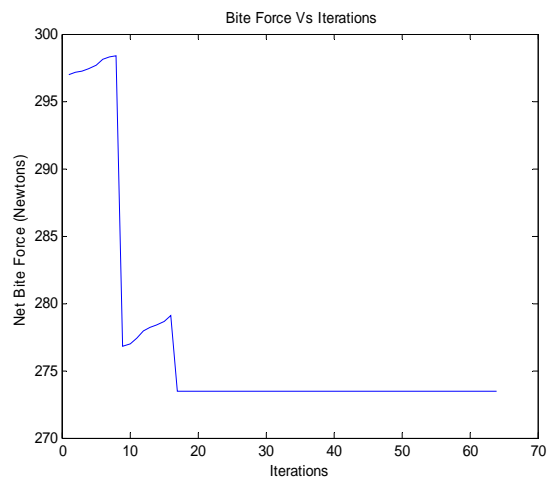


b

Figure 4-15 Parametric sweep studies showing the variation of bite force with Temporalis insertion for tiger (a) surface plot (b) Line plot

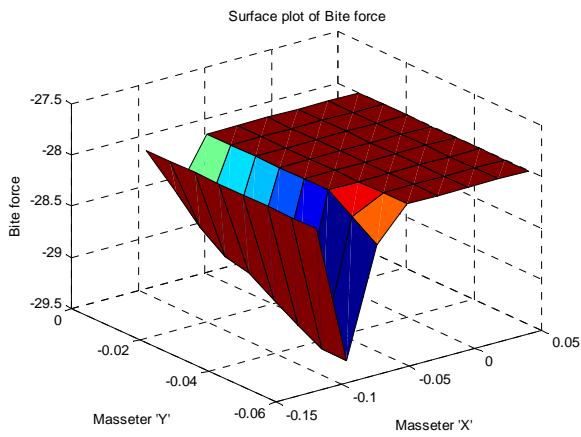


a

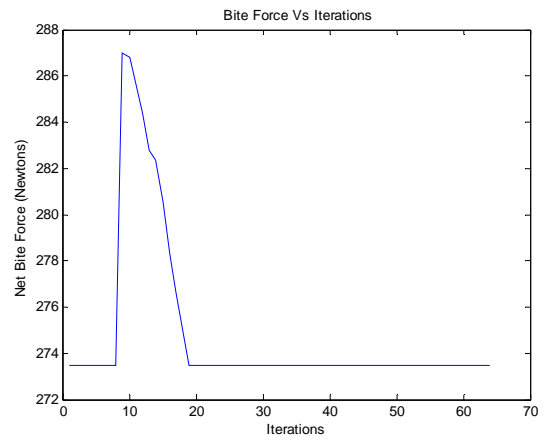


b

Figure 4-16 Parametric sweep studies showing the variation of bite force with Masseter origin for tiger (a) surface plot (b) Line plot

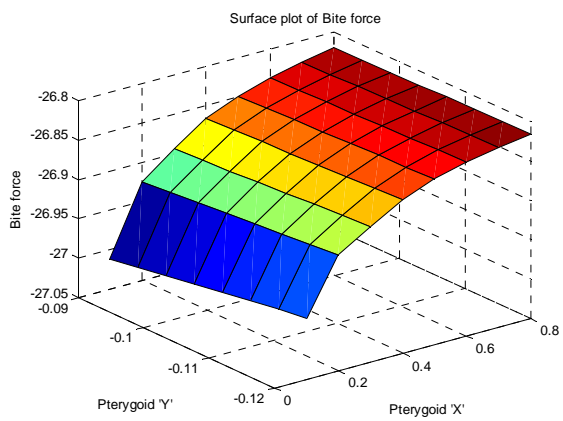


a

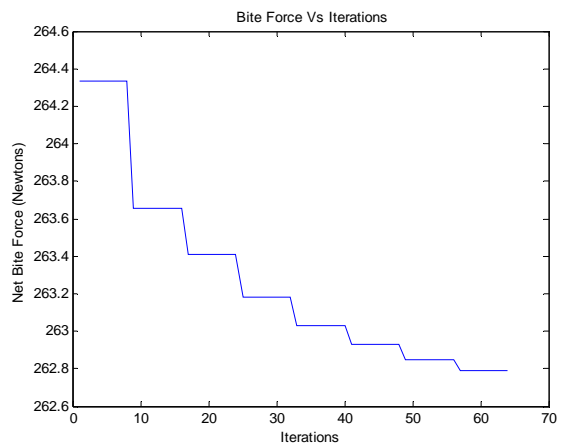


b

Figure 4-17 Parametric sweep studies showing the variation of bite force with Masseter insertion for tiger (a) surface plot (b) Line plot

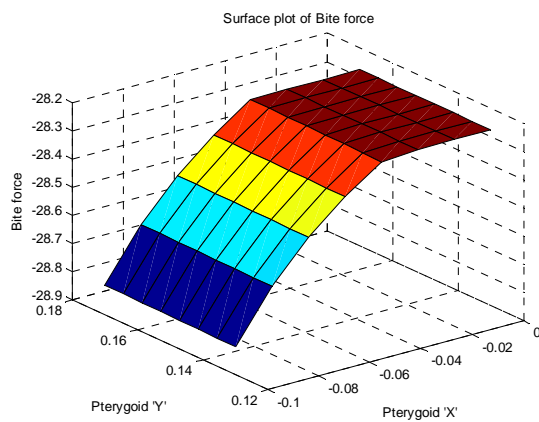


a

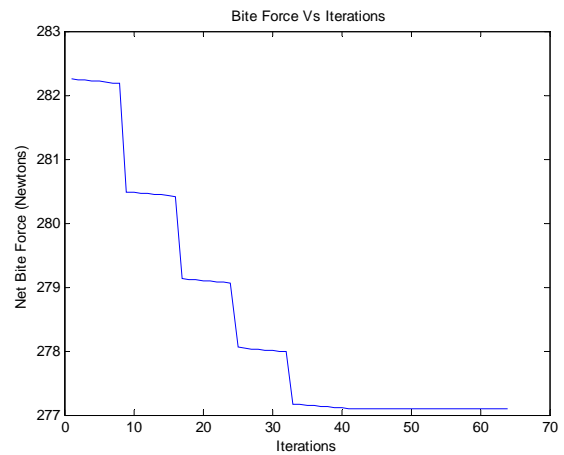


b

Figure 4-18 Parametric sweep studies showing the variation of bite force with Pterygoid origin for tiger (a) surface plot (b) Line plot



a



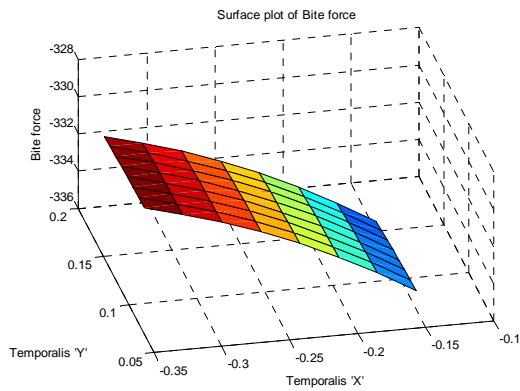
b

Figure 4-19 Parametric sweep studies showing the variation of bite force with Pterygoid insertion for tiger (a) surface plot (b) Line plot

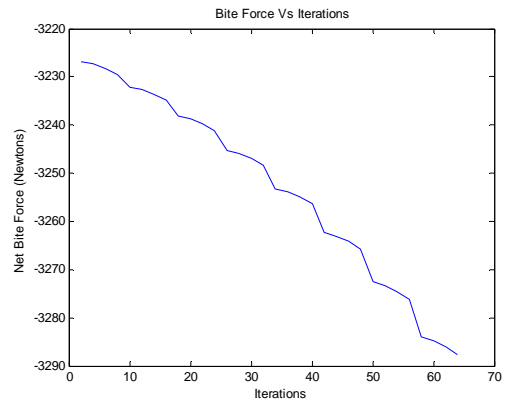
As we see the dependence of the bite force on the design variables in almost all the cases is relatively monotonic. Hence, the minimum of the objective (which would also correspond to the maximum bite force) typically would occur at the boundaries. Each plot contains two subplots – the first subplot shows the value of the bite force for the corresponding sweep of the two selected design variables.

4.4.1.2 Sabertooth tiger - VPMuscle – Simple

Figure 4-14 to Figure 4-19 indicate the variation of bite force with changing muscle origin and insertion for the case 3 i.e. Sabertooth tiger – VPMuscle- Simple. The same brute force method as used for case 1 is employed to span the entire probable area of muscle location(s)

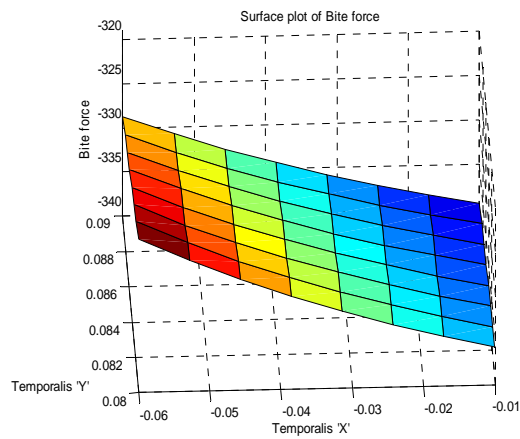


a

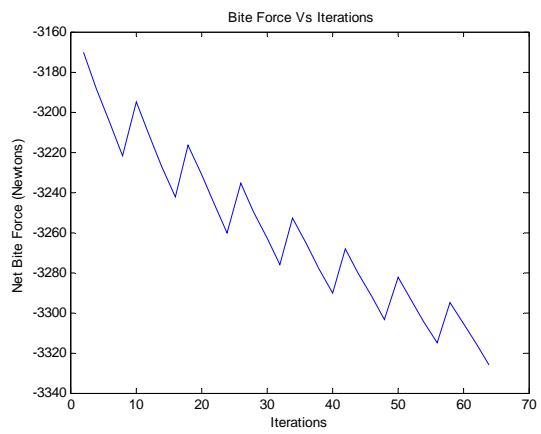


b

Figure 4-20 Parametric sweep studies showing the variation of bite force with Temporalis origin for sabertooth tiger (a) surface plot (b) Line plot

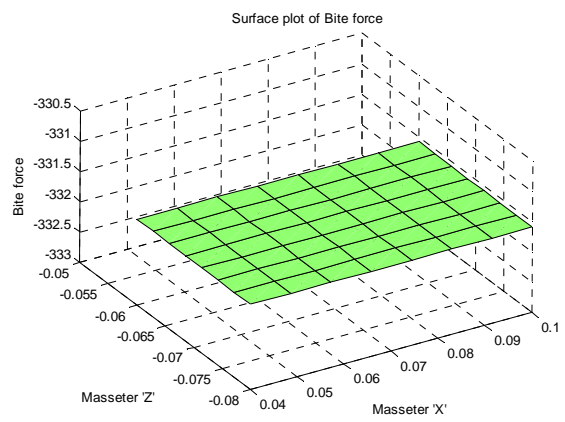


a

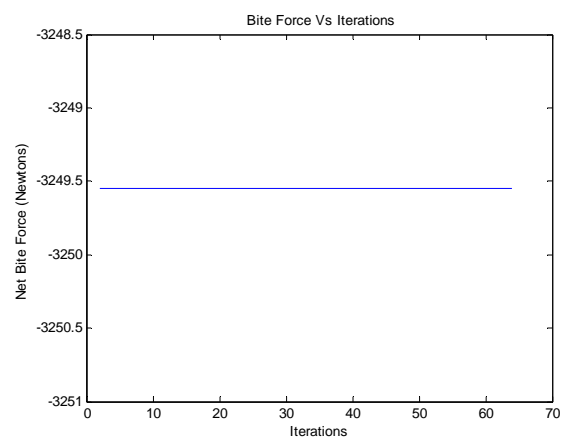


b

Figure 4-21 Parametric sweep studies showing the variation of bite force with Temporalis insertion for sabertooth tiger (a) surface plot (b) Line plot

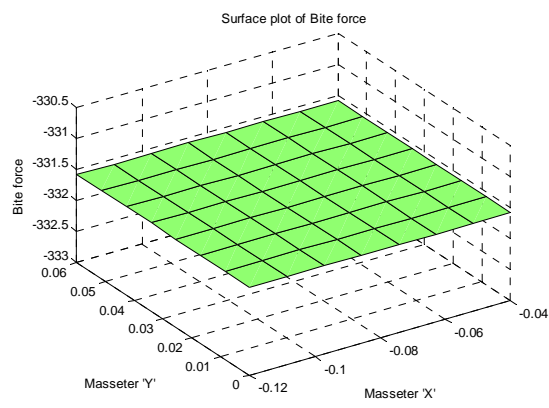


a

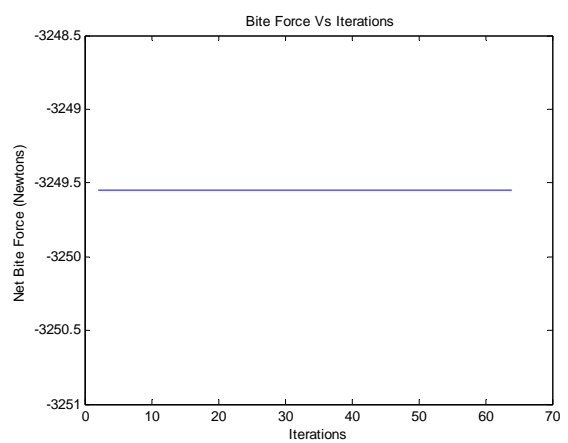


b

Figure 4-22 Parametric sweep studies showing the variation of bite force with Masseter origin for sabertooth tiger (a) surface plot (b) Line plot

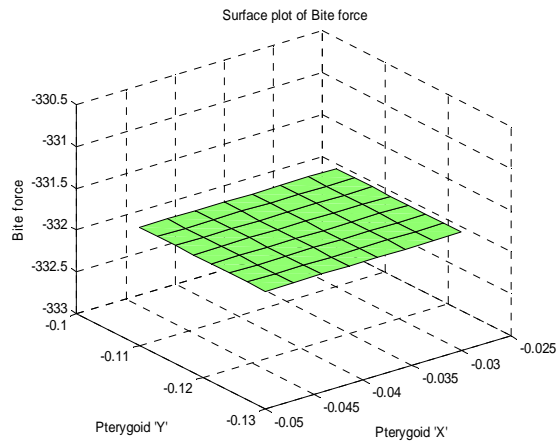


a

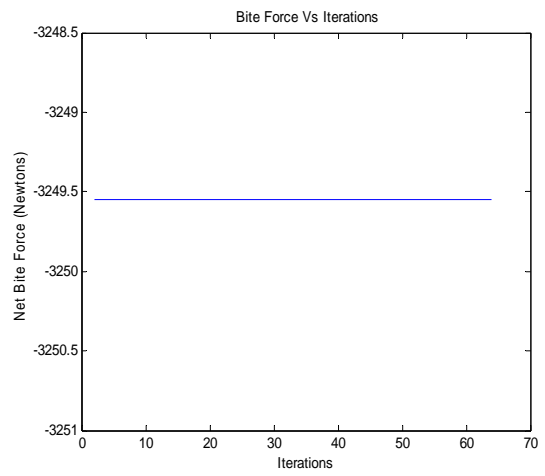


b

Figure 4-23 Parametric sweep studies showing the variation of bite force with Masseter insertion for sabertooth tiger (a) surface plot (b) Line plot

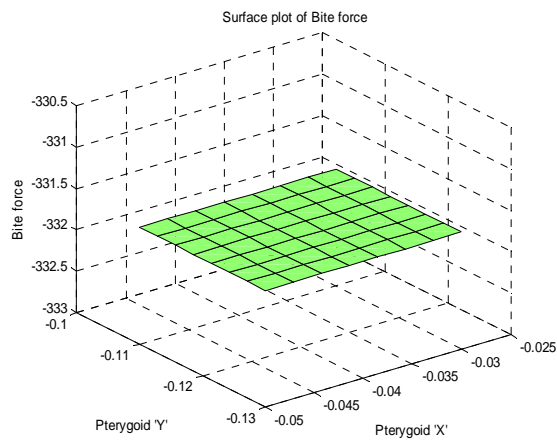


a

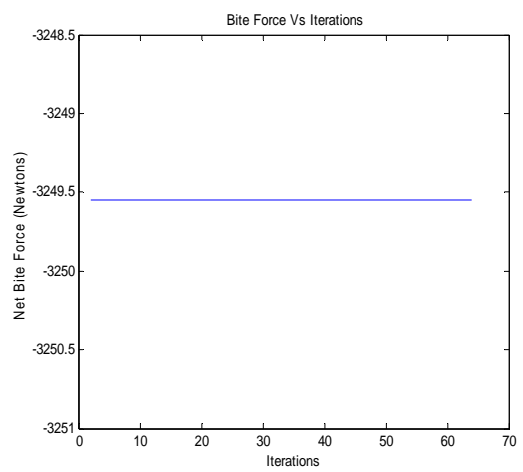


b

Figure 4-24 Parametric sweep studies showing the variation of bite force with Pterygoid origin for sabertooth tiger (a) surface plot (b) Line plot



a



b

Figure 4-25 Parametric sweep studies showing the variation of bite force with Pterygoid insertion for sabertooth tiger (a) surface plot (b) Line plot

Note that all these plots were computed using nominal values for all design variables to as shown in Table 4-2. While the specific sets of selected parametric sweep variables were swept between their maximum and minimum ranges.

For example Figure 4-20 suggests that the bite force would be maximum for the (x,y) location of origin of Temporalis at (-0.2,0.042). Figure 4-23, Figure 4-24 suggest that the location of the origin or insertion of the Masseter do not influence maximum bite force significantly.

4.4.2 Full Optimization:

The full optimization was attempted but we found that the solution typically converged to one of the boundaries. Thus for the problem selected, this solution appears to be the muscle location for creating maximum bite force this appears to be validated by the parametric sweep solution too. However, in other problems the optimal location of muscles may not occur at the boundaries. This is some thing that will be required careful validation.

The bite force graphs for the eight cases as shown in the following figures.

Case 1: Tiger – VPMuscle - Simple

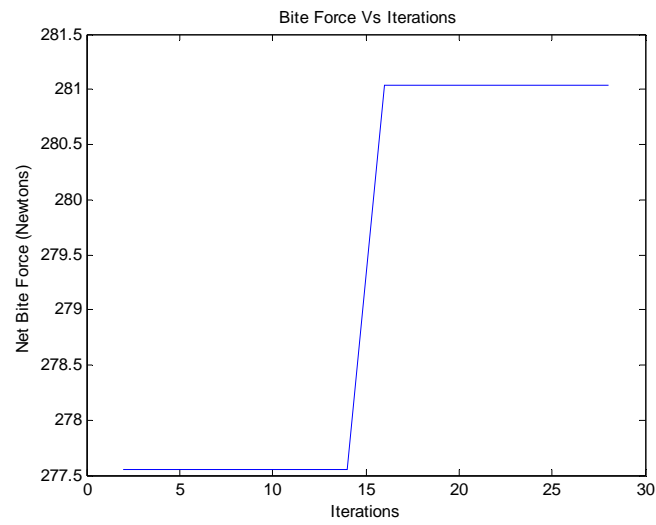


Figure 4-26 Bite force graph for Tiger – VPMuscle - Simple

Case 3: Tiger – SPMuscle - Simple

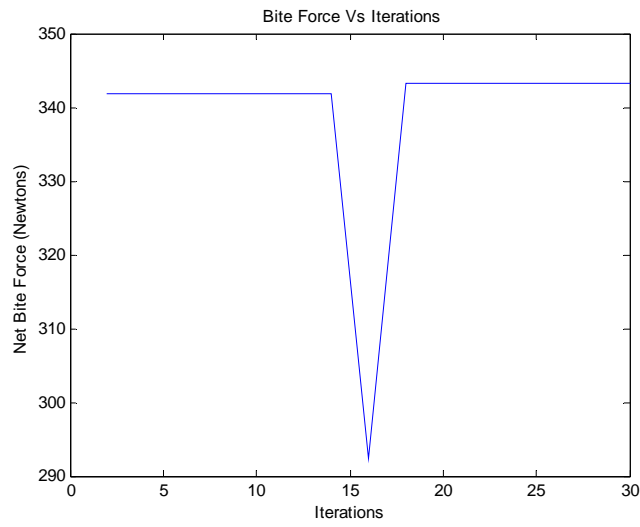


Figure 4-27 Bite force graph for Tiger – SPMuscle - Simple

Case 5: Sabertooth tiger - VPMuscle - Simple

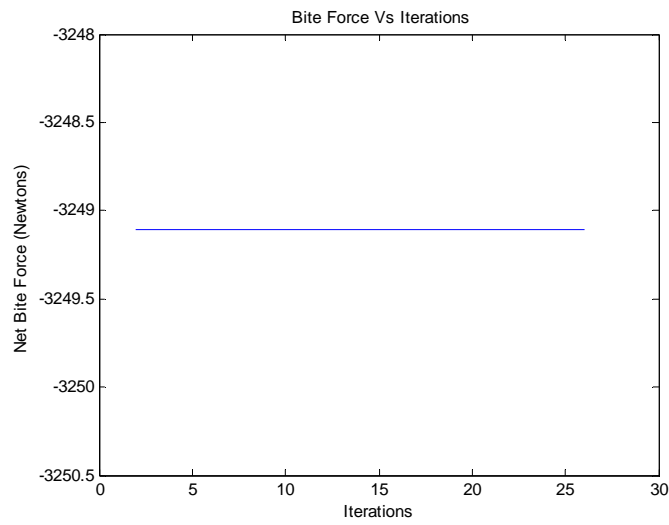


Figure 4-28 Bite force graph for Sabertooth tiger – VPMuscle - Simple

Case 7 : Sabertooth tiger – SPMuscle - Simple

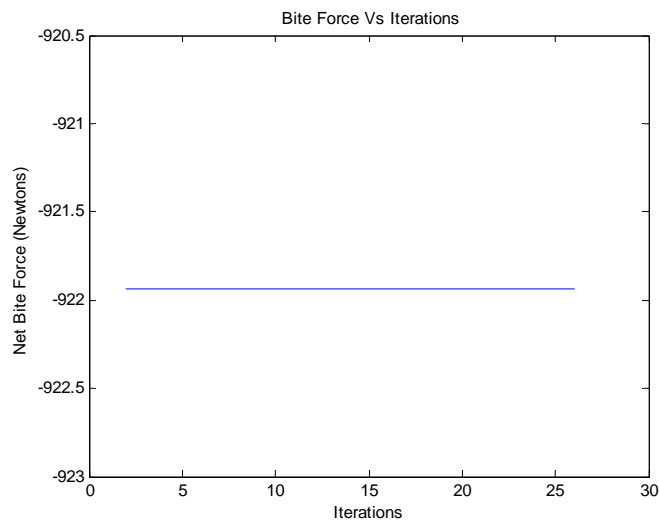


Figure 4-29 Bite force graph for Tiger – SPMuscle - Simple

Case 2: Tiger – VPMuscle - Complex

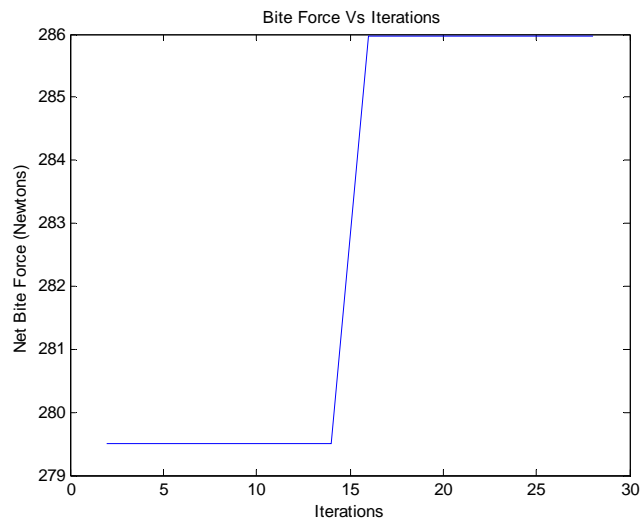


Figure 4-30 Bite force graph for Tiger – VPMuscle - Complex

Case 4: Tiger – SPMuscle - Complex

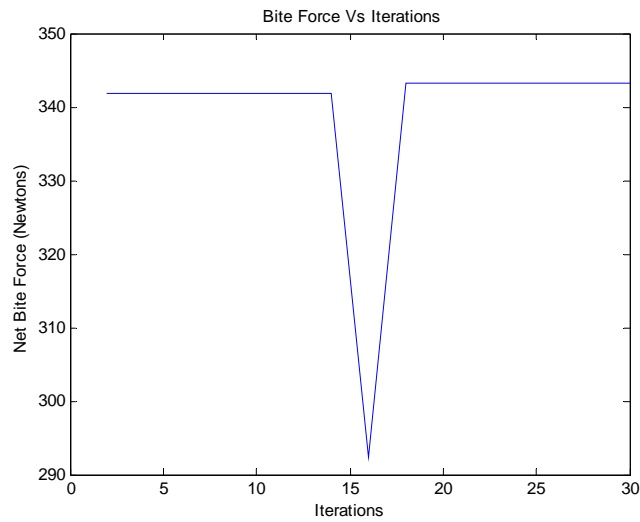


Figure 4-31 Bite force graph for Tiger – SPMuscle - Complex

Case 6: Sabertooth tiger - VPMuscle - Complex

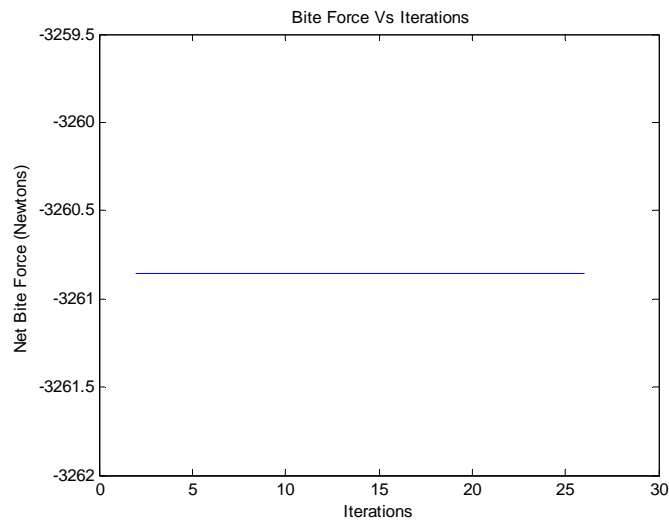


Figure 4-32 Bite force graph for Sabertooth tiger – VPMuscle - Complex

Case 8 : Sabertooth tiger - SPMuscle – Complex

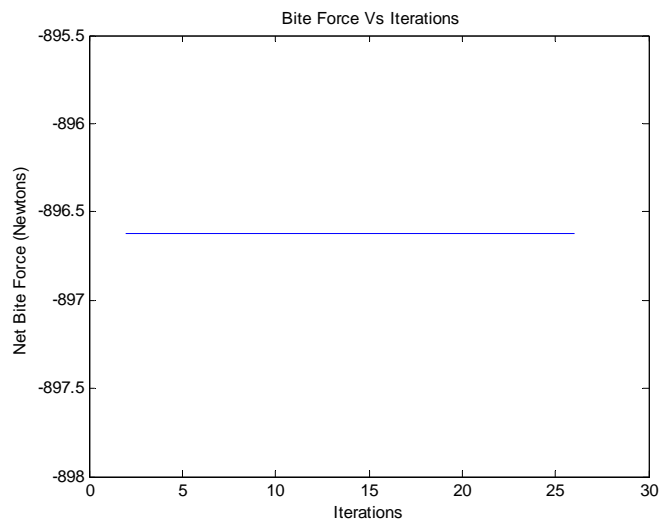


Figure 4-33 Bite force graph for Sabertooth tiger – SPMuscle - Complex

The above results are obtained based on the following assumptions

1. The skull and mandible joint is assumed to be revolute joint
2. The muscles are attached to the bone as point contacts
3. The bone density values for the tiger and sabertooth tiger are taken based on the values found in [36].
4. The maximum limit on muscle force which has a considerable impact on the bite force results is taken based on intuition.

The important thing is that these values can be changed easily and the analyses can be rerun for different animals.

5 Conclusion And Future Work

5.1 Conclusion

The principal premise of this work was that contemporary tools of biomechanical engineering analysis such as computational musculoskeletal analysis have reached a maturity that can be used to ratify or eliminate hypotheses pertaining to how various animals moved, hunted and killed. The approach adopted was one of constructing accurate biomechanically constrained computational virtual prototypes of living/extinct animals and then performing a range of virtual/ computational experiments under known and estimated conditions in order to study the functional performance.

In the past, several factors including: (i) inadequate biological computational tools; (ii) inherent complexity of the system; or (iii) lack of exposure to advanced technology; had hindered the ability of the biological scientists to bring to bear the enormous computational power effectively and efficiently. Hence our efforts were focused on developing a user-friendly computational framework that would permit computational testing of various hypotheses and what-if scenarios, identifying the critical issues, validating the tools, and finally presenting the biological scientists with a novel validated toolbox.

We began the process with point data (CT/MRI data) which was converted into an intermediate geometric model. A constrained virtual prototype that preserves the geometric features was then created by progressively adding physiological constraints (such as muscles, tendons), physical constraints (such as impenetrability, motion limits

and actuator capacities) and other constraints pertaining to material, strength, rigidity. A commercial-off-the-shelf package developed for musculoskeletal analysis (AnyBody) was used to develop the models using the proprietary AnyScript language. While offering a convenient method of construction of a complex musculoskeletal model, the scripting process using AnyScript requires considerable programming experience and at the same time is fraught with potential for user-input errors. Hence a user-friendly and intuitive MATLAB interface was created to guide a relatively uninitiated user through the various stages of defining the multibody musculoskeletal model with all the constraints and running the case studies.

Many challenges arising from the inherent geometric complexity of biological data, the level of detail of modeling of individual muscles as well as clustering of muscles in groups and determining their recruitment pattern needed to be resolved prior to the successful implementation of a virtual prototype. However, once these challenges are resolved, the resulting constrained musculoskeletal model is well suited for virtual/computational analysis. A range of virtual experiments/ analysis under a wide range of initial conditions and actuation schemes can now be systematically and iteratively performed on the computational virtual prototype in order to study the functional performance.

Case studies using musculoskeletal jaw models of members of the feline family – sabertooth tiger (*Smilodon fatalis*) and the tiger (*Panthera tigris*) – were used to highlight critical steps in this multistage process. In particular, our attention was focused on three sets of studies pertaining to: (i) redundancy resolution to determine muscle recruitment (ii) bite force analysis in presence of redundant actuation and (iii) determination of optimum muscle location points based on maximizing bite-force. Finally, the results section also

discusses briefly how the analysis results can be used by biologists to test various ‘if then’ scenarios virtually on the computer and either ratify or eliminate some of the hypotheses.

5.2 Future Work

Optimization routine: In this work we have used brute force method to evaluate the design space of muscle origin and insertion points. A more sophisticated algorithm will be used in future.

Limitations of the software: The MATLAB interface restricts the user from modifying the modeling parameters and other variable pertaining to segments (rigid bodies), joints, drivers, and muscle etc. once they are defined. These issues need to be dealt in the future versions.

Issues pertaining to modeling: Through out our project we have approximated the muscle origin and insertion as points, which is not the case in reality. The future work will include the solution strategy to this issue like proving a curtain of muscles instead of single muscle etc.

Task space redundancy: In the present work, we can calculate the bite force at the tip of one tooth. To simultaneously calculate the bite force at the tips of two teeth, we need to first resolve the issue of task space redundancy. One can adopt a screw theory (similar to the approach followed by delSignore [37]) in order to solve the problem of task space redundancy.

Bibliography

- [1] G. G. Wang, "Definition and Review of Virtual Prototyping," in *J. o. Comp. A. Inf. Sc. In Eng.* University of Manitoba, Winnipeg, MB,R3T 5V6 Canada, 2002.
- [2] C. P. Tang, Bhatt, R. M., and Krovi, V., "Web-Base Self-paced Virtual Prototyping Tutorials," presented at ASME 2003 Design Engineering Technical Conferences and Computer and Information in Engineering Conference, Chicago,Illinois USA, 2003.
- [3] R. E. Schleihauf, *Biomechanics of Human Movement*.
- [4] S. P. McGuan, "Achieving Commercial Success with Biomechanics Simulation," presented at XXth International Symposium on Biomechanics in Sports, Caceres, Spain, 2002.
- [5] J. M. Winters, *Terminology and Foundatin of Movement Science, in Biomechanics and Neural Control of Posture an dMovement*. New York: Springer-Verlag, 2000.
- [6] B. P. Heilman, and Kirsch,R. F, "Selection of an Optimal Muscle Set for a Standing Neuroprosthesis Using a Human Musculoskeletal Model," presented at Proceedings of the 25th Annual International Conference of the IEEE EMBS, Cancun, Mexico., 2003.
- [7] M.M.Ayoub, "A 2-D Simulation Model for Lifting Activities," *Computers and Industrial Engineering*, vol. 35, pp. 619-622, 1998.
- [8] H. E. J. Veeger, Yu Bing, An, K.N, and Rozendal, R.H., "Parameters for modeling the upper extremity," *Journal of Biomechanics*, vol. 30, pp. 647-652, 1997.
- [9] G. K. Kazunori hase, Gang-Moon Eom, "A simulation study on the evolution of hopping motions in animals," *IEEE Transactions on System,Man, and Cybernetics*, vol. 34, 2004.

- [10] P. C. Dechow, and Carlson D.S., "A method of bite force measurement in primates," *J Biomech*, vol. 16, pp. 797–802, 1983.
- [11] J. J. Thomason, and Russell, A.P., "Mechanical factors in the evolution of the mammalian secondary palate: a theoretical analysis," *J Morphol*, vol. 189, pp. 199-213, 1986.
- [12] E. R. Dumont, and Anthony Herrel, "The effects of gape angle and bite point on bite force in bats," *The Journal of Experimental Biology*, vol. 206, pp. 2117-2123, 2003.
- [13] M. B. Meers, "Maximum Bite Force and Prey Size of *Tyrannosaurus rex* and Their Relationships to the Inference of Feeding Behavior," *Historical Biology: A Journal of Paleobiology*, vol. 16, pp. 1-12, 2003.
- [14] D. Verwajen, Van Damme, R., and Herrel, A., "Relationships between head size, bite force, prey handling efficiency and diet in two sympatric lacertid lizards," *Journal of Functional Ecology*, vol. 16, pp. 842–850, 2002.
- [15] M. C. Raadsheer, van Eijden, T.M., van Ginkel, F.C., and Prah-Andersen, B., "Contribution of jaw muscle size and craniofacial morphology to human bite force magnitude," *J Dent Res*, vol. 78, pp. 31-42, 1999.
- [16] R. Davoodi, and Loeb, G. E., "Conversion of SIMM to SIMULINK for faster development of musculoskeletal models," in *IFESS*, 2001.
- [17] S. W. R. Shadmehr, "A Simple Muscle Model."
- [18] R. Davoodi, Brown, I.E., Lan, N., Mileusnic, M., and Loeb, G.E., "An Integrated Package of Neuromusculoskeletal Modeling Tools in Simulink," presented at 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Istanbul, Turkey, 2001.
- [19] K. F. Hulme, Mendel, F.C., and Chugh, K.P., "Development of a Computational Toolkit for Biomechanical Analysis and Simulation: The Vertebrate Analyzer," presented at ICESA 16th International Conference on Computer Applications in Industry and Engineering, Las Vegas, Nevada, 2003.
- [20] J. Rasmussen, Vit Vondrak, Michael Damsgaard, Mark de Zee, and Søren T. Christensen, "The AnyBody project - Computer Analysis of the Human Body," presented at The Czech Society of Biomechanics, Cejkovice, Czech Republic, 2002.

- [21] J. D. John Rasmussen*, Michael Damsgaard, Mark de Zee & Søren Tørholm Christensen, "Musculoskeletal Modeling as an Ergonomic Design Method," presented at International Ergonomics Association XVth Triennial Conference, Seoul, Korea, 2003.
- [22] "www.anybodytech.com."
- [23] "<http://medical.nema.org/>."
- [24] "http://www.materialise.com/mimics/mimbase_ENG.html."
- [25] A. Turner, and M., Anton, "The Big Cats and their Fossil Relatives," 1997.
- [26] J. J. Collins, "The redundant nature of locomotor optimization laws," *Journal of Biomechanics*, vol. 28, pp. 251–267, 1995.
- [27] T. W. Lu, O'Connor, J. J., Taylor, S. J., Walker, P. S., "Validation of a lower limb model with in vivo femoral forces telemetered from two subjects," *Journal of Biomechanics*, vol. 31, pp. 63–69, 1998.
- [28] M. G. Pandy, F. C. Anderson, et al., "A Parameter Optimization Approach for the Optimal-Control of Large-Scale Musculoskeletal Systems." *Journal of Biomechanical Engineering-Transactions of the Asme*, vol. 114(4), pp. 450-460, 1992.
- [29] M. G. Pandy, Zajac, F.E., Sim, E., Levine, W., "An Optimal Control Model for Maximum-Height Human Jumping." *Journal of Biomechanics*, vol. 23(12), pp. 1185-1198, 1990.
- [30] E. Todorov, and Jordan, M.I, "Optimal feedback control as a theory of motor coordination," *Nature Neuroscience*, vol. 5(11), pp. 1226-1235, 2002.
- [31] M. L. Nagurka, and Yen, V., "Fourier-Based Optimal Control of Nonlinear Dynamic Systems.," *Journal of Dynamic Systems, Measurement, and Control*, vol. 112, pp. 17-26, 1990.
- [32] R. Raikova, and Aladjov, H.Ts., "Hierarchical Genetic Algorithm Versus Static Optimization – Investigation of Elbow Flexion and Extension Movements," *Journal of Biomechanics*, vol. 35, pp. 1123-1135, 2002.
- [33] R. Happee, "Inverse Dynamic Optimization Including Muscular Dynamics, A New Simulation Method Applied to Goal Directed Movements," *Journal of Biomechanics*, vol. 27(7), pp. 953-960, 1994.

- [34] J. C. Bean, Chaffin, D. B., "Biomechanical Model Calculation of Muscle Contraction Forces: A Double Linear Programming Method," *Journal of Biomechanics*, vol. 21(1), pp. 59-66, 1988.
- [35] J. Rasmussen, Damsgaard, M., and Voigt, M., "Muscle recruitment by the min/max criterion – a comparative study," *Journal of Biomechanics*, vol. 34, pp. 409-415, 2001.
- [36] <http://hypertextbook.com/facts/2002/AnnaYaruskaya.shtml>.
- [37] Michael J. Del Signore, Venkat Krovi, and Frank Mendel, "Virtual Prototyping and Hardware-in-the-Loop Testing for Musculoskeletal System Analysis," *International Conference on Mechatronics and Automation*, 2005.

Appendix A

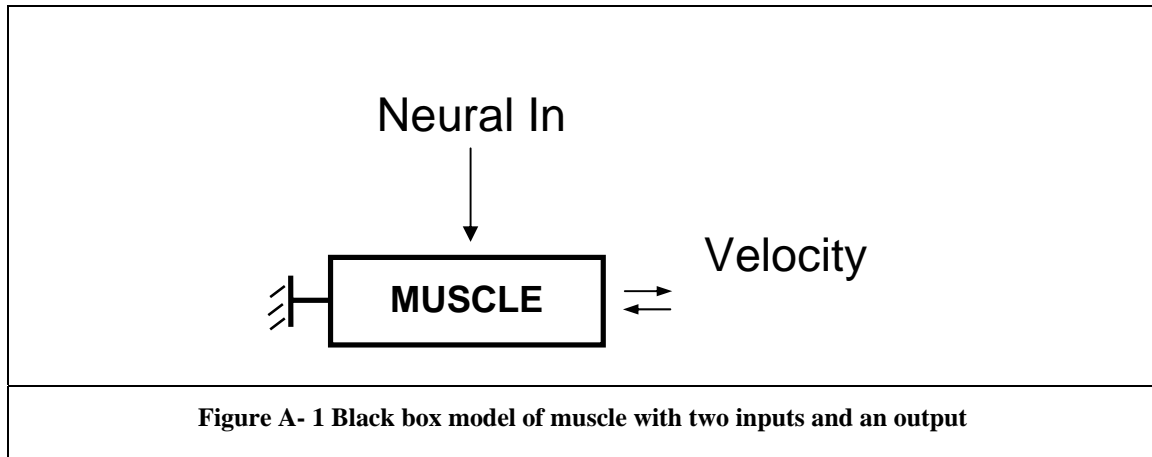
Sliding filament hypothesis (Huxley) specifies how the distribution of cross-bridges evolves as a function of length, velocity, and neural input. This leads to a PDE. Keeping track of that distribution explicitly is not computationally feasible. This models is complete but hard to simulate.

Distribution moment model (Zahalak) approximates the cross-bridge distribution with a Gaussian, and derives ODEs for the mean and variance of the Gaussian. Fast and surprisingly accurate, but the math is hard to understand, so people don't use it.

Bio mechanics researchers are not only interested in constructing musculoskeletal models but they are also interested in muscle forces and how the geometric relationship between muscles and bones transform into moments about the joints. In order to unveil these issues many mathematical models and software's are being developed. One such mathematical model of muscle that is being widely used in biomechanics domain is the Hill's Muscle model.

A.1 Hills Muscle Model

In studying the Muscle properties Hill used the reductionist approach i.e obtaining the physiological data without directly attributing the observed behavior to any biophysical mechanisms. Ignoring the thermodynamic effects, Hill suggested there are two inputs and one output for a muscle as shown in Figure A- 1.



The “Neural In” is the input obtained from the motor neuron (MN). There are two classical types of motor neuron inputs. 1. Twitch or the Impulse input, a brief electrical pulse causing sudden activation of the muscle. 2. Tetanus or step input which has the ability to generate maximum muscle force. Depending on the type of work muscle performs these inputs are provided by the central nervous system. This is a unidirectional input meaning the motor neuron has direct effect on the muscle activation whereas the muscle activation doesn’t have a direct effect on MN.

The other type of input to the muscle is the mechanical input from the mechanical interface i.e. where the muscle couples to the environment. It is a bidirectional input. Contact force f , and a kinematic variable v , velocity, have been chosen as the inputs that interface through this port. If force is the input Velocity is the output and vice-versa.

By controlling these inputs under various testing conditions such as Isometric twitch, Isometric tetanus, free twitch etc. in an experimental setup (described below), Hill came up with two classical muscle models.

A.2 Experimental Setup:

A muscle is attached to a bar or pivot as shown in Figure A- 2. One end of the bar has a catch mechanism that will hold the bar fixed until the catch is released. A weight is attached to the other end of the bar. When the catch is released, with no muscle activation, the weight will pull the muscle by a force 'T' equal to the weight. The muscle is activated maximally which resulted in a muscle force ' T_o '. At this moment the catch is released. The muscle shortened as shown in figure C and the force dropped from ' T_o ' to 'T' as shown in figure B. After the rapid phase of shortening, the muscle shortened gradually as shown in Figure C. The immediate shortening of muscle by Δx_1 suggest that there is spring like element. This spring is the series element (SE), and its stiffness is given by

$$K_{SE} = \frac{\Delta f}{\Delta L}.$$

After the immediate change in muscle length we observe a gradual change without any change in the muscle force. This shows that there is a damper that is acting on the spring that is slowing down the response. This passive element can be referred to as the parallel elastic element (PE) with spring stiffness of K_{PE} . The parallel elastic element, muscle viscosity and the series elastic element constitute the muscle's passive elements.

In a muscle, when the tension in the system suddenly decreases, the series elastic element responds immediately, but the parallel elastic element responds gradually because of its viscous nature.

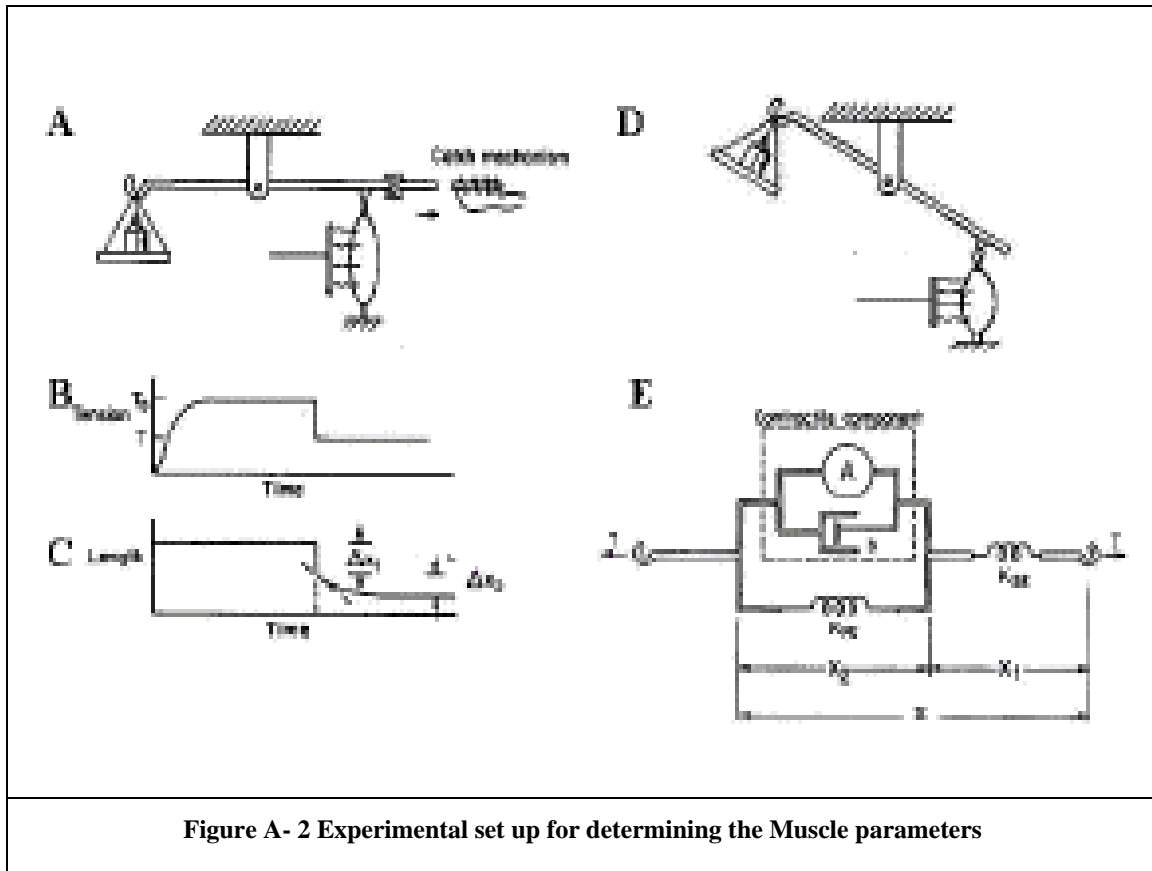
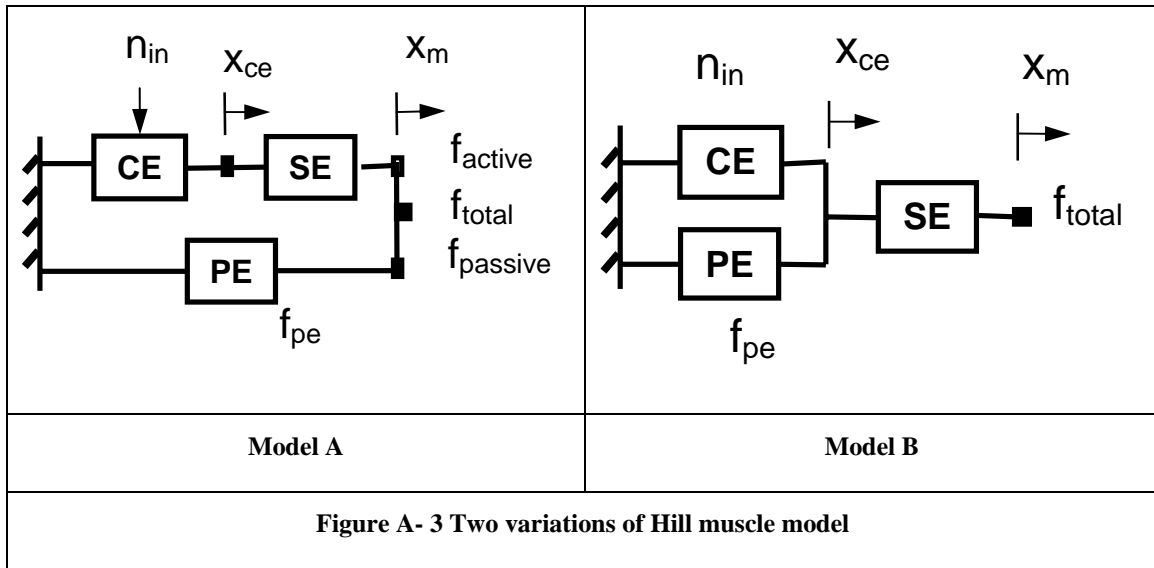


Figure A- 2 Experimental set up for determining the Muscle parameters

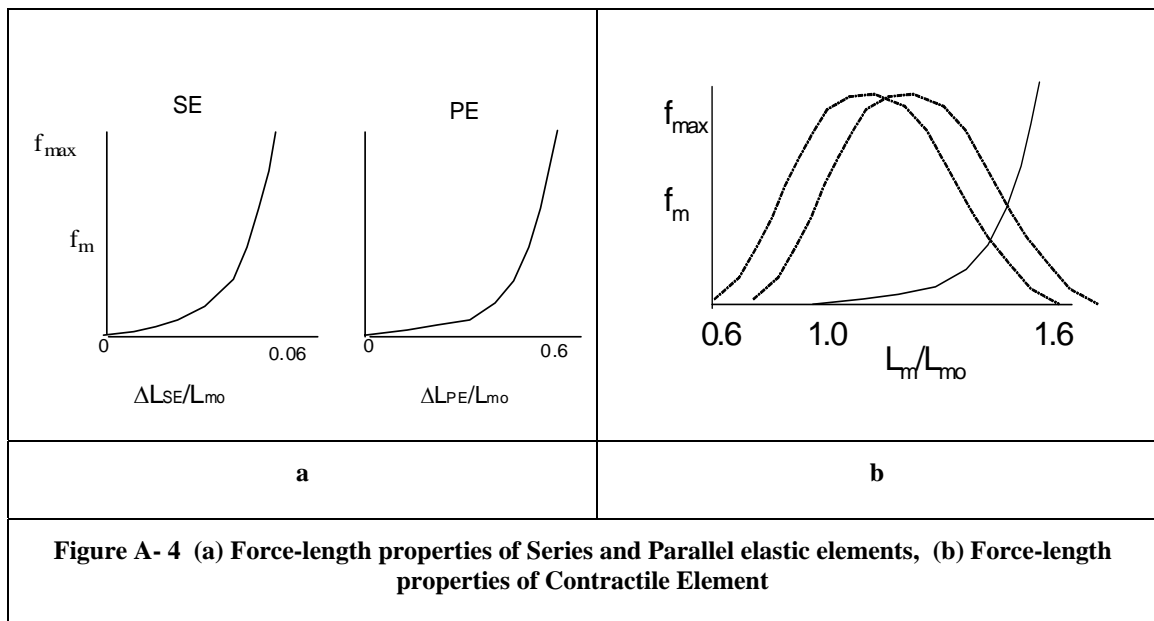
The muscle's active component contributes the final piece of the mathematical muscle model. This active force acts against the passive components of the model (and muscles) to produce the final force that acts on the bar in Figure 2D. Function A indicates the active component in Figure 2E.

From the above experimental setup, Hill came up with two different 3 element muscle models. The three elements of the model are Contractile Element (CE, muscle fibers), the parallel elastic element (PE, connective tissues around the fibers and fiber bundles) and the series elastic element (SE, muscle tendon). The relationships thus developed for CE, SE and PE are called constitutive relations (relations that tend to describe observed behavior by idealized lumped elements).



From the above two models, Model A is better to Model B because of the following reasons.

1. In Model A , when $n_{in} = 0$ and the muscle is extended f_{total} is due to the passive stretch $f_{passive}$ only, which is true in real muscles, where as it is not in Model B.
2. There are many other passive spring-like elements, structurally in parallel in a real muscle. All these can be easily lumped together in Model B.



From the graphs shown in Figure A- 4**Error! Reference source not found.**, we observe that CE represents a more complex type relationship. It has multiple curves at each level of activation. So the force-length relationships are activation dependent in CE.

For SE and PE, the force-length relationships are more like a passive spring. The slope of the curve indicates the spring stiffness. The shape of the curve suggests that the force-length relationship is nonlinear.