A NOVEL MUSCULOSKELETAL JOINT MODELING FOR ORTHOPAEDIC APPLICATIONS

A thesis submitted for the degree of Doctor of Philosophy

By

Neriman Ozada

School of Engineering and Design

Brunel University

August, 2008

Abstract

The objective of the work carried out in this thesis was to develop analytical and computational tools to model and investigate musculoskeletal human joints. It was recognised that the FEA was used by many researchers in modelling human musculoskeletal motion, loading and stresses. However the continuum mechanics played only a minor role in determining the articular joint motion, and its value was questionable. This is firstly due to the computational cost and secondly due to its impracticality for this application. On the other hand, there isn't any suitable software for precise articular joint motion analysis to deal with the local joint stresses or non standard joints. The main requirement in orthopaedics field is to develop a modeller software (and its associated theories) to model anatomic joint as it is, without any simplification with respect to joint surface morphology and material properties of surrounding tissues. So that the proposed modeller can be used for evaluating and diagnosing different joint abnormalities but furthermore form the basis for performing implant insertion and analysis of the artificial joints. The work which is presented in this thesis is a new frame work and has been developed for human anatomic joint analysis which describes the joint in terms of its surface geometry and surrounding musculoskeletal tissues. In achieving such a framework several contributions were made to the 6DOF linear and nonlinear joint modelling, the mathematical definition of joint stiffness, tissue path finding and wrapping and the contact with collision analysis. In 6DOF linear joint modelling, the contribution is the development of joint stiffness and damping matrices. This modelling approach is suitable for the linear range of tissue stiffness and damping properties. This is the first of its kind and it gives a firm analytical basis for investigating joints with surrounding tissue and the cartilage. The 6DOF nonlinear joint modelling is a new scheme which is described for modelling the motion of multi bodies joined by non-linear stiffness and contact elements. The proposed method requires no matrix assembly for the stiffness and damping elements or mass elements. The novelty in the nonlinear modelling, relates to the overall algorithmic approach and handling local non-linearity by procedural means. The mathematical definition of joint stiffness is also a new proposal which is based on the mathematical definition of stiffness between two bodies. Based on the joint stiffness matrix properties, number of joint stiffness invariants was obtained analytically such as the centre of stiffness, the principal translational stiffnesses, and the principal rotational

stiffnesses. In corresponding to these principal stiffnesses, their principal axes have been also obtained. Altogether, a joint is assessed by six principal axes and six principal stiffnesses and its centre of stiffness. These formulations are new and show that a joint can be described in terms of inherent stiffness properties. It is expected that these will be better in characterising a joint in comparison to laxity based characterisation. The development of tissue path finding and wrapping algorithms are also introduced as new approaches. The musculoskeletal tissue wrapping involves calculating the shortest distance between two points on a meshed surface. A new heuristic algorithm was proposed. The heuristic is based on minimising the accumulative divergence from the straight line between two points on the surface and the direction of travel on the surface (i.e. bone). In contact and collision based development, the novel algorithm has been proposed that detects possible colliding points on the motion trajectory by redefining the distance as a two dimensional measure along the velocity approach vector and perpendicular to this vector. The perpendicular distance determines if there are potentially colliding points, and the distance along the velocity determines how close they are. The closest pair among the potentially colliding points gives the "time to collision". The algorithm can eliminate the "fly pass" situation where very close points may not collide because of the direction of their relative velocity. All these developed algorithms and modelling theories, have been encompassed in the developed prototype software in order to simulate the anatomic joint articulations through modelling formulations developed. The software platform provides a capability for analysing joints as 6DOF joints based on anatomic joint surfaces. The software is highly interactive and driven by well structured database, designed to be highly flexible for the future developments. Particularly, two case studies are carried out in this thesis in order to generate results relating to all the proposed elements of the study. The results obtained from the case studies show good agreement with previously published results or model based results obtained from Lifemod software, whenever comparison was possible. In some cases the comparison was not possible because there were no equivalent results; the results were supported by other indicators. The modelling based results were also supported by experiments performed in the Brunel Orthopaedic Research and Learning Centre.

Acknowledgments

I am very grateful to my supervisor, Professor Ibrahim Esat for his invaluable help, support and encouragement throughout of my research and giving me this opportunity to work with him.

I want to express my gratitude to Professor Ofer Levy for his guidance and useful advises in orthopaedics field.

I would like to thank Dr. Paul Smirthwaite of Biomet Company for his guidance and useful discussions relating to the artificial joints.

Also I want to thank Dr. Chris Mares for his help throughout of my research.

Very deepest thanks are due to my parents, Tahsin Ozada, Nalan Ozada and my sister Ayse Ozada for their endless love and support. I specially would like to thank to my dad for the financial support which made my studies possible.

Table of Contents

Abstract…… i

Acknowledgments……iii

Table of Contents……iv

List of f Figures……xi

List of Tables……xvi

Abbreviations……...xviii

CHAPTER 1 INTRODUCTION……1

1.1Need for accurate musculoskeletal joint modelling……. 1

1.2Joint modelling background……. 3

1.3Organisation of thesis……... 6

CHAPTER 2 LITERATURE SURVEY……7

2.1Skeletal joint modelling……7

2.1.1Structure and mobility of human skeletal joints……7

2.1.2Kinematic modelling of multi body and skeletal systems……9

2.1.3Kinematic modelling and joint motion measurement techniques: Application to upper limb ……12

2.1.4Dynamic modelling of multi body, skeletal and musculoskeletal systems 17

2.1.5Application of subsidiary and enabling theories in multi body simulations 18

2.1.6Virtual human models……... 20

2.2 Musculoskeletal tissue modelling……21

2.2.1 Musculoskeletal tissues……... 22

2.2.1.1 Ligament and tendon…… 23

2.2.1.2 Cartilage…… 24

2.2.1.3 Muscle…….. 25

2.2.2 Deformation modelling of musculoskeletal tissues……. 27

2.2.3 Modelling of musculoskeletal tissue paths and line of actions…… 32

2.2.4 Deformation simulation of flexible bodies……...38

2.2.4.1 Geometry based deformation simulation methods…… 39

2.2.4.2 Physics based deformation simulation methods……41

2.3Joint contact modelling and collision……... 44

2.3.1Diarthrodial joint contact modelling……. 44

2.3.2Collision detection…….49

2.3.3Contact and collision response…….. 56

2.3.3.1Contact constraints and contact constraint based collision response 58

2.3.3.2 Rigid and flexible contact and penalty based collision response 59

2.3.3.3 Rigid body impact and impulse based collision response..61

CHAPTER 3 THEORY…….. 63

3.1 Introduction……. 63

3.2 Unconstrained (6dof) joint modelling theory…….. 66

3.2.1 Linearity and non-linearity……... 66

3.2.2 Linear joint modelling…….. 67

3.2.2.1 Spring or spring-dashpot elements…… 67

3.2.2.2 Generalization of the equation of linear momentum……. 69

3.2.2.3 Generalization of the equation of moment of momentum.70

3.2.2.4 Assembling the equations of motion……. 71

3.2.3 Development of joint stiffness formulations…… 78

3.2.4 Nonlinear joint modelling…….84

3.2.4.1 Formulation of nonlinear spring elements…….85

3.2.4.2 A non-matrix based motion analysis…… 87

3.2.4.3 Displacements of a point on a body…….. 89

3.3 Development of geometric tissue path finding, line of action and wrapping algorithms…….. 95

3.3.1 Minimum mass and residual covariance algorithms…… 95

3.3.2 Calculation of the centre of rotation and the moment arm……... 107

3.3.2.1 The instantaneous centre of rotation of a "joint"……...107

3.3.2.2 The moment arm calculations……... 108

3.4 Development of collision detection, contact and collision response algorithms 110

3.4.1 Background and rationale of the new algorithm……110

3.4.2 The collision algorithm…….112

3.4.3 The cartilage model…….. 117

3.5 The summary of the proposed theories……... 118

CHAPTER 4 SOFTWARE DEVELOPMENT AND IMPLEMENTATION 119

4.1 Introduction……. 119

4.2 Data structures……. 120

4.2.1 Future development and upgrading of the data structure……. 128

4.3 Linear and nonlinear multi rigid body formulations, algorithms and software implementation…….. 131

4.3.1 Linear multi rigid body formulation for joint stiffness…… 131

4.3.2 Linear multi rigid body, joint stiffness software…….. 132

4.3.2.1 Pseudo code for assembling $\mathbf{R}_{pi}$, $\mathbf{R}_{pj}$ and $\mathbf{R}_{pi}^{T}$ ……133

4.3.2.2 Pseudo code for assembling stiffness matrix for a given spring 133

4.3.2.3 Pseudo code for assembling the global stiffness matrix…134

4.3.2.4 Assembly matrix has to be called for each spring (muscle, ligament) ……...134

4.3.2.5 Preparing for the analysis…….... 134

4.3.3 Nonlinear multi rigid body formulation……... 137

4.3.3.1 Initialise geometry……138

4.3.3.2 Updating geometry during the solution…….139

4.3.3.3 Overall pseudo code…….. 141

4.3.4 Mass and inertia tensor calculations……142

4.3.5 Calculation of principal inertias……... 143

4.3.6 Finding Euler angles from direction cosines…… 144

4.4 Geometric tissue path finding, line of action and wrapping algorithms and software implementation ……. 145

4.4.1 Muscle mesh generation, topology and data…… 145

4.4.2 An evaluation of minimum mass algorithm for muscle wrapping... 149

4.5 Collision detection, contact and collision response algorithms and software implementation…….. 154

4.5.1 Evaluation of the collision algorithm……... 154

4.6General software features……. 156

4.6.1 Screen shots from the program……. 156

4.6.1.1 Interactive options……. 156

4.6.1.2 Example for the “difference” or “subtract” Boolean operation 159

4.6.1.3 Comments about software implementation……... 159

CHAPTER 5 APPLICATIONS, RESULTS AND VALIDATION…… 160

5.1 Introduction……. 160

5.2 Generation of musculoskeletal model and data in MJM……. 161

5.2.1 Bone mesh generation…….. 161

5.2.2 Skeletal system and joint motion terminology……. 164

5.2.3 Musculoskeletal system and tissue modelling……..166

5.2.3.1 Head and neck tissues……167

5.2.3.2 Elbow tissues……. 167

5.2.4 Some functionalities of the developed software MJM……. 168

5.2.4.1 A data creation session on MJM……... 169

5.2.4.2 Boolean operations on MJM……. 172

5.2.4.3 Tissue wrapping…….173

5.3 Lifemod software……..... 173

5.3.1 Comparing the MJM versus the Lifemod…….176

5.4 Joint modelling, comparative results and validation……... 177

5.4.1 Rationale and the basis of validation…….. 177

5.4.2 Head and cervical vertebrae (neck) motion modelling…….178

5.4.2.1 Motion modelling based results of head and cervical vertebrae using the MJM software…… 178

5.4.2.1.1 Half Sine acceleration acting on T1……... 181

5.4.2.1.2 Half Sine force acting on head (T1 is fixed)….. 182

5.4.2.1.3 Static force acting on head (T1 is fixed), deflection, the COR……... 184

5.4.2.1.4 Joint stiffness, the centre of stiffness, joint stiffness invariants……... 187

5.4.2.2 Motion of head and cervical vertebrae using Lifemod software 188

5.4.2.3 Modelling or experimental based results from the literature 191

5.4.2.4 Comparison of results from MJM software, Lifemod software and published data in head and neck modelling applications……193

5.4.3 Elbow joint modelling…….. 193

5.4.3.1 Modelling based results from MJM software……194

5.4.3.1.1 Static deflection and centre of rotation of the ulna-radius complex……. 195

5.4.3.1.2 Centre of stiffness and axis of stiffness……..196

5.4.3.1.3 Dynamic calculations of centre of rotation with respect to the elbow angles…… 196

5.4.3.1.4 Valgus and varus laxity studies with muscle wrapping and non-linear cartilage……200

5.4.3.1.5 Moment arm based results……..201

5.4.3.2 Modelling based results from Lifemod software…….. 203

5.4.3.3. Experiment based results (carried out in the Brunel Orthopaedic Research and Learning Centre)……. 203

5.4.3.3.1 The Stewart platform based centre of rotation of the ulna-radius complex…….. 203

5.4.3.3.2 Valgus and varus laxity studies with the Elbow Brace 206

5.4.3.3.3 Comments on the experimental results…….. 208

5.4.3.4 Modelling or experimental based results from the literature 209

5.4.3.4.1 Centre of rotation of elbow joint…… 209

5.4.3.4.2 Valgus and varus laxity studies with and without tissue deficiency ……. 211

5.4.3.4.3 Moment arm results……..... 214

5.4.4 Summary of results……...215

CHAPTER 6 DISCUSSION…….. 218

6.1 Introduction……. 218

6.2 Objective……. 218

6.2.1 Structures of discussion……218

6.3 Mathematical modelling ……218

6.3.1 Linear modelling…….. 219

6.3.2 Non-linear modelling……219

6.3.3 Joint stiffness concept ……..220

6.3.4 Muscle wrapping ……. 221

6.3.5 Collision detection...…….221

6.4 Software validation…….. 222

6.4.1 Individual modules …… 222

6.5 Evaluation of results…… 224

6.5.1 The joint laxity……. 224

6.5.2 The centre of joint stiffness…….. 224

6.5.3 Non-linear analysis with muscle wrapping…….. 225

CHAPTER 7 CONCLUSIONS AND FUTURE WORK……226

7.1 Conclusions……. 226

7.1.1 Summary of the contributions made in this work…… 226

7.2 The future work……... 228

References…… 229

Appendix…….. 250

A1 Human skeletal bones and joints……. 250

A2 Head and neck …… 252

A3 Elbow ……. 264

A4 Euler angles and direction cosines…… 271

A5 Selected software code…….. 275

A5.1 Connectivity for closed object (box, sphere, cone, cylinder and muscle) 275

A5.2 Inertia matrix code……276

A6 Geometric data preparations from Geomagic Studio 9 ……... 278

List of Figures

Figure 3.1 Body ***i*** and body ***j*** are in contact and connected by spring k_p..... 72

Figure 3.2 Non-linear deflections in three orthogonal axes of mounting……. 85

Figure 3.3 Free body diagram of body ***i*** and body ***j***……..... 88

Figure 3.4 Flow diagram of a program for analysing a multi body system incorporating springs with polynomial type of nonlinearity……93

Figure 3.5 Illustration of proposed shortest path between attachment points …. 97

Figure 3.6 Shortest path wrapping of a ligament between ulna-humeral joint…. 106

Figure 3.7 (a) Instantaneous centre of rotation, (b) Moment arm calculations… 108

Figure 3.8 Interactive attachments of soft tissues and resulting direction of motion 112

Figure 3.9 Illustration of collision algorithm along with applied directional velocity $\boldsymbol{V}_i$ 114

Figure 3.10 Illustration of collision algorithm along with directional velocity $\boldsymbol{V}_i$ and rotation $\boldsymbol{\omega}_i$……... 116

Figure 4.1 Main database tables…… 121

Figure 4.2 Non-linear data, OLE link to Excel……. 123

Figure 4.3 Mass, COG, and inertia tensor calculations…….126

Figure 4.4 Front page of MJM…… 130

Figure 4.5 The axes of principal inertias of two bone segments…….. 143

Figure 4.6 Nodal and topological information of a closed volume mesh generation 147

Figure 4.7 Some examples of mesh generation, the mesh density can be controlled 148

Fig. 4.8 Evaluation the analytical shortest path against the proposed muscle wrapping algorithm 149

Figure 4.9 Ellipsoidal and helix curves plotted on a cylindrical mesh generated by the mesh generator 153

Figure 4.10 Error ratio between the optimal curve and the algorithm proposed in this thesis 154

Figure 4.11 Various collision algorithm performances against each others……. 156

Figure 4.12 Various interactive options, the selections are self explanatory…… 158

Figure 4.13 Neck and head (a) Meshed (b) Rendered and selected with an added cylinder (c) Head–cylinder Boolean operation……. 159

Figure 5.1 Step1of tissue attachment processes, (a) Initial selection (b) Identifying the selection with changing the object colour……. 170

Figure 5.2 Step 2 of tissue attachment processes, (a) User is asked to enter the second point (b) The selection of the second point…….. 170

Figure 5.3 Recording the attached mount (spring) between mass (bone) 2 and mass 1 171

Figure 5.4 Tissue attachments between two bones (masses) ……... 172

Figure 5.5 Examples of MJM functions, (a) Multi tissue wrapping, (b) Boolean operation and implant positioning…… 173

Figure 5.6 Prosthetic knee joint model and tissue wrapping on geometric surface of prosthesis from Lifemod software…… 175

Figure 5.7 Extracts from MJM, (a) Numbering of head and neck masses (bones) (b) Muscle attachments……... 180

Figure 5.8 (a) Half Sine acceleration constraint shock acting on T1, graph shows mass no 4, head movement…….....181

Figure 5.8 (b) Velocity versus time response of head (mass 4) to half Sine shock. x and y directions are plotted, note the motion in the y direction…….. 182

Figure 5.9 Displacement versus time response of head (mass 4) to half Sine shock force acting on the head…….. 183

Figure 5.10 Head and neck flexion with the original geometries……. 183

Figure 5.11 Response of the centre of mass of head under an applied constant horizontal force…….. 185

Figure 5.12 Neck and head motion after half sine acceleration applied to the torso (upper body…….. 189

Figure 5.13 Centre of mass position of head versus time response to frontal impact to the upper torso……... 189

Figure 5.14 Centre of mass velocity of head versus time response to frontal impact applied to the upper torso……. 190

Figure 5.15 Angle of upper neck (joint between Head-C1) versus time response to frontal impact applied to the upper torso which shows one dimensional rotation in sagittal plane (horizontal direction)…….. 191

Figure 5.16 Instantaneous centres of each cervical vertebra during neck rotations (from Bogduk & Mercer, 2000)……. 192

Figure 5.17 Elbow joint model from MJM software…… 194

Figure 5.18 6DOF elbow joint model and the application of centre of movement analysis relative to flexion angle of forearm (ulna, radius and hand)…….. 197

Figure 5.19 3D change in centre of rotations (CoR) versus elbow flexion angle results obtained from MJM …… 199

Figure 5.20 Valgus and varus laxity curves from unloaded and varus-valgus loaded intact elbow joint and from varus-valgus loaded elbow joint with complete LCL (varus stiffness) and MCL (valgus stiffness) incisions…….. 201

Figure 5.21 Force vector (in red) of Bicep muscle at neutral position of the forearm 202

Figure 5.22 Moment arm change of Bicep muscle versus elbow flexion angle…202

Figure 5.23 Steward Platform with 6 Celsco SP1 transducers (Alrashidi et al. 2009) 204

Figure 5.24 In vivo, elbow joint valgus and varus angle measurement of two subjects from Stewart Platform……. 204

Figure 5.25 3D changes in centre of rotations during arm flexion of two subjects from Steward Platform……. 205

Figure 5.26 In vivo experimental set up for measuring elbow joint movements with a potentiometer (PF) for flexion-extension movement and with a strain gauge (SGV) for varus and valgus movement. The extended arm (Left) and flexed arm (Right) .206

Figure 5.27 In vitro experimental set up for measuring elbow joint movements with a potentiometer (PF) for flexion-extension movement and with a strain gauge (SGV) for varus and valgus movement. The experiment is performed under applied varus and valgus loads……. 207

Figure 5.28 Location of the average screw displacement axis (SDA) with respect to humerus and ulna in the frontal plane. β1 represents the angle between the SDA and the longitudinal axis of humerus. β2 represents the angle between the SDA and the longitudinal axis of ulna. The carrying angle is represented as the summation of beta angles as β1 + β2. The excursion of the SDA (2.60) is shown exaggerated by factor of 2.0 to help visualize the site of the smallest dispersion of the axis (Bottlang et al., 2000) 209

Figure 5.29 (a) Valgus and varus movements versus flexion angles of elbow joint. (b) The SDA paths in 2D relative to flexion angles (Bottlang et al., 2000)…….. 210

Figure 5.30 the Idealization pathway of SDA. αh and αf frustum vertex angles, th and tf translations of the SDA in horizontal and frontal planes (2D) (Bottlang et al., 2000) 210

Figure 5.31The linkage system based experimental set up which is commonly utilized to measure flexion-extension, varus-valgus and rotational movements of elbow joint with potentiometers (PF, PV, PR) (Jensen et al., 2005)…….. 212

Figure 5.32 Mean varus displacements relative to elbow flexion with firstly performing radial head excision (a), though firstly performing LCL incision (b), with combination of radial head excision and LCL incision and also combination of radial head replacement and LCL repair (c) (Jensen et al., 2005)…….. 213

Figure 5.33 Valgus and varus displacement (deg) versus flexion angles (deg) of elbow joint in various conditions (Stavlas et al., 2007)…… 214

Figure 5.34 Moment arms plot of bicep muscle during flexion angles of elbow joint based on ten different specimens (Murray et al., 2002)…… 215

Figure A1Human skeletal bones and joints…….. 250

Figure A2.1 Cervical vertebrae (neck) ligaments, (a) Interspinous ligaments, (b) Flaval ligaments, (c) Facet joint capsule ligaments, (d) Longitudional ligaments (created in Lifemod software)…… 254

Figure A2.2 Cervical vertebrae (neck) muscles, (a) Semispinalis Cervicis muscles, (b) Trapezius muscles, (c) Longus Colli muscles, (d) Semispinalis Capitis muscles, (e) Sternocleidomastoid muscles (created in Lifemod software)…….. 259

Figure A2.3 Intervertebral discs between each cervical vertebra……. 260

Figure A2.4 Activation curve of cervical vertebrae (neck) muscles which shows activated muscles after the impact to the upper torso at 0.175 seconds (Lifemod software)…… 262

Figure A2.5 Extension (Left) neutral position (Middle) and flexion (Right) movements of head and neck…… 263

Figure A2.6 Lateral bending to the right of the cervical vertebra (Left), neutral position (Middle), lateral bending to the left of the cervical vertebrae (Right)……. 263

Figure A2.7 Axial rotation to the right of the cervical vertebra (Left), neutral position (Middle), axial rotation to the left of the cervical vertebrae (Right)…… 263

Figure A3.1 Major Flexor muscles (bicep brachii, brachioradialis, brachialis) of arm 266

Figure A3.2 Major Extensor muscles (triceps brachii, ancenous) of arm…… 266

Figure A3.3 Elbow joint major muscle moment arms during flexion movement from ten Specimen (Murray, Buchanan & Delp, 2002)……. 268

Figure A3.4 Elbow joint cartilages on articulating surfaces of humerus, ulna and radius (highlighted in red)…….. 269

Figure A3.5 Elbow joint extension (Left), neutral (Middle) and flexion (Right) movements…….270

Figure A3.6 Elbow joint, valgus (Left), neutral (Middle) and varus (Right) movements 270

Figure A3.7 Elbow joint supination (Left), neutral (Middle) and pronation (Right) movements…… 271

Figure A6.1 Screen shot of Geomagic Studio 9…….. 278

Figure A6.2 Disconnected surface point (red) elimination with Select Disconnected, Select Outliers and Reduce Noise functions…… 278

Figure A6.3 Wrapping function for triangulation and Fill Hole function for filling surface holes……. 279

Figure A6.4 Geomagic Studio 9; Fill Hole, Fill Partial, Create Bridge, Clean Up and Move functions. Triangles have been deleted and Create Bridge has been used to fill the defected surface…….. 279

Figure A6.5 Triangle reduction is performed for selected surface by using Decline Triangle function. In this figure 75,000 triangles are reduced into 7000 triangles with 6000 triangles of contact surfaces and 1000 triangles for bone shaft……. 280

Figure A6.6 Cervical vertebrae joints from Lifemod software (Left). Triangulation is performed by Refine Polygons function (upper) and surface defects are smoothed by Sandpaper function (lower) through Geomagic Studio 9. For more realistic and anatomic cervical column, C1 and C2 needed to be shifted (Right)……... 280

List of Tables

Table 5.1 Centre of mass directions of 9 masses (head and cervical vertebra bones) extracted from MJM database…….. 185

Table 5.2 Centre of mass directions under an arbitrary static force……. 186

Table 5.3 Centre of rotation due to static displacements……. 186

Table 5.4 Joint stiffness and associated results…… 188

Table 5.5 Mass deflections, centre of rotation and mount deflections at insertion points under static loads…….. 195

Table 5.6 Joint stiffness and related results…….. 196

Table 5.7 Centre of rotation of mass 2……. 198

Table 5.8 Valgus and varus displacements relative to elbow flexion angles; in vivo results (average of more than 30 patients + students)…….. 207

Table 5.9 Valgus and varus displacements relative to elbow flexion angles; in vitro results (single sample)…….. 208

Table 5.10 Frastum parameters and waist location for neutral elbow flexion (roller configuration A) compared with flexion under applied varus or valgus moments of 0.5 Nm (Bottlang et al., 2000)…… 211

Table 5.11 Summary of peak moment arm data of elbow (Murray et al., 2002) 215

Table A1.1 Major human skeletal joints and common joint mobility descriptions 251

Table A2.1 Range of movements of cervical vertebrae joints……. 252

Table A2.2 Origin and insertion points of cervical vertebrae (neck) ligaments.. 255

Table A2.3 Stiffness and damping parameters of cervical vertebrae (neck) ligaments 256

Table A2.4 Origin and insertion points of cervical vertebrae (neck) muscles…. 259

Table A2.5 Tensile failure stiffness parameters, compression failure stiffness parameters and area of each intervertebral disk along cervical vertebrae (Yoganandan et al., 2001) 261

Table A2.6 Tensile failure stiffness parameters, compression failure stiffness parameters and area of each intervertebral disk along cervical vertebrae (Moroney et al., 1999) 261

Table A 2.7 Medial-lateral shear stiffness, axial tension-compression stiffness, anterior posterior shear stiffness, lateral bending stiffness, flexion-extension stiffness and axial

rotation stiffness parameters of each intervertebral disk along cervical vertebrae (Lifemod software)…… 261

Table A 2.8 Medial-lateral shear damping, axial tension-compression damping, anterior posterior shear damping, lateral bending damping, flexion-extension damping and axial rotation damping parameters of each intervertebral disk along cervical vertebrae (Lifemod software)…… 262

Table A3.1 Valgus and varus movement range of elbow joint relative to flexion angles 264

Table A3.2 Origin and insertion points of elbow joint ligaments…… 265

Table A3.3 Stiffness and damping parameters of elbow joint ligaments (Regan et al., 1991)…… 265

Table A3.4 Origin and insertion points of arm flexor and extensor muscles….. 267

Table A3.5 Anthropometric parameters of arm flexor and extensor muscles….. 267

Table A3.6 Tendon displacement and moment arm estimations of arm flexor and extensor muscles…….. 267

Table A3.7 Average contact areas with standard deviations between olecranon part of ulna and trochlea part of humerus (ulna-humeral joint) during elbow flexion (In vitro, mm2)…… 269

Table A3.8 Average contact pressure with standard deviations between olecranon part of ulna and trochlea part of humerus (ulna-humeral joint) during elbow flexion (In vitro, MPa)…… 269

Abbreviations

AABB	Axis aligned bounding boxes
ADAMS	Automatic dynamic analysis of mechanical systems
AL	Annular ligament
ATP	Adenosine triphosphate
BVHs	Bounding volume hierarchies
C1	Cervical vertebra 1
C2	Cervical vertebra 2
C3	Cervical vertebra 3
C4	Cervical vertebra 4
C5	Cervical vertebra 5
C6	Cervical vertebra 6
C7	Cervical vertebra 7
CE	Contractile element
COG	Centre of gravity
COM	Centre of mass
COR	Centre of rotation
COS	Centre of stiffness
CPU	Central processing unit
CSG	Constructive solid geometries
CT	Computed tomography
DADS	Dynamic analysis and design software
DAE	Differential algebraic equations
DEA	Discrete element analysis
DH	Denavit-Hartenberg
DOF	Degree of freedom
ECRL	Extensor carpi radialis longus
EMG	Electromyography
FEA	Finite element analysis
FFD	Free form deformation
GAMMA	Geometric algorithms for modeling motion and animation
GJK	Gilbert, Johnson & Keerthi
GPU	Graphics processing unit
GUI	Graphical user interface
IDE	Integrated development environment
Iges	Initial graphics exchange specifications
ISB	International society of biomechanics
KDS	Kinetic data structure
LC	Linear complementarity
LCL	Lateral collateral ligament
LUCL	Lateral ulnar collateral ligament
MCL	Medial collateral ligament
MJM	Musculoskeletal joint modeler

MRI	Magnetic resonance imaging
MUCL	Medial ulnar collateral ligament
NC	Nonlinear complementarity
NJR	National joint registry
OBB	Oriented bounding boxes
OLE	Object linked embedded
PCSA	Physiological cross sectional area
PE	Elastic element in parallel
PF	Potentiometer for measuring flexion and extension movements of elbow
PR	Potentiometer for measuring rotational movements of elbow
RAPID	Robust and accurate polygon interface detection
RCL	Radial collateral ligament
ROM	Range of motion (or movement)
SDA	Screw displacement axis
SE	Elastic element in series
SGV	Strain gauge for measuring valgus and varus movements of elbow
SIMM	Software for interactive musculoskeletal modelling
SPH	Smoothed particle hydrodynamics
ST	Scapulothoracic
T1	Thoracic vertebra 1
TKR	Total knee replacement
UHMWPE	Ultra high molecular weight polyethylene
VB	Visual basic
VIMS	Virtual interactive musculoskeletal system
VRML	Virtual reality modelling language
VSR	Virtual soldier research
XML	Extensible markup language
1D	One dimensional
2D	Two dimensional
3D	Three dimensional

CHAPTER 1

INTRODUCTION

1.1 Need for accurate musculoskeletal joint modelling

It has been reported in the 5th Annual Report of National Joint Registry (NJR) on 31th March 2008 that in 2007/08, about 151,496, hip and knee joint replacement procedures were carried out in England and Wales in NHS and independent healthcare sector and about 583,724 total numbers of procedures were performed between 2003 and 2008. As the age of population in the western world gets older and as the life expectancy continues to rise, the need for joint replacement will also continue to increase. In addition to primary (first time a joint is replaced), the revision (repeated replacement) surgery will increase even further since increasing population with the replacement joint. The aging population is more prone to osteoarthritis, trauma, bone fracture and etc. Basically the prosthetic joint replacement is performed to recover the lost joint functionality. The joint replacement procedure has evolved since end of the 19th century. The most important factors involved in reconstructive surgery are the geometry of the prosthesis, type of prosthetic material and the positioning. In considering material types, since 1930s many material types such as stainless steel, titanium, vitalium, ceramic, platinum and polyethylene have been used in manufacturing of joint prosthesis (Pinchuk et al., 2005). The major problem with the prosthetic materials revealed with the accumulation of wear debries such as iron abscess. In addition to material based problems, replacement of a natural joint with an artificial prosthesis can kinematically alter the normal joint articulation due to modifying the natural centre of joint movements and forcing the joint to articulate relative to a fixed centre of movement. This is seen as the approximation of anatomical joints as idealised kinematic joints. Moreover, results from joint replacement also depend on the performance and capability of the surgeon which has high influence in positioning of prosthesis. However many of these issues still remain unsolved and there is a need for a more reliable joint replacement, less dependent on surgeon, with respect to right selection and right positioning of prosthesis which could restore the normal joint functionality. Most of the alteration and problems in joint replacement is due to the lack of scientific perspective of joint biomechanics. In considering prosthetic joint

loosening, from mechanical point of view, this is probably due to artificially modified joint kinematics (i.e. centre of movement and mobility) which yield further problems in tissue interactions, and articular contact which then result abnormal joint articulation and loosening. Specifically considering such prosthetic joint complications and loosening the major problem arise due to insufficient or high contact forces, relatively loose or tense ligaments, insufficient muscle force generation and overall problem in interaction between all these musculoskeletal tissues and the joint articulation mechanism. Thus, in order to increase the success in joint replacement creative approaches have to be taken into account. In most of the surgery planning, the main decision is made by a surgeon through assessing image data such as x-ray taken from the patient. Then during the operation, the alignment and positioning depend on the capability of the surgeon and guidance of people from the prosthesis manufacturing company through considering anatomical landmarks of bone segments such as approximate longitudinal axes in positioning the prosthesis. Through development of computational modelling of joints, it has become possible to understand the joint biomechanics which should guide any treatment as surgical or otherwise. However, due to impracticality of all these computational models as well as incapability of assessing joints in anatomic fidelity, decision making and reconstructive surgery still demands surgeons skill.

In computational joint modelling and joint biomechanics, kinematics, musculoskeletal tissue behaviour, articulate contact, friction, and lubrication are analysed. The most important objectives in analysing the joint biomechanics are to understand the normal, abnormal, traumatic and prosthetic joint mechanisms along with surrounding tissue and contact interactions. As mentioned above, artificial joint loosening is a frequently seen complication. For example in elbow joint reconstruction, due to using semi constraint or constraint hinge joint prosthesis yield high number of dissatisfaction (instability or muscle loss) and joint loosening due to altered joint. Another commong problem in joint replacement is the incompatibleness between the natural joint and the artificial joint in terms of material properties. During the last fifty years or more research has gone into understanding of bone and tissue, surrounding a joint and their mechanical behaviour and mechanical and biological compatability with the implant material. Starting from early days where implants has progressed becoming more and more biocompatible with less loosening or stress shielding. However, joint kinematics has

attracted much less attention compared to the biomaterials research, probably because motion kinematics was seen less important. Having a less natural mobility seems to be less life threatening than having an implant in body which can lead to body's rejection, infection or even cancer. Nevertheless, it became obvious that wrong kinematics does exactly the same thing by increasing friction and resulting a high wear debris, spread of implant molecules into body fluid or blood and local infection and may even lead to cancer. Attempts of understanding human body motion kinematics, is not new and gait analysis has been performed using specialised equipment and many analysis techniques had been developed. Analyis of human motion based on the musculoskeletal parameters and its dynamical equations are relatively new and generally based on modelling paradigms of mechanism modellers. Human body motion analysis, on these packages is treated as mechanism analysis with standard joints (spherical, revolute). There is a real need to define motion of musculoskeletal system based on the real joint geometry without any "standard joint" assumption. The research work presented here started with this intention of developing formulations and theories to facilitate this. Treating a joint as a standard joint, alters joint tendon, cartilage and muscle loading and results in the loss of muscles which become redundant.

1.2 Joint modelling background

In considering the kinematics of joints, anatomic joints except immovable joints (i.e. skull), are kinematically unconstrained joints and described by 6DOF mobility. The stability (and apparently reduced degree of freedom) of anatomic joints is maintained by strong ligaments and tendons and cartilage controlling the contact motion. The geometric cavity of joints can only contribute the joint stability but does not give full constraint in an anatomic joint mechanism. However for the sake of the simplicity, kinematic based simplifications have been performed through joint modelling applications. This simplification is based on assuming anatomic joints as simplified idealised kinematic joints with fixed axes (centre) of movements. Through these assumptions, the motion modelling and mobility descriptions are permanently reduced in equations of motion. This kinematic assumption is known as the bilateral joint constraint where the local contact geometry between articulating bodies are not involved into the analysis. On the other hand, through the unilateral contact based joint modelling, contact is included into the analysis. Through the unilateral based joint

modelling, mobility of articulating bodies are assumed as 6DOF and when the contact is active (detected by collision detection algorithms) the kinematic mobility of bodies are reduced to avoid interpenetration between bodies in contact (a rigid contact results in the reduction of DOF). In considering prosthetic joints, many types of prosthetic joint mechanisms are utilized. For example, prosthesis can be designed for surface reconstruction with no artificial constraints between articulating bodies, or can be designed as constrained where articulating bodies are constrained through introducing fixed centre of movement based idealised joints.

In terms of joints analysis, there are several approaches currently available, such as the FEA, the experimental based tests and the multi body dynamic modelling based techniques. The FEA analysis is one of the precise modelling techniques based on the continuum mechanics and its use requires a high level of expertise. It is rarely used by orthopaedic surgeons, physiotherapists or clinicians in order to assess the normal, pathologic and prosthetic joint mechanism. In experimental investigation of the musculoskeletal behaviour, the scope is limited and there are only a limited range of investigations which can be conducted on live patients. Although motion tracking is widely used, at which the skin movement obscures skeletal joint kinematics preventing detailed study. Nevertheless, some measurements such as motion tracking offers approximate information to identify gross deviation from the normal motion and generally used in the gait analysis. In multi body dynamic based modelling techniques, joints are either represented as idealised joint or contact based unilateral joints. Multibody modellers are probably more appropriate for modelling musculoskeletal structures compared to the FEA and they are more convenient compared to the experimental analysis. However they are not easy to use and the usage requires a high level skill and expertise. For example modelling a precise joint in the Lifemod software may take days rendering its use beyond very busy surgeons. Thus all these methods are not practical for surgeons to employ in order to make pre-surgery study.

In considering the musculoskeletal tissues around joints, many studies have been performed in terms of modelling tissue behaviour during joint articulation (i.e. tissue elongation) and more importantly, assessing the individual muscle force contributions to the joint torque. On the other hand in such applications, experimental tests are also performed. Due to the invasive nature of measuring tissue elongation or measuring EMG based muscle force generation, this kind of applications cannot be performed in

vivo and possible in vitro studies show erroneous results. However, through CT scan or MRI based geometric applications it is possible to understand the moment arm changing relative to tissue length change and joint angle changes. However these applications still lack geometrical and kinematics information in describing the whole joint articulation with respect to variational axes of movements. The tissue moment arm is very important in determining the joint motion, but the moment arm changes with the tissue wrapping bone surfaces or other tissues. The correct calculation of the moment arm necessitates the modelling of tissue paths in the presence of other tissues.

In prosthetic designs, the FEA is most commonly used. The use of the multi body dynamic software is rear. Although the FEA analysis have helped to develop implants with better stress distribution and better fatigue life, in real life applications the mechanics integrity is only one factor. The analysis of a prosthesis embedded in biological tissue and articulating against a cartilage requires accurate modelling of these. Currently there is no commercially available software which uses constitutive models of biological tissues. And simplified models of these tissues, likening them to mechanical elements (spring dashpots) to model viscoelastic behaviour, remain a very crude imitation of the reality.

Thus there is a need for a modelling tool which is capable of assessing and analysing anatomical joints, intuitive and easy to use for the surgeons. Such tool can be used to perform simple pre operation study, test different implants, and evaluate the effect of erroneous insertion in terms of joint mobility. The software can also be used with a surgical navigation or can be linked to robotic based surgery which can of course perform more accurate joint replacement.

Thus the project presented here aims to develop theories and formulations needed to drive such a system. The thesis shows that not only these theories are developed but prototype software was assembled.

1.3 Organisation of thesis

This thesis is composed of 7 chapters, the appendix and the references. Chapter 1 is the introduction chapter where the intension is to provide a background on the joint modelling and replacement. Chapter 2 is the literature chapter consists of three sections

as skeletal joint modelling, musculoskeletal tissue modelling and joint contact modelling. Moreover each section in the literature chapter consists of several subsections. Chapter 3 is the theory chapter at which the developed theories and formulations are addressed in three sections as unconstrained 6DOF joint modelling theory, development of geometric tissue path finding, line of action and wrapping algorithms, development of collision detection, contact and collision response algorithms. The developed theory and formulations which are addresses in the Chapter 3 are assembled as prototype software given in Chapter 4. Chapter 5, the results chapter, presents modelling applications, results and validation of the developed joint modelling software. In this chapter the results are compared against the Lifemod software, against published results and some experimental results carried out in the Brunel Orthopaedic Research and Learning Centre. The Chapter 6 is the discussion chapter. And finally the Chapter 7 presents the conclusions and the suggestions for the future work. In the Appendix, literature based parameters and results are placed. Additionally, the preparation of bone geometries and mesh generation process are included in the Appendix. Most of the joint and musculoskeletal tissue parameters and some standard formulations are also provided in the Appendix. The references section consists of around 300 references which have been utilized and studied in order to guide and construct the structure and the content of this thesis.

CHAPTER 2

LITERATURE SURVEY

2.1 Skeletal joint modelling

In this section a brief background to human skeletal joints and their modelling concepts are provided. The skeletal joints have long been approximated as idealised kinematic joints. With respect to these general approximations, human joint mobility (DOF) information is also studied. Following these idealised kinematic joint expressions, kinematic human upper limb movement and relative joint configuration based studies are addressed here. It is well acknowledged that by means of large scale dynamic joint analysis, wide range of research applications have been focused on rigid body bone assumptions. Thus, rigid body skeletal joint modelling techniques are presented. In order to describe joint articulations the intention was to focused on the multi body dynamics rather than the FEA. Therefore, the multi body dynamics based joint articulation modelling techniques were surveyed and studied. In the skeletal joint modelling section and in other sections, it is proposed to give brief introduction, background and subject specific literature review.

2.1.1 Structure and mobility of human skeletal joints

In musculoskeletal systems, the main function of skeletal bones is to provide the structural support to the body. Bone is a self-repairing and specialized supportive tissue which provides great rigidity, strength and durability against dynamic forces. The structural level of bone (Rho, Kuhn-Spearing & Zioupos, 1998) is composed of the cortical and cancellous bone levels, lamella layers, collagen fibres, collagen molecules, bone crystals and minerals. The overall shape of bone is represented with cortical and cancellous bone structures. The mechanical stability of bone is provided by the second cancellous level through osteons. The organization of osteons in the second level, affects the structural properties, shock absorption, and resistance function of bone. The combination of minerals and collagen fibres provides the hardness and resistance of bone. The most commonly faced bone disease is the osteoporosis, characterized by low bone mass and structural deterioration of bone tissue that can result possibly increasing

of bone fractures (Riggs & Melton, 1995). Usually extra care is taken when treating and reconstructing an osteoporotic bone fracture to avoid any further fracture and implant failure (Schneider, Goldhahn & Burckhardt, 2005). Nevertheless, excessive stress and dynamic loading can result in abnormal bone movements or deformations and damage the bone with musculoskeletal tissues (Section 2.2.1). In multi body musculoskeletal modelling applications, skeletal joints are assembled by two or more bones. The skeletal bone is usually modelled as a rigid body due to their very small bone deformations when comparing with musculoskeletal tissue deformations such as soft tissue and muscles. In order to express the joint articulation kinematics, each joint has been defined with its particular kinematic degrees of freedom. The kinematic based mobility of skeleton joints are classified in three groups as synarthroses (immovable), amphiarthroses (slightly movable) and diarthroses or diarthrodial (movable) joints (Saladin, 2007). The extended joint mobility information is given in Appendix A1. Skeletal joints are also classified based on the cartilage type which determines the type of contact between bone surfaces. This classification is referred to as structural classification concerns fibrous, cartilaginous and synovial type of joints (Moore, Dalley & Agur, 2006). Basically, bones in synarthroses type of joints are almost in direct contact and fastened together by intervening connective tissues. Skull is given as an example of synartroses joint type. In such joints, there is no appreciable movement between bones. However, synarthroses type joints exhibits high shock absorbing properties. Amphiarthroses (slightly movable) joints are also referred to as cartilaginous type of joints where bone surfaces are covered by fibrocartilage. Although amphiarthroses joints have more movement capability than synarthroses joints, they have limited mobility due to being surrounded by very strong tissues such as ligaments. Vertebral column is a typical example of amphiarthroses joint composition where fibocartilage adheres to the ends of vertebral bones through the column. Movement capabilities of diarthrodial joints are their distinguishing character from other type of joints. In biomechanical studies, extensive focus has been put on the modelling of diarthrodial (movable) joints which are of high degrees of freedom. Diarthrodial joints are also known as synovial joints, limb joints or insecure joints. Skeletal system consists of greater number of diarthrodial joints. In diarthrodial joints, end of the each union bone is covered with articular (hyaline) cartilage. Particular limb type of joints comprised of outer fibrous capsule and inner synovial membrane.

Based on the capability of joint movements an additional classification is performed for diarthrodial joints such as planar, uniaxial, biaxial and multi axial joint types. The planar (arthrodial) joints admit of a very moderate gliding movement and formed by the flat and small surfaces where one bone is slightly concave and the other one is slightly convex bone and gliding on each other. The amount of movement is limited by the surrounding ligaments. Planar joint articulation is seen between carpal and carpo-metacarpal bones, temporo-maxillary, sterno and acromio-clavicular, tarsal and meta-tarsal bones. On the other hand, the mobility of uniaxial (ginglimus or hinge) joints is defined with only rotational movements about a single axis. The most perfect form of uniaxial joint example has been given as the ulna-humeral joint at elbow joint complex. Moreover, rotatorius (pivot) joint is one of the uniaxial types of joints. Their mobility is limited to rotational movement about a single axis like pivot process turning within a ring perpendicular to the length of the bone. For example the radio-ulnar joint is formed partly by the lesser sigmoid cavity of the ulna where the neck of the radius rotates within the ring. Furthermore, biaxial (ellipsoid or condylar) joints allow translational movement in one plane such as varus and valgus movement during main flexion and extension movements. Especially knee joint, temporomandibular joint, radio-carpal joint and metacarpaphalangeal joint are considered as ellipsoid type of biaxial joints. Nevertheless some biaxial joints move with a small amount of rotation in addition to its flexion extension and abduction adduction movements. This is also called saddle type of biaxial joints where, carpa-metacarpal articulation of thumb is given as an example. Finally, multi axial (enarthrosis) joints are highly mobile joints such as ball-and-socket joints which perform rotational movement in all three axes. Hip and shoulder joints are in the form of multi axial joints. In such joints, one of the union bones has rounded head (ball) shape which fits into the other bone that has concavity (socket) shape.

2.1.2 Kinematic modelling of multi body and skeletal systems

A rigid body mechanism such as a robot is defined as linkage mechanisms with links interact and articulate with each other through joints. Kinematics analysis is used extensively in robotic field to understand the articulation and motion of jointed mechanisms. Typically, the structural body of robot and skeletal links (bones) are considered as rigid bodies which do not deform. It should be noted that, an

unconstrained rigid body consists of three degrees of translational freedom and three degrees of rotational freedom which can perform six degrees of freedom in total. In the modelling of robot and skeletal mechanisms, joints are usually represented as idealised kinematic joints which are constrained by means of number of degrees of constraints (Craig, 1989). In the previous Section 2.1.1, various idealized types of joints are mentioned as they are well acknowledged and adapted for representing the anatomic skeletal joints. Basically these joints are planar (translational), uniaxial (i.e. revolute, ginglimus or hinge, pivot), multi axial (i.e. spherical) joints. In this case, total degrees of freedom of a jointed mechanism decreases permanently with respect to kinematic joint constraints which is referred to as bilateral constraints. Thus, it is essential to define total degrees of freedom for each joint and the mechanism in order to construct its kinematic equations of motion. The kinematic modelling of skeletal mechanisms has been widely performed for representing joint articulations and relative body motion. Kinematic descriptions are utilized to establish the fundamental equations of motion for further dynamics and control analyses. Kinematic description of a mechanism is composed of the translational and/or rotational displacement, velocity and acceleration parameters. Once the mechanism is represented in terms of its kinematics, then it can be simulated both using the inverse and the forward kinematics and dynamics formulations. In order to control the motion, overall kinematic and dynamic analyses must be completed where the transformations between rigid body links can be managed by controlling the desired transformation during particular movements. Up to date, most of the kinematic mobility representations of skeletal joints have been adapted from idealised joints as mentioned above. In kinematic based studies, motion is described based on the geometry without concerning forces. Furthermore, in describing the kinematic motion of a mechanism, the first thing is to distinguish if the system consists of idealised joints or not. If a joint is not an idealised joint then its mobility definition in equations of motion might not be described as permanently constrained or bilaterally constrained joint. Thus, in equations of motion, the mobility of such joint can be represented as state dependent or unconstrained (6DOF). In terms of state dependent joint mobility, the state dependency comes from the contact constraints. This is also referred to as unilateral contact based constraints. In considering the bilateral and unilateral constraints, the obvious distinguishing factor is their kinematic mobility representations. On the other hand if a joint is not bilaterally nor unilaterally constrained, it might perform all translational and rotational movements in six degrees

of freedom. Thus, the kinematic joint equations of motion differs depending whether it is permanently constrained, unconstrained or dependently constrained. Moreover, in order to define the location and displacement of a system proper coordinate definitions are required. Various coordinate systems have been selected to represent skeletal body locations, translation and orientations (Zatsiorski, 1997). The definition of a coordinate system is based on accuracy of movement representation. For example in some applications, body or particular limb movements are defined with the reference frames fixed on the body. The cardinal planes composed of sagittal, transverse and frontal planes intersect at a particular point to represent relative limb and joint movements. The somatic reference system is also widely used to represent relative movements of skeletal body based on anatomic landmarks, centre of mass or principle axes of local body inertia. In considering the three dimensional orientations, the matrix method (Engin, 1980), Euler method and screw method as well as helical method is extensively employed. Due to the scope of this thesis, unconstrained three dimensional translations and rotations of skeletal bones and relative joint configurations are described by Euclidian space or coordinates R^3.

The kinematic analysis is classified into two categories as forward and inverse kinematics. Generally the position analysis is the main objective in forward kinematics. For example, the position of end-effector is obtained when the known kinematic parameters such as velocity and acceleration are written in kinematic equations of motion. The kinematic analysis is usually driven by popular Denavit-Hartenberg (DH) notation to establish the transformation matrix for each link in constrained systems (Denavit & Hartenberg, 1955). The DH notation has been a pioneering method in representing the transformation matrix of articulated systems. It particularly establishes the coordinate system in terms of four specific parameters among each articulation. Comprehensive explanation of the Denavit-Hartenberg notation can be found in many mechanics, robotics and engineering text books. In contrast to forward kinematics, in inverse kinematics, the position and orientation of the end-effector frame is given. Inverse kinematics is used in virtual animations of articulated bodies where the end-effector posture is satisfied through given positions and coordinates. However the given end-effector position can present ill conditions or undesirable link and joint movements. Thus in order to represent realistic postures and animations of articulated bodies, ill conditions must be taken into account and solved. In this case, methods such as

decoupling, inverse transformation and iterative methods can probably solve such inverse kinematic problems (Jazar, 2006). Moreover, the transformation matrix which is obtained from inverse kinematics can be compared with the transformation matrix given through forward kinematics. Therefore the error between transformation matrices which is usually called the residue δ (error) is minimized by using Jacobian matrices.

2.1.3 Kinematic modelling and joint motion measurement techniques: Application to upper limb

The kinematic modelling of joints permits to define the movement in the range of predicted kinematic degrees of freedom. In general, ulna-humeral joint is modelled as a hinge joint permits rotation about a single axis and performs one DOF. The ball and socket (spherical) joint permits three DOF and it is the most commonly used joint model to represent three dimensional shoulder and hip joint movements. In kinematic modelling of joints such as shoulder joint requires to handle the joint limits carefully. One of the earliest 3D kinematic models of human shoulder joint was proposed by Engin & Tumer, 1988, Tumer & Engin, 1989. In kinematic representation of the shoulder complex, humerus, clavicle, scapula and torso are assumed as rigid links interacting with idealized joints. These joints are described as 3DOF ball and socket joint consists of 1DOF sleeve joint with 2DOF universal joint. In these studies, the objective was to build a jointed mechanism of human shoulder complex where the rotational motion range of each joint is limited to sinus cone boundaries. Joint rotations were then calculated by the optimization criterion based on minimizing the error between reference position and performed movement. However their model has some limitations such as lack of measuring joint translations, inapplicability of analysing humeral axial rotations and neglecting scapular thoracic (ST) joint. The complexity of joint modelling arises with increasing the number of joints and limbs. In addition to describing the shoulder joint limits with sinus cone boundaries, the polynomial based surface regression fitting method was performed by Wang et al., in 1998. Furthermore, Maurel (1999) developed a specific shoulder joint complex. The kinematic bone data of upper body was extracted from medical images including soft tissues. In their case, soft tissues are also assumed as rigid bodies which are not allowed to deform during joint articulations. Bone configurations with idealised joints were driven from inverse kinematic algorithms to describe the end-effector posture through iterative applications.

For rotational range of motion, the use of sinus cone boundaries is further discussed by Maurel & Thalmann, (2000). In their human shoulder model, joint translations are assumed to be negligible when comparing with range of joint rotations. The difference between the Maurel`s shoulder models with other presented models is the addition of the scapula-thoracic (ST) joint model. More recently, a comprehensive human model "SANTOS" was released (Kim et al., 2006; Abdel-Malek et al., 2006). SANTOS consists of high DOF joint models constructed based on Denavit-Hartenberg notation. Thus every joint in SANTOS human model possesses fixed centre of rotations. Furthermore, Rab, Petuskey & Bagley, 2002 presented an upper body model consist of head, neck, shoulder, left upper arm, right upper arm, left and right lower arm with left and right hand and pelvis. All joints are modelled as idealised joints and assumed to have fixed centres of rotations. The experimental application was carried out with 3D video based motion capture technique and retro reflective skin markers. In 2005, Wu et al. published a comprehensive joint coordinate data for upper limb extremity consists of shoulder, elbow, wrist and hand. The work is entirely based on the kinematical data which has been presented by Wu et al., in 1995. In order to make the usage of kinematics easier and practical, the gleno-humeral joint is usually assumed as 3DOF ball and socket joint, ulna-humeral joint is assumed as 1DOF hinge joint and radio-ulnar joint is described as 1DOF pivot joint. In order to describe the relative linkage movements and joint motions bony landmarks, the centre of mass (COM) or principle axes of inertia are needed to define the coordinate system. However in most of the skeletal joint modelling applications natural translational movements and kinematic definition of unfixed centre of rotations are neglected. On the other hand, for the sake of simplicity, even in some detailed upper limb modelling studies, function of wrist joint and hand factor have been neglected to avoid the complicated coupled joint movement analysis. This elimination is sometimes valid due to the objective of study conducted which does not concern with wrist joint movement. On the other hand, studies which take functional kinematics of joints during daily activities (grasping, drinking, combing hair and etc.) must concern functional movement of wrist joint as well. Andel et al. (2008) proposed to define simple daily activities kinematically with using markers placed on hand, upper arm and acromium (to represent trunk and scapula). Local coordinate systems and joint rotations were defined based on the ISB (International Society of Biomechanics) standardization (Wu et al., 2005). The developed functional evaluation method is technically convenient in defining range of

motion (ROM) of upper extremity. The complete 3D kinematic evaluation of upper extremity was presented. It is also essential to measure the kinematic motion of human joints for diagnosis and healing purposes and also for designing implants and diagnostic devices. In general the kinematic motion analysis can be performed through applying a gross movement. In most cases, the gross movement is utilized for many applications; it is particularly useful for clinical applications such as diagnose many orthopaedic and neurological disorders. Additionally the evaluation of daily functional activities provides valuable information for diagnosis (Petuskey et al., 2007; Cooper et al., 1993; Magermans et al., 2005). Therefore experimental motion measurement applications are widely performed to investigate the kinematic concept of upper limb functionalities. For example, an analytic and experimental joint kinematic study has been performed by Kinzel, Hall & Hillberry (1972 I, II), as a part of the arthritis investigation in canine shoulder. The study was performed to measure the relative gross motion between scapula and humerus of a canine dog. Their application is based on the instrumented spatial linkage mechanism where the linkage was attached to the scapula and humerus to mimic the relative motion between bones. The plotted motion data provided the valuable insight to understand the relative motion between these two articulated bones. Nevertheless, the quantitative kinematic description of all diarthodial joints would guide orthopaedic diagnosis, treatment and joint replacement surgery. In sport activities, certain joint and tissue injuries occur. Thus, the motion analysis can provide information to understand the biomechanical concept of joint injuries and trauma along with passive and active motion measuring applications (Morrey & Chao, 1976) and detailed 3D biomechanical studies (Chao & Morrey, 1978). The study was a cadaveric based non-invasive diagnostic study where 3D rotational measurements of elbow joint are accounted under passive motions. It has been focused on measuring 3D rotational motion of the elbow joint without concerning the joint translations. Measuring of the functional relationship between the kinematical degrees of freedom of shoulder joint complex (Shoulder rhythm) has been also performed by many researchers. According to the studies performed by Hogfors et al., (1987, 1991) 12 DOF shoulder complex through 3DOF kinematic joints including muscles can provide acceptable shoulder rhythm information. In their proposed studies, tantalum balls were inserted under the skin around shoulder joints for motion capturing. The extended shoulder rhythm analysis was then carried to denote functional relationship between bones in shoulder joint complex. Such study provided acceptable information about shoulder kinematics

but the model could be investigated by adding translational joint movements too. Moreover, detailed investigation of joint kinematics and coordination has been carried by Dvir & Berme (1978) to measure the relevant body segment configurations of shoulder complex during arm elevation under various loading conditions. Through the development of various devices such as ganiometer, palpator, roentgen stereophotogrammetric devices and 3D motion recording cameras, the measurement of the kinematic orientations of upper and lower limbs has become more applicable. For daily activities, generally exoskeleton linkage mechanism is utilized and attached externally to the articulated limbs to perform rotational and translational measurements between the limbs. However most of the experimental devices have limitations in measuring joint functions accurately. In order to tackle the accuracy problems, marker based stereo-metric method has been utilized with little influence on the joint behaviour under investigation. Peterson & Palmerud (1996) attempted to use video-based stereo-metric system called MacReflex, in order to examine arm orientations. Results provided highly accurate rigid-body kinematics information. Further kinematic upper limb model has been presented by Fazel-Rezai, Shwedyk & Onyshko (1997). Their kinematical model is based on the camera based motion capture system (Safaee-Rad et al., 1990). Their upper limb model had 10 DOF which are: three rotational and three translational degrees of freedom for the arm with two rotational degrees of freedom for the forearm and two rotational degrees of freedom for the hand. The measurements have been performed during a drinking action from a cup and during feeding task. Instrumented 3D motion analysis systems support the quantitative assessment of diarthrodial joint motions. Only sophisticated measurement techniques can facilitate the unbiased evaluation and monitor 6DOF mobility of diarthrodial joints. Such a measurement technique was developed by Williams et al. (2006), to perform non-invasive marker-based motion analysis system and makes it possible to ascertain and document unconstraint kinematic functions of upper extremity joints. On the other hand some researchers had paid more attention to examine the changes of joint centre during particular motions. In addition to the importance of knowing kinematics of joints, the information about how centre (or axis) of elbow joint changes during motion revealed very important factor in biomechanics. The lack of precise modelling due to idealisation of the joint and resulting inaccurate knowledge of anatomical joint articulations can constitute problems in measuring accurate joint forces which is one of the main factors of understanding elbow implant loosening (Trail, Nuttall & Stanley,

1999). One of the earliest studies of (Morrey & Chao, 1976) focused on passive motion measurement and the definition of 3D rotational characteristics of elbow joint. In their study, the elbow is taken as a 2DOF joint which consists of 1DOF flexion-extension movement through ulna-humeral joint and 1DOF pronation-spunation movement through radio-ulnar joint. This study was the first study analysing 3D rotational characteristics of an elbow joint with a non-invasive measurement technique. In addition to their kinematic based study, it was pointed out that the centre of rotation of the elbow joint passes through the centre of the trochlea and the centre angle is changing linearly with respect to the rotational degrees of flexion-extension. In further studies, researchers aimed to determine the normal range of motion of forearm during flexion-extension and pronation-spunation movements. Youm et al. (1978) studied the elbow joint kinematics during various movements. Translational axis changing of ulna was measured which provided a valuable insight into designing the elbow implants. Their complete analytical and experimental results support the hypothesis of a fixed centre of rotation of elbow joint around the centre of the trochlea. The Reuleux`s technique was used which was able to define the radius and ulna movement about the distal end of the humerus while the elbow flexes (Eberharter & Ravani, 2006). From their cadaver based experiments, the instant rotational centre of ulna-humeral joint and radio-humeral joints was found at the centre of arc formed by trochlear sulcus and capitellum. Also the instant centres of both the radio-humeral and the ulna-humeral joints show that the elbow flexion occurs about an axis of rotation that passes through the centre of each arc formed by trochlear sulcus and the capitellum. The flexion-extension movement of elbow joint was presented as uniaxial (hinge) joint. In considering the shoulder joint, De Duca & Forrest (1973), pointed out that during shoulder abduction there are at least two separate centres of rotation at gleno-humeral joint. Additionally, Poppen & Walker (1976) presented a comprehensive study about normal and abnormal shoulder complex. The abnormal shoulder joint movements were subjected to several muscle and joint disorders. Veeger et al. in 1997 attempted to put more effort in quantitative descriptions of joint coordinates and centre of rotation of upper extremity joints. The centre of rotation of gleno-humeral joint has been examined with respect to ulna-humeral and radio-ulnar joint articulations. They found out that the centre of rotation of the gleno-humeral joint is very close to the geometric centre. In addition to kinematic descriptions, including 3D locations of muscle attachment sites, muscle length and volume information provided a precise analysis of shoulder complex.

From their point of view, it is reasonable to model the gleno-humeral joint as 3 DOF spherical joint with the fixed geometric centre of the sphere. Results presented a good agreement with the data provided by Poppen & Walker (1976). Furthermore, Veeger (2000) attempted to study the geometric and kinematic rotation centres of the gleno-humeral joint. The indication was to examine if the representation of the gleno-humeral joint as a spherical joint with the fixed spherical centre of rotation presents an acceptable results in shoulder joint kinematics. More specifically, they focused to find out if the geometric rotation centre of gleno-humeral joint could be represented as the centre of the sphere. They performed an in vitro study which comprised both the estimation of the kinematic and geometric centre of the gleno-humeral joint. Their results indicate that the gleno-humeral joint can be assumed as a 3 DOF spherical joint articulating through its geometric (spherical) centre of rotation. Further experimental studies have been performed in the Brunel Orthopaedic Research and Learning Centre, where the aimed to measure unconstrained elbow joint motion and unfixed centre of rotation. The measuring device was based on the Steward platform where the position and orientation of forearm is measured (Chapter 5). Experimental results which have been obtained, do not support the hypotheses that ulna-humeral joint centre passes through the trochlear sulcus during flexion-extension movement as the results show variational cetre (axis) of movements.

2.1.4 Dynamic modelling of multi body, skeletal and musculoskeletal systems

In order to drive physics based dynamic simulations of multi body systems, kinematic concepts such as mobility (DOF), displacement, velocity and acceleration are to be followed by kinetic concepts such as force, torque, centre of mass and moment of inertia to construct the dynamic equations of motion. Dynamics is the study of motion normally classified under kinematic and kinetic branches. Kinematic branch is mentioned in Section 2.1.2 where prior knowledge of motion exists and forces driving the motion are to be calculated (or not required). Kinetic branch is the study of motion which concerns the forces as a source of the motion. Thus, for rigid, flexible and hybrid multi body dynamic simulations, mass and inertia tensor and specifically for non-rigid systems, stress and strain tensors or stiffness and damping tensors are needed to construct the dynamic equations of motion. In this section, the main intention is to focus on the rigid multi body motions which only concerns mass and inertia tensors.

Due to inflexible nature of rigid bodies, parameters such as stiffness and damping are not involved into the equations. Thus the main attention is paid in constructing the mass and inertia tensors. The widely adopted inertia tensor calculation with through practical mathematical algorithm has been presented by Mirtich (1996a). The algorithm has been combined with the Gauss Divergence Theorem and adapted for polyhedral meshes. There are various theories and methods used to solve dynamic equations of motion such as Newton-Euler (Nikravesh, 1988), Lagrange (Asada & Slotine, 1986), D'Alembert principle, Gibbs-Appell, Featherstone algorithm (Featherstone, 1987), and Kane`s method (Kane & Levinson, 1985). These are well known methods and utilized to construct dynamic equations of motion. Vast numbers of literature and text books provide extensive background and discussions about these theories and their applicability. By means of the implementation of these theories and their numerical integration, Newton-Euler method is relatively the simplest method. Thus it has been extensively utilized for multi body dynamic equations of motion analysis (Flores et al., 2008).

2.1.5 Application of subsidiary and enabling theories in multi body simulations

The intention in this subsection is to present the state of the art in dynamic multi body modelling and simulation applications. The last four decades, many commercial and non-commercial software packages have been developed for dynamic multi body analysis and simulations. Based on the specific aim in modelling and simulation of multi body systems, adopted or developed theories and numerical methods vary. The development in computational power (Moravec, 1998), shows greater accomplishments in scientific applications. Thus, the fidelity of applying computational models yields cheaper virtual prototyping, manipulation, executable biomedical analysis, in robotics, ergonomics, and medicine as well as in sports applications. In computational biomechanics, multi body dynamics and the finite element analysis are essential tools. Such as in multi body dynamics, general objective is to simulate rigid and deformable bodies with respect to several dynamic theories, numerical methods and control theories. In early 1970s ADAMS (Automatic Dynamic Analysis of Mechanical Systems) and DADS (Dynamic Analysis and Design Software) software tools have been developed for general mechanical multi body design and simulations. The origin of ADAMS is the DRAM library which was developed by Chace (1978) for general

kinematic and dynamic analyse purposes and for articulated planar mechanisms. In order to handle large scale multi body system analysis, Orlandea et al. (1977) presented Lagrangian dynamics based solution for constrained multi body systems. Furthermore, Lifemod software released as a plug in to ADAMS for advanced dynamic modelling of musculoskeletal and its interacted environment. The Lifemod software (LIFEMOD) is widely used in orthopaedics, sports, and ergonomics applications. Dynamic simulations can be performed with utilizing inverse and forward dynamic theories and number of additional methods. In the Lifemod software, skeletal joints are modelled as idealised kinematic joints with fixed centre of rotations. In addition to idealised joint modelling, unilateral contact based joint constraints are also employed (Section 2.3). Thus, through constructing the unilateral joints, it is possible to obtain contact surface guided joint motion, however only for restricted applications and for particular geometries. In such cases, bodies are operating in reduced coordinate system rather than of full coordinates due to the mobility reduction when unilateral contact is active. On the other hand, SD/Fast (Symbolic Dynamics, Mountainview CA), a well known dynamic engine is driven by Kane`s dynamic equation of motion formulation. SD/Fast has been incorporated in to SIMM (Software for Interactive Musculoskeletal Modelling). The SIMM is an interactive computer graphics package, which has been developed (Delp et al., 1990; Delp & Loan, 2000) to model musculoskeletal bodies in order to analyse neurological, orthopaedic and surgical based concepts. Additionally, VIMS (Virtual Interactive Musculoskeletal System) software has been developed in order to provide a musculoskeletal modelling tool and to understand complex musculoskeletal disorders (Chao, 2003). This software is widely used in planning of total joint replacements and for joint simulation and rehabilitation purposes (Chao et al., 2007). The capability of the software provided tools for clinicians and researchers to study specific shoulder rhythms (Lin et al., 2005) and calculate joint contact stresses and evaluate external fixator effects to joint functionality. More recently, a graphics based software called MSMS has been developed by Davoodi & Loeb (2002). The software is capable of modelling and control musculoskeletal movements and provides comprehensive framework for virtual prototyping of prosthesis. For the purpose of the development of rehabilitation devices in orthopaedics, ergonomics and analysis of musculoskeletal disorders, “Anybody” software (Damsgaard et al., 2006) can be an appropriate choice. The “Anybody” software (ANYBODY) is based on dynamic theories and optimization principles that provide control over dynamic simulations.

FEA (Finite Element Analysis) is also essential in musculoskeletal modelling which is based on continuum mechanics. ABAQUS and ANSYS are best known FEA software packages. The advanced software MADYMO (MADYMO) is used for wide range of engineering applications. It consists of both the finite element analysis and the multi body dynamic methods. In the same way, RecurDyn (RECURDYN) is a general purpose advanced software which also combines the multi body dynamics and the FEA analysis. They are designed to handle constrained multi body mechanisms but they are also capable of simulating contact based articulated bodies. FEA is usually employed when the goal is to obtain the mechanical behaviour of materials such as stress distributions and strain. In FEA, bodies consist of finite elements and each element possesses material properties. Due to the general purpose in musculoskeletal modelling which is the performing large scale dynamic simulations with muscle and soft tissue interactions, applying FEA analysis is not practical. In FEA based modelling approaches, including very complex muscle, soft tissue and bone deformations into the equations of motions is very difficult, almost impossible. Both FEA and multi body dynamic modelling techniques can be employed to model musculoskeletal bodies but the complexity of handling continuum mechanics, makes FEA less desirable method in motion analysis.

2.1.6 Virtual human models

Virtual human models have been developed to perform and analyse specific human tasks based on multi body kinematics and dynamics techniques. In the early 1990s most of virtual human and animation systems were based on stick-figures or simplified articulated rigid bodies. One of the earliest developed virtual human models was Cyberman (Blakeley, 1980) developed by Cyrysler Corporation for analysing articulated limb locations in virtual environments like vehicles. Cyberman consists of 15 idealized kinematic joints and rigid skeleton links. Therefore, it has been possible to analyse activities of driver and passengers and provide particular anthropometric data in such applications. Combiman is a further developed virtual human model which was designed at Aerospace Medical Research Laboratory (Bapu et al., 1980). Combiman skeleton system possesses 35 articulated links proposed to examine joint movements, human reach capacity in aircraft cockpit. However these rigid virtual human models did not satisfy the realistic human motions and joint reactions in anatomic fidelity.

Furthermore, Zelter (1982) attempted to extend the stick-figured human models through computer programming advances to improve the reliability of virtual modelling. The main idea was to generate realistic kinematic analysis of motion through establishing mathematical model tree of joint transformations. However the inability of these systems in representing the flexibility of musculoskeletal structures has lead researchers to focus more on modelling of virtual humans in anatomic fidelity. Thalmann et al. (1995) and Thalmann & Cordier (2000) proposed to develop their own anatomical virtual human models including some of flexible bodies. Recently, a comprehensive study carried in the Virtual Soldier Research (VSR) laboratory at University of Iowa where the complete virtual human called SANTOS has been developed. SANTOS is the most sophisticated Denavit-Hartenberg notation based human model (Abdel Malek et al., 2006). Real time interactions and joint manipulations are essential capabilities provided with SANTOS. A detailed literature survey, most of the developed virtual humans and more comprehensive applications in virtual modelling are well presented by Thalmann`s book (Thalmann & Thalmann, 2004).

2.2 Musculoskeletal tissue modelling

This section provides an overview of fundamental modelling applications of musculoskeletal tissues. The first part gives a brief histological and mechanical background about modelling of musculoskeletal tissues. The musculoskeletal tissues are: skeletal muscle, tendon, ligament and cartilage. Then the attention is paid to explain how such flexible material deformations are modelled by following generic deformation modelling techniques. Therefore, comprehensive explanation of important aspects in characterization and modelling of musculoskeletal tissues is provided. Following the deformation modelling theories, particular musculoskeletal tissue modelling methods for the estimation of tissue moment arms, modelling of line of actions and tissue wrapping are provided with a detailed literature survey. After then the state of the art in physical and geometrical based deformation simulation techniques for musculoskeletal tissues and many other flexible materials are given.

2.2.1 Musculoskeletal tissues

In biomechanics field, most of the researchers have focused on the understanding of biomechanical behaviour of musculoskeletal tissues especially at diarthrodial joints due to their high degree of freedom functionalities (Section 2.1). It is well known that, musculoskeletal tissues are described as collagen fibred microstructures which provide impact resistant behaviour to protect body or other tissues. Considerable stiffness and damping properties are presented in collagen fibred musculoskeletal tissues. The existence of such properties yields tissues to dissipate the strain energy in to heat. While musculoskeletal tissues are under overloading or excessive stress conditions, they can be damaged and disrupted. When loads at diarthrodial joints exceed the joint strength level, musculoskeletal tissues especially tendon and ligaments which surround diarthrodial joints are stressed beyond their capacity that results tissue collagen failure (Viidik, 1990). Moreover, arthritis which is reported as the most common disease in elderly adults is the main reason of pain and abnormal joint functionalities due to ligament and/or cartilage defects (Brandt et al., 2006). Furthermore, in sport activities, knowing joint and tissue interactions holds the key in understanding injury mechanisms and yield proper orthopaedic treatment, rehabilitation, repairing and restoring actions. Biomechanical behaviour of musculoskeletal tissues involves measuring material and structural properties based on experimental and analytical techniques. In general, material properties of a particular body can be obtained by isolating it. In contrast, structural properties are obtained from structure or mechanism which composed set of interacting components. These material and structural properties are also referred to as mechanical properties. Stress-strain and force-deflection curves are essential to extract the mechanical properties such as young`s modulus, toughness, hardness, elasticity and strength ability to withstand variety of loads. These properties are essential in understanding, evaluation and characterization of materials as well as structures. Therefore, characterization is essential factor in material and structural modelling. Nevertheless, characterization and modelling of musculoskeletal tissues is a fundamental issue in representing their interactions and mechanical behaviours relative to joint articulations. Tissue properties have significant effect in correctly predicting tissue behaviour and joint loading during joint articulations. Thus, even for tissues, characteristics of which well known from experimental studies, their behaviour in a model can be realistic only when joints are anatomically realistic. For example it is

impossible to obtain accurate ligament loadings if elbow joint taken as a hinge joint. As mentioned in the previous section, the idealized joint intended to model anatomic joints neither can predict the realistic behaviour of joints nor loading of the surrounding musculoskeletal tissues.

2.2.1.1 Ligament and tendon

Ligaments are soft collagen fibred tissues; connect bones across skeletal joints. Their main functions are to stabilize joints, provide restriction to abnormal movements and maintain proper joint functions. Tendons are also defined as soft collagen fibred tissues which attach skeletal muscles to bone(s) and transmit forces across joints. By means of hierarchical structure and material properties, ligament and tendon possess similar characteristics (Cowin & Doty, 2007). Furthermore, many experimental techniques such as tensile tests have been performed to characterize mechanical behaviour of these tissues (Weiss & Gardiner, 2001). However due to the structural complexity of all human tissues, experimentally obtained data may exhibits lack of accuracy. For example, in tensile testing, in order to apply tensile force to tendon and ligament, individual tissue is usually fixed between rigid clamps that can introduce poor interference (Taylor, Duda & Heller, 2009). In addition to the tensile tests, optical and video based techniques have been performed to understand mechanical behaviour of ligament and tendon (Woo et al., 1983). Due to the excessive forces, collagen base failure of ligament and tendon has been also investigated by many researchers and recently a constitutive model has been proposed to describe particular ligament failures (De Vita & Slaughter, 2007). From many experimental investigations, material based stress-strain relation of these tissues has been defined as time and history dependent. In general, these characteristics are fundamentals to model musculoskeletal tissues. In considering ligament and tendon, their mechanical behaviour is reflected to collagen matrix structure which is responsible factor in their material characteristics and behaviour. Therefore, in terms of material based modelling, ligament and tendon are usually modelled as viscoelastic materials with respect to their time and history dependent behaviour. In addition to this, for the sake of the simplicity, in structural based joint analysis, they are usually modelled as elastic materials.

2.2.1.2 Cartilage

Cartilage is a living material and specialized type of dense connective tissue. In general, cartilage consists of chondrocytes and chondroblast cells, extracellular matrix and fluid. There are three different types of cartilage tissue that are: Articular (hyaline) cartilage, elastic cartilage and fibrous cartilage. Articular cartilage is most prominently found at diarthrodial joints, which covers contact surfaces of articlating bones. Approximately, articular cartilage consists of 70-85% water and electrolytes with 25-30% proteoglycans, glycoproteins and chondrocytes. Therefore water content of articular cartilage depends on the concentration of proteoglycans which also determines the swelling pressure (Mirzayan, 2006). It has been reported that due to arthritis disease, type II collagen degrades (Dodge & Poole, 1989) that cannot resist the swelling which results water increasing and ion exposing to draw in fluid (Poole et al., 2002). Generally, the articular cartilage structure is divided into four zones: 1) Tangential (superficial) zone 2) Middle zone 3) Deep zone and 4) Calcified zone (Cowin & Doty, 2006). Each zone contains different collagen fibril structure and different amount of proteoglycans. Main function of articular cartilage is to minimize occurred stress between bone contact surfaces along with fluid and solid phase interactions. Under stress and energy dissipation conditions, cartilage structure allows shielding collagen matrix from high stresses. Therefore fluid component provides major load bearing function and reduces the friction. In terms of mechanical properties, articular cartilage appears to exhibit inhomogeneous, anisotropic, viscoelastic and nonlinear characteristics (Mow & Guo, 2002). In addition to viscoelasticity, cartilage has been considered as poroelastic and biphasic material (Atkinson, Haut & Altiero, 1997; Setton, Zhu & Mow, 1993; Mow et al, 1980). Substantial works have been performed to determine and model tensile and compressive behaviour of cartilage during diarthroidal joint articulations (Huang et al., 2005; Mow, Ratcliffe & Poole, 1992). Due to compressive stress occur at diarthrodial joint contact where articular cartilages are formed, it is important to understand the compressive responses among them. When compressive stress occurs, cartilage fluid is forced out of the stressed area and response is provided by solid phase with proteoglycans. Hence, fluid flow dependent viscoelastic compressive responses have been accurately modelled based on biphasic and even triphasic theories (Mow et al., 1980; Hon et al., 1999). The hydraulic permeability relates to the rate of fluid forced out of the stressed area and the aggregate modulus

corresponds to the stiffness of the cartilage when all fluid flow is ceased (equilibrium). The hydraulic permeability and aggregate modulus are extracted parameters from viscoelastic responses that are important factors utilized in modelling of articular cartilages.

2.2.1.3 Muscle

In considering the musculoskeletal tissues, muscles are distinguished from other connective tissues (ligament, tendon, cartilage) due to their force generation capability. Muscles are classified into three main categories: 1) cardiac muscle 2) smooth muscle and 3) skeletal muscle. In this section, skeletal muscles are presented due to their contribution to joint articulations and body movements. The structure of skeletal muscle (Martini, 1998) consists of myofibrils, that every individual skeletal muscle has hundreds or thousands of myofibrils. A single myofibril is around 1 micrometer in diameter. Myofibril is surrounded by sarcomeres where sarcomere membrance and myofibril groups are surrounded by sacroplasmic reticulum containing multiple nucleus and mithochondria. Sarcolemma is another membrance which surrounds each muscle fibre (muscle cell). Muscle fibres are grouped into bundles, which is also called endomysium and they are surrounded by fascicle bundles. Connective tissue called prymysium surrounds fascicle groups and run blood vessels and nerves. The fibrous connective tissue called epimysium wrap around the skeletal muscle. The epimysium connects muscle to bone through tendon connective tissue where the connection depends on the skeletal muscle shape. The main function of skeletal muscle is to generate forces, maintain body position, support soft tissues as well as organs and keep body temperature in the normal range by contracting required energy in body. Skeletal muscles are attached to all over the skeletal system with tendons. When forces are generated by skeletal muscles, they are transferred through tendons to bones to produce skeletal movement and joint function. The force generation depends on force impulses that are naturally generated by the nervous system. Mechanical behaviour of skeletal muscle is mainly defined based on active (contraction) and passive (material) elements.

- Passive muscle properties

Passive muscle structure is considered as passive extensibility or passive flexibility of skeletal muscle. The resistance to passive extension in skeletal muscle is described as

muscle tone. Some works express the passive extension as muscle belly lengthening without concerning tendon compliance (Gajdosik, 2001). However more recent studies showed the influence of tendon unit in passive extension of some skeletal muscles (Herbert et al. 2002). Maximum passive extensibility is defined as the ability to yield maximum joint mobility which is important for athlete's performance (Dadebo, White & Geoerge, 2004). Difficulty of measuring active muscle tension in skeletal muscles, have moved researchers to focus on passive tension based measurements through plotting tension-length curves. These experiments provide passive tension and passive length extension information in total muscle tension generation. Thus, subtraction of passive tension from the total tension provides the magnitude of the active component. Furthermore, passive extensibility of skeletal muscle relates to its mechanical behaviour and characteristics. Like other musculoskeletal tissues, skeletal muscles present time and history dependent behaviour under tension and compression loads. Thus, in general, skeletal muscle has been modelled as a viscoelastic material (Bosboom et al., 2001). In addition to its viscoelastic characteristics, nonlinearity, transverse isotropic (Loocke, Lyons & Simms, 2006) and anisotropic (Loocke, Lyons & Simms, 2008) characteristics of skeletal muscle have been also reported.

- Active muscle properties

Active component is described based on passive muscle force-length characteritics as well as active muscle force-velocity characteristics and neural stimulation factors. According to the crossbridge theory (Martini, 1998) force generation starts with neural stimulation then continues with calcium releasing through sarcoplasmic reticulum. The calcium releasing binds troponin to change troponin-tropomysoin complex. After this change actin and myosin filaments start associating. While ATP hydrolization occur along with ATPase (sodium-potassium pump) in the myosin head, the energy is generated at that stage. Generated tension is proportional to the length of actin (thin filament) and myosin (thick filament) in muscle fibre. The first experimental work was performed by Gordon, Huxley & Julian (1966) to measure the tension-length relationship while tension is generated by a single muscle fibre at different sarcomere lengths. This length-tension curve is related with the generated tension in skeletal muscle which is described as a function of overlapping between actin and myosin filaments in sarcomere. On the other hand, muscle contraction is usually classified into

two modes; isometric (static) contraction mode and isotonic contraction mode depends on the performed task. During isometric muscle contraction, static position is obeyed where joint angle and muscle-tendon length remain almost constant but muscle shortens and tendon stretches. Isometric exercising which is also referred to as strength training, leads muscles to strengthen during static positioning. Sarcomere length-tension relationship during isometric exercises can be plotted through experiments. The isometric contraction is also illustrated by force-stiffness relationship. In isometric contraction, the total force generation is based on sarcomere lengthening and muscle shortening velocity. As the musle-tendon length is not changing, muscle contraction is mainly depends on sarcomere lengthening and muscle contraction along with time dependent activation constant. Thus this relationship in isometric contraction has been investigated with tremendous focus as the maximum force generation based on activation is depicted as parabolic curve (Bahler, 1968). In isotonic contraction, muscle-tendon is lengthening and shortening which is known as the dynamic contraction. Weight lifting, walking, running are isotonic exercises. There are two types of isotonic contraction so called concentric and eccentric contractions. For example, weight lifting exercise with dumbbells is a concentric contraction. While the weight is lifted, activated muscles are shortening and produce forces to counter act the external loads. Thus this relationship is based on the activation level of the muscle controlled by central nerve system. While the lifted dumbbells are being lowered, skeletal muscles are lengthening that is called eccentric contraction. McHugh et al., (2002) provide valuable information for difference between concentric and eccentric contractions of quadriceps at muscle fibre level.

2.2.2 Deformation modelling of musculoskeletal tissues

In order to model deformation of a material, dynamic equations of motion are set up. For rigid, flexible and hybrid multi body dynamic simulation in Euclidean space R^3, mass and inertia tensor are needed specifically and for flexible and hybrid systems, stress and strain tensor or stiffness and damping tensor are needed to set these equations of motion. Flexible body deformation modelling reveals fundamental importance in musculoskeletal modelling. It is well known that physical based mechanical behaviour of rigid or flexible bodies is represented by physical constitutive equations. In order to understand musculoskeletal tissue response to applied mechanical loads, many

constitutive equation based deformation models as well as supportive experimental techniques have been developed (Van Den Bogert, 1994). These constitutive equations are solved by common dynamic theories through finite element analysis (FEA) and discrete element analysis (DEA). Constitutive equations can reach to a very large number of equations to represent physical behaviour of a material/structure comprehensively. In this section a brief literature review will be given on various constitutive models describing the musculoskeletal tissue behaviours and deformations. The intention is to provide general engineering background on the deformation modelling of musculoskeletal tissues. It is well acknowledged that most of the musculoskeletal tissues as well as biological materials possess elastic and viscous characteristics. Nevertheless, directionality of material response is a significant factor in behaviour characterization. Thus, materials are assessed as isotropic or homogeneous if material response is same in all directions. Otherwise, if the response differs in different directions, it is referred to as anisotropic and non-homogenous material. Moreover, if the material response is same in two (orthogonal) planes, it is an orthogonal and non-homogeneous material. Therefore the directional dependency might be due to material, thermal, electrical and chemical properties. Going into more detail in description of material behaviours, it is convenient to revise several fundamental representations such as elasticity and viscosity. Basically, elasticity of a material is described by Hooke`s law and viscosity is described by Newton`s law. Hooke`s law provides mechanical constitutive equation that applies to represent elasticity of materials such as plastic, rubber, metal and biological tissues which yields to depict elastic stress-strain relations. When an elastic (solid) material is under loading condition, there is no energy (heat) dissipation as material returns its origin when the load is removed. If a material consists of both elastic and viscous (fluid) characteristics, when load is applied and removed, energy loss occurs due to the viscositic characteristics. Materials which exhibit both elastic and viscous characteristics are simply represented by the combination of Hookean and Newtonian constitutive equations. Therefore, the viscoelastic material modelling is the most commonly utilized material behaviour representation which consists of Hookean elastic and Newtonian viscous elements. Most commonly utilized linear visoelastic models (Fung, 1993) are Maxwell, Kelvin-Voigt, Standard linear solid, and Burger model. These models consist of Hookean elastic element which is represented by a spring $k(E)$ and Newtonian viscous element which is represented by a dashpot (η). The arrangement of these

elements varies and depends on the model. In considering Maxwell and Kelvin-Voigt models, spring stress is written by stress-strain relations with constitutive equations as:

$$\sigma = E\varepsilon \tag{2.1}$$

Where σ represents spring stress and ε represents strain. Young`s modulus of material is shown by *E*. The spring stiffness *k* in a force deflection relationship is a function the Young's Modulus. Stress in a dashpot element is proportional to time dependent strain given as:

$$\sigma = \eta \frac{d\varepsilon}{dt} \tag{2.2}$$

Where $\frac{d\varepsilon}{dt}$ shows the strain rate, and η represents dashpot viscosity. These stress-strain equations yield to establish equation of motion of material or structure. In considering flexible materials in Euclidian space R^3, elastic spring stiffness, viscous dashpot damping, stress and strain are represented by 3x3 tensors. If the concerned body is volumetric, then finite element (continuum mechanics) based representation accounts all individual mass particles (discretisized) of body. Thus, the stress (σ) can be represented by 3x3 Cauchy stress tensor (Ciarlet, 1987) that accounts all external forces acting per unit area of each mass particle. In discrete element analysis, the body is not composed of finite elements as it does not need to be discretisized into mass particles. Therefore, in discrete element analysis body is already composed of rigid mass elements.

Furthermore, the viscoelastic behaviour is characterised by creep, stress relaxation and hysteresis tests to obtain stress-strain relations. If a viscoelastic body is suddenly stressed with a constant stress and stress is maintained constant for a period of time this action deforms the body during a period of time. This kind of test is known as creep, where the measurement of strain versus time gives the creep response of the body. On the other hand, when a body is deformed with a constant strain for a period of time,

stress is changing with respect to time. In this case, the stress versus time curve shows the stress relaxation response. Loaded and unloaded component of stress-strain relationship provides the hysteresis response. These viscoelastic response experiments are used to obtain particular material properties from stress-strain or force-deformation curves. On the other hand, in order to obtain material characteristics (properties) through dynamic frequencies from performed tests (i.e. creep, stress relaxation) several signal processing methods can be performed such as Prony series (concern signals as decay functions). Let`s consider the stress relaxation test, the constitutive equation is written for the relation between constant strain and time dependent stress as:

$$\sigma(t) = Y(t)\varepsilon \tag{2.3}$$

Where time dependent stress σ (t) is proportional to time dependent Y(t) function. Y (t) is the stress relaxation function which is normally represented using the Prony series which gives the shear modulus as sum of decay exponentials. In other words it is possible to translate Y(t) into its frequency and decay components. The Prony series are important in extracting the material constants such as the natural frequencies, the damping ratios and amplitudes (Tschoegl, 1989). Moreover, under dynamic (periodic) and static (time-invariant) loads, nonlinearities exist in most of the engineering and biological material behaviours. Nonlinearity means that the stress and strain or the force and deformation relations are not linear. Probably the nonlinearity originates from intrinsic material properties at cellular (for biological materials) and molecular level as well as from geometric effects.

In some cases the nonlinearity of a structure is localised and associated with few modes and coordinates but the rest of the structure behaves linearly. For situations where non-linearity exist in directions with little influence on the motion under investigation, nonlinearity can be neglected or in other situation it may have to be tolerated simply because of lack of effective means to cope with it. Hence, it is impossible to take all nonlinearity into account during the inspection of the system. Indeed, in the identification of nonlinearity, if no conjunction can be achieved between theoretical and experimental results and if nonlinearity could not be characterised, it is better off

treating the system as linear. Existing models and theories to predict nonlinear viscoelastic behaviour have been comprehensively surveyed by Drapaca, Sivaloganathan & Tenti (2007). One of the qualitative studies was provided by Viidik & Ekholm (1968) to understand the mechanical behaviour of collagen fibred tissues. In their model elasticity of tissue material has been represented by nonlinear springs. Engin (2001) considered the nonlinear force-elongation relation for elastic ligaments in his knee model. Fung (1993) suggested that for small soft tissue deformations the linear viscoelastic models are sufficient however for finite or large deformations, nonlinear viscoelastic behaviour has to be taken into account. In addition to this, the first continuum mechanics based quasi-linear viscoelastic model has been developed by Fung (1993, 1981). Especially, Mooney-Rivlin and Ogden models (Ogden, 1984) are widely used strain energy based material models to predict stress-strain behaviour of incompressible nonlinear viscoelastic, elastic and hyperelastic materials such as rubber, soft tissue and brain tissue (Holzapel & Ogden, 2005). Further nonlinear and quasi-linear modelling applications have been provided by Funk et al. (2000) to predict biomechanical behaviour of injured ankle ligaments. In general soft tissue modelling, a comprehensive study which contains incompressibility, nonlinear and quasi-linear viscoelasticity has been provided with engineering text books. A detailed review in modelling of musculoskeletal tissues like ligament and tendon through finite element based applications, constitutive modelling methods and various experimental applications has been presented by Weiss & Gardiner (2001).

In addition to passive material deformation modelling literature, force generation has to be taken into account in order to model complete dynamic skeletal muscle tissues. The most popular Hill`s muscle model was introduced by Hill (1938). The model composed of three elements: elastic element (SE) in series with active contractile element (CE) that are in parallel with passive element (PE). Parallel passive element (PE) represents passive muscle-tendon mechanical behaviour. Elastic element in series (SE) represents sarcomere properties. And contractile element (CE) is referred to as the force actuator element. Hill based muscle modelling is the most popular phenomenological based approach which is practical and adequate for most applications. Hill muscle model have been almost always used to predict skeletal muscle behaviours in diarthroidal joints such as elbow (Kaufman, An & Chao, 1989), shoulder, wrist and ankle. Extending the Hill model, muscle contraction phenomenon have been investigated concerning muscle

and tendon architecture (Zajac, Topp & Stevenson, 1986; Zajac, 1989) resulting in a more sophisticated model. Other sophisticated models have been presented by Huxley (1969) and Hatze (1981). These models are mainly built up on cross-bridge theory (Section 2.2.1.3) and thermodynamic energy transfer in molecular level of muscle contraction. The important aspect of such models is accounting mechanical properties of muscle tissue such as permeability of elements in cross bridge activations. Neidhard-Doll et al. (2004) proposed a microscopic phenomenologic skeletal muscle model to predict isometric excitation-contraction process. Their contribution was the modelling of each process involving cross bridge or fibrous force generation such as calsium releasing and energy transformation.

2.2.3 Modelling of musculoskeletal tissue paths and line of actions

In addition to the modelling of biological tissue deformation, the description of moment arms and line of actions is important in musculoskeletal models. Simply, muscle forces are transmitted to bones and generate moments to move corresponding joints which is referred to as joint torque. The use of moment arm knowledge is highly important in medical diagnoses, surgical applications, pre and post processing of surgical treatments. For example, ligament and muscle-tendon moment arm knowledge yields to understand accurate loading and the coordinated range of joint articulations and help surgeons to plan for tissue reconstructive surgery or tissue transfer surgery for the best treatment to satisfy natural tissue length and resultant moment arms as similar as in normal joints. Without an accurate line of action knowledge and treatment, post operative limbs can exhibit abnormal joint articulations due to altered tissue length that can result in wrong moment arms, and inconvenient interaction with surrounded tissues and bone (Adams et al., 2007). Thus, it is essential to understand the concept of muscle-tendon deflection and change in moment arm relative to performed task (Krogt, Doorenbosch & Harlaar, 2007). Therefore, experimental and geometrical tissue length measurements, moment arm calculations as well as computational modelling techniques have been widely developed (Pandy, 1999; An et al., 1984). Having obtained tissue origin and insertion points with the determination of moment arms are used to provide kinetic insight into many joint articulation models. In considering moment arm estimation, the kinematic modelling of joints requires an accurate location of the centre of joint rotation in order to describe moments of tissue forces. The joint centre of movement definition is based

on the joint kinematic model. In bilateral constraint based kinematic joint models, centre of movement or rotation is fixed and remain the same during joint articulations. In despite of bilateral constraints, the unilateral contact based constraints are utilized where the joint centre of rotation is not fixed but the movement is constrained when contact is active. This factor restricts the 6DOF anatomic joint articulation motion analysis. As proposed in this thesis, totally unconstraint joint model should possess 6DOF to predict realistic joint articulations in anatomic fidelity. From mechanical point of view, if there is a force acting on a joint, it`s moment arm is the perpendicular distance from the force vector relative to a point. Thus, the description of this point depends on the performed kinematic joint model. As such, in bilateral joints this point is at the fixed centre of rotation. However in 6DOF anatomic joint models, this point is not fixed and move in space. Nevertheless, force acting on a joint can be converted into moment by multiplying it by its moment arm. Due to the difficulty of defining the free instantaneous centre of movement of a joint in the range of 6DOF, it is much easier to define fixed centre of movement to measure moment arms. Thus, the most of researchers applied bilateral constraint based kinematic modelling as a basis of estimating moment arms. To our knowledge there is only one sophisticated method (Marai et al., 2003; Marai, 2007) presents geometric contact based kinematic joint model which is utilized to estimate ligament length and moment arms according to performed task. However, in this particular application the interest was the modelling of ligament line of actions of distal radio-ulnar joint (DRUJ) without concerning tissue deformations. Rather, it is interesting to describe how moment arms and tissue lengths are changing along with muscle contractions and joint articulations. The most commonly performed moment arm measurement technique firstly presented by An et al. (1984) where the moment arm estimation is performed with respect to change in tissue length (excursion) and to change in joint angle (Chapter 3). This procedure is widely used to predict moment arms for finger joint, knee (Buford et al., 2001; Spoor et al., 1990), shoulder and elbow joints (Murray et al., 1998). In such applications, it is not required to define kinematic joint centre of rotation explicitly. However, if the moment arm estimation is proposed to link to the dynamic joint motion analysis, kinematic joint model is needed. On the other hand, experimental, geometry based moment arm measuring techniques such as biplanar x-rays, digitization techniques, magnetic resonance (MR) imaging and computed tomography (CT) scanning are utilized. Where, extracted musculoskeletal data is converted into 3D geometric mesh data through image

processing software tools (GEOMAGIC, MIMICS). The definition of appropriate origin and insertion points as well as the centre line of tissue paths yields to define interactive line of actions and measure moment arms relative to established joint kinematics (Buford & Anderson, 2006). Electromyography (EMG) technique is also utilized in order to measure muscle forces to describe their moment arms and moments. Thus, individual muscle force measurements can be performed through EMG experiments to understand each muscle moment contribution at joint torque (Laursen et al., 1998). In considering dynamic joint modelling, there is an important problem with the definition of line of actions that influence the moment arm estimations. First, it is the tissue path modelling along with origin and insertion points and the second is the path wrapping. In the modelling of line of actions or tissue paths, it is required to express the line as a shortest distance between its origin and the insertion points. Therefore, this approach is aiming to minimize the total potential energy of the system. In this field, the straight line modelling of tissue paths is very commonly performed application. In the straight line modelling, a musculoskeletal tissue path is represented by straight line(s) consists of fixed origin and insertion points. The straight line method with simplified deformation and wrapping capability is used in Lifemod software (LIFEMOD) where tissue deflection and wrapping is allowed in the range of the straight path rather than following realistic tissue paths. Thus musculoskeletal tissues such as deltoid muscles do not appropriately wrap within a curvature that highly effect the force vector representations. Other most commonly used geometric model is the centroid line model (An et al., 1981; Jensen & Davy, 1975) where tissue path is passing from centre of the tissue as a curved line via cross sectional centroid points between the origin and the insertion attachments. The centroid line model is relatively more substantial model than the straight line model where tissue orientation is more accurately represented. Deflection of a tissue path is simulated with the basis of centroid points rather than on a straight line. In considering some pioneering musculoskeletal modelling applications, Hogfors et al. (1987) attempted to predict shoulder muscle forces as a function of arm position and external loads in static and quasistatic conditions. Muscle and ligament line of actions are modelled as stretched strings lying on straight line between origin and insertion points. Bone movements are defined with respect to constrained kinematic joint coordinates. The musculoskeletal model composed of more than 30 soft tissue and muscles wrap on bones along with geometrical constraints (sphere, cylinder). Furthermore, Bassett et al. (1990) investigated three dimensional orientations of

muscles during abduction movement of shoulder through computer-assisted analysis. However Bassett`s study (1990) and other earlier shoulder joint modelling applications (Poppen & Walker, 1976; De Duca & Forrest, 1973) haven't been assessed as comprehensive shoulder models due to large kinematic joint simplifications and difficulty in predicting accurate soft tissue and muscle forces. Thus, Raikova (1992) aimed to tackle such problems in prediction of muscle and joint forces on human upper limb. Approximately 30 upper limb muscles have been accounted and their action lines were modelled as straight lines. Additionally, Karlsson & Peterson (1992) attempted to test whether their purposed 3D shoulder joint complex model could be utilized to predict all the internal musculoskeletal forces acting on humerus in different loading conditions and movements. Arm elevation with hand loading (10N) was considered. During arm evaluation, extracted muscle forces around glenohumeral joint showed concordance with the data base provided by Inman, Saunders & Abbott (1944). The agreement was achieved in presenting 150N of muscle force reactions to 10N hand load during the abduction movement. Van Der Helm et al. (1992) studied the finite element based, dynamic model of shoulder mechanism. Three main bones at shoulder joint complex (scapula, clavicula, humerus) have been modelled as rigid beams and muscles have been modelled by one or more active-truss or curved-truss elements. In their performed study, they were able to measure muscle moment arms and mechanical behaviour of muscles during various joint articulations. Moreover, a major accomplishment was achieved by Van Der Helm (1994b) with developing the complete finite element based dynamic model of human shoulder complex. They provided a valuable insight with confirming that moment arms change with respect to rotational movement and emphasized the effect of modelling kinematically constraint joints in predicting moment arms. However, their finite element based results didn't show concordance with EMG results due to the possibility of poor force measures from EMG experiments. Furthermore, Van Der Helm (1994a) attempted to analyse both kinematic and dynamic behaviour of shoulder mechanism specifically for the humeral abduction under loading and unloading conditions. The developed model provided good insight into shoulder biomechanics within functional analysis of morphological structures. Similarly with their previous studies, some of shoulder ligaments were modelled as flexible truss elements and muscles were modelled as active truss and curve truss elements with respect to their PCSA (Physiological cross sectional area). The difference between truss and curved truss elements is that the truss elements are

represented as straight lines but the curved truss elements are allowed to wrap around bony or other soft tissue structures. These structures are simple geometric shapes (sphere, cylinder). Dynamic model of shoulder has been also studied by Wuelker et al. (1995) which was the further dynamic investigation of the shoulder mechanism after Van Der Helm`s applications (1994a; 1994b). Hogfors, Karlsson & Peterson (1995) extended the previously presented shoulder joint model which was published previously (Hogfors et al., 1987). The new model was proposed to be used in muscle force predictions within shoulder rhythm. The model utilized to predict internal muscle forces as well as ligament tensions along with constraint kinematic shoulder joint movements. Siemienski et al. (1995) showed the validation of muscle force predictions with EMG experiments which had presented some problems in previous studies (Van Der Helm et al., 1994b). They predicted upper arm muscle forces under loading conditions. In every particular arm positions, shoulder muscle forces were plotted with respect to the applied loads. According to their experiments, the joint force decreasing is assumed to be related to smaller muscle moments acting on elbow and shoulder joints. In addition to this, they have depicted different loading effects on the gleno-humeral joint contact area and the joint forces. Murray, Delp & Buchanan (1995) aimed to determine if muscle moment arms around the elbow joint vary as a function of elbow flexion-extension and the pronation-spunation movements. Moment arms of five elbow muscles as biceps, brachialis, brachioradilis, pronator teres and triceps have been obtained from the tissue excursion measurements according to joint angles (An et al., 1984). In their application, ulna-humeral joint is a hinge joint which can only rotate about a single axis with a fixed centre of rotation. As one of the common assumptions, this axis is taken which is passing through the centres of the capitulum and trochlear groove. They utilized the SIMM software (Section 2.1.5) to model the elbow joint with surrounding muscles to predict muscle forces and moment arms relative to the joint movements. Their results showed that during flexion of elbow, muscle moment arms increased. The further extended study had been presented by Murray et al. (1998), in order to create a “specimen-specific” kinematic model of the elbow joint along with surrounded musculoskeletal data from MR images. They intended to determine elbow muscle moment arms with respect to flexion and extension movements and clarify the functional capacity of muscles and their peak moment arms (Murray, Buchanan & Delp, 2002). Muscle line of actions was represented as set of line segments with series of points passing through muscle centroid points. Moment arm estimation was based on

the derivation of muscle-tendon excursion with respect to joint angle (An et al., 1984). Thus tendon displacement measurements were performed experimentally using position transducers. In this field more comprehensive computational models of musculoskeletal bodies have appeared with more sophisticated tissue wrapping applications. Nevertheless the accuracy of deflection simulation of tissues mainly depends on performed tissue wrapping model which defines the dynamic points along the tissue path. In considering the tissue wrapping, some problems arise in representing tissue constraining geometries. In this case wrapping on multiple anatomic structures must be considered. In order to represent geometric contraints, simple geometric shapes such as sphere and cylinder are used. These geometric shapes are also referred to as obstacles. The obstacle avoidance, shortest path finding and wrapping are problems originate from robotics, graphics, and the computational geometry and network optimization areas. As there is no unique analytic solution to such geometric problems, heuristic approaches and algorithmic solutions have been developed. However it is no more than the last three decades that such problems and techniques found their way into musculoskeletal modelling. In earlier musculoskeletal modelling applications such as in upper limb modelling (Delp & Loan, 1995, Van Der Helm, 1994a, 1994b), single obstacle has been employed as constraining surface. More recently much attention was paid in the shortest path finding and wrapping algorithm developments, which are all performed based on simple geometric obstacles (Buford & Anderson, 2006). The preliminary study provided by Garner & Pandy (2000) with their human upper limb model where the obstacle-set method was first introduced. In their study, tissue paths wrap on a single obstacle (cylinder or sphere) with series of straight and curved line segments. These segments are joined together via fixed and movable dynamic points (Carman & Milburn, 2005) to describe the mobility of tissues during joint movements. More specifically, Charlton & Johnson (2001) discussed analytical and numerical solutions for tissue wrapping among multiple constraining objects. Furthermore, Gao et al. (2002) proposed to model muscle wrapping with realistic bone and tissue surfaces as well as simple geometries. In their specific joint modelling application, muscles are modelled as deformable strings wrapped around bony geometries. The contact between bone geometry and muscle path is considered as frictionless. However, predicted moment arms through bone geometry based wrapping model did not provide good concordance with similar applications. In one of the recently performed studies, Audenaert & Audenaert (2008) proposed an optimal wrapping method on spherical and

cylindrical geometric shapes to obtain all possible shortest paths wrapped around these geometric surfaces. The global optimization method has been applied to construct the shoulder deltoid muscles wrapping on spherical and cylindrical surfaces. Then deltoid muscle moment arm predictions have been compared with moment arm data presented by Garner & Pandy (2000) which have been obtained from their obstacle-set method. Thus, considerable agreement has been achieved in describing muscle length changing, moment arms and wrapping radii between these two methods. Additionally, a novel multi-object wrapping method has been developed to represent tissue wrapping passing all connected obstacles through employing the analytic theory of geodesics (Marsden & Swailes, 2008). In the geodesic method, path is described as part of arc and helix (referred to as geodesic) wrap around a sphere and cylinder. The path is through interconnected dynamic centroid points. In this analytic method, friction is neglected and standard energy minimization approach is used. It has been possible to obtain the unique path which possesses the minimum energy from geodesic approach. Further experimental based study performed by Ackland et al. 2008 who proposed to determine instantaneous muscle moment arms surrounding glenohumeral joint during abduction and flexion range of movement in vitro. In their study, moment arm estimations are based on the common tendon-excursion method (An et al., 1984; Pandy, 1999). They provided more insight into shoulder abduction and forward flexion movements performed in everyday daily tasks such as lifting and pushing. In this area, a survey has been presented by Gatti et al. (2007) who proposed to compare existing moment arm measurement and modelling methods for shoulder rotator cuff muscles during arm elevation.

2.2.4 Deformation simulation of flexible bodies

In this section brief information about how flexible objects are simulated will be provided. As mentioned in previous sections (Section 2.2.2), physical based modelling of musculoskeletal tissue deformations is based on describing its equations of motion. For deformation modelling a set of constitutive equations are utilized. For example, these constitutive equations can represent stress-strain dependent behaviour. However, sophistication in describing these constitutive equations can affect the simulation performance. As the real time simulations are essential in virtual surgery, computer games, force feedback and haptic applications, simulation time is the most important

factor. Through recent advances in computer hardware developments (in particular, graphics processors), it has been possible to simulate musculoskeletal tissues, organs, skin, cloth, fire, waterfalls and many elastic and plastic materials through employing physical or pure geometrical based deformation simulation methods (Gibson & Mirtich, 1997). Therefore, simulation procedures and applied techniques in physical and pure geometrical based methods vary widely depending on the scope of the simulation. For the sake of the simplicity, most commonly performed methods are briefly introduced in this thesis. For more wider survey, Thalmann`s book (Thalmann & Thalmann, 2004) is recommended.

2.2.4.1 Geometry based deformation simulation methods

In deformation simulation area, pure geometric methods have been widely developed. If physical representations of objects are not strictly required, applying geometric based deformation methods can fit better into the simulation due to its efficiency. The advantage of geometry based simulation methods, is not being based on the physical equations of motion, they do not suffer from numerical instability. Most commonly, in geometry based methods, surface geometry is manipulated by controlling surface points on NURBS and B-spline curves. Usually geometric surfaces are represented by B-splines and NURBS as it is easier to control patches along with curved control points rather than triangled mesh surfaces. Basically, spline surface composed set of control points through spline curves on the surface. For example Bezier spline representations have been widely utilized which consists of series of Bezier curves (Bartels, Beatty & Barsky, 1987). This representation is simply used in computer graphics to create smooth curves for smooth solid surface models. Free form deformation (FFD) is one of the most commonly performed geometric based simulation methods. The free form deformation is applied through key-frame interpolation of surface control points. In one of the earlier studies, Barr (1984) represented objects in hierarchical forms to perform small component based free form deformations. Object deformation is simply expressed as the deformation function “F” between the un-deformed solid surface points “x” and deformed solid surface points “X”. The method was also used in the Boolean operations, twisting, tapering and similar geometric transformations. Furthermore Sederberg & Parry (1986) proposed more general applications in this field through

employing the FFD. In their method, free form deformation is defined by trivariate Bernstein polynomials, where arbitrary deformations are performed through moving the control points on parametric surfaces, planes and quadrics. Coquillat (1990) extended the FFD method which was previously presented by Sederberg & Parry (1986) with purposing a different approach. In their study, 3D lattices are used to define the shape deformations. The body is either represented as parametric surfaces (B-spline) or triangulated meshes depend on the scope of the simulation. Moreover, the implicit surface based techniques are also utilized to simulate complex shape deformations. Implicit surfaces are created by simple block functions so called primitives. Thus, blending the primitive of an object provides the resultant deformation. Most commonly utilized implicit surfaces are metaballs (Nishimura et al., 1985), blobs (Blinn, 1982) and similar type of objects (Wyvill, McPheter & Wyvill, 1986). In implicit surface based techniques, the geometric representation is similar within the FFD method, where the implicit surfaces are defined by points and each point is expressed by field function such as $F(x,y,z)$. Implicit surface based technique has been applied to build the virtual ballerina (Yoshimoto, 1992). This method has been also utilized to model bone and muscles in human (Shen & Thalmann, 1995) and animal musculoskeletal body modelling (Wilhelms, 1994; Wilhelms & Van Gelder, 1997). In the modelling of similar anatomical bodies, ellipsoids have been used to model muscle deformations (Scheepers et al., 1997) and jointed skeletal animations. Beside the implicit surface (cylinders, spheres, ellipsoids) based applications, energy constraint based methods (Witkin, Fleischer & Barr, 1987) have been performed for geometric deformations. In such applications geometric constraints are expressed as energy functions in deformation equations. Summation of energy functions leads to move the object in a defined parameter space where the minimum energy is obeyed during parametric movement. Furthermore, interactive axial deformation method has been used by several researchers (Lazarus, Coquillart & Jancene, 1993) to perform the geometric body deformations. In this approach each point on an object is defined by its local coordinates which correspond to each surface vertex coordinate. Thus, the deformation of a body is associated with the change in the local coordinates. The deformation is independent of the geometric shape where the deformation on each local coordinate is transformed to the global coordinate system to define the resultant object deformations.

2.2.4.2 Physics based deformation simulation methods

First of all, the physical based representation of objects depends on the scope of the simulation. Objects can be represented by mass particles, mass-spring or mass-spring-dashpot elements. From physical point of view, materials composed of atoms and molecules. In computational simulations, the smallest element of a solid and liquid material is usually represented by mass particles. In particular simulations of natural phenomena such as smoke, fire, waterfalls and explosion, particle based representations are essential. Interacting particles are represented as attached mass-spring and mass-spring-dashpot elements. Such representations are widely performed to simulate the physical based deformation of musculoskeletal tissues (Ganovelli, Dingliana & O`Sullivan, 2000) and many engineering materials in virtual environment. As mentioned in Section 2.2.2 spring and dashpot elements represent the elasticity and viscosity characteristics of a material. Thus, configuration and quantity of mass-spring-dashpot elements as well as their dimensional representations (e.g. 1D, 2D, 3D) depend on the proposed simulation. However in the case of the finite element analysis, body is firstly composed of finite elements. Thus, in order to perform physical based deformation simulations, geometric (spatial) discretization is applied to describe the body in terms of nodal mass elements. Then equations of motion can be constructed through constitutive equations. Moreover, in the FEA, the constitutive equations are introduced based on continuum mechanics through describing stress-strain relations of elements. Through applying the geometric discretization (e.g. Finite differences method and finite element method) the constitutive equations represent the body based on nodal mass particles. In this case each nodal mass motion can be represented by force-deflection based equations. The other important issue is the representation of the time dependent change of body position and velocities. The time dependent solution of equations of motion is constructed by replacing differential equations with algebraic equations for numerical iterative solution. In this case, stiff system problem and instability problems arise which are handled by temporal discretization (time discretization) of the system equations. The temporal discretization applies very small time steps in numerical solutions to avoid the numerical problems. Furthermore, topological (meshed) based deformation simulation of bodies along with mass-spring elements can take excessive amount of computer power and time. Thus, in real time virtual surgery simulations, organs and tissues are modelled through meshless mass-spring elements to perform real time feedback for cutting and interventions without

concerning real physical interactions. However, in some mass-spring element based modelling applications accurate physical behaviour is essential where topological body representation is preferred to maintain the real physical based behaviours. In this field, Terzopoulos, Platt & Barr (1987), and Terzopoulos & Fleischer (1988), introduced most popular physical based deformation modelling methods with applying the Lagrange formulation through finite differences. In their work, advanced inelastic deformation modelling has been applied to simulate viscoelasticity, plasticity and fracture. Later, linear Lagrange formalism was adopted by Celniker & Gossard (1991) for craniofacial and maxillofacial surgery simulations. Miller (1988) applied mass-spring based mesh representations to model deformable dynamic behaviour of worms and snakes. Moreover, modal analysis approaches have been also introduced (Pentland & Williams, 1989) in order to improve the performance of physical deformation simulations. The development of hybrid model has yielded to perform deformation simulations along with utilizing modal vibration mode based shape deformations (Sclaroff & Pentland, 1991). Further physical based deformation modelling proposed by Essa, Sclaroff & Pentland (1992), where geometric surface has been represented by superquadratic functions (parametric shape function for 2D parametric curves) and its physical vibration (modal) modes. The parametric implicit (inside-outside) function has been created then substituted in to polynomial deformation functions (vibration mode based shape functions) to describe physical body deformation. Additional modal analysis based geometric deformation method has been proposed by Muller et al. (2005) to perform meshless body deformations. In the presented work by Hieber, Walther & Koumoutsakos (2004), solid body organs, human kidney and liver have been modelled as linear viscoelastic materials. The proposed virtual surgery simulations were based on interacting particles (mass-spring elements) through representing meshless body surface through applying particle level set method and smoothed particle hydrodynamics (SPH) method. Furthermore, Hieber & Koumoutsakos (2008) extended the smoothed particle hydrodynamics (SPH) method and applied for elastic soft tissue deformations through applying dynamic Lagrange formulations. Most commonly, interacting particles (mass-spring) and smoothed particle hydrodynamics (SPH) are used for meshless simulation of fluids (Muller, Charypar & Gross, 2003). Kuhnapfel, Cakmak & Maab (2000) presented tissue deformations in virtual endoscopic surgery applications. In their work, mass-spring-dashpot elements are used to represent deformable tissues. Their proposed virtual deformation simulation method have been implemented and performed through

applying the "KISMET" software. Mass-spring based modelling methods have been also widely used in facial tissue deformation simulations. Terzopoulos & Waters (1990) modelled facial tissues with springs and layered spring representations for facial muscle, skin and subcutaneous tissue simulations. Wu, Thalmann & Thalmann (1995) presented a dynamic model of facial expressions and facial tissue based ageing phenomenon. The facial tissue and skin deformations have been modelled through accounting the muscle contractions and modelling the muscle, fat and skin tissues by attached springs. Further modelling application for facial behaviours was performed by Nedel & Thalmann (1998) who applied muscle volumetric deformation with modelling tissues as angular springs attached between mass particles with concerning tissue paths and line of actions. Mass-spring systems have also been used for cloth simulations (Provot, 1995; Volino, Courchesne & Thalmann, 1995). Additionally, in dynamic musculoskeletal modelling, Chen & Zelter (1992) have developed linear FEA based hybrid soft tissue deformation model. The method was applied to simulate skeletal muscle deformation during various joint articulations with the addition of skeletal muscle contraction model (Zajac, Topp & Stevenson, 1986; Zajac, 1989). The material of the skeletal muscle tissue was assumed as an incompressible, homogeneous and isotropic material. The particular study performed to model bicep muscle contraction and shortening behaviour during elbow flexion movement through inverse kinematics. Additionally, Zhu, Chen & Kaufman (1998) modelled the human anconeus muscle deformation using FEM with 3D geometric brick elements. Muscle contraction model has been adopted from Zajac`s muscle model (Zajac, Topp & Stevenson, 1986; Zajac, 1989). In addition to linear based modelling of muscle mechanical behaviours, Hirota et al. (2001) concerned geometric and material nonlinearities in his FE model. In this case, tedrahedral elements have been used to perform non-linear volumetric body deformation analysis. More recently, typical nonlinear dynamic analysis through concerning nonlinearity and anisotropy of biological materials (i.e. skeletal muscle) with the addition of geometric nonlinearities has been handled by Lemos et al. (2001) through applying FEA based linearization of the equations of motion similarly performed by Hirota et al. (2001). Recently, Lemos et al. (2005) improved their muscle deformation modelling approaches with focusing on more anatomic and physiological aspects of skeletal muscles on human lower limb through employing the FEA. The muscle contraction modelling yielded to control muscle deformation and validated with some experimental data.

2.3 Joint contact modelling and collision

The purpose of this section is to provide literature survey on the joint contact in diarthrodial joint modelling. The intension is to emphasize the importance of the contact analysis in the multi body modelling of diarthrodial joints and its significance in accurate prediction of joint torque and muscle forces. In this section various type of geometric and deformation based contact modelling methods as well as specific diarthrodial joint contact models will be introduced. In processing the contact problem in computational simulations the collision has to be detected and response to be calculated. Thus, through the scope of multi body modelling and simulations, collision detection and collision response problems have to be handled carefully. It is well acknowledged that the contact problem is an extensively studied research area especially in the field of multi body modelling in robotics and biomechanics. In the biomechanics, the modelling of diarthrodial joint contact has been shown to be very important in accurate definition of joint articulations. For example, any musculoskeletal tissue and bone disorder affects the overall joint functionality along with abnormal compressive loading transfer to joint contact surfaces. Thus, only a complete and precise anatomic joint modelling including realistic kinematic joint mobility, anatomic based musculoskeletal tissue and contact models can bring valuable insight in to the joint biomechanics if all these elements are integrated to work together. By means of a respective work, it is intended to highlight the importance of each modelling phenomenon (i.e. tissue wrapping, tissue deformation, cartilage contact) in predicting reliable joint articulation model. Briefly, the state of the art in modelling of diarthrodial joints including the contact is introduced in the following section (Section 2.3.1). In Section 2.3.2 and 2.3.3, more detailed information about collision and contact is provided. Therefore, here, the fundamental issues in modelling of anatomic joint contact and in their graphic based computational simulations is surveyed.

2.3.1 Diarthrodial joint contact modelling

In considering the diarthrodial joints such as knee, hip and shoulder, contact stresses and friction occur at joint contact surfaces during joint articulations. Articular (hyaline) cartilage is prominently found at diarthrodial joints and covers bone contact surfaces to provide the reduced friction resistance. It is well known that, active joint contact forces

are due to skeletal muscle compressive actions. Normally, under compressive stresses, cartilages respond to contact forces through widening the contact area where the stress is distributed over this area is reduced. However, the cartilage response differs which depends on type of loading (i.e. static, dynamic) and joint conditions such as normal, injured, abnormal and intact joints. This is probably due to the difference in contact surface, cartilage degeneration, connective tissue defect, and overall tissue interactions. The articular cartilage and diarthrodial joint degeneration is also linked to the excessive contact stresses (Setton, Tohyama & Mow, 1998). The excessive contact stresses can probably occur due to high external dynamic loading at which skeletal muscles are subjected to and compress joints to respond to these loads (i.e. weight lifting). Moreover, the contact problem is a very important in designing joint prosthesis. In total or semi joint replacements, the patient satisfaction is essential which can only be happen through satisfying normal joint functions after the replacement. Thus, in designing the joint prosthesis (implant), material selection and geometry are fundamental issues. In terms of the material selection, the prosthetic contact material needs to satisfy the realistic cartilage contact force distributions. For example, some flexible materials such as polyethylene based materials are used in designing the knee joint implants. In particular, this issue relates to the fact that if the contact surface material is reasonably flexible, contact surface is allowed to deform under compressive loading and contact stresses are distributed relative to the deformed area. However in some prosthetic cases very hard contact may lead to a very high impact forces. The high level of forces can result in implant failure, abnormal joint functionality, and musculoskeletal tissue disorder as well as joint pain. It is also possible that after the tissue reconstructive surgery, tissue lengths might be shortened or lengthened which affects the natural tissue moment arm. In this case, moments, according to each skeletal muscle, tendon and ligament are not in the range of their natural values and do not contribute the natural joint torques which directly affect the artificial or natural contact responses. It must be noted that in a musculoskeletal joint modelling applications the above interactions have to be taken into account. However the modelling of anatomical joint contact which involves tissue interactions and complex contact surfaces, is more difficult than modelling a mechanical contact between two simple shaped (i.e. sphere) solid bodies. In precise anatomic joint models, it is required to deal with non-spherical, non-convex, non-rigid anatomic contacts. But, it is very difficult to handle all these in building a realistic joint model. That is why anatomic contact surfaces are mostly approximated to

simple geometric shapes such as sphere, ellipsoid and planar surfaces. In despite of its practicability, this kind of approximations restricts the correct contact point and force predictions in anatomic fidelity. In many performed studies, analysis of deformable cartilage contact dynamics as well as artificial cartilage material in prostheses (i.e. UHMWPE) has been investigated through FEA (Rawlinson & Bartel, 2002; Otto, Callaghan & Brown, 2001; Sathasivam & Walker, 1998 cited in Bei, 2003) and with elastic foundation based contact model (Blankevoort et al., 1991). Through applying the FEA, geometric based contact pressure distributions are measured accurately. In the FEA, it is possible to carry the dynamic analysis including muscle and ligament deformations and contact along with discretisized anatomic geometries. In addition to the FEA, elastic foundation theory is widely employed to model diarthrodial joint contact (An et al., 1990; Marai, 2007; Pandy, Sasaki & Kim, 1997). In this theory, springs are used to represent elastic layer of cartilage including the material and geometrical nonlineariies to calculate the pressure distribution between cartilage-cartilage and artificial joint contact. In Blankevoort`s study (1991), the proportionally increasing linear elastic cartilage stiffness with respect to the increasing contact forces and surface deformation has been presented. Various researchers published cartilage thickness based studies for human and animal diarthrodial joints, which is an important factor to define the deformable cartilage area and distance between contacting bodies. In one of the recent applications Li et al. (2005) presented an experimental work to clarify the articular cartilage thickness distribution at knee joints. According to Li (2005) this kind of information is still lacking which is necessary to define the correlation between joint loading, movements and contact relative to cartilage thickness. Thus, Li (2005) extracted 3D mesh data from MR images, and showed that cartilage contact surfaces which are subjected to frequent contact, possess thicker tissue properties than other regions which are not mainly involve in articulation contact. Furthermore, Han et al. (2005) presented a FEA based articular cartilage model including its real surface geometry. The application was proposed to investigate the cartilage contact at cat's patella-femoral joint. Cartilage has been modelled as realistic biphasic layer and contact pressures are measured according to the biphasic theory (Mow et al., 1980). This study is one of the rare cartilage contact modelling applications that considers the real surface geometries. The biphasic theory has been cited in many cartilage modelling applications. During the last few decades, number of techniques has been developed in order to model joint articulations with geometric

contact surface representations. In most of these studies, the main focus has been the modelling of human knee joint. The very early knee joint model was presented by Crowninshield, Pope & Johnson (1976) who studied the quasi-static response of the knee joint contact. In their model, joint translations and rotations are evaluated with the guidance of ligament forces. Ligament forces were based on the geometric configuration change between the initial and displayed knee joint articulations. In this analysis anatomic contact surface geometry and external force measurements weren't available. The first quasi-static inverse method application including 3D bone surface data was presented by Wismans, Veldpaus & Janssen (1980). In this model, knee joint is concerned as tibia-femoral joint with the rigid body contact assumption. Ligament elongations are analysed as a function of knee flexion-extension movements. For the sake of the simplicity, the contact friction between articular cartilages and menisci influence are neglected. In Wisman`s model (1980) the articular cartilage is assumed as elastic material due to the lack of viscoelastic tissue properties. Additionally, Essinger et al. (1989) focused on investigating the prosthetic knee joint behaviour through with a 3D quasi-static model. Furthermore, Hirokawa (1991) presented a 3D mathematical model of the knee joint with focusing on articular contact surface descriptions with high order polynomial equations. The study was one of the most comprehensive contact based multi body knee joint models yet presented which included the patella and the ligament interactions. Blankevoort et al. (1991) also focused on the 3D mathematical modelling of human knee joint including the deformable joint contact. Their study is based on the work presented by Wismans, Veldpaus & Janssen (1980) and the mathematical description of deformable contact is similar to the Essinger`s work (1989). Recently, Blankevoort`s contact model (1991) which is also referred to as elastic foundation model has been utilized to model distal radio-ulnar joint contact (Marai, 2007). The two dimensional joint models studied by Moeinzadeh, Engin & Akkas (1983), where rigid body contact is employed. Abdel-Rahman & Hefzy (1993) presented the changing in geometric contact points and contact forces during tibia-femoral articulations. After Abdel-Rahman`s work (1993), Ling, Guo & Boersma (1997) introduced kinematic and dynamic model of the knee joint which provides the further understanding of knee joint mechanism. Until the early 2000, most of the existed joint models were two dimensional dynamic models and few 3D models existed previously. More detailed rigid contact based 3D dynamic knee joint model has been proposed by Abdel-Rahman & Hefzy (1998). They used approximated contact surface

geometries where femur contact surface is approximated as sphere and tibia contact surface is approximated as planar surface. During joint articulations, geometric contact point changing as well as ligament force effects have been investigated. They explained the concept of ligament injury mechanism under dynamic and impact loading conditions. However the model does not contain the geometric nonlinearities, menisci and the tissue wrapping. Further, a rigid contact based 3D dynamic knee joint model was presented by Piazza & Delp (2001). This model includes both tibia-femoral and patella-femoral joints of knee joint complex. The aim was to evaluate the implanted knee joint functions during the step-up task. In their graphics based joint contact analysis, collision detection package RAPID (Section 2.3.2) has been utilized to find potentially colliding points between triangulated rigid bone meshes. For the contact and collision response, the contact constraint based equations have been used and solved through the complementarity formalism (Section 2.3.3). In order to minimize the computational time while solving the complementarity formalism, the Dantzig`s solution algorithm (Dantzig, 1998) was performed. Most commonly, contact based joint models have been applied to the knee joint. However less focus paid in modelling of elbow joints probably due to its complex anatomic behaviour and the geometry. More widely, some studies have been carried out to model the gleno-humeral joint (Novotny, Beynnon & Nichols, 2000). Moreover, most of the contact based elbow joint analyses are experimental. These experimental based studies usually seek the natural and artificial elbow joint articulations. More interestingly, radial head fractures and following radial head reconstruction are the most commonly faced orthopaedic cases. In such traumatic situations, the type of the implant reveals significant importance for contact and surrounded tissue interactions. For example, according to Morrey, Askew & Chao (1981), the insufficient elbow joint contact loading, weakening ligaments and implant failures are almost always seen in silicone type implants due to its insufficient contact loading performance. In order to tackle the joint contact loading problems, metallic type of implants has been preferably utilized (Harrington et al., 2001). Therefore, In order to analyse the contact characteristics with several metallic radial head implantations, Liew at al. (2003) presented a valuable experimental study. In their study, the radio-capitellar joint contact forces and contact area has been analysed before and after the implantation.

2.3.2 Collision detection

Collision detection is a fundamental geometry based application in computational simulations (Ericson, 2005). Collision detection applications depend on the purpose of the simulation and object geometry. In general, objects are represented as collection of regular primitive shapes (i.e. cube, sphere, and cylinder) consists of primitive elements such as patches and triangles. Commonly, during the collision detection, these primitive elements so called geometric features are checked to obtain potentially colliding object pairs. In considering more sophisticated collision detection algorithms, parametric and implicit surfaces (B-spline, NURBS) as well as Constructive Solid Geometries (CSG) are used for intersection calculations between objects. However for the sake of the simplicity, polygonal surface approximations are utilized to simplify the representation of object geometry for more efficient collision detection in terms of the distance calculations. In a very basic closest feature (triangle, patch, edge) check, total n number of triangle or edge is scanned between object pairs, as this is defined as an n^2 comparison. If the objective is to perform computational simulation of interacting objects, after detecting the potential collisions, then the computational simulation procedure is followed by contact and collision response. In considering the collision response procedure, approximated object surfaces are not suitable as this can result inaccurate contact response calculations. Nevertheless, for accurate computational simulations, sufficient collision detection variables are required, which is only possible through sophisticated algorithms and detailed object geometries. The complexity of such algorithms can be between $0(n\log(n))$ and $0(n^2)$. In collision detection, number of geometrical procedures needs to be performed. These are; determination of relative object configurations, detection of intersections, distance calculations between objects as well as boundaries, computation of separation distance between colliding objects, penetration depth calculations and etc. Thus the common name *"proximity queries"* (Lin & Manocha, 2003) is used to represent such collision detection procedures. Most of the developed collision detection algorithms are originally designed based on proximity queries for convex objects such as searching close features of convex objects, seperating axis testing between polytopes (Chung & Wang, 1996) and computing translational distance and penetration dept between convex object pairs (Cameron, 1986). However object representations had to be improved in order to present more realistic simulations along with the proximity queries of non-convex objects that exist in

almost every application. However collision detection applications for non-convex objects have been rarely taken into account which was mainly due to required decomposition (Bajaj & Dey, 1992) of non-convex objects which result in time consuming in collision detection. In this field, extensive research studies have been proposed to speed up the collision detection process along with the proximity queries. On the other hand, most of the developed collision detection algorithms used to be performed with discrete time instances. However, in computational games, virtual surgery simulations, and in various applications where the real time object simulations are essential, continuous collision detection algorithms are needed.

Going into more detailed description of collision detection methods, basic primitives are known as the simple collision detection tests. Basic primitive tests are closest feature intersection tests which check closest triangles, rays, polygons between object pairs. Collision detection between compex geometries are costly due to applying proximity queries between high number of polygonal objects. In boundary based collision detection, objects are represented in boundaries like sphere, box or its convex hull boundary. Then the queries such as intersection test are performed between these boundaries (Teschner et al., 2004). Bounding volume (BV) construction is one of the most commonly adopted techniques to perform efficient collision detection between such objects with eliminating the unnecessary test for un-colliding parts by, first, checking their bounding volumes. This elimination is also referred to as culling. Sweep and proone algorithm is one of the most commonly performed algorithms which is employed to check potential intersections between bounding boxes (Cohen, Lin & Manocha, 1994). Through applying this kind of algorithms, object parts are not considered for collision test if they are not in the potential collision trajectory. Therefore, the combination of such bounding volumes in a hierarchical tree is referred to as bounding volume hierarchies (BVHs). In constructing bounding volume hierarchies, axis aligned bounding boxes (AABB), oriented bounding boxes (OBB), k-Dop trees, sphere trees (Hubbard, 1995), and convex hull representations are most commonly employed. The convex hull is the tightest boundary representation wraps around object surface. However, it is well acknowledged that tighter boundary representations result higher number of primitives to be tested. Bounding volume hierarhies are constructed using either top-down or bottom-up approaches. In computer simulations, concerning the state change of objects reletive to time, some trees lead to

perform faster updateing during simulations. Some researchers extended and improved the bounding volume based collision detection tests (Lin & Manocha, 2003). Quinlan (1994) presented a method which is able to calculate the distance between non-convex objects along with hierarchy based spherical boundary representations. Applying spherical bounding hierarchy provided the efficient collision test through predicting unnecessary object parts not to be included in collision detection. In further applications (Bradshow & O`Sullivan, 2002, 2004), sphere-tree like boundary representations have been utilized to construct convex and non-convex object boundary hierarchies. Due to the efficiency of using k-DOP tree, it has been used in many collision detection applications. Thus, faster collision detection tests have been achieved while objects are translating and rotating (Kolosowski et al., 1998, Zachman, 1998). Mezger, Kimmerle & Etzmub (2003) with introduced k-dop tree based collision detection method to simulate cloth deformations. In addition to hierarchical structure representations, Barequet et al. (1996) introduced the "boxtree" like boundary representation which is so called "hierarchical structure of nested boxes". This method has been proposed for efficient ray traycing and collision detection between 3D polyhedral objects. Then, Van Der Bergen (1998) presented a collision detection algorithm to handle collision detection during rigid and deformable body simulations. In this case the efficiency has been achieved by employing AABB trees. Bonner & Kelley (1998) introduced spherical volume hierarchy for 3D object representations. They specifically tested the collision between moving robots through predicted paths in complex environments. James & Pai (2004) used bounded deformation trees (BD-trees) to handle deformable object simulations. Their results showed that the collision detection performance between deformable objects along with BD-trees is equivalent to the standard collision detection tests performed for rigid bodies. It is normally acknowledged that the rigid object collision tests are easier and faster than handling deformable object tests. Recently Larrson & Moller (2005) presented a bounding volume hierarchy based dynamic collision detection method for breakable object simulations. In addition to bounding volume hierarchies, spatial partitioning is a commonly performed collision detection method which aims to subdivide the space occupied by objects. The space is either represented by regular grid cells or voxel grids. Along with spatial partitioning, intersection check and closest feature tests are performed by spliting space into a number of grid cells and check if a cell is shared by two objets or not. If the cell is empty it is not included into further of collision tests.

This is an alternative approach to bounding volume hierarchies in culling operations. Due to the storage problem of employing regular grids for higher number of objects, hierarcical grids and trees are also employed as a space subdivision sheme. Most commonly applied spatial partitioning based representations are Octrees, Quadtrees, k-d trees and binary space partitioning (BSP) trees. The Octree hierarchy is extensively utilized for culling operations to reduce the number of cells. Earlier literature survey about spatial data structures and their hierarchical representations, especially about the quad-trees can be found in Samet`s book (Samet, 1989). Moreover, space discretization has also been performed technique to describe the path along space grid cells to navigate the motion of human with obstacles in 3D virtual environment. This method is useful when performing intersection test for complex and moving objects in real time. In robotics field, Shaffer & Herb (1992) presented an octree based data structure for collision avoidance. Their system was generated to perform the real time collision detection during the robot arm movements in 3D environment. This kind of applications are important to adapt for safe robot operation in dangerous environments such as nuclear industry. Furthermore, kinetic data structure (KDS) based collision detection algorithms have been used for real time time collision tests. In physical based multi body simulations, sometimes large size and accurate topologycal meshes are required for precise dynamic analysis. However, in such simulations even if efficient collision detection algorithms are employed they can still cost excessive in terms of computer processing time using the CPU (central processing unit). Nevertheless, transfering the execution from the CPU to graphics processing unit (GPU) can accelerate the efficiency and robustness of such computational simulations. Thus sophisticated physical based simulation methods and collision detection have been easily performed. Due to its highly fast output capability, the GPU based collision detection has become very popular and less time consuming. Comprehensive comparison between BVH, spatial partitioning and the GPU assisted collision detection methods is well documented (Teschner et al., 2004). As mentioned earlier, culling operation is essential for collision detection where some parts of objects and object spaces are eliminated before collision detection test. In addition to these traditional culling operations, rendering pass based GPU assisted techniques have been also developed. Usually the GPU assisted collision detection algorithms are referred to as image-space based algorithms. Image-space collision detection methods and geometric textures are widely performed GPU assisted methods where the query information

coming from depth buffers, colour buffers and stencil buffers. The information is read back from the GPU memory and passed to the CPU (readbacks) that can result very slow process. Thus occlusion query is extensively utilized along with recasting algorithms to avoid slow readbacks. For example, occlusion query simply counts the number of pixels of triangled object. Along with the requirement of fast collision detection, image-space based interference test has been developed by Baciu, Wingo & Hanqui (1999). The implemented code is also known as RECODE. Xie, Yang & Zhu (2007) presented sphere-tree bounding volume constructions and the GPU assisted collision detection algorithm for surgical simulations. In their application, occlusion queries have been utilized which lead faster culling operation and collision detections. Govindaraju et al. (2007) used chromatic decomposition for real time (continuous) collision detection between deformable objects. Redon et al. (2004) performed continuous collision detection based on CULLIDE algorithm to handle very fast object state changes during simulation. Moreover, GPU assisted, image-space based techniques have been performed for the real time collision detection between surgical tools and deformable tissues in virtual surgery (Lombardo et al., 1999). Rodriguez-Navarro & Susin (2005) presented image-space based collision detection method with gathering the depth-buffer information to simulate the real time cloth behaviour on moving body. Despite of the performed efficient image-space based collision detections; their inapplicability for further collision response operation presents the limitation of this method (Teschner et al., 2004). Further literature surveys on the performed and developed collision detection methods have been provided by a number of researchers (Lin & Gottschalk, 1998; Jimenez, Thomas & Torras, 2001). In terms of their applicability in industry, the most commonly adopted collision detection libraries are RAPID, I-Collide, V-Collide, Q-Collide, Voronoi-clip, SOLID, SWIFT, SWIFT ++, Canvas 3d JS, PQP, QUICKCD and GIMPACT. Some libraries either use existing well known collision detection algorithms such as Lin-Canny algorithm and enhanced Gilbert-Johnson-Keerthi algorithm (GJK). The Lin-Canny algorithm is an early developed algorithm for closest feature tests between polyhedral convex and non-convex objects. In some cases, the algorithm fails to find the closest features before a penetration occurs. Therefore, after the penetration, it applies to backtracking to find exact collided features (i.e. points). But in despite of this limitation, it is a fast algorithm in object distance computations. The Lin-Canny algorithm has been adopted for I-Collide library (Cohen et al., 1995) to handle the collision in “large scale”

environments. Mirtich (1998) presented the V-clip (Voronoi clip) algorithm. The algorithm is adopted to perform closest feature test between objects in a similar way as of Lin-Canny algorithm. it demonstrated an improved capability in detecting collision between polyhedras overcoming some limitations of Lin-Canny algorithm in distance computation. Unlikely to Lin-Canny and V-clip algorithms, GJK (Gilbert, Johnson & Keerthi, 1988) is a simplex-based rather than closest feature based algorithm. The performance of the GJK algorithm is better than the Lin-Canny algorithm in computing penetration depths. RAPID (Robust and Accurate Polygon Interface Detection) is one of most commonly used software packages which is non-commercial and free C++ library easy to use for rigid object collision detection tests. It can perform collision detection with disconnected triangle clouds and polygon soups with not requiring structured models. Oriented bounding box (OBB) hierarchy is a fundamental structure employed in RAPID (Gottschalk, Lin & Manocha, 1996). RAPID package performs collision detection in two stages which are testing contacting triangles with OBB trees then performing exact collision test to list exact colliding triangles. RAPID is also the collision detection tool for ADAMS and Lifemod dynamic engines (Section 2.1.5). Other widely used collision detection library is V-COLLIDE (Hudson et al., 1997) which performs *N* number of polygonal object processor (rather than a pair) in large virtual environments (e.g. in VRML). V-Collide code was built based on RAPID library. V-Collide can perform collision detection between unstructured models and arbitrary models (polygon soup). The code contains three processing stages which start with constructing bounding box hierarchy (e.g. AABB, OBB) then employing sweep-prune algorithm and checking potentially colliding N number of object pairs. After limiting the M number of potentially colliding objects into L number of object pairs, the RAPID library is utilized for exact colliding object future list. As the distance calculation is not reported by RAPID, V-Collide also lacks in this option. However, in addition to RAPID, V-Collide package is capable to handle position change of objects and updates the potentially colliding feature information. In addition to V-Collide, I-Collide collision detection library (Cohen et al., 1995) can also perform N number of convex polygonal object processor. After constructing the bounding boxes, sweep and prune algorithm is used and N number of objects are reduced into M number of object pairs similar with V-Collide stages. If object boundaries are in contact than closest features and point calculations are performed for exact list of object features in contact. At each position change, I-Collide can list potentially colliding pairs and built updated

contacting object future information. Furthermore, Q-Collide (Chung et al. 1996) has been built up on I-Collide library. The robustness of Q-Collide is close to RAPID, I-Collide and V-Collide due to being driven by similar procedures. Q-Collide utilizes the seperating plane theorem to depict disjointed polytopes (e.g. ellipsoid, thin rod, flat plate). For dynamic applications, temporal and geometric coherences are used to speed up the collision detection. In considering the Mirtich`s feature based Voronoi clip algorithm (Mirtich, 1998), the algorithm has been adopted to develop V-Clip library through its C++ implementation. Then java port of the V-Clip had released. The purpose of developing the V-clip algorithm was to provide more efficient, fast and robust collision detection and polyhedral distance calculation than existing libraries. V-Clip can perform collision detection and closest point checks between convex, non-convex and even between polygonal soup models. It is able to calculate distances between close objects and also can calculate penetration depth. In considering rigid and deformable object simulations, SOLID library is also a very popular choice. SOLID (Software Library for Interference Detection) is a C++ collision detection library constructed to tackle collision detections for moving rigid and deformable objects (Van Den Bergen, 1999). The library is built based up on the GJK algorithm. In the SOLID package, objects are represented by standard primitives such as cones, boxes and spheres as well as complex polytopes. The library can be adopted for further simple collision response operations. Similarly, SWIFT (Ehmann & Lin, 2000) is a library, which is utilized for collision detections between convex polyhedral solids undergoing rigid motions. It`s functionality is similar to I-Collide library. It provides faster intersection test than I-Collide and V-Clip libraries and it`s intersection test is based on an algorithm similar to the Lin-Canny algorithm. It can detect weather pair of objects are intersecting and can compute approximate and exact distance between object. Original SWIFT is able to test convex objects and convex pieces of non-convex objects. Additionally, SWIFT++ (Ehmann & Lin, 2001) was built top of the SWIFT to cope with non-convex object intersection test through improved SWIFT functionalities. More information about collision detection libraries can be found in the GAMMA (Geometric Algorithms for Modeling, Motion, and Animation) web page (GAMMA).

2.3.3 Contact and collision response

Most commonly performed contact and collision response methods are introduced in this section. Contact and collision problems are common in virtual multi body modelling and simulations as they assessed as major problems in robotics, computer graphics, general engineering, and biomechanics. As mentioned in Section 2.1.2 and 2.1.3, kinematic descriptions (i.e. displacement, velocity) are needed to construct equations of motion. As desired, multi body system can be dynamically analysed with introducing additional kinetic factors such as forces into equations of motion. Concerning equations of motion for relatively moving bodies, total mobility (DOF) of the mechanism can be expressed with introducing kinematic idealised joints and substituting bilateral joint constraints into equations of motion. In this case, the total mobility of the system is permanently constrained. However in some situations introducing the time dependent bilateral joint constraints is not adequate to describe the relative interactions between bodies. Thus, state dependent unilateral contact constraints are convenient to describe the relative body movements or interactions with respect to the contact conditions. In this case, the system mobility is not permanently constrained. Rather, system mobility constraints are state dependent constraints and driven when contact is active. In unilateral based contact modelling, the aim of introducing the contact constraints is for preventing the body interpenetrations and response the collision. Basically, in unilateral contact based modelling approaches, contact and collision response are incorporated. In order to apply unilateral contact modelling, there has to be at least two bodies either approaching to each other resulting in an impact, or they need to be in resting contact. Otherwise if there is no impact or contact, there is no collision response. For the impact problem, there are two general approaches such as continues and discontinues (Lankarani & Nikravesh, 1990 cited in Flores et al., 2008). In continuous based approaches the continuous contact force models are utilized such as the Hertz theory. The unilateral contact constraint based methods such as penalty based method is well known continuous force based approach. Generally continuous force models represent forces and deformation in a continuous manner. This approach is generally used for slowly approaching and relatively deforming bodies. That's why in continuous force based penalty responses the analysis takes a long time to construct reaction forces. However, if bodies are approaching to each other with a very high velocity, especially for hard and rigid bodies, resulting in a

very short time impact where penalty based collision response methods are not adequately fast enough to respond to the collision. Physically, these high velocity impacts result impulses between bodies which occur in a very short time and develop very high impulse forces. Thus, instead of using penalty based collision response methods, other unilateral contact based approaches are used. However the impact situation yields discontinuous change in body velocities which needs to be taken into account. Therefore in order to tackle with the impact and system discontinuities, non-smooth model (Moreau & Panagiotopoulos, 1988) is used. According to the Newton`s law of restitution, the total momentum before the impact is equal to the total momentum after the impact. The non-smooth model also handles the system equations before and after the impact with introducing various time integrators such as event-driven and time-stepping. The event-driven integrator distinguishes motions before the impact as smooth event and after the impact as non-smooth event. The event-driven integrator is an acceleration based approach and the velocity is assumed as continuous where contact forces and acceleration are solved. The time-stepping integrators are more commonly used to handle multiple contacts. After the impact or at the switching point (incidence) step size refinement is performed. This means that the smooth part before the impact is considered with large step sizes and after the impact, step sizes are refined to provide accurate solution. The time stepping approaches are velocity based approaches where Newton`s equations of motion which compose of force and acceleration relations are replaced by equations with impulse velocity relations. Apart from the non-smooth modelling, impulse based collision response methods are performed for such impact case. In summary, constraint based collision response methods, penalty based collision response method (Marchal, Aubert & Chaillou 2004; Hirota, Fisher & State, 2003; Hirota et al., 2001) and impulse based non-smooth model are all described under the unilateral contact constraint based methods. A well known impulse based collision response method has been presented by Mirtich (1996b). The impulse based collision response method is different than the non-smooth approach where the method has not been designed to perform dynamic contact analysis. On the other hand, constraint based and penalty based methods are used for both rigid and deformable bodies, but impulse based method is only performed for rigid body contact responses.

2.3.3.1 Contact constraints and contact constraint based collision response

The time dependent bilateral joint constraint based equations of motion are normally expressed by ordinary differential algebraic equations (DAE). However, through unilateral contact constraints, it is required to apply regularized modelling and complementarity formalism to solve the system equations (Pfeiffer & Glocker, 1996). The regularized modelling handles the contact phenomena by using spring attachment(s) between contacting bodies. The contribution of spring forces is only evaluated at active contact conditions and neglected if bodies are not in contact trajectory. This information provides the clue about state dependency of unilateral contact constraints. Through penalty based method, penetration depth is calculated to place spring or spring-dashpot elements to develop the contact responses. On the other hand, in constraint based contact formulations, usually the approximation space is constructed. Then set of kinematic constraint equations which are called variational inequality equations are formulated based on the described constraints such as the space. Moreover, there are some parameters that are either exist when there is a contact or don't exist when there is no contact such as friction parameter. Thus, the friction is expressed by equality equations to assess its exact contribution to the contact. The variational inequality constraint based contact formulation is different than the penalty based contact formulations where bodies are not allowed to violate the defined space. In this case, interpenetration is not allowed through obeying the variational inequality equation which defines the space or distance between bodies. Concerning the non-smooth mechanic problems due to impact and friction, these inequality and equality equations are formulated either by linear complementarity (LC) formalism or nonlinear complementarity (NC) formalism (Pfeiffer & Glocker, 1996; Murty, 1988; Baraff, 1989; Baraff, 1994; Panagiotopoulos, 1975). Through these formalisms, equality and inequality equations are solved by specific algorithms such as Lemke`s Algorithm (Lemke, 1968). However, applying the mathematical programming or linear complementarity formalism, can yield higher computational cost. It is well known that time dependent position and velocity change is formulated through integration of equations of motion. In unilateral contact based dynamic analysis, inequality and equality constraint equations can be expressed as dynamic equations of motion. The general formulation of linear and non linear complementarity problems their solution algorithms can be found in many textbooks (Featherstone, 1987, 2007; Cottle, Pang &

Stone, 1992) and scientific papers (Cottle, Pang & Stone, 1996; Cottle & Dantzig, 1968 cited in Featherstone, 2007). In handling the contact, Trinkle (a leading researcher in this field) has developed a complementary based contact method (Trinkle, 2003) to handle frictional contact between multi rigid bodies. The analytic multi rigid contact formulation has been introduced by Pang & Trinkle (1995), which is used to solve frictional rigid contact problem with complementarity formalisms. The aim was to obtain the instantaneous acceleration of object, and calculate the contact forces during rolling and sliding conditions. Stavroulaki & Stavroulakis (2002) deal with the unilateral contact modelling by using FEM and by handling the non-smooth mechanic problems. The Signorini boundary conditions are widely employed in terms of variational inequality equations for solving the contact problem between rigid environment and deformable body or contact problem between two deformable bodies. Signorini`s theory has been introduced as non-interpenetration constraint equation for elasto-mechanic contact problems (Eck, Jarusek & Krbec, 2005). Signorini`s contact theory is a computationally efficient method preferably applied in haptic (Duriez, Andriot & Kheddar, 2004) and real time simulations. Also, Signorini`s contact theory works very well in frictional contact problems (Refaat & Meguid, 1995).

2.3.3.2 Rigid and flexible contact and penalty based collision response

- Rigid and flexible contact

As discussed earlier, in unilateral based contact formulations, non-penetration constraints can be introduced based on regularized modelling or complementarity formulations. The penalty based collision response method is also referred to as the regularized modelling approach which is widely employed unilateral contact formulation. In considering the spring attachments in penalty based method, it shouldn't be confused with spring or spring-dashpot elements which are also employed to model the flexible body deformations (Section 2.2.2). Through penalty based method, these elements are only used to prevent interpenetrations between colliding bodies. From physical point of view, in the real environment all bodies are deformable and there isn't any structure which can be described as a rigid body. Even if a body is very hard it is more likely to present very small deformations under loading. However, for the sake of the simplicity, in extensive multi body modelling applications, hard

bodies are assumed as rigid bodies. In some cases, if a hard body is assessed as flexible body, its mechanical behaviour is mainly modelled based on local and very small deformations (i.e. Hertz Theory). The local deformations are relatively small dimensional deformations when comparing with whole body dimensions. On the other hand in considering a flexible body contact, its whole body deformation (global deformation) has to be taken into account for describing its mechanical behaviour. In both cases the penalty based springs can be employed to prevent the interpenetration between colliding bodies. In flexible body contact, the difficulty arises with variational change in body geometry due to the deformation. Thus it becomes difficult to place reactive contact forces between bodies where position of contact points can change and applied force cannot respond to the contact appropriately. Thus appropriate definition of the global deformations plays important role in contact problems. Briefly, flexible body deformation is mainly modelled with Hookean springs and Newtonian dashpot elements (Section 2.2.2). Based on the theory of elasticity (Timoshenko & Goodier, 1970; Hunt & Crossley, 1975 cited in cited in Flores et al., 2008), Hertz describes the best known contact theory which is originally designed for the contact analysis between half space spherical bodies. Hertz's theory was introduced by Hertz in 1881 (cited in Johnson, 1987), for solving elastic contact problem and pressure distribution between two elastic solid (elastic half-space) bodies. In some elastic body contact cases, when the contact energy is transferred into strain energies this can result high frequencies and vibrations. In order to minimize the system vibrations, dashpot elements are introduced into the system. In general linear viscoelastic Kelvin–Voigt model is utilized to represent the flexible body contact. Emerging of the attention in modelling of viscoelastic contact phenomena has lead researchers to develop several numerical and analytical methods. Furthermore, Zhu, Zwiebel & Barnhardt (1999) proposed an impact model between two bodies which are modelled as linear viscoelastic bodies. They proposed to model the energy dissipation and energy loss at the impact and the separation, with the function describes the energy dissipation during the impact correspond to the restitution coefficient in the form of hysteresis response provide the relative information between force and the deformation change. Lankarani & Nikravesh (1990) developed a contact force model based on the Hertz contact theory with the addition of hysteresis damping function to represent the energy dissipation during the impact.

- Penalty Based Collision Response

The penalty based collision response method is fast and easy to implement. However due to employing large stiff springs to allow only small penetrations can yield stiff system problems. The problem can be easily solved with effective numerical solutions such as higher order Runge Kutta. Additionally, spring constants can be selected arbitrarily. The procedure of penalty based method of unilateral contact problem starts with calculating the penetration depths of bodies at every time step and generate spring forces to prevent the penetration. In penalty based collision response approach, commonly discrete-time based simulations are driven to generate the response forces. These responses or penalty forces are computed as a function of the penetration depth. Depending on the simulation time step size, unwanted large penetrations can occur which results the undesired penetration depth violations. Thus, time stepping schemes have to be applied to resort to a very small step-length. This enforces the numerical stability. Furthermore, the proposed penalty based collision response method, which is based on the measuring penetration depth between colliding bodies, requires an adequate collision detection method to provide the potential collision and penetration depth information. The GKJ collision detection method (Section 2.3.2) is mainly employed to obtain the penetration depth information through Minkowski sums (Ericson, 2005) for further penalty based collision response applications.

2.3.3.3 Rigid body impact and impulse based collision response

Let`s assume that there are spherical solid bodies randomly moving in an environment, if they approach to each other, the relative distance may become zero and they collide which is also referred to as the impact (Stronge, 2004). The moment of the impact is so called incidence. The impact consists of two phases as compression (i.e. collision) and restitution phase. In considering an impact where at least one of the bodies is non-rigid body, the compression is followed by body deformation(s) through the normal direction of contact surface. Particularly concerning non-rigid body impact phenomenon, after the compression, the restitution phase stars at which the body velocities are reversed. The restitution phase ends when bodies are separated and contact is lost. In the case of body impact, the period of contact depends on the normal compliance of body contact area which is proportional to the body flexibility. The body contact area is a

distinguishing element in characterising the flexible, hard and rigid body contact types. Let`s consider two rigid bodies approaching to each other with very high velocities. In the case of impact, the contact period is very small comparing with flexible body contact period. Thus, in order to avoid the interpenetration in rigid body impact, penalty based collision response method is not employed. In this case impulse force development is more adequate response to act immediately after the impact to change and reverse penetrating body velocities. As introduced at the beginning of this section, impulse results in discontinuity in body velocities which is tackled with various time integrators such as event driven and step-timing. These methods are considered as unilateral based methods. In addition to these approaches, impulse based collision response methods are also used. After detecting the potential collisions impulses are generated at contact points to prevent the interpenetration. In impulse based collision response applications contact and collision response methods are performed separately unlikely the unilateral contact based methods. The impulse based collision response methods are capable to perform real time physical simulations. One of the earliest impulse based collision response methods has been introduced by Moore & Wilhelms (1988). The method handles collision response between articulating rigid bodies. In their application, Newton`s law of restitution is adopted where impact velocities are determined. Mirtich presented his well known impulse based collision response method in his thesis (Mirtich, 1996b).

CHAPTER 3

THEORY

3.1 Introduction

Through the literature survey, it is highlighted that musculoskeletal models still lack reliability in analysing joint articulations in anatomic fidelity. Extensive research is focused on developing a vast number of techniques in various areas to contribute to modelling of musculoskeletal and joint articulations. From biomechanics point of view, fundamental musculoskeletal modelling concepts which involve joint motion need to account deformation of musculoskeletal tissues, definition of tissue moment arms and wrapping conditions as well as account contact and collision conditions. In general, some of these concepts have long been faced in robotics, engineering and computational research areas and only for the last few decades that they have been adopted for many medical and biomechanical fields. According to the literature survey in Chapter 2, it has been emphasized that through developing a unified modelling, a precise musculoskeletal model can be achieved. It is appropriate to highlight that, in the proposed multi body modelling of musculoskeletal joints, all the formulations are based on multi body dynamics and discrete element analysis (DEA). Finite element analysis (FEA) and continuum mechanics are not involved in the dynamic formulations. Particularly for kinematic joint modelling, it is proposed to assess the anatomic joints as unconstrained joints (6DOF) rather than as the idealised kinematic joints. Thus, it is required to describe equations of motion based on the joint surface geometry. Therefore dynamic equation of motion formulation has been developed in this thesis to deal with the 6DOF joint mobility. The proposed dynamic formulation contains linear and nonlinear formulations based on geometric and material linearity and nonlinearity (Section 2.2.2). Furthermore, in dynamic equations of motion, Newton-Euler formulations has been adapted with the Euler`s coordinate system (R^3), employing the Euler angles or the direction cosines matrix whichever found appropriate. Two way transformations from one definition to the other are also given in the Appendix A4. In the proposed multi body dynamic formulation, bodies are connected by spring or spring-dashpot elements. These elements represent the soft tissues such as muscle,

ligament and cartilage. In particular, the joint contact between bodies is described by the contact surface and the cartilage properties (viscoelastic material). This approach is one of a number of novelties of the proposed modelling. The well known Kelvin-Voigt linear viscoelastic model has been adapted for the linear dynamic analysis. Moreover, Kelvin-Voigt based spring-dashpot arrangement is utilized for the nonlinear dynamic analysis. In this case, the newly developed nonlinear equations are used for spring and dashpot elements in nonlinear viscoelastic model. The proposed algorithm to process the vicoelastic nonlinearity is capable of modelling non-linear spring and dashpot in polynomial descriptions. In addition to the deformation modelling of tissues, estimation of tissue line of actions, moment arms and employing the muscle wrapping is very important in a precise musculoskeletal modelling. Due to the number of problems, accurate estimation of tissue moment arm is not available in the current literature. As previously mentioned in the introduction and in the literature survey chapters, the most important problems revealed in musculoskeletal modelling are: (1) Not being able to define joint articulations in anatomic fidelity as unconstrained joints and (2) Not being able to link tissue moment arms, tissue deformation and realistic wrapping conditions as well as joint contact to unconstrained joint articulations. Neither any currently available modeller nor experimentally produced results could relate the change in tissue lengths, calculation of moment arms relative to the unconstrained centre of joint translations and rotations. Thus, using the proposed unconstrained joint model as a basis of tissue length and moment arm estimations can be utilized to uncover the relation between these important variables. Particularly, it has been recognised that the wrapping of tissues is of significant importance in accurate definition of tissue deformation and in moment arm estimations. Thus, the description of the desired anatomically based relation between unconstrained joint movements with surrounded tissues can only be possible if tissue behaviours and tissue wrapping modelling are proposed to be as anatomic as tissue behaviours themselves. Currently the tissue wrapping is only possible through using simple geometric shapes where the wrapping is considered to be of a shortest path on a sphere or a cylinder. In addressing this problem, a novel algorithm is proposed for tissue wrapping on anatomic bone geometries rather than on simple geometric shapes. Through this approach, tissue moment arms can be obtained accurately. In addition to the deformation modelling of passive tissues such as ligaments, the active skeletal muscle force generations must be modelled too. The most popular skeletal muscle modelling application the Hill`s type muscle model (Section 2.2.2). The simplified

Hill`s muscle model has been implemented into the developed software which models a skeletal muscle with active and passive element. For the proposed overall dynamic analysis, inverse and forward or mixed dynamics analysis can be performed. Furthermore, contact and collision are specifically focused areas in this Thesis. Nevertheless the proposed unconstrained joint model can be defined as contact based driven joint model implicitly. In order to construct the complete dynamic formulations of joints, contact, collision detection and collision response phenomena need to be considered through the joint modelling process. In this case, anatomic surface geometry (non-convex) based collision detection algorithm has been developed as a basis of further contact and collision response procedures (Section 2.3). The new developed contact and collision algorithms are able to perform rigid as well as deformable contact analysis. The specific intension in this procedure is to provide geometric feature (i.e. triangle or vertex) based contact information during joint articulations. Where, this geometric information can yield to place spring or spring-dashpot elements between potentially colliding points. These elements are used to perform two main functions as collision response and the cartilage tissue deformation.

To summarise, the following formulations, theory and modelling are offered in this chapter:

1. Linear modelling of joints based on 6DOF, where the joint stiffnesses holding the joint together are calculated from the joint geometry, cartilage and ligaments. This model is expected to be useful for modelling joints with small movement but also useful to calculate joint stiffness, joint laxity, centre of instantaneous rotation, joint stiffness invariant properties (such as the centre of joint stiffness and the principal axes of stiffness).
2. Nonlinear modelling of joint based on 6DOF, and development of algorithms to analyse non-linear joint behaviour.
3. Muscle wrapping, joint contact, collision detection and collision response.

3.2 Unconstrained (6dof) joint modelling theory

3.2.1 Linearity and non-linearity

Linearity and nonlinearity can be expressed for material based and geometric based behaviours. Material linearity or nonlinearity originates from material properties. The geometric nonlinearity occurs due to large changes in kinematics such as large displacements and rotations. In a nonlinear system, the nonlinearity is either due to material or geometric nonlinearities or both. Thus, if material deflection is not proportional to applied forces it behaves nonlinearly and even there is no geometric nonlinearity, system is considered as nonlinear. On the other hand, if material exhibits linear behaviour, but there are geometric nonlinearities, the system is considered as nonlinear. Nevertheless, some nonlinear systems possess both material and geometrical nonlinearities. In the proposed linear unconstrained (6DOF) joint model, force deflection relationship is linear and the system geometry remains almost unchanged due to the very small motion applications. This approach leads to describe the multi body system as linear. The purpose of developing the linear multi body system is firstly for finding the initial equilibrium positions of joints and secondly (which is more important) for introducing the concept of the joint stiffness. By means of the joint stiffness concept, a new formulation is introduced for the assessment of anatomic joint stiffness as a quantitative description of joint stability and joint laxity. In the linear dynamic formulations, the multi body system is introduced by considering two rigid bodies, each body possessing 6DOF mobility. Spring and/or spring-dashpot attachments are used in order to connect the bodies and analyse the system as a jointed mechanism. In order to set up the linear dynamic equations of motion, generalisation of the equations of internal and external forces are needed. The internal forces or so called reactions are due to the dashpot and spring elements which have to be expressed in a unified and structured fashion for formulation of the damping and stiffness matrices. The damping matrix structure is identical to stiffness matrix structure except that stiffness coefficients need to be replaced by damping coefficients. After then the generalisation of the equations of linear momentum (force-acceleration equations) and angular momentum (moment of momentum) are required. It is assumed that deflections in the three principal axes of spring and/or dashpot remain orthogonal and orthogonality is preserved through the deflection cycle. It is assumed that the principal axes are axi

symmetric with the main stiffness along the main tissue direction with equal shear stiffness in the other two orthogonal directions.

3.2.2 Linear joint modelling

3.2.2.1 Spring or spring-dashpot elements

In this section the transformation of the stiffness matrix of a spring from its local axes frame to the global axes frame will be obtained. The equivalent damping matrix can be obtained by replacing the stiffness coefficients with damping coefficients. In this exercise the three dimensional springs are employed. Thus, three dimensional spring stiffnesses can be described in a local 3D Cartesian frame system. It is considered that the Cartesian frame system coincides with the principal axes of the spring. Therefore, the force deflection equation of the 3D spring element is given as:

$$\mathbf{f} = k\mathbf{x} \tag{3.1}$$

Where **f** is a force vector and **k** is the stiffness matrix (diagonal with the principal stiffness values) and **x** is the displacement vector. In general it is convenient to describe the behaviour of a system in the global axes frame (OXYZ). Hence, in order to express the spring stiffness in global axes frame, it needs to be transformed into the global axes frame. It is assumed that the principal axes of the spring and the global axes are all orthogonal, an orthogonal transformation exists between the two frames. Let`s assume that transformation matrix **T** is used for the transformations from local to global axes frame and $\mathbf{T}^T$ is used for the transformations from global to local axes frame. A vector **x** in the local axes frame could be expressed as a vector **X** in the global axes frame with multiplying it by the transformation matrix **T** as:

$$\mathbf{X} = \mathbf{T}\mathbf{x} \tag{3.2 a}$$

And the transformation from global to local is given as:

$$\mathbf{x} = \mathbf{T}^{T}\mathbf{X} \tag{3.2 b}$$

Then, the force equation, from the Eqn. 3.1 can be written in a form as:

$$\mathbf{f} = k\mathbf{T}^{T}\mathbf{X} \tag{3.3}$$

Therefore, if the Eqn. 3.3 is pre multiplied by the transformation matrix **T**,

$$\mathbf{T}\mathbf{f} = \mathbf{T}k\mathbf{T}^{T}\mathbf{X} \tag{3.4}$$

The force vector **F** is described in the global axes frame as:

$$\mathbf{F} = \mathbf{T}k\mathbf{T}^{T}\mathbf{X} \tag{3.5}$$

In short

$$\mathbf{F} = \mathbf{K}\mathbf{X}$$

Where

$$\mathbf{K} = \mathbf{T}k\mathbf{T}^{T} \tag{3.6}$$

It is noted that the transpose of the transformation matrix ($\mathbf{T}^T$) is equal to the inverse of the transformation matrix ($\mathbf{T}^{-1}$) which can be given as:

$$\mathbf{T}^T = \mathbf{T}^{-1} \tag{3.7}$$

3.2.2.2 Generalization of the equation of linear momentum

If the mass/inertia matrix in Newton-Euler formulation is obtained relative to the axes passing through the centre of mass of body ***i***, then the subsection of the mass matrix corresponding to linear momentum is a diagonal matrix containing the mass elements, shown in the Eqn. 3.8.

$$\mathbf{h}_l = \mathbf{m}\mathbf{v} \tag{3.8}$$

Where, $\mathbf{h}_l$ is linear momentum, **m** is a diagonal matrix and **v** is the velocity vector of the centre of mass (COM, casually known as the centre of gravity, COG) of body ***i*** in its local axes frame. The usual transformation from local axes frame to global axes frame is also used for transforming linear momentum from the local axes frame to global axes frame as:

$$\mathbf{H}_l = \mathrm{T}\mathbf{m}\mathrm{T}^T V \tag{3.9}$$

It is to be emphasized that mass matrix **m** is not changed (all diagonal elements contain the same mass value – mass is not directional). The force acceleration relationship can

be obtained from differentiation of the liner momentum. Thus, force acting on body ***i*** in the global axes frame can be expressed as:

$$\mathbf{F} = \dot{\mathbf{H}}_l = \frac{\partial \mathbf{H}_l}{\partial t} = \mathbf{ma} \tag{3.10}$$

Where **a** is the acceleration vector of the COM

3.2.2.3 Generalization of the equation of moment of momentum

For the analysis of rotational motion of a rigid body, the relationship between the angular velocity of the body and the moment of momentum vector is needed. The moment of momentum is expressed as:

$$\mathbf{h}_a = \mathbf{j}\boldsymbol{\omega} \tag{3.11}$$

$\mathbf{h}_a$ is the moment of momentum (angular momentum) vector, **j** is the moments of inertia matrix and **ω** is the angular velocity vector which are all described in the local axes frame of body ***i***. Here the moments of inertia matrix **j** may or may not be a diagonal matrix, however it is always symmetric. Therefore, the moment of momentum vector $\mathbf{h}_a$ has to be converted in global axes frame, for the final matrix assembly. As presented by the Eqn. 3.5 for the stiffness matrix, the transformation from local to global axes frame for the inertia matrix follows exactly the same steps. In this case, again, **T** is the transformation matrix and the angular momentum in the global axes frame is given as:

$$\mathbf{H}_a = \mathrm{T}\mathbf{j}\mathrm{T}^{\mathrm{T}}\boldsymbol{\Omega} \tag{3.12}$$

Similar to the procedure described for the **k** matrix, inertia matrix in the global axes is given by,

$$\mathbf{J} = \mathrm{T}\mathbf{j}\mathrm{T}^{\mathrm{T}} \tag{3.13}$$

If the vector differentiation is taken for $\mathbf{H}_a$, the moment vector is obtained in the global axes as:

$$\mathbf{M} = \dot{\mathbf{H}}_a = \frac{\partial \mathbf{H}_a}{\partial \mathrm{t}} + \boldsymbol{\omega} \times \mathbf{H}_a \tag{3.14}$$

In considering the Eqn. 3.14, $\boldsymbol{\omega}$ is the angular velocity of the body in the global axes frame. For the small angular velocities $\boldsymbol{\omega} \times \mathbf{H}_a$ is small and may be ignored.

3.2.2.4 Assembling the equations of motion

In order to assemble the equations of motion, the internal forces acting on individual bodies need to be calculated. The assembly of equations is carried out with tconsidering the body ***i*** and body ***j*** connected by spring $\mathbf{k}_{\mathrm{p}}$. The bodies are rigid and all inertial properties are assumed to be known. The body ***i*** and body ***j*** are shown in Figure 3.1.

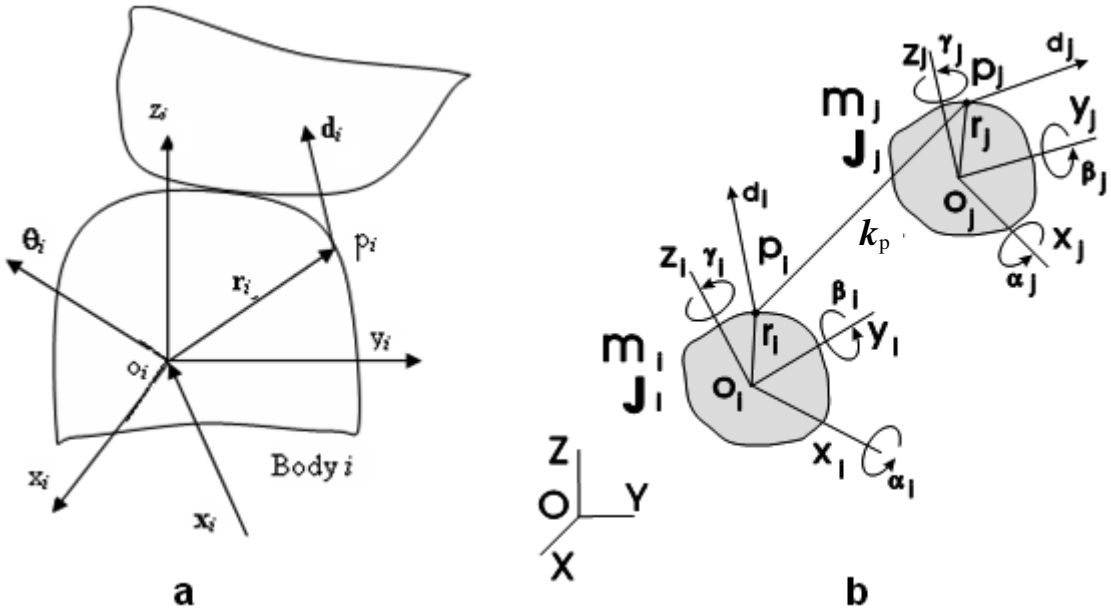


Figure 3.1 Body ***i*** and body ***j*** are in contact and connected by spring k_p

The dynamic motion of a rigid body is described by the translational and rotational motion of its centre of mass. The centre of mass of body ***i*** is given as o_i and its translational motion is defined as $\mathbf{x}_i$ and its rotational motion is described as $\boldsymbol{\theta}_i$. Similarly the translational motion of body ***j*** is described by $\mathbf{x}_j$ and rotational motion is described as $\boldsymbol{\theta}_j$. In this description, $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_j$ are infinitesimally small rotations and therefore they can be treated as vectors. There is no such restriction on $\mathbf{x}_i$ and $\mathbf{x}_j$. In order to describe the spring force acting on each body, the location of the force due to the spring *p* is given by two position vectors $\mathbf{r}_{pi}$ and $\mathbf{r}_{pj}$ which are measured relative to local axes frames $o_i x_i y_i z_i$ and $o_j x_j y_j z_j$ respectively. Although the local axes frames can be located at any point on each body, for the sake of simplicity their origin point is assumed to coincide with the centres of mass of body ***i*** and body ***j***. It is also assumed that the local axes system is fixed to the body and the body is rigid. As body ***i*** and body ***j*** are connected by spring k_p, for small motion, displacements of end points of spring are described in the axis frame of each body. The absolute displacement of $\mathbf{p}_i$ is measured relative to the local axes $o_i x_i y_i z_i$ as $\mathbf{d}_i$. The same description is valid for $\mathbf{p}_j$ relative to the local axes $o_j x_j y_j z_j$. Now small displacement of $\mathbf{d}_i$ and $\mathbf{d}_j$ can be described as follows.

$$\mathbf{d}_i = \mathbf{x}_i + \theta_i \times \mathbf{r}_{pi} \tag{3.15}$$

$$\mathbf{d}_j = \mathbf{x}_j + \theta_j \times \mathbf{r}_{pj} \tag{3.16}$$

Where, $\mathbf{r}_{pi}$ and $\mathbf{r}_{pj}$ are coordinates of spring attachment points in their respective axes frames, given as, $\mathbf{r}_{pi} = (x_{pi}, y_{pi}, z_{pi})$ and $\mathbf{r}_{pj} = (x_{pj}, y_{pj}, z_{pj})$. Displacements $\mathbf{d}_i$ and $\mathbf{d}_j$ are measured in local axes frames, and expressed in matrix form as:

$$\mathbf{d}_i = \begin{bmatrix} 1 & 0 & 0 & 0 & z_{pi} & -y_{pi} \\ 0 & 1 & 0 & -z_{pi} & 0 & x_{pi} \\ 0 & 0 & 1 & y_{pi} & -x_{pi} & 0 \end{bmatrix} \begin{Bmatrix} x_{oi} \\ y_{oi} \\ z_{oi} \\ \theta_{xi} \\ \theta_{yi} \\ \theta_{zi} \end{Bmatrix} \quad \mathbf{d}_j = \begin{bmatrix} 1 & 0 & 0 & 0 & z_{pj} & -y_{pj} \\ 0 & 1 & 0 & -z_{pj} & 0 & x_{pj} \\ 0 & 0 & 1 & y_{pj} & -x_{pj} & 0 \end{bmatrix} \begin{Bmatrix} x_{oj} \\ y_{oj} \\ z_{oj} \\ \theta_{xj} \\ \theta_{yj} \\ \theta_{zj} \end{Bmatrix} \tag{3.17}$$

Alternatively, notation $\mathbf{d}_i$ and $\mathbf{d}_j$ can also be described as:

$$\mathbf{d}_i = \mathbf{I}\mathbf{x}_i + \mathbf{R}_{pi}\theta_i \tag{3.18}$$

$$\mathbf{d}_j = \mathbf{I}\mathbf{x}_j + \mathbf{R}_{pj}\theta_j \tag{3.19}$$

Where $\mathbf{x}_i^T = \{x_{oi} \quad y_{oi} \quad z_{oi}\}$ and $\boldsymbol{\theta}_i^T = \{\theta_{xi} \quad \theta_{yi} \quad \theta_{zi}\}$ and $\mathbf{d}_i$ and $\mathbf{I}$ in the matrix form are given by:

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{R}_{pi} = \begin{bmatrix} 0 & z_{pi} & -y_{pi} \\ -z_{pi} & 0 & x_{pi} \\ y_{pi} & -x_{pi} & 0 \end{bmatrix} \tag{3.20}$$

The matrix notations of the displacement $\mathbf{d}_j$ can be obtained by replacing the suffix i with j. These matrix and vector notations can be combined as $\mathbf{A}_{pi}$ and $\mathbf{u}_i$ for displacement $\mathbf{d}_i$ and $\mathbf{A}_{pj}$ and $\mathbf{u}_j$ for displacement $\mathbf{d}_j$ as:

$$\mathbf{A}_{pi} = \begin{bmatrix} 1 & 0 & 0 & 0 & z_{pi} & -y_{pi} \\ 0 & 1 & 0 & -z_{pi} & 0 & x_{pi} \\ 0 & 0 & 1 & y_{pi} & -x_{pi} & 0 \end{bmatrix} \qquad \mathbf{A}_{pj} = \begin{bmatrix} 1 & 0 & 0 & 0 & z_{pj} & -y_{pj} \\ 0 & 1 & 0 & -z_{pj} & 0 & x_{pj} \\ 0 & 0 & 1 & y_{pj} & -x_{pj} & 0 \end{bmatrix} \tag{3.21}$$

$$\mathbf{u}_i = \begin{Bmatrix} x_{oi} \\ y_{oi} \\ z_{oi} \\ \theta_{xi} \\ \theta_{yi} \\ \theta_{zi} \end{Bmatrix} \qquad \mathbf{u}_j = \begin{Bmatrix} x_{oj} \\ y_{oj} \\ z_{oj} \\ \theta_{xj} \\ \theta_{yj} \\ \theta_{zj} \end{Bmatrix} \tag{3.22}$$

Thus $\mathbf{d}_i$ and $\mathbf{d}_j$ can be expressed in their new forms as:

$$\mathbf{d}_i = \mathbf{A}_{pi}\mathbf{u}_i \tag{3.23}$$

$$\mathbf{d}_j = \mathbf{A}_{pj}\mathbf{u}_j \tag{3.24}$$

Forces and moments acting on body ***i*** at position ***p*** may be expressed as:

$$\begin{Bmatrix} F_{pix} \\ F_{piy} \\ F_{piz} \\ T_{pix} \\ T_{piy} \\ T_{piz} \end{Bmatrix} \quad \text{where,} \quad \begin{Bmatrix} T_{pix} \\ T_{piy} \\ T_{piz} \end{Bmatrix} = \mathbf{r}_{pi} \times \mathbf{F}_{pi} \quad \text{and} \quad \mathbf{F}_{pi} = \begin{Bmatrix} F_{pix} \\ F_{piy} \\ F_{piz} \end{Bmatrix} \tag{3.25}$$

Then, the Eqn. 3.25 may be written in matrix form as:

$$\begin{Bmatrix} T_{pix} \\ T_{piy} \\ T_{piz} \end{Bmatrix} = \mathbf{r}_{pi} \times \mathbf{F}_{pi} = \begin{bmatrix} 0 & -z_{pi} & y_{pi} \\ z_{pi} & 0 & -x_{pi} \\ -y_{pi} & x_{pi} & 0 \end{bmatrix} \begin{Bmatrix} F_{pix} \\ F_{piy} \\ F_{piz} \end{Bmatrix} \tag{3.26}$$

Therefore,

$$\begin{Bmatrix} F_{pix} \\ F_{piy} \\ F_{piz} \\ T_{pix} \\ T_{piy} \\ T_{piz} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -z_{pi} & y_{pi} \\ z_{pi} & 0 & -x_{pi} \\ -y_{pi} & x_{pi} & 0 \end{bmatrix} \begin{Bmatrix} F_{pix} \\ F_{piy} \\ F_{piz} \end{Bmatrix} \tag{3.27}$$

Noting that the matrix in the Eqn. 3.27, is the transpose of the matrix introduced in the Eqn. 3.21. The force equation is written as:

$$\mathbf{f}_{pi} = \mathbf{A}_i^T \mathbf{F}_{pil} \tag{3.28}$$

Note that $\mathbf{F}_{pil}$ is a force vector (acting at point ***p*** on body ***i***) described in the local axes frame. Normally force is conveniently described in the global axes. This equation can be re-written as:

$$\mathbf{f}_{pi} = \mathbf{A}_i^T \mathbf{T}_i^T \mathbf{F}_{pi(3)} \tag{3.29}$$

Now $\mathbf{F}_{pi\,(3)}$ is in the global axes frame of dimension 3, now pre multiplying the both side of the equation with $\mathbf{T}'$

$$\mathbf{F}_{pi} = \mathbf{T}'_i \, \mathbf{A}_i^T \mathbf{T}_i^T \mathbf{F}_{pi(3)} \tag{3.30}$$

The left hand side of the equation has dimension 6. Where;

$$\mathbf{T}_i' = \begin{bmatrix} \mathbf{T}_i & 0 \\ 0 & \mathbf{T}_i \end{bmatrix} \tag{3.31}$$

Now, forces acting on body ***i*** is written as (for this, point deflections are described in the global axes):

$$\mathbf{F}_{pi(3)} = \mathbf{k}_p \left(\mathbf{T}_j \mathbf{d}_{pj} - \mathbf{T}_i \mathbf{d}_{pi} \right) \tag{3.32}$$

$$\mathbf{F}_{pi(3)} = \mathbf{k}_p \left(\mathbf{T}_j \mathbf{A}_{pj} \mathbf{u}_j - \mathbf{T}_i \mathbf{A}_{pi} \mathbf{u}_i \right) \tag{3.33}$$

u Vectors are still in the local frame and it needs to be replaced by **U**.

$$\mathbf{F}_{pi(3)} = \mathbf{k}_p \left(\mathbf{T}_j \mathbf{A}_{pj} \mathbf{T}_j'^T \mathbf{U}_j - \mathbf{T}_i \mathbf{A}_{pi} \mathbf{T}_i'^T \mathbf{U}_i \right) \tag{3.34}$$

In this case;

$$\mathbf{T}_i' = \begin{bmatrix} \mathbf{T}_i & 0 \\ 0 & \mathbf{T}_i \end{bmatrix} \tag{3.35}$$

$\mathbf{k}_p$ is described in the global axes frame, $\mathbf{F}_{pi}$ is a force vector in the global axes frame, the following steps are self explanatory.

$$\mathbf{M}_i \ddot{\mathbf{u}}_i = \sum_p \mathbf{f}_{pi} \tag{3.36}$$

$$\mathbf{M}_i \mathbf{T}_i'^T \ddot{\mathbf{U}} = \sum_p \mathbf{f}_{pi} \tag{3.37}$$

$$\mathbf{T}'_i \mathbf{M}_i \mathbf{T}_i'^T \ddot{\mathbf{U}} = \mathbf{T}'_i \sum_p \mathbf{f}_{pi} \tag{3.38}$$

$$\mathbf{T}'_i \mathbf{M}_i \mathbf{T}_i'^T \ddot{\mathbf{U}} = \sum_p \mathbf{F}_{pi} \tag{3.39}$$

Substituting forces from the Eqn. 3.34 into the Eqn. 3.30 and into the Eqn. 3.39 for suffix *i* and *j*, the final equation of motions may be written as:

$$\mathbf{T}'_i \mathbf{M}_i \mathbf{T}_i'^T \ddot{\mathbf{U}}_i + \sum_p \left(\mathbf{T}'_i \mathbf{A}_i^T \mathbf{T}_i^T \mathbf{k}_p \mathbf{T}_i \mathbf{A}_{pi} \mathbf{T}_i'^T \right) \mathbf{U}_i - \sum_p \left(\mathbf{T}'_i \mathbf{A}_i^T \mathbf{T}_i^T \mathbf{k}_p \mathbf{T}_j \mathbf{A}_{pj} \mathbf{T}_j'^T \right) \mathbf{U}_j = 0 \tag{3.40}$$

$$\mathbf{T'}_j \mathbf{M}_j \mathbf{T'}_j^T \ddot{\mathbf{U}}_j - \sum_p \left(\mathbf{T'}_j \mathbf{A}_j^T \mathbf{T}_j^T \mathbf{k}_p \mathbf{T}_i \mathbf{A}_{pi} \mathbf{T}_i^{'T}\right)\mathbf{U}_i + \sum_p \left(\mathbf{T'}_j \mathbf{A}_j^T \mathbf{T}_j^T \mathbf{k}_p \mathbf{T}_j \mathbf{A}_{pj} \mathbf{T}_j^{'T}\right)\mathbf{U}_j = 0 \quad (3.41)$$

These are the final equations of motion described in matrix form ready to be assembled in the global matrix for the overall system.

In the implementation chapter (Chapter 4) further discussion will be given in simplifying the matrix for more efficient computation and it will be shown that the Eqn. 3.40 and Eqn.3.41 can be written in the following formats as:

$$\begin{bmatrix} \mathbf{m}_i & 0 \\ 0 & \mathbf{J}_i \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{x}}_i \\ \ddot{\boldsymbol{\theta}}_i \end{Bmatrix} - \begin{bmatrix} \mathbf{k}_p & \mathbf{k}_p \mathbf{R}_{pi} \\ \mathbf{R}_{pi}^{\mathrm{T}} \mathbf{k}_p & \mathbf{R}_{pi}^{\mathrm{T}} \mathbf{k}_p \mathbf{R}_{pi} \end{bmatrix} \begin{Bmatrix} \mathbf{x}_i \\ \boldsymbol{\theta}_i \end{Bmatrix} + \begin{bmatrix} \mathbf{k}_p & \mathbf{k}_p \mathbf{R}_{pj} \\ \mathbf{R}_{pi}^{\mathrm{T}} \mathbf{k}_p & \mathbf{R}_{pi}^{\mathrm{T}} \mathbf{k}_p \mathbf{R}_{pj} \end{bmatrix} \begin{Bmatrix} \mathbf{x}_j \\ \boldsymbol{\theta}_j \end{Bmatrix} = \begin{Bmatrix} \mathbf{F}_i \\ \mathbf{M}_i \end{Bmatrix} \quad (3.42)$$

$$\begin{bmatrix} \mathbf{m}_j & 0 \\ 0 & \mathbf{J}_j \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{x}}_j \\ \ddot{\boldsymbol{\theta}}_j \end{Bmatrix} + \begin{bmatrix} \mathbf{k}_p & \mathbf{k}_p \mathbf{R}_{pi} \\ \mathbf{R}_{pi}^{\mathrm{T}} \mathbf{k}_p & \mathbf{R}_{pi}^{\mathrm{T}} \mathbf{k}_p \mathbf{R}_{pi} \end{bmatrix} \begin{Bmatrix} \mathbf{x}_i \\ \boldsymbol{\theta}_i \end{Bmatrix} - \begin{bmatrix} \mathbf{k}_p & \mathbf{k}_p \mathbf{R}_{p_j} \\ \mathbf{R}_{p_j}^{\mathrm{T}} \mathbf{k}_p & \mathbf{R}_{p_j}^{\mathrm{T}} \mathbf{k}_p \mathbf{R}_{p_j} \end{bmatrix} \begin{Bmatrix} \mathbf{x}_j \\ \boldsymbol{\theta}_j \end{Bmatrix} = \begin{Bmatrix} \mathbf{F}_j \\ \mathbf{M}_j \end{Bmatrix} \quad (3.43)$$

Overall equations of motion are now completed. It is worth re-stating that the stiffness and the damping matrices are identical in their structure. In order to obtain the damping matrix all one needs to do is to replace the stiffness coefficients with corresponding damping coefficients.

3.2.3 Development of joint stiffness formulations

As the stiffness matrix is already assembled, it might be a convenient approach to use the existing stiffness matrix between articulating bodies in order to study the joint stiffness properties. It is appropriate to mention that in the proposed unconstrained joint model, the number of joint bones is not restricted to two. Thus the proposed joint modelling can be employed to model a joint complex which is composed of more than

two bones and more than one joint. For example, by means of human elbow joint, three bones (humerus, ulna and radius) and three joints are considered (ulna-humeral, radio-ulnar and radio-humeral joints). Therefore such a joint complex modelling can be performed through the proposed multi body joint model. In studying the joint laxity in terms of the joint stiffness, only two bones and one joint can be modelled at a time. Since the stiffness matrix formulation is provided by the Eqn. 3.40 and Eqn. 3.41, there is no need to obtain these from scratch. However, if these equations are utilized then elements in the matrix need to be reduced to represent a single joint which is needed to contain two joint bones such as body ***i*** and body ***j*** only. Thus, a 12 x 12 matrix obtained for these bodies. However this is further reduced to 6 x 6 Matrix by fixing one of the bodies. Let`s assume that body ***i*** is the mobile bone and body ***j*** is the fixed bone. Therefore, through eliminating the motion of the body ***j***, the Eqn. 3.40 is expressed for body ***i*** as:

$$\mathbf{T}'_i \mathbf{M}_i \mathbf{T}'^{T}_i \ddot{\mathbf{U}}_i + \sum_p \left(\mathbf{T}'_i \mathbf{A}_i^T \mathbf{T}_i^T \mathbf{k}_p \mathbf{T}_i \mathbf{A}_{pi} \mathbf{T}_i'^T \right) \mathbf{U}_i = 0 \tag{3.44}$$

It is possible to use this formulation as it is, especially if the system equations are already assembled, supplying the boundary condition should take care of any reduction required. Taking the stiffness element for body ***i*** only, in the new format introduced in the Eqn. 3.42.

$$\begin{bmatrix} \mathbf{k}_p & \mathbf{k}_p \mathbf{R}_{pi} \\ \mathbf{R}_{pi}^{\mathrm{T}} \mathbf{k}_p & \mathbf{R}_{pi}^{\mathrm{T}} \mathbf{k}_p \mathbf{R}_{pi} \end{bmatrix} \tag{3.45}$$

The question one that needs to answer is how to use the matrix such a way that it gives practical information with respect to joint laxity measurement. Firstly the stiffness needs to be measured to axes relative to the joint itself and translational stiffness (trying to pull the joint apart) has to be separated from the rotational ones. And the rotational

stiffnesses have to be identified relative to the instantaneous axis of rotation. In order to achieve this, the translational stiffness should not be coupled with the rotational stiffness. Although the Eigen value analysis of the stiffness matrix achieves this precisely, it is not always easy to give a physical interpretation to the Eigen values or vectors. However there is a more direct way of achieving this. This concept, to be called as the centre of stiffness (similar to centre of mass) and it has the property that stiffness formulation with respect to this centre will have translational stiffness completely uncoupled from the rotational stiffness. The calculation of the centre of the rotation can be defined as the centre, relative to which any rotation will not generate a resultant force. In vectorial terms this can be expressed as:

$$\boldsymbol{k}_i\left(\boldsymbol{\theta}\times(\mathbf{r}_i-\boldsymbol{\rho}_i)\right)=0 \tag{3.46}$$

Where $\boldsymbol{k}_i$ is the stiffness matrix of the i_{th} tissue in the global axis (this is expressed in the global axis for convenience before assembling the formulation for the sake of simplicity). $\boldsymbol{\theta}$ is the arbitrary rotation vector and $\mathbf{r}_i$ is the position of the i_{th} tissue attachment on body $\boldsymbol{i}$ (remembering that body $\boldsymbol{j}$ is fixed in space) all measured relative to the local body axes (with axes parallel to the global axes, which does not need any transformation). $\boldsymbol{\rho}$ is the position vector of the centre of stiffness. The equation now can be expressed, for all the stiffness elements as:

$$\sum_{alltissues} k_i(\boldsymbol{\theta}\times\mathbf{r}_i)-k_i(\boldsymbol{\theta}\times\boldsymbol{\rho})=0 \tag{3.47}$$

Now replacing the vector equation with its matrix equivalent is given as:

$$\sum_{alltissues}\left[\begin{bmatrix} k_{xxi} & k_{xyi} & k_{xzi} \\ k_{yxi} & k_{yyi} & k_{yzi} \\ k_{zxi} & k_{zyi} & k_{zzi} \end{bmatrix}\begin{bmatrix} 0 & z_{pi} & -y_{pi} \\ -z_{pi} & 0 & x_{pi} \\ y_{pi} & -x_{pi} & 0 \end{bmatrix}\right]\begin{Bmatrix} \boldsymbol{\theta}_{xi} \\ \boldsymbol{\theta}_{yi} \\ \boldsymbol{\theta}_{zi} \end{Bmatrix}=\left[\sum_{alltissues}\begin{bmatrix} k_{xxi} & k_{xyi} & k_{xzi} \\ k_{yxi} & k_{yyi} & k_{yzi} \\ k_{zxi} & k_{zyi} & k_{zzi} \end{bmatrix}\right]\begin{bmatrix} 0 & \boldsymbol{\rho}_z & -\boldsymbol{\rho}_y \\ -\boldsymbol{\rho}_z & 0 & \boldsymbol{\rho}_x \\ \boldsymbol{\rho}_y & -\boldsymbol{\rho}_x & 0 \end{bmatrix}\begin{Bmatrix} \boldsymbol{\theta}_{xi} \\ \boldsymbol{\theta}_{yi} \\ \boldsymbol{\theta}_{zi} \end{Bmatrix} \tag{3.48}$$

Which gives,

$$\begin{bmatrix} \sum\left(-z_{pi}k_{xyi}+y_{pi}k_{xzi}\right) & \sum\left(z_{pi}k_{xxi}-x_{pi}k_{xzi}\right) & \sum\left(-y_{pi}k_{xxi}+x_{pi}k_{xyi}\right) \\ \sum\left(-z_{pi}k_{yyi}+y_{pi}k_{yzi}\right) & \sum\left(z_{pi}k_{yxi}-x_{pi}k_{yzi}\right) & \sum\left(-y_{pi}k_{yxi}+x_{pi}k_{yyi}\right) \\ \sum\left(-z_{pi}k_{zyi}+y_{pi}k_{zzi}\right) & \sum\left(z_{pi}k_{zxi}-x_{pi}k_{zzi}\right) & \sum\left(-y_{pi}k_{zxi}+x_{pi}k_{zyi}\right) \end{bmatrix}$$
$$= \begin{bmatrix} -\rho_z\sum k_{xyi}+\rho_y\sum k_{xzi} & \rho_z\sum k_{xxi}-\rho_x\sum k_{xzi} & -\rho_y\sum k_{xxi}+\rho_x\sum k_{xyi} \\ -\rho_z\sum k_{yyi}+\rho_y\sum k_{yzi} & \rho_z\sum k_{yxi}-\rho_x\sum k_{yzi} & -\rho_y\sum k_{yxi}+\rho_x\sum k_{yyi} \\ -\rho_z\sum k_{zyi}+\rho_y\sum k_{zzi} & \rho_z\sum k_{zxi}-\rho_x\sum k_{zzi} & -\rho_y\sum k_{zxi}+\rho_x\sum k_{zyi} \end{bmatrix} \tag{3.49}$$

It is possible to set up 9 equations from the matrix equation (Eqn. 3.49), one for each element of the matrix. There are many combinations of 3 equations out of the possible 9, needed to calculate the three unknowns $\{\rho_x \quad \rho_y \quad \rho_z\}$. However not all 9 can be used for this. For example a set of equations given in the Eqn. 3.50 are ill conditioned. Equally any solution not having k_{xx} or k_{zz} in the final expression (i.e. the expression made of cross inertias only) is not acceptable since cross stiffnesses my not exist if the axes system chosen coincides with the principal axes of the spring (the cross stiffnesses all become zero).

$$\begin{bmatrix} 0 & \sum k_{yzi} & -\sum k_{yyi} \\ -\sum k_{yzi} & 0 & \sum k_{yxi} \\ \sum k_{yyi} & -\sum k_{yxi} & 0 \end{bmatrix} \begin{Bmatrix} \rho_x \\ \rho_y \\ \rho_z \end{Bmatrix} = \begin{Bmatrix} \sum\left(-z_{pi}k_{yyi}+y_{pi}k_{yzi}\right) \\ \sum\left(z_{pi}k_{yxi}-x_{pi}k_{yzi}\right) \\ \sum\left(-y_{pi}k_{yxi}+x_{pi}k_{yyi}\right) \end{Bmatrix} \tag{3.50}$$

The following equations are found to satisfy all the conditions for a valid solution.

$$\begin{aligned} -\rho_z\sum k_{yyi}+\rho_y\sum k_{yzi} &= \sum\left(-z_{pi}k_{yyi}+y_{pi}k_{yzi}\right) \\ -\rho_z\sum k_{zyi}+\rho_y\sum k_{zzi} &= \sum\left(-z_{pi}k_{zyi}+y_{pi}k_{zzi}\right) \end{aligned} \tag{3.51}$$

$$\begin{bmatrix} \sum k_{yzi} & -\sum k_{yyi} \\ \sum k_{zzi} & -\sum k_{zyi} \end{bmatrix} \begin{Bmatrix} \rho_y \\ \rho_z \end{Bmatrix} = \begin{Bmatrix} \sum \left(-z_{pi} k_{yyi} + y_{pi} k_{yzi} \right) \\ \sum \left(-z_{pi} k_{zyi} + y_{pi} k_{zzi} \right) \end{Bmatrix} \tag{3.52}$$

$$\begin{Bmatrix} \rho_y \\ \rho_z \end{Bmatrix} = \frac{1}{-\sum k_{yzi} \sum k_{zyi} + \sum k_{zzi} \sum k_{yyi}} \begin{bmatrix} -\sum k_{zyi} & \sum k_{yyi} \\ -\sum k_{zzi} & \sum k_{yzi} \end{bmatrix} \begin{Bmatrix} \sum \left(-z_{pi} k_{yyi} + y_{pi} k_{yzi} \right) \\ \sum \left(-z_{pi} k_{zyi} + y_{pi} k_{zzi} \right) \end{Bmatrix} \tag{3.53}$$

The formulation above ensures that there is always a solution provided, k_{zz}, k_{yy} exist as they both appear in the denominator and the nominator of the expression. And final from is given by:

$$-\rho_y \sum k_{yxi} + \rho_x \sum k_{yyi} = \sum \left(-y_{pi} k_{yxi} + x_{pi} k_{yyi} \right) \tag{3.54}$$

$$\rho_x = \frac{1}{\sum k_{yyi}} \left(\rho_y \sum k_{yxi} + \sum \left(-y_{pi} k_{yxi} + x_{pi} k_{yyi} \right) \right) \tag{3.55}$$

Where, again the nominator and denominator contain diagonal stiffness elements which ensure that there is always a feasible solution. To illustrate what is happening, one can consider a single spring with its principal axes coinciding with the global axes. This way one can see what may happen in a situation where matrix is reduced to following form as:

$$\begin{bmatrix} k_{px} & 0 & 0 & 0 & k_{px}z_p & -k_{px}y_p \\ 0 & k_{py} & 0 & -k_{py}z_p & 0 & k_{py}x_p \\ 0 & 0 & k_{pz} & k_{pz}y_p & -k_{pz}x_p & 0 \\ 0 & -k_{py}z_p & k_{pz}y_p & k_{pz}y_p^2 + k_{py}z_p^2 & -k_{pz}x_p y_p & -k_{py}x_p z_p \\ k_{px}z_p & 0 & -k_{pz}x_p & -k_{pz}x_p y_p & k_{pz}x_p^2 + k_{px}z_p^2 & -k_{px}y_p z_p \\ -k_{px}y_p & k_{py}x_p & 0 & -k_{py}x_p z_p & -k_{px}y_p z_p & k_{py}x_p^2 + k_{px}y_p^2 \end{bmatrix} \tag{3.56}$$

Here it is easy to see that shifting the axes system to the centre of stiffness involves zeroing all the elements of the top and bottom of the non diagonal sub matrices. Assuming that there is *p* number of such element as:

$$\rho_z = \frac{\sum_p k_{px}z_p}{\sum_p k_{px}} \quad \rho_y = \frac{\sum_p k_{px}y_p}{\sum_p k_{px}} \quad \rho_x = \frac{\sum_p k_{py}x_p}{\sum_p k_{py}} \tag{3.57}$$

Once this is calculated, the Eigen values of the lower 3x3 matrix of the re assembled stiffness matrix gives the principal joint stiffnesses and the Eigen vectors gives the axes of articulation. Summarising the procedure, initially there is a stiffness matrix as:

$$\begin{bmatrix} \mathbf{k}_p & \mathbf{k}_p\mathbf{R}_{pi} \\ \mathbf{R}_{pi}^{\mathbf{T}}\mathbf{k}_p & \mathbf{R}_{pi}^{\mathbf{T}}\mathbf{k}_p\mathbf{R}_{pi} \end{bmatrix} \tag{3.58}$$

From the matrix formulation above, the centre of stiffness is given by:

$$\sum_p \left[\mathbf{k}_{\mathrm{p}}\mathbf{R}_{\mathrm{pi}}\right] = \left[\sum_p \mathbf{k}_{\mathrm{p}}\right]\boldsymbol{\rho} \tag{3.59}$$

The torsional stiffnesses and the principal axes of rotation are given by:

$$\left[\mathbf{R}_{\mathrm{pi}}^{\mathrm{T}}\mathbf{k}_{\mathrm{p}}\mathbf{R}_{\mathrm{pi}} - \lambda\mathbf{I}\right]_{\rho}\mathbf{X} = 0 \tag{3.60}$$

Where λ gives the principal stiffnesses and $\mathbf{X}$ gives the principal axes of stiffness, this is expected to coincide with the axes or articulation. The suffix $\boldsymbol{\rho}$ indicates that the Eigen value/vector analysis is to be performed on recalculated stiffness matrix for the new tissue attachment coordinates translated by $\boldsymbol{\rho}$.

3.2.4 Nonlinear joint modelling

Most of the biological tissues and engineering materials behave nonlinearly. However, considering the nonlinearity may cause problems in the analysis. Due to the complexity of handling nonlinear analysis, most of the material nonlinearities have been treated as linear. However, ever-growing number-processing power of computers yields to handle complex nonlinear behaviour of materials with nonlinear geometric effects through new developments in non-linear modelling theories. The predicted nonlinear behaviour may actually be desirable for the design objective. Introducing the nonlinearity can predict the certain response of a mechanism. As mentioned in Section 2.2.1 and Section 2.2.2, it is well acknowledged that musculoskeletal tissues behave nonlinearly. Through the comprehensive literature survey on the nonlinear behaviour of musculoskeletal tissues and their modelling applications, it has been decided to develop a new polynomial formulation to express the nonlinear material behaviour. In the proposed nonlinear joint formulation, both material and geometric nonlinearities are considered where the nonlinear force-deflection behaviour of spring can account both for material and geometrical nonlinearities. In musculoskeletal, nonlinear tissue behaviour has been also related to the geometric nonlinearities due to large changes in its cross section. Therefore, during large geometric displacements and rotations of joints, the proposed nonlinear model of springs yields high stiff spring responses to prevent gaining large momentum to prevent serious tissue damage. It is now appropriate to mention that due to rigid body model of bones, bone deformations are neglected and deformation is only specialized to tissues. Additionally, it is assumed that deflections in the three principal axes of a spring remain orthogonal and orthogonality is preserved through the deflection cycle. Typical non-linear deflections

in the principal axes of a spring are shown in the Figure 3. 2. In all types of soft tissue modelling, the stiffness assumed to be axi symmetric with the main stiffness in the longitudinal axis (axis of the main tension) with having little stiffness in the other two axes.

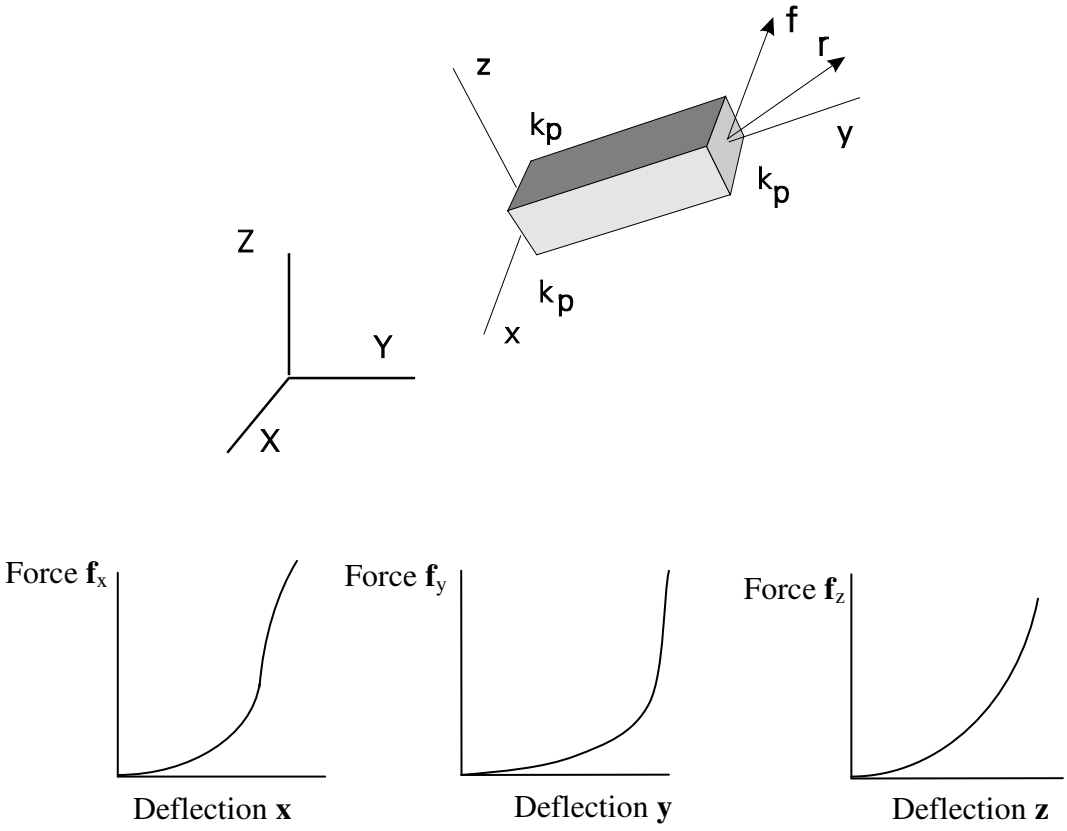


Figure 3.2 Non-linear deflections in three orthogonal axes of mounting

3.2.4.1 Formulation of nonlinear spring elements

If the stiffness characteristics are nonlinear this can conveniently be expressed by the force deflection curve. According to the assumption given previously, the deflection in three orthogonal planes remains uncoupled. The proposed model of force deflection relations in three orthogonal planes are expressed as:

$$\mathbf{f}_x = \mathrm{k}_{x1}\mathbf{x} + \mathrm{k}_{x2}\mathbf{x}^2 + \mathrm{k}_{x3}\mathbf{x}^3 + ... + \mathrm{k}_{xn}\mathbf{x}^n \tag{3.61a}$$

$$\mathbf{f}_y = \mathrm{k}_{y1}\mathbf{y} + \mathrm{k}_{y2}\mathbf{y}^2 + \mathrm{k}_{y3}\mathbf{y}^3 + ... + \mathrm{k}_{yn}\mathbf{y}^n \tag{3.61b}$$

$$\mathbf{f}_z = \mathrm{k}_{z1}\mathbf{z} + \mathrm{k}_{z2}\mathbf{z}^2 + \mathrm{k}_{z3}\mathbf{z}^3 + ... + \mathrm{k}_{zn}\mathbf{z}^n \tag{3.61c}$$

In other words, three force equations are described in terms of a series of spring parameters with higher order terms of deflection in the principal axes directions. No equation contains any variable from any other two directions. This representation scheme is generally found to be satisfactory for many practical applications. Matrix representation of this is possible, since the stiffness coefficient matrices of the displacements or their higher order terms are all diagonal.

$$\mathbf{f} = \mathbf{k}_1\mathbf{x} + \mathbf{k}_2\mathbf{x}^2 + \mathbf{k}_3\mathbf{x}^3 + ... + \mathrm{k}_n\mathbf{x}^n \tag{3.62}$$

$\mathbf{x}$, $\mathbf{x}^2$, $\mathbf{x}^3$ and higher order terms are vectors containing terms such as $\mathbf{f} = \left(\boldsymbol{f}_x\ \boldsymbol{f}_y\ \boldsymbol{f}_z\right)$ $\mathbf{x}^n = \left(\boldsymbol{x}^n\ \boldsymbol{y}^n\ \boldsymbol{z}^n\right)$. In order to facilitate assembly of stiffness elements to the global matrix, a typical conversion of the type is needed which is:

$$\mathbf{F} = \mathbf{K}_1\mathbf{X} + \mathbf{K}_2\mathbf{X}^2 + \mathbf{K}_3\mathbf{X}^3 + ... + \mathbf{K}_n\mathbf{X}^n \tag{3.63}$$

Unfortunately such transformation is neither possible nor meaningful. This is one of the important reasons why nonlinearity is generally formulated based on piecewise linearization. Although the piecewise linearization ensures that tensorial operation is

applicable and the formulations given for linear motion can be used, the system is now described by one of *S* number of stiffness combinations, where S is given by:

$$S = \prod_{s=1}^{n} r_s \tag{3.64}$$

Where, $\boldsymbol{r}_s$ gives the number of linear segments for s number of spring. A single mass with four mounts and each with five linear segments on each, the formulation based on piecewise linearization would require 625, 6x6 stiffness matrices. Although in terms of memory allocation this may not be too unmanageable, during the simulation process at each time step the stiffness matrix will need to be re-assembled by selecting the correct segments and of course this will slow the simulation.

3.2.4.2 A non-matrix based motion analysis

The non matrix based motion analysis is essentially based on the free body diagrams and the internal reactions explicitly appear in the equations of motion. Each body possesses six degrees of freedom and spring reactions (and other forces acting on it) are treated as external forces. A typical free body diagram of bodies ***i*** and ***j*** is shown in the Figure 3.3. Axes systems $O_i x_i y_i z_i$ and $O_j x_j y_j z_j$ are fixed to the bodies ***i*** and ***j*** respectively. The orientation of the body is measured in terms of the Euler angles. In this analysis, it is assumed that the local axes and the principal axes coincide. Spring force acting at a point on each body is due to relative displacements of the ends of the spring. In a system with body ***i*** and ***j***, this is determined due to motion of the attachment positions of the spring on body ***j*** relative to body ***i***.

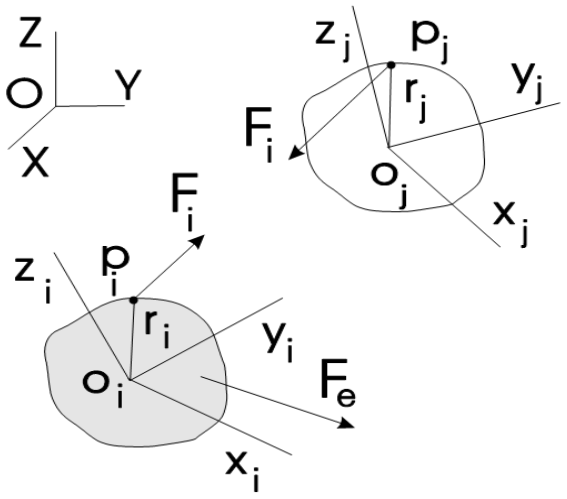


Figure 3.3 Free body diagram of body ***i*** and body ***j***

Basically, equations of motion which describes the motion of body ***i*** is given by:

$$\mathbf{M}_i\ddot{\mathbf{x}}_i = \mathbf{F} \tag{3.65}$$

Where

$$\mathbf{F} = \mathbf{F}_e + \mathbf{F}_i \tag{3.66}$$

In Equation 3.65, suffices ***e*** and ***i*** mean the external and the internal forces (not to be confused with the body index, *i*). Where $\mathbf{F}_i$ is written by Equation 3.67 as:

$$\mathbf{F}_i = \mathbf{k}_1\mathbf{x} + \mathbf{k}_2\mathbf{x}^2 + \mathbf{k}_3\mathbf{x}^3 + \ldots + \mathbf{k}_n\mathbf{x}^n \tag{3.67}$$

x is described as the displacement of springs along its principal axes. This is important because it is assumed that experimental data exist for displacement of soft tissue along the main direction of its stretch. The lateral stiffnesses are taken to be much smaller than the main tensile stiffness.

3.2.4.3 Displacements of a point on a body

Force acting on a rigid body at a point **p** due to springs and dampers is a function of the displacement and velocity of that point. This may be described by:

$$\mathbf{r}_p = \mathbf{R}_i + \mathbf{r}_i \tag{3.68}$$

Where $\mathbf{R}_i$ is the position vector of the centre of origin of the axes system fixed on the rigid body and $\mathbf{r}_i$ is the position vector of a point on the body relative to the moving axes system. In order to analyse the dynamics of a multi body system, interaction of at least two bodies should be considered. Let us assume that these bodies are numbered as i and j. $\mathbf{p}_i$ and $\mathbf{p}_j$ are two points on these bodies as shown in Figure 3.3. In order to analyse the motion of body i and body j, the internal forces acting on individual bodies due to their motion relative to each other needs to be expressed. The motion of the origin of the axes system i is given by (x_i, y_i, z_i), and the angular rotation of axes is given by (α_i, β_i, γ_i). Similarly the motion of body j is described by (x_j, y_j, z_j) and (α_j, β_j, γ_j). Thus the displacements $\mathbf{d}_i$ and $\mathbf{d}_j$ are expressed as:

$$\mathbf{d}_i = (x_i\ y_i\ z_i) + (\alpha_i\ \beta_i\ \gamma_i) \times (a_i\ b_i\ c_i) \tag{3.69}$$

$$\mathbf{d}_j = (x_j\ y_j\ z_j) + (\alpha_j\ \beta_j\ \gamma_j) \times (a_j\ b_j\ c_j) \tag{3.70}$$

In order to calculate reactions on each body, relative displacements between connecting points needs to be calculated. The relative displacement in the global axes frame is given by;

$$\mathbf{d}_g = \mathbf{T}_j \mathbf{d}_j - \mathbf{T}_i \mathbf{d}_i \tag{3.71}$$

Or

$$\mathbf{d}_{\mathrm{g}} = \mathbf{d}_{gj} - \mathbf{d}_{gi} \tag{3.72}$$

Where, $\mathbf{T}_i\mathbf{d}_i$ and $\mathbf{T}_j\mathbf{d}_j$ transform the local displacements $\mathbf{d}_i$ and $\mathbf{d}_j$ into the global axes frame which are represented as $\mathbf{d}_{gi}$ and $\mathbf{d}_{gj}$ in the Eqn. 3.72. After expressing the end displacements in the global axes frame, then the stretch can be expressed in the global axes frame as $\mathbf{d}_{\mathrm{g}}$. However, it is more convenient to deal with the nonlinearity in the principal axes system of springs. Therefore deflection or the stretch in the reference (local) frame of the spring is given by:

$$\mathbf{d} = \mathbf{T}_{\mathbf{s}}^{\mathbf{T}}\mathbf{d}_{gj} - \mathbf{T}_{\mathbf{s}}^{\mathbf{T}}\mathbf{d}_{gi} \tag{3.73}$$

This can also be given by:

$$\mathbf{d} = \mathrm{T}_s^T\mathbf{d}_g \tag{3.74}$$

Where, T_s^T is the transformation matrix which transforms the $\mathbf{d}_g$ from global to local axes frame of the spring. Then the polynomial expression for displacements may be defined. If $\mathbf{d} = (x\ y\ z)$ then forces acting in the spring (or soft tissue) can be calculated in the local axes frame of the spring which has been provided by the Eqn. 3.67 (force acting on body ***i***) as:

$$\mathbf{F}_{si} = \mathbf{k}_1\mathbf{d} + \mathbf{k}_2\mathbf{d}^2 + \mathbf{k}_3\mathbf{d}^3 + ... + \mathbf{k}_n\mathbf{d}^n \tag{3.75}$$

And on body ***j***

$$\mathbf{F}_{sj} = -\left(\mathbf{k}_1\mathbf{d} + \mathbf{k}_2\mathbf{d}^2 + \mathbf{k}_3\mathbf{d}^3 + ... + \mathbf{k}_n\mathbf{d}^n\right) \tag{3.76}$$

Now these forces need to be transferred from the local axes frame of the spring to the body frames ***i*** and ***j*** respectively.

$$\boldsymbol{F}_i = \boldsymbol{T}_i^T \boldsymbol{T}_s \boldsymbol{F}_{si} \tag{3.77}$$

$$\boldsymbol{F}_j = \boldsymbol{T}_j^T \boldsymbol{T}_s \boldsymbol{F}_{sj} \tag{3.78}$$

In the Eqn. 3.77 and Eqn. 3.78, transformation of forces from the local axes frame of springs to the global (by $\boldsymbol{T}_s$) and from the global to the local body axes frame (by $\boldsymbol{T}_j^T$) is carried out. Having calculated the forces on bodies, now moments can also be calculated. Each moment equation is carried out in each body axes frame.

$$\mathbf{M}_i = \mathbf{r}_i \times \mathbf{F}_i \tag{3.79}$$

And

$$\mathbf{M}_j = \mathbf{r}_j \times \mathbf{F}_j \tag{3.80}$$

If the system is linear then the equations of motion can be compiled to obtain the overall stiffness matrix. However in nonlinear systems, no equation of motion compiled in matrix form, forces are available only in numerical form. Provided displacements and velocities of attachment points of springs are known. The numerical integration can proceed, as the numerical integration techniques such as Runge Kutta, requires velocities and displacements for integration. Such numerical approaches enable us to analyse motion of a multi-body system in the time domain however this may complex the non-linearity. The main steps of this approach are given in the flow diagram in Figure 3.4. Compared to the piecewise linearization, this method needs no checking which linear segments should be added to the stiffness matrix, since the

stiffness or inertia matrix is never assembled. The inertia matrix may be needed if one does not choose to use the principal axes systems of individual bodies as the local axes systems of the bodies. However, operating in the principal axes systems, each equation of each body (six for each) is to be divided with appropriate mass or inertia elements to calculate their corresponding accelerations:

$$\ddot{x}_i = \frac{F_{xi}}{m_i}, \ddot{y}_i = \frac{F_{y_i}}{m_i}, \ddot{z}_i = \frac{F_{z_i}}{m_i}, \ddot{\theta}_{xi} = \frac{T_{x_i}}{I_{xx_i}}, \ddot{\theta}_{y_i} = \frac{T_{y_i}}{I_{yy_i}}, \ddot{\theta}_{z_i} = \frac{T_{z_i}}{I_{zz_i}} \tag{3.81}$$

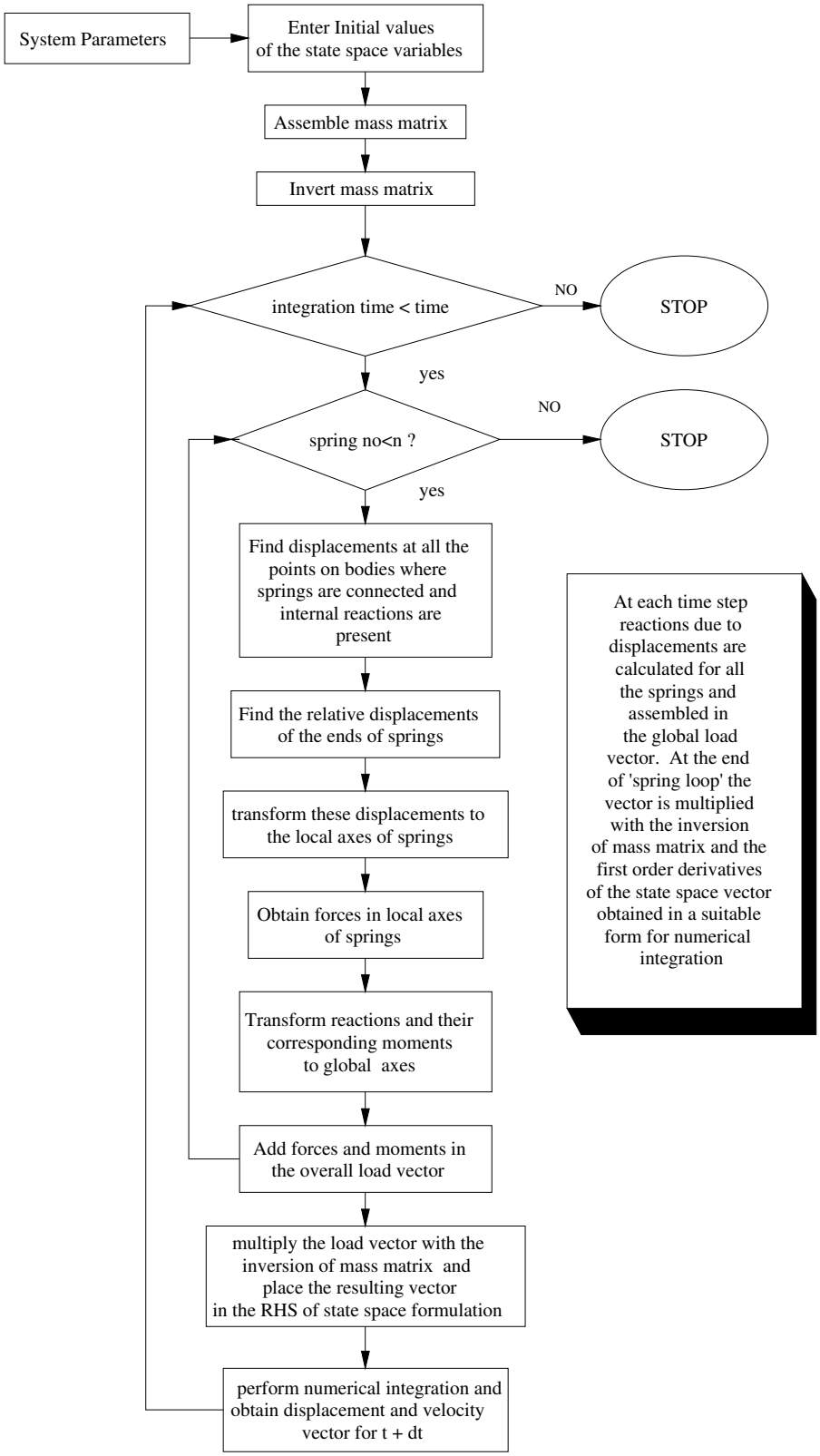


Figure 3.4 Flow diagram of a program for analysing a multi body system incorporating springs with polynomial type of nonlinearity

In the flow diagram, steps of the analysis are illustrated. These steps start with finding displacements for each tissue attachment points for ligament, tendon, and muscle and also include the cartilage contact model (cartilage contacts are geometrically selected as number of distance pairs less than a predefined distance). Then relative displacements are obtained between attachment points. When this is implemented for bone surfaces this step becomes more complicated as the displacement is not the Euclidian distance between the displacements of these points. The effective stretching has to account for the tissue wrapping around bone surfaces. This is to be explained in the next section as well as in the software implementation chapter (Chapter 4). After then, displacements are transformed to the local axes frames of each spring which represents ligament, tendon, muscle and cartilage. As the stiffness and damping properties of individual tissue elements are described in their principal axes frames, forces relative to the principal axes can easily be calculated. After obtaining forces in local tissue axes frame, they are transformed into the global axes and from the global axes they are again transformed to the individual body axes systems. Forces are then divided with the appropriate mass/inertia elements and added to the state space formulation. Thus, numerical integration is performed and new displacement and velocity vectors are obtained. These to be converted to the global axes frame (although not needed for dynamics analysis) for surface calculations. In the integration of the equations of motion, these steps are performed at each time interval. Note that, if the local axes systems of individual bodies coincide with their principal axes then the mass matrix inversion and mass matrix assembly are not needed.

Both for the linear formulations and the procedural solution for the non-linear formulations to perform, force action points of springs (tissues) need to be known. In biological systems this information is not available as a simple coordinate on the bone surface. Very rarely, attachment is a single point, in the most cases attachment (origin and insertion points) spreads over an area. In our modelling, it will be assumed that the mean attachment point can be found or multiple tissues can be attached to represent multiple insertion or origin points. The second problem relates to the geometry of the objects interaction. The tendons articulating a joint because of muscle force

generations, do so by pulling bones at the attachment points (in our equations, the positions of $\mathbf{p}_i$ and $\mathbf{p}_j$), the line of action should go through these points provided there is no other surface intersecting the line. However if there is any soft or hard tissue between those points then the connecting tissue "wraps round" the obstacle(s). Therefore the next section deals with this problem of calculating the effective line of force action in the presence of obstructive surfaces or objects. This is also known as the obstacle set problem.

3.3 Development of geometric tissue path finding, line of action and wrapping algorithms

In Section 2.2.3, the importance of describing shortest tissue paths and tissue attachments as well as the significance of wrapping has been widely addressed. According to the background information, it is well acknowledged that accurate tissue path estimations and the moment arm calculations are crucial for understanding of the joint mobility. An accurate moment arm is necessary for the implant design, implant insertion and planning surgical procedures. It should be kept in mind that an accomplished musculoskeletal joint model should facilitate the analysis for calculation of the shortest tissue path, tissue wrapping and accurate contact modelling in describing the joint in anatomic fidelity. Thus, through this section, the intention is to introduce how the shortest tissue path is found and wrap around bony or other tissue surfaces and how these are effective factors in the proposed joint model. In order to perform these desired evaluations, new algorithms have been developed and implemented. More detailed information about the software implementation is provided in the Chapter 4. In this section, the formulations necessary for the construction of tissue shortest path and the wrapping algorithm is provided. As there is no analytical shortest path theory applicable for the free surfaces, what is offered here is a heuristic algorithm based on geometry alone with no surface friction. After introducing the algorithm, its integration with the proposed dynamic analysis will be explained.

3.3.1 Minimum mass and residual covariance algorithms

The proposed heuristics assume that, starting from a point, traversing the landscape towards the end point the shortest path should follow the path which is closest to the

straight line between these two points. It is further assumed that the path follows the surface with the minimum covariance (squared perpendicular distance to the straight line) or minimum surface mass (total number of vertices along the path traversed). The initial path is estimated by finding the cross section properties (minimum mass or covariance) of the surface at the equal angular intervals about an axis coinciding with the vector along the two points. It is, however, possible that having started travelling from the starting point with the selected angle, the minimum path may change for the remaining journey. Therefore residual covariance has to be tested at equal interval along the journey to decide if the path is to be varied. Further information is to be given in the Chapter 4. Here the main formulation necessary to accomplish the algorithm, such as plane to surface intersection will be given.

In anatomic field, tissue attachment points are known as origin and insertion points which have been also referred to as the end points of springs or spring attachment points in this thesis. In order to generate a tissue path from the proposed algorithms, the origin and the insertion points are chosen on surface(s), these points are initially chosen interactively from the screen but saved in a data base for the future use. Geometric surfaces are described as triangulated mesh surfaces consist of triangles and points. For the muscle wrapping process of a specific muscle to start, the coordinates (in the global axis) of the attachment points are selected. First of all, a straight line is drawn between attachment points along the surfaces. After describing the attachment points and drawing the straight line, a local cylindrical coordinate system has to be defined for the tissue path. The axes system is defined in the global coordinate system. The origin of a local cylindrical coordinate system is usually assumed to be at the origin point of tissue attachments. The local axes system is described by the orthogonal vectors **u**, **v** and **w,** the origin of which is located at the first tissue attachment point. Let us assume that the $\boldsymbol{l}$ vector joins the two attachment points, $\mathbf{r}_a$ and $\mathbf{r}_b$. **v** vector is taken to be the unit vector along $\boldsymbol{l}$ as shown in the Figure 3.5.

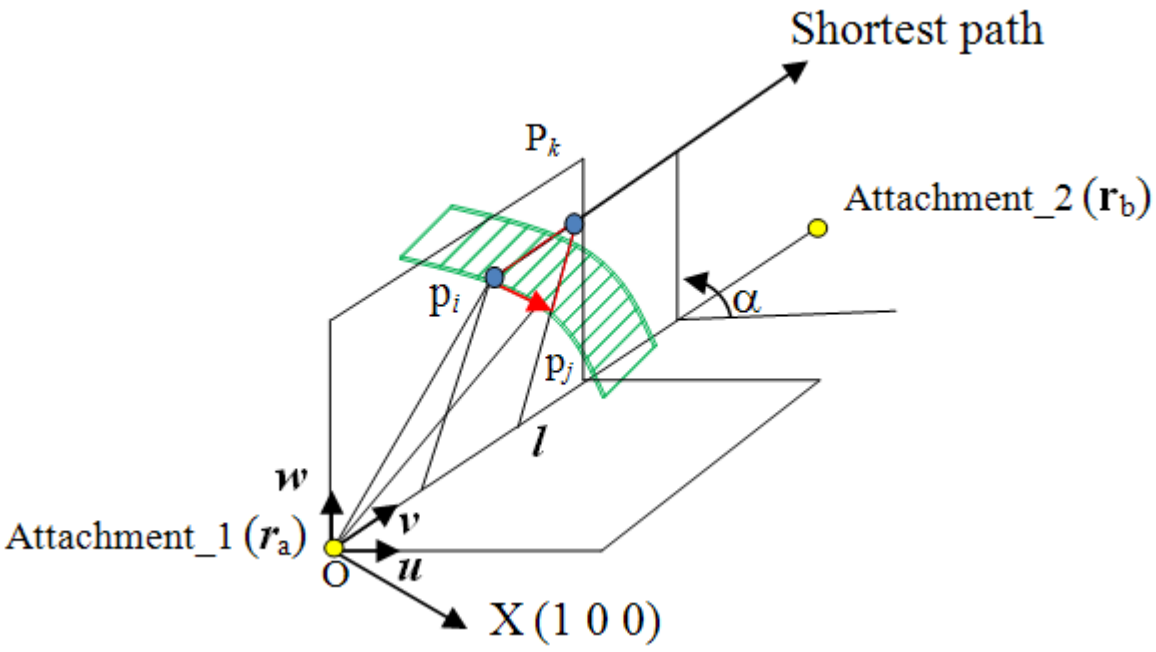


Figure 3.5 Illustration of proposed shortest path between attachment points

$\boldsymbol{l}$ vector is described between origin point $\mathbf{r}_a$ and insertion point $\mathbf{r}_b$ as:

$$\boldsymbol{l} = \mathbf{r}_b - \mathbf{r}_a \tag{3.82}$$

In Euclidean coordinates (R^3) the $\boldsymbol{l}$ vector can be described between the attachment points as:

$$\boldsymbol{l} = ((x_b - x_a),(y_b - y_a),(z_b - z_a)) \tag{3.83}$$

In order to establish the local coordinate system, $\boldsymbol{u}$, $\boldsymbol{v}$ and $\boldsymbol{w}$ vectors need to be unit vectors. The unit vector $\boldsymbol{v}$ is given by:

$$\boldsymbol{v} = \frac{(\mathbf{r}_b - \mathbf{r}_a)}{\left|\mathbf{r}_b - \mathbf{r}_a\right|} = \frac{(x_b - x_a),(y_b - y_a),(z_b - z_a)}{\sqrt{(x_b - x_a)^2 + (y_b - y_a)^2 + (z_b - z_a)^2}} \tag{3.84}$$

In order to obtain the **w** vector perpendicular to **v** vector, an arbitrary vector **X** as **X =** (1 0 0) is introduced. The cross product of **X** vector and ***v*** unit vector gives the **w** vector as:

$$\mathbf{w} = \mathbf{X} \times \boldsymbol{v} \tag{3.85}$$

In programming terms, it must be checked that ***v*** vector does not coincide with the **X** vector. If it does then another arbitrary vector is to be chosen. If this happens the program chooses **X** = (0 1 0) as the second option, both cannot be aligned with the ***v*** axis so no further test is necessary. It should be noticed that the unit vector ***w*** is not directly obtained from the cross product above, as they are not perpendicular vectors and even though each has unity magnitude, the cross product may not be a unit vector. Thus the unit vector ***w*** is extracted from the Eqn. 3.86 as:

$$\boldsymbol{w} = \frac{\mathbf{X} \times \boldsymbol{v}}{\left|\mathbf{X} \times \boldsymbol{v}\right|} \tag{3.86}$$

Furthermore, the cross product of ***v*** and ***w*** unit vectors directly gives the ***u*** unit vector as:

$$\boldsymbol{u} = \boldsymbol{v} \times \boldsymbol{w} \tag{3.87}$$

Now, all the unit vectors are extracted. Through considering that the surface consists of triangles and points, before running algorithms, it is required to obtain projections of points between origin and insertion points on the bone surfaces described in terms of a local cylindrical coordinate system, with the $\boldsymbol{p}_a$ at the origin and $\boldsymbol{v}$ along the central axis of the cylinder. These projections need to be obtained on the O$\boldsymbol{uw}$ plane. Initially, the projection of an arbitrary point is obtained on $\boldsymbol{u}$ axis and $\boldsymbol{w}$ axis separately. Let's assume that an arbitrary point $\mathbf{r}_i$ is chosen in the point cloud. A straight line between the origin point $\mathbf{r}_a$ and an arbitrary point $\mathbf{r}_i$ can be defined by $\boldsymbol{l}_i$ vector as:

$$\boldsymbol{l}_i = \mathbf{r}_i - \mathbf{r}_a \tag{3.88}$$

The projection of point $\mathbf{r}_i$ on u is obtained by dot product of $\boldsymbol{l}_i$ vector and $\boldsymbol{u}$ unit vector as:

$$p_u = (\boldsymbol{r}_i - \boldsymbol{r}_a) \cdot \boldsymbol{u} \tag{3.89}$$

Likewise, the projection of point $\mathbf{r}_i$ on w axis is obtained by dot product of $\boldsymbol{l}_i$ vector and $\boldsymbol{w}$ unit vector as:

$$p_w = (\boldsymbol{r}_i - \boldsymbol{r}_a) \cdot w \tag{3.90}$$

In order to apply the proposed algorithms, point projections of all points relative to O$\boldsymbol{uw}$ plane need to be extracted. Where, each point has its own projection angle α_i which is the angle measured from positive $\boldsymbol{u}$. The angle α_i can be obtained by tangent (α_i) which is measured between $\boldsymbol{u}$ and $\boldsymbol{w}$ axis as follows.

$$\tan(\alpha_i) = \frac{p_w}{p_u} \tag{3.91}$$

$$\alpha_i = \tan^{-1} \frac{p_w}{p_u} \tag{3.92}$$

Thus the first part of the procedure starts with segment slicing. The surface between the attachment points ($\mathbf{r}_a$ and $\mathbf{r}_b$) is divided into equal segments where each segment is assumed to be 10^0 along the ***l*** vector and about the O***uw*** plane. Thus the bone surfaces are divided into 36 segments around 360 degrees. Therefore, having obtained the projection angle α_i for each point on the surface, the calculation of which point belongs to which segment can be performed. After separating point clouds into segments, total number of points on each segment is calculated. This is the measure of the mass value of each segment and it is used as a measure to determine the less costly slice to be followed. In addition to total number of point calculations, segment covariance is also calculated. Again this is another measure of the distance function. Thus for the minimum mass/covariance algorithm to work, points are assumed to be uniformly distributed on the surface. This is generally true as they were pre-processed on the Geomagic software to ensure the uniform distribution, this will further be explained in the results chapter (Chapter 5). According to the proposed minimum mass algorithm, potential path segments which contain minimum number of points are chosen. Then, the segment which has minimum number of points can be established as a reference segment for the shortest path. The second proposed algorithm, the residual covariance algorithm which is briefly explained above, is expected to work better when surface profile exhibits high amplitude oscillations. In this case the height (from the ***v*** vector) of each segmented (slice) surface is calculated based on the projections of each point on the plane O***uw***. The square of this height is the measure of the surface covariance. The covariance for each slice can be calculated as, where the summations sign refers to the total number of points in the given slice:

$$\text{cov} = \sum (p_w^2 + p_u^2) \tag{3.93}$$

Hence the segment with minimum covariance can be accepted as the potential reference surface where the shortest path can pass through. According to the performed numerical experiments, for almost all the cases studied, the same segment is identified by minimum mass and covariance algorithms as the reference segment for the shortest path. Even in few cases where different segments are identified, these slices are visually appeared to be almost identical. Although the covariance is a better measure, if the points are not evenly distributed, there is no significant advantage over the simple point summation method if spread is uniform. Having obtained the particular segment for the proposed shortest path, the convex hull algorithm must be followed along with the residual covariance algorithm. Thus, driving the convex hull algorithm allows wrapping around the triangulated mesh surface through the proposed shortest path segment. The residual covariance algorithm assumes that the shortest path is a convex hull line on the mid plane of the reference slice. In other word this is a 2D convex hull algorithm. Another property of the convex hull line is that it starts from the tissue attachment point and finishes at the other end of the tissue. In other word, the attachment points are on the convex hull. For each segment (slice) the path described by the mid plane and the surface is followed. For this, the mid plane between the angles α_i and α_{i+1} needs to be defined. This plane is assumed to pass at angle α_i+5 with remembering that each segment is 10^0 where α_i relates to the starting angle and α_{i+1} relates to the end angle of the segment *i* which is $\alpha_i + 10^0$.

Up to this point the main parts of the algorithm is explained, the only unexplained is the "residual" term. To explain why "residual" is used as a term, consider a situation where the path start from the origin point and following the minimum covariance (or mass) slice, now the path comes to a stage where it has to climb a high hill, it is only logical to change to the next slice if it could go round the hill. Ideally algorithm should provide mechanism to "go around" the hill section without climbing it. The proposed covariance algorithm, as well as slicing the surface, it segments the progress along its main direction (into stages) checking the neighbouring paths if they are shorter to the target (thus the title of the algorithm, "residual covariances") enabling diversion from the current trajectory and updating the current trajectory at each stage. However the algorithm does not only check the residuals of the neighbouring slices but also checks the cost of changing (sideway motion involves additional journey).

In order to implement the algorithm the mid plane, surface intersection curve must be obtained. For this all the edge segments of mesh primitives (in our case triangles) needs to be intersected with the plane. Let`s take two arbitrary points such as point $\mathbf{r}_i$ and $\mathbf{r}_j$ describing a edge segment and establish if it is intersected with the plane in question (conveniently described in the cylindrical coordinates). This can be accomplished by checking if the plane intersects the line going through these two points, between the points, outside intersections are not valid.

In order to calculate the line segment plane intersection, let us describe the vector $\boldsymbol{l}_j$ between the origin point $\mathbf{r}_a$ and the point $\mathbf{r}_j$.

$$\boldsymbol{l}_{\mathrm{i}} = \mathbf{r}_i - \mathbf{r}_a \tag{3.94}$$

$$\boldsymbol{l}_j = \mathbf{r}_j - \mathbf{r}_a \tag{3.95}$$

And then the vector $\boldsymbol{l}_p$ of the intersection point is also required to be described. In this case, the λ value determines the point of intersection.

$$\boldsymbol{l}_p = \boldsymbol{l}_i + \lambda(\boldsymbol{l}_j - \boldsymbol{l}_i) \tag{3.96}$$

If the λ value for the line segment mid plane intersection is between 0 and 1, then the point $\mathbf{r}_p$ will be between the points $\mathbf{r}_i$ and $\mathbf{r}_j$ and on the mid plane, thus on the path. In order to calculate the $\mathbf{r}_p$ and λ, the procedure starts with defining the projection points (projection of the point p on the O$\boldsymbol{uw}$ plane) and the segment angles. Let`s assume that projection angle of point $\mathbf{r}_p$ is α_p, then equations are followed as:

$$\tan\alpha_p = \frac{\boldsymbol{p}_w}{\boldsymbol{p}_u} = \frac{\boldsymbol{l}_{\mathrm{p}} \cdot \boldsymbol{w}}{\boldsymbol{l}_{\mathrm{p}} \cdot \boldsymbol{u}} \tag{3.97}$$

$$\alpha_p = \tan^{-1} \frac{\boldsymbol{p}_w}{\boldsymbol{p}_u} \tag{3.98}$$

$$t_p = \tan \alpha_p \tag{3.99}$$

The projection of the point $\boldsymbol{p}$ on the $\boldsymbol{u}$ and the $\boldsymbol{w}$ axes are given as:

$$\boldsymbol{p}_u = (\boldsymbol{l}_i + \lambda(\boldsymbol{l}_j - \boldsymbol{l}_i)) \cdot \boldsymbol{u} \tag{3.100}$$

$$\boldsymbol{p}_w = (\boldsymbol{l}_i + \lambda(\boldsymbol{l}_j - \boldsymbol{l}_i)) \cdot \boldsymbol{w} \tag{3.101}$$

From the Eqn. 3.97, tan α_p can be written as:

$$t_p = \frac{(\boldsymbol{l}_i + \lambda(\boldsymbol{l}_j - \boldsymbol{l}_i)) \cdot \boldsymbol{w}}{(\boldsymbol{l}_i + \lambda(\boldsymbol{l}_j - \boldsymbol{l}_i)) \cdot \boldsymbol{u}} \tag{3.102}$$

Rearranging the equations, λ can be obtained as in the Eqn. 3.107.

$$t_p((\boldsymbol{l}_i + \lambda(\boldsymbol{l}_j - \boldsymbol{l}_i)) \cdot \boldsymbol{u}) = (\boldsymbol{l}_i + \lambda(\boldsymbol{l}_j - \boldsymbol{l}_i)) \cdot \boldsymbol{w} \tag{3.103}$$

$$t_p \boldsymbol{l}_i \cdot \boldsymbol{u} + \lambda t_p(\boldsymbol{l}_j - \boldsymbol{l}_i) \cdot \boldsymbol{u} = \boldsymbol{l}_i \cdot \boldsymbol{w} + \lambda(\boldsymbol{l}_j - \boldsymbol{l}_i) \cdot \boldsymbol{w} \tag{3.104}$$

$$t_p \boldsymbol{l}_i \cdot \boldsymbol{u} - \boldsymbol{l}_i \cdot \boldsymbol{w} = \lambda(\boldsymbol{l}_j - \boldsymbol{l}_i) \cdot \boldsymbol{w} - \lambda t_p(\boldsymbol{l}_j - \boldsymbol{l}_i) \cdot \boldsymbol{u} \tag{3.105}$$

$$t_p \boldsymbol{l}_i \cdot \boldsymbol{u} - \boldsymbol{l}_i \cdot \boldsymbol{w} = \lambda(\boldsymbol{l}_j - \boldsymbol{l}_i) \cdot (\boldsymbol{w} - t_p \boldsymbol{u}) \tag{3.106}$$

$$\lambda = \frac{\mathrm{t}_p \boldsymbol{l}_i \cdot \boldsymbol{u} - \boldsymbol{l}_i \cdot \boldsymbol{w}}{(\boldsymbol{l}_j - \boldsymbol{l}_i) \cdot (\boldsymbol{w} - \mathrm{t}_p \boldsymbol{u})} \tag{3.107}$$

With this the point can be calculated and the λ parameter can be written as:

$$\lambda = \frac{\boldsymbol{l}_i \cdot (\boldsymbol{w} - \mathrm{t}_p \boldsymbol{u})}{(\boldsymbol{l}_i - \boldsymbol{l}_j) \cdot (\boldsymbol{w} - \mathrm{t}_p \boldsymbol{u})} \tag{3.108}$$

$$\boldsymbol{l}_p = \boldsymbol{l}_i + \frac{\boldsymbol{l}_i \cdot (\boldsymbol{w} - \mathrm{t}_p \boldsymbol{u})}{(\boldsymbol{l}_i - \boldsymbol{l}_j) \cdot (\boldsymbol{w} - \mathrm{t}_p \boldsymbol{u})} (\boldsymbol{l}_j - \boldsymbol{l}_i) \tag{3.109}$$

Now the point $\boldsymbol{r}_\mathrm{p}$ can be calculated by:

$$\boldsymbol{r}_p = \boldsymbol{r}_a + \boldsymbol{l}_i + \frac{\boldsymbol{l}_i \cdot (\boldsymbol{w} - \mathrm{t}_p \boldsymbol{u})}{(\boldsymbol{l}_i - \boldsymbol{l}_j) \cdot (\boldsymbol{w} - \mathrm{t}_p \boldsymbol{u})} (\boldsymbol{l}_j - \boldsymbol{l}_i) \tag{3.110}$$

Where, point $\mathbf{r}_a$ is the attachment point or so called origin point. After calculating the intersection point $\mathbf{r}_p$, the residual covariance calculation is performed for that slice. This is repeated for each slice selecting the minimum covariance slice. As for the residual elements, the distance along $\boldsymbol{l}$ is also divided into segments, each segment containing the remaining covariance along the slice. After the remaining or residual covariances for each stage in each segment is calculated the search for the shortest path starts with the overall shortest path slice and along the way, moving to the next stage, The

neighbouring covariances are checked if the neighbours contains shorter path then the slice is changed. The so called, the stage covariances are measured as:

$$\mathrm{cov}_i(j) = \sum_j \left(\boldsymbol{p}_w^2 + \boldsymbol{p}_u^2 \right) \tag{3.111}$$

Where, j is the stage index of the slice i. This is the summation for all points within a segment or so called the slice. An example of the implementation of the shortest path (red line) algorithm is shown in the Figure 3.6. Having calculated the shortest path the next is to describe the tissue dimension (the cross section). Currently this is taken to be an assumed radius, either taken to be constant or a function varying with the length of the shortest path. The action line or the centroid is described by the surface normals and the radius value. The moment arm of a tissue acting between two (or more) bodies is determined by checking the body membership of the surface points along the path at the point where the line leaves one body "flying" to the other. At this very point the normal to the surface with the muscle radius determines the force action point. Similarly the force action point on the other body is determined by the point of arrival of the tissue, again determined by the initial contact point. The moment arm for this tissue is determined by the perpendicular distance between the line described by these points and the momentary centre of rotation (shown in the Figure 3.6).

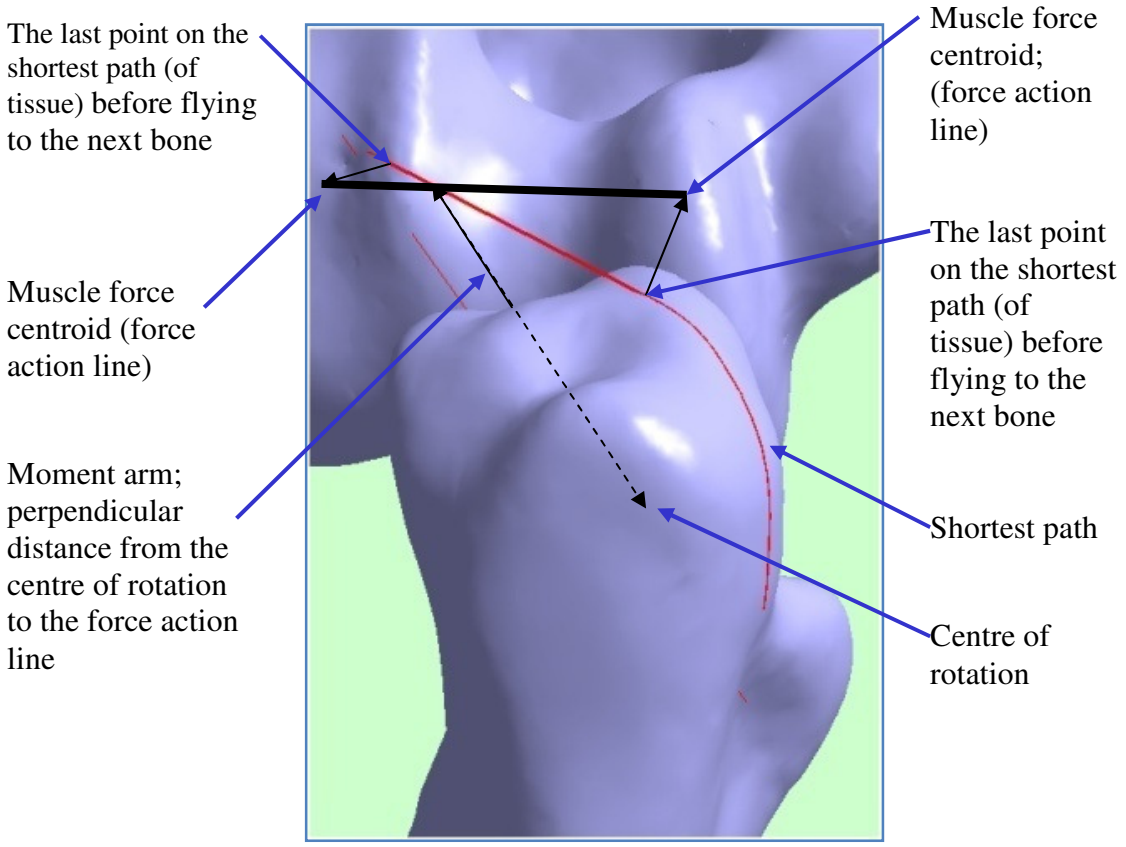


Figure 3.6 Shortest path wrapping of a tissue between ulna-humeral joint

It must be noted that through the developed shortest path wrapping algorithms at every time step, change in path length, and change in moment arm can be measured. This approach is a novel addition to the moment arm concept where the kinematics dependent change in the shortest path and wrapping algorithms and the corresponding moment arm can now be measured and analysed dynamically. Moreover the 6DOF joint kinematic model also allows the study of tissue length and wrapping behaviour and correct ligament and cartilage loading calculations.

3.3.2 Calculation of the centre of rotation and the moment arm

3.3.2.1 The instantaneous centre of rotation of a "joint"

When a joint is articulating then the motion of one of the bones is needed to be expressed relative to the axes frame of the other, so that one becomes stationary (or appears to be stationary). Once the relative velocity of one of the bodies is obtained which is relative to the other, then the instantaneous centre can be calculated. Let us assume that the motion of the moving body is described by the translation of its centre and angular velocity as $\boldsymbol{v}$ and $\boldsymbol{w}$ respectively. In finding the centre it is important to recognise that not all the velocity vector $\boldsymbol{v}$ can be described in terms of $\boldsymbol{w}$ alone. It is from basic kinematics that the only part of the translational velocity $\boldsymbol{v}$ can be associated with $\boldsymbol{w}$. Only a part of $\boldsymbol{v}$ which is perpendicular to $\boldsymbol{w}$ vector, this is shown in Figure 3.7a. In order to find the centre, this components needs to be calculated. The vector component of $\boldsymbol{v}$ along $\boldsymbol{w}$ is given by:

$$\boldsymbol{v}_w = \frac{\boldsymbol{w}\cdot\boldsymbol{v}}{|\boldsymbol{w}|}\frac{\boldsymbol{w}}{|\boldsymbol{w}|} \tag{3.112}$$

And $\boldsymbol{v}$ perpendicular to $\boldsymbol{w}$ is given by:

$$\boldsymbol{v}_h = \boldsymbol{v} - \frac{\boldsymbol{w}\cdot\boldsymbol{v}}{|\boldsymbol{w}|}\frac{\boldsymbol{w}}{|\boldsymbol{w}|} \tag{3.113}$$

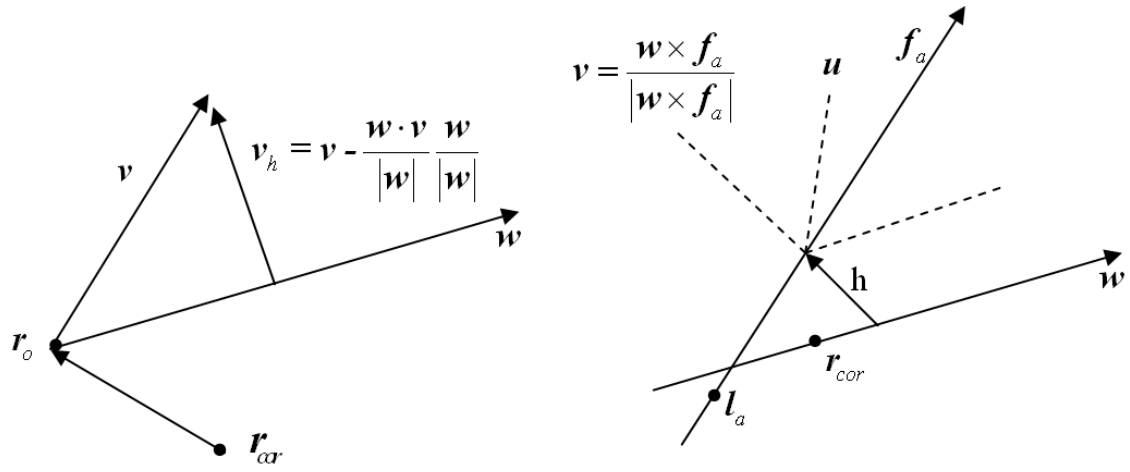
Therefore the distance r can be calculated as:

$$r = \frac{|\boldsymbol{v}_h|}{|\boldsymbol{w}|} \tag{3.114}$$

And the instantaneous centre position is given by:

$$\boldsymbol{r}_{cor} = \boldsymbol{r}_o + r\frac{\boldsymbol{w}\times v}{|\boldsymbol{w}\times v|} \tag{3.115}$$

Where, $\boldsymbol{r}_o$ is the origin of the moving body`s axis system.



(a) (b)

Figure 3.7 (a) Instantaneous centre of rotation, (b) Moment arm calculations

3.3.2.2 The moment arm calculations

The distance between the force action line of a tissue, defined by, and the line parallel to $\boldsymbol{w}$ and passing through $\boldsymbol{r}_{cor}$ is casually known as the moment arm. The line of action is either the vector between attachment points or the vector between the point $\boldsymbol{l}_a$ at which the tissue centroid "fly" from one body to the other.

$$\boldsymbol{l}_{\mathrm{w}} = \boldsymbol{r}_{cor} + \lambda_1 \boldsymbol{w} \tag{3.116}$$

$$\boldsymbol{l}_{\mathrm{f}} = \boldsymbol{r}_{\mathrm{a}} + \lambda_2 \boldsymbol{f}_{\mathrm{a}} \tag{3.117}$$

The distance between these lines is the moment arm. However this is not true unless these two lines are perpendicular to each other (see Figure 3.7b). It can be shown that;

$$\boldsymbol{v} = \frac{\boldsymbol{w} \times \boldsymbol{f}_a}{\left|\boldsymbol{w} \times \boldsymbol{f}_a\right|} \quad \text{and} \quad \boldsymbol{u} = \frac{\boldsymbol{v} \times \boldsymbol{w}}{\left|\boldsymbol{v} \times \boldsymbol{w}\right|} \tag{3.118}$$

$$\mathrm{h} = \left(\boldsymbol{l}_{\mathrm{a}} - \boldsymbol{r}_{cor}\right) \cdot \frac{\boldsymbol{w} \times \boldsymbol{f}_{\mathrm{a}}}{\left|\boldsymbol{w} \times \boldsymbol{f}_{\mathrm{a}}\right|} \tag{3.119}$$

The moment arm is the relationship between the tissue length change which is given as $\boldsymbol{f}_{\mathrm{a}}$ (this is in the same direction of the muscle force) and the corresponding angular rotation $\boldsymbol{w}$. The correct moment arm calculation is given as;

$$momentArm = \frac{\boldsymbol{f}_a \cdot \boldsymbol{u}}{\left|\boldsymbol{w}\right|} \tag{3.120}$$

Note that this equation is similar to the one generally used in the literature and developed by An et al. (1984) as:

$$momentArm = \frac{dl}{d\theta}$$

Where dl is the muscle length change and the $d\theta$ is the joint angle change. This is equivalent to $momentArm = \frac{\boldsymbol{f}_a}{\left|\boldsymbol{w}\right|}$ which assumes that the $\boldsymbol{f}_{\mathrm{a}}$ and $\boldsymbol{w}$ are perpendicular to each other.

3.4 Development of collision detection, contact and collision response algorithms

3.4.1 Background and rationale of the new algorithm

The contact and collision literature is covered and addressed in the Section 2.3. Also importance of the contact analysis in the multi body modelling of diarthrodial joints and its significance in accurate prediction of joint torque and muscle forces have been argued. What has been recognised through the literature survey that complete joint modelling cannot be achieved without considering the contact, collision detection and collision response. As the proposed multi body joint modelling is a composition of multi disciplines, contact or collision are referred to as one of them. Specifically, in contact analysis, collision detection and collision response are important factors. Thus, in considering the proposed complete multi body joint model, it is required to obtain time or kinematics dependent tissue elongations, moment arms, cartilage deformation and geometric contact point locations. In considering the contact, it is proposed to locate the cartilage contact elements as a function of time and geometry and evaluate the deformable contact during joint articulation. In this case, the complete set of the procedures including contact and collision are implemented and they are encapsulated to drive the proposed unconstrained dynamic joint analysis. In anatomic joint contact analysis, it is not always possible to have the human skeletal surface models being presented in an assembled and correctly positioned. Even if this is the case, the actual equilibrium positions of bones will be dependent on several conditions such as loading and dynamics of the assembly. Therefore it is important to start the dynamic motion analysis from an equilibrium position. In the proposed analysis the equilibrium is taken as the static equilibrium position. Initially, the contact analysis starts with extracting triangulated bone surfaces (in VRLM 2.0) as the kinematic elements of a joint such as ulna and humeral bones of the elbow joint. Through assuming that bones are placed close to their natural positions, then the algorithm is employed to move bones to approach each other and to find the equilibrium positions. Since the objects (bones) are of rather complex shapes, the collision detection between two bones cannot be performed through some algorithms which can only handle convex shapes. For the collision detection, commonly used collision detection packages such as the RAPID

(Section 2.3.2) considered. Although implementation of the RAPID algorithm is currently undertaken with the supplier of the graphics platform used in this thesis, it was not possible to wait for the implementation thus the algorithm to be presented in this section was developed. This was also because the RAPID was believed to perform well only in the presence of a multi non overlapping objects. Another problem with RAPID was that it was only capable of detecting collision by following objects in motion and performing sophisticated culling of graphic primitives (triangles) at each time step. In problems where motion is relatively slow and accuracy was important, "time to collision" was more important than "gross collision detection". Hence, there was a need to develop number of algorithms to perform collision detection and collision response respectively. As mentioned in Section 2.3, the unilateral contact based analysis can be adapted to perform collision detection and collision response together. For example, penalty based collision response methods where constraint equations such as penalty depth equation is introduced to the system and bodies are not allowed to interpenetrate more than the acceptable distance. Then, through the penalty depth calculations the collision response springs are attached between bodies. This kind of methods is driven with reducing the kinematic mobility of bodies. However, if the objective is analysing the joint contact in anatomic fidelity, more appropriate contact and collision algorithms are required. Thus, in order to uncover the complex dynamic contact phenomena of anatomic joints, it has been proposed to develop the following algorithms which are encapsulated with the rest of the developed methods and algorithms to drive the complete unconstrained joint analysis. The proposed contact and collision algorithms initially start finding the equilibrium position through based on the loading of the springs or tissues (i.e. ligament, tendon, and muscle) allow directing bones to each other. Hence, contact and collision algorithms have been proposed to find shortest distance between bones along the direction of motion through attached springs (Figure 3.8), this will be explained later in detail.

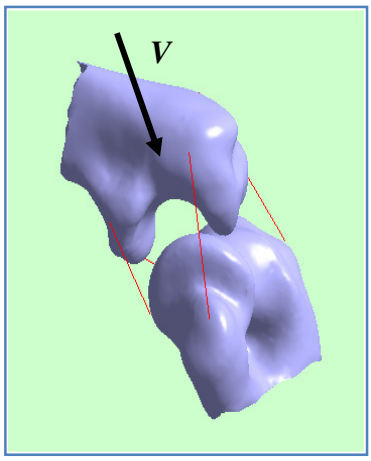


Figure 3.8 Interactive attachments of soft tissues and resulting direction of motion

3.4.2 The collision algorithm

As the details of algorithm will be given in the implementation chapter, here only basic formulation relating to the algorithm and the outlines of the procedures will be highlighted. The algorithm designed to ensure that ligament loading pulls bones together when they are displaced outside their static equilibrium position. It is assumed that the ligaments are loaded even when the bones are in their equilibrium position.

For the collision calculation, although it is possible to move bodies towards each other (the direction of motion to be calculates based on ligament forces) and incrementally calculate the nearest distance, this is a very inefficient method. The first step in the proposed method is to test every point on the moving surface which has a surface normal with an angle less than 90^0 to the direction of motion against to the other object(s) surface.

Having calculated the forces, in the absence any reacting force from the contact surface there will be a pulling force between the bodies. The algorithm assumes that one of the bones is stationary and others are moving relative to the stationary bone. It is always possible to describe the motion of a body relative to the axes system of another body. This definition is equivalent to seeing one of the bodies as stationary and the other as mobile relative to the stationary one. The proposed method allows entering ligament connections interactively and entering the unloaded lengths and tension at each tissue connection. This enables the direction of motion $\boldsymbol{V}$ to be calculated. This was a very

useful feature of the software especially at the testing stage. The software can also read these from the previously stored data base. Let`s assume the presence of two bodies, body $\boldsymbol{i}$ and body $\boldsymbol{j}$. On the moving body $\boldsymbol{i}$ each point on the surface is projected towards body $\boldsymbol{j}$ and one would expect to collide with a point on the surface of the body $\boldsymbol{j}$. However not all points on the bodies need to be investigated. A point can potentially collide only and only if the angle between its surface normal and the velocity is less than $\frac{\pi}{2}$. In considering a point $\boldsymbol{p}_i$ on body $\boldsymbol{i}$ and point $\boldsymbol{p}_j$ on body $\boldsymbol{j}$, the algorithm determines if these two points are in the range of collision trajectory and if they need to be considered for collision. Not all points satisfying the collision criterion will collide as only the shortest distance(s) among all eligible pairs will collide. This application is similar to the culling operation (Section 2.3.2). Thus, through defining the potentially colliding points in the shortest distance trajectory, the following statement determines if points are on the positive side of the surface with respect to the motion vector $\boldsymbol{e}_v$ which is the unit vector of $\boldsymbol{V}$. If considering point $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$, this condition can be defined and satisfied only and only if following inequalities are satisfied:

$$\boldsymbol{n}_{pi} \cdot \boldsymbol{e}_v > 0 \tag{3.121}$$

$$\boldsymbol{n}_{pj} \cdot -\boldsymbol{e}_v > 0 \tag{3.122}$$

If the above conditions are satisfied then, the points will be checked if they are in their collision path trajectory. Then the condition for this is given as:

$$\text{If } \left| \boldsymbol{l}_{ij} \times \boldsymbol{e}_v \right| < tol \tag{3.123}$$

Where, point $\boldsymbol{i}$ and $\boldsymbol{j}$ are concerned as potentially colliding points and *tol* is the tolerance which is determined by the point density as:

$$tol = \rho \sqrt{\frac{Surface\_Area}{No\_of\_points}} \tag{3.124}$$

Where ρ is a factor, represents irregularity in surface point distribution density, this is taken to be approximately 5 but for a relatively uniform distribution but approximation of 1 should work. In considering the fact that around 4 points should neighbour every point under test any fluctuation of density will be catered for. Taking values 5 ensures a selection with a reasonable safety factor. On the prototype software, *tol* itself is selected by trial and error. Note that the cross product term gives the distance h_{ij} as shown on the Figure 3.9.

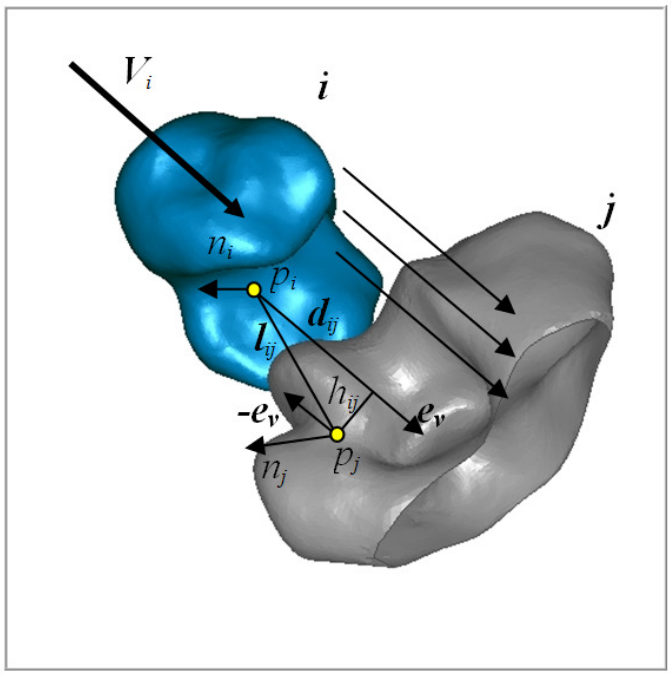


Figure 3.9 Illustration of collision algorithm along with applied directional velocity $\boldsymbol{V}_i$

The current algorithm is based on the point clouds but future algorithms may use intersection of the motion vector with the surface simplexes to ensure more precise calculation of collision surfaces. The final stage of the algorithm needs to compare all

eligible pairs to find the shortest distance among them which will give the potentially colliding points. The distance can be given as:

$$\mathbf{d}_{ij} = \boldsymbol{l}_{ij}.\boldsymbol{e}_v \tag{3.125}$$

Collecting all the elements of the algorithm as:

$$\begin{aligned} &\textit{if } \boldsymbol{n}_{pi}.\boldsymbol{e}_v > \boldsymbol{0} \textit{ and } \boldsymbol{n}_{pj}.-\boldsymbol{e}_v > \boldsymbol{0} \textit{ and } \left|\boldsymbol{l}_{ij} \times \boldsymbol{e}_v\right| < tol \textit{ and } abs\left(\boldsymbol{l}_{ij}.\boldsymbol{e}_v\right) < Mind_{ij} \textit{ then} \\ &Mind_{ij} = \boldsymbol{l}_{ij}.\boldsymbol{e}_v \end{aligned} \tag{3.126}$$

When all the points of bodies $\boldsymbol{i}$ and $\boldsymbol{j}$ are tested, $Mind_{ij}$ will give the points $\boldsymbol{i}$ and $\boldsymbol{j}$ which will satisfy the collision conditions.

After satisfying all the collision detection conditions presented above, collision response method is driven. In the proposed joint model, cartilage contact or deformable contact is considered as a part of the contact and collision algorithms and dynamic model. Therefore, after detecting the potentially colliding points, deformable contact based analysis is carried out. First of all, the contact and collision response analysis starts after the equilibrium position is achieved. Thus, the list of potentially contacting points and triangles are obtained which is based on the distance calculation between these geometric features. In the proposed contact and collision analysis, interpenetrations are not allowed through defining the acceptable closest distance between bodies. The closest distance constraint is similar to the inequality constraint equations such as the gap functions employed for unilateral contact constraints (Section 2.3.3). However the main difference between these constraint equations and the employed constraints in the proposed model is the different mobility description of the motion. By means of kinematic description, it is intended to emphasize that in unilateral based contact constraint equations at the moment of contact (a state of the kinematics), mobility is reduced. But in the proposed study, due to the unconstrained basis of the kinematics,

even there is a contact between bodies the mobility is not reduced and each body still possess 6DOF. Then, the analysis is carried out by enabling further processes to deal with the rest of the contact analysis. Thus, spring elements are attached at the colliding pairs of points. These spring elements are referred to as the contact springs representing cartilage stiffness among $\boldsymbol{l}_{ij}$ and constant lateral stiffness as 0.001of the normal stiffness. The cartilage stiffness calculation is given in the next section. After placing these spring elements the motion analysis continues but checking and updating the contact condition as the analysis progresses. Thus, body $\boldsymbol{i}$ is allowed translating and rotating as shown in Figure 3.10.

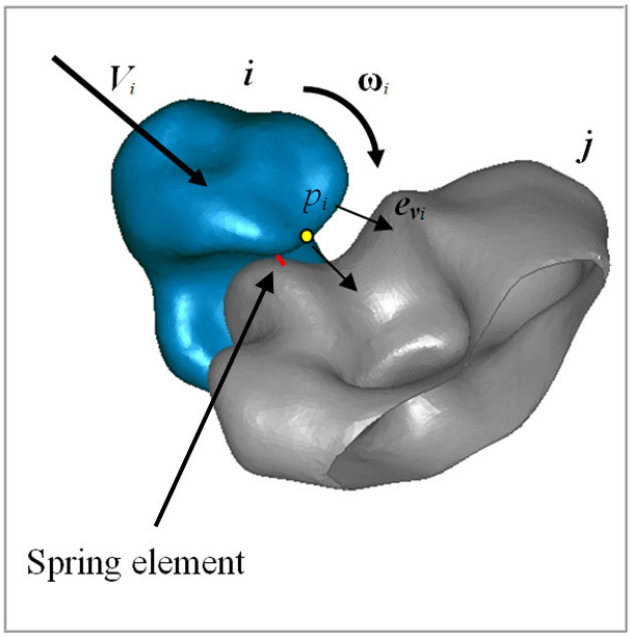


Figure 3.10 Illustration of collision algorithm along with directional velocity $\boldsymbol{V}_i$ and rotation $\boldsymbol{\omega}_i$

Normally there will be more than one spring attachments and the total number is user controlled, the resultant sliding will ensure sliding translation on all contact surfaces as the mathematical formulation given previously ensures this. Since the motion now includes rotation as well as translation each body on the moving surface has different velocity vector. Also, instead of $\boldsymbol{e}_v$ which was provided earlier in the algorithm that

applies for all the moving points, $\boldsymbol{e}_v$ needs to be calculated for each point on the body $\boldsymbol{i}$ as:

$$e_{vi} = unitVector\left(\boldsymbol{V}_i + \boldsymbol{\omega}_i \times \boldsymbol{p}_i\right) \tag{3.127}$$

3.4.3 The cartilage model

For the cartilage contact force model the equation given in the Eqn. 3.128 is adopted from the literature which is also performed by COSMOSMotion, cited in Fisk & Wayne, (2009).

$$F = kg^e + \left(\frac{dg}{dt}\right) f(c,d) \tag{3.128}$$

Where F is the magnitude force written as a function of k (stiffness), depth of penetration between bodies (g), exponent e is 2. The second term is the viscoelastic term which is a function of c, the damping coefficient and d, the maximum damping at the maximum penetration. The k value is taken as 8000 N/mm. c value represents the maximum damping which is assumed as 400 N-s/mm. The penetration at maximum damping can be represented by d which is assumed as 0.001 mm. The friction is assumed to be negligible. The implementation of the contact load cannot use this in its current form. However the program calculates total penetration of all nodes and finds average penetration (irrespective of their directionality). Then it calculates the force due to the average penetration and distributes these back to nodes according to their individual contributions based on the penetration magnitude and vector directions.

3.5 The summary of the proposed theories

1. A new linear stiffness matrix formulation suitable for human joint modelling.

2. A new non-linear, polynomial material stiffness and geometrically nonlinear analysis formulations and solution scheme suitable for human joint modelling.

3. A new centre of stiffness axes of stiffness formulation. The proposed tensorial treatment of stiffness matrix representing flexibly connected rigid bodies is novel not only in the biomechanics field but also in general engineering.

4. A new muscle wrapping algorithm and its associated formulations.

5. A new collision detection algorithm suitable for calculating accurate collision conditions using a “look ahead” or “time to collision” scheme.

CHAPTER 4

SOFTWARE DEVELOPMENT AND IMPLEMENTATION

4.1 Introduction

It is impossible to give all details of the implementation. In this chapter only important considerations of the implementation will be described. Nevertheless engineering a software solution is no different than engineering machinery. Despite of the initial planning and innovative ideas, a long period of evolution, testing and redesign follows. This inevitable cycle is further fuelled by parallel development work that goes on by the "third party" component suppliers and their continuing re-design. In the software development this is significantly more serious than, hardware development where new products and new versions released to the market at a very high rate. It is possible for one to reduce the reliance to third party software components but the avoidance is not possible. Even for the most basic software, the development would rely on the operating system and the integrated development environment (IDE) and its compilers. The IDE on the Microsoft Windows platform has limited graphics capability unless one uses their main graphics platform which is the DirectX. Although the DirectX was a possible option, it was decided that a third party graphics kernel would be used. The main reason for this decision is to protect the code developed in this project from costly re-coding in the future from the changes which can happen. Considering the fact that the current version of the DirectX is number10, illustrates the point made. This is approximately corresponding to a new release every 18 months. The same applies for the operating system. Choosing a third party component shifts the burden of interface maintenance to the supplier. After investigating the cost and capability of various graphics kernel options the KernelCAD was selected. KernelCAD is an OpenGL based graphics "interface" component. As for the programming language, VB.Net was used. Selecting a language was not a major problem as under the Microsoft Visual Studio platform, there is a high degree of convergence among all the languages (C, C++, C# and VB) and the same maths library drives them all and all generate a common pre-compiled code to one unified compiler. Deciding on the data representation, data format or data base was a difficult problem. If the data is well structured then it is a

good idea to use conventional data base such as ACCESS. However if the data is less structured, then XML type representation gives an added flexibility. Of course there is also an option of saving data in ordinary text (ASCII) files. Among these options the ACCESS option is chosen, because, although XML is probably better in the long run, visual studio provides little support as interface between XML to its native components. The same is true for the text files. Another important option is the reporting, although the software presented here has no reporting capability it is important that this should be considered and again the ACCESS has link to the Crystal Reports (an industry leading reporting engine). There are disadvantages of the ACCESS, it is not ideal to describe "spread sheet" type data. For this the Microsoft Excel is used with an OLE (Object Linked Embedded) type interface. Such interface is cumbersome and it is not easy to maintain but this was a compromised solution. Finally the software is structured and designed to allow further developments and extensions. Normally inflexibility of a software is more related to its data structure than to its algorithms. In the data structures section below, it will be explained how such flexibility is achieved.

The best way of describing software is to describe its data structures and its algorithms, thus the chapter is divided into these two main sections.

4.2 Data structures

Here it is not easy to describe the rationale and reasons of choosing every parameter and variable without making the thesis excessively large, however the data base will be provided for information. The database is made of two types of data base tables, general tables holding non-project based information and project based information. Although the intention with general tables is to store information which the project based tables can use. Thus these tables can remain as read-only tables which ensure certain level of data integrity and security. The second set of tables are the project tables which links to the analysis program and modifiable by the program/user and holds project or case specific information. At this stage only project specific data tables are populated. Although non-specific skeletal data has been created, there is no urgency of populating the general data base because of its peripheral and marginal relevance to the main contributions of the thesis. Here only the main fields of the project specific tables will be presented as these are the main drivers of the analysis. The following are

the project specific data base tables are: ProjectHeaders, ProjectSpring, ProjectMass, ProjectPoints, ProjectMounts, ProjectBuffers, ProjectForces and ProjectResults.

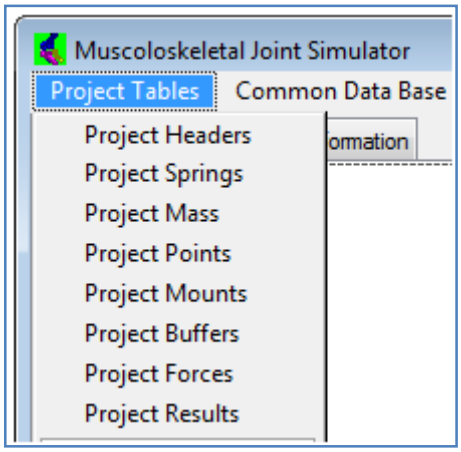


Figure 4.1 Main database tables

- ProjectHeaders

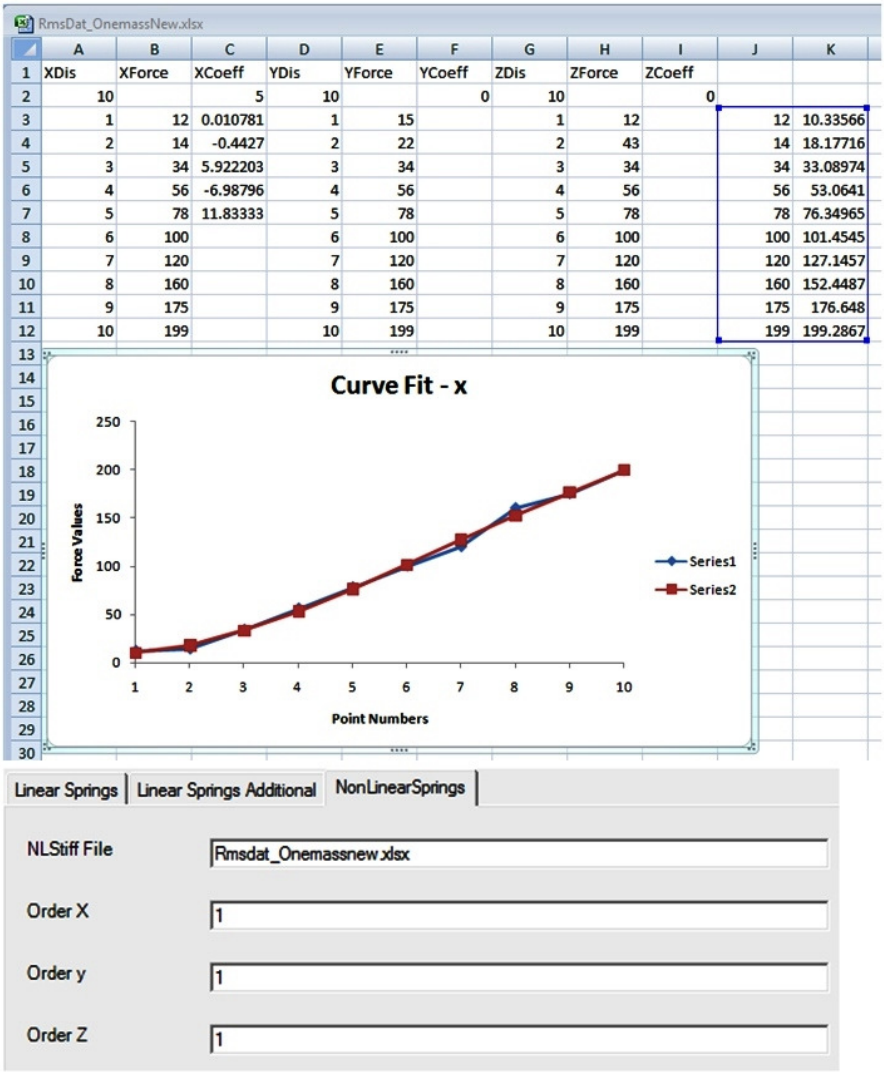
This table holds information necessary to identify the project and the case being studied. Two most important parameters are the Name of the project which links all the other table records to the header but also to each other. The other important parameter is the file path (full path) to the skeletal VRML file which holds the skeletal surface information.

- ProjectSprings

This table holds information about the tissue parameters. The main records hold information about linear definition of tissue parameters. Rather than separating muscle, tendon, ligament and cartilage all described as springs (or spring-dashpot) and type indicator is used to indicate what a given record represents. For example, 1 means linear spring, 2 means linear dashpot, 3 means rotational spring, 4 means rotational dashpot, 5 means combination of 1 and 2. All springs have 3 stiffnesses (or dampings) in their principal axes directions. The orientations of springs are decided during their attachments and not here. Unfortunately the table is not suitable for describing non-

linear information. For this one would need a record with fields such as “coefficient 1”, coefficient 2” and etc., up to the order of curve fitting used. However one cannot pre decide how many coefficients are necessary to describe a given data as this would depend on the data being analysed. Originally it was decided that curve fitting would not be allowed beyond the order 15. Although this seems sufficient, there are other conditions that needs to be abided in order to ensure a successful analysis, such as the order of curve fitting must always be an odd number if the tissue allowed having tensile as well as compressive (cartilage) and that the curve should not oscillate near the origin. And the slope near the origin should accurately represent the linear slope (in case a linear analysis is used). All these considerations make the curve fitting difficult from the user point of view. The curve fitting has two main advantages, firstly compresses the data and secondly as explained in the theory section, it is more efficient when dealing with the non linear analysis problems. However it is a good practice to keep the raw data, even if it is not in the main data base, in the long terms this should be integrated into the main data base. Although the efficiency of the polynomial option is a distinct advantage, difficulty of the usage in terms of its reliance on the users skill in curve fitting compromises an accurate analysis. In other word user must be confident in the use of the software to ensure the results are correct. With this consideration, the software presented here allows the raw data to be used as it is as well the polynomial curve fitting. To implement both the raw and the polynomial option, for the data storage, the Microsoft Excel is used and an interface to Excel is created by the OLE (Object Linking and Embedding) interface.

The excel file contains two columns of force deflection (or velocity) data for each principal axes of the spring (XDis, XForce), (YDis, YForce) and (ZDis, ZForce). Once the link is established with the Excel file, program asks the user to choose the direction and the order of curve fitting. The program performs the curve fitting, plots the curve of the raw data against the fitted data and asks the user to accept or re run the curve fitting process. An example file is shown in the Figure 4.2. What program should use in performing the final analysis depends on entries in the “NonLinearSprings” pane. If “Order X” is set to 1 then the coefficients are used directly from the Access data base, if it is set to 0 then raw data is used and finally if it is any other number then polynomial coefficients are used from the excel file name of which is indicated in the “NLStiff File” text box.



	A	B	C	D	E	F	G	H	I	J	K
1	XDis	XForce	XCoeff	YDis	YForce	YCoeff	ZDis	ZForce	ZCoeff		
2	10		5	10		0	10		0		
3	1	12	0.010781	1	15		1	12		12	10.33566
4	2	14	-0.4427	2	22		2	43		14	18.17716
5	3	34	5.922203	3	34		3	34		34	33.08974
6	4	56	-6.98796	4	56		4	56		56	53.0641
7	5	78	11.83333	5	78		5	78		78	76.34965
8	6	100		6	100		6	100		100	101.4545
9	7	120		7	120		7	120		120	127.1457
10	8	160		8	160		8	160		160	152.4487
11	9	175		9	175		9	175		175	176.648
12	10	199		10	199		10	199		199	199.2867

Figure 4.2 Non-linear data, OLE link to the Excel

The algorithm which calculates the force value from a raw data simply check the deflection of the spring along the principal axes of the spring at each integration step and establishes in between at which two data points the current deflection is located and calculates the corresponding force value. This is not explained previously in the theory section because it involves a simple interpolation. A simple pseudo code is given as follows as this is executed for all springs and for each principal axis of each spring.

For i=1 to all data points

If current-spring-deflection along the principal axes is less than rawdeflection(i+1) but greater than rawdeclection(i) then interpolate the force at the current-spring-defection, between these points, and exit For

Next i

- ProjectMass

ProjectMass contains inertial information of all bones in the project such as the mass and inertia values, orientation of their local axes system relative to the global axes (this is described by the Euler angles of the local axes orientations). The axes systems of individual bones are important for a number of reasons. Firstly the analysis requires an axis system, but having volumetric information (obtainable from the surface data) lead to calculate the principal axes. Although this is possible, the user will have no information where the axes system is to be located and no way of deciding where ligament tendon attachments to be located relative to the axes system. In practice there is no easy way of solving this problem and there is no one reliable standard is on offer. It has been suggestions to use the bony features or the bony landmarks but currently there is no widely acceptable standard. The "central axis of bone" commonly referred to as the reference in measuring the joint angle in the orthopaedic literature is not an invariant vector in mathematical sense and it does not likely to match with the principal axes of the bone along its shaft. There are implications of this, especially when it comes to comparing the software with commercially available software, unless the bone surface geometries, the axes system, muscle, ligament attachment points are the same as the software developed here and the commercial package, it would be impossible to have a reliable test. In order to "solve" this problem, it was decided that the local axes of all bones and the global axes of the system should be coincided. And this was made to coincide with the axes system used by the Lifemod software. This way the ligament and tendon attachments can be coordinated relative to the global axes system. Program then can, internally use the local axes system in the assembly of equations which coincide with the centre of mass of each bone. For the linear analysis, the local axes systems are aligned to be parallel to the global axes whereas for the non linear analysis the principal axes are used. This ensures that in state space formulations and iterative

solution process, inertia matrix is not needed to be inverted. The ProjectMass table includes the motion constraints of each body. As the constraints need to be decided by the user in advance and similar to deciding any geometrical information about a local body, this also possesses similar difficulties. To make this manageable, it is assumed that the constraint data should also be relative to the global axes. In addition to having options constraining local body motion, the user is also given the option to remove any particular body from the analysis altogether. This option ensures that no unnecessary bodies are collected and assembled during the analysis. This way a single skeleton can be used for many different analyses but selecting only appropriate bone combinations. Another option for the constraint management is the option which allows a body to be fixed but still considered for the analysis. This option is necessary because of the muscle wrapping algorithm. It is possible that a tendon attachment coming from a "fixed body" may change its location due to change of motion geometry of the moving bodies. This means that even though the fixed body will not appear in the dynamics equations, its surface information is necessary in identifying the attachment points by the tissue wrapping algorithm. The "ProjectMass" also includes the initial displacements and velocities. Displacements are assumed to be small and relative to an equilibrium position. The software is not designed to accept large initial deflections as its muscle wrapping implications. Normally the program assumes that the equilibrium position is unknown and that iteration starts from the position given by the skeletal data file. The initial condition simply modifies this starting position. The need for this may arise because the convergence to an equilibrium position may not be happening or happening fast enough or failure of collision detection process or simply one may decide to investigate the uniqueness of the equilibrium position.

The Figure 4.3 shows an object inertia calculations being displayed. Inertia calculation algorithm uses triangle meshes and the Gauss Divergence theorem. Basically, mass density of a 3D volumetric solid is usually assumed as uniform and constant value 1 to simplify the procedure when multiplying it with mass and inertia tensor. In order to compute mass and inertia tensor, several algorithms and methods have been constructed based on generalised theorems and principles. The Gauss Divergence theorem is one of them, which is most commonly used to represent 3D volume integrals in terms of 2D surface integrals and latter line integrals. Thus this reduction, simplifies the procedure where the surface information is convenient to represent volumetric integrals. Hence,

this reduction provides practical computation along with computing moments of surfaces rather than moments of concerned volumetric objects. For inertia tensor calculations, practical mathematical algorithm is employed which as developed by Mirtich, (1996a) for polyhedral meshes with employing Gauss Divergence theorem.

The pseudo code of the algorithm is given by Eberly (2003). The calculated mass/inertias may be used to update the data base entries. These values are further modified by the Euler angle information in the same database. The default Euler angles are all zero.

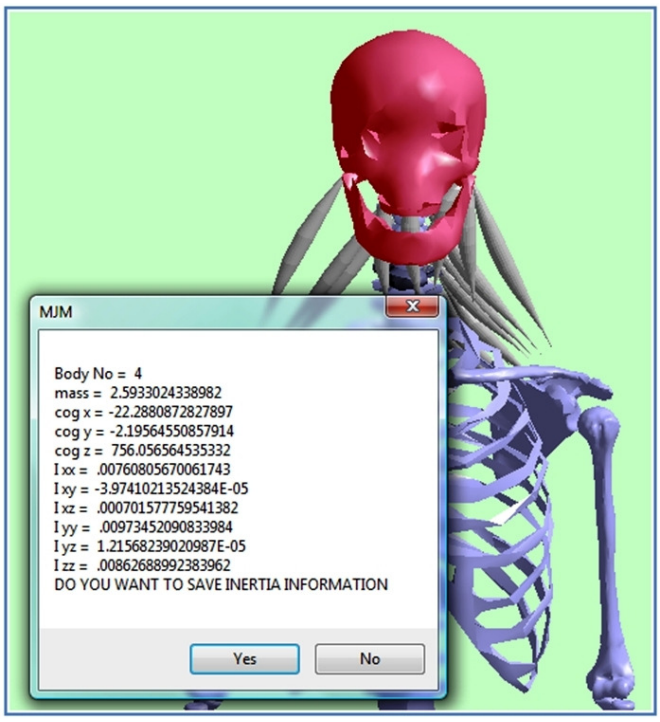


Figure 4.3 Mass, COG, and inertia tensor calculations

- ProjectPoints

This table contains point on each body. Point has no contribution to the analysis. It is included to be used to display motion of points on a selected body.

- ProjectMounts

The "ProjectMounts" table contains information about the points at which ligament, muscles or cartilage (three primary types) ends are attached. Although cartilage is calculated from the joint surface geometry, the program also allows these to be specified manually. Initially these points are selected interactively and stored in the data base. It would have been almost impossible to create these points in any other way. Although using the mounting information from the Lifemod software was considered, the idea was abandoned when it was discovered that during the process of transferring bone information from Lifemod to MJM the bone positions had moved. The program also allows these points to be updated with the new selections. The mount information in the table, as well as specifying the coordinates of attachment on two points also specify the orientation of springs in terms of the Euler angles. If the mount information is created interactively, the force action ends of the mount, calculated by insertion points or by muscle wrapping algorithm which are used to obtain the principal directions of the local stiffnesses assuming axisymmetic stiffness. The force action direction being the main axis, with the two orthogonal axes are placed arbitrarily but perpendicular to the main one. In the current program the Euler angles for orienting the tissue attachment are not used, both in linear and non linear this is done automatically based on the end position of attachment points.

- ProjectBuffers

This was developed to handle mounts with gaps or clearances. Effectively they are mounts, activated only when the gap is closed during the deflection of the structure to which they are attached to. It was part of an experiment to deal with cartilage modelling. Although it worked, this was replaced by the collision algorithm and collision based cartilage definition.

- ProjectForces

This table includes forces acting on each body. The force definition can also include the prescribed motion of bodies. The type of force in all 6 directions is controlled by the "type" flags. These are 0, 1, 2, 3 and 4. 0 no force, 1 ordinary force with "N" as the unit of the force, 2 means the force magnitude specifies a prescribed displacement in

"m", 3 means prescribed velocity (m/s) and 4 means prescribed acceleration (m/s^2). Currently there is no check between the type of force and the type of analysis performed and this may result in error. It is necessary to ensure that for correct analysis the type of force chosen is appropriate. For example the joint stiffness calculation does not require any forcing function. Static loading requires only static loading or prescribed displacement but not prescribed velocity or acceleration as these are not relevant to the static analysis. Prescribed displacement, velocity and acceleration can be used in inverse analysis. The force selection also allows previously saved numerical data to be used as a time variant forcing function. For the numerical input option too, all five types are available. The type 1 force can be used for the forward or the types 2, 3, 4 for the inverse analysis. The table also contains the analysis to be performed and various analysis parameters such as accuracy, sampling rate, total number of sampling points, activation times of each force element, integration start and end times.

- ProjectResults

This table is not kept permanently. It is over written every time an analysis is performed. At the beginning the results for each analysis was saved in the table. As the program being tested continuously for the development purpose, it was found that, in a short time, the result file had grown excessively.

4.2.1 Future development and upgrading of the data structure

What makes software flexible and upgradeable is the flexibility of its data structure. Although algorithms are also factors in the upgrading they are less significant. Here the flexibility to upgrade the data structure is described in practical terms, that is, if a new data element is needed by the program, how it can be added without making a major change to the structure of the program. The data item needs to be editable by the user (or the program) and must reside in an appropriate database table. In other word, how can it be added to the database system of the program? In practical terms, how can one add another database field in a given table without making major change to the program? This is achieved by making the database interface and GUI (graphical user interface) of the program to be dynamically created. This also implies that the database query is also dynamically assembled.

The GUI is made of to pane containers, one managing the database interface and the other managing the results and any post processing such as animation and result plots. However this container has some pre processing functionality such as mass and inertia calculations, muscle attachment positioning etc. The panes, in the database container are dynamically managed. This container displays one table at a time and the number of panes and distribution of data in each pane and the database query that needs to be assembled to generate these dynamically managed as follows: The program contains following global data declarations.

```
NoOfTapControls = 8 'the same as the number of tables to be read from the data base
NoOfTapsInEachControl()={1, 3, 6, 1, 2, 3, 5, 3}'how many blocks (groups) of data for
each table - they are the number of tabs
NoOfFieldsInEachTap(,)={{8, 0, 0, 0, 0, 0}, {8, 9, 4, 0, 0, 0}, {9, 9, 6, 9, 9, 6}, {8,
0, 0, 0, 0, 0}, {9, 6, 0, 0, 0, 0},{9, 6, 6, 0, 0, 0}, {9, 6, 9, 5, 9, 0}, {6, 6, 6, 0,
0, 0}}  'each tab contains number of fields (grouped together)
```

NoOfTapControls declares how many data base tables are to be used and **NoOfTapControls** declares, for each table, how many tabs are to be created and **NoOfFieldsInEachTap** declares, in each tab how many fields to be displayed. For example the first table will have single tab and that tab will contain 8 fields. The listing below gives further declarations and assignments. The tab name, which relates to tables are listed in the **ProjectTablesNames** and for each table the **projectFieldsNames** identify what is to be displayed in the GUI. There are additional arrays contain help information for each field and their default values if a new record is created, declared as **projectFieldsInfo** and **DefaultFieldsVALUES**, respectively.

```
ProjectTablesNames()={"ProjectHeaders", "ProjectSpring", "ProjectMass", "ProjectPoints",
"ProjectMounts", "ProjectBuffers", "ProjectForces", "ProjectResults"}
projectFieldsNames(,,)={{{"Project Name", "File name", "Created by", "Created for",
"Date created", "Date last saved", "Other comments", "Model File Name", " "}, _
projectFieldsInfo(,,)={{{"Project Name the same for all field project names", "File name
- not used", "Created by", "Created for", "Date created not updated by the program",
"Date last saved not updated by the program ", "Other comments", "Model File Name", "
"}, _
DefaultFieldsVALUES(,,) As String={{{"Project name", "File name", "Created by", "Created
for", "Date created", "Date last saved", "Other comments", " ", " "}, _
```

With the declarations above the creation of the database query can be fully automated and the database selection which is given below all become automated. But corresponding to the database fields selected by the declarations above, GUI creation is also automated and generated dynamically. A "SELECT" command creation is demonstrated below.

```
Function createSelectCommand(iTable) As String
'add all the projectFieldsNames to built the ProjectSelectCommands string
ProjectTableSelectCommands(iTable)=(For all i and j)"["&
projectFieldsNames(iTable,i,j)&"]"
ProjectTableSelectCommands(0)="SELECT"&"ID,"&ProjectTableSelectCommands(0)&"FROM"&Projec
tTablesNames(0)
'here other select commands for other tables
createSelectCommand = ProjectTableSelectCommands(iTable)
End Function
```

What all these means in practice is that any modification to the program in terms of creating new interface for a new data item becomes a very simple process. The GUI with the project headers table is displayed as shown in the Figure 4.4.

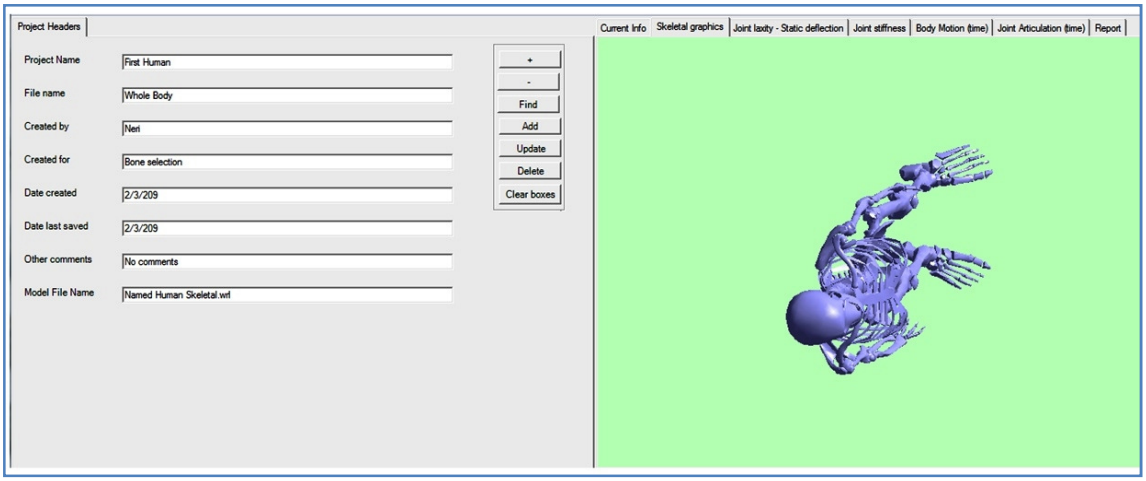


Figure 4.4 Front page of MJM

4.3 Linear and nonlinear multi rigid body formulations, algorithms and software implementation

In this section, for the sake of clarity only some elements, perceived to be important, will be covered.

4.3.1 Linear multi rigid body formulation for joint stiffness

It is possible to simplify the linear equations of motion, given in the Eqn. 4.1 and Eqn. 4.2 if the local axes of bodies are aligned with the global axes frames (the centres need not coincide). In that case all axes transformation terms **T** can be removed from the equations.

$$\mathbf{T'}_i \mathbf{M}_i \mathbf{T'}_i^T \ddot{\mathbf{U}}_i + \sum_p \left(\mathbf{T'}_i \mathbf{A}_i^T \mathbf{T}_i^T \mathbf{k}_p \mathbf{T}_i \mathbf{A}_{pi} \mathbf{T'}_i^T \right) \mathbf{U}_i - \sum_p \left(\mathbf{T'}_i \mathbf{A}_i^T \mathbf{T}_i^T \mathbf{k}_p \mathbf{T}_j \mathbf{A}_{pj} \mathbf{T'}_j^T \right) \mathbf{U}_j = 0 \tag{4.1}$$

$$\mathbf{T'}_j \mathbf{M}_j \mathbf{T'}_j^T \ddot{\mathbf{U}}_j - \sum_p \left(\mathbf{T'}_j \mathbf{A}_j^T \mathbf{T}_j^T \mathbf{k}_p \mathbf{T}_i \mathbf{A}_{pi} \mathbf{T'}_i^T \right) \mathbf{U}_i + \sum_p \left(\mathbf{T'}_j \mathbf{A}_j^T \mathbf{T}_j^T \mathbf{k}_p \mathbf{T}_j \mathbf{A}_{pj} \mathbf{T'}_j^T \right) \mathbf{U}_j = 0 \tag{4.2}$$

Furthermore it is possible to restructure the equations if force and moment equations are written in two parts rather than combined by the use of the A matrices.

$$\mathbf{A}_i = \begin{bmatrix} 1 & 0 & 0 & 0 & z_{pi} & -y_{pi} \\ 0 & 1 & 0 & -z_{pi} & 0 & x_{pi} \\ 0 & 0 & 1 & y_{pi} & -x_{pi} & 0 \end{bmatrix} \quad \mathbf{A}_i^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -z_{pi} & y_{pi} \\ z_{pi} & 0 & -x_{pi} \\ -y_{pi} & x_{pi} & 0 \end{bmatrix} \tag{4.3}$$

This can be done by using the **R** matrices

$$\mathbf{R}_{pi}=\begin{bmatrix} 0 & z_{pi} & -y_{pi} \\ -z_{pi} & 0 & x_{pi} \\ y_{pi} & -x_{pi} & 0 \end{bmatrix} \tag{4.4}$$

Now two matrix equations above simplified to the form (also explained in the Theory Chapter):

$$\begin{bmatrix} \mathbf{m}_i & 0 \\ 0 & \mathbf{J}_i \end{bmatrix}\begin{Bmatrix} \ddot{\boldsymbol{x}}_i \\ \ddot{\boldsymbol{\alpha}}_i \end{Bmatrix}+\begin{bmatrix} \mathbf{K}_p & \mathbf{K}_p\mathbf{R}_{pi} \\ \mathbf{R}_{pi}^{\mathrm{T}}\mathbf{K}_p & \mathbf{R}_{pi}^{\mathbf{T}}\mathbf{K}_p\mathbf{R}_{pi} \end{bmatrix}\begin{Bmatrix} \boldsymbol{x}_i \\ \boldsymbol{\alpha}_i \end{Bmatrix}-\begin{bmatrix} \mathbf{K}_p & \mathbf{K}_p\mathbf{R}_{pj} \\ \mathbf{R}_{pi}^{\mathrm{T}}\mathbf{K}_p & \mathbf{R}_{pi}^{\mathbf{T}}\mathbf{K}_p\mathbf{R}_{pj} \end{bmatrix}\begin{Bmatrix} \boldsymbol{x}_j \\ \boldsymbol{\alpha}_j \end{Bmatrix}=\begin{Bmatrix} \mathbf{F}_i \\ \mathbf{M}_i \end{Bmatrix} \tag{4.5}$$

$$\begin{bmatrix} \mathbf{m}_{\mathrm{j}} & 0 \\ 0 & \mathbf{J}_{\mathrm{j}} \end{bmatrix}\begin{Bmatrix} \ddot{\boldsymbol{x}}_{\mathrm{j}} \\ \ddot{\boldsymbol{\alpha}}_{\mathrm{j}} \end{Bmatrix}-\begin{bmatrix} \mathbf{K}_{\mathrm{p}} & \mathbf{K}_p\mathbf{R}_{pi} \\ \mathbf{R}_{\mathrm{pj}}^{\mathrm{T}}\mathbf{K}_{\mathrm{p}} & \mathbf{R}_{pj}^{\mathbf{T}}\mathbf{K}_p\mathbf{R}_{pi} \end{bmatrix}\begin{Bmatrix} \boldsymbol{x}_i \\ \boldsymbol{\alpha}_i \end{Bmatrix}+\begin{bmatrix} \mathbf{K}_p & \mathbf{K}_p\mathbf{R}_{pj} \\ \mathbf{R}_{pj}^{\mathrm{T}}\mathbf{K}_p & \mathbf{R}_{pj}^{\mathbf{T}}\mathbf{K}_p\mathbf{R}_{pj} \end{bmatrix}\begin{Bmatrix} \boldsymbol{x}_j \\ \boldsymbol{\alpha}_j \end{Bmatrix}=\begin{Bmatrix} \mathbf{F}_j \\ \mathbf{M}_j \end{Bmatrix} \tag{4.6}$$

These equations are easier to implement into software provided all elements of the matrices are already expressed in the global axes before starting to assemble them. This is achieved by ensuring that all bone bodies have their local axes coinciding with the global axes at the beginning of the analysis. When the analysis is executed, the centre of mass of each body is found and each local axis is moved (translated) to that position and spring attachment points are re calculated together with the inertia matrices and the stiffness matrices above are assembled.

4.3.2 Linear multi rigid body, joint stiffness software

Assembling $\mathbf{R}_{\mathrm{pi}}$, $\mathbf{R}_{\mathrm{pj}}$ and $\mathbf{R}_{\mathrm{pi}}^{\mathbf{T}}$: Note that there is no need to obtain $\mathbf{R}_{\mathrm{pj}}^{\mathbf{T}}$ this will be clear in the description of the assembly subroutine. Actually $\mathbf{R}_{\mathrm{pj}}$ may also be excluded from explicit derivation but this will make the programming logic less obvious.

4.3.2.1 Pseudo code for assembling $\mathbf{R}_{pi}$, $\mathbf{R}_{pj}$ and $\mathbf{R}_{pi}^{T}$

```
'Coordi, Coordj are inputs and Ri, Rj and Ri(transpose) are output
   Private Sub R_Mat(Coordi,Coordj Ri, Rjm  RTi)
        Ri(1, 2) = Coordi(3) : Ri(1, 3) = -Coordi(2) : Ri(2, 3) = Coordi(1)
        Ri(2, 1) = -Ri(1, 2) : Ri(3, 1) = -Ri(1, 3) : Ri(3, 2) = -Ri(2, 3)
        Rj(1, 2) = Coordj(3) : Rj(1, 3) = -Coordj(2) : Rj(2, 3) = Coordj(1)
        Rj(2, 1) = -Rj(1, 2) : Rj(3, 1) = -Rj(1, 3) : Rj(3, 2) = -Rj(2, 3)
        For all i and j
                RTi(j, i) = Ri(i, j)
    End Sub
```

4.3.2.2 Pseudo code for assembling stiffness matrix for a given spring

The code needs to be studied carefully. There are four (6 x 6) matrices, each containing four (3 x 3) matrices given in the Eqn. 4.4 and Eqn. 4.5. The four (6 x 6) matrices are structurally identical, what differentiates them is the order in which the $\mathbf{R}_{pi}$ and $\mathbf{R}_{pj}$ matrices are positioned. The order is controlled by positioning r_i or r_i (*Coordi, Coordj*) in the parameter list of the routine ***AssembleSubStiffness***. For example the first matrix of Eqn. 4.4 is obtained by points r_i and r_i, (the first, 6 x 6 matrix contains $\mathbf{R}_{pi}$ only). The second matrix of the Eqn. 4.4 contains r_i and r_j, the first matrix of the second equation is given by r_j and r_i and finally the last matrix in the second equation is given by r_j and r_j points. Therefore we need only one routine, listed below, for assembling the 6 x 6 matrices and one routine for assembling the sub matrices $\mathbf{R}_i$ and $\mathbf{R}_j$ and $\mathbf{KR}_i$ in the Eqn. 4.4 and the Eqn. 4.5. Of course the routine below needs to be called four times with coordi and coordj, copying points (i,i), (i,j), (j,i) and (j,j). This now assembles all the matrices required to describe the final structure.

```
'inputs stiffness matrix K, coordi, coordj and returns the global stiffness matrix GK
'GK is a 6x6 matrix if coordi=coordinates on mass i and coordj also coordinates on mass
i
'then the first stiffness matrix in the first equations is assembled
Private Sub AssembleSubStiffness(K, Coordi, Coordj, GK)
     Call R_Mat(Coordi, Coordj, Ri, Rj, RTi)
     Call MatMult(K, Rj, KR)
     Call MatMult(RTi, K, RTK)
     Call MatMult(RTi, KR, RKR)
         For all i and j, each going from 1 to 3
               GK(i, j) = K(i, j)
               GK(i, j + 3) = KR(i, j)
               GK(i + 3, j) = RTK(i, j)
               GK(i + 3, j + 3) = RKR(i, j)
 End Sub
```

4.3.2.3 Pseudo code for assembling the global stiffness matrix

Having obtained the stiffness equations of individual stiffness matrices, these now can be assembled into the final global matrix equations. It is required to locate the position in the global matrix and add each new entry to the existing content of the matrix. The position of each of four sub matrices depends on the i^{th} and j^{th} mass numbers (*imas* and *jmas*)

```
'inputs imas, jmas, GK and returns stiffness
'Stiffness now is the global stiffness matrix
Private Sub AssembleGlbMatrix(imas, jmas, GK, Stiffness)
  For all i and  j each going from 1 to 6
    Stiffness((imas-1)*6+i,(jmas - 1)*6+j)=Stiffness((imas-1)*6+i,(jmas-1)*6+j)+GK(i, j)
  End Sub
```

4.3.2.4 Assembly matrix has to be called for each spring (muscle, ligament)

This code assembles a given spring/tissue (mount) at the correct positions in the final global matrix "stiffness".

```
For MountCount = 1 To TotalMount
 For i=1 to 2
  For j=1 to 2
ip = PointNoAtEachEndOfMount(MountCount,i)
jp = PointNoAtEachEndOfMount(MountCount,j)   'points numbers at each end of a tissue
FOR ALL k values, coordi(k) = CoordOfPoint(ip, k) : coordj(k) = CoordOfPoint(jp, k)
AssembleSubStiffness(ks, coordi, coordj, GK)
AssembleGlbMatrix(MassNoOnWhicPointLOcated(ip),MassNoOnWhicPointLOcated(jp),GK,
                  stiffness)
  Next
 Next
Next
```

4.3.2.5 Preparing for the analysis

Having assembled the global equations, it is needed to solve these with geometrical constraints or end conditions superimposed on the bodies. As long as the constraints are described in the global axes, processing these is a relatively standard procedure for linear systems. The constraints analysis normally involves fixing one (or more) of the bodies in the space so that the motion of the other(s) can be studied. Or fixing one or more of the freedoms of a body. The software reads these constrained directions and identify them, both as equation numbers as well as variable numbers. These constraints then introduced into the equations of motion. The introduction involves two types of processing, either the equation of motion matrices are shrunk by removing equations

corresponding to these variable (appropriate rows and columns in the global mass, stiffness and damping matrices) or instead of reducing the equations of motion, the equations are re arranged and solved for unknown force corresponding to known deflection (fixing means zero motion). The second option also allows the introduction of prescribed motion and thus inverse analysis. This is a relatively simple matrix manipulation exercise. The first option, the matrix shrinking is demonstrated by two calls, where the constraint vector is passed on to the ConstraintMapping routine and the routine calculates what is the new variable number corresponding to its old (original) variable number and other way round, so that relevant matrices and vectors can be expanded or shrunk.

```
Call ConstraintMapping(constraints, NewDimension, Shrink, Expand)
Call ShrinkMatrix(NewDimension, Shrink, stiffness, StiffnessFinal)
```

Shrinking is demonstrated in matrix form as shown below, where j^{th} and k^{th} variables are assumed to zero (columns to be removed) and corresponding equations of the j^{th} and k^{th} motion equations (rows to be removed) are removed from the analysis. "*x*" indicates the removed elements.

$$\begin{array}{c} \\ j \\ \\ k \\ \\ \end{array}\begin{array}{c} \begin{array}{ccccc} j & & k & & \end{array} \\ \begin{bmatrix} l_{11} & x & & x & \\ x & x & x & x & x \\ & x & & x & \\ x & x & x & x & x \\ & x & & x & l_{nn} \end{bmatrix} \end{array} \xrightarrow{\text{constrained } j \,\&\, k} \begin{bmatrix} l_{11} & & \\ & & \\ & & l_{n-2,n-2} \end{bmatrix} \tag{4.7}$$

It is also possible to process simple constraints (selected variables being zero) without reducing the matrices. Strictly speaking, in a system, zeroing a given set of variables implies that the motion is restricted by applying a force and preventing them from moving. In other word, an unknown displacement is replaced by an unknown force and by re arranging equations, the solution can proceed. This is demonstrated below in term

of matrix manipulations. It is assumed that j^{th} displacement is known (zero or else). Note that the process presented here is the only option if pre constrained motion is non zero and thus the shrinking cannot be used.

$$\begin{bmatrix} l_{11} & l_{12} & .. & l_{1j} & l_{1n} \\ l_{21} & l_{22} & .. & l_{2j} & l_{2n} \\ .. & .. & .. & .. & .. \\ l_{j1} & l_{j2} & .. & l_{jj} & l_{jn} \\ l_{n1} & l_{n2} & .. & l_{nj} & l_{nn} \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ .. \\ x_j \\ x_n \end{Bmatrix} = \begin{Bmatrix} f_1 \\ f_2 \\ .. \\ f_j \\ f_n \end{Bmatrix} \tag{4.8}$$

In this equation x_j is known and f_i is unknown, re arranging the equations,

$$\begin{bmatrix} l_{11} & l_{12} & .. & 0 & l_{1n} \\ l_{21} & l_{22} & .. & 0 & l_{2n} \\ .. & .. & .. & 0 & .. \\ l_{j1} & l_{j2} & .. & -1 & l_{jn} \\ l_{n1} & l_{n2} & .. & 0 & l_{nn} \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ .. \\ f_j \\ x_n \end{Bmatrix} = \begin{Bmatrix} f_1 \\ f_2 \\ .. \\ 0 \\ f_n \end{Bmatrix} - x_j \begin{Bmatrix} l_{1j} \\ l_{2j} \\ .. \\ l_{jj} \\ l_{nj} \end{Bmatrix} \tag{4.9}$$

To achieve the manipulations above, the following steps are to be executed, multiply the column of the matrix with known prescribed displacement and take it to the right hand side of the matrix equation, of course with –ve sign. Now insert zero in every element of the column corresponding to that variable. Now enter -1 at $(j, j)^{th}$ position. The final matrix is as shown in the Eqn. 4.9. Note that if there are more than one prescribed displacements, the procedure is performed for each in the same fashion. The order of the matrix does not change.

In terms of implementations, both methods are implemented in the program. The reason is that the skeletal system contains a large number of bones (bodies) most of which are not functional for a given joint analysis. It means that the analysis formulation will result in extremely sparse matrices if shrinking is not used. Having

implemented the both solutions, it was realised that for non linear analysis it would have been much better if all non-participating bones were left out of analysis altogether. However there is nowhere in the database where exclusion of a body from analysis can be indicated without reading all the bodies into the database in the first place. In this case, two possible options are considered, one is to use a naming convention for bodies in the VRML file indicating which bodies will be excluded from the analysis. The second option is to change the constraints defaults from "free" to "constrained" thus during the data preparation session the user can update few entries from the "constrained" to the "free". However the resulting sparse matrix may give problems if it is used for the inverse analysis. There is a final alternative option and that is to leave this question unanswered for now and instead of using the whole skeleton, create VRML files only with the bones relevant to the project. This option would be sufficient for the research carried out here. Therefore, for the two case studies presented in the results section, two bone files were created instead of using the whole skeleton.

4.3.3 Nonlinear multi rigid body formulation

The linear analysis option resented above is only useful for studying the joint stiffness, the centre of stiffness, the laxity and small motion at a given geometrical orientation. However general motion analysis of the musculoskeletal system requires non linear analysis by the fact that the large geometrical changes are expected during the motion. Along with the geometrical changes, tissue surrounding the joint and cartilage also exhibit non-linear behaviour (material non-linearity). The geometrical non-linearity requires geometry updating, material non-linearity involves applying either polynomial non-linearity curves or following the stiffness curves along their deflection path based on the raw data as explained previously. As explained in the previous chapter, assembling the equations in a simple matrix form is not possible when stiffnesses are non-linear. This type of nonlinearity may exist even if the motions are small. Non-linear stiffness cannot be expressed in the tensorial form.

The flow diagram in Figure 3.4, gives the steps for the non linear analysis. In that procedure the main logic involves calculation of non linear spring stiffness along its "principal" stiffness directions. These are described as those directions along which forces do not create deflection in any other directions. In general this kind of definition

for discrete stiffness is not always applicable. Even if such directions exist, they exist in a finite configuration and do not remain unchanged as the deflection progresses. Therefore this is an approximation. However for muscles and for soft tissue the assumption is well adequate, in fact muscles can carry tensile loads only along the main load direction. As for the shear loads, they are either very small or non-existent. In the current program only the stiffness along the muscle/tendon and ligament is considered.

4.3.3.1 Initialise geometry

The very first step in the non-linear analysis is to initialise the geometry

1. Establish initial conditions

This part of the program simply calculates the body parameters that need not be re-calculated during the simulation process such as mass, inertia and principal inertias and points on the body relative to the local axes system and principal axes system. These relative to the both axes systems are fixed and need not be re-calculated. Initialisation also calculates data which are necessary to start the calculation but needs to be updated during the simulation. Therefore, part of the initialiseGeometry routine is identical to UpdateGeometry routine. The *muscleLigamentAttachments* is the main routine calculating muscle wrapping and its effects. The muscle wrapping is capable of dealing with the multiple body wrapping.

2. Read ligament end coordinates on bodies ***i*** and ***j***

3. Calculate number of segments for the tissue going from ***i*** to ***j*** over a number of other bony surfaces. For example if it is only ***i*** and ***j*** then the segment is one and the point where segment goes tangent to the ith surface “fly” and the point where it arrive at the ***j***th surface and the directions of these tangents need to be calculated. This information is necessary for force and moment calculations. If the tissue, going from ***i*** to ***j*** travels over other bony surfaces then the number of segments will be increased accordingly. For 3 bodies then 2 segment and for 4 bodies then 3 segment etc. total number of segments will always be one less then the number of bodies.

Output of the algorithms would be as:

```
muscleLigamentAttachments(noOfSegments, SegmentTension, BetweenBodies, PointI, pointJ,
globFrcDirI, globFrcDirJ)
```

Note that the *muscleLigamentAttachments* call has only output parameters, no input parameters as these are declared as the global variables. It returns the *noOfSegments*, how many segments exist in a given tissue path, *SegmentTension*, segment tension (even if it travels over a number of bodies, the tension remain constant), body pairs *BetweenBodies* for each segment, *PointI*, *pointJ* are end points of each segments in the global coordinates and finally *globFrcDirI*, *globFrcDirJ*, these are force directions at *pointI* and *pointJ*. Resulting segment information is collected for the later use as follows:

```
LigamentSegmentCounter = 0
   For j = 1 To noOfSegments
    LigamentSegmentCounter = LigamentSegmentCounter + 1
    ligamentSegmentTension(LigamentSegmentCounter) = SegmentTension
    LigamentSegmentBetweenBodies(LigamentSegmentCounter, 1) = BetweenBodies(j, 1)
    LigamentSegmentBetweenBodies(LigamentSegmentCounter, 2) = BetweenBodies(j, 2)
       For k = 1 To 3
          ligamentSegmentPoints(LigamentSegmentCounter, 1, k) = PointI(j, k)
          ligamentSegmentPoints(LigamentSegmentCounter, 2, k) = pointJ(j, k)
          ligamentSegmentFrcDir(LigamentSegmentCounter, 1, k) = globFrcDirI(j, k)
          ligamentSegmentFrcDir(LigamentSegmentCounter, 2, k) = globFrcDirJ(j, k)
       Next
    Next j
```

4.3.3.2 Updating geometry during the solution

The updating geometry involves calculation of tissue segment information as of the geometry initialisation part of the program. However the calculation of inertias, since the local and the principal axes are fixed to the body, is not needed. In addition to the muscle wrapping algorithm the collision detection needs to be performed. In general, the initialisation, to be absolutely sure that the simulation starts with no bone intersecting any other bone, should perform Boolean intersection. This may be implemented if the software to be made publicly available to other users, however for the purpose of this thesis, there is no urgency of such implementation. Since the bodies are in move, the collision detection needs to be performed at each stage. The software implemented two collision detection method, one developed in this research as explained in the theory chapter and the other supplied by the KernelCAD library. KernelCAD library, like all commercially available collision detection algorithms performs shortest distance queries incrementally. This is true for the OBB based RAPID which is the most widely used algorithm available in industry. The algorithm

which is proposed in this project can perform "time to collision" analysis which eliminates iterative detection and it is fundamentally different than the RAPID's incremental solution. However the RAPID gains its efficiency by performing very effective culling which is missing in the algorithm proposed here. In the process of compiling the thesis, the KernelCAD was in the process of implementing OBB as the collision detection method. Some basic elements of the ***updateGeometry***, is listed below:

```
Sub updateGeometry(dt, y(), dy())
  ' for each body cog, cogvelocity, angular velocity of principal axes, find its new
    orientation (new orientation of principal axes system of each body)
  ' calculate muscle attahment ends (in the global axes)
  ' perform muscle wrap in the same order as of the original muscle construction
  ' calculate muscle attachment points (muscle centroids corresponding points on the
    convex hull, the last point on one bone and the last point on the jth bone

Motion of bodies contained in the array y(), obtained by runge kutta is translated here
and bone positions are updated, note that y() is the state space vector containing
displacments and velocities of bodies relative to their principal axes sysems

    For i = 1 To NumberOfBones
            translationalVel(j) = y(6 * NumberOfBones + (i - 1) * 6 + j)
            rotationalVel(j) = y((6 * NumberOfBones + (i - 1) * 6 + 3 + j))
            RotateLocalAxes(dt, rotationalVel, dircosIn, dircosOut)
            'this rotates local axes and ensures that the orthogonality is satisfied
    Next
   'all axes are rotated relative to the global axes, giving new direction cosines
   'now mesh geometry to be updated in order to start ligament and cartilage
    Calculation to start.
    For i = 1 To NumberOfBones
     Update mesh geometry of each bone(i)
    Next

    'now we need to update ligament attachement points relative to the global axes
     (theseare invariant relative to the local axes).
    For i = 1 To LigamentCounter
      Update ligament attachment points on each body relative to the glbal axes
    Next

    Calculate ligament forces due to muscle wrapping
    LigamentSegmentCounter = 0
    For i = 1 To LigamentCounter
      muscleLigamentAttachments(noOfSegments, SegmentTension, BetweenBodies, PointI,
                                pointJ, globFrcDirI, globFrcDirJ)
    Next i

    CollisionKernelCAD(m_iSectionFrom, m_iSectionTo, cog1, cog2, _
            cogVel1, cogVel2, AngVelocityVect1, AngvelocityVect2, _
            CrtlgPi, CrtlgPj, Crtlgni, Crtlgnj, nCrtlgConnections,
            CrtlgtimeToCollision)
    The call here can be replaced by
    Collision(m_iSectionFrom, m_iSectionTo, cog1, cog2, _
            cogVel1, cogVel2, AngVelocityVect1, AngvelocityVect2, _
            CrtlgPi, CrtlgPj, Crtlgni, Crtlgnj, nCrtlgConnections,
            CrtlgtimeToCollision)
End Sub
```

Having calculated the tissue paths and able to update this as the motion progresses, ***muscleLigamentAttachments*** and ***Collision*** both give necessary information to calculate

muscle and ligament forces, their action positions and directions. Calculation of forces is not given here but it is easy to follow from the main program supplied with the thesis. The motion of the bodies described in the standard state space form is given in the FCT function used by the Runge Kutta algorithm. The program has two version of the Runge Kutta, one with variable step length and the other is user controlled step length. As it will be discussed later the variable step length proved to be relatively ineffective because change of tissue orientation by muscle wrapping and the collision caused stiffness fluctuations resulting very small integration step length.

The pseudo code is listed below which covers everything about the non-linear implementation. The only missing software implementation detail left to be explained relates to the implementation of the joint stiffness dynamically. The `updateGeometry` generates instantaneous stiffness information at each time step and the corresponding orientation at which the data about the tissue attachment points, force directions (tissue directions) at those points are all known and sufficient to generate the stiffness matrix given previously. The instantaneous joint stiffness was implemented but not fully tested at the time of writing the thesis.

4.3.3.3 Overall pseudo code

Most of the elements of the overall program have been already given. Through collecting these and giving further information about tissue wrapping and collision detection, a summary pseudo code is provided below.

Read system parameters, Obtain mass and inertia of bodies, obtain principal inertias and axes, convert spring attachment points to local axes frames. With this no need for mass matrix conversion.

If time >= total integration time then stop, else continue

For i = 1 to N tissue attachments (including muscle and ligament)
For each tissue attachment peform muscle wrapping algorithm
{Obtain cylindrical coordinates joining two end points of the attachment and perform minimum mass
Or minimum covariance to obtain the "shortest path",
Along this path peform "convex hull"
Along the path and r distance away from the surface (using surface normals), construct "centroid" path.
Record the mass number from which the centroid points are created from.
Starting from the point of attachment, follow the centroid points until the mass number number changes.
The last point before the mass number change is the point which is to be used in calculating the moment Arm. This is the "spring attachment point". At the end of the muscle wrapping algorithm,

following information is obtained: 1. spring attachment points, 2. on which body the point is located, 3. Direction of the tissue at that point, 4. Extension of the tissue, 5. Tension in the tissue (based on polynomial or any other formulation) and 6. Direction of force.
For each tissue attachment collect how many sub segments exist and create new tissue segment list (this way now we can handle each tissue path going over multi bodies)
Next tissue

Perform collision algorithm to find a collection of contact points (surface contact points within a range), there areTwo algorithms currently being used, user can choose one or the other.
A) Collision based on the algorithm came with the graphics platform component.
This is based on stepwise shortest distance calculations. A method used by all collision analysis algorithms currently available in commercial code.
B) Algorithm developed and presented in this paper which perform "time to collision" predictive analysis.

If the angle between the surface normal and the direction of relative motion between bodies is more than 90^0 then exclude the vertex from the search.
If the radial distance between the approaching vertices (fly-past) greater than a tolerance then exclude the pair from the search.
For the remaining vertices compare Minimum distances (Mind) to find the smallest
Mind/approach velocity = time to collision
The following statement find the Mind.

$$if\ \mathbf{n}_{pi}.\mathbf{e}_v > 0\ and\ \mathbf{n}_{pj}.-\mathbf{e}_v > 0\ and\ \left|\mathbf{l}_{ij}\times\mathbf{e}_v\right| < tol\ and\ abs\left(\mathbf{l}_{ij}.\mathbf{e}_v\right) < Mind_{ij}\ then$$
$$Mind_{ij} = \mathbf{l}_{ij}.\mathbf{e}_v$$

<u>At this stage all connections between bodies are established</u>

For i = 1 to total ligament segments
Find body numbers attached at each end of the ligament segment (ligament segment defined above)
Attachement coordinates and force vectors are known
Assemble these into the equations of motion (depending on the mass numbers) in state space form.
Next i
For i=1 to total number of contact points
Find body numbers attached at each end of the ligament segment (ligament segment defined above)
Attachement coordinates and force vectors are known
Assemble these into the equations of motion (depending on the mass numbers) in state space form.
Next i
Use RUNGEKUTTA to integrate
Increment time step

4.3.4 Mass and inertia tensor calculations

This is implemented from the pseudo code of Gauss Divergence Theorem developed by Mirtich (1996a) and cited in "Game Physics" book (Eberly, 2003). The subroutine reads m_iCurObject and returns mass, cog and inertia tensor. m_iCurObject provides surface information in terms of triangulated surface mesh data. The routine returns the mass, the centre of mass and the inertia matrix.

```
computeInertia(m_iCurObject, mass, COG(),Inertia(,))
```

For simplicity some dimensioning are removed but the subroutine is mainly complete and listed in the Appendix A5.

4.3.5 Calculation of principal inertias

In order to calculate the principal inertia, the Eigen value analysis is adopted form the LAPACK library and converted from Fortran to VB.Net. Although the principal axes are calculated by simply passing the inertia matrix to the Eigen solver, the results are obtained in terms of the direction cosines matrix and in order to use it with the KernelCAD library, these needed to be converted to the Euler angle definitions. Furthermore the principal axes obtained from the Eigen solver, needed to be orthogonalised. Although this was supposed to be orthogonal, after lengthy experimentation, it was discovered that the axes system may not always be exactly orthogonal. An example of the results from the principal axes calculation is given in Figure 4.5.

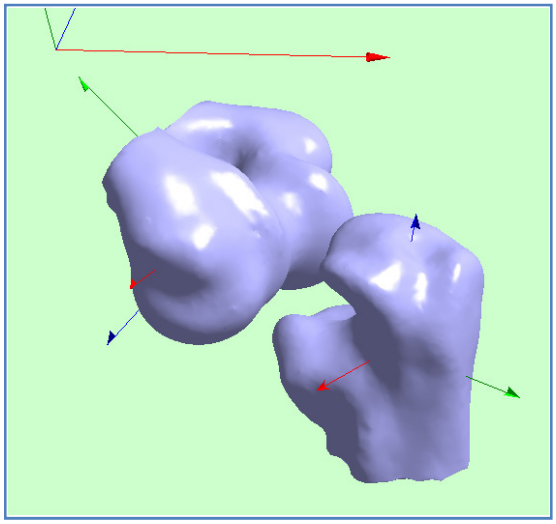


Figure 4.5 The axes of the principal inertias of two bone segments

4.3.6 Finding Euler angles from direction cosines

The Euler angles describe the orientation of an axes system in space in terms of three finite rotations. The order of rotation is important and the final configuration of the axes system depends on the order in which the rotations are performed along the orthogonal axes of the rotating frame. Therefore in converting a transformation matrix (direction cosines matrix) to the Euler rotations, it is necessary to specify which particular rotation orders are selected in order to orient a rotated frame relative to the global axes. The case to be illustrated as an example, the first rotation is taken to be in the z direction, the second in y and the final rotation is in x direction. Generally particular order depends on the application. If the sole purpose of obtaining the Euler angles from the direction cosines is to display the local axes on the body, then one can choose any one of the options. As the orientation of the axes frame in the KernelCAD Platform is manipulated by axes rotations only, obtaining the Euler angles is necessary in order to display these axes.

$T(\theta_z)T(\theta_y)T(\theta_x) \rightarrow (3, 2, 1)$

For this particular transformation the transformation matrix is calculated as:

$$T = T(\theta_z)T(\theta_y)T(\theta_x) = \begin{bmatrix} \cos\theta_y \cos\theta_z & \sin\theta_x \sin\theta_y \cos\theta_z - \cos\theta_x \sin\theta_z & \cos\theta_x \sin\theta_y \cos\theta_z + \sin\theta_x \sin\theta_z \\ \cos\theta_y \sin\theta_z & \sin\theta_x \sin\theta_y \sin\theta_z + \cos\theta_x \cos\theta_z & \cos\theta_x \sin\theta_y \sin\theta_z - \sin\theta_x \cos\theta_z \\ -\sin\theta_y & \sin\theta_x \cos\theta_y & \cos\theta_x \cos\theta_y \end{bmatrix}$$

By inspection

$$\theta_y = \sin^{-1}(-T(3,1))$$

$$\theta_z = \tan^{-1}(T(2,1)/T(1,1))$$

And finally,

$$\theta_x = \tan^{-1}(T(3,2)/T(3,3))$$

The subroutine processing this receives the TRANS, transformation matrix, and the order which the user request the Euler angles to be calculated and the routine returns the Euler angles. Similar calculation shown above are performed for all possible combinations of the order of rotations (Appendix A4).

```
Sub FindEulerAnglesFROM_TRANSF(RotationOrder(),TRANS(,),EulerAngle())
    If RotationOrder(1) = 3 And RotationOrder(2) = 2 And RotationOrder(3) = 1 Then
        'first z, second y and finally x rotations
        EulerAngle(3) = System.Math.Atan2(TRANS(2, 1), TRANS(1, 1))
        EulerAngle(2) = System.Math.Asin(-TRANS(3, 1))
        EulerAngle(1) = System.Math.Atan2(TRANS(3, 2), TRANS(3, 3))
    Else If ( 5 REMAINING OPTIONS))
        -------
    End If
End Sub
```

4.4 Geometric tissue path finding, line of action and wrapping algorithms and software implementation

4.4.1 Muscle mesh generation, topology and data

In this section, the muscle mesh generation scheme will be explained. In order to describe mesh information, one needs to describe the topology or nodal connectivity and the coordinates corresponding to the nodes. In practical terms the connectivity involves numbering the mesh elements and listing the node numbers, in anticlockwise direction, for each element. Therefore mesh generation involves automating the element and node numbering. The next step is to obtain the geometry of each node. Once these are obtained the mesh surface can be generated. The topology proposed in this project is designed to be suitable to generate any closed volume and the current program can generate the standard geometrical shapes such as, sphere, box, cylinder,

cone, and with a small modification to the topology it can also generate the torus. What determines the shape is the geometry since the topology for these shapes is the same. The muscle geometry is determined by the two end points and the tangent path curve on the bone surface and the normals along the path. The muscle geometry is generated by employing a number of possible geometry schemes, either it is cylindrical with a fix radius, except the last two end points where the cylinder section joins to these points forming two end cones, or the radius of the muscle section is determined by some function which ensures that the section starts with zero diameter and gradually increases and then decreases back to zero. For this various functions are used such as the half sine curve, or full shifted cosine curve, such as A-$A\cos(\theta)$. Surface normals are used to determine the centroid of the muscle. The nodal numbering scheme is given in the Figure 4.6.

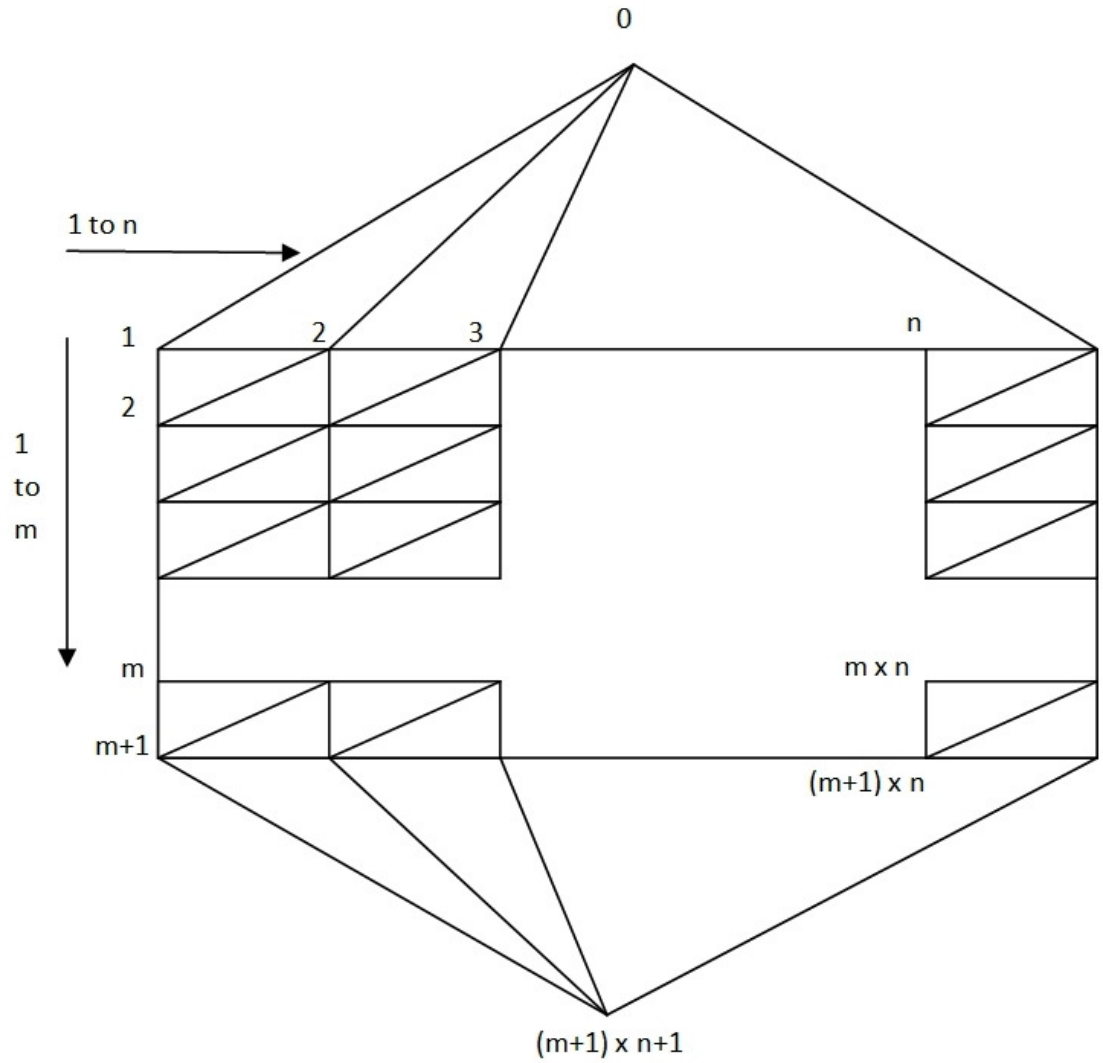


Figure 4.6 Nodal and topological information of a closed volume mesh generation

There are m rows and n columns of the rectangles, each divided into two triangles. The actual node numbers, listed row by row are given as:

(1,2,3 ... n, 1)
(n+1, n+2, ... 2n, n+1)
(2n+1, 2n+2, 3n, 2n+1)
....
(mn+1, mn+2, (m+1)n, mn+1)

Numbering the rectangular elements (element numbers),

(1, 2, 3 ... n)
(n+1, n+2, ... 2n)
....
((m-1)n+1, (m-1)n+2, ... mn)

The triangle numbers (triangular element numbers) as there are two triangles for each rectangle, now the numbering of triangles corresponding to each rectangle is placed in bracket as it can be inspected by comparing the list above and the list below.

(1,2), (2,3), ... (2n-1, 2n)
(2n+1, 2n+2), ... (4n-1, 4n+2)
...
(2(m-1)n+1, 2(m-1)n+2), ... (2mn-1,2mn)

At this stage all nodes are numbered and all elements are also numbered. It is just needed to associate each element with the elemental node numbers. The full code doing this is given in the Appendix A5. An example is given in the Figure 4.7 for a sphere and muscles (used by the muscle wrapping algorithm). Muscle, box, sphere, cylinder, cone all have the same topological structure and the nodal connectivity describes any one of them. Of course in each case the nodal coordinates are different.

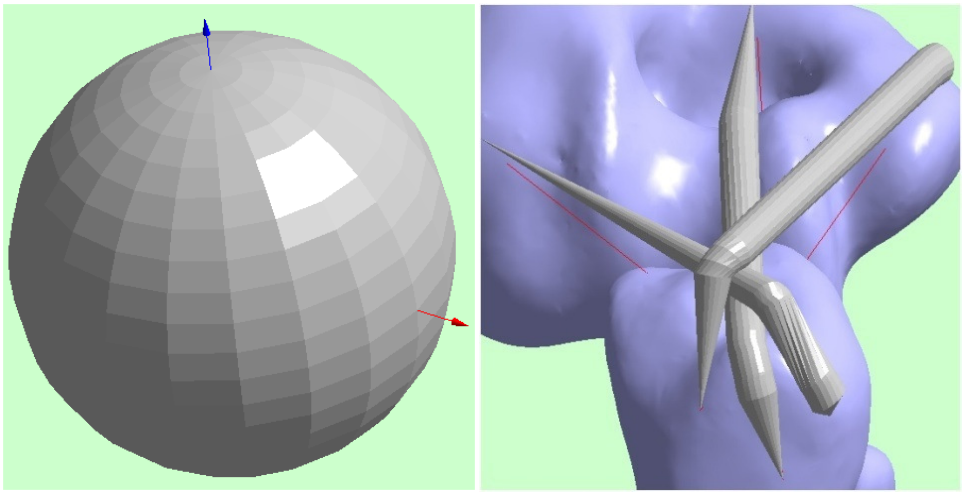


Figure 4.7 Some examples of mesh generation, the mesh density can be controlled

4.4.2 An evaluation of the minimum mass algorithm for muscle wrapping

The muscle wrapping scheme is given in detail in the theory chapter. The effectiveness of the scheme needs to be tested. However this is a very difficult, almost impossible task since unless another similar scheme is found and evaluated on the same free surface, the comparison would not be valid. The only option is to test the scheme against an analytically optimal shortest path. The shortest path on a flat surface is a straight line between two end points. When the flat surface is rolled into a cylinder the line become a helix. Therefore between two points on a cylinder the shortest path is a helix. This is shown in the Figure 4.8.

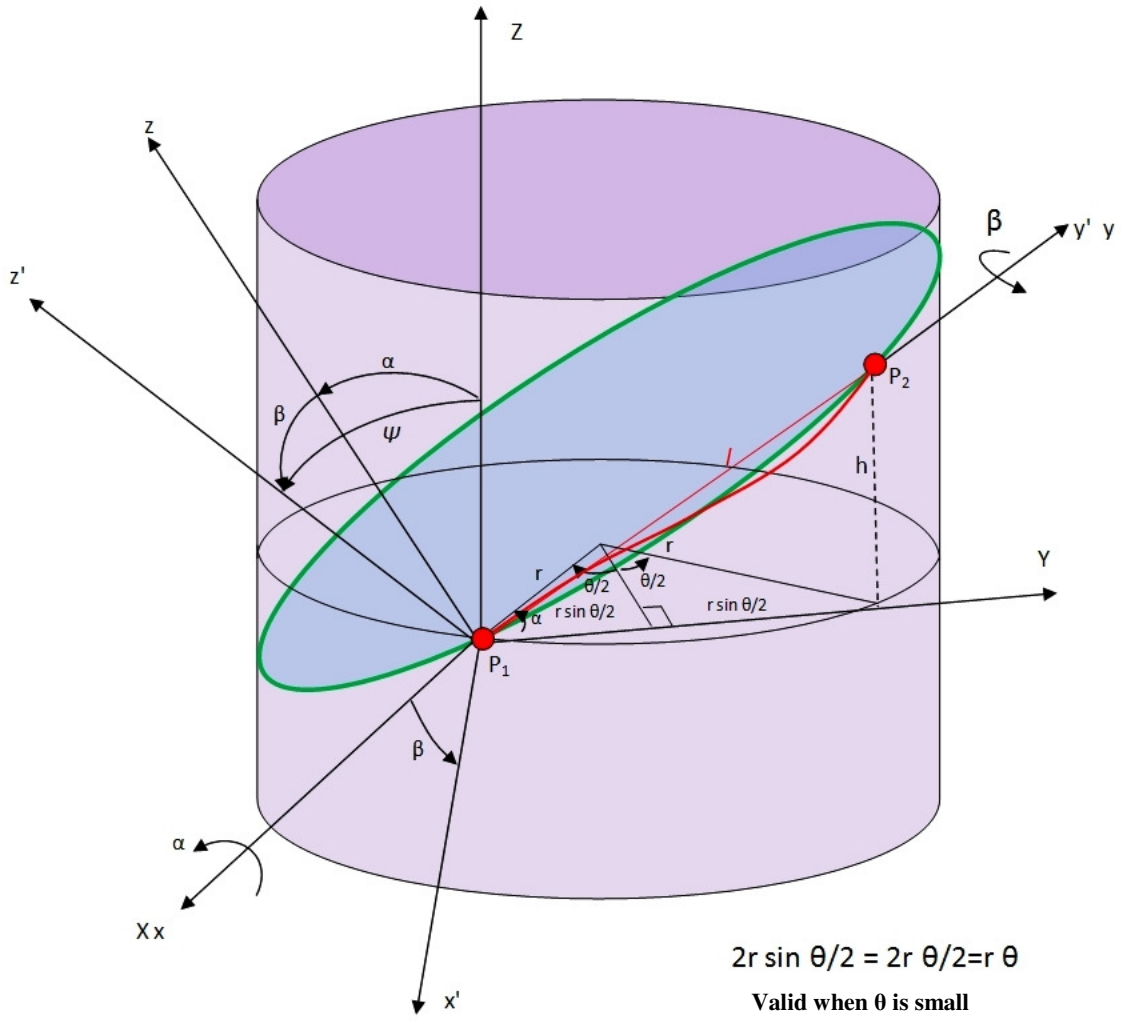


Fig. 4.8 Evaluation the analytical shortest path against the proposed muscle wrapping algorithm

The scheme proposed by the minimum mass / minimum covariance based shortest path which involves cutting the cylinder with a plane and the surface intersection between the cylinder and the plane is the path to be evaluated. The search for the minimum path involves rotating the plane around the straight line between these two points until the surface path between these points becomes the minimum. The path is an ellipsoidal curve since the intersection of a cylinder with a plane produces an ellipsoid. The analysis produced here studies the relationship between the ellipsoidal path and optimal helix path and calculates the error between them.

From the Figure 4.8, the relationship between the length (l) of the ellipsoid arc (in green) from point P_1 to point P_2 can be calculated from the relationship that the projection of this arc on to the circular cross section is given as:

$$l\cos\psi = l' \tag{4.10}$$

Where

$$l' = r\theta \tag{4.11}$$

Although the relationship seems rather straight forward the length l depends on the ellipsoid, which itself depend on the orientation of the plane, intersection of which generates the ellipsoid. The only parameter which determines the length of the ellipsoid arc is the angle between the ellipsoid plane and the cross sectional circular plane. This is given by ψ.

The algorithm has been developed to search for planar sections between two points involve generating planar sections passing through these two points and finding the shortest path between them. For the cylinder considered above this is equivalent to varying the angle β. The plane is described by Ox'y' and the angle between this plane and the circular cross section plane is determined by the angle between their normals.

This angle given in the Figure 4.8 is ψ, and the relationship between three angles are given as:

$$\cos\psi = \cos\alpha\cos\beta \tag{4.12}$$

Since

$$l = \frac{l'}{\cos\psi} \tag{4.13}$$

In order to minimise *l*, the $\cos\psi$ must be the maximum or ψ itself be the minimum. Since in the chosen axes system α is fixed, ψ can be minimum, only when β is zero.

$$l_{shortest\_ellipsoid} = \frac{r\theta}{\cos\alpha} \tag{4.14}$$

Therefore the ellipsoid path can be calculated. It is possible to do further simplification as for this position the following relationship can be written as:

$$\tan\alpha = \frac{h}{2r\sin\frac{\theta}{2}} \tag{4.15}$$

$$\cos\alpha = \frac{2r\sin\frac{\theta}{2}}{\sqrt{h^2 + 4r^2\sin^2\frac{\theta}{2}}} \tag{4.16}$$

The shortest distance between the two points can be calculated geometrically. If the cylinder, cut opened to a flat plane then the shortest distance would be a straight line. Therefore the shortest distance is given as:

$$l_{shotest} = \sqrt{(r\theta)^2 + h^2} \tag{4.17}$$

$$error = l_{shortest\_ellipsoid} - l_{shotest} = \frac{r\theta}{\cos\alpha} - \sqrt{(r\theta)^2 + h^2} \tag{4.18}$$

$$error\_ratio = \frac{l_{shortest\_ellipsoid} - l_{shotest}}{l_{shotest}} = \frac{\frac{r\theta}{\cos\alpha} - \sqrt{(r\theta)^2 + h^2}}{\sqrt{(r\theta)^2 + h^2}} \tag{4.19}$$

$$error\_ratio = \frac{r\theta}{\cos\alpha\sqrt{(r\theta)^2 + h^2}} - 1 \tag{4.20}$$

$$error\_ratio = \frac{r\theta\sqrt{h^2 + 4r^2\sin^2\frac{\theta}{2}}}{2r\sin\frac{\theta}{2}\sqrt{(r\theta)^2 + h^2}} - 1 \tag{4.21}$$

Error ratio will be maximum, when θ is 180^0. Therefore substituting 180^0 for the angle θ to study the Eqn 4.21 for the maximum error,

$$error\_ratio = \frac{r\pi\sqrt{h^2 + 4r^2}}{2r\sqrt{(\pi r)^2 + h^2}} - 1 \tag{4.22}$$

Now introduce a variable (k) to describe h in terms of r

$$h = kr \tag{4.23}$$

$$error\_ratioMax = \frac{\pi}{2}\sqrt{\frac{4+k^2}{\pi^2+k^2}} - 1 \tag{4.24}$$

Ellipsoidal and helix curves are shown plotted in the Figure 4.9 and the error between these curves is plotted versus k in the Figure 4.10.

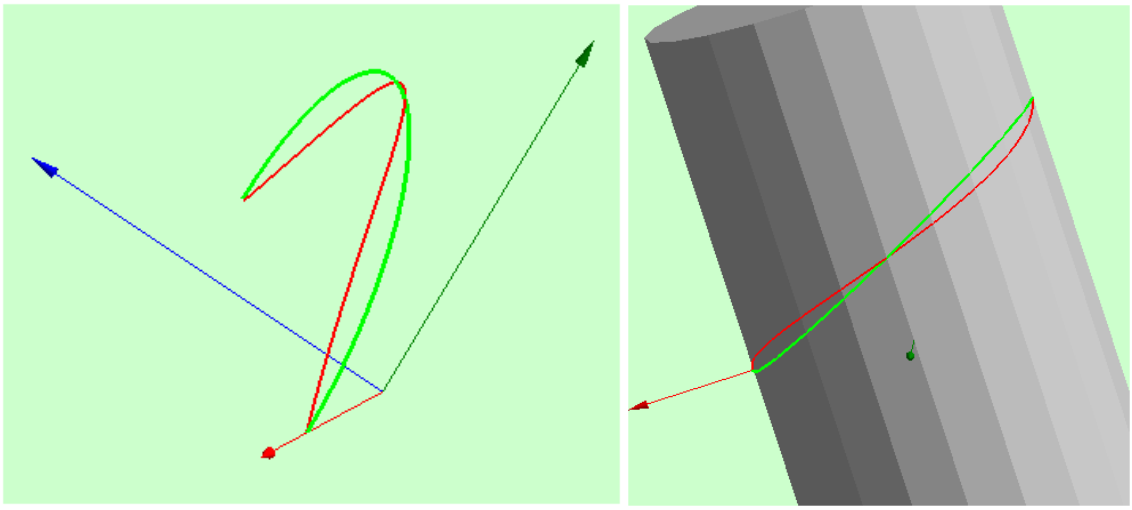


Figure 4.9 Ellipsoidal and helix curves plotted on a cylindrical mesh generated by the mesh generator

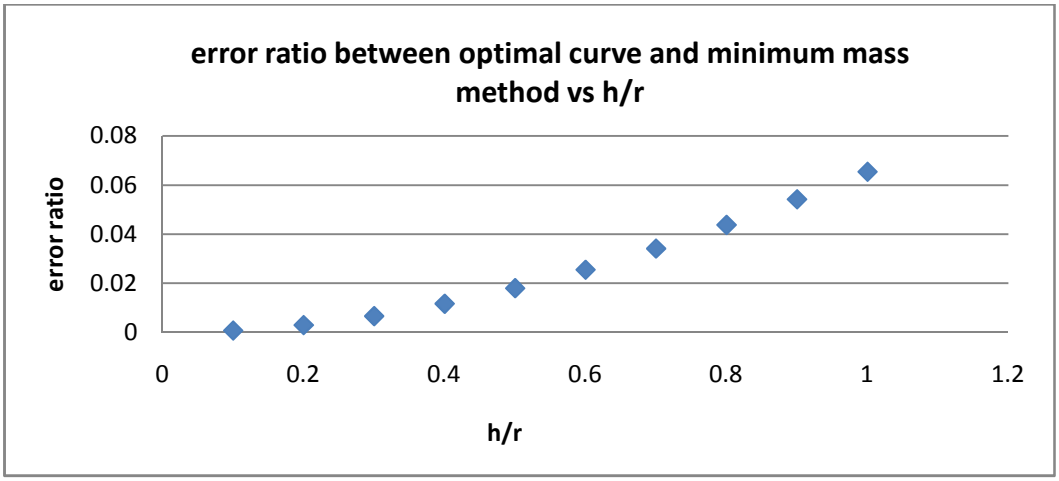


Figure 4.10 Error ratio between the optimal curve and the algorithm proposed in this thesis

The algorithm developed here is compared to the optimally analytic solution gives less than 8% error when the maximum error situation is considered, when path goes from front to the back of the cylinder. When the point at the back of the cylinder moved along the length of the cylinder then the error started increasing which was expected but less than 10% even for the extreme attachment points.

4.5 Collision detection, contact and collision response algorithms and software implementation

4.5.1 Evaluation of the collision algorithm

For collision detection, several methods were tested. A) The collision algorithm developed based on the KernelCAD library's nearest point calculation routine, B) The exhaustive search method which involved finding the nearest distance by searching every vertex on one body against every vertex on the other and similar to (A), iterating towards the collision point and C) the method proposed in this project. There are two version of the method, one assumes no rolling and projection of vertices of one body on the other along the lines of velocity (or relative velocity) ensures a single step solution and the developed algorithm returns "time to collision" and the collision point. The method is of $(0)n^2$ complexity, however the number of processes n in the complexity is

determined after substantial data reduction as this is explained in the theory chapter. The second version has conditional iteration added to the algorithm to deal with rolling as well as translation of bodies. If angular rotation is small, the combined motion of translation and rotation of individual points may be assumed to be linear. The assumption is true only and only if the angle of rotation is less than (10^0), this is taken to be the angle which sin(theta) can be approximated to (theta). However if the predicted time implies an angular rotation is more than this than the motion is halved and collision prediction algorithm re runs. This is different than iteratively moving bodies and testing for possible collision.

The following results are found for 70000 vertices (50k on one body and 20k on the other), using the same bone segments shown in Fig. 4.5. Note that such a large number of vertices are not necessary and one could perform the surface collision and contact analysis with much less vertices. Our data base containing most of human body bones and surfaces has two level vertex densities. High density at articulating surfaces, but low density anywhere else (i.e. the bone shaft). Two levels of density reduce the total vertex number from 70000 to few thousand only. For the comparison study, a large number of vertices were chosen to ensure that a measurable execution time can be recorded. A 10 fold reduction in the surface data would result in 100 fold reduction in the execution time. The recorded results for 3 methods run on a Dell T7600 2.33GHz computer (there are many other factors that also influence the performance readings) are as follows:

A. Exhaustive search took 21 minutes.

B. The KernelCAD software took 70 second for a single step nearest distance calculation, it is estimated (by the authoring company, DInsight) the software to be converted to work as a collision tool with e-4 accuracy (although one would not need this) would require approximately 10 iterations. This translates to approximately 11 minutes, half the time of the exhaustive search. Their results were plotted with assumptions of 5 iterations; the total time for 5 iterations is calculated based on the time recorded for a single iteration. The software company promised to implement full collision during the next few months. They further promised to implement our algorithm in their library as an option. If this is done then one could perform a reliable comparison.

C. Predictive search proposed in this thesis took 45 seconds

The chosen examples had not required any iteration during the translation with rolling combined motion. The efficiency and algorithm performance as a time to execution is plotted against the number of vertices in the search. The results show that there is some overhead which influences the lower number of vertex sets, however as the vertex number increases the overhead influence become a small part of the overall processing time.

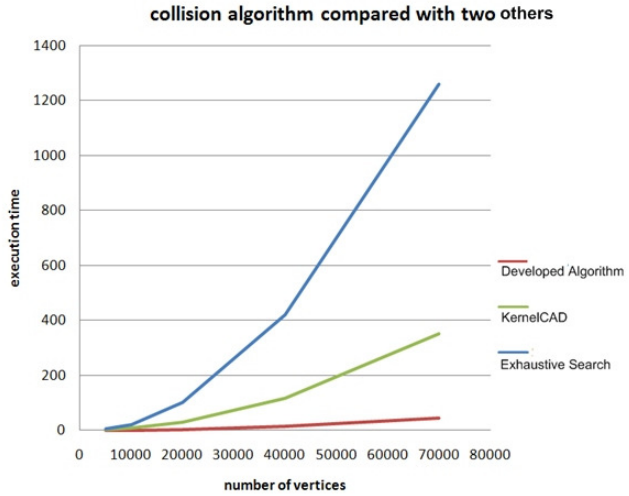


Figure 4.11 Various collision algorithm performances against each others.

4.6 General software features

4.6.1 Screen shots from the program

4.6.1.1 Interactive options

It is not possible to create data for the software to run without having interactive means of doing this. Especially certain data, such as muscle and ligament attachments to bones necessitate these coordinates to be located interactively. The interactive features of the program are shown below. There are 7 main menu headings and each has its sub menus. Some are there to interrogate the object, some for adding new objects (only

standard ones as mentioned above, the muscle geometry creation is automatic and program generates them as it is needed), some simply moves and rotates individual selected objects, Boolean operations implemented to test implants insertions and surgical procedures. "Boolean operations" also allows individual object to be "cut" again for the future considerations of implementing surgical procedures, "calculate inertias" perform volumetric calculations of mass, inertia matrix and cog of the selected object and the program ask the user to decide if these to be saved in the data base, "show", turns certain display options on or off. One of the most important options is the "Muscle_Ligament_Cartilage" option. This simply allows the coordinates of the tissue attachment points to be selected. This option allows either an existing record to be updated or a new record to be created. Finally "Add force" option allows driving forces to be created interactively. This option selects the point, at which the force is acting and the direction of the force. The direction is calculated either by selecting another point on another object or by manually entering the force direction. Interactive options only create part of a record; the rest of the record is created from the default values. Current interactive options and their sub menus are displayed in the Figure 4.12.

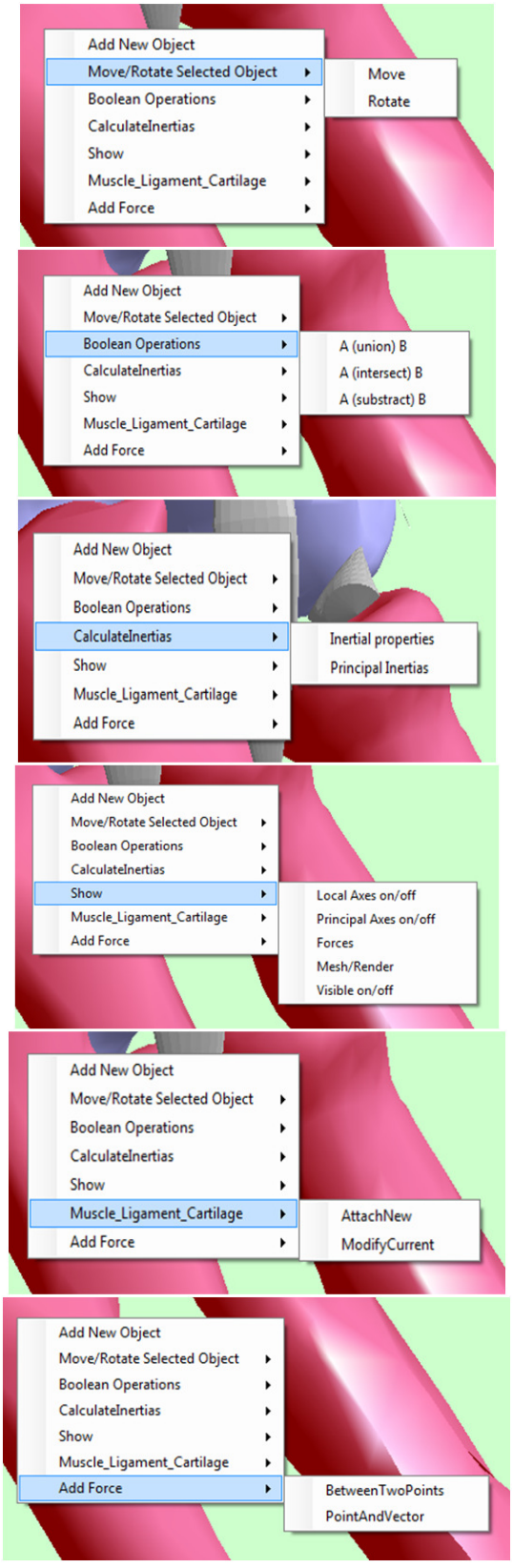
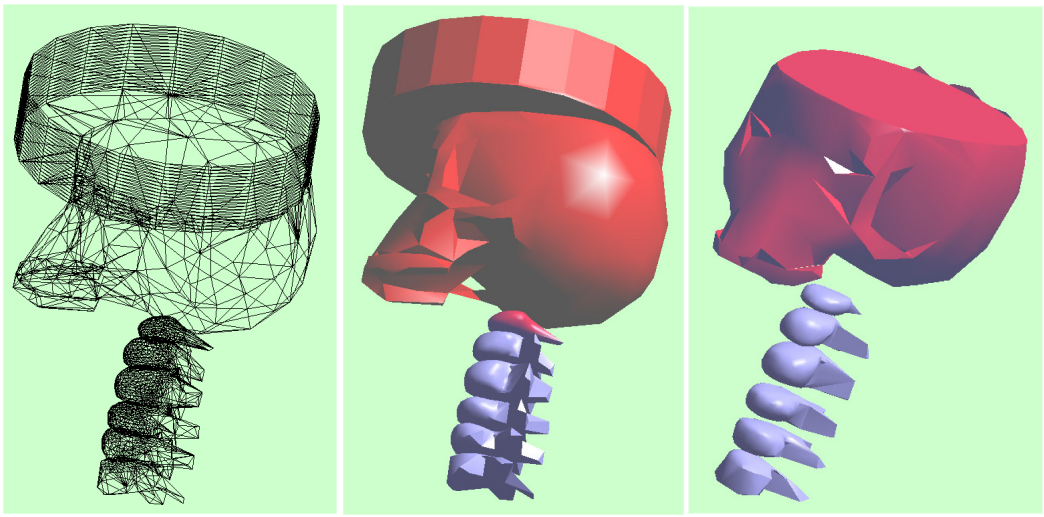


Figure 4.12 Various interactive options, the selections are self explanatory

4.6.1.2 Example for the “difference” or “subtract” Boolean operation

The example in the Figure 4.13 shows the creation of a cylinder which uses the mesh generation scheme described previously, displays it as the wire frame, then renders it and then performs a Boolean operation, in this case subtracting the cylinder from the head.



(a) (b) (c)

Figure 4.13 Neck and head (a) Meshed (b) Rendered and selected with an added cylinder (c) Head–cylinder Boolean operation

4.6.1.3 Comments about software implementation

The software implementation is an ongoing process. At the time when this thesis was written, the implementation was still continuing but reached to a stage where the ideas fundamental to the thesis were ready to be demonstrated. When documenting the implementation of such large software, it is impossible to include every consideration that went into developing the code. It is inevitable that some code were left out for the sake of simplicity. The details of Eigen solvers, numerical integration routines are also excluded as they are standard and available in the literature.

CHAPTER 5

APPLICATIONS, RESULTS AND VALIDATION

5.1 Introduction

In the theory chapter, number of new formulations is proposed. Especially the linear stiffness definition of multi bone structure which is interconnected by biological tissues such as tendon, ligament and cartilage and reduction of the stiffness tensor to investigate its invariant properties is novel. The literature survey had not revealed similar treatment either in the biomechanics or general engineering. Equally the developed wrapping algorithm is the only one on offer for general meshed surfaces. The algorithmic solution of the non-linearity (materially and geometrically), full 6DOF multi body model that is maintained by the contact forces were also proposed. Also a collision detection scheme was developed to deal with the joint interactions. In this section, the evaluation of the accuracy, usefulness, effectiveness and significance of these formulations are addressed. Unfortunately there is no easy way of validating these. Firstly because in case of the joint stiffness and centre or stiffness matrices, there is no similar treatment available, even if there was, the only way of comparing what is offered with anything else would require identical data to be used by the methods. The same argument applies for the muscle wrapping algorithm. Only way of comparing two muscle wrapping algorithms is to test them on identical surface geometries, muscle geometries and muscle attachment points. These prerequisites make the comparative study very difficult. Also there is underlying assumption that the program implementation of all these algorithms are done correctly and program contains to bugs, numerical stability is resolved and the user interface is reliable enough to create the test data accurately. All these aside, like many commercially available human motion modelling software platforms, the reliability of the proposed platform can be tested at least to some satisfactory level. And like many modelling software platforms, if the main functionalities are shown to be stable and within the range of expected behaviour, it can be tuned and improved over the time. Improvement over the time is applied specially in biological tissue modelling since their constitutive modelling of which are still in a very early stage. Therefore the basis of comparative study would be evaluating

the software for establishing if its results are within the "expected range". Therefore the questions to be studied are:

a) Is the software stable and reliable in executing the main functions and various enabling functions?
b) Can individual algorithms be tested in isolation?
c) Can stiffness, mass, damping matrix be assembled and tested for the stability.
d) Is the non-linear solution, executing muscle wrapping, numerically stable?

If it is established that the software implementation of the proposed theories done correctly and that the system stable in every way:

a) How can it be assessed that the obtained solutions are meaningful and useful to the orthopaedics field?
b) Even if obtained results are not identical, are the results within the expected range in terms of previously published material?
c) Is it comparable to the experimental results obtained in the Brunel Orthopaedic Research and Learning Centre?
d) Can it be compared against the Lifemod software, one of the leading software in industry, at least for certain types of analysis?

5.2 Generation of musculoskeletal model and data in MJM

5.2.1 Bone mesh generation

In order to create the proposed 6DOF joint models, surface geometries of bones (segments) are needed. The morphologic study of cadaveric specimens has been performed and bones have been digitized using FARO Platinum Arm by previous researchers in the Brunel Orthopaedic Research and Learning Centre. The output data from the digitization is obtained as point clouds with imperfect surface conditions in the IGES file format. Thus, in order to generate meshed surface data, the Geomagic Studio 9 has been utilized (GEOMAGIC). The procedure starts with importing the .iges files into the Geomagic Studio 9 (Figure A6.1). The point cloud of raw data is usually obtained in different conditions and each bone can possess more than thirty thousand points on its surface. The most of the disconnected points are eliminated through

following the Select Disconnected, Select Outliers and Reduce Noise functions which enable to delete unwanted points outside of the bone surface (Figure A6.2). Then the wrapping function is used in order to create the triangulated surface with optional point spacing. The automatic triangulation also depends on the point cloud where smoother surface parts are obtained with denser point clouds. After obtaining the triangulated surface, possible holes are generated on the surfaces, which are results of un-digitised surface areas. The Fill Hole function is used to fill current holes on the bone surface (Figure A6.3). For the simple shaped holes the basic Fill Hole function is appropriate but if the surface is aggressive or the simple Fill Hole option is not sufficient, then alternative options can be followed along with Fill Partial, Create Bridge, and Clean Up and Move functions. Additionally, undesirably wrapped or triangulated surface parts can be revealed. In this case triangles can be deleted manually and then the created holes can be filled with the advanced Fill Hole options. After deleting triangles and using the advanced Fill Hole functions such as the Create Bridge option, the constructed surface might possess straight line bridges overall the surface (Figure A6.4). These imperfections can easily be smoothed through Sandpaper function. The Sandpaper function enables to relax or clean imperfect regions of the surface through adjusting the desired strength or smoothness level and selecting the boundary fixing option. If a surface is wavy and desired to be smoothed, utilizing this function is convenient which can provide balanced surface area. The Sandpaper option is also utilized with prosthetic applications where the surface is cleaned to reduce the unwanted surface area. This application can mimic the anatomically removed bone parts to place the prosthesis. Therefore, Polygonal options can be performed in order to obtain the desired smooth surface with reconstructing triangles or moving surface points. For example, Relax option is performed to relax the aggressive regions and to obtain smooth surface. After obtaining the desired surface with no defects, the meshed data can be converted in several file formats to be used in many modelling, design and other programs. Commonly used file formats are .stl, .wrl, .obj. These formats are used effectively if the triangulated surface format or similar surface features are fine enough to represent a body surface. However for some software packages, the triangulated mesh data should be post processed and patches must be created. The advantage in this type of post processing is creating the patches and splines which provide control points through the surface. For example the control points and spline curves are very important features in geometric based deformation simulations (Sec. 2.2.4.2). NURBS are the commonly

used surface data for geometric based deformations and for the FEA packages (i.e. ABAQUS). In considering the current application in this thesis, in order to import the mesh data into the developed prototype software, VRML 2.0 file format and triangulated surface data is sufficient. Nevertheless, number of triangles and mesh density influence the processing time. In order to tackle this problem, the density of the triangulated surface is reduced through reducing the number of triangles with the Decimate Polygons function. The similar example is also seen in Lifemod software where the bone surface geometries are highly simplified for the efficiency. However, grosser the mesh less accurate the analysis, especially if the analysis depends on the surface smoothness. In Lifemod software, geometric based analysis is performed through parasolid and Shell (.shl) geometries or with similar formats which are detailed surface file formats. In the particular application of surface geometry based prosthetic joint analysis (TKR), the parasolid surface geometry is used which can be described as a semi analytic surface. As the parasolid based surface representation for each bone segment requires more than several Mb storage space, the geometric based analysis seems difficult or inapplicable at this stage. In order to perform the 6DOF joint modelling in the developed prototype software, accurately represented joint contact area is essential. In addition to this, in the proposed modelling application, it is important to have dense surface to perform surface geometry based tissue wrapping. Thus, the triangle reduction on bone surfaces has been carefully performed with leaving contact geometries as smooth as possible. In Appendix A6 the mesh generation processing through the Geomagic Studio 9 is illustrated. The radius bone of elbow joint complex has been chosen to show the processing and performed functions. As seen from the figures in Appendix A6, after completing the required applications, original 75000 triangles on radial bone surface has been reduced into 7000 triangles without a significant (or visible) reduction in smoothness.

Above procedures and many others have been followed for ulna, humerus, radius, scapula and clavicle bones. In addition to these particular upper limb bones, hand as one segment has been exported from the Lifemod and imported into the Geomagic Studio 9 in order to assemble it with the prepared upper limb extremity for elbow joint analysis. In this thesis the focus is to analyse the elbow joint as well as head and cervical vertebrae joints (neck). The head and neck bones have been adapted from Lifemod software where bone surface geometries are highly simplified. Hence, head

and neck bones have been imported into the Geomagic Studio 9 and triangulation of bone surfaces has been increased with the Refine Polygons function (Appendix A6). Additionally, upper body bone data which includes the thoracic vertebra T1 that involves in the head and neck whiplash simulation has also been imported from Lifemod. Through the proposed 6DOF and surface geometry based joint analysis, bone segments of elbow joint complex as well as bone segments of all cervical vertebra joints have been created as unconstrained and separate bodies.

The required geometric data for elbow joint analysis and head and neck whiplash test has been processed using the Geomagic Studio 9 and then data has been imported into the developed software in VRML 2.0 (.wrl) file format. The VRML (Virtual Reality Modelling Language) is the standard file format and it is mainly used for 3D vector graphics. Up to date several versions of VRML have been developed as VRML 1.0, VRML 2.0, VRML97 and the latest version is the XML based X3D which is used for real time 3D computer graphics. With the VRML file format, 3D objects can be read in a simple text format or it is compressed in unreadable "worlds" (.wrl) format. VRML is an object-oriented language, where the objects are defined as nodes, edges, triangles or vertices in 3D scene.

Further modelling applications such as the dynamic analysis, the tissue wrapping, the tissue deformation, the collision and the contact have been performed with the developed software and applications are mainly addressed in the following sections.

5.2.2 Skeletal system and joint motion terminology

In the developed joint modelling formulation, it has been emphasized that the joint kinematics is unconstrained and each joint possesses 6DOF mobility. As the main purpose is to analyse the joint articulations, based on surface geometry, providing dense geometric data at the contact surfaces are essential. As mentioned in the earlier Section 2.1.1, human skeletal system is composed of synarthroses (immovable), amphiarthroses (slightly movable) and diarthroses or diarthroidal (movable) joints. Through this consideration, each type of joint is described relative to commonly used idealised kinematic mobility descriptions. For example amphiarthroses (slightly movable) joints

have been assumed to possess less DOF than diarthroidal (movable) joints. However, in this thesis, the intention is to emphasize that each joint even it is assumed as slightly movable joint, it is considered as a 6DOF joint. The degree of joint flexibility is dictated by the ligaments and the tendons. However modelling of the precise joint geometries and creating all interactive tissues is a very time consuming process thus the analysis will be limited to two types of joints. In this thesis, the cervical vertebrae joints (slightly movable) and elbow joint (movable) are selected, in order to provide the joint analysis for the two main joint types. The human skeletal system with bone and joint names are provided in Appendix A1. For the specialized focus on head-neck and elbow joints, following joint movement descriptions and modelling procedures have been performed. For example the mobility of vertebral column including cervical vertebrae, thoracic vertebrae and lumbar vertebrae is commonly described by flexion-extension, lateral bending and axial rotations performed at each vertebral joint. In general each type of movement is described in 1DOF. Additionally, each type of movement is usually assumed relative to fixed axes or centre of rotations. In Appendix A2, common cervical vertebrae joint movements are shown.

In considering the elbow joint, the joint movements are described by flexion-extension, varus-valgus and supination-pronation movements. In Appendix A3, these common elbow joint movements are shown. Range of movements for each cervical vertebrae joints and particular varus-valgus range of movement during neutral flexion of elbow joint are provided in the Appendix A2, A3, compiled from published literature. However, due to the nature of the joint modelling provided in this thesis, each joint range of motion is described in three dimensional translations and three dimensional rotations in R^3 described in terms of the Euler angles or the direction cosines, convertible from one to other as given in the Appendix A4. It should be noted that the common joint movements are described based on 1DOF, and they are not comparable with the proposed 6DOF joint movements with the time dependent changing of axis (centre) of rotations. In the proposed applications, precise data for the head-neck whiplash test has been obtained from the created model in Lifemod software. In Lifemod software, 6DOF mobility is described for cervical vertebrae discs which are cartilage like soft tissues between each cervical vertebra.

5.2.3 Musculoskeletal system and tissue modelling

In the developed joint modelling applications, musculoskeletal tissue modelling is classified as linear viscoelastic which is Kelvin-Voigt model and nonlinear spring and dashpot element based viscoelastic Kelvin-Voigt like spring and spring-dashpot arrangement. This modelling approach is utilized to describe passive tissue behaviours of all musculoskeletal tissues (ligament, muscle, cartilage, and disk). In ligament tissue modelling, only tensional forces are generated and tissue path is created through applying the developed tissue path and wrapping algorithms (Sec. 3.3). Therefore, ligament is defined as a tissue between origin and insertion points along it`s shortest paths. Each ligament response is described in three dimensional stiffness and three dimensional damping matrices in linear material modelling applications. The Geometric representation of ligaments is generated as three dimensional geometric shapes where the line of action is passing through the centroid points (geometric centre) of these shapes. These geometric shapes are created using the mesh generation algorithms described in the software chapter (Chapter 4). For active tissues which are skeletal muscles, the same procedure is applied. Currently four types of geometries are utilised, a cosine function describing muscles, ellipsoid describing ligament, a uniform cylindrical radius with conical ends and tangent to surface and finally a short cylinder representing cartilage contact with aligning its central axis with the average surface normal of contacting surfaces. The line of action for all types of tissues passes through their geometric centre. The shortest path and tissue wrapping algorithms are performed to define each tissue path lying on triangulated bone and other tissue surfaces. In non linear analysis, tissue dimensions are created dynamically. For cartilage tissue, contact analysis is performed. Cartilage tissue is also described as viscoelastic Kelvin-Voigt model based spring and dashpot elements lying on contact surfaces between articulating bones but with an exponential stiffness function. Through the implementation, collision detection algorithm is performed to detect the colliding triangles and points which yield to activate the contact springs to response in order to avoid the penetration. After performing tissue attachments for all musculoskeletal tissues, static equilibrium position is obtained and dynamic or static analysis is then performed after taking the equilibrium position as the reference position. Specifically head and neck (cervical vertebrae) joints and elbow joint are analysed in this thesis, which is found appropriate to introduce all ligaments, muscles, cartilage and disks in Appendix A2 and A3.

5.2.3.1 Head and neck tissues

For the head and neck whiplash analysis, four types of ligaments as interspinous ligaments, flaval ligaments, facet joint capsule ligaments and longitudional ligaments have been created from head through cervical vertebrae (C1-C7) to thoracic vertebra (T1) (Figure A1.1). For interspinous ligaments, flaval ligaments and longitudional ligaments, eight individual tissues have been generated. Furthermore, five types of muscles have been created as semispinalis cervicis, trapezius, longus colli, semispinalis capitis and sternocleidomastoid muscles. Then, a cartilage (disk) between each vertebra has been created. The number of total ligament, muscle and cartilage is the same with the model which is created in Lifemod software for the whiplash test in order to compare the results.

In order to reduce the time and avoid data creation errors, the "Whiplash Necksim" example provided by the Lifemod is used with all its soft tissue and muscle parameters as well as its dynamic impact force loading. The model is created from the developed modelling example. In the MJM, all tissue attachments were done visually by inspecting the Lifemod model visually. These approximate positions make each model slightly different.

5.2.3.2 Elbow tissues

For the elbow joint analysis, same procedures have been followed. The main elbow joint ligament types are radial collateral ligament (RCL), lateral ulnar collateral ligament (LUCL), annular ligament (AL) and medial collateral ligament (MUCL). Number of individual ligament attachments is different for each ligament type such as medial ulnar collateral ligament is modelled with anterior, posterior and transverse bundles. Ligament positions have been adapted from the text book (Tortora & Derrickson, 2009) as Lifemod software does not provide any information about the elbow joint ligament attachments and ligament tissue parameters. Additionally, elbow muscles are modelled as biceps, brachialis, brachioradialis, triceps, pronator teres and extensor carpi radialis longus muscles. Again in the developed joint modelling software the number of individual muscle part depends on the muscle dimensions and anatomic attachments. Furthermore, cartilage on the humeral bone surface and cartilage on the radial and ulnar bone surfaces have been created for elbow joint. The screen snap shot of elbow joint cartilages from the developed modelling software is shown in the

Appendix A3. Detailed information about the attachment points of ligaments and muscles and place of cartilage tissue or disks for cervical vertebrae (neck) joints and elbow joint are given in the Appendix A2 and A3. Literature base stiffness and damping properties of ligaments, muscles, cartilage and cervical vertebra disks are provided in the Appendix A2 and A3. Additionally, literature based muscle moment arm estimations are provided in the Appendix A3 for elbow joint (arm) only.

5.2.4 Some functionalities of the developed software MJM

The software was developed to test the theories proposed in the theory chapter. However the software was developed well beyond an academic tool. Partly because, soon after the development started, it became obvious that the problem was "data rich" and unless it was driven by a database system it would have been impossible to carry out extensive tests that required for the development work. The second reason was the fact that one of the original objectives was to create a platform which can be used for virtual surgery and this could not be accomplished on a software without a powerful graphical user interface. Finally there were so many enabling routines that unless these were collected on a well structured software system, the main focus of the research could not be achieved. For example some of these functions implemented are: simultaneous equation solver, matrix inversion, Eigen solver, mass and inertia calculator, numerical integration routines, mesh generators for tissues, Boolean operators (although a library function was used). These were either developed from scratch or modified from existing Fortran libraries such as Lapack (LAPACK).

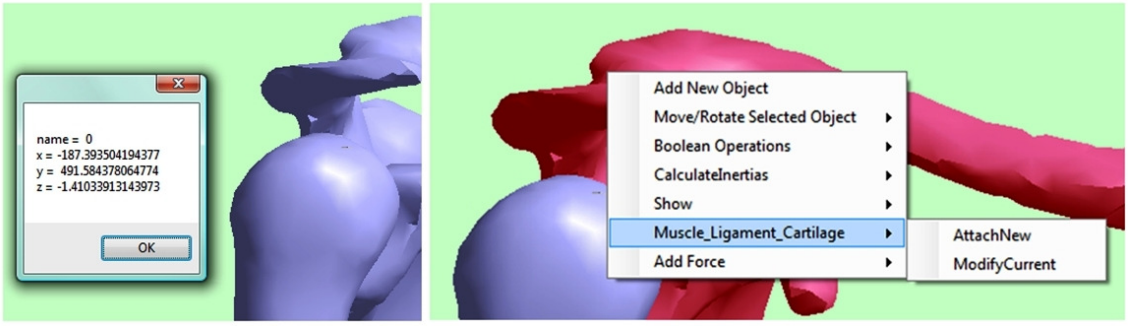
Current MJM analysis capabilities include, solving for linear deflection to calculate joint laxity, solving non-linear static deflection (only material non-linearity), calculating centre of rotation, axis of stiffnesses, non-linear simulation with tissue wrapping.

The data entry to the software is fully interactive (there is no other easy way of doing this) and stored in database for the future use. The graphical user interface is user friendly and easy to use.

5.2.4.1 A data creation session on MJM

The software has tens of functionalities and without a "user manual" type document it would be impossible to explain all the functions, here only a few selected functions will be given. There are three types of interaction with graphics screen. First type involves moving the objects in the screen, rotating the object in the screen using the left mouse key and translating them by using the left mouse key with the shift key. The second type involves selecting a single object. For example calculating mass/inertia properties involves picking single object from the screen. For this selection, the object is selected from the screen by holding the "ctrl" key and clicking on it using the left mouse key. Selection changes the colour of the object and also displays the selected coordinates of the point on the object in the global coordinates. The third type of selection involves two points. In this case, the program prompts the user to select the second object. Once the second object is selected the program asks the user to decide if the data to be stored. For example if a ligament is selected by this method, the program will automatically write the data into the text boxes on the screen. These text boxes are mapped to the data base but the saving is not automatic and the user may make changes before saving, such as modifying the tissue type, where the default is the muscle type. And if anything else to be modified the user needs to make changes to the appropriate fields before saving them in the data base. The following steps show how a selection is performed for tissue attachment.

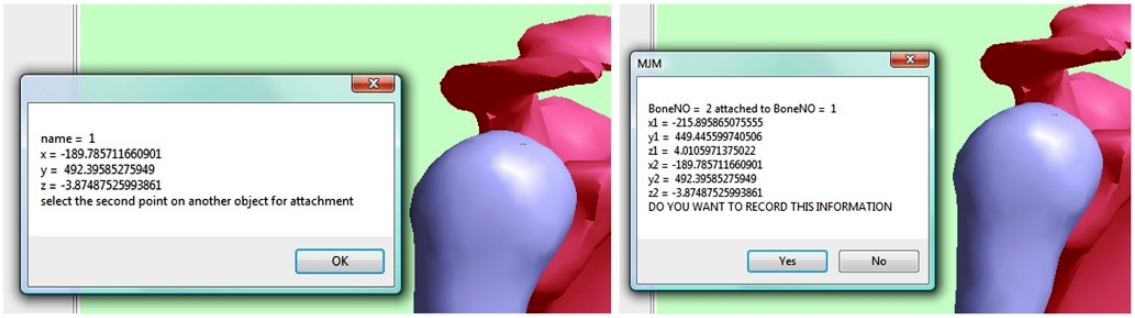
Figure 5.1(a), shows the selection process starting by user clicking an object. The clicking displays the selected point coordinates in the global axes, this happens only if the cursor is on an object. In Figure 5.1 (b), the object colour changed and all options for manipulating object or creating object related information options are displayed. For this session, the "Muscle_Ligament_Cartilage" option is selected, giving further options for either creating a new tissue attachment record or modifying the currently selected record data. For the "ModifyCurrent" option, the intended connection record must be currently selected.



(a) (b)

Figure 5.1 Step1of tissue attachment processes, (a) Initial selection (b) Identifying the selection with changing the object colour

Once the new record option is selected the message window repeats the selected point coordinates and asks user to select the second point, shown in Figure.5.2 (a). When the second object is selected the second object coordinates displayed with the first object coordinates and the user is invited to decide if this data to be placed on the text boxes on data pane of the screen. Clicking yes creates pointers to the appropriate data table and creates relevant text boxes and places data in these boxes shown in Figure 5.3.



(a) (b)

Figure 5.2 Step 2 of tissue attachment processes, (a) User is asked to enter the second point (b) The selection of the second point

At this stage data is not recorded, for this "Add" button needs to be clicked. Before recording the data, mount type 1, for muscle, 2 for ligament and 3 for cartilage can be selected. Also Mounting name (in this case ligament name) needs to be modified. These text boxes are displayed in the Figure 5.3.

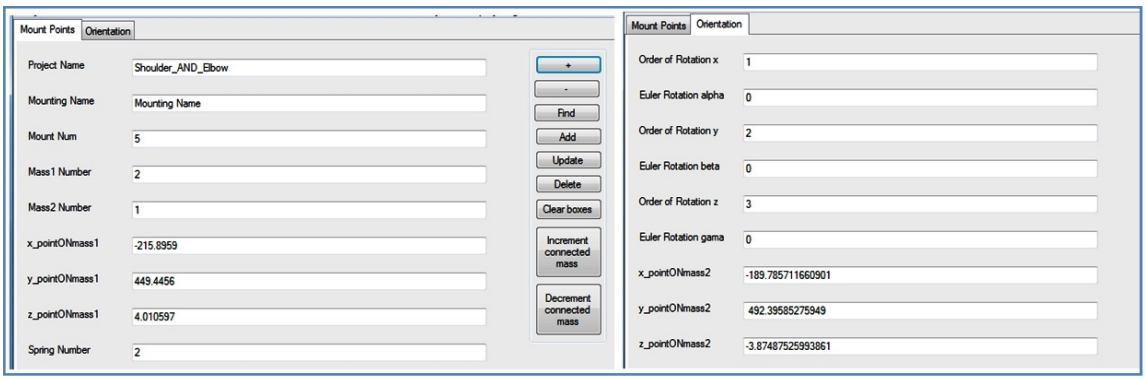


Figure 5.3 Recording the attached mount (spring) between mass (bone) 2 and mass 1

Note that mass 1 is the mass number 2 and the mass 2 is mass number 1. The connection is always from the higher number to the lower number. For example if mass 4 and mass 2 are connected by a mount then the mass number 1 is mass 4 and mass number 2 is mass 2.

Clicking the "add" button, the ligament tissue is created; in this case the tissue is relatively "flat ellipsoid" in cross section. The selection of four types of geometries (cylinder, cylinder with conical ends, flat ellipsoid, translated cosine and describing various ligament types with these shapes, at the time of writing the thesis, are experimental and likely to change. Currently the tissue shapes do not relate to the force carried by the tissue. It is expected that in the future developments this may be modified to relate the load carried by the tissue. The tissue (flat ellipsoidal cross section for ligament) is shown in Figure 5.4 and it is noted that that the tissue performs no muscle wrapping (possibly cutting the bone) at this stage. Only geometrically-non-linear analysis performs the muscle wrapping.

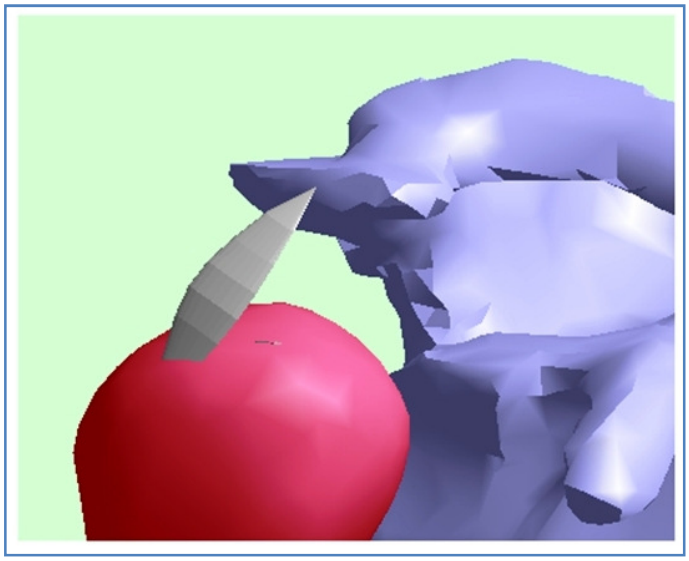


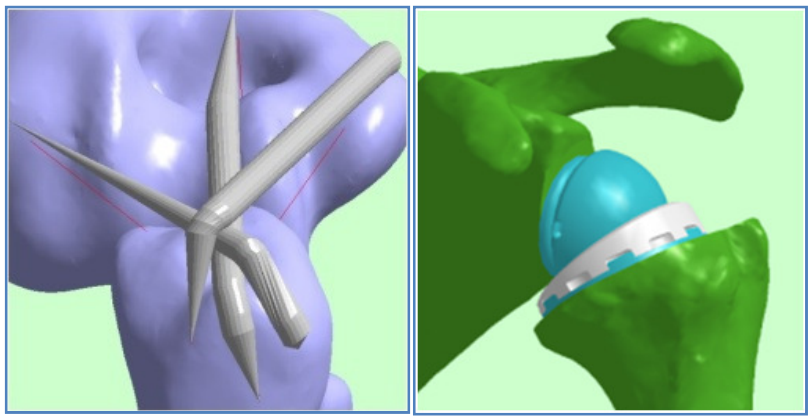
Figure 5.4 Tissue attachments between two bones (masses)

5.2.4.2 Boolean operations on MJM

The software allows all three Boolean operations and a specialised version of the subtraction (Union, Intersection, and subtraction and specialised half plane subtraction). With the half plane subtraction, part of bone material intersecting with the half plane (defined by a point and a normal vector), and can be removed. The Boolean options are implemented in order to create the artificial joints by cutting bone and inserting implants enabling the software to analyse artificial joints. A simple Boolean operation is given in the software implementation chapter and an example here is given in Figure 5.5 (b) where implants are inserted at the humeral head and glenoid cavity. The positioning of the implant is performed by moving and rotating the implant. Implanted joint analysis will not be given in this thesis due to time restriction and also for not having any digitised elbow or neck implants (only two joints that are analysed for validation purpose). The only digital implant presented here is the "Verso" shoulder implant which has been developed by Levy and BIOMET Company (BIOMET) as seen in Figure 5.5 (b).

5.2.4.3 Tissue wrapping

Certain degree of tissue wrapping takes place in all joints however almost all the available literature relates to the shoulder. In the implementation presented here the elbow joint is used to represent any generic joint to demonstrate the algorithm proposed earlier. The Figure 5.5 (a) demonstrates the tissue wrapping. Three tissues are selected in a particular order, the first tissue entry following the bone surfaces between two attachment points and the second and the third are wrapping on the bone surface and the previously entered tissue(s). The red lines are straight lines between the attachment points.



(a) (b)

Figure 5.5 Examples of MJM functions, (a) Multi tissue wrapping, (b) Boolean operation and inserted implant

5.3 Lifemod software

Lifemod software (LIFEMOD) is a multi body dynamics based musculoskeletal modelling software built up on the Adams dynamic engine (Sec. 2.1.5). In order to describe the capabilities of the Lifemod software, the body segment creation, joint modelling, soft tissue modelling, contact modelling, finally, inverse and forward dynamic and static analysis are presented in this section. First of all, in order to perform a body or joint analysis, body segments (bones) are created. Lifemod software contains coarse anthropometric human skeletal database. After selecting a desired body characteristics (i.e. gender, age, and weight), parameters such as mass, coordinates and principal inertias are created automatically. Usually the body segments are treated as

assembled bone complexes. For example, radius and ulna are assumed as parts of the forearm and inertial calculation is performed for the forearm through considering ulna and radius together. However the software also allows bone segments to be defined individually. On the other hand, bone geometries are highly simplified and it is not possible to perform geometry based joint analysis through employing such geometries. Nevertheless, in Lifemod software, geometric based analysis can be performed through importing surface data such as the parasolid geometries. The global coordinate system is taken at the middle of the pelvic bones. The local coordinate system of each segment is probably taken as the principle axes system of the segments.

After creating body segments, joints are modelled. Before modelling the joints, articulation cannot be performed. This basically explains that through describing a joint, kinematics with the coordinate system is created. All created joints are kinematically idealized with fixed axes or centre of rotations. In the joint modelling the maximum mobility is 3DOF spherical joint. There are several joint mobility types as free, passive, fixed, Hybrid III dependent, driven and driver. In passive and Hybrid III dependent joints, angular joint stiffness parameters are introduced into the system which defines spring and dashpot dependent stiffness limits of the joint. In the fixed joints, kinematics is totally fixed to the ground and such a joint possesses no mobility. In the free joints there is no kinematic joint constraint in three rotational axes. If the joint is free, it means that the joint has all 3 degrees of freedom and unconstrained, “free” does not mean a 6DOF. Furthermore, driven joints are modelled based on the prescribed kinematic mobility which is introduced through the spread sheet tables. The driver type joints are modelled based on the joint axis controller. The Lifemod software also describes the joint stiffness. This is fundamentally different definition than the joint stiffness described in this thesis. In the Lifemod software, constant stiffness parameters are used to describe the stiffness of a joint in three angular directions and has no relationship to the tissue stiffnesses. Also the Lifemod software cannot calculate tissue loading accurately since the joint is constrained the tissue loading is indeterminate. After creating a joint, stiffness values remain constant where the analysis is performed with these parameters. In the developed formulations in this thesis the joint stiffness concept is based on the joint conditions such as ligament stiffness and muscle forces as well as joint geometry. This is not the case in Lifemod software where the stiffness parameters are mainly adapted from literature and from dummy crash test based

measurements. The dummy crash test based parameters entirely depend on the unrealistic body segment as ellipsoids. This kind of analysis can be performed for simplified joint mechanisms however it does not provide much in joint stiffness and response based on realistic geometries. In addition to the 3D joint modelling definition, there is an additional option in Lifemod software which allows creating contact geometry based joints. As it is demonstrated in the total knee replacement (TKR) tutorial in Lifemod, the contact joint kinematics is driven by contact constrained joint definition. In the contact modelling, there are three types. The first is the ellipsoid-plane contact which represents contact between ellipsoidal dummy and the ground. The second is the contact between ellipsoidal dummies which is called ellipsoid-ellipsoid contact. The third is the solid-solid contact. The solid-solid contact is originally created between the parasolid surfaces. The solid-solid contact is performed to describe the contact between tibia and femoral implants in the TKR analysis (Figure 5.6). The contact kinematics needs to be known in order to calculate collision response forces. As discussed in Section 2.3.3 unilateral contact formulation is an advanced contact modelling method which depends on the kinematic formulation. Thus the Lifemod software performs this kind of contact formulation for solid-solid contact analysis where the kinematic mobility is reduced when the contact is active.

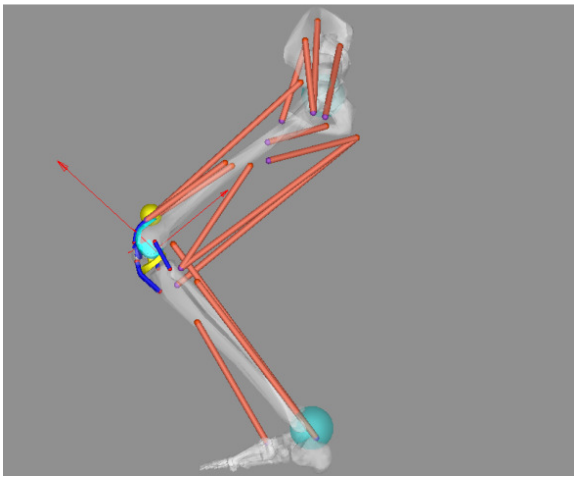


Figure 5.6 Prosthetic knee joint model and tissue wrapping on geometric surface of prosthesis from Lifemod software

As for the soft tissue modelling in Lifemod software, ligaments are described as passive spring and dashpot elements attached between origin and insertion points as a straight line. The straight line attachment is the default tissue path modelling approach in Lifemod software, any "wrapping" is guided manually. For the muscle modelling the Hill's muscle model is employed. The wrapping algorithm is claimed to be performed on the geometric surface which is only possible where the parasolid surface is used (Figure 5.6), everywhere else tissue lines are straight, even when they go through the bone.

After completing the model preparation procedures briefly explained above, the analysis can be followed. Firstly the inverse dynamic analysis of joint and soft tissues is performed to train the body and prepare it for the forward dynamic analysis. After completing the desired dynamic analyses, the model can be saved and results can be depicted. In general, results are generated in three dimensional joint rotations due to spherical joint modelling, and 6DOF body motions relative to each body centre of mass. All these results can be depicted relative to the posture or time and can be exported to the excel files.

5.3.1 Comparing the MJM versus the Lifemod

The Lifemod software has been in industry many years and development which includes its ADAM platform, since its start in 1980s hundreds of man years has gone into its development. Therefore the intention is not to compare capabilities of this software with the MJM which is developed only as a research tool primarily to test various formulations proposed in this thesis. The intention of comparison is to identify fundamental differences between these two softwares in terms of their approach to modelling.

The Lifemod software has been developed to model dynamics of multi rigid bodies interconnected, preferably, by spherical joints. The Lifemod software is built on ADAMS, which is developed to analyse mechanical rigid body systems and mechanisms. The focus of analysis is the motion of bodies. Surface contact driven motion is possible by employing contact constraints (such as tangency of contacting surfaces) on parasolid surfaces dealing with rigid contact. It is not clear if the Lifemod

can handle non-rigid contact. Also it has no capability of manipulating the joint geometry.

MJM software is developed to model human joints described in terms of contacting surface geometries maintained by surrounding tissue, such as ligaments, tendons, muscles and cartilage. Cartilage stiffness is activated during contact. The joint has full 6DOF. The focus of the analysis is the joint behaviour. The joint geometries can be manipulated and artificial joints can be introduced. It allows the joint to be cut, removed and replaced by orthopaedic implant. It can perform joint related analysis such as joint laxity, joint stiffness, calculation of joint stiffness invariants such as the centre of stiffness and principal axes of stiffness.

5.4 Joint modelling, comparative results and validation

5.4.1 Rationale and the basis of validation

Before presenting various case studies in comparing the results obtained by the modelling theories and the software presented in this thesis and the other modellers and the published results and some experiments performed in the Brunel Orthopaedic Research and Learning Centre. It is necessary to establish what may be expected out of the proposed comparison. Normally in a comparative study if one method is compared against the other and the results are not in agreement then it is possible to argue that one is right and the other is not, or of course both may be wrong. A real life study of the same problem may identify the validity of results. The same understanding of validity applies to the study presented here except that since the problem involves living subjects some of the modelling results cannot be tested on a live person. For example, in studying joint laxity, it is not possible to obtain reliable laxity results from live persons. On the other hand cadaveric laxity can be studied but there is no certainty that cadaveric material preserve their natural laxity. Another common problem relates to muscle modelling, activation and load sharing. It is well known that the activation depend on the individual, and the joint loading again depends on the person. In a joint where more than one muscle can share the joint torque, the muscle load share may depend on the people's daily routine or exercise. In modelling the formulation of muscle loading must employ some biological or physical concept such as minimum

energy in order to distribute the load among the joint muscles. But these are just modelling concepts, whether the loading follow any one of these concepts are debatable. The muscle wrapping is also a similarly debatable concept. There is no question that the wrapping does happen, but it is not clear if the path change during the geometry changes as the motion progresses. There are many issues which makes the comparison between modellers and between modellers and the real life difficult. Because of this, all modellers, currently performs inverse motion analysis. Therefore to validate the theory and the software presented in this thesis, only "trend analysis" will be performed. I) Are the results "believable" and similar to other modellers in terms of general behaviour rather than the exact magnitude? II) In case of the presented formulations, some of which are completely novel, are there experimental results which can be used for comparison? Again looking for a general behaviour rather than the exact values. III) There are results produced in the current work which are based on novel concepts and thus no comparative results are available. In that case, are the results meaningful and are there indicators of their validity? The trend based comparison will be based on two case studies, Head and neck motion when the torso is subjected to sudden acceleration and the elbow when articulated for its full range under the various valgus-varus loading in order to study joint laxity as a function of applied load and the joint angle.

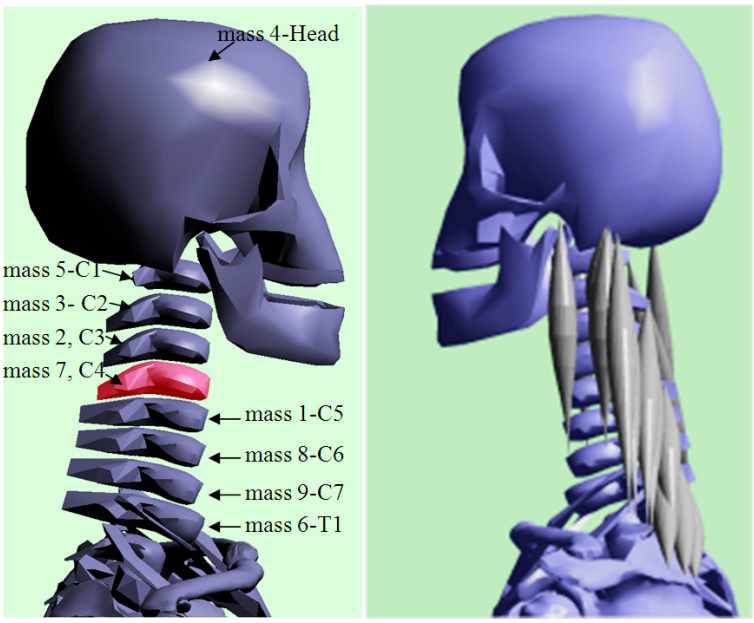
5.4.2 Head and cervical vertebrae (neck) motion modelling

In this section, various issues are addressed such as the motion of the neck and head complex when the torso is subjected to sudden accelerations. This example is chosen because the Lifemod software provides this as a case study. The second reason is the type of data which is used by Lifemod software. The software assumes that the neck segments are connected by stiff pads and this provides the main stiffness of the structure. This is very easy to implement in MJM and the case was taken to be studied for the comparison.

5.4.2.1 Motion modelling based results of head and cervical vertebrae using the MJM software

The modelling involves transferring the bone surface information of the Lifemod software to the MJM. The main reason for doing this is to minimise the variation between two models. Even after doing this, there are many parameters and factors which are not either available in the Lifemod documentation or were unclear. After transferring, the bone segments were positioned together as "natural" as possible. This was a necessary step since bones surfaces in the Lifemod software do not participate in the analysis. In Lifemod, they are for display purpose only in order to make the analysis more meaningful. The MJM makes the use of the surface information for locating the ligament attachment points and also for establishing approximate normals for positioning of the cartilage "pad"s. In the Lifemod analysis it appears that the attachment points are pre-entered with a high degree of symmetry as it will be obvious from their results. In the head and neck mechanism each joint possesses 6DOF mobility which is placed between head and C1 also through C1 and C7 and between C7 and thoracic vertebra (T1). There are a total of nine free bodies (54 DOF) where the reference body is taken as the T1. Two studies were carried out in MJM, in the first study, the T1 prescribed to move with the half sine acceleration and in the second study, the T1 was fixed and a force applied on the head. In the Lifemod example, all cervical vertebrae bones and the head are connected by ligaments, muscles and cartilage (disk). For the sake of the simplicity only four major types of ligaments as interspinaus, flaval, facet joint capsule and anterior longitudional ligaments are modelled. And only five types of muscles as trapezius, semispinalis cervicis, longus colli, semispinalis capitis and sternocleidomastoid muscles are modelled. For the analysis, the linear tissue and linear geometric modelling (small displacement) is assumed. For MJM analysis, the tissue stiffness and damping parameters have been adapted from Lifemod software (Appendix A2). As for the tissue attachment points, these are also adapted from Lifemod software, graphically inspecting and attaching them interactively. Therefore these were selected visually. The attachment points as origin and insertion points for each ligament and muscle are given in Appendix A2 which were used by the Lifemod analysis, it was not possible to use these data points in the MJM because of the

difference (although slight) in the coordinate systems and because the position of bone segments moved in re-positioning them during data preparation. The cartilage tissue has been also modelled based on the linear viscoelastic material model (Kelvin-Voigt). After building the complete neck and the head model, bone segments are packed as close as possible to what appeared to be the "natural" shape. The manipulation of the neck/head geometry extracted from the Lifemod was not possible without using a graphics software, for this, the data was transferred to the Geomagic Studio 9 and after manipulation and adjustment, the data were converted to VRML 2.0 file format and transferred to the MJM. In doing so, the bone numbering has changed and the final bone numbering as well as the final geometry used in the analysis is shown in Figure 5.7 (a). The interactively attached muscles are also shown in the Figure 5.7 (b).



(a) (b)

Figure 5.7 Extracts from MJM, (a) Numbering of head and neck masses (bones) (b) Some muscle attachments

Another major divergence from the Lifemod data is the scaling all mass, stiffness, damping and force values. All of them were divided by 2, the reason being that the largest moving mass (the head) calculated by the MJM inetial calculation module gave

half of the mass used by the Lifemod software. The inertial values of the Lifemod software are not based on the meshed objects. Therefore 5.068615 kg for the head is almost exactly twice of what was calculated by the volume integration carried out by the MJM (2.59kg, taken from MJM database). If the mass is multiplied to be the half, then stiffness, damping and the force values had to be halved. Therefore 20kN/m cartilage stiffness used by the Lifemod was also halved. This is mathematically creates an equivalent system since every term of the equation of motion is divided by the same factor.

5.4.2.1.1 Half Sine acceleration acting on T1

A half sine shock applied on T1, in other word the T1 motion was constrained to move with an acceleration in –x direction. This is equivalent of car crashing into an obstacle (and what was used by the Lifemod analysis). This shock is equivalent to the example given in the Lifemod software but with half of the maximum amplitude for the reasons explained above. The result will be compared later in a greater detail but at this stage there are observations which give some validity to the results. Head having the largest movement is what was expected, the main movement is in the x direction and all 9 bodies move with a linear velocity following the initial acceleration. The results are plotted by the MJM and displayed in Figure 5.8 (a) and 5.8 (b).

Figure 5.8 (a) Half Sine acceleration constraint shock acting on T1, the graph shows mass no 4, head movement

The velocity of the motion for the head (mass 4) is plotted in Figure 5.8 (b). The motion decays down to a constant velocity (approximately-7000mm/s) in the x direction. Also motion in y direction indicates coupling (although very small) non existent in the symmetric Lifemod results.

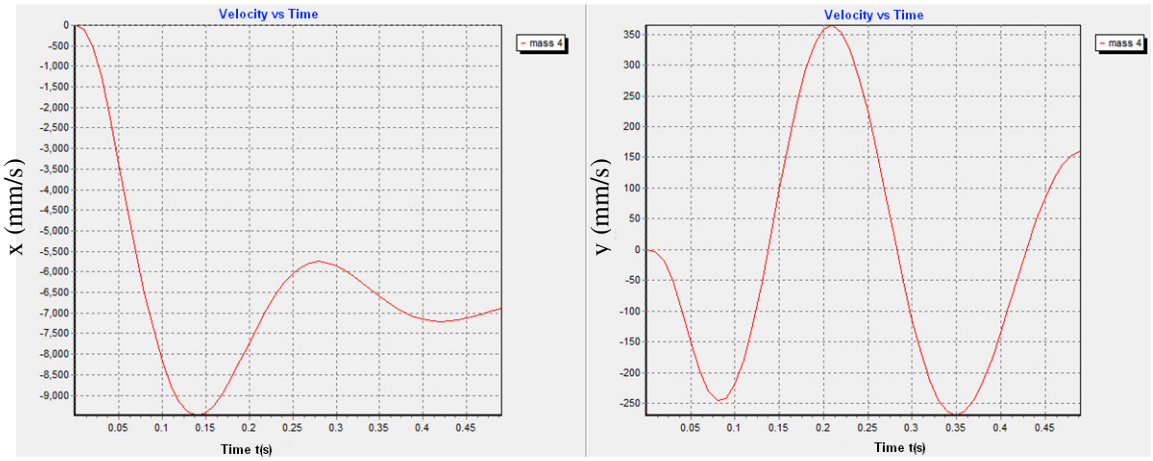


Figure 5.8 (b) Velocity versus time response of head (mass 4) to half Sine shock. x and y directions are plotted, note the motion in the y direction

5.4.2.1.2 Half Sine force acting on head (T1is fixed)

The results with a force acting on the head, magnitude of which is calculated based on the (maximum acceleration/2) multiplied with the head mass is plotted in the Figure 5.9. The head movement for this shock is displayed in Figure 5.10.

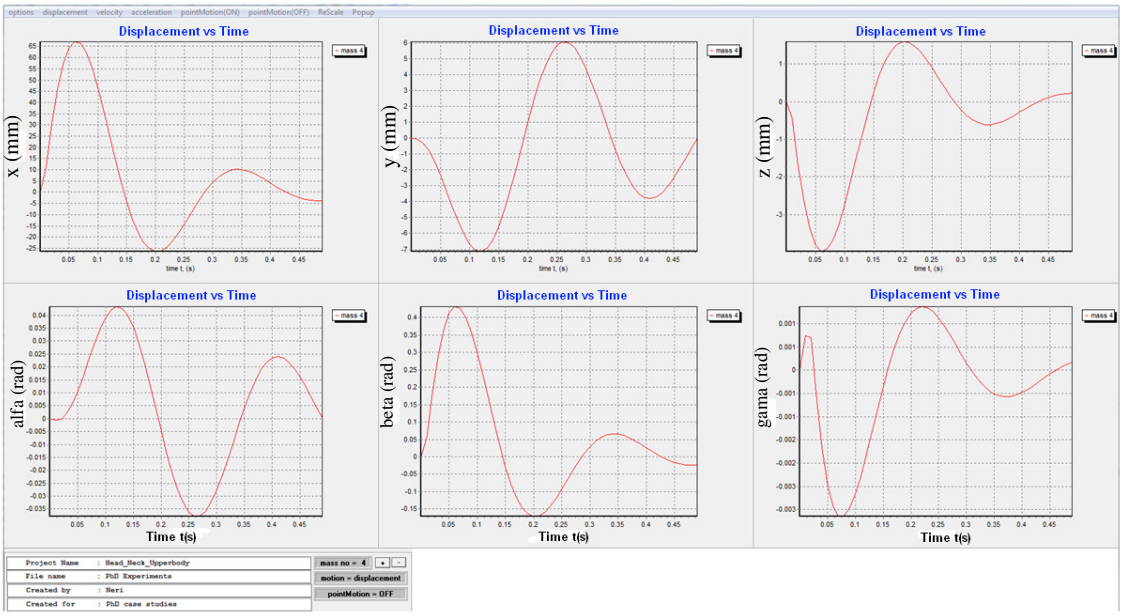


Figure 5.9 Displacement versus time response of head (mass 4) to half Sine shock force acting on the head

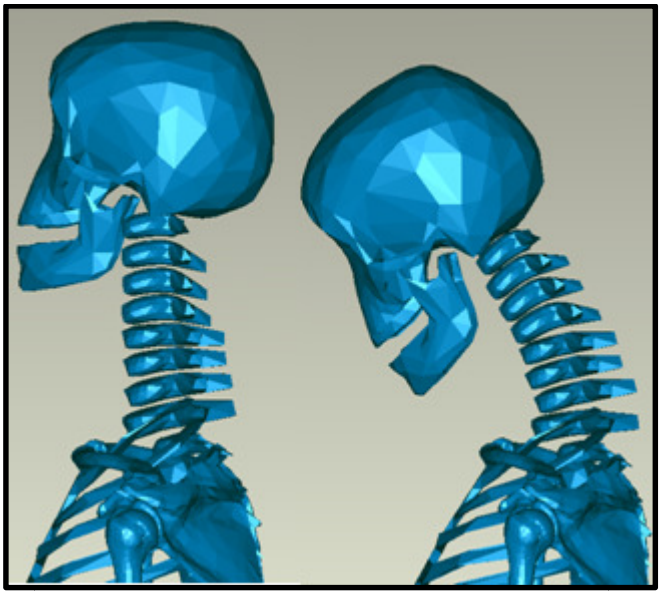


Figure 5.10 Head and neck flexion with the original geometries

The reason for doing this is to investigate relative displacement of the model under the shock described earlier, acting on T1. As it was explained, T1 shock makes all the bodies to translate (reaching to a constant velocity) and the tranlation movement is much larger than the relative movement between bodies, it is not possible to observe graphically the relative movement unless the constrained body motion is subtracted from the motion of each body. Instead, it is possible to fix the "ground" body or in this case T1 and apply an approximate shock acceleration induced forces to all the bodies. In this case the head mass is substantially greater than vertebrae masses and the force applied only on the head. There are a number of observations one can make out of these results, the result are very similar (almost identical in the x direction) to the results obtained by the Lifemod but there are some differences too. The difference is in the y direction. The Lifemod software gives no reading in that direction. The reason is simple, the Lifemod analysis is not based on bone surface geometries, attachment points are assumed to be relative to the centre of mass of each vertebra and symmetric. In MJM, these points are selected as visually observed. Even though the selection method appear to be approximate, it gives greater realism and probably demonstrates that slight variation of tissue attachment may result in motion that one may not observe if the structure assumed to be symmetric in every way.

5.4.2.1.3 Static force acting on head (T1 is fixed), deflection, the COR

In order to make sense of the results obtained from the MJM, the COM (cog) positions are tabulated in the Table 5.1, taken from the MJM database. In order to evaluate the centre of rotations, the COM positions are needed. To analyse the static deflection a small horizontal force (an arbitrary magnitude) applied on the head, with the torso fixed in every direction. The neck movement is shown in Figure 5.11. The deflections are magnified for the display purpose. The deflection results of the centre of masses are tabulated in the Table 5. 2. The centre of rotations is tabulated in the Table 5.3. There are number of interesting observations one can make of these results. As observed with the dynamic results, the horizontal force, results in the lateral deflection. The results observed here are in agreement with the dynamic results presented in the previous section as far as coupling concerned. However having the force acting in the horizontal direction on the head only distorts the deflection of the head in the x direction compared to the dynamically observed results.

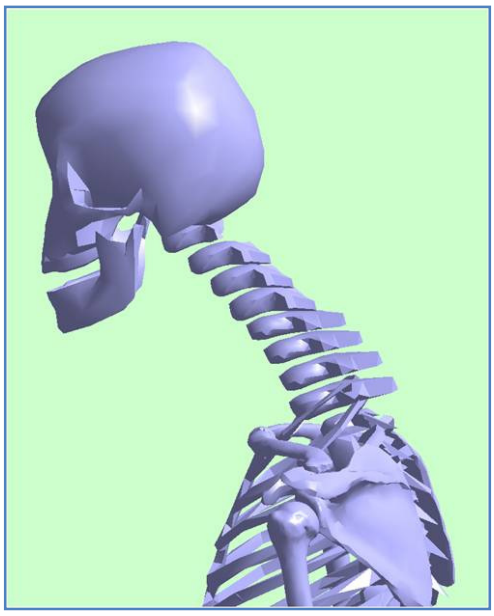


Figure 5.11 Response of the centre of mass of head under an applied constant horizontal force

Table 5.1 Centre of mass directions of 9 masses (head and cervical vertebra bones) extracted from MJM database

Mass no	COGx(mm)	COGy(mm)	COGz(mm)
6	-3.41003	0.831027	197.6785
9	-26.0425	-2.65552	549.8908
8	-24.3809	-2.61092	570.9855
1	-24.0875	-2.6109	590.3632
7	-20.958	-1.02725	611.0099
2	-21.1058	-0.94873	632.4982
3	-21.1058	-0.94873	654.3983
5	-16.7199	-0.78814	677.9849
4	-22.2881	-2.19565	756.0566

Table 5.2 Centre of mass movement under an arbitrary static force

Mass no	X(mm)	Y(mm)	Z(mm)	Alfa a(rad)	Beta b (rad)	Gama g (rad)
6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
9	0.2322	-0.3580	-0.2189	16.6690	19.5803	-16.5166
8	1.1131	-1.3839	-0.5077	39.4451	41.8723	-36.0307
1	2.3274	-2.7583	-0.6976	59.7789	60.1111	-56.8295
7	4.3249	-4.1728	-1.4025	59.0893	115.0576	-57.5116
2	8.0915	-5.5180	-1.9336	63.5862	171.1433	-57.5774
3	12.4604	-6.9278	-2.4048	66.1445	217.1731	-57.2551
5	17.9395	-8.7337	-3.7998	65.2526	260.7370	-56.9281
4	41.6993	-13.6278	-2.6433	66.8152	303.8847	-56.8358

Note. Angles are (1000 x *rad)*

The centres of rotations are also tabulated in the Table 5.3; in this case there are no Lifemod results to compare with. The important observation from this table relates to the interpretation of the centre coordinates. If the Table 5.1 is revisited, it is possible to observe that the gap between the centres of mass of each vertebra is approximately 20mm. And the centre of rotation (the difference between each vertebra) of each vertebra is approximately -10mm. In other word each centre is positioned at the bottom part of the vertebra which as it will be shown later is in the same range of (only by visual inspection) some published experimental results (Figure 5.16).

Table 5.3 Centre of rotation due to static displacements

Mass no	X(mm)	Y(mm)	Z(mm)
6	0	0	0
9	-10.9248	-0.1997	-11.2624
8	-15.4457	-4.3602	-21.9766
1	-19.0854	-8.6999	-29.2782
7	-20.3075	-8.3922	-37.6540
2	-17.9499	-9.4903	-48.0322
3	-16.9828	-10.2453	-58.4807
5	-19.9682	-10.3779	-70.4198
4	-15.8312	-22.0080	-136.2813

5.4.2.1.4 Joint stiffness, the centre of stiffness, joint stiffness invariants

The joint stiffness results are listed below. There are no published or similar results obtainable by any obvious source to compare with. It is expected that the formulation offered here would provide a scientific basis for a better assessment of joint laxity. As shown above joint laxity and joint deflection are seen as the same thing. Of course, concepts like the centre of rotation, the laxity are all function of applied loads and as demonstrated above, the magnitude of applied load and the selection of the force location, which is not a trivial matter, may influence the results. The interesting observation for the stiffness centre between 4 and 5 is that the stiffness centre coincide (difference between -136 and -70) with the head and C1 vertebra. The stiffness tensor is also generated but not displayed here. The principal axis shows that there is substantial symmetry in the stiffness system, indicating that the non-symmetric laxity results are probably due to the location of the COM of the head mass relative to the centre of stiffness. In general the application of the force vector is always going to be a problem when laxity is judged in terms of static deflection of a joint. The principal axes of stiffness, shows that at least one of the axes is completely isolated and the other two representing lateral stiffnesses, agreeing with the principal stiffness that the system is relatively soft in z axis but very stiff in two lateral axes.

All the points raised here apply for the second joint stiffness listed in the Table 5.4, between bodies 6 and 9 and also bodies 5 and 4. Just to point out that the vertebra thickness is approximately 10mm therefore, the centre of stiffness is positioned at the bottom of the vertebra again the centre of stiffness is at -4.45mm whereas the centre of rotation is at -11.26mm. Again both results are within the vertebra region. The principal stiffnesses and the principal directions are similar to the head-C1 joint.

Table 5.4 Joint stiffness and associated results

Centre of stiffness of the joint between body 5 and 4	**Centre of stiffness of the joint between body 6 and 9**
X Coordinate = -17.01 Y Coordinate = 0.86 Z Coordinate = -66.83	X Coordinate = -12.57 Y Coordinate = 3.31 Z Coordinate = -4.49
Principal Stiffnesses	Principal Stiffnesses
R 828.803522988841 I 0 R 33.1539292810169 I 0 R 828.636422733053 I 0	R 36.0733338614719 I 0 R 144.02408335007 I 0 R 145.710606103802 I 0
Principal Axes	Principal Axes
1.000e 00 0.000e 00 3.468e-03 0.000e 00 0.000e 00 1.000e 00 7.109e-01 1.000e 00 8.058e-03	0.000e 00 0.000e 00 1.000e 00 1.000e 00 6.189e-02 2.267e-01 0.000e 00 1.000e 00 0.000e 00

Note. R for real and I for imaginary

5.4.2.2 Motion of head and cervical vertebrae using Lifemod software

According to the results obtained from Lifemod software, head and neck movements are depicted as below. The analysis has been performed for frontal impact. The head mass is 11.1743833572 lb (5.068615 kg) and neck mass is 2.9014822662 lb (1.316 kg). Joints are created based on the Hybrid III crash Dummy based strength values with the Hybrid III scale factor 1 from the Lifemod software. As mentioned earlier these factors are used in order to provide the stiffness values for each rotational direction as in sagittal, transverse and frontal planes. Ligaments are interspinous ligaments, flaval ligaments, facet joint capsule ligaments and longitudional ligaments have been created from head through cervical vertebrae (C1-C7) to thoracic vertebra (T1). Basically ligaments are modelled as viscoelastic material with parallel spring and dashpot (Kelvin-Voigt model) elements with 100 lbf/in stiffness and 20 lbf/in/s damping values (these are 20kN/m 4kN/m/s). Muscles are semispinalis cervicis, trapezius, longus colli, semispinalis capitis and sternocleidomastoid muscles and modelled based on the Hill type muscle model. All required passive material properties and active contraction properties have been adapted from the Lifemod library. The dynamic analysis is based on the given translational acceleration to upper torso. In order to impose the acceleration the upper torso is represented as a translational joint where the translational velocity is created based on this constrained joint movement. The total simulation time

is 0.5 sec, where the acceleration is applied around 0.15-0.17 seconds and off at 0.2 seconds. The profile of the acceleration is a "half sine" shock which is convenient as it is one of the available options in the MJM force library. The head movement due to the analysis carried out by Lifemod software is shown in Figure 5.12.

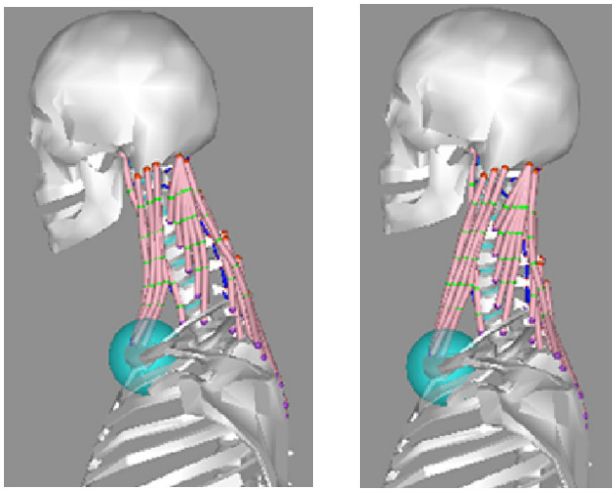


Figure 5.12 Neck and head motion after half sine acceleration applied to the upper torso

After completing the dynamic analysis variety of results are depicted as follows.

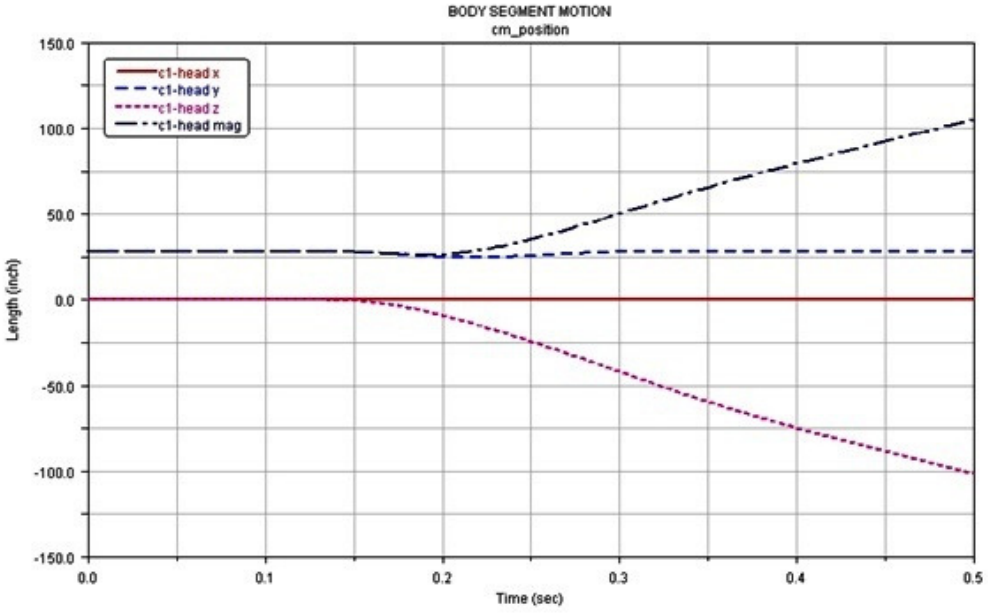


Figure 5.13 Centre of mass position of head versus time response to frontal impact applied to the upper torso

Results shown in the Figure 5.13 are very similar to the result obtained from the MJM simulations. The only difference is the starting time for the shock. In MJM the shock starts immediately whereas in the Lifemod software there is approximately a delay of 0.15s. As these results are from the demonstration case study provided by the Lifemod software, it was not possible to make changes to study alternative cases. Ignoring the actual response initiation, the results are almost identical. Velocity and angle plots of the head are similarly in good agreement as shown in Figure 5.14 and Figure 5.15 respectively. The substantial difference between these results and MJM is the lateral motion. In the Lifemod software, there is a complete symmetry, whether this is realistic in the real life, is debateable, the results are two dimensional, there are movements in the z and the x directions but none in the y direction. And the angle rotation is only in y direction (beta in MJM terms) due to fixed joint between the head and C1.

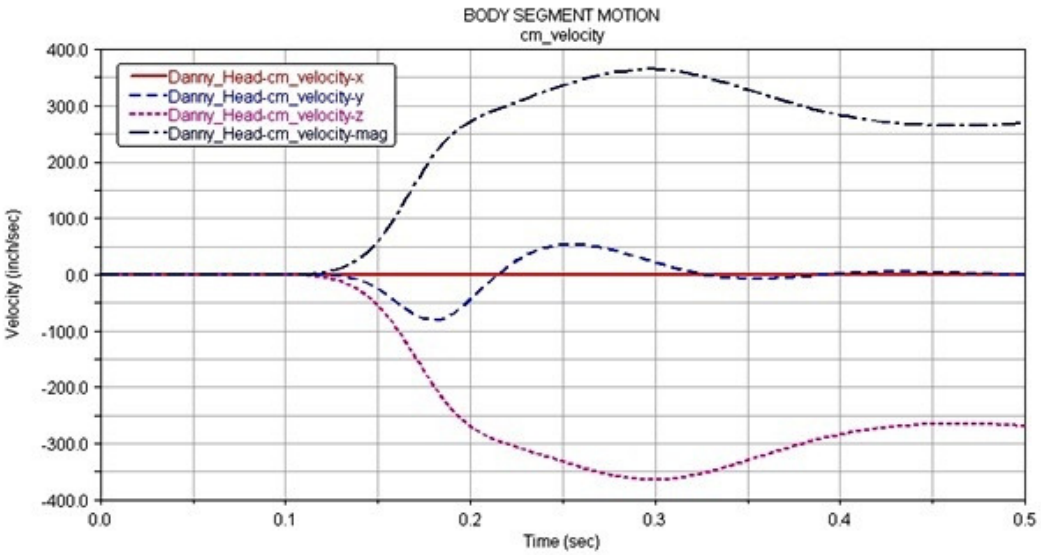


Figure 5.14 Centre of mass velocity of head versus time response to frontal impact applied to the upper torso

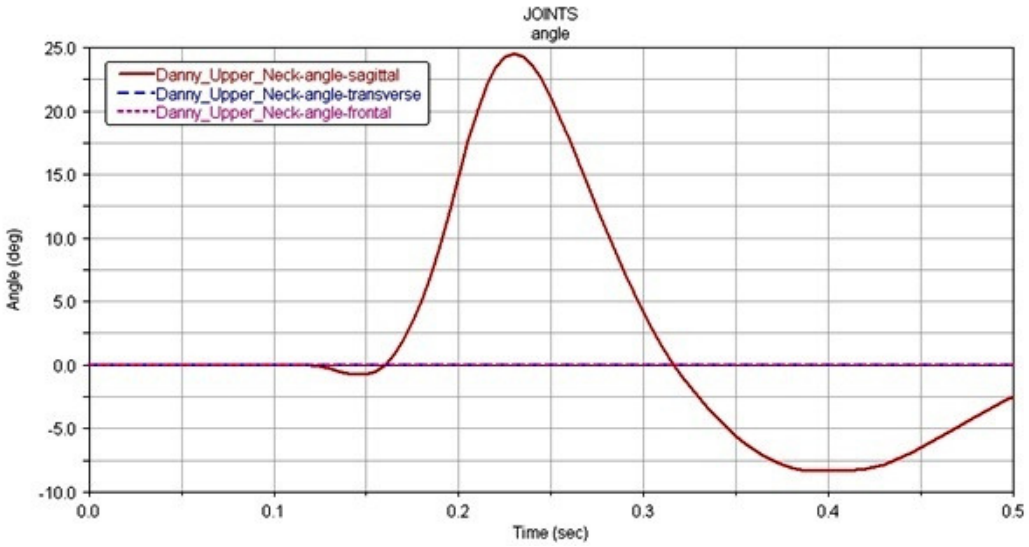


Figure 5.15 Angle of upper neck (joint between Head-C1) versus time response to frontal impact applied to the upper torso which shows one dimensional rotation in sagittal plane (horizontal direction)

5.4.2.3 Modelling or experimental based results from the literature

Regarding to the understanding of whiplash trauma due to car collisions, Woltring et al. (1994) studied instantaneous centre of rotation of neck and head. It has been emphasized that due to the serious spinal injuries with car accidents, it is very important to describe the head and neck motion and centre of rotations. In terms of kinematics, their model composed of head and neck, where the neck is defined as the cervical vertebrae column from C1 to C7. Additionally the thoracic vertebra (T1) is taken as a reference point of the model. Specifically dual pivot model has been used to describe the vertebrae column between C7 and T1 as one link and to describe the vertebral column between C6 to C1 as one link as well as for describing the head as the third link in the kinematic jointed chain. Their experimental study is based on the 3D video data and the formulation is based on the instantaneous helical axes estimations. The instantaneous position of the centre of rotation of various segments and also further information about neck kinematics can be found in the text book (Zatsiorski, 1997). The linked planar based kinematic modelling of head and neck has been performed by many researchers like Merrill et al. 1984. Additionally, the neck movement has been

assumed as performed on an arc with all equal contributions of individual vertebra. Additionally the common perspective is that, the cervical vertebrae from C3 to C7 moves as a unit and vertebra joint motions are nearly equal (Zatsiorski, 1997). Geometric based instantaneous centre of rotation measurement has been provided by Bogduk & Mercer (2000). They also show the sketch of the normal mean location and standard range of distribution of the typical cervical motion segments as in Figure 5. 16. It has been argued that the abnormal motion of cervical vertebrae can be assessed based on the instantaneous centre of rotations (Amevo, Aprill & Bogduk, 1992). It is difficult to interprate these results but the centre of rotation of each vertebra, approximately, appears to be within its own body. These findings, in approximate terms, in agreement both with the COR and the COS obtained by the MJM analysis. Additional information about the range of motion of cervical vertebrae joints is given in the Appendix A2.

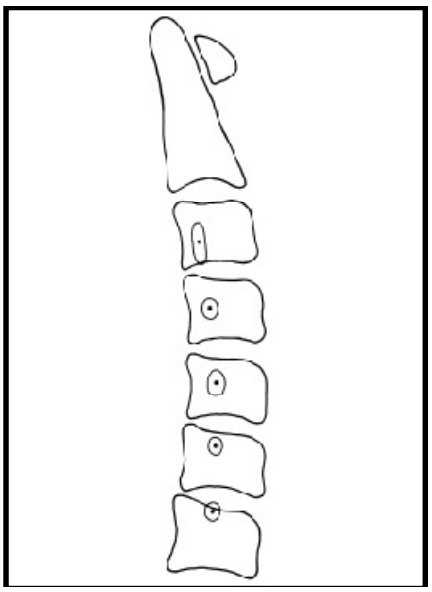


Figure 5.16 Instantaneous centres of each cervical vertebra during neck rotations (from Bogduk & Mercer, 2000)

5.4.2.4 Comparison of results from MJM software, Lifemod software and published data in head and neck modelling applications

The results obtained from the MJM software is in good agreement with the Lifemod software and are in reasonable agreement with the literature base results. The main difference between the Lifemod and the MJM is in the lateral motion. The MJM mount database is constructed by selecting attachment position interactively from the screen and similarly the force is acting on the mass centre which is calculated based on the meshed surface information. In the Lifemod software these are assumed to be completely symmetric, thus generating symmetric results. There are no equivalent results corresponding to the joint stiffness tensor and its invariants from the Lifemod or anywhere else. This capability of the MJM is novel and promising as it will further be discussed in the discussion chapter.

5.4.3 Elbow joint modelling

In the elbow joint model, major elbow ligaments such as radial collateral ligament (RCL), lateral ulnar collateral ligament (LUCL), annular ligament (AL) with anterior, posterior and transverse bundle of medial ulnar collateral ligament (MUCL) have been modelled as linear viscoelastic tissues. This was simply because no non-linear property (material non-linearity) was available in the published literature for the elbow ligaments. Six types of major flexor and extensor muscles as biceps brachii, brachialis, brachioradialis, triceps brachii, pronator teres and extensor carpi radialis longus muscles were modelled based on simplified Hills model. The cartilage contact was also created between radio-humeral and ulna-humeral joints. The cartilage contact between radio-ulnar joint was ignored. All the liner spring stiffness and dashpot viscosity parameters were adapted from the literature and the Lifemod software. In addition to this, wrist joint, hand with its finger joints were fixed to the forearm (ulna, radius, hand) and they moved as a single body. Hence, in the specific elbow joint flexion-extension analysis, only the elbow joint ligaments and major arm flexor and extensor muscles were considered. In terms of the anthropometric data, the humeral bone length between top of the grater tubercle and bottom of the capitulum is 294.6 mm. And the length of the radius between the radial head and the radial styloid process is 252.2 mm. The ulna bone length between the olecranon and ulnar styloid process is 256.7 mm. The

approximate length of the forearm between the olecranon and third distal hand phalanges is 429 mm. In considering the musculoskeletal, the weights of the humerus, forearm and hand were automatically calculated by the software. This means that mass parameters used here were lower than of the Lifemod but this was not a major problem as the motion considered was relatively slow and inertial effects were not expected to be significant and no scaling was used. Some of the anthropometric information was taken from the Lifemod software. Also in this case, the muscle and ligaments were manually attached. The elbow model constructed on the MJM software is shown in the Figure 5.17. All the utilized model parameters are stored in the MJM data base.

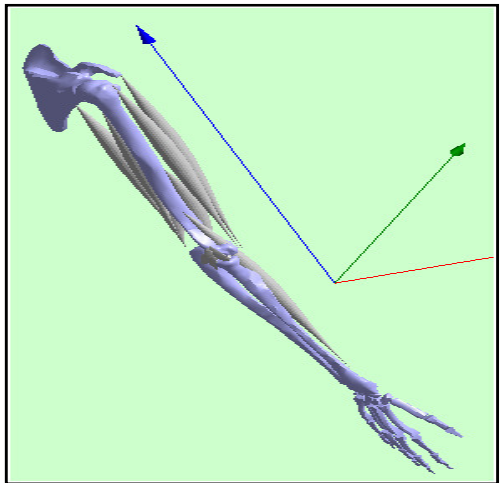


Figure 5.17 Elbow joint model from MJM software

5.4.3.1 Modelling based results from MJM software

- Initial considerations

In order to analyse the instantaneous centre of movement of the elbow joint, the centre position was calculated due to small linear deflection analysis or calculated at each time increment during the numerical integration. The centre and the principal axes of stiffness, the centre of rotation (under a given lateral load acting at the centre of the combined forearm complex), varus-valgus under varus and valgus load was calibrated to match with published results. Finally varus valgus deflection under the same load and addition of ligament incisions were performed to study the effect on varus valgus

deflection values. Before starting, the COM was calculated as (30.9, -209.5, 6.15) mm relative to the coordinate system used previously for the head and neck study.

5.4.3.1.1 Static deflection and centre of rotation of the ulna-radius complex

The main force is acting at the centre of mass position (given above) with (-1, 1, 0) direction with a magnitude which was adjusted for the laxity consideration, ensuring that the laxity did not go beyond 2.5^0 in varus or valgus directions. The static deflection results are given in the Table 5.5. In this table, angles are in rad*1000. Taking the alpha and beta values of 28 and 33.26, it can be shown that the vector sum of this is 2.49^0. The results also show that deflection at any insertion point does not exceed 0.4mm remaining well inside the linear region (although only linear stiffnesses were available for this study). Finally the results shows the centre of rotation coincide with the centre of stiffness as it will be shown later. Here the humerus bone is mass 1 and the forearm is taken as mass 2.

Table 5.5 Mass deflections, centre of rotation and mount deflections at insertion points under static loads

Deflection of mass	**X (mm)**	**Y(mm)**	**Z(mm)**	**alfa a(rad)**	**beta b(rad)**	**gama g(rad)**
1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	-4.5103	3.7649	-1.2254	28.0202	33.2598	-1.0318

Centre of rotation of mass	**X(mm)**	**Y(mm)**	**Z(mm)**
2	-19.4837	20.6030	135.0162

Deflection at insertion position of mount	**Position X(mm)**	**Position Y(mm)**	**Position Z(mm)**	**Deflection X(mm)**	**Deflection Y(mm)**	**Deflection Z(mm)**
3	-24.642	13.807	131.874	-0.110	0.095	-0.019
4	-38.485	10.100	133.591	-0.057	0.061	0.338
5	-44.980	2.339	144.892	0.311	-0.249	0.336
6	-32.868	-4.741	138.253	0.083	-0.075	-0.265
7	-31.201	-7.216	125.983	-0.328	0.267	-0.390

Note. Mass no 2(forearm)

5.4.3.1.2 Centre of stiffness and axis of stiffness

As with the head and neck model, the centre of stiffness, the inertia tensor of the joint and finally the principal values of stiffness and principal axes of stiffness are calculated as shown below.

The principal stiffness and the principal axes of stiffness are given in Table 5.6. Again the centre of stiffness and the centre of rotation are very close to each other, especially in the z axis position.

Table 5.6 Joint stiffness and related results

Centre of stiffness of joint	**X(mm)**	**Y(mm)**	**Z(mm)**
Joint between body 1 and 2	-34.44	2.86	134.92

Principal stiffnesses	Principal axes
R 305.967282234803 I 0	1.000e 00 2.925e-01 2.937e-01
R 17.6352419578363 I 0	3.177e-02 0.000e 00 8.876e-01
R 305.526200056914 I 0	0.000e 00 8.702e-01 1.000e 00

Note. R for real and I for imaginary

5.4.3.1.3 Dynamic calculations of centre of rotation with respect to the elbow angles

In this case the load was calculated to give an approximate 2.5^0 varus and valgus laxities and the same load was applied dynamically, the load was kept in the horizontal plane ensuring the turning moment remained approximately constant. In this case, full dynamics and non linear analysis were employed. Figure 5.18 shows how static tissue attachments, when run in dynamic mode employ another geometrical profile and start wrapping around the bony surfaces (impossible to capture a screen damp when all the muscles were dynamically plotted. Plot, re-plot takes a fraction of second and continues during the whole simulation).

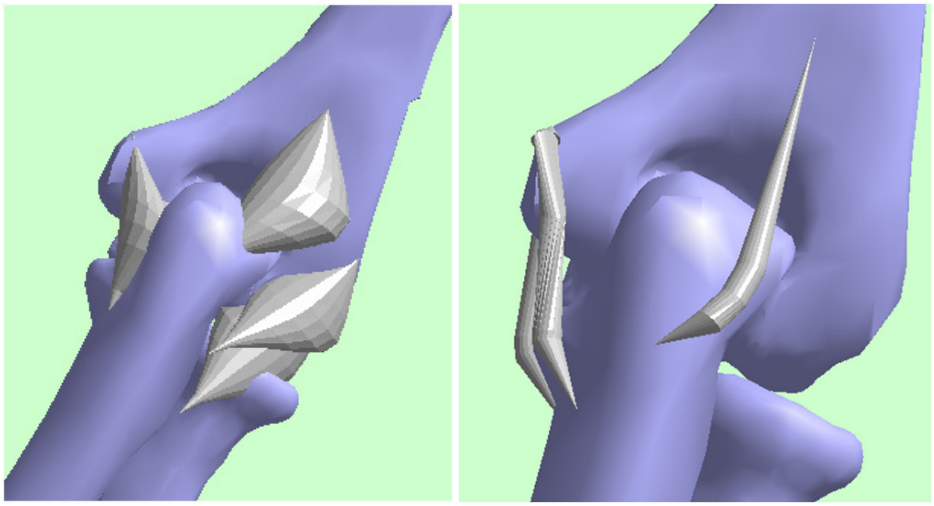


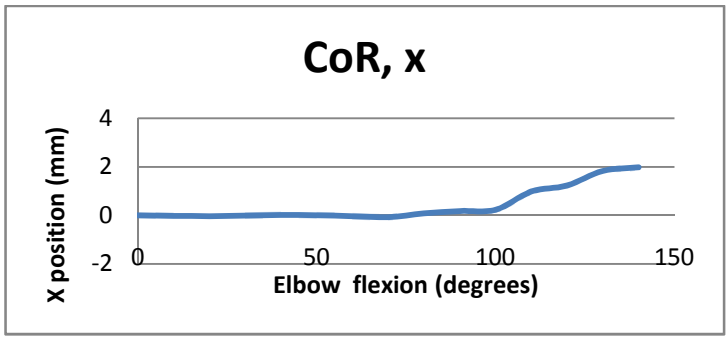
Figure 5.18 6DOF elbow joint model and the application of centre of movement analysis relative to flexion angle of forearm (ulna, radius and hand)

Again since there is no equivalent tissue wrapping results available in the literature the overall trend of laxity variation during the full flexion motion will be investigated with an intention of comparing it with the latest published material. This is the reason why the load was calibrated to give similar deflection both in the static mode and in slow dynamic mode. The muscle load was applied by describing it as a spring in tension. This is an effective loading mechanism which ensures that, firstly the load generated can be controlled accurately and secondly, eliminates the need to use the feedback control. However position control using this method is not always precise. The exact position where the motion terminates depends on the final equilibrium position. The following table is tabulated from the time series results generated by MJM by matching the time at which the required angle is calculated. A simple algorithm is written to perform dot product between the appropriate principal axes of radius and ulna complex and humerus. This method is considered to be accurate enough for the problem under consideration. The results are listed below, mass 1 is the humerus which is fixed and the forearm complex is the mass number 2.

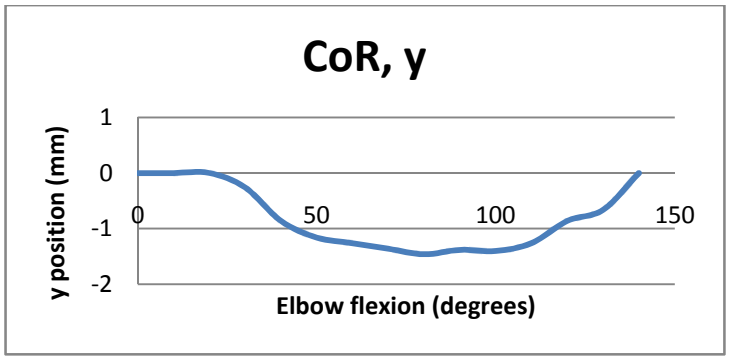
Table 5.7 Centre of rotation of mass 2

Flexion Angle (within ±1^0)	**x(mm)**	**y(mm)**	**z(mm)**
0	-19.402	20.603	135.0574
10	-19.506	20.603	135.0574
20	-19.516	20.603	135.0574
30	-19.496	20.3427	135.1558
40	-19.471	19.7427	135.2558
50	-19.402	19.4427	135.3558
60	-19.516	19.3427	135.3558
70	-19.556	19.2427	135.4058
80	-19.406	19.1427	135.4558
90	-19.306	19.2227	135.5058
100	-19.256	19.2027	135.5558
110	-18.506	19.3427	136.0558
120	-18.256	19.7427	136.5558
130	-17.656	19.9427	136.5558
140	-17.506	20.603	137.0558

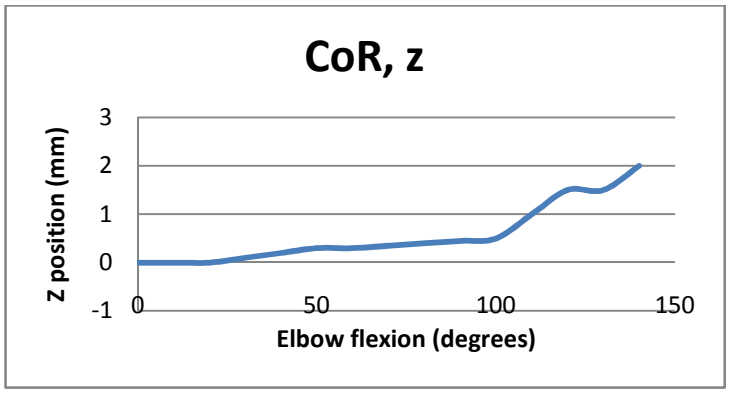
Plot of these results are shown in Figure 5.19 as:



(a)



(b)



(c)

Figure 5.19 3D change in centre of rotations (CoR) versus elbow flexion angle results obtained from MJM

5.4.3.1.4 Valgus and varus laxity studies with muscle wrapping and non-linear cartilage

The last part of the elbow analysis involved the full functionality of the software and the muscle wrapping algorithm was activated for all tissues. Unfortunately, it was difficult to turn off part of non-linearity of the individual components in order to identify, if for example, the muscle wrapping was necessary. Furthermore although ligament type tissue wrapping is an issue in elbow, muscles do not wrap around elbow joint and even though the bicep tendon connecting to the scapula wraps around the shoulder joint, it does not influence the results as the humerus is stationary. Another consideration is that, the calibration of force used for these analyses, for intact joint, produces responses which one may consider linear. The initial angle, when the flexion angle is 0, shown to be around $10^0 – 12^0$, which is the approximate neutral position of the forearm relative to humerus as it will be discussed further. This is in agreement with our software output. The non-linear analysis was run 5 times, first with no load and the angles between the principal axes of humerus and the principal axis of the forearm complex were measured (by simple dot product). An automatic selection of the principal axes along forearm complex and the principal axes along humerus for the elbow flexion angle calculation is not possible this was done manually. As it was mentioned previously, 2.5^0 degree laxity is used to calibrate the overall ligamental structure stiffness. Having done this, medial collateral ligament (MCL) is removed (stiffness set to zero) and the test for the full flexion range was carried out. The procedure was repeated for laxity due to ligament damage, lateral collateral ligament (LCL). The results obtained are very close to the result reported by Jensen et al. (2005) and Stavlas, Jensen & Sojbjerg (2007). The results are plotted in the Figure 5.20.

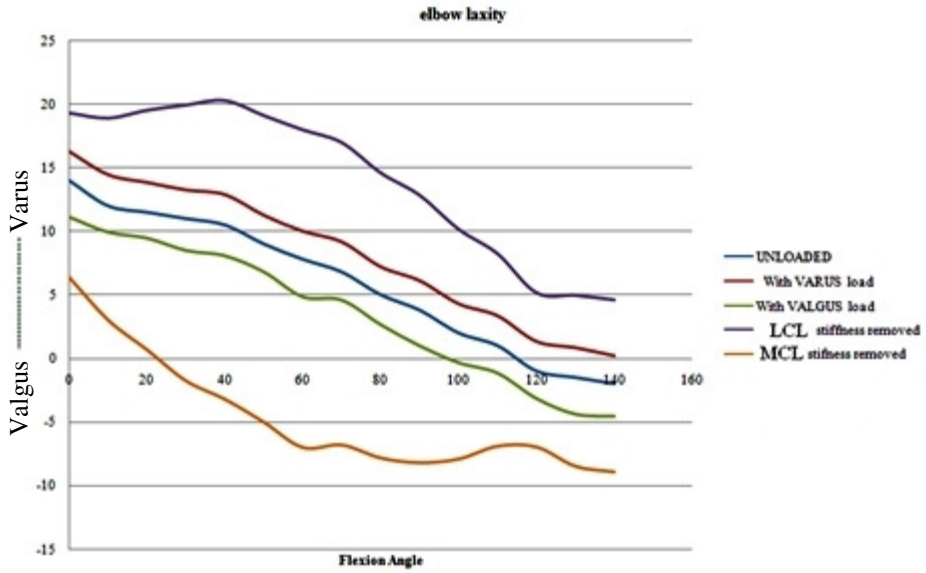


Figure 5.20 Valgus and varus laxity curves from unloaded and varus-valgus loaded intact elbow joint and from varus-valgus loaded elbow joint with complete LCL (varus stiffness) and MCL (valgus stiffness) incisions

5.4.3.1.5 Moment arm based results

The last numerical experiment was carried out for the calculation of the moment arm. Again the current state of the software did not allow this to be automated as explained above. The results from the MJM were obtained with respect to time. The same calculation explained in the previous section had to be performed. The angle calculated as the angle between two principal axes of the humerus and the ulnar/radius at the nearest required angles (every 20^0 taken in this study). The formulation for the moment arm calculations are given in the theory chapter. The MJM, in each step of the analysis generates the muscle force vector and also the angular velocity of the body. The articulation velocity of the body is the same as the joint velocity since the humerus is stationary. Angular velocity, the centre of rotation, a point (insertion point) on the muscle force vector and the direction of the force vector gives all necessary points and vectors for calculating the moment arm. Since this was done by transferring the result information to the excel sheet manually, it was time consuming and performed only for one muscle. The muscle action line for the bicep is shown in Figure 5.21. The moment arm results are plotted in Figure 5.22 as these results exhibits the same pattern as

previously reported results by Murray, Buchanan & Delp (2002), shown in Fig. 5.34. The results are in general agreement.

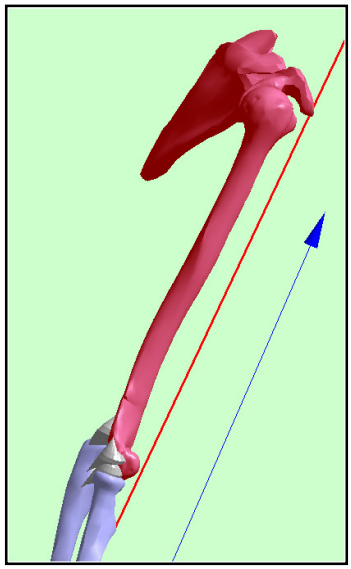


Figure 5.21 Force vector (in red) of Bicep muscle at neutral position of the forearm

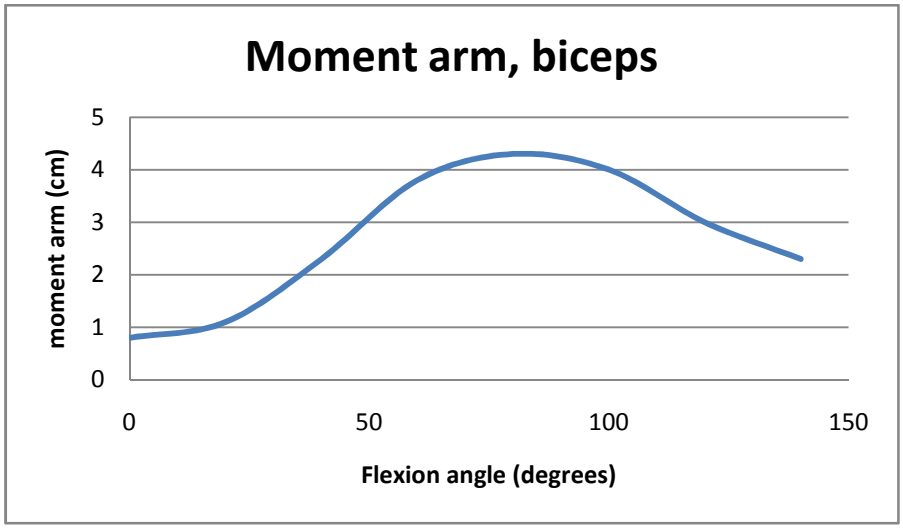


Figure 5.22 Moment arm change of Bicep muscle versus elbow flexion angle

5.4.3.2 Modelling based results from Lifemod software

Unfortunately the available example in the Lifemod software on elbow is a very basic one treating the joint as a hinge joint. Once a joint is treated as a standard joint then it is not a natural joint and analysis relating to the joint loading or realistic joint geometry will not be reliable or even possible. In the elbow joint model created, humerus is taken as a body and forearm (radius, ulna and hand) is taken as another body. In order to drive the dynamic flexion based joint analysis, firstly the elbow joint is created as a hinge joint and results demonstrate the "linkage" articulation and variation in joint torques and gross motion kinematics, they are not interesting in terms of joint kinematics and joint behaviour. Therefore no graphical results are included here. The standard Lifemod documentation provides detailed information.

5.4.3.3 Experiment based results (carried out in the Brunel Orthopaedic Research and Learning Centre)

The reason for presenting these results is to demonstrate the difficulties involved in measuring joint kinematics experimentally and also show the range of laxities measured in-house experiments. Number of different devices was developed in the Brunel Orthopaedic Research and Learning Centre, only two will be presented here. The first device was developed as a generic device to measure six degree of freedom motion of a rigid body. This was based on the Steward Platform (flight simulator). The second device was an elbow brace specifically developed to measure the joint, valgus varus laxity.

5.4.3.3.1 The Stewart platform based centre of rotation of the ulna-radius complex

The device was developed (or more accurately, adopted) in the Brunel Orthopaedic Research and Learning Centre. The Steward platform is a kinematic based measurement device which is able to plot forearm movements with respect to the humeral bone. The device consists of six linear potentiometers where the change in displacement on these potentiometers is obtained and plotted using the forward kinematics analysis available in the Matlab. In the Figure 5.23 the Steward Platform

and axes system is shown and in Figure 5.24 the angle alpha is plotted which directly associates with valgus varus movement. In order to be effective in measuring a specific joint kinematics, the bones at each end of the joint must be fixed to the fix and moving parts of the platform respectively. The experiments were carried out using volunteers and strapping their arms to the device. Therefore the expected accuracy was limited due to the skin movement.

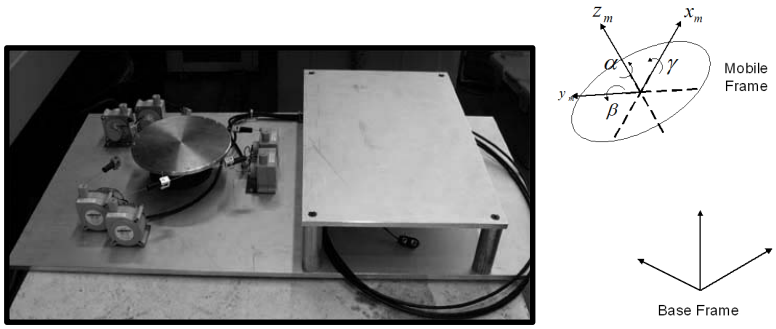


Figure 5.23 Steward Platform with 6 Celsco SP1 transducers (Alrashidi et al. 2009)

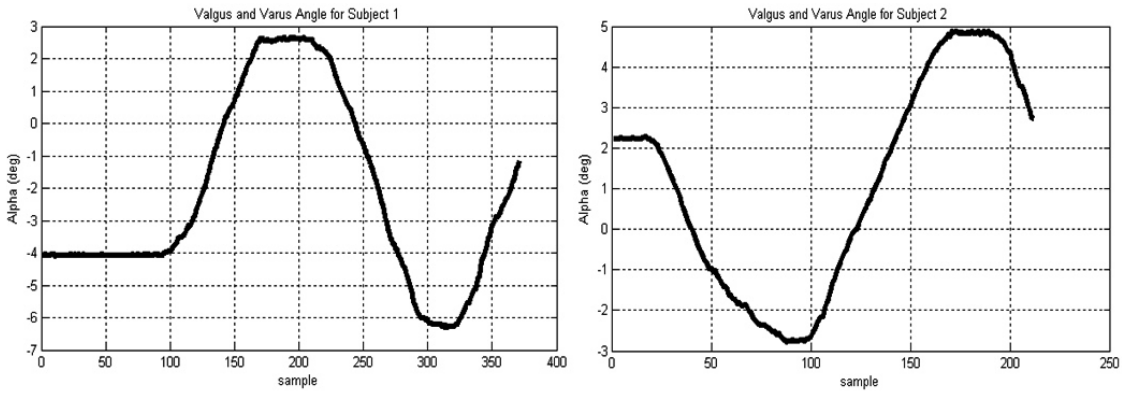


Figure 5.24 In vivo, elbow joint valgus and varus angle measurement of two subjects from Stewart Platform

The device was also used to calculate the centre of rotation. The results are plotted in Figure 5.25 for two subjects from in vivo measurements.

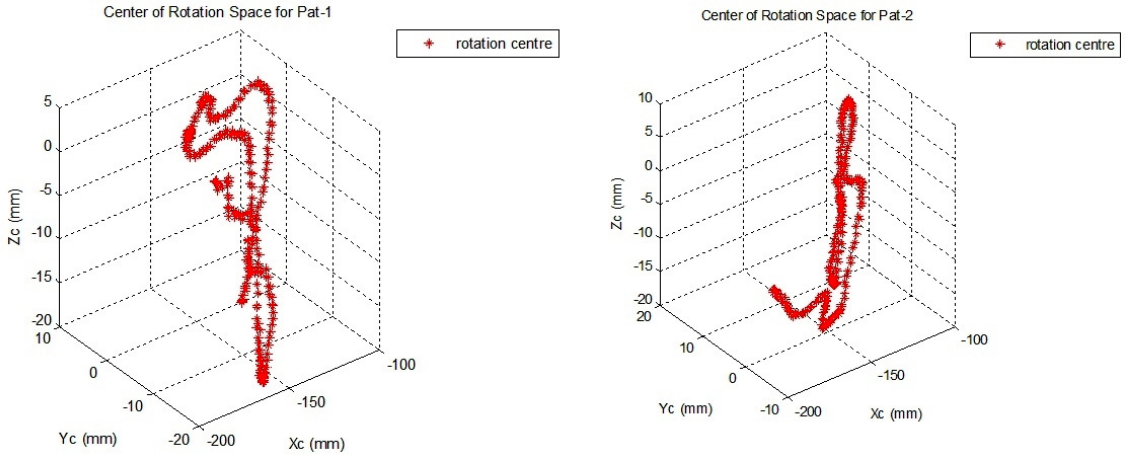


Figure 5.25 3D changes in centre of rotations during arm flexion of two subjects from Steward Platform

Above results show that during elbow joint articulation, there is no fixed centre of movement. The device has recorded 10^0 of valgus and varus angle ranges, shown in the Fig. 5.24. Although the apparent laxity relatively high, this is probably due to skin movement of the strapped arm. Unfortunately at the time of writing the thesis no experiments were carried out on any cadaveric joint where bones could be screwed onto the platform to evaluate the real laxity using the device. Similarly the results for centre of rotation were obtained using the platform display a large range, larger than anything reported previously for a healthy joint. Showing that strapping is not a reliable method of fixing an arm to the platform. Nevertheless results indicate that the real laxity or the range of the centre of rotation will be within the result generated by the platform. In that sense previously calculated MJM results are well within what is given by the platform.

5.4.3.3.2 Valgus and varus laxity studies with the Elbow Brace

The second experimental application has been performed with another device (Elbow brace) was developed in the Brunel Orthopaedic Research and Learning Centre. The device composed of potentiometers to measure arm flexion and extension movements as shown by PF and strain gauges to measure the varus-valgus angles (calculated from the strain readings) which is shown by SGV. The experiments were performed in vivo (with a living subject) and in vitro (with a cadaveric specimen). In these experiments the arm movements are measured relative to forearm neutral position with various positions of humerus. In vivo and in vitro experimental set ups are shown in Figure 5.26 and Figure 5.27 respectively. In the experimental set ups, the rotational (supination-pronation) movements are not measured and the applications are based on the forearm neutral position with approximately negligible rotational movements. For the skin movement problem two approaches have been considered such as using straps to tight the arm which enables to fix the device on living people arm non-invasively (Figure 5. 26). The other approach is screwing the device on to the upper arm which is required to make two holes on the humerus (Figure 5.27). This application is only possible to perform on cadaver arms. With neglecting arm supination and pronation rotational movements, the device needed to be fixed to the ulna and humerus. Through applying this constraining application, humerus and ulna have been forced to move relatively with restricting translational and rotational movements.

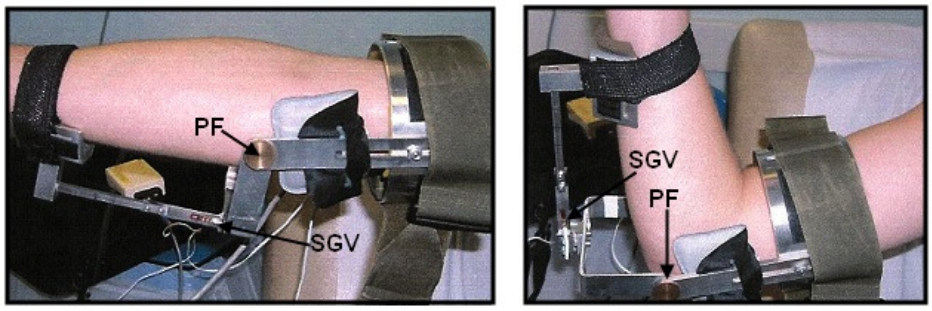


Figure 5.26 In vivo experimental set up for measuring elbow joint movements with a potentiometer (PF) for flexion-extension movement and with a strain gauge (SGV) for varus and valgus movement. The extended arm (Left) and flexed arm (Right)

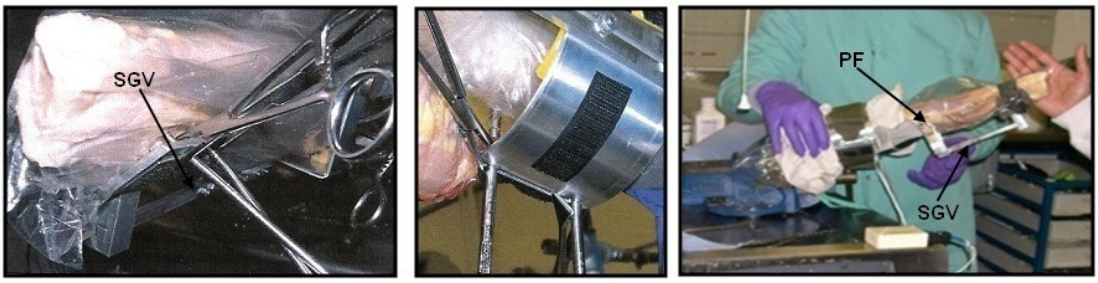


Figure 5.27 In vitro experimental set up for measuring elbow joint movements with a potentiometer (PF) for flexion-extension movement and with a strain gauge (SGV) for varus and valgus movement. The experiment is performed under applied varus and valgus loads

In vivo tests, the device is fixed with device reference position (0,0,0). The humerus and the lower arm are mobile relative to the arm's neutral position. 10N varus and 10N valgus forces are applied (using a digital weighing scale) to measure varus-valgus movements relative to the neutral flexion-extension movements. The Reference axis of varus-valgus movement is the beam axis where the strain gauge is positioned. The flexion-extension axis is relative to the potentiometer axis adjusted close to the medial and the lateral epicondyles on the humeral bone.

Table 5.8 Valgus and varus displacements relative to elbow flexion angles; in vivo results (average of more than 30 patients + students)

Neutral Flexion Angle (deg)	Varus Angle (deg)	Valgus Angle (deg)	Forced Varus Angle (deg)	Forced Valgus Angle (deg)
10	2.5	-2.5	4	-2.0
60	5.5	-1	6	0
90	4	0	5	2
120	2.2	2	3	3

In Vitro results, the ulna and the humeral bones are fixed to the device and device is fixed to the table. 10N varus and 10N valgus forces are applied to measure varus-valgus movements relative to the neutral flexion-extension movements. The Reference axis of the varus-valgus movement is the beam axis where the strain gauge is attached. The flexion-extension axis is relative to the potentiometer axis located close to medial and lateral epicondyles on humeral bone. Only external forced based movements have been generated.

Table 5.9 Valgus and varus displacements relative to elbow flexion angles; in vitro results (single sample)

Forced Flexion Angle (deg)	Neutral Varus Angle (deg)	Neutral Valgus Angle (deg)	Forced Varus Angle (deg)	Forced Valgus Angle (deg)
10	2.5	-2.5	3	-2
60	3	-1.5	4	1
90	2	1	3	2
120	1.5	2	3	3

5.4.3.3.3 Comments on the experimental results

These experiments were carried out in the Brunel Orthopaedic Research and Learning Centre. For the in vivo tests there were no shortage of volunteers and researchers used their friends and also some clinical tests were carried out in the Hillingdon hospital with patients consent. The cadaveric tests were carried out on a single sample which makes it difficult to make generalised conclusions. However the results found was well within previously reported findings.

5.4.3.4 Modelling or experimental based results from the literature

5.4.3.4.1 Centre of rotation of elbow joint

Elbow joint axis of flexion-extension movement has been described by the contact surfaces of the trochlea and the capitellum on the humeral bone. The joint articulation traditionally is described in terms of the axis of rotation and the centre of rotation and the literature analysing the joint kinematics, focuses specifically on this. There is a very little literature relating the motion kinematics directly to articulating bone surfaces. The study performed by Bottlang et al. (2000) is one of very few who actually discusses surfaces of trochlea and capitellum and their relationship to the screw displacement axis (SDA) and with respect to the joint kinematics Figure 5. 28. The figure also illustrates why at least a 10^0 valgus exist at extended (0 flexion angle) position when the ulnar angular orientation is measured with respect to the central axes of humeral bone.

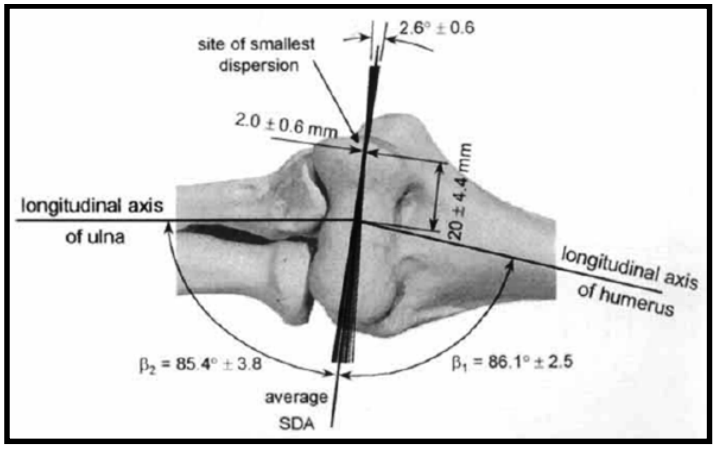
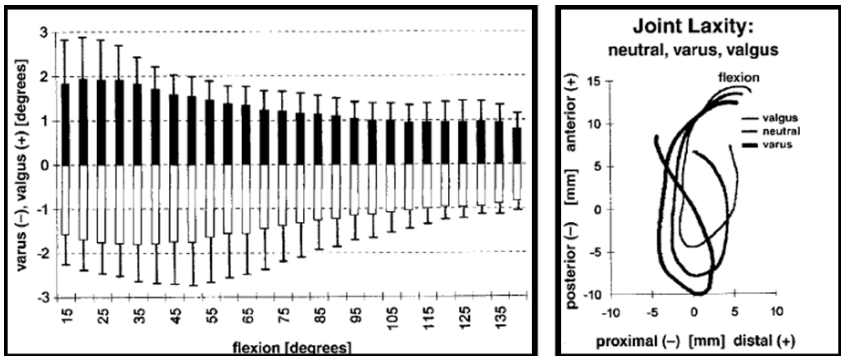


Figure 5.28 Location of the average screw displacement axis (SDA) with respect to humerus and ulna in the frontal plane. β_1 represents the angle between the SDA and the longitudinal axis of humerus. β_2 represents the angle between the SDA and the longitudinal axis of ulna. The carrying angle is represented as the summation of beta angles as $\beta_1 + \beta_2$. The excursion of the SDA (2.6^0) is shown exaggerated by factor of 2.0 to help visualize the site of the smallest dispersion of the axis (Bottlang et al., 2000)

They were also able to depict the change in SDA path in 2D during flexion and extension movements (Figure 5.29)



(a) (b)

Figure 5.29 (a) Valgus and varus movements versus flexion angles of elbow joint. (b) The SDA paths in 2D relative to flexion angles (Bottlang et al., 2000)

Summary of the SDA displacements and rotations relative to the flexion under applied varus and valgus moments and neutral conditions are given in Table 4.10. 2D illustrations of these angular variations are shown in Fig. 5.30. In considering the elbow joint kinematics, these are the most widely reported results.

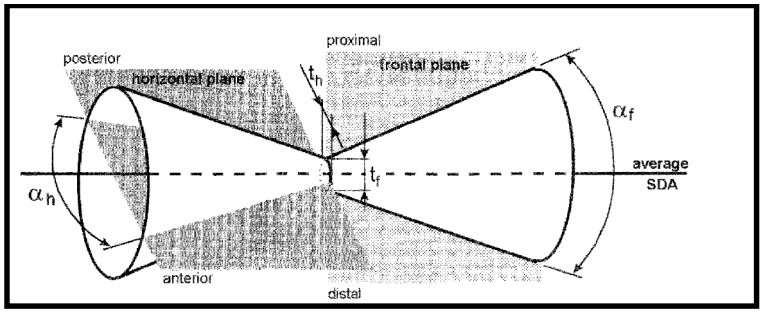


Figure 5.30 the Idealization pathway of SDA. α_h and α_f frustum vertex angles, t_h and t_f translations of the SDA in horizontal and frontal planes (2D) (Bottlang et al., 2000)

Table 5.10 Frastum parameters and waist location for neutral elbow flexion (roller configuration A) compared with flexion under applied varus or valgus moments of 0.5 Nm (Bottlang et al., 2000)

Flexion Mode	t_h (mm)	t_f (mm)	α_h (deg)	α_f (deg)	Frustum waist location (mm)
Neutral	1.4 ± 0.3	2.0 ± 0.6	5.7 ± 2.2	2.6 ± 1.0	19.9 ± 4.4
Varus	1.8 ± 0.6	2.0 ± 0.7	7.3 ± 2.3	4.2 ± 1.7	18.7 ± 8.3
Valgus	1.6 ± 0.4	2.1 ± 0.8	6.8 ± 1.9	2.6 ± 0.7	26.0 ± 8.6

In the experimental work above, humerus has been fixed and the dynamic based experiment which has been performed through applying loads to biceps, brachialis and triceps muscles. They placed the screw displacement axis (SDA) on the frontal (vertical) plane along the line between the bottom of the trochlear sulcus and the periphery of the capitellum and the medial facet of the trochlea. They have depicted the translational and orientational change in two dimensional positions of the SDA along the forearm flexion-extension. The experiment also employed the flexion / extension movements with forced varus and valgus conditions.

5.4.3.4.2 Valgus and varus laxity studies with and without tissue deficiency

Furthermore, Jensen et al. 2005 performed an experimental work to investigate the LCL division and radial head excision in elbow joint laxity. They also state the importance of MCL in valgus laxity of elbow joint and role of LCL in elbow dislocation as well as constraining capacity of radial head in varus displacement. They hypothesised that the isolated radial head excision is not the main cause of laxity but it causes the insufficient LCL performance which can cause main joint laxity. In the experimental application, specimen arm is adjusted on the experimental set up with placing the humeral bone parallel to the floor. Relative to this adjustment, the forearm flexion extension and varus valgus movements have been performed; the test rig is shown in the Figure 5. 31

and the results are plotted in the Figure 5.32. They have found that the radial head excision increase the varus laxity by a mean of 4.8° and increase the external rotator laxity by a mean of 7.1° with respect to the reference intact joint positions. Through the radial head prosthesis, decrease in laxity has been achieved as mean 1.3° varus and 1.2° external rotation. Additionally, with the radial head excision and LCL incision the increase in varus laxity as mean of 15.6° has been obtained with the increase of external rotatory laxity as mean of 18.6°. Also the isolated incision of LCL showed higher laxity results than isolated excision of radial head as mean of 14.1° varus and 14.7° external rotation. As a conclusion, they emphasized the indirect radial head constraint role in varus displacement and external rotation as it puts the LCL under tension.

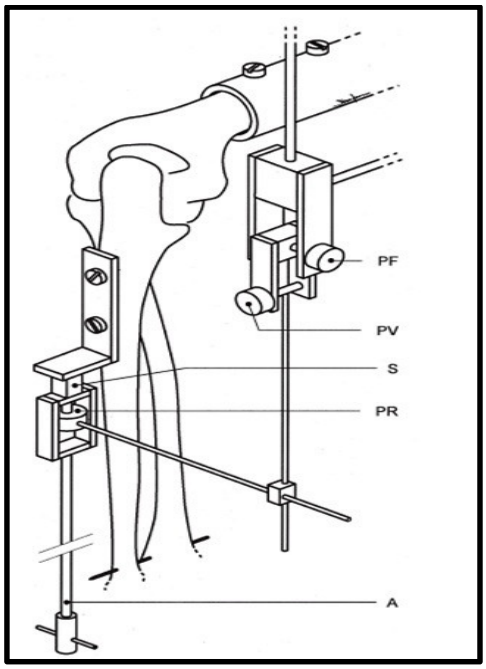
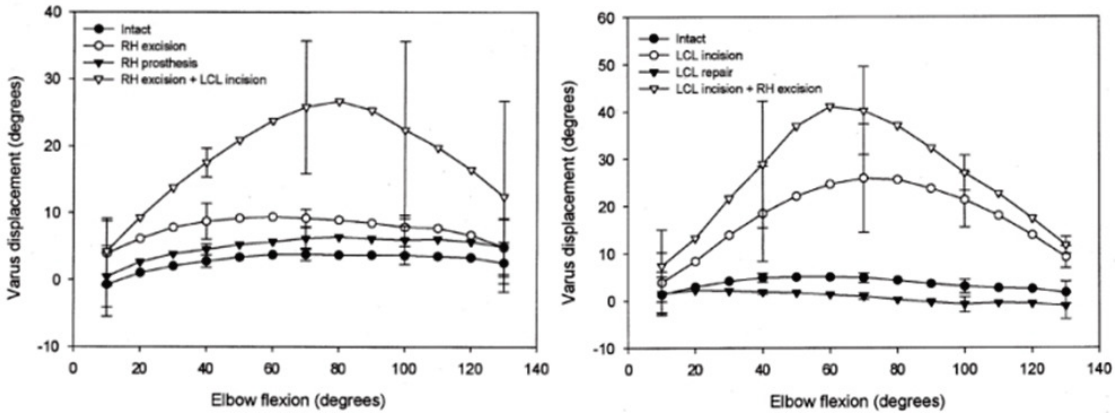
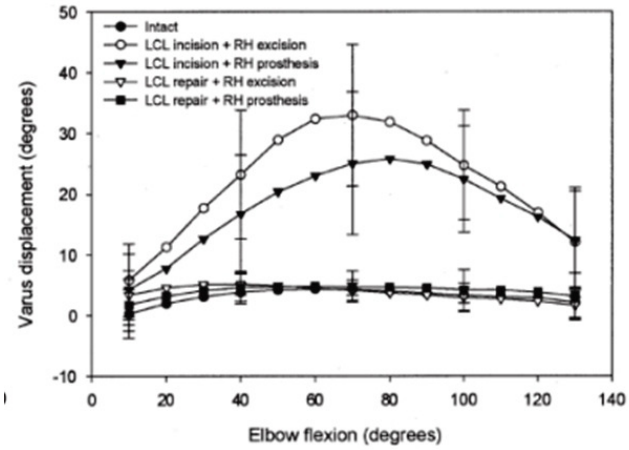


Figure 5.31The linkage system based experimental set up which is commonly utilized to measure flexion-extension, varus-valgus and rotational movements of elbow joint with potentiometers (PF, PV, PR) (Jensen et al., 2005)



(a) (b)



(c)

Figure 5.32 Mean varus displacements relative to elbow flexion with firstly performing radial head excision (a), though firstly performing LCL incision (b), with combination of radial head excision and LCL incision and also combination of radial head replacement and LCL repair (c) (Jensen et al., 2005)

More recently, Stavlas et al. (2007) performed experimental test in order to depict the varus and valgus displacement relative to elbow flexion with a fixed centre of rotation based on the commonly performed linkage based experimental set up (Figure 5.31). The intension was to provide the information about the stabilizing effect of LCL and MCL ligaments in varus and valgus laxity as well as providing the information about stabilizing effect of the elbow fixator in varus valgus laxity. Results from the experimentally tested elbow specimens are depicted in Figure 5.33.

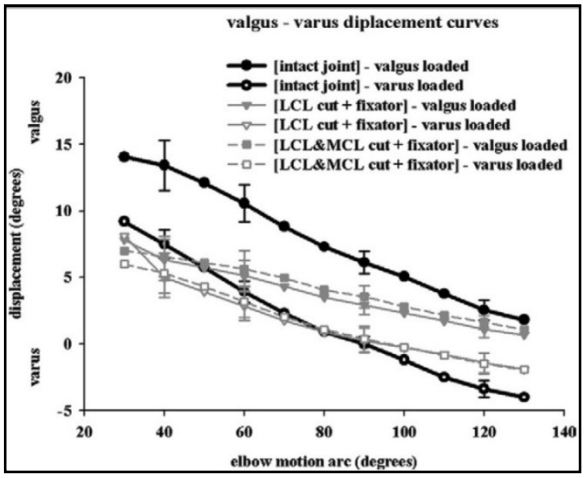


Figure 5.33 Valgus and varus displacement (deg) versus flexion angles (deg) of elbow joint in various conditions (Stavlas et al., 2007)

5.4.3.4.3 Moment arm results

Murray, Buchanan & Delp (2002) performed an experimental work to measure the peak moment arms of the flexor and the extensor muscles of the elbow joint. In their study, the elbow joint has been modelled as a hinge joint with a fixed centre of rotation where the muscle moment arms are calculated based on the fixed centre of rotation. The work is two dimensional and the moment arm of each muscle is defined as the perpendicular distance from the centre of rotation of the joint to the muscle action line. They investigated muscle moment arms of the bicep brachii, brachialis, brachioradialis, extensor carpi radialis longus and pronator teres and triceps (given in the Appendix A3). The moment arm estimation has been calculated as the partial derivative of measured tendon displacement with respect to the joint angle, also given in the Chapter 3. The moment arm plot of the bicep muscle relative to elbow flexion angles is shown in the Figure 5.34.

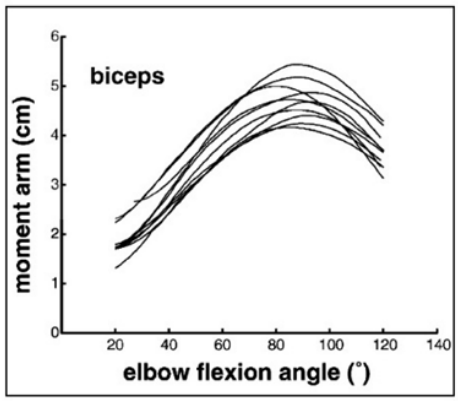


Figure 5.34 Moment arms plot of bicep muscle during flexion angles of elbow joint based on ten different specimens (Murray et al., 2002)

Table 5.11 Summary of peak moment arm data of elbow (Murray et al., 2002)

Muscle	Mean (cm)	Range of Peaks (cm)	Angle of Peak (deg)	Range of Angles (deg)
Brachioradialis	7.7	7.0 - 9.0	108	100-118
Biceps	4.7	4.2 - 5.4	88	80-93
ECRL	3.2	2.6 - 4.5	106	99-115
Brachialis	2.6	2.1 – 3.0	88	76-102
Pronator Teres	1.7	1.3 – 2.0	100	94-113
Triceps	-2.3	-1.8 to -2.8	44	1-62

5.4.4 Summary of results

The results chapter has been performed for the following tasks as:

1. The software developed to test a number of new formulations and theories put forward in this project. And presented some functionalities and capabilities of

the software. The novel theories includes the modelling of linear joint stiffness, non linear analysis of deflection and dynamic response based on surface contact driven motion and tissue wrapping algorithm and derivation of the joint stiffness tensor and the tensor invariants.

2. Data preparation involved extracting relevant published data and default values used in the Lifemod software. For the surface geometries, for the elbow joint, digitised cadaveric bone surface data used for the articulating joint areas.
3. The proposed formulations and theories needed to be tested using case studies, for this, two problems were investigated. a) Head-neck complex and b) Elbow compex.
4. With the head-neck complex; a) A static deflection analysis was performed to investigate the centre of rotation, b) A joint stiffness analysis was performed to investigate the joint stiffness, the centre of stiffness, the principal stiffnesses and c) A dynamic analysis was performed to investigate the response under the half sine acceleration acting on the torso and investigate the response under the half sine force acting on the head.
    a. The static deflection analysis results gave results comparable and in the same range as of previously published experimental results.
    b. The joint stiffness analysis had nothing comparable in the orthopaedic literature but never the less produced results which gave interesting association with the centre of rotation. Also identified possible reasons of non-symmetric deflection results observed in the static analysis.
    c. The dynamic analysis gave very similar results in comparison with the Lifemod analysis on the same problem.
5. With the elbow complex, the Lifemod software had little in terms of case studies and the only case referred to was an elbow study with a hinge joint which was not relevant. Following analyses were carried out. a) The static deflection analysis was performed to investigate the centre of rotation. b) The joint stiffness analysis was performed to investigate the joint stiffness, the centre of stiffness, the principal stiffnesses. c) The dynamic analysis to investigate the response under a static load acting laterally to study the valgus-varus deflection range under different conditions. Some cases studied were, the valgus-varus measures for an unloaded intact joint, varus and valgus loaded intact joint and

with ligament incisions. The study also included the moment arm calculations for the intact unloaded bone.

a. The static deflection analysis are used to calibrate the initial stiffness values of the joint to match with reported valgus / varus range and further deflection analysis showed to be in agreement with other reported case studies.
b. Again the joint stiffness analysis had nothing comparable in the orthopaedic literature but the results found to be interesting and related to the deflection and the centre of rotation obtained in the laxity analysis.
c. Dynamic analysis performed in the time domain, but the results required for the laxity, the centre of rotation in dynamic situation and the calculation of the moment arm for the comparison purpose, needed to be presented as a function of the flexion angle. This was done by selecting the time corresponding to the flexion angles.

CHAPTER 6

DISCUSSION

6.1 Introduction

In this chapter, the contributions which have been made and their significance to the current knowledge is discussed. Some discussion has already been conducted in the results section and here some summarised and further discussion will be given where needed.

6.2 Objective

The objective of the project was to develop a human joint modelling frame work based on rigorous mathematical formulations describing its kinematics, stiffness and the motion of bodies connected by the joint. The need for this kind of investigation arises because such a framework does not exist and the current joint modelling is primitive and it is only an extension of multi body analysis adopted from the mechanism analysis.

6.2.1 Structures of discussion

- Mathematical modelling
- Software implementation
- Results and summary of comparative work
- General discussion

6.3 Mathematical modelling

It was decided that anatomic joints would be modelled as a structural construct where its stiffness would be determined by surrounding tissues, cartilage and its surface geometry. Thus this would be appropriate to describe the joint as a 6DOF system and the articulation direction of the joint is controlled by its stiffness system. In considering the stiffness ratios between ligaments and cartilage as two main stiffness elements, it

was decided that approach was feasible and that the "stiff system" problem would not be encountered.

6.3.1 Linear modelling

The linear modelling was developed from scratch and the author is not aware of the particular formulation being published anywhere else. The formulation facilitates the construction of the stiffness and damping matrix of a discrete stiffness and damping elements attached at arbitrary points on joining bodies. The damping and stiffness in the elements are defined in their principal axes and orientation by the Euler angles or alternatively by their end points. The main principal axes for muscle/tendon is described as the direction of the two attachment points as a straight line. The other two are arbitrarily placed to describe tissue as axi symmetric. As for cartilage the main principal axis is described as the average normal at the contact point. The averaging is needed as there is no certainty that the contacting surface will be smooth. Again it is assumed that the cartilage stiffness is axi symmetric.

6.3.2 Non-linear modelling

There are two types of non-linearity, material non-linearity and the geometrical non-linearity. The material non-linearity may exist even if the motion is small which the case of the cartilage behaviour. A small deflection creates a highly non-linear reaction force. The situations with ligaments are less non-linear but are subjected to larger deflection and therefore can also operate in their non-linear region. The treatment proposed here, unlike to the linear model, requires no matrix assembly for the stiffness and damping elements or mass elements. The method uses the free body diagram representation of the problem. This means that each body equation is written separately and any interacting force treated as external force on each body. The motion of each body is treated in its own principal axis system. Therefore forces have to be transformed accordingly. The spring stiffnesses for cartilage and ligaments have to be calculated in their own principal axis system and they have to be converted. Therefore forces from the local (principal) axis of each spring have to be converted to the local (principal) axis of individual bodies of the joint. This deals with the material non-linearity and such transformation is required even if the system is geometrically linear.

However, in dealing with a completely non-linear system, attachment points of tissues change but also tissues wrap around bone or other tissue surfaces and their action line has to be calculated as the simulation progresses. As the contact stiffness and the contact points change, these points need be calculated at each step with the average contact normal. The novelty here relates to the overall algorithmic approach and handling of local non-linearity by procedural means.

6.3.3 Joint stiffness concept

The mathematical definition of joint stiffness is a new proposal, in general determining how strong a joint is also known as the "joint laxity". This is an informal term and simply reflects how much deflection one would expect when a moment applied on a joint. Sometimes the "joint stability" used as a term to describe if a joint functions without violating its natural range. These terms are not strictly scientific and they do not provide grounds for evaluating joints effectively. Although joint laxity is used widely, published results always give full conditions under which the laxity is measured to ensure that the results are quantifiable and repeatable as it will depend on these parameters and most importantly to the applied moment. In order to describe the joint stiffness, this must be independent of the applied forces. In other word, it must be an inherent property of the joint. The mathematical definition of stiffness between two bodies is already obtained. In general, where system has more than one joint, it is not difficult to isolate the required joint under investigation and reduce the stiffness matrix to the specific joint. Having done this yields the joint stiffness matrix which can be used to obtain invariant properties of the joint. The formulation proposed in this project in terms of obtaining the invariant properties of a stiffness structure is completely novel. The joint stiffness matrix is a 6 x 6 matrix, which are made of 4, 3 x 3 matrices. Two non diagonal matrices couple rotation and translation of joint movement. The elimination of non diagonal elements gives the conditions of decoupling. The position which de-couples the translation from the rotation is known as the centre of stiffness. Finding this point was not trivial because among a number of possible solutions, a stable solution was needed to be identified. Some solutions would result in ill condition solution system. Having found the centre of stiffness, at this position, with the updated stiffness matrix, which will have only top diagonal sub matrix representing the translational stiffness matrix and bottom diagonal, representing the torsional stiffness

matrix. The Eigen value of the top diagonal matrix gives the principal, translational stiffnesses and translational directions. Equally the bottom diagonal matrix will give the principal, torsional stiffness and torsinal direction. The modelling of these joint invariants is completely new and never proposed before. It is expected that they can be used to describe joint stiffness properties in a more meaningful way as are inherent property of anatomic joints.

6.3.4 Muscle wrapping

This is presented as part of the theory because of the formulations that is required to manipulate the surface geometry and intersections. The muscle (or tissue) wrapping involves calculating the shortest distance between two points on a surface described by the meshed surface data. Describing the muscle wrapping as the shortest distance between two points on an analytical surface may not have an analytic solution but there are rear cases where an analytical optimum is possible because the surface can unwrap to a plane, such as a cylinder. Dealing with complex and numerically described surfaces means that only heuristic solution is possible. The heuristic proposed in this thesis involves finding tissue paths which will be as close as possible to the straight line between the two selected points. However, a path segment being on the closest to the shortest distance line (the straight line) at a given point in space is not sufficient to be on the minimum path. It is important that the path segments belonging to the "closest to the straight line" set should be connected for the whole path. The proposed method offers a compromised solution. The solution is described in detail in the theory chapter and found to work well testing it in the evaluations carried out in the software section and also during the simulations. The proposed method is new.

6.3.5 Collision detection

The collision detection algorithm was developed to ensure that bodies (made of mesh surfaces) do not penetrate each other during the motion. The basic idea of the algorithm is to calculate the distance between points as one point is placed on one body and the second point is on the other body. In this case the distance is a two dimensional measure, the projection of the Cartesian distance between these two points on the approach velocity and the perpendicular distance to the approach velocity. The

perpendicular distance determines if there is a likelihood of collision (potentially colliding points) and the distance along the velocity vector determines how close these points are so that the closest can be selected. The closest distance is divided by the velocity which gives the time to collision. The method is fundamentally different than commercial collision algorithms which perform the nearest distance calculations in time increments and decide what do to when penetration takes place. Unfortunately the "culling" proposed in the proposed method is very basic and there is room for improving its performance.

6.4 Software validation

Human joint modelling was recognised to be data rich and needed to be interactive. It would have been impossible to develop an analysis system where all data is obtained accurately and pre assembled and still gave flexibility to execute a large number of evaluation tests. Not all capabilities of the developed software were documented partly because the development has been an ongoing process and during the write-up process software development has continued. And secondly some of the capabilities related to the usage of the program rather than the fundamentals.

6.4.1 Individual modules

In software development it is necessary to test individual routines in isolation before assembling them into the main program. Some of the testing carried out will be discussed here. The testing interactive capabilities, such as ensuring that the point picked is on the object, or ensuring that the rotation is real (and not because of the "camera") and many other seemingly trivial problems will not be discussed (although more than half of development time had gone into this kind of activity). Following simple tests were carried out

1. Mass inertia calculations. The algorithms were tested against simple analytical objects.
2. Inertia tensor and principal inertia. Partly from published results and partly by visual inspection.
3. Boolean operations. Visual inspection and wireframe display

4. Muscle or tissue wrapping. Visual inspection and again inspecting the wireframe displays.

5. Collision detection. Visual inspection and plotting contact pairs. In collision detection the proposed algorithm was tested against the shortest distance algorithm provided by the KernelCAD platform and also against the exhaustive search method.

User interface and database: It was necessary to create a system where data could be created interactively. This was important because muscle, ligament, tendon attachment could be located on the screen and stored for the later use. Also forces can be attached interactively. Therefore it was necessary to develop a flexible graphical user interface. Interface has been shown in the result chapter and a typical tissue attachment was demonstrated. Most of surface related data base elements are generated by the user interface. The user interface has large number of functionalities, including the manipulation of objects using Boolean operations. The user interface was developed with an intention to facilitate implant insertion and evaluation. The database fields which are not bone surface related, have default values and any new record is loaded with these values. The association of the data base and the graphical user interface text boxes are dynamically linked and the program is designed to be upgradeable with new data fields.

Analysis modules evaluation: Currently following modules are active,

a) Linear deflection.

b) Non-linear deflection (material nonlinearity only), both (a) and (b) comes with the instantaneous centre of rotation.

c) Joint stiffness, joint tensor, centre of stiffness, principal axes of rotational stiffnesses.

d) Non-linear analysis with tissue wrapping.

The validation is presented in the result chapter through modelling application of head and neck and elbow joints. These are tested using the linear deflection to study the joint laxity. The joint stiffness is studied with the stiffness joint centre and associated properties. The non-linear analysis module with muscle wrapping tested only for the

elbow motion. The full flexion motion was analysed and corresponding laxity results were compared with the published results.

6.5 Evaluation of results

The discussion section gives the rationale of comparing results from various sources. It was argued that because of the nature of the problem it would be very difficult if not impossible to obtain identical results for many reasons. Therefore it was decided that comparison would be on the general behaviour and the response trend analysis. All mathematical modelling about the human joint put forward in the theory section were studied and evaluated using two examples, human neck and head, made of 9 bodies and elbow complex which is made of humerus jointed to radius ulnar. Where, radius ulnar is taken as a single body.

6.5.1 The joint laxity

This is effectively a deflection analysis. The deflection analysis was carried out on the neck example gave very close result in comparison to Lifemod analysis. The results also revealed differences which are previously discussed. The results obtained from MJM are also in close agreement.

6.5.2 The centre of joint stiffness

This is a new concept, at least in the biomechanics area and a new tool in characterising joint stiffness. Numerically it is demonstrated that the joint stiffness matrix does change as predicted by shifting the stiffness definition to the centre of stiffness by zeroing all coupling coefficients of the matrix, thus removing and coupling between rotation and translation. It is also shown that the principal stiffness coefficient generated as symmetric as expected. Any evaluation beyond this is difficult without having anything to compare with. It is also observed that the centre of stiffness is very close to the centre of rotation. This found to be an interesting result however, it was thought to offer some explanation to non-symmetric deflection observed in MJM results when the neck whiplash was analysed.

6.5.3 Non-linear analysis with muscle wrapping

This was the most difficult investigation partly because it took some times to get a stable solution. It was difficult because in MJM, all results (of elbow) were generated with respect to the time whereas all published elbow results were with respect to the flexion angle. It meant that the result had to be evaluated and exact angle matching published results to be selected for the comparison. Five sets of analysis were performed giving remarkably close results to the previously published results. One set was run without any load; two sets run to generate 2.5^0 valgus and varus deflection for the calibration purpose. Further tests were run with incision of the medial collateral ligament (MCL) and lateral collateral ligament (LCL). The results obtained were in the same range as of published results agreeing with the increase in varus and valgus laxity in the presence of ligament incision. For the same elbow with no lateral load, moment arm was calculated also giving results similar to the published results. Although results were in agreement, it is difficult to argue that the elbow was the best joint to study wrapping algorithms, or even the "joint stability". Probably shoulder would be a better choice as most of the modelling based applications and tissue wrapping applications have been performed mainly for this joint. Nevertheless analysis generated enough results to show its effectiveness for the elbow joint. For the future development, the shoulder joint can be studied to test the algorithm to its limit.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

The work reported in this thesis, encapsulates the theories and algorithms developed to drive the core analysis modules of the software which was developed to model a musculoskeletal structure and particularly the diarthrodial and amphiarthroses joints. What makes the proposed modeller different than currently available modellers is that the joint kinematics is based on the local joint surfaces and contact geometry. There are many modellers capable of modelling gross human body motion. Nevertheless, none of the available modellers offers complete elements of joint modelling, it appears that joint modelling is an extension of their core analysis capability which is the musculoskeletal motion dynamics. It is felt that an analysis framework focused on human joints would offer a significant benefit and potential to be used in surgical environments. The local mobility of joints has a significant influence in human motion analysis, and in understanding of joint loading or contact forces. Thus, an accurate analysis of joint motion is very important in medical applications such as implant evaluation or surgery assistance and assisting medical devices and instrumentations.

7.1.1 Summary of the contributions made in this work

1. Linear modelling. Development of the joint stiffness and damping matrix. Suitable for the linear range of tissue stiffness and damping properties. This is the first of its kind and it gives a firm analytical basis for investigating joints with surrounding tissue and the cartilage.
2. Non-linear modelling. A new scheme is described for modelling the motion of multi bodies joined by non-linear stiffness and contact elements. The proposed method requires no matrix assembly for the stiffness and damping elements or mass elements. The novelty here relates to the overall algorithmic approach and handling of local non-linearity by procedural means.
3. Joint stiffness. The mathematical definition of joint stiffness is a new proposal; it is based on the mathematical definition of stiffness between two bodies.

Based on the joint stiffness matrix properties, number of joint stiffness invariants were obtained analytically such as the centre of stiffness, the translational principal stiffnesses, the rotational principal stiffnesses and corresponding to these principal stiffnesses, their axes. Altogether, a joint is assessed with its six principal axes and six principal stiffnesses and the centre of stiffness. These formulations are new and show that a joint can be described in terms of inherent stiffness properties. It is expected that these will be better in characterising a joint in comparison to the laxity.

4. Muscle wrapping. The muscle (or tissue) wrapping involves calculating the shortest distance between two points on a surface described by the mesh data. A new heuristic algorithm was described and demonstrated. The heuristic is based on minimising the accumulative divergence from the straight line between two points on the surface and the direction of travel on the surface.
5. Collision detection. The novel algorithm proposed here identifies the possible collision points on motion path by redefining the distance as a two dimensional measure, the distance along the velocity approach vector and perpendicular to the approach vector. The perpendicular distance determines if there is a likelyhood of collision (potential collision) and the distance along the velocity determines how close they are. The closest among potentially colliding points give the "time to collision". The algorithm can eliminate "fly pass" situation where very close points may not collide because of the direction of velocity.
6. Software. Software platform providing capability for analysing joints with six degree of freedom based on joint surfaces is new. The software is highly interactive and driven by well structured database, designed to be highly flexible for the future developments.
7. Two case studies were carried out generating results relating to all the proposed elements of the study. The results which were previously published or generated by the Lifemod software, show good agreement with the proposed model results whenever comparison was possible. In some cases where the comparison was not possible because there were no equivalent results, the results were supported by other indicators. The results were supported also by experiments carried out in the Brunel Orthopaedic Research and Learning Centre.

7.2 The future work

It is believed that the understanding of the mechanics of musculoskeletal joint is probably one of the most important single items in orthopaedics. The task, to certain extend, was achieved by developing the mathematical models and the software framework which enabled to better understand the joint behaviour. However the research not only answered some questions but also raised certain questions to be investigated in the future. It is possible to list a number of areas where the future work can focus on, however it is believed that three particular issues stands out among many others:

1. The first one is the joint stiffness and joint stiffness invariants and their relationship to the local articulation surface and the surface invariants and their relationship to the instantaneous centres of motion. Research in this field can reveal quantitative measures for accurate implant insertions.
2. Collision response. Although collision response handled by describing the contact points by springs and dashpots representing the cartilage. However there is a possibility that this method may reach to its limit as the contact become harder, especially when metal to metal contacts are tested.
3. Improvement of the muscle wrapping algorithm. Not only allowing the muscle to wrap but also devising schemes for guiding it through natural paths irrespective its minimum path trajectory.

References

Abdel Rahman E., Hefzy M., (1993) "A Two Dimensional Dynamic Anatomical Model of the Human Knee Joint," *Journal of Biomechanics* **115**, pp. 357-364.

Abdel Rahman E., Hefzy M., (1998) "Three Dimensional Dynamic Behaviour of the Human Knee Joint under Impact Loading," *Medical Engineering & Physics* **20**(4), pp. 276-290.

Abdel-Malek K., Yang J., Marler T., Beck S., Mathai A., Zhou X., Patrick A., Arora J., (2006) "Towards a New Generation Virtual Humans: Santos," *International Journal of Human Factors Modeling and Simulation* **1**, pp. 2-39.

Ackland D., Pak P., Richardson M., Pandy M., (2008) "Moment Arms of the Muscles Crossing the Anatomical Shoulder," *Journal of Anatomy* **213**, pp. 383-390.

Adams C., Baldwin M., Laz P., Rullkoetter P., Langenderfer J., (2007) "Effects of Rotator Cuff Tears on Muscle Moment Arms: A Computational Study," *Journal of Biomechanics* **40**, pp. 3373-3380.

Ahmad C., Park M., ElAttrache N., (2004) "Elbow Medial Ulnar Collateral Ligament Insufficiency Alters Posteromedial Olecranon Contact," *The American Journal of Sports Medicine* **32**(7), pp. 1607-1612.

Alrashidi M., Yildiz I., Alrashdan K., Esat I., (2009) "Evaluating Elbow Joint Kinematics with the Stewart Platform Mechanism," *Eighth International Conference on Modelling in Medicine and Biology*.

Amevo B., Aprill C., Bogduk N., (1992) "Abnormal Instantaneous Axes of Rotation in Patients with Neck Pain," *Spine* **17**(7), pp. 748-756.

An K., (2005) "Kinematics and Constraint of Total Elbow Arthroplasty," *Journal of Shoulder and Elbow Surgery* **14**, pp. 168S-173S.

An K., Himenso S., Tsumura H., Kawai T., Chao E., (1990) "Pressure Distribution on Articular Surfaces: Application to Joint Stability Analysis," *Journal of Biomechanics*, 23, pp. 201-203.

An K., Hui F., Morrey B., Linscheid R., Chao E., (1981) "Muscles across the Elbow Joint: A Biomechanical Analysis," *Journal of Biomechanics* **14** (10), pp. 659-669.

An K., Takahashi K., Harrigan T., Chao E., (1984) "Determination of Muscle Orientations and Moment Arms," *Journal of Biomechanical Engineering*, **106** (3), pp. 280-282.

ANYBODY http://www.anybodytech.com

Asada H., Slotine J. E., (1986) "Robot Analysis and Control," John Wiley and Sons.

Atkinson T., Haut R., Altiero N., (1997) "A Poroelastic Model that Predicts some

Phenomenologycal Responses of Ligaments and Tendons," *Journal of Biomechanical Engineering* **119**(4), pp. 400-405.

Audenaert A., Audenaert E., (2008) "Global Optimization method for Combined Spherical-Cylindrical Wrapping in Musculoskeletal Upper Limb Modelling," *Computer Methods and Programs in Biomedicine* **92** (1), pp. 8-19.

Baciu G., Wong W., Sun H., (1999) "RECODE: An Image Based Collision Detection Algorithm," *Journal of Visualisation and Computer Animation* **10**(4), pp. 181-192.

Bahler A., (1968) "Modeling of Mammalian Skeletal Muscle," *IEEE Transactions on Biomedical Engineering* **15**(4), pp. 249-257.

Bajaj C., Dey T., (1992) "Convex Decomposition of Polyhedra and Robustness," *SIAM Journal on Computing* **21** (2), pp. 339-364.

Bapu P., Evans S., Kitka P., Korna M., McDaniel J., (1980) "User`s Guide for Combiman Programs, Version 4," *University of Dayton Research Institute*.

Baraff D., (1989) "Analytical Methods for Dynamic Simulation of Non Penetrating Rigid Bodies," *ACM SIGGRAPH Computer Graphics*, pp. 223-232.

Baraff D., (1994) "Fast Contact Force Computation for Nonpenetrating Rigid Bodies," *Proceedings of the 21th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 23-34.

Barequet G., Chazelle B., Guibas L., J. Mitchell, A. Tal, (1996) "BOXTREE: A Hierarchical Representation for Surfaces in 3D," *17th Annual Conference of European Association for Computer Graphics*, pp. 387-396.

Barr A., (1984) "Global and Local Deformations of Solid Primitives," *ACM SIGGRAPH Computer Graphics* **18**(3), pp. 21-30.

Bartels R., Beatty J., Barsky B., (1987) "An Introduction to Splines for Use in Computer Graphics & Geometric Modeling," Morgan Kaufmann-USA.

Bassett R., Browne A., Morrey B., An K., (1990) "Glenohumeral Muscle Force and Moment Mechanics in a Position of Shoulder Instability," *Journal of Biomechanics* **14**, pp. 659-669.

Bei Y., (2003) "Dynamic Simulation of Knee joint Contact during Human Movement," PhD Thesis, University of Florida.

BIOMET http://www.biomet.co.uk/resource/7474/Verso%20Design%20Rationale.pdf

Blakeley F.M., (1980) "Cyberman," Chrysler Corp. Detroit, Michigan.

Blankevoort L., Kuiper J., Huiskes R., Grootenboer H., (1991) "Articular Contact in a Three Dimensional Model of the Knee," *Journal of Biomechanics* **24**(11), pp. 1019-1031.

Blinn J., (1982) "A Generalization of Algebraic Surface Drawing," *ACM Transactions*

on Graphics 1(3), pp. 235-256.

Bogduk N., Mercer S., (2000) "Biomechanics of the Cervical Spine. I: Normal Kinematics," *Clinical Biomechanics* **15**, pp. 633-648.

Bonner S., Kelley R., (1988) "A Representation Scheme for Rapid 3-d Collision Detection," *Proceedings of IEEE International Conference Symposium on Intelligent Control*, pp. 320-325.

Bosboom E., Hesselink M., Oomens C., Bouten C., Drost M., Baaijens F., (2001) "Passive Transverse Mechanical Properties of Skeletal Muscle Under in Vivo Compression," *Journal of Biomechanics* **34**, pp. 1365-1368.

Bottlang M., Madey, M., Steyers C., Marsh J., Brown T., (2000) "Assessment of Elbow joint Kinematics in Passive Motion by Electromagnetic Motion Tracking," *Journal of Orthopaedic Research* **18**, pp. 195-202.

Bradshaw G., O`Sullivan C., (2002) "Sphere Tree Construction Using Dynamic Medial Axis Approximation," *Proceedings of ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 33-40.

Bradshaw G., O`Sullivan C., (2004) "Adaptive Medial Axis Approximation for Sphere-Tree Construction," *ACM Transactions on Graphics*, pp. 1-26.

Brandt K., Radin E., Dieppe P., Van De Putte L, (2006) "Yet More Evidence that Osteoarthritis is not a Cartilage Disease," *Ann Rheum Dis.* **65**, pp. 1261-1264.

Buford W., Anderson C., (2006) "Predicting Moment Arms in Diarthroidal Joints-3D Computer Simulation Capability and Muscle-Tendon Model Validation," *Proceeding of 28th IEEE EMBS Annual International Conference*.

Buford W.L., Andersen C., Elder K., Patterson R., (2001) "Verification of Spline-path Muscle Models for a 3D Simulation of the Extremities," *Proceedings of the International Society of Biomechanics* (ISB), pp. 206.

Cameron S., (1986) "Collision Detection by Four Dimensional Intersection Testing," *IEEE Transactions on Robotics and Automation,* pp. 291-301.

Carman A., Milburn P., (2005) "Dynamic Coordinate data for Describing Muscle-Tendon Paths: A Mathematical Approach," *Journal of Biomechanics* **38** (4), pp. 943-951.

Celniker G., Gossard D., (1991) "Deformable Curve and Surface Finite-Element for Free Form shape Design," *Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques,* pp. 257-266.

Chace M. A., (1978) "Using DRAM and ADAMS Programs to Simulate Machinery Vehicles," *Agricultural Engineering* **59**(11), 16-18.

Chao E. Y., Morrey B. F., (1978) "Three Dimensional Rotation of the Elbow," *Journal of Biomechanics* **11**, pp. 57-73.

Chao E., (2003) "Graphic Based Musculoskeletal Model for Biomechanical Analyses and Animation," *Medical Engineering & Physics* **25**(3), pp. 201-212.

Chao E., Armiger R., Yoshida H., Lim J., Haraguchi N., (2007) "Virtual Interactive Musculoskeletal System (VIMS) in Orthopaedic Research, Education and Clinical Patient Care," *Journal of Orthopaedic Surgery and Research* **2**(2).

Charlton I., Johnson G., (2001) "Application of Spherical and Cylindrical Wrapping Algorithm in a Musculoskeletal Model of the Upper Limb," *Journal of Biomechanics* **34** (9) pp. 1209-1216.

Chen D., Zelter D., (1992) "Pump it Up: Computer Animation of a Biomechanically Based Model of Muscle Using the Finite Element Model," *ACM SIGGRAPH Computer* Graphic, pp. 89-98.

Chung K., Wang W., (1996) "Quick Collision Detection of Polytopes in Virtual Environments," *Proceedings of ACM Symposium, Virtual Reality Soft. Tech.*

Ciarlet P., (1988) "Mathematical Elasticity," *Three Dimensional Elasticity Series, Studies in Mathematics and its Applications*, Elsevier Science Pub. Co.

Cohen J., Lin M., Manocha D., (1994) "Exact Collision Detection for Interactive Environments," *Proceeding of the 10th Annual Symposium on Computer Geometry.*

Cohen J., Lin M., Manocha D., Ponamgi M., (1995) "I-Collide: An Interactive and Exact Collision Detection system for Large Scale Environments," *Proceedings of ACM Symposium on Interactive 3D Graphics,* pp. 189-196.

Cooper J. E., Shwedyk E., Quanbury A. O., Miller J., Hildebrand D., (1993) "Elbow Joint Restriction: Effect on Functional Upper Limb Motion during Performance of three Feeding Activities," *Archives of Physical Medicine and Rehabilitation* **74** (8), pp. 805-809.

Coquillat S., (1990) "Extended Free-Form Deformation: A Sculptuning Tool for 3D Geometric Modelling," *Computer Graphics* **2**(4), pp. 187-196.

COSMOSMotion
http://highered.mcgrawhill.com/sites/dl/free/0073375330/624525/MotionTutorial2008.pdf

Cottle R., Dantzig G., (1968) "Complementary Pivot Theory on Mathematical Programming," *Linear Algebra and Its Applications* **1**(1), pp. 103-125.

Cottle R., Pang J., Stone R., (1992) "The Linear Complementarity Problem," Academic Press Inc.

Cottle R., Pang J., Stone R., (1996) "Review of the Linear Complementarity Problem," *Linear Algebra and its Applications* **235**, pp. 275-276.

Cowin S., Doty S., (2007) "Tissue Mechanics," 2nd. Edition Springer.

Craig J. J., (1989) "Introduction to Robotics: Mechanics and Control," 2nd Edition,

Addison-Wesley.

Crowninshield R., Pope M., Johnson R., (1976) "An Analytical Model of the Knee," *Journal of Biomechanics* **9**, pp. 397-405.

Dadebo B., White J., George K., (2004) "A Survey of Flexibility Training Protocols and Hamstring Strains in Professional Football Clubs in England," *British Journal of Sports Medicine* **38**, pp. 388-394.

Damsgaard M., Rasmussen J., Christensen S., Surma E., Zee M., (2006) "Analysis of Musculoskeletal systems in the Anybody Modeling System," *Simulation Modelling Practice and Theory* **14**(8), pp. 1100-1111.

Dantzig, G., (1998) "Linear Programming and Extensions," 11th Edition, Princeton University Press.

Davoodi R., Loeb G., (2002) "A Software Toll for Faster Development of Complex Models of Musculoskeletal Systems and Sensorimotor Controllers in Simulink," *Journal of Applied Biomechanics* **18**, pp. 357-365.

De Duca C. J., Forrest W. J., (1973) "Force Analysis of Individual Muscles Acting Simultaneously on the Shoulder Joint during Isometric Abduction," *Journal of Biomechanics* **6**, pp. 385-393.

De Vita R., Slaughter W. S., (2007) "A Constitutive Law for the Failure Behaviour of Medial Collateral Ligaments," *Biomechanics and Modeling in Mechanobiology* **6**(3), pp. 189-197.

Delp S., Loan J., (1995) "A Graphics Based Software System to Develop and analyze Models of Musculoskeletal Structures," *Computers in Biology and Medicine* **25**(1), pp. 21-34.

Delp S., Loan J., (2000) "A Computational Framework for Simulating and Analyzing Human and Animal Movement," *Computing in Science and Engineering*, **2**, pp. 46-55.

Delp S., Loan J., Hoy M., Zajac F., Topp E., Rosen J., (1990) "An Interactive Graphics-Based Model of The Lower Extremity to Study Orthopaedic Surgical Procedures," *IEEE Transactions on Biomedical Engineering* **37**(8), pp. 757-767.

Denavit J., Hartenberg R. S., (1955) "A Kinematic notation for lower-pair mechanisms based on matrices," *Journal of Applied Mechanics* **77**, pp. 215-221.

Dodge G., Poole A., (1989) "Immunohistochemical detection and immunochemical analysis of type II collagen degradation in human normal, rheumatoid, and osteoarthritic articular cartilages and in explants of bovine articular cartilage cultured with interleukin 1," *The Journal of Clinical Investigation* **83**(2), pp. 647-661.

Drapaca C., Sivaloganathan S., Tenti G., (2007) "Nonlinear Constitutive Laws in Viscoelasticity," *Mathematics and Mechanics of Solids*, **12**, 475-501.

Duriez C., Andriot C., Kheddar A., (2004) "Signorini`s Contact Model for Deformable Objects in Haptic Simulations," *IEEE/RSJ International Conferences on Intelligent*

Robots and Systems.

Dvir Z., Berme N., (1978) "The Shoulder Complex in Elevation of the Arm: A Mechanism Approach," *Journal of Biomechanics* **11**, pp. 219-225.

Dvorac J., Antiness J., Panjabi M., et al. (1988) "Age and Gender Related Normal Motion of the Cervical Spine," *Spine* **17**, pp. S393-S398.

Eberharter J. K., Ravani B., (2006) "Kinematic Registration in 3D Using the 2D Reuleaux Method," *Journal of Mechanical Design* **128**, pp. 349-355.

Eberly D., (2003) "Game Physics," Morgan Kaufmann.

Eck C., Jarusek J., Krbec M., (2005) "Unilateral Contact Problems, Variational Methods and Existence Theorems," CRC Press.

Ehmann S., Lin M., (2000) "Accelerated Proximity Queries Between Convex Polyhedra By Multi-Level Voronoi Marching," *IEEE/RSJ International Conferences on Intelligent Robots and Systems*.

Ehmann S., Lin M., (2001) "Accurate and Fast Proximity Queries Between Polyhedra Using Convex Surface Decomposition," *Computer Graphics Forum* **20**, pp. 500-511.

Engin A. E., (1980) "On Biomechanics of the Shoulder Complex," *Journal of Biomechanics* **13**, pp. 575-590.

Engin A. E., Tumer S. T., (1988) "Three Dimensional Kinematic Modelling of the Human Shoulder Complex-Part I: Physical Model and Determination of Joint Sinus Cones," *Journal of Biomechanical Engineering* **111**, pp. 107-112.

Engin A.E., (2001) "Techniques in the Dynamic Modeling of Human Joints with a Special Application to the Human Knee," Book Chapter - Musculoskeletal Models and Techniques **111**(3).

Ericson C., (2005) "Real Time Collision Detection," Elsevier.

Essa I., Sclaroff S., Pentland A., (1992) "Physically Based Modeling for Graphics and Vision," *M.I.T Media Laboratory Perceptual Computing Group Technical Report*.

Essinger J., Leyvraz P., Heegard J., Robertson D., (1989) "A Mathematical Model for the Evaluation of the Behaviour during Flexion of Condylar Type Knee Prosthesis," *Journal of Biomechanics* **22**, pp. 1229-1241.

Fazel-Rezai R., Shwedyk E., Onyshko S., (1997) "Three Dimensional Kinematic Model of the Upper Limb With Ten Degrees of Freedom," *Proceedings of 19th International Conference IEEE/EMBS*.

Featherstone R., (1987) "Robot Dynamics Algorithms," Kluwer Academic Publisher.

Featherstone, R. (2007) "Rigid Body dynamics Algorithms," Springer.

Ferrario V., Sforza C., Serrao G., Grassi G., Mossi E., (2002) "Active Range of Motion

of the Head and Cervical Spine: A Three Dimensional Investigation in Healthy Young Adults," *Journal of Orthopaedic Research* **20**, pp. 122-129.

Fisk J., Wayne J., (2009) "Development and Validation of a Computational Musculoskeletal Model of the Elbow and Forearm," *Annals of Biomedical Engineering* **37**(4), pp. 803-812.

Flores P., Ambrosio J., Claro J. C. P., Lankarani H. M., (2008) "Kinematics and Dynamics of Multi Body Systems with Imperfect Joints," *Lecture Notes in Applied and Computational Mechanics - Springer* **34**.

Fung Y., (1981) "Biomechanics: Mechanical Properties of Living Tissues," 1st Edition, Springer-Verlag USA.

Fung Y., (1993) "Biomechanics: Mechanical Properties of Living Tissues," 2nd Edition, Springer-Verlag USA.

Funk J., Hall G., Crandall J., Pilkey W., (2000) "Linear and Quasi-Linear Viscoelastic Characterization of Ankle Ligaments," *Journal of Biomechanical Engineering* **122,** pp. 15-22.

Gajdosik R., (2001) "Passive Extensibility of Skeletal Muscle: Review of the Literature with Clinical Implications," *Clinical Biomechanics* **16**, pp. 87-101.

GAMMA http://gamma.cs.unc.edu/

Ganovelli F., Dingliana J., O`Sullivan C., (2000) "Bucket Tree: Improving Collision Detection between Deformable Objects," *Spring Conference in Computer Graphics*, pp. 156-163.

Gao F., Damsgaard M., Rasmussen J., Christensen S., (2002) "Computational Method for Muscle-Path Representation in Musculoskeletal Models," *Biological Cybernetics* **87** (3), pp. 199-210.

Garner B., Pandy M., (2000) "The Obstacle Set Method for Representing Muscle Path in Musculoskeletal Models," *Computer Methods in Biomechanics and Biomedical Engineering* **3**(1), pp. 1-30.

Garner B., Pandy M., (2003) "Estimation of Musculotendon Properties in the Human Upper Limb," *Annuals of Biomedical Engineering* **31**, pp. 207-220.

Gatti C., Dickerson C., Chadwick E., Mell A., Hughes R., (2007) "Comparison of Model Predicted and Measured Moment Arms for the Rotator Cuff Muscles," *Clinical Biomechanics* **22** (6), pp. 639-644.

GEOMAGIC www.geomagic.com

Gibson S., Mirtich B., (1997) "A Survey of Deformable Modeling in Computer Graphics," Technical Report, TR-97-19, Mitsubishi Electric Research Laboratories.

Gilbert E., Johnson D., Keerthi S., (1988) "A Fast Procedure for Computing the Distance Between Objects in Three Dimensional Space," *IEEE Transactions Robotics*

and Automation **4**(2), pp. 193-203.

Gordon A., Huxley A., Julian F., (1966) "The Variation in Isometric Tension with Sarcomere Length in Vertebrae Muscle Fibres," *Journal of Physiology* **184**, pp. 170-192.

Gottschalk S., Lin M., Manocha D., (1996) "OBB-Tree: A Hierarchical Structure for Rapid Interference Detection," *30th Annual Conference Series Computer Graphics* (30), pp. 171-180.

Govindaraju N., Kabul I., Lin M., Manocha D., (2007) "Fast Continuous Collision Detection Among Deformable Models using Graphics Processors," *Computers & Graphics* **31**, pp. 5-14.

Han S., Federico S., Epstein M., Herzog W., (2005) "An Articular Cartilage Contact Model Based on Real Surface Geometry," *Journal of Biomechanics* **38**, pp. 179-184.

Harrington I., Seyki-Out A., Barrington T., Evans D., Tuli V., (2001) "The Functional Outcome with Metallic Radial Head Implants in the Treatment of Unstable Elbow Fractures: A Long Term Review," *Journal of Trauma* **50**(1), pp. 46-52.

Hatze H., (1981) "Estimation of Myodynamic Parameter Values from Observations on Isometrically Contracting Muscle Groups," *Applied Physics* **46**, pp. 325-338.

Hieber S., Koumoutsakos P., (2008) "A Lagrangian Particle Method for the Simulation of Linear and Nonlinear Elastic Models of Soft Tissue," *Journal of Computational Physics* **227**, pp. 9195-9215.

Hieber S., Walther J., Koumoutsakos P., (2004) "Remeshed Smoothed Particle Hydrodynamics Simulation of the Mechanical Behaviour of Human Organs," *Technology and Health Care* **12**, pp. 305-314.

Hill A., (1938) "The Heat of Shortening and the dynamic Constants of Muscle," *Proceedings of the Royal Society of London, Series B, Biological Sciences* **126**(843), pp. 136-195.

Hirokawa S., (1991) "Three Dimensional Mathematical Model Analysis of the Patellofemoral Joint," *Journal of Biomechanics* **24**(8), pp. 659-671.

Hirota G., Fisher S., State A., (2003) "An Improved Finite Element Contact Model for Anatomical Simulations," *The Visual Computer* **19**(5), pp. 291-309.

Hirota G., Fisher S., State A., Lee C., Fuchs H., (2001) "An Implicit Finite Element Method for Elastic Solids in Contact," *14th Conference on Computer Animation*, pp. 136-254.

Hogfors C., Karlsson D., Peterson B., (1995) "Structure and Internal Consistency of a Shoulder Model," *Journal of Biomechanics* **28**(7), pp. 767-777.

Hogfors C., Peterson B., Sigholm G., Herberts P., (1991) "Biomechanical Model of the Human Shoulder Joint – II. The Shoulder Rhythm," *Journal of Biomechanics* **24** (8), pp.

699-709.

Hogfors C., Sigholm G., Herberts P., (1987) "Biomechanical Model of the Human Shoulder – I. Elements," *Journal of Biomechanics* **20** (2), pp. 157-166.

Holzapel G., Ogden R., (2005) "Mechanics of Biological Tissue," Springer-Verlag, Heidelberg.

Holzbaur K., Murray W., Gold G., Delp S., (2007) "Upper Limb Muscle Volumes in Adult Subjects," *Journal of Biomechanics* **40**, pp. 742-749.

Hon Y. C., Lu M. W., Xue W. M., Zhou X., (1999) "A New Formulation and Computation of the Triphasic Model for Mechano-electrochemical Mixtures," *Computational Mechanics*, **24**(3), pp. 155-165.

Huang C. Y., Stankiewicz A., Ateshian G. A., Mow V. C., (2005) "Anisotropy, Inhomogenity, and Tension-Compression Nonlinearity of Human Glenohumeral Cartilage in Finite Deformation," *Journal of Biomechanics* **38**(4), pp. 799-809.

Hubbard P., (1995) "Collision Detection for Interactive Graphics Applications," PhD Thesis, Brown University.

Hudson T., Lin M., Cohen J., Gottschalk S., Manocha D., (1997) "V-Collide: Accelerated Collision Detection for VRML," *Proceedings of 2nd Symposium on the Virtual Reality Modeling Language*, pp. 119-125.

Hunt K., Crossley F., (1975) "Coefficient of Restitution Interpreted as Damping in Vibroimpact," *Journal of Applied Mechanics* **7**, pp. 440-445.

Huxley H., (1969) "The Mechanism of Muscular Contraction," *Science* **164** (3886), pp. 1356-1366.

Inman V., Saunders J., Abbott L., (1944) "Observations on the Function of the Shoulder Joint," *Journal of Bone and Joint Surgery Am.* **26**, pp. 1-30.

James D., Pai D., (2004) "BD-Tree Output Sensitive Collision Detection for Reduced Deformable Models," *International Conference on Computer Graphics and Interactive Techniques ACM SIGGRAPH*, pp. 393-398.

Jazar G. N., (2006) "Theory of applied Robotics: Kinematics, Dynamics and Control," Springer.

Jensen R., Davy D., (1975) "An Investigation of Muscle Lines of Action about the Hip: A Centroid line Approach vs the Straight Line Approach," *Journal of Biomechanics* **8**(2), pp. 103-110.

Jensen S., Olsen B., Tyrdal S., Sojbjerg J., Sneppen O., (2005) "Elbow Joint Laxity After Experimental Radial Head Excision and Lateral Collateral Ligament Rupture: Efficacy of Prosthetic Replacement and Ligament Repair," *Journal of Shoulder and Elbow Surgery* **14**(1), pp. 78-84.

Jimenez P., Thomas F., Torras C., (2001) "3D Collision Detection: A Survey,"

Computer and Graphics **25**(2), pp. 269-285.

Johnson K., (1987) "Contact Mechanics," Cambridge University Press.

Kane T. R., Levinson D. A., (1985) "Dynamics: Theory and Applications," McGraw-Hill.

Karlsson D., Peterson B., (1992) "Towards a Model for Force Predictions in the Human Shoulder," *Journal of Biomechanics* **25**, pp. 189-199.

Kaufman K., An K., Chao E., (1989) "Incorporating of Muscle Architecture into the Length-Tension Relationship," *Journal of Biomechanics* **22**, pp. 943-948.

Kim J. H., Abdel-Malek K., Yang J., Marler T., (2006) "Prediction and Analysis of Human Motion Dynamics Performing Various Tasks," *International Journal of Human Factors Modelling and Simulation* **1**(1), pp. 69-97.

Kinzel G. L., Hall A. S., Hillberry B. M., (1972) "Measurement of the Total Motion between Two Body Segments – I. Analytical Development," *Journal of Biomechanics* **5**, pp. 93-105.

Kinzel G. L., Hillberry B. M., Hall A. S., (1972) "Measurement of the Total Motion between Two Body Segments – II. Description of Application," *Journal of Biomechanics* **5**, pp. 283-293.

Klosowski J., Held M., Mitchell J., Sowizral H., Zikan K., (1998) "Efficient Collision Detection Using Bounding Volume Hierarchies of k-DOPs," *IEEE Transactions on Visualization and Computer Graphics* **4**(1), pp. 21-36.

Krogt M., Doorenbosch C., Harlaar J., (2007) "Muscle Length and Lengthening Velocity in Voluntary Crouch Gait," *Gait & Posture* **26**(4), pp. 532-538.

Kuhnapfel U., Cakmak H., Maab H., (2000) "Endoscopic Surgery Training Using Virtual Reality and Deformable Tissue Simulation," *Computers & Graphics* **24**, pp. 671-682.

Lankarani H, Nikravesh P., (1990) "A Contact force Model with Hysteresis Damping for Impact Analysis of Multibody Systems," *Journal of Mechanical Design* **112**(3), pp. 369-376.

LAPACK http://www.netlib.org/lapack/

Larrson T., Moller T., (2006) "A Dynamic Bounding Volume Hierarchy for Generalized Collision Detection," *Computers & Graphics* **30**(3), pp. 450-459.

Laursen B., Jensen B., Nemeth G., Sjogaard G., (1998) "A Model Predicting Individual Shoulder Muscle Forces Based on Relationship between Electromyographic and 3D External Forces in Static Position," *Journal of Biomechanics* **31**, pp. 731-739.

Lazarus F., Coquillart S., Jancene P., (1993) "Interactive Axial Deformations," *IFIP Working Conference on Geometric Modeling and Computer Graphics*, pp. 241-254.

Lemke C., (1968) "On Complementarity Pivot Theory," *Mathematics of Design Sciences,* pp. 95-114.

Lemos R., Epstein M., Herzog W., Wyvill B., (2001) "Realistic Skeletal Muscle Deformation Using Finite Element Analysis," *14th Brazilian Symposium on computer Graphics and Image Processing,* pp. 192-199.

Lemos R., Rokne J., Baranoski G., Kawakami Y., Kurihara T., (2005) "Modeling and simulation the Deformation of Human Skeletal Muscle Based on Anatomy and Physiology," *Computer Animation and Virtual Worlds* **16**, pp. 319-330.

Li G., Park S., DeFrate L., Schutzer M., Ji L., Gill T., Rubash H., (2005) "The Cartilage Thickness Distribution in the Tibiofemoral Joint and Its Correlation with Cartilage to Cartilage Contact," *Clinical Biomechanics* **20**(7), pp. 736-744.

Liew V., Cooper I., Ferreira L., Johnson J., King G., (2003) "The Effect of Metallic Radial Head Arthroplasty on Radiocapitellar Joint Contact Area," *Clinical Biomechanics* **18**, pp. 115-118.

LIFEMOD www.lifemodeler.com

Lin H., Nakamura Y., Su F., Hashimoto J., Nobuhara K., Chao E., (2005) "Use of Virtual, Interactive, Musculoskeletal System (VIMS) in Modeling and Analysis of Shoulder Throwing Activity," *Journal of Biomechanical Engineering* **127**, pp. 525-530.

Lin M., Gottschalk S., (1998) "Collision Detection between Geometric Models: A Survey," *In Proceedings of IMA Conference on Mathematics of Surfaces*.

Lin M., Manocha D., (2003) "Collision and Proximity Queries," *Handbook of Discrete and Computational Geometry: Collision detection*.

Ling Z., Guo H., Boersma S., (1997) "Analytical Study on the Kinematic and Dynamic Behaviours of a Knee Joint," *Medical Engineering & Physics* **19**(1), 29-36.

Lombardo J., Cani M., Neyret F., (1999) "Real Time Collision Detection for Virtual Surgery," *Proceedings of the Computer Animation*, 82.

Loocke M., Lyons C., Simms C., (2006) "A Validated Model of Passive Muscle in Compression," *Journal of Biomechanics* **39**, pp. 2999-3009.

Loocke M., Lyons C., Simms C., (2008) "Viscoelastic Properties of Passive Skeletal Muscle in Compression: Stress-Relaxation Behaviour and Constitutive Modelling," *Journal of Biomechanics* **41**, pp. 1555-1566.

MADYMO http://www.madymo.com/cms/index.php?pageid=131

Magermans D. J., Chadwick E. K. J., Veeger H. E. J.,. Van Der Helm F. C. T, (2005) "Requirements for Upper Extremity Motion During Activities of Daily Living," *Clinical Biomechanics* **20**, pp. 591-599.

Marai *G., (2007) "Data Driven Predictive Modeling of Diarthrodial Joints,"* PhD

Thesis, Brown University.

Marai G., Laidlaw D., Demiralp D., Andrews C., Grimm C., Crisco J., (2003) "Estimating Joint Contact Areas and Ligament Lengths From Bone Kinematics and Surfaces," *IEEE Trans. on Biomedical Engineering,* **51**(5), pp. 790-799.

Marchal D., Aubert F., Chaillou C., (2004) "Collision Between Deformable Objects Using Fast-Marching on Tetrahedral Models," *Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 1-19.

Marsden S., Swailes D., (2008) "A Novel Approach to the Prediction of Musculotendon Paths," *Journal of Engineering in Medicine* **222** (1), pp. 51-61.

Martini F., (1998) "Fundamentals of Anatomy and Physiology," 1th Edition, Prentice Hall.

Maurel W., (1999) "3D Modelling of the Human Upper Limb including the Biomechanics of Joints, Muscles and Soft Tissues," PhD Thesis, Ecole Polytechnique Federale De Lausanne.

Maurel W., Thalmann D., (2000) "Human Shoulder Modelling Including Scapulo-Thoracic Constraint and Joint Sinus Cones," *Computers & Graphics* **24** (2), pp. 203-218.

McHugh M., Tyler T., Greenberg S., Gleim G., (2002) "Differences in Activation patterns between Eccentric and Concentric Quadriceps Contractions," *Journal of Sports Sciences* **20**(2), pp. 83-91.

Merrill T., Goldsmith W., Deng Y., (1984) "Three Dimensional Response of a Lumped Parameter Head Neck Model Due to Impact and Impulsive Loading," *Journal of Biomechanics* **17**(2), pp. 81-95.

Mezger J., Kimmerle S., Etzmub O., (2003) "Hierarchical Techniques in Collision Detection for Cloth Animation," *Journal of WSCG* **11**(2), pp. 322-329.

Miller G., (1988) "The Motion Dynamics of Snakes and Worms," *Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques,* pp. 169-173.

MIMICS www.materialise.com

Mirtich B., (1996 a) "Fast and Accurate Computation of Polyhedral Mass Properties," *Journal of Graphics Tools* **1**(2), pp. 31-50.

Mirtich B., (1998) "V-Clip: Fast and Robust Polyhedral Collision Detection," *ACM Transactions on Graphics* **17**(3), pp. 177-208.

Mirtich, B. (1996 b) "Impulse-Based Dynamic Simulation of Rigid Body Systems," PhD Thesis, Arizona State University.

Mirzayan R., (2006) "Cartilage Injury in the Athlete," Theime.

Moeinzadeh M., Engin A.E., Akkas N., (1983) "Two Dimensional Dynamic Modelling of Human Knee Joint," *Journal of Biomechanics* **16**(4), pp. 253-264.

Moore K. L., Dalley A.F., Agur A.M.R., (2006) "Clinically Oriented Anatomy," 5th Edition, Lippincott Williams & Wilkins.

Moore M., Wilhelms J., (1988) "Collision Detection and Response for Computer Animations," *Proc. of the 15th Ann. Conference on Computer Graphics and Interactive Techniques*, pp. 289-298.

Moravec H., (1998) "Robot: Mere Machine to Transcendent Mind," Oxford University Press.

Moreau J., Panagiotopoulos P., (1988) "Nonsmooth Mechanics and Applications," Springer.

Moroney S., Schultz A., Miller J., Andersson G., (1988) "Load-Displacement Properties of Lower Cervical Spine Motion Segments," *Journal of Biomechanics* **21**(9), pp. 769-779.

Morrey B. F., Chao E. Y., (1976) "Passive motion of the Elbow Joint," *Journal of Bone and Joint Surgery Am.* **58** (4), pp. 501-508.

Morrey B., Askew L., Chao E., (1981) "Silastic Prosthetic Replacement for the Radial Head," *Journal of Bone and Joint Surgery Am.* **63**, pp. 454-458.

Mow V. C., Kuei S. C., Lai W. M., Armstrong C. G., (1980) "Biphasic Creep and Stress Relaxation of Articular Cartilage: Theory and Experiments," *Journal of Biomechanical Engineering*, **102**(1), pp. 73-84.

Mow V. C., Ratcliffe A., Poole A. R., (1992) "Cartilage and Diarthrodial Joints as Paradigm for Hierarchical Materials and Structures," *Biomaterials* **13**, 67-97.

Mow V., Guo X., (2002) "Mechano-Electrochemical Properties of Articular Cartilage: Their Inhomogenities and Anisotropies," *Annual Review of Biomedical Engineering* **4**, pp. 175-209.

Muller M., Charypar D., Gross M., (2003) "Particle Based Fluid Simulation for Interactive Applications," *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 154-159.

Muller M., Heidelberger B., Teschner M., Gross M., (2005) "Meshless Deformations Based on Shape Matching," *ACM Transactions on Graphics* (TOG), pp. 471-478.

Murray W., Arnold A., Salinas S., Durbhakula M., Buchanan T., Delp S., (1998) "Building Biomechanical Models Based on Medical Image Data: An Assessment of Model Accuracy," Book Chapter-Medical Image Computing and Computer-Assisted Interventation **1496**, 539-549.

Murray W., Buchanan T., Delp S., (2002) "Scaling of Peak Moment Arms of Elbow Muscles with Upper Extremity Bone Dimensions," *Journal of Biomechanics* **35**(1) pp.

19-26.

Murray W., Delp S., Buchanan T., (1995) "Variation of Muscle Moment Arms with Elbow and Forearm Position," *Journal of Biomechanics* **28**(5), pp. 513-525.

Murty K., (1988) "Linear Complementarity, Linear and Nonlinear Programming," Heldermann.

Nedel L., Thalmann D., (1998) "Modeling and Deformation of the Human Body Using an Anatomically Based Approach," *Proceedings of the Computer Animation*.

Neidhard-Doll A., Phillips C., Repperger D., Reynolds D., (2004) "Biomimetic Model of Skeletal Muscle Isometric Contaction: II. A Phenomenological Model of the Skeletal Muscle Excitation-Contraction Coupling Process," *Computers in Biology and Medicine* **34**(4), pp. 323-344.

Nikravesh P. E., (1988) "Computer-aided Analysis of Mechanical Systems," Prentice Hall, Englewood Cliffs.

Nishimura H., Hirai M., Kawai T., Kawata T., Shirakawa I., Omura K., (1985) "Object Modeling by Distribution Function and a Method of Image Generation," Transactions IECE **J68-D** (4), pp. 718-725.

NJR http://www.njrcentre.org.uk/NjrCentre/

Novotny J., Beynnon B., Nichols C., (2000) "Modelling the Stability of the Human Glenohumeral Joint during External Rotation," *Journal of Biomechanics* **33**, pp. 345-354.

O`Driscoll S., Horii E., Morrey B., Carmichael S., (1992) "Anatomy of the Ulnar Part of the Lateral Collateral Ligament of the Elbow," *Clinical Anatomy* **5**, pp. 296-303.

Ogden R., (1984) "Non-Linear Elastic Deformations," Ellis Horwood New York.

Orlandea N., Chace M. A., Calahan D. A., (1977) "A Sparsity Oriented Approach to the Dynamic Analysis and Design of Mechanical Systems- Part I," *Journal of Engineering for Industry* **99**, pp. 773-784.

Otto J., Callaghan J., Brown T., (2001) "Mobility and Contact Mechanics of a Rotating Platform Total Knee Replacement," *Clinical Orthopaedics and Related Research* **392**, pp. 24-37.

Panagiotopoulos P., (1975) "A Nonlinear Programming Approach to the Unilateral Contact, And Friction, Boundary Value Problem in the Theory of Elasticity," *Archive of Applied Mechanics* **44**(6), pp. 421-432.

Pandy M., (1999) "Moment Arm of a Muscle Force," *Exercise & Sport Sciences Reviews*, **27**, pp. 79-118.

Pandy M., Sasaki K., Kim S., (1997) "A Three Dimensional Musculoskeletal Model of the Human Knee Joint. Part 1: Theoretical Construction," *Computer Methods in*

Biomechanics and Biomedical Engineering **1**(2), pp. 87-108.

Pang J., Trinkle J., (1996) "Complementarity Formulations and Existence of Solutions of Dynamic Multi Rigid Body Contact Problems with Coulomb Friction," *Mathematical Programming* **73**(2), pp. 199-226.

Penning I., (1978) "Normal Movement of the Cervical Spine," *American Journal of Roentgenology*, **130**(2), pp. 317-326.

Pentland A., Williams J., (1989) "Good Vibrations: Modal Dynamics for Graphics and Animation," *Proceedings of ACM SIGGRAPH conference*, pp. 207-219.

Peterson B., Palmerud G., (1996) "Measurement of Upper Extremity orientation by Video Stereometry System," *Medical & Biological Engineering & Computing* **34** (2), pp. 149-154.

Petuskey K., Bagley A., Abdala E., James M. A., Rab G., (2007) "Upper Extremity Kinematics during Functional Activities: Three Dimensional Studies in a Normal Pediatric Population," *Gait & Posture* **25**, pp. 573-579.

Pfeiffer F., Glocker C., (1996) "Multibody Dynamics with Unilateral Contacts," Wiley Series in Nonlinear Science.

Piazza S., Delp S., (2001) "Three Dimensional Dynamic Simulation of Total Knee Replacement Motion during a Step Up Task," *Journal of Biomechanical Engineering* **123**, pp. 599-606.

Pinchuk L., Nikolaev V., Tsvetkova E., Goldade V., (2005) "Tribology and Biophysics of Artifical Joints," Elsevier Science.

Pintar F., (1986) "The Biomechanics of Spinal Elements," PhD Thesis, Marquette University.

Poole A., Kobayashy M., Yasuda T., Mwale F., Kojima T., Sakai T., Wahl C., El-Maadawy S., Webb G., Tchetina E., Wu W., (2002) "Type II Collagen Degradation and its Regulation in Articular Cartilage in Osteoarthritis," *Ann Rheum Dis* **61**, 78-81.

Poppen N. K., Walker P. S., (1976) "Normal and Abnormal Motion of the Shoulder," *Journal of Bone and Joint Surgery* **58**, pp. 195-201.

Provot X., (1995) "Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behaviour," In Proceedings of Graphics Interface, pp. 147-154.

Quinlan S., (1994) "Efficient Distance Computation between Non-Convex Objects," *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 3324-3329.

Rab G., Petuskey K., Bagley A., (2002) "A method for determination of Upper Extremity Kinematics," *Gait and Posture* **15**, pp. 113-119.

Raikova R., (1992) "A General Approach for Modelling and Mathematical

Investigation of the Human Upper Limb," *Journal of Biomechanics* 25(8), pp. 857-867.

Rawlinson J., Bartel D., (2002) "Flat Medial-Lateral Conformity in Total Knee Replacements Does Not Minimise Contact Stresses," *Journal of Biomechanics* **35**, pp. 27-34.

RECURDYN http://functionbay.co.kr

Redon S., Kim Y., Lin M., Manocha D., (2004) "Fast Continuous Collision Detection for Articulated Models," *Proceedings of ACM Symposium on Solid Modeling and Applications*.

Refaat M., Meguid S., (1995). "On the Modeling of Frictional Contact Problems Using Variational Inequalities," *Finite Elements in Analysis and Design* **19**(1-2), pp. 89-101

Regan W., Korinek S., Morrey B., An K., (1991) "Biomechanical Study of Ligaments around the Elbow Joint," *Clinical Orthopaedics and Related Research* (271), pp. 170-179.

Rho J., Kuhn-Spearing L., Zioupos P., (1998) "Mechanical Properties and the Hierarchical Structure of Bone," *Medical Engineering & Physics* **20**, pp. 92-102.

Riggs B. L., Melton L. J., (1995) "The Worldwide Problem of Osteoporosis: Insights Afforded by Epidemiology," *Bone* **17**(5), pp. S505-S511.

Rodriguez-Navarro J., Susin A., (2006) "Non Structured Meshes for Cloth GPU Simulation using FEM," *3rd Workshop in Virtual Reality, Interactions, and Physical Simulations*, pp. 1-7.

Safaee-Rad R., Shwedyk E., Quanbury A. O., Cooper J. E., (1990) "Normal Functional Range of Motion of Upper Limb Joints During Performance of Three Feeding Activites," *Archives of Physical Medicine and Rehabilitation* **71**(7), pp. 505-509.

Saladin K. S., (2007) "Anatomy & Physiology: The Unity of Form and Function," 4th Edition, McGraw-Hill.

Samet H., (1989) "The Design and Analysis of Spatial Data Structures," Addison-Wesley.

Sathasivam S., Walker P., (1998) "Computer Model to Predict Subsurface Damage in Tibial Inserts of Total Knees," *Journal of Orthopaedic Research* **16**, pp. 564-571.

Scheepers F., Parent R., Carlson W., May S., (1997) "Anatomy-Based Modeling of the Human Musculature," *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 163-172.

Schneider E., Goldhahn J., Burckhardt P., (2005) "The Challenge: Fracture Treatment in Osteoporotic Bone," *Osteoporos International* **16**, pp. S1-S2.

Sclaroff S., Pentland A., (1991) "Generalized Implicit Functions for Computer Graphics," *ACM SIGGRAPH Computer Graphics*, pp. 247-250.

Sederberg T., Parry S., (1986) "Free-Form Deformation of Solid Geometric Models," *SIGGRAPH* **20**(4).

Setton L. A., Zhu W., Mow V., (1993) "Biphasic Poroviscoelstic Behaviour of Articular Cartilage: Role of the Surface Zone in Governing the Compressive Behaviour," *Journal of Biomechanics* **26**(4/5), pp. 581-592.

Setton L., Tohyama H., Mow V., (1998) "Swelling and Curling Behaviour of Articular Cartilage," *Journal of Biomechanical Engineering* **120**, pp. 355-361.

Sforza C., Grassi G., Fragnito N., Turci M., Ferrario V., (2002) "Three Dimensional Analysis of Active Head and Cervical Spine Range of Motion: Effect of Age in Healthy Male Subjects," *Clinical Biomechanics* **17**, pp. 611-614.

Shaffer C., Herb G., (1992) "A Real Time Robot Arm Collision Avoidance System," *IEEE Transactions on Robotics and Automation* **8**(2), pp. 149-160.

Shen J., Thalmann D., (1995) "Interactive Shape Design Using Metaballs and Splines," *Proceedings of Implicit Surfaces* **95**, pp. 187-196.

Siemienski A., Hogfors C., Karlsson D., Peterson B., Makhsous M., Kadefors R., (1995) "Towards an Experimental Validation of a Shoulder Model," *Biomechanics Seminar* **9**, pp. 87-104.

Spoor C., Van Leeuwen J., Meskers C., Titulaer A., Huson A., (1990) "Estimation of Instantaneous moment arms of Lower-Leg Muscles," *Journal of Biomechanics* **23** (12), pp. 1247-1259.

Stavlas P., Jensen S., Sojbjerg J., (2007) "Kinematics of the Ligamentous Unstable Elbow Joint after Application of a Hinged External Fixation Device: A Cadaveric Study," *Journal of Shoulder and Elbow Surgery* **16**(4), pp. 491-496.

Stavroulaki M., Stavroulakis G., (2002) "Unilateral Contact Applications Using FEM Software," *International Journal of Applied Mathematics and Computer Science* **12**(1), pp. 115-125.

Stronge W., (2004) "Impact Mechanics," Cambridge University Press.

Taylor W., Duda G., Heller M., (2009) "A Comparison of Techniques for Fixation of the Quadriceps Muscle-Tendon Complex for in vitro Biomechanical Testing of the Knee Joint in Sheep," *Medical Engineering and Physics* **31**, pp. 69-75.

Terzopoulos D., Fleischer K., (1988) "Modeling Inelastic Deformation: Viscoelasticity, Plasticity, Fracture," *Computer Graphics (SIGGRAPH Proceedings)*, pp. 269-278.

Terzopoulos D., Platt J., Barr H., (1987) "Elastically Deformable Models," *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 205-214.

Terzopoulos D., Waters K., (1990) "Physically Based Facial Modelling, Analysis, and Animation," *The Journal of Visualization and Computer Animation* **1**(2), pp. 73-80.

Teschner M., Heidelberger B., Muller M., Gross M., (2004) "A Versatile and Robust Model for Geometrically Complex Deformable Solids," *Proceedings of the Computer Graphics International*, pp. 312-319.

Teschner M., Kimmerle S., Heidelberger B., Zachmann G., Raghupathi L., Fuhrmann A., Cani M., Faure F., Thalmann N., Strasser W., Volino P., (2005) "Collision Detection for Deformable Objects," *Computer Graphics Forum* **24**(1), pp. 61-81.

Thalmann D., Boulic R., Huang Z., Noser H., (1995) "Virtual and Real Humans Interacting in the Virtual World," *International Conference on Virtual Systems and Multimedia* **95**, pp. 48-57.

Thalmann N., Cordier F., (2000) "Construction of a Human Topological Model from Medical Data," *IEEE Transactions on Information Technology in Biomedicine* **4**(2), pp. 137-149.

Thalmann N., Thalmann D., (2004) "Handbook of Virtual Humans," John Wiley & Sons.

Timoshenko S., Goodier J., (1970) "Theory of Elasticity," McGraw Hill.

Tortora G., Derrickson B., (2009) "Principles of Anatomy and Physiology: Organization, Support and Movement and Control systems of the Human Body," John Wiley and Sons.

Trail I. A., Nuttall D., Stanley J. K., (1999) "Survivorship and Radiological Analysis of the Standard Souter-Strathclyde Total Elbow Arthroplasty," *Journal of Bone and Joint Surgery Br.* **81**(1), pp. 80-84.

Trinkle J., (2003) "Formulation of Multibody Dynamics as Complementarity Problems," *Proceedings of DETC`03 ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pp. 2-6.

Tschoegl N., (1989) "The Phenomenological Theory of Linear Viscoelastic Behaviour," Springer-Verlag Berlin.

Tumer S. T.,. Engin A. E, (1989) "Three Dimensional Kinematic Modelling of the Human Shoulder Complex-Part II: Mathematical Modelling and Solution via Optimization," *Journal of Biomechanical Engineering* **111**, pp. 113-121.

Van Andel C. J., Wolterbeek N., Doorenbosch C. A., Veeger D. H., Harlaar J., (2008) "Complete 3D Kinematics of Upper Extremity Functional Tasks," *Gait & Pasture* **27** (1), pp. 120-127.

Van Den Bergen G., (1998) "Efficient Collision Detection of Complex Deformable Models Using AABB Trees," *Journal of Graphics Tools* **2**(4), pp. 1-13.

Van Den Bergen G., (1999) "A Fast and Robust GJK Implementation for Collision Detection of Convex Objects," *Journal of Graphics Tools* **4**(2), pp. 2-25.

Van Der Bogert A., (1994) "Analysis and Simulation of Mechanical Loads on the Human Musculoskeletal System: A Methodological Overview," *Exercise and Sport*

Sciences Reviews **22**, pp. 23-51.

Van Der Helm F., (1994 a) "Analysis of the Kinematic and Dynamic Behaviour of the Shoulder Mechanism," *Journal of Biomechanics* **27**(5), pp. 527-550.

Van Der Helm F., (1994-b) "A Finite Element Musculoskeletal Model of the Shoulder Mechanism," *Journal of Biomechanics* **27**, pp. 551-569.

Van Der Helm F., Veeger H., Pronk G., Van Der Woude L., Rozendal R., (1992) "Geometry Parameters for Musculoskeletal Modelling of the Shoulder System," *Journal of Biomechanics* **25**(2), pp. 129-144.

Veeger H. E. J., (2000) "The Position of the Rotation Center of the Glenohumeral Joint," *Journal of Biomechanics* **33**12), pp. 1711-1715.

Viidik A., (1990) "Structure and Function of Normal and Healing Tendons and Ligaments," *Biomechanics of Diarthrodial Joints,* Springer, pp. 3-38.

Viidik A., Ekholm R., (1968) "Light and Electron Microscopic Studies of Collagen Fibers under Strain," *Anatomy and Embryology*, **127**(2), pp. 154-164.

Volino P., Courchesne M., Thalmann N., (1995) "Versatile and Efficient Techniques for Simulating Cloth and Other Deformable Objects," *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, pp. 137-144.

Wang X., Maurin M., Mazet F., De Castro Maia N., Voinot K., Pierre J., Fayet M., (1998) "Three Dimensional Modelling of the Motion Range of Axial Rotation of the Upper Limb," *Journal of Biomechanics* **31**, pp. 899-908.

Weiss J., Gardiner J., (2001) "Computational Modeling of Ligament Mechanics," *Clinical Reviews in Biomedical Engineering*, **29** (4), pp. 1-70.

White A., Panjabi M., (1978) "Clinical Biomechanics of the Spine," Lippincott.

White A., Panjabi M., (1990) "Clinical Biomechanics of the Spine," J. B. Lippincott Company.

Wilhelms J., (1994) "Modeling Animals with Bones, Muscles, and Skin," Technical-Report Santa Cruz -USA.

Wilhelms J., Van Gelder A., (1997) "Anatomically Based Modeling," *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 173-180.

Williams S., Schmidt R., Disselhorst-Klug C., Rau G., (2006) "An Upper Body Model for the Kinematic Analysis of the Joint Chain of the Human Arm," *Journal of Biomechanics* **39**, pp. 2419-2429.

Wismans J., Veldpaus F., Janssen J., (1980) "A Three Dimensional Mathematical Model of the Knee Joint," *Journal of Biomechanics* **13**, pp. 677-684.

Witkin A., Fleischer K., Barr A., (1987) "Energy Constraints on Parameterized

Models," *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 225-232.

Woltring H., Long K., Osterbauer P., Fuhr A., (1994) "Instantaneous Helical Axis Estimation from 3D Video Data in Neck Kinematics for Whiplash Diagnostics," *Journal of Biomechanics* **27**(12), pp. 1415-1425.

Woo S., Gomez M. A., Seguchi Y., Endo C. M., Akeson W. H., (1983) "Measurement of Mechanical Properties of Ligament Substance from Bone-Ligament-Bone Preparation," *Journal of Orthopaedic Research* **1**(1), pp. 22-29.

Wu G., Cavanagh P. R., (1995) "ISB Recommendations for Standardization in the Reporting of Kinematic Data," *Journal f Biomechanics* **28**(10), pp. 1257-1261.

Wu G., Van Der Helm F. C. T., Veeger H. E. J., Makhsous M., Van Roy P., Anglin C., Nagels J., Karduna A. R., McQuade K., Wang X., Werner F. W., (2005) "ISB Recommendation on Definitions of Joint Coordinate Systems of Various Joints for the Reporting of Human Joint Motion – Part II: Shoulder Elbow, Wrist and Hand," *Journal of Biomechanics* **38**, pp. 982-992.

Wu Y., Thalmann N., Thalmann D., (1995) "A dynamic Wrinkle Model in Facial Animation and Skin Aging," *The Journal of Visualization and Computer Animation* **6**(4), pp. 195-205.

Wuelker N., Wirth C., Plitz W., Roetman B., (1995) "A Dynamic Shoulder Model: Reliability Testing and Muscle Force Study," *Journal of Biomechanics* **28**, pp. 489-499.

Wyvill G., McPheeters C., Wyvill B., (1986) "Data Structure for Soft Objects," *The Visual Computer* **2**(4), pp. 227-234.

Xie K., Yang J., Zhu Y., (2007) "Fast Collision Detection Based on Nose Augmentation Virtual Surgery," *Computer Methods and Programming in Biomedicine* **88**(1), pp. 1-7.

Yoganandan N., Kumaresan S., Pintar F., (2001) "Biomechanics of the Cervical Spine Part 2. Cervical Spine Soft Tissue Responses and Biomechanical Modeling," *Clinical Biomechanics* **16**, pp. 1-27.

Yoshimoto S., (1992) "Ballerinas Generated by a Personal Computer," *The Journal of Visualization and Computer Animation* **3**, pp. 85-90.

Youm Y., McMurthy R. Y., Flatt A. E., Gillespie T. E., (1978) "Kinematics of the Wrist: I. an Experimental Study of Radial-Ulnar Deviation and Flexion-Extension," *Journal of Bone and Joint Surgery* **60**(4), pp. 423-431.

Zachmann G., (1998) "Rapid Collision Detection by Dynamically Aligned DOP-Trees," *Proceedings of IEEE Virtual Reality Annual International Symposium*, pp. 90-97.

Zajac F., (1989) "Muscle and Tendon: Properties, Models Scaling and Application to Biomechanics and Motor Control," *CRC Critical Reviews in Biomedical Engineering* **17**, pp. 359-411.

Zajac F., Topp E., Stevenson P., (1986) "A Dimensionless Musculotendon Model," 8th

Annual Conference IEEE Engineering in Medicine and Biology Society, pp. 601-604.

Zatsiorsky V. M., (1997) "Kinematics of Human Motion," Human Kinetics Publishers.

Zelter D., (1982) "Motor Control Techniques for Figure Animation," *Computer Graphics and Applications* **2** (9), pp. 53-59.

Zhu Q., Chen Y., Kaufman A., (1998) "Real Time Biomechanically Based Muscle Volume Deformation using FEM," *Computer Graphics Forum* **190**(3), pp. 275-284.

Zhu S., Zwiebel S., Barnhardt G., (1999) "A theoretical Formula for Calculating Damping in the Impact of Two Bodies in a Multibody System," *Journal of Mechanical Engineering Science* **213**(3), pp. 211-216.

Appendix

A1 Human skeletal bones and joints

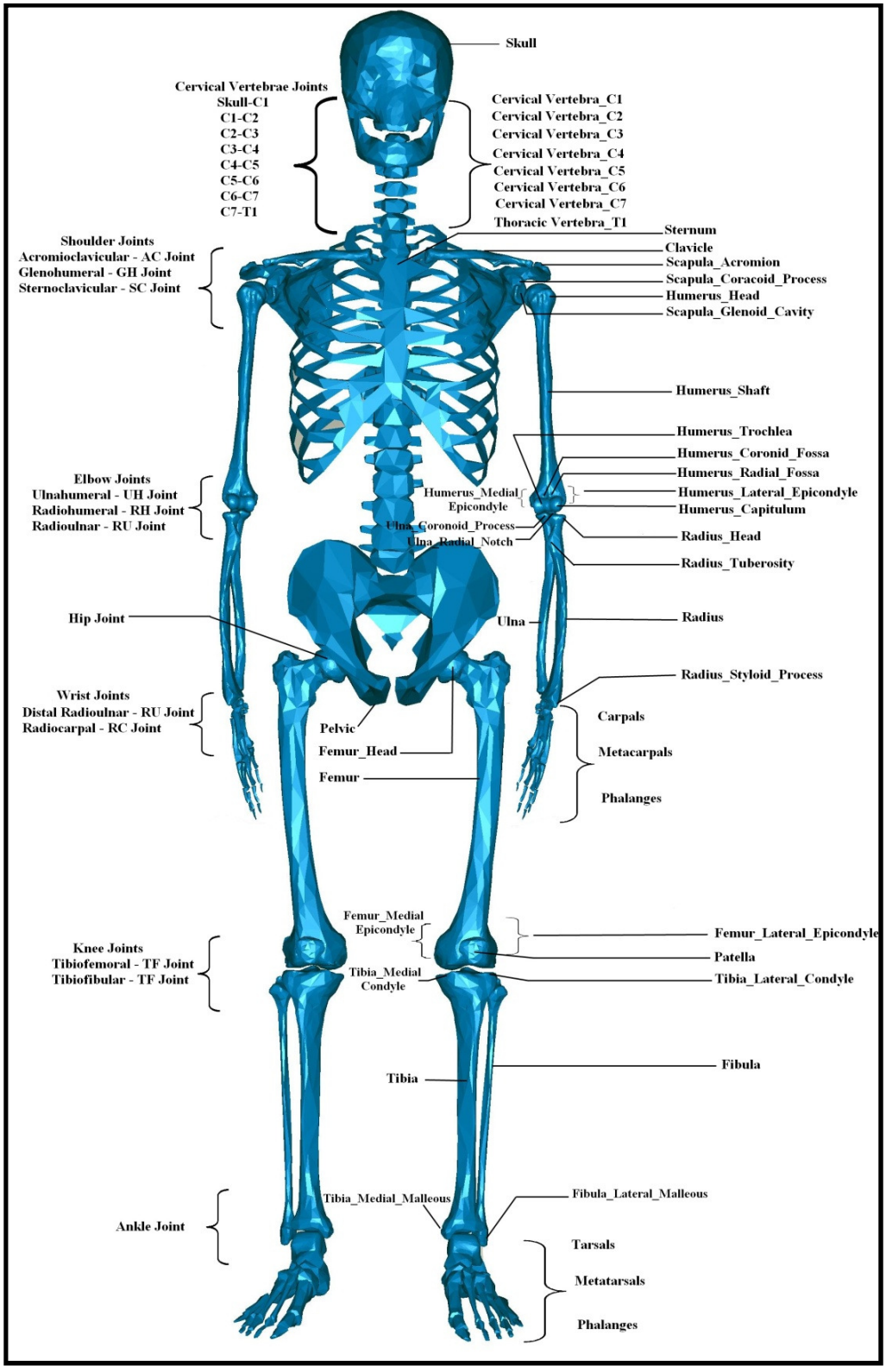


Figure A1 Human skeletal bones and joints

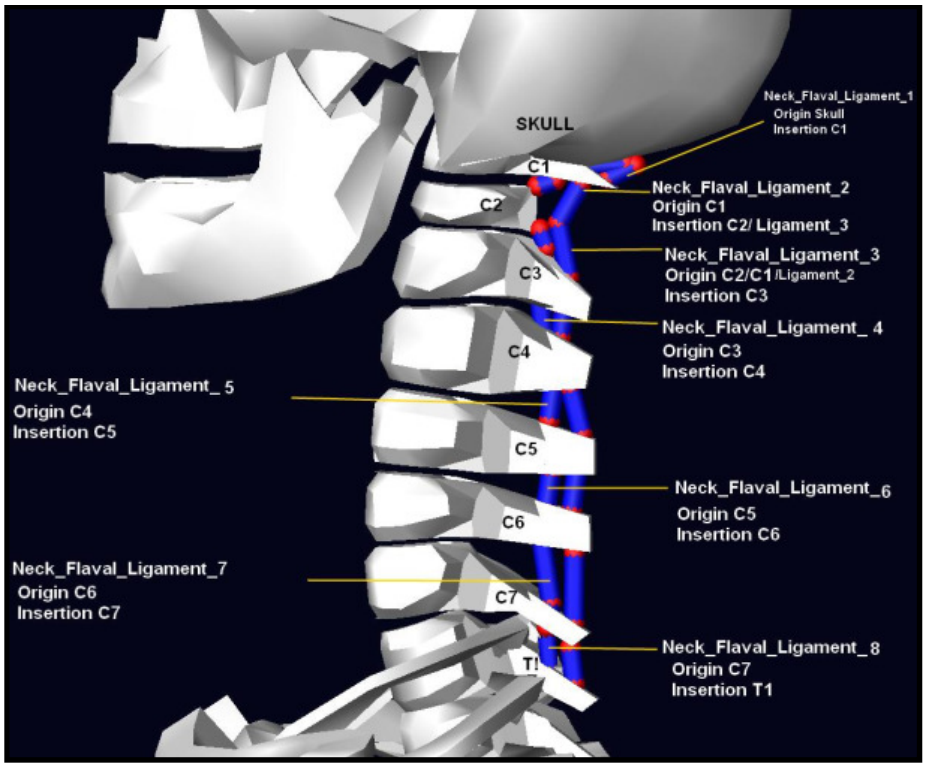
Table A1.1 Major human skeletal joints and common joint mobility descriptions

Place	Bone	Joint Type	Common DOF
Hand	DP (Distal Phalanges) IP (Intermediate Phalanges)	Interphalangeal Joint (Hinge Joint)	1
	IP (Intermediate Phalanges) PP (Proximal Phalanges)	Interphalangeal Joint (Hinge Joint)	1
	PP (Proximal Phalanges) M (Metacarpals)	Metacarpopalangeal (Arthrodial Joint)	3
	M (Metacarpals) C (Carpals)	Carpometacarpal (Thump) Joint (Saddle Joint)	2
Foot	DP (Distal Phalanges) IP (Intermediate Phalanges)	Interphalangeal Joint (Hinge Joint)	1
	IP (Intermediate Phalanges) PP (Proximal Phalanges)	Interphalangeal Joint (Hinge Joint)	1
	PP (Proximal Phalanges) M (Metatarsals)	Metatarsopalangeal (Arthrodial Joint)	3
	M (Metatarsals) T (Tarsals)	Tarsometatarsal Joint (Saddle Joint)	2
Elbow	H (Humerus) U (Ulna)	Humeroulnar Joint (Hinge Joint)	1
	H (Humerus) R (Radius)	Humeroradial Joint (Hinge (Ginglimus) Joint)	1
	U (Ulna) R (Radius)	Radioulnar Joint (Pivot (rotatorius) Joint)	1
Knee	Fe (Femur) T (Tibia)	Tibiofemoral Joint (Condylar Joint)	2
	T (Tibia) Fi (Fibula)	Proximal Tibiofibular Joint (Synavial Joint)	3 plane
Shoulder	HH (Humeral Head) G (Glenoid Fossa) S (Scapula)	Glenohumeral Joint (Ball-and-Socket Joint)	3
Hip	P (Pelvis) A (Acetabulum) FH (Femoral Head)	Femoral Acetabulum Joint (Ball-and-Socket Joint)	3
Wrist	R (Radius) C (Carpal)	Radiocarpal Joint (Ellipsoid Joint)	2
Ankle	Ti (Tibia) Ta (Talus)	Tibiotalar Joint (Hinge Joint)	1
Skull	FB (Frontal Bone) PB (Parletal Bone) SB (Sphenold Bone) TB (Temporal Bone)	Scull Type Joints (Synartroses joints)	0

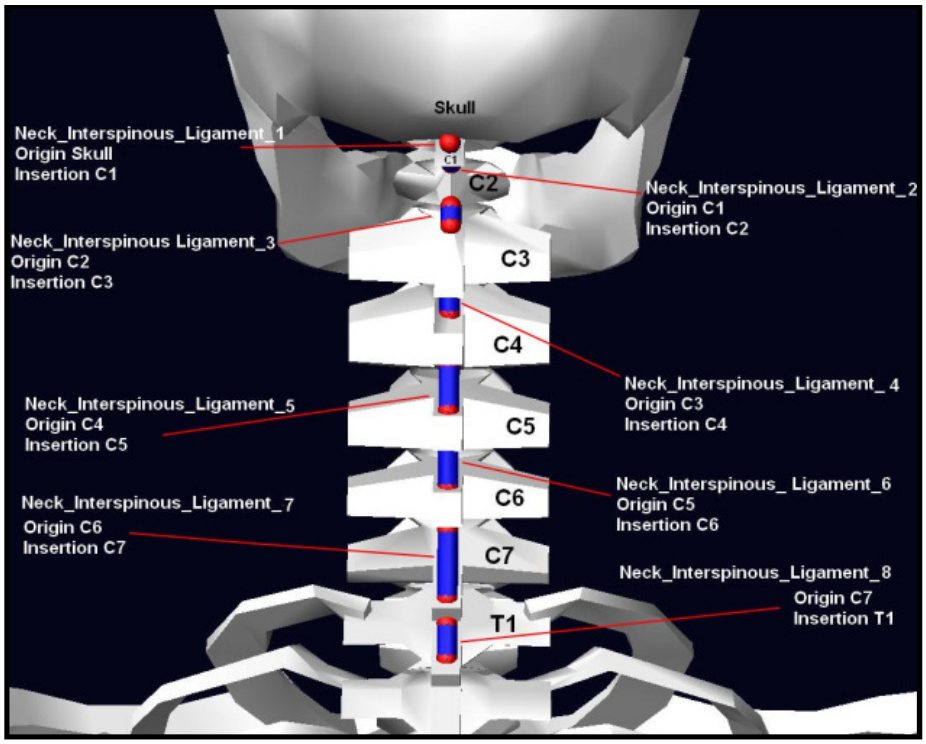
A2 Head and neck

Table A2.1 Range of movements of cervical vertebrae joints

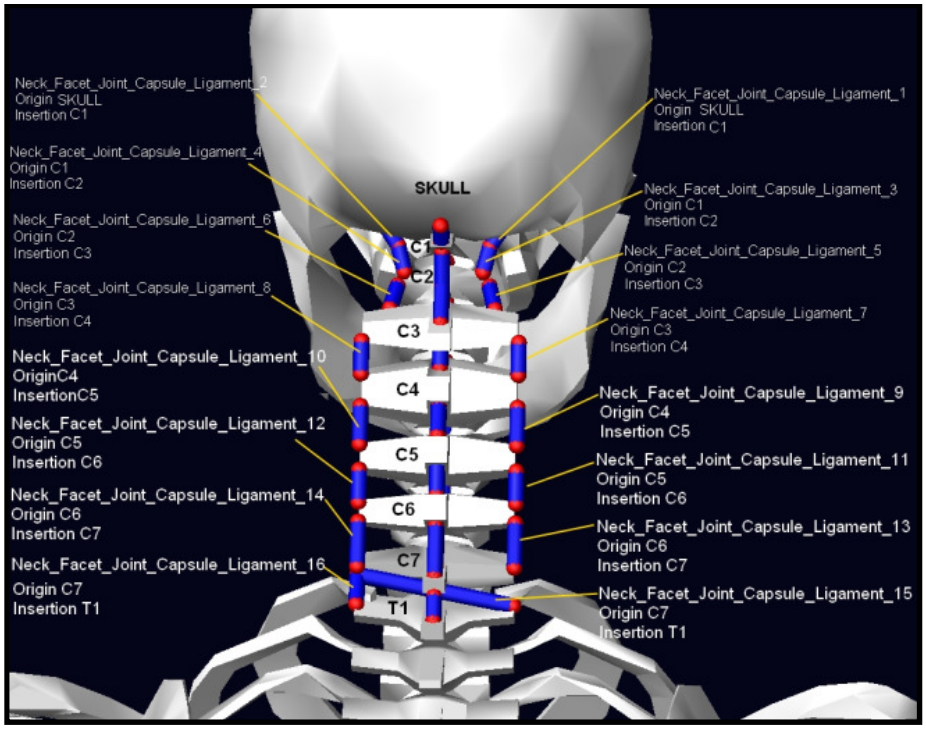
White & Panjabi, 1978	**Flexion-Extension (degree)**	**Lateral Bending (degree)**	**Axial Rotation (degree)**
Head-C1	13	8	0
C1-C2	10	0	47
C2-C3	8	10	9
C3-C4	13	11	11
C4-C5	12	11	12
C5-C6	17	8	10
C6-C7	16	7	9
C7-T1	9	4	8
Total	98	59	106
White & Panjabi, 1990	**Flexion-Extension (degree)**	**Lateral Bending (degree)**	**Axial Rotation (degree)**
Head-C1			
C1-C2			
C2-C3	10		
C3-C4	15		
C4-C5	20		
C5-C6	20		
C6-C7	17		
C7-T1	9		
Total			
Dvorac et al., 1988	**Flexion-Extension (degree)**	**Lateral Bending (degree)**	**Axial Rotation (degree)**
Head-C1			
C1-C2			
C2-C3	10 (active) 12 (passive)		
C3-C4	15 (active) 17 (passive)		
C4-C5	19 (active) 21 (passive)		
C5-C6	20 (active) 23 (passive)		
C6-C7	19 (active) 21 (passive)		
C7-T1			
Total			
Penning, 1978	**Flexion-Extension (degree)**	**Lateral Bending (degree)**	**Axial Rotation (degree)**
Head-C1			
C1-C2			
C2-C3	12 (active)		
C3-C4	18 (active)		
C4-C5	20 (active)		
C5-C6	20 (active)		
C6-C7	15 (active)		
C7-T1			
Total			
Sforza et al., 2002	**Flexion-Extension (degree)**	**Lateral Bending (degree)**	**Axial Rotation (degree)**
Total			
Adolescents	131.6 (active)	85 (active)	160.3 (active)
Young Adults	130.2(active)	77.2 (active)	155.1 (active)
Mid-Aged	117.4 (active)	78.7 (active)	153.3 (active)
Ferrario et al., 2002	**Flexion-Extension (degree)**	**Lateral Bending (degree)**	**Axial Rotation (degree)**
Total			
Women	136 (active)	90.8 (active)	161.9 (active)
Men	130.3(active)	77.2 (active)	155.1 (active)


SKULL
C1
C2
C3
C4
C5
C6
C7
T!
Neck_Flaval_Ligament_1
Origin Skull
Insertion C1
Neck_Flaval_Ligament_2
Origin C1
Insertion C2/ Ligament_3
Neck_Flaval_Ligament_3
Origin C2/C1 /Ligament_2
Insertion C3
Neck_Flaval_Ligament_ 4
Origin C3
Insertion C4
Neck_Flaval_Ligament_ 5
Origin C4
Insertion C5
Neck_Flaval_Ligament_6
Origin C5
Insertion C6
Neck_Flaval_Ligament_7
Origin C6
Insertion C7
Neck_Flaval_Ligament_8
Origin C7
Insertion T1

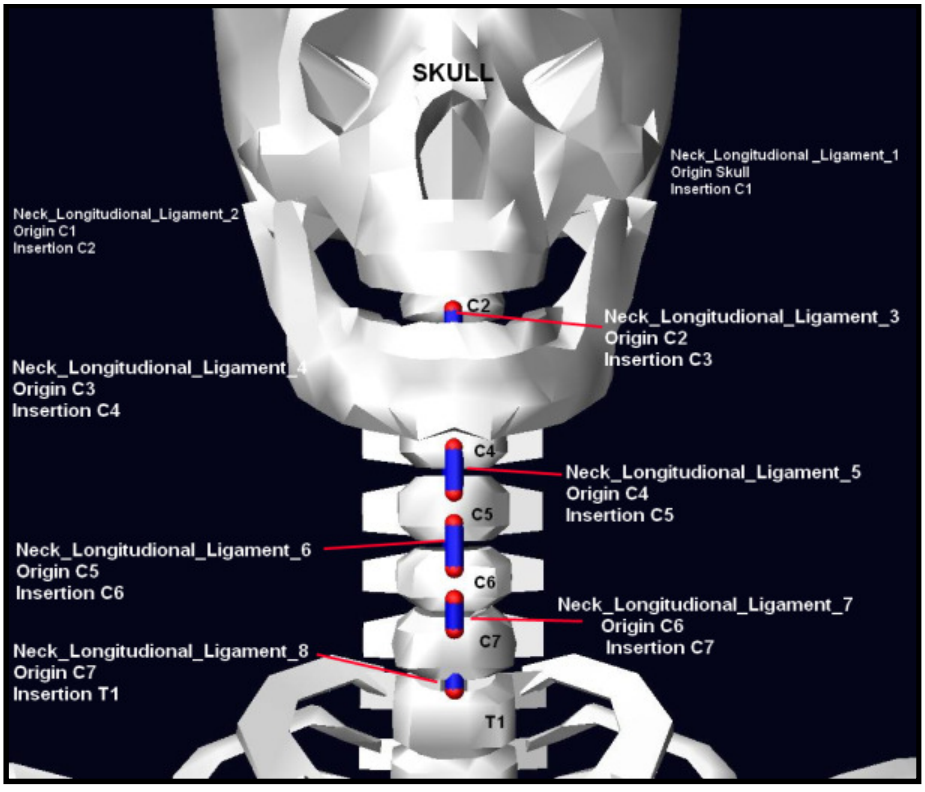
(a)


Skull
C1
C2
C3
C4
C5
C6
C7
T1
Neck_Interspinous_Ligament_1
Origin Skull
Insertion C1
Neck_Interspinous_Ligament_2
Origin C1
Insertion C2
Neck_Interspinous Ligament_3
Origin C2
Insertion C3
Neck_Interspinous_Ligament_ 4
Origin C3
Insertion C4
Neck_Interspinous_Ligament_5
Origin C4
Insertion C5
Neck_Interspinous_ Ligament_6
Origin C5
Insertion C6
Neck_Interspinous_Ligament_7
Origin C6
Insertion C7
Neck_Interspinous_Ligament_8
Origin C7
Insertion T1

(b)



(c)



(d)

Figure A2.1 Cervical vertebrae (neck) ligaments, (a) Interspinous ligaments, (b) Flaval ligaments, (c) Facet joint capsule ligaments, (d) Longitudional ligaments (reated in Lifemod software)

Table A2.2 Origin and insertion points of cervical vertebrae (neck) ligaments

Cervical vertebrae (neck) ligaments	Origin point	Insertion point
Neck_Interspinous_Ligament_1	Skull	C1
Neck_Interspinous_Ligament_2	C1	C2
Neck_Interspinous_Ligament_3	C2	C3
Neck_Interspinous_Ligament_4	C3	C4
Neck_Interspinous_Ligament_5	C4	C5
Neck_Interspinous_Ligament_6	C5	C6
Neck_Interspinous_Ligament_7	C6	C7
Neck_Interspinous_Ligament_8	C7	T1
Neck_Flaval_Ligament_1	Skull	C1
Neck_Flaval_Ligament_2	C1	C2/Neck_Flaval_Ligament_3
Neck_Flaval_Ligament_3	C1/C2/Neck_Flaval_Ligament_2	C3
Neck_Flaval_Ligament_4	C3	C4
Neck_Flaval_Ligament_5	C4	C5
Neck_Flaval_Ligament_6	C5	C6
Neck_Flaval_Ligament_7	C6	C7
Neck_Flaval_Ligament_8	C7	T1
Neck_Facet_Joint_Capsule_Ligament_1	Skull	C1
Neck_Facet_Joint_Capsule_Ligament_2	Skull	C1
Neck_Facet_Joint_Capsule_Ligament_3	C1	C2
Neck_Facet_Joint_Capsule_Ligament_4	C1	C2
Neck_Facet_Joint_Capsule_Ligament_5	C2	C3
Neck_Facet_Joint_Capsule_Ligament_6	C2	C3
Neck_Facet_Joint_Capsule_Ligament_7	C3	C4
Neck_Facet_Joint_Capsule_Ligament_8	C3	C4
Neck_Facet_Joint_Capsule_Ligament_9	C4	C5
Neck_Facet_Joint_Capsule_Ligament_10	C4	C5
Neck_Facet_Joint_Capsule_Ligament_11	C5	C6
Neck_Facet_Joint_Capsule_Ligament_12	C5	C6
Neck_Facet_Joint_Capsule_Ligament_13	C6	C7
Neck_Facet_Joint_Capsule_Ligament_14	C6	C7
Neck_Facet_Joint_Capsule_Ligament_15	C7	T1
Neck_Facet_Joint_Capsule_Ligament_16	C7	T1
Neck_Longitudional_Ligament_1	Skull	C1
Neck_Longitudional_Ligament_2	C1	C2
Neck_Longitudional_Ligament_3	C2	C3
Neck_Longitudional_Ligament_4	C3	C4
Neck_Longitudional_Ligament_5	C4	C5
Neck_Longitudional_Ligament_6	C5	C6
Neck_Longitudional_Ligament_7	C6	C7
Neck_Longitudional_Ligament_8	C7	T1

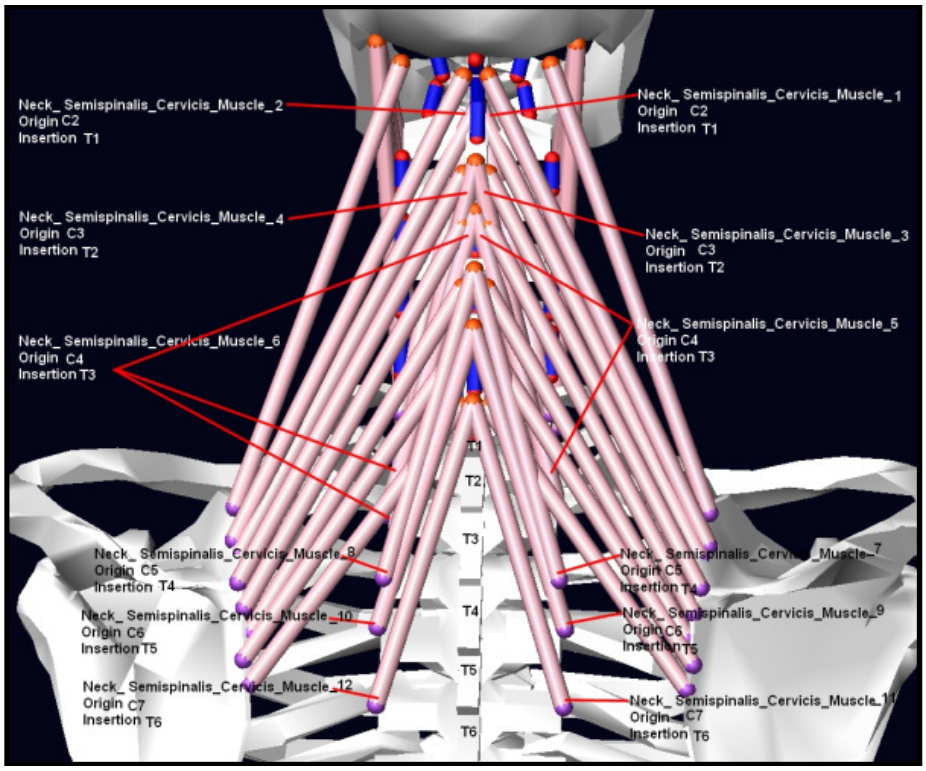
Table A2.3 Stiffness and damping parameters of cervical vertebrae (neck) ligaments

Yoganandan, Kumaresan & Pintar, 2001	Ligament Stiffness N/mm (Anterior Longitudional Ligament)	Ligament Stiffness N/mm (Posterior Longitudional Ligament)	Ligament Stiffness N/mm (Flaval Ligament)	Ligament Stiffness N/mm (Interspinous Ligament)	Ligament Stiffness N/mm (Joint Capsule Ligament)
Head-C1					32.6
C1-C2	24		11.6		32.3
C2-C3	16	25.4	25	7.74	
C3-C4	16	25.4	25	7.74	
C4-C5	16	25.4	25	7.74	
C5-C6	17.9	23	21.6	6.4	
C6-C7	17.9	23	21.6	6.4	
C7-T1	17.9	23	21.6	6.4	

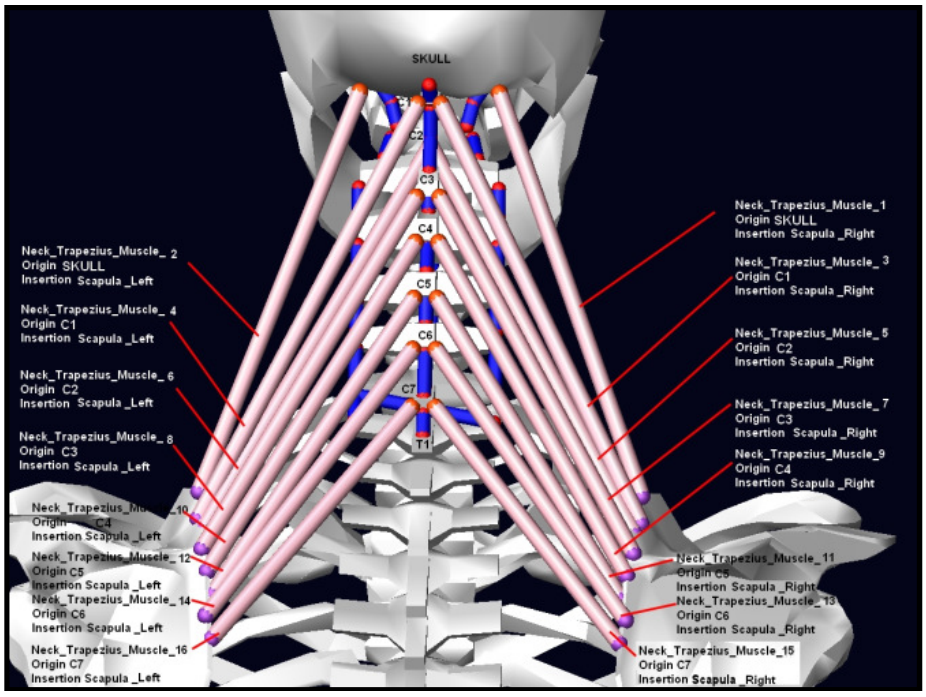
Pintar, 1986	Ligament Stiffness N/mm (Anterior Longitudional Ligament)	Ligament Stiffness N/mm (Posterior Longitudional Ligament)	Ligament Stiffness N/mm (Flaval Ligament)	Ligament Stiffness N/mm (Interspinous Ligament)	Ligament Stiffness N/mm (Joint Capsule Ligament)
Head-C1					32.6
C1-C2	24		11.6		32.3
C2-C3	16.6	21.4	16.3	7.3	30.7
C3-C4	16.6	21.4	16.3	7.3	30.7
C4-C5	16.6	21.4	16.3	7.3	30.7
C5-C6	17	25.9	25.3	5.3	29.2
C6-C7	17	25.9	25.3	5.3	29.2
C7-T1	17	25.9	25.3	5.3	29.2

Lifemod software	Ligament Stiffness lbf/in (Anterior Longitudional Ligament)	Ligament Stiffness lbf/in (Posterior Longitudional Ligament)	Ligament Stiffness lbf/in (Flaval Ligament)	Ligament Stiffness lbf/in (Interspinous Ligament)	Ligament Stiffness lbf/in (Joint Capsule Ligament)
Head-C1	100	100	100	100	100
C1-C2	100	100	100	100	100
C2-C3	100	100	100	100	100
C3-C4	100	100	100	100	100
C4-C5	100	100	100	100	100
C5-C6	100	100	100	100	100
C6-C7	100	100	100	100	100
C7-T1	100	100	100	100	100

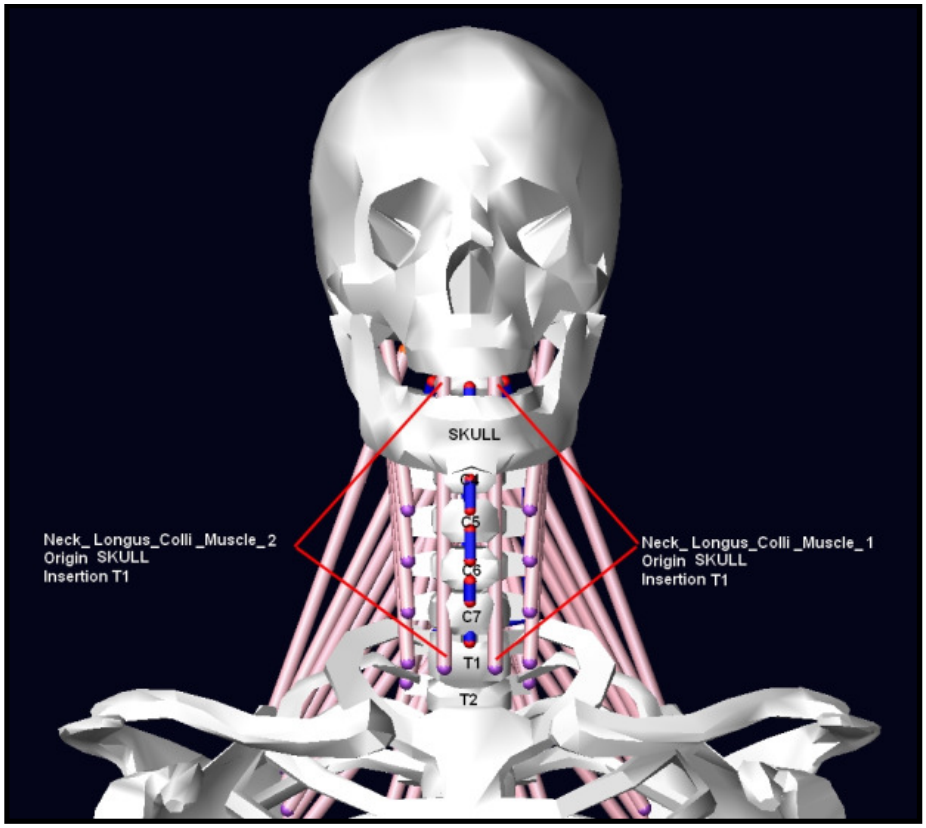
Lifemod software	Ligament Damping lbf/in (Anterior Longitudional Ligament)	Ligament Damping lbf/in (Posterior Longitudional Ligament)	Ligament Damping lbf/in (Flaval Ligament)	Ligament Damping lbf/in (Interspinous Ligament)	Ligament Damping lbf/in (Joint Capsule Ligament)
Head-C1	2	2	2	2	2
C1-C2	2	2	2	2	2
C2-C3	2	2	2	2	2
C3-C4	2	2	2	2	2
C4-C5	2	2	2	2	2
C5-C6	2	2	2	2	2
C6-C7	2	2	2	2	2
C7-T1	2	2	2	2	2


Neck_ Semispinalis_Cervicis_Muscle_2
Origin C2
Insertion T1
Neck_ Semispinalis_Cervicis_Muscle_1
Origin C2
Insertion T1
Neck_ Semispinalis_Cervicis_Muscle_4
Origin C3
Insertion T2
Neck_ Semispinalis_Cervicis_Muscle_3
Origin C3
Insertion T2
Neck_ Semispinalis_Cervicis_Muscle_6
Origin C4
Insertion T3
Neck_ Semispinalis_Cervicis_Muscle_5
Origin C4
Insertion T3
T1
T2
T3
T4
T5
T6
Neck_ Semispinalis_Cervicis_Muscle_8
Origin C5
Insertion T4
Neck_ Semispinalis_Cervicis_Muscle_7
Origin C5
Insertion T4
Neck_ Semispinalis_Cervicis_Muscle_10
Origin C6
Insertion T5
Neck_ Semispinalis_Cervicis_Muscle_9
Origin C6
Insertion T5
Neck_ Semispinalis_Cervicis_Muscle_12
Origin C7
Insertion T6
Neck_ Semispinalis_Cervicis_Muscle_11
Origin C7
Insertion T6

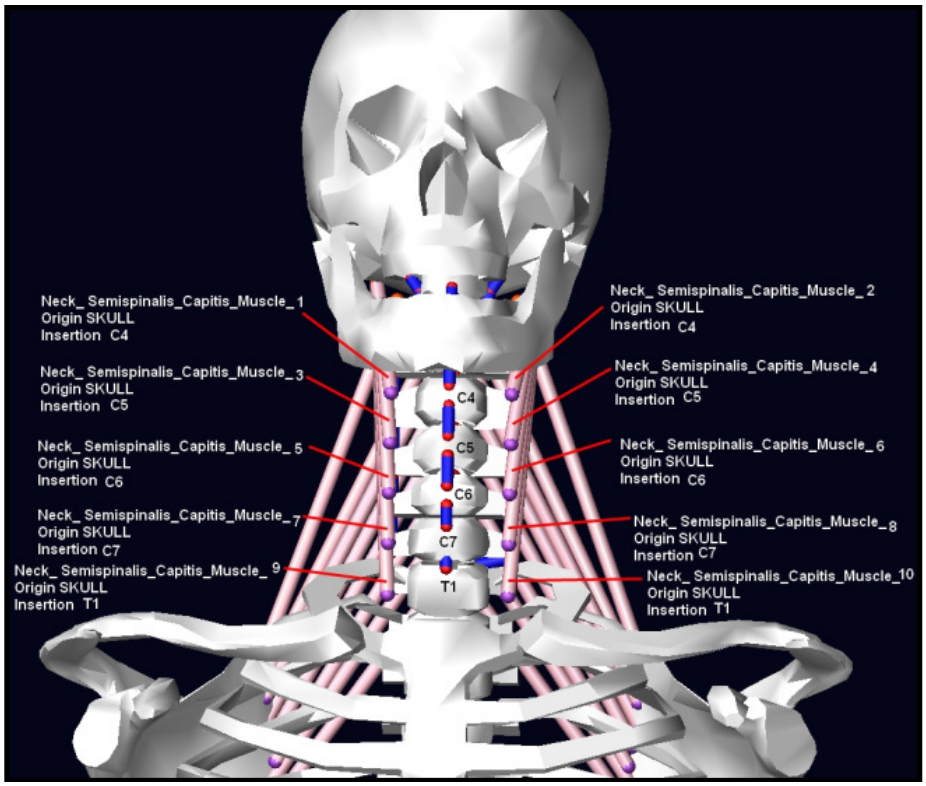
(a)


SKULL
C2
C3
C4
C5
C6
C7
T1
Neck_Trapezius_Muscle_1
Origin SKULL
Insertion Scapula _Right
Neck_Trapezius_Muscle_ 2
Origin SKULL
Insertion Scapula _Left
Neck_Trapezius_Muscle_ 3
Origin C1
Insertion Scapula _Right
Neck_Trapezius_Muscle_ 4
Origin C1
Insertion Scapula _Left
Neck_Trapezius_Muscle_ 5
Origin C2
Insertion Scapula _Right
Neck_Trapezius_Muscle_ 6
Origin C2
Insertion Scapula _Left
Neck_Trapezius_Muscle_ 7
Origin C3
Insertion Scapula _Right
Neck_Trapezius_Muscle_ 8
Origin C3
Insertion Scapula _Left
Neck_Trapezius_Muscle_9
Origin C4
Insertion Scapula _Right
Neck_Trapezius_Muscle_12
Origin C5
Insertion Scapula _Left
Neck_Trapezius_Muscle_11
Origin C5
Insertion Scapula _Right
Neck_Trapezius_Muscle_14
Origin C6
Insertion Scapula _Left
Neck_Trapezius_Muscle_13
Origin C6
Insertion Scapula _Right
Neck_Trapezius_Muscle_16
Origin C7
Insertion Scapula _Left
Neck_Trapezius_Muscle_15
Origin C7
Insertion Scapula _Right

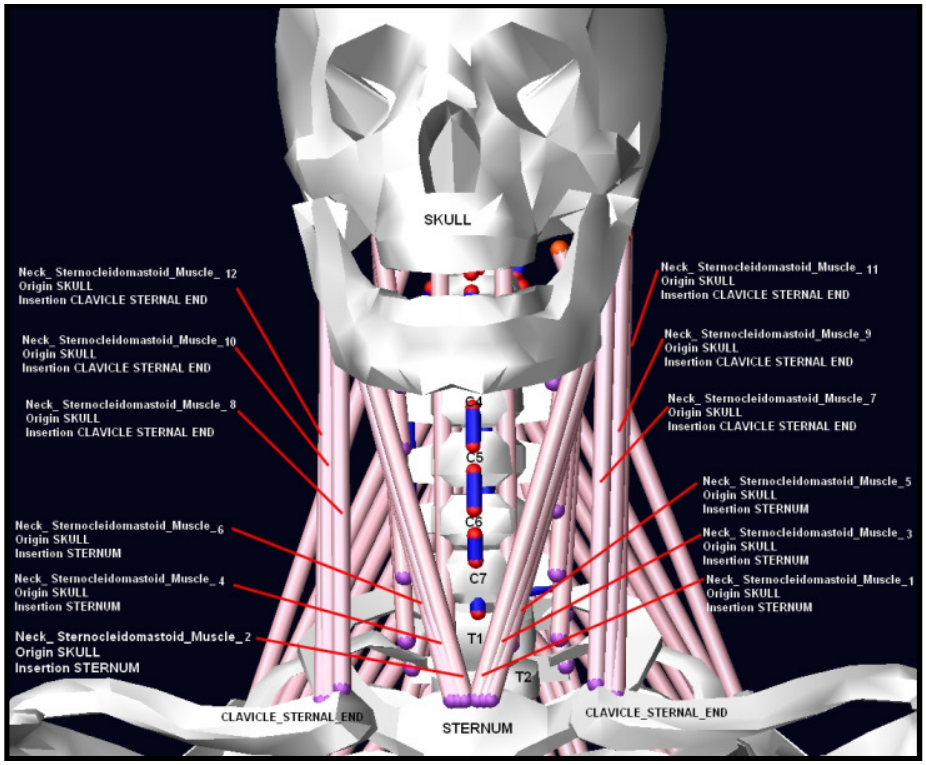
(b)


SKULL
C4
C5
C6
C7
T1
T2
Neck_ Longus_Colli _Muscle_2
Origin SKULL
Insertion T1
Neck_ Longus_Colli _Muscle_1
Origin SKULL
Insertion T1

(c)


Neck_ Semispinalis_Capitis_Muscle_1
Origin SKULL
Insertion C4
Neck_ Semispinalis_Capitis_Muscle_2
Origin SKULL
Insertion C4
Neck_ Semispinalis_Capitis_Muscle_3
Origin SKULL
Insertion C5
Neck_ Semispinalis_Capitis_Muscle_4
Origin SKULL
Insertion C5
Neck_ Semispinalis_Capitis_Muscle_5
Origin SKULL
Insertion C6
Neck_ Semispinalis_Capitis_Muscle_6
Origin SKULL
Insertion C6
Neck_ Semispinalis_Capitis_Muscle_7
Origin SKULL
Insertion C7
Neck_ Semispinalis_Capitis_Muscle_8
Origin SKULL
Insertion C7
Neck_ Semispinalis_Capitis_Muscle_9
Origin SKULL
Insertion T1
Neck_ Semispinalis_Capitis_Muscle_10
Origin SKULL
Insertion T1
C4
C5
C6
C7
T1

(d)



(e)

Figure A2.2 Cervical vertebrae (neck) muscles, (a) Semispinalis Cervicis muscles, (b) Trapezius muscles, (c) Longus Colli muscles, (d) Semispinalis Capitis muscles, (e) Sternocleidomastoid muscles (created in Lifemod software)

Table A2.4 Origin and insertion points of cervical vertebrae (neck) muscles

Cervical vertebrae or neck Muscles	Origin Point	Insertion Point
Neck_Semispinalis_Cervicis_Muscle_1	C2	T1
Neck_Semispinalis_Cervicis_Muscle_2	C2	T1
Neck_Semispinalis_Cervicis_Muscle_3	C3	T2
Neck_Semispinalis_Cervicis_Muscle_4	C3	T2
Neck_Semispinalis_Cervicis_Muscle_5	C4	T3
Neck_Semispinalis_Cervicis_Muscle_6	C4	T3
Neck_Semispinalis_Cervicis_Muscle_7	C5	T4
Neck_Semispinalis_Cervicis_Muscle_8	C5	T4
Neck_Semispinalis_Cervicis_Muscle_9	C6	T5
Neck_Semispinalis_Cervicis_Muscle_10	C6	T5
Neck_Semispinalis_Cervicis_Muscle_11	C7	T6
Neck_Semispinalis_Cervicis_Muscle_12	C7	T6
Neck_Trapezius_Muscle_1	Skull	Scapula_Right
Neck_Trapezius_Muscle_2	Skull	Scapula_Left
Neck_Trapezius_Muscle_3	C1	Scapula_Right
Neck_Trapezius_Muscle_4	C1	Scapula_Left
Neck_Trapezius_Muscle_5	C2	Scapula_Right
Neck_Trapezius_Muscle_6	C2	Scapula_Left
Neck_Trapezius_Muscle_7	C3	Scapula_Right
Neck_Trapezius_Muscle_8	C3	Scapula_Left
Neck_Trapezius_Muscle_9	C4	Scapula_Right
Neck_Trapezius_Muscle_10	C4	Scapula_Left
Neck_Trapezius_Muscle_11	C5	Scapula_Right
Neck_Trapezius_Muscle_12	C5	Scapula_Left

Neck_Trapezius_Muscle_13	C6	Scapula_Right
Neck_Trapezius_Muscle_14	C6	Scapula_Left
Neck_Trapezius_Muscle_15	C7	Scapula_Right
Neck_Trapezius_Muscle_16	C7	Scapula_Left
Neck_Longus_Colli_Muscle_1	Skull	T1
Neck_Longus_Colli_Muscle_2	Skull	T1
Neck_Semispinalis_Capitis_Muscle_1	Skull	C4
Neck_Semispinalis_Capitis_Muscle_2	Skull	C4
Neck_Semispinalis_Capitis_Muscle_3	Skull	C5
Neck_Semispinalis_Capitis_Muscle_4	Skull	C5
Neck_Semispinalis_Capitis_Muscle_5	Skull	C6
Neck_Semispinalis_Capitis_Muscle_6	Skull	C6
Neck_Semispinalis_Capitis_Muscle_7	Skull	C7
Neck_Semispinalis_Capitis_Muscle_8	Skull	C7
Neck_Semispinalis_Capitis_Muscle_9	Skull	T1
Neck_Semispinalis_Capitis_Muscle_10	Skull	T1
Neck_Sternocleidomastoid_Muscle_1	Skull	Sternum
Neck_Sternocleidomastoid_Muscle_2	Skull	Sternum
Neck_Sternocleidomastoid_Muscle_3	Skull	Sternum
Neck_Sternocleidomastoid_Muscle_4	Skull	Sternum
Neck_Sternocleidomastoid_Muscle_5	Skull	Sternum
Neck_Sternocleidomastoid_Muscle_6	Skull	Sternum
Neck_Sternocleidomastoid_Muscle_7	Skull	Clavicle_Sternal_End
Neck_Sternocleidomastoid_Muscle_8	Skull	Clavicle_Sternal_End
Neck_Sternocleidomastoid_Muscle_9	Skull	Clavicle_Sternal_End
Neck_Sternocleidomastoid_Muscle_10	Skull	Clavicle_Sternal_End
Neck_Sternocleidomastoid_Muscle_11	Skull	Clavicle_Sternal_End
Neck_Sternocleidomastoid_Muscle_12	Skull	Clavicle_Sternal_End

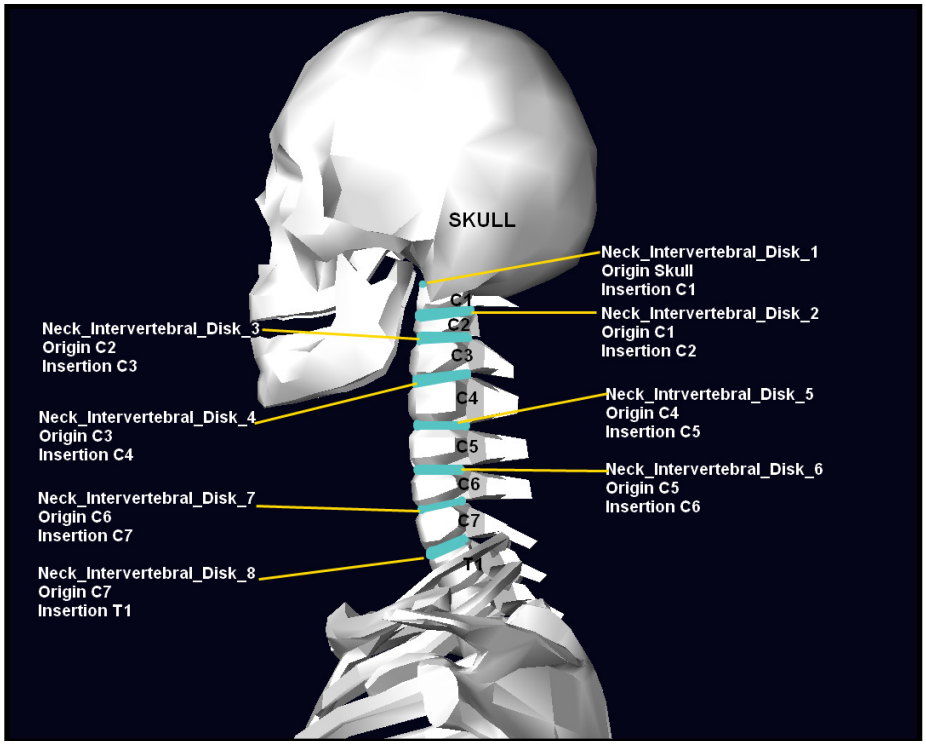


Figure A2.3 Intervertebral discs between each cervical vertebra

Table A2.5 Tensile failure stiffness parameters, compression failure stiffness parameters and area of each intervertebral disk along cervical vertebrae (Yoganandan et al., 2001)

Yoganandan, Kumaresan & Pintar, 2001	Intervertebral Disk Area mm*2	Tensile Failure Stiffness N/mm	Compression Failure Stiffness N/mm
Head-C1			
C1-C2			
C2-C3	108-262	63.5	637.5
C3-C4	98-442	69.8	765.3
C4-C5	118-332	66.8	784.6
C5-C6	129-440	22	800.2
C6-C7	168-502	69	829.7
C7-T1	188-482	82.2	973.6

Table A2.6 Tensile failure stiffness parameters, compression failure stiffness parameters and area of each intervertebral disk along cervical vertebrae (Moroney et al., 1999)

Moroney et al., 1988	Compression Stiffness N/mm	Anterior Shear Stiffness N/mm	Posterior Shear Stiffness N/mm	Right Lateral Shear Stiffness N/mm
Lower neck	492	62	50	73

Table A 2.7 Medial-lateral shear stiffness, axial tension-compression stiffness, anterior posterior shear stiffness, lateral bending stiffness, flexion-extension stiffness and axial rotation stiffness parameters of each intervertebral disk along cervical vertebrae (Lifemod software)

Lifemod software	Medial-Lateral Shear Stiffness (lbf/in)	Axial Tension - Compression Stiffness (lbf/in)	Anterior-Posterior Shear Stiffness (lbf/in)	Lateral Bending Stiffness (lbf/in)	Flexion-Extension Stiffness (lbf/in)	Axial Rotation Stiffness (lbf/in)
Head-C1	4.17E+004	3.02E+004	3.20E+004	5.10E-002	7.20E-002	6.50E-002
C1-C2	4.17E+004	3.02E+004	3.20E+004	5.10E-002	7.20E-002	6.50E-002
C2-C3	4.17E+004	3.02E+004	3.20E+004	5.10E-002	7.20E-002	6.50E-002
C3-C4	4.17E+004	3.02E+004	3.20E+004	5.10E-002	7.20E-002	6.50E-002
C4-C5	4.17E+004	3.02E+004	3.20E+004	5.10E-002	7.20E-002	6.50E-002
C5-C6	4.17E+004	3.02E+004	3.20E+004	5.10E-002	7.20E-002	6.50E-002
C6-C7	4.17E+004	3.02E+004	3.20E+004	5.10E-002	7.20E-002	6.50E-002
C7-T1	4.17E+004	3.02E+004	3.20E+004	5.10E-002	7.20E-002	6.50E-002

Table A 2.8 Medial-lateral shear damping, axial tension-compression damping, anterior posterior shear damping, lateral bending damping, flexion-extension damping and axial rotation damping parameters of each intervertebral disk along cervical vertebrae (Lifemod software)

Lifemod software	Medial-Lateral Shear Damping (lbf/in)	Axial Tension - Compression Damping (lbf/in)	Anterior-Posterior Shear Damping (lbf/in)	Lateral Bending Damping (lbf/in)	Flexion-Extension Damping (lbf/in)	Axial Rotation Damping (lbf/in)
Head-C1	57.5	57.5	57.5	0.232	0.232	0.232
C1-C2	57.5	57.5	57.5	0.232	0.232	0.232
C2-C3	57.5	57.5	57.5	0.232	0.232	0.232
C3-C4	57.5	57.5	57.5	0.232	0.232	0.232
C4-C5	57.5	57.5	57.5	0.232	0.232	0.232
C5-C6	57.5	57.5	57.5	0.232	0.232	0.232
C6-C7	57.5	57.5	57.5	0.232	0.232	0.232
C7-T1	57.5	57.5	57.5	0.232	0.232	0.232

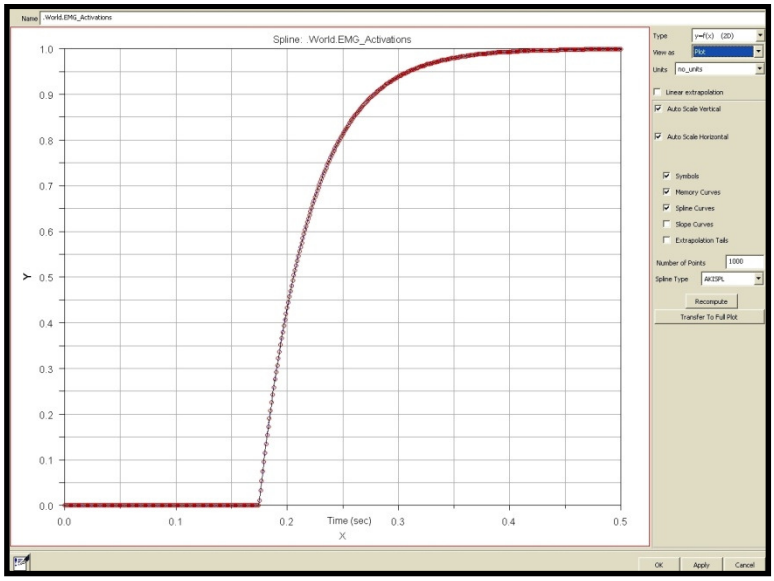


Figure A2.4 Activation curve of cervical vertebrae (neck) muscles which shows activated muscles after the impact to the upper torso at 0.175 seconds (Lifemod software)

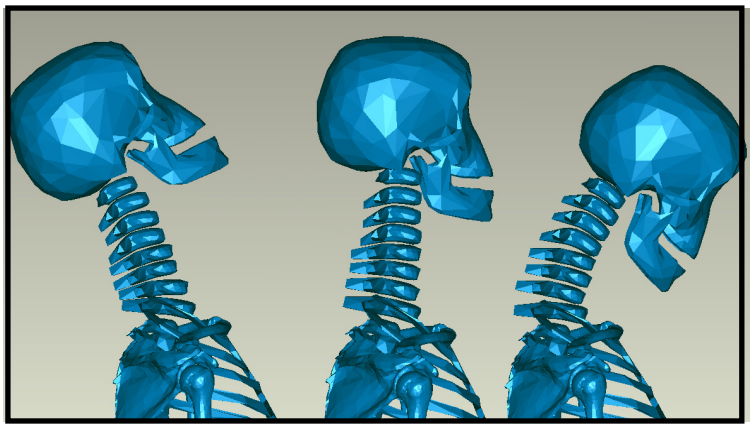


Figure A2.5 Extension (Left) neutral position (Middle) and flexion (Right) movements of head and neck

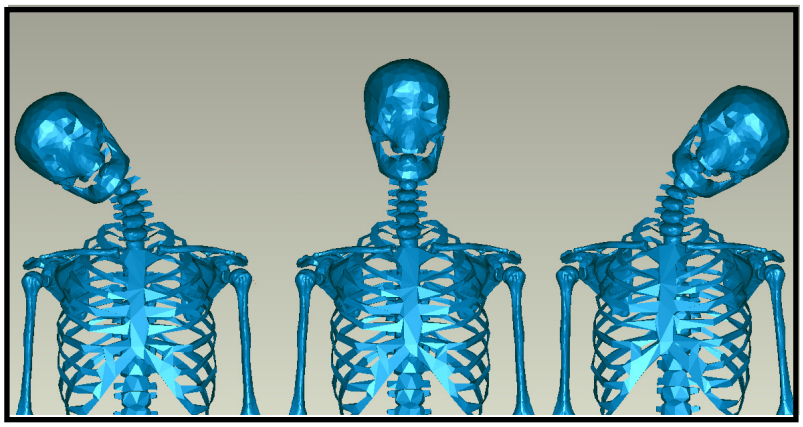


Figure A2.6 Lateral bending to the right of the cervical vertebra (Left), neutral position (Middle), lateral bending to the left of the cervical vertebrae (Right)

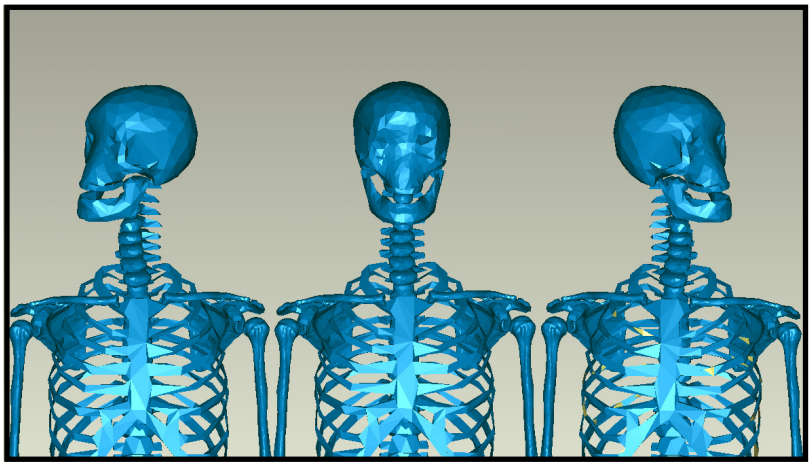


Figure A2.7 Axial rotation to the right of the cervical vertebra (Left), neutral position (Middle), axial rotation to the left of the cervical vertebrae (Right)

A3 Elbow

Table A3.1 Valgus and varus movement range of elbow joint relative to flexion angles

O`Driscoll et al., 1992 (Through comparison with literature relatively lower varus-valgus laxity in implanted joint has been corresponded in to muscle force absorption due to semi constrained hinge implant)	Normal (Intact) elbow joint maximum varus-valgus laxity during active flexion (deg)	Implanted (loose-hinge) elbow joint maximum varus-valgus laxity during active flexion (deg)	Normal elbow joint maximum varus-valgus Laxity during active muscle flexion (deg)	Implanted (loose-hinge) elbow joint maximum varus-valgus Laxity during active muscle flexion (deg)
Loose-hinge (linked)	$2.7^0 \pm 1.5^0$	$3.8^0 \pm 1.4^0$	$6.9^0 \pm 3.7^0$	$10.8^0 \pm 1.8^0$

An, 2005	Normal elbow joint maximum varus-valgus laxity during flexion (deg)	Implanted elbow joint maximum varus-valgus laxity during flexion (deg)	Normal elbow joint maximum varus-valgus laxity with muscle loading during flexion (deg)	Implanted elbow joint maximum varus-valgus laxity with muscle loading during flexion (deg)	Implanted elbow joint maximum varus-valgus laxity with ligament release during flexion (deg)	Elbow joint maximum varus-valgus laxity with radial head excision during flexion (deg)	Elbow joint maximum varus-valgus laxity with MCL incision and radial head excision during flexion (deg)
Coonrad Morrey semiconstrained TEA	$2.7^0 \pm 1.5^0$	$3.8^0 \pm 1.4^0$	$6.9^0 \pm 3.7^0$	$10.8^0 \pm 1.8^0$			
GBS III	$5.8^0 \pm 2.2^0$	$9.6^0 \pm 2.5^0$			$12.8^0 \pm 2.9^0$		
Norway TEA							
Ewald Capitellocondylar TEA				$4.3^0 \pm 2.4^0$			
Souter Strathclyde TEA			$4.3^0 \pm 2.3^0$ $3.5^0 \pm 1.7^0$ (with doubled muscle load)	$6.5^0 \pm 1.5^0$ $5.5^0 \pm 1.6^0$ (with doubled muscle load)			
Sorbie Questor (ulinked)				$8.6^0 \pm 4.0^0$		$13.3^0 \pm 5.5^0$	
Additional assessment	$3.4^0 \pm 1.6^0$						$11.1^0 \pm 5.6^0$

Ahmad, Park & ElAttrache, 2004 (Full tear of MUCL shows maximum valgus range of movement)	Load (N.m)	Change in valgus angle with partial tear of MUCL (medial ulnar collateral ligament) relative to intact MUCL (deg)	Change in valgus angle with full tear of MUCL relative to intact MUCL (deg)
Elbow flexion angle (30^0)	1.25	$2.30^0 \pm 0.71^0$	$6.59^0 \pm 2.76^0$
Elbow flexion angle (90^0)	1.25	$0.71^0 \pm 1.69^0$	$3.64^0 \pm 2.59^0$
Elbow flexion angle (30^0)	2.00	$2.21^0 \pm 0.81^0$	$7.37^0 \pm 3.45^0$
Elbow flexion angle (90^0)	2.00	$0.05^0 \pm 2.00^0$	$4.26^0 \pm 3.10^0$

Table A3.2 Origin and insertion points of elbow joint ligaments

Elbow joint ligaments	Origin Point	Insertion Point
Elbow_Annular_Ligament (AL)	Unla_Supinator_Crest (approximate)	Ulna_Sublime_Tubercle (approximate)
Elbow_Radial_Collateral_Ligament (RCL) (part of LCL)	Humerus_Lateral_Epicondyle	Radius_Head and AL
Elbow_Lateral_Ulnar_Collateral_Ligament (LUCL) (part of LCL)	Humerus_Lateral_Epicondyle	Unla_Supinator_Crest
Elbow_Anterior_Medial_Collateral_Ligament (part of MCL) (also called Anterior Oblique)	Humerus_Medial_Epicondyle	Ulna_Sublime_Tubercle
Elbow_Posterior_Medial_Collateral_Ligament (part of MCL) (also called Posterior Oblique)	Humerus_Medial_Epicondyle	Ulna_Olecranon
Elbow_Transverse_Medial_Collateral_Ligament (part of MCL)	Ulna	Ulna
Elbow_Oblique_Ligament	Ulna	Radius

Table A3.3 Stiffness and damping parameters of elbow joint ligaments (Regan et al., 1991)

Regan et al., 1991	AL Stiffness (N/mm)	RCL Stiffness (N/mm)	LUCL Stiffness (N/mm)	Anterior MCL Stiffness (N/mm)	Posterior MCL Stiffness (N/mm)	Transverse MCL Stiffness (N/mm)	Oblique Ligament Stiffness (N/mm)
	28.5	15.5	57.0	72.3	52.2		

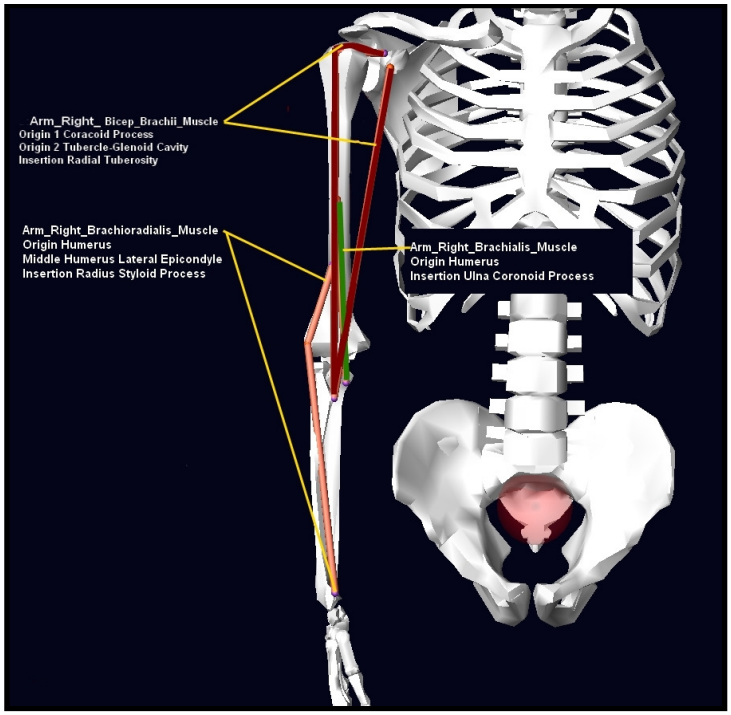


Figure A3.1 Major Flexor muscles (bicep brachii, brachioradialis, brachialis) of arm

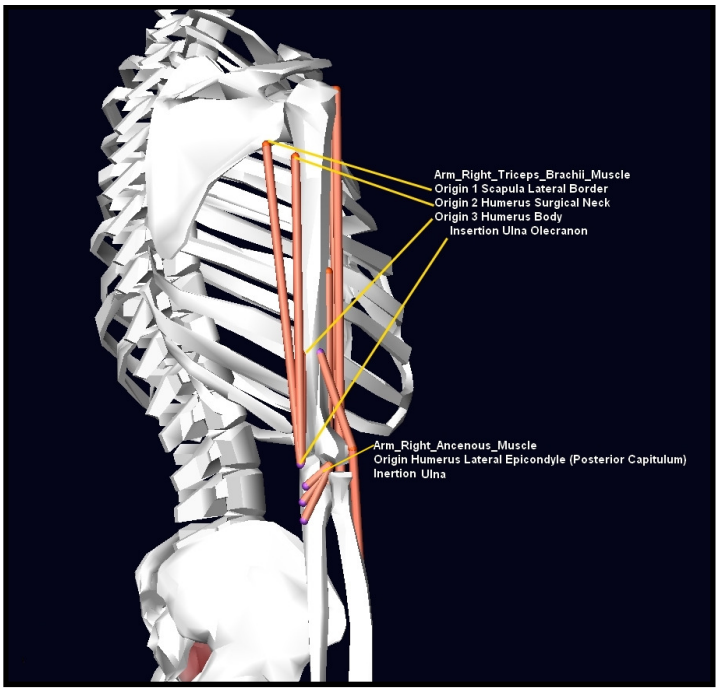


Figure A3.2 Major Extensor muscles (triceps brachii, ancenous) of arm

Table A3.4 Origin and insertion points of arm flexor and extensor muscles

Arm flexor and extensor muscles	Origin Point	Insertion Point
Arm_Biceps_Brachii_Muscle	Scapula_Coracoid_Process Scapula_Tubercle_Glenoid_Cavity	Radius_Tuberosity
Arm_Brachioradialis_Muscle	Humerus Humerus_Lateral_Epicondyle	Radius_Styloid_Process
Arm_Brachialis_Muscle	Humerus	Ulna_Coronoid_Process
Arm_Triceps_Brachii_Muscle	Scapula_Lateral_Border Humerus_Surgical_Neck Humerus_Body(Shaft)	Ulna_Olecranon
Arm_Ancenous_Muscle	Humerus_Lateral_Epicondyle	Ulna

Table A3.5 Anthropometric parameters of arm flexor and extensor muscles

Muscle (Holzbaur et al., 2007)	PCSA (cm^2)	Length (cm)	Average Volume (cm^3)
Biceps Brachii	8.2	27	143.7
Brachioradialis	3.9	23.5	65.1
Brachialis	14.4	22.3	143.7
Triceps Brachii	40	27	372.1
Ancenous	1.3	8.3	10.8
Pronator_Teres	6.5	16.1	38.4
Extensor_Carpi_Radialis_Longus	2.7	22.2	37.5

Muscle (Garner & Pandy, 2003)	PCSA (cm^2)	Muscle Fibre Length (cm)	Tendon Slack Length (cm)	Average Volume (cm^3)
Biceps Brachii	25.9	14.22	22.98	365.84
Brachioradialis	3.08	27.03	6.04	83.19
Brachialis	25.88	10.28	1.75	265.96
Triceps Brachii	76.30	8.77	19.05	619.99
Ancenous				
Pronator_Teres	17.96	4.48	11.58	80.41
Extensor_Carpi_Radialis_Longus	24.89	7.28	26.80	166.61

Table A3.6 Tendon displacement and moment arm estimations of arm flexor and extensor muscles

Muscle (Murray, Delp & Buchanan, 1995)	Biceps Brachii Tendon displacement (cm)	Biceps Brachii Moment arm (cm)	Brachioradialis Tendon displacement (cm)	Brachioradialis Moment arm (cm)	Brachialis Tendon displacement (cm)
Flexion Angle (deg)					
20	5.8	2	7.5	1	4.4
40	5	2.8	6.8	2.2	3.8
80	2.6	3.6	4.1	4	2.4
120	0.1	3	0.1	5.8	0.1

Muscle (Murray, Delp & Buchanan, 1995)	Brachialis Moment arm (cm)	Triceps Brachii Tendon displacement (cm)	Triceps Brachii Moment arm (cm)	Pronator_Teres Tendon displacement (cm)	Pronator_Teres Moment arm (cm)
Flexion Angle (deg)					
20	1.2	0.4	-2.4	3.5	1.1
40	1.8	1.1	-2.2	3	1.7
80	2.8	2.6		1.8	1.9
120	3.2	3.8	-1.8	0.1	2.6

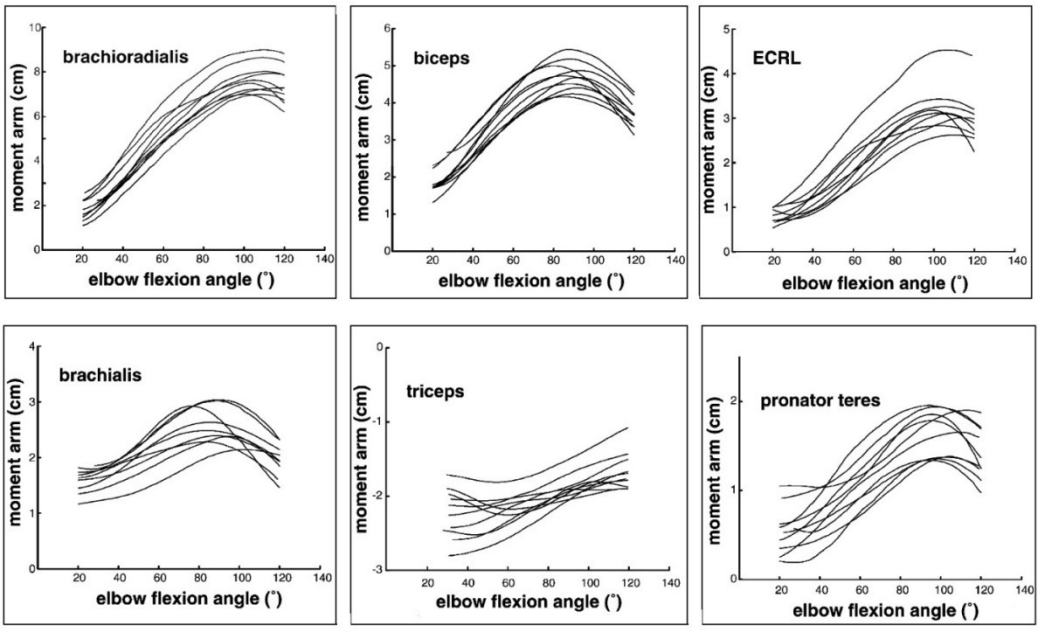


Figure A3.3 Elbow joint major muscle moment arms during flexion movement from ten Specimen (Murray, Buchanan & Delp, 2002)

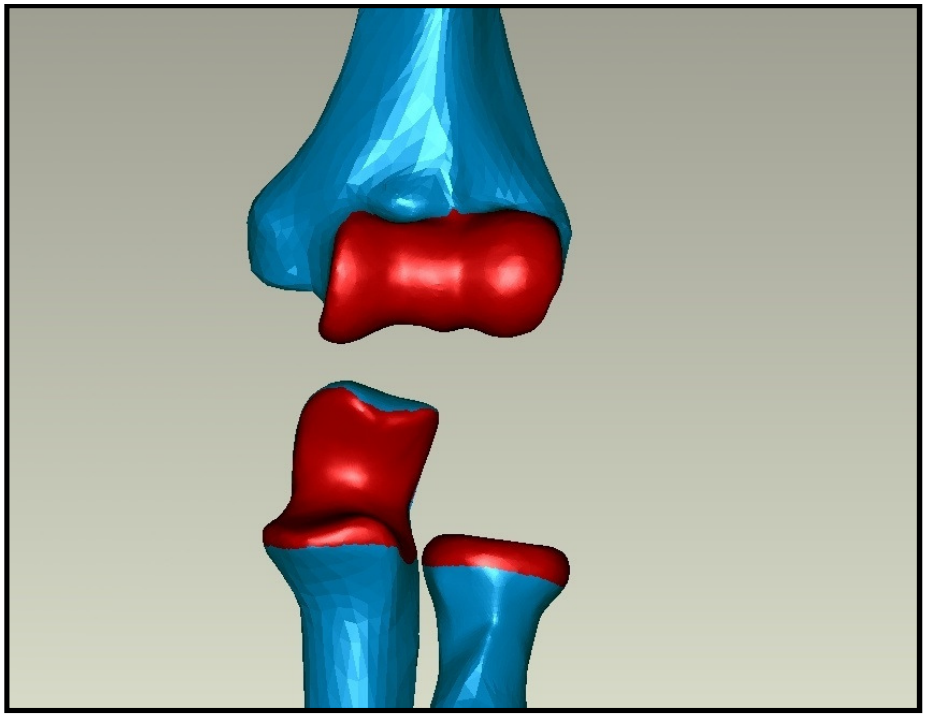


Figure A3.4 Elbow joint cartilages on articulating surfaces of humerus, ulna and radius (highlighted in red)

Table A3.7 Average contact areas with standard deviations between olecranon part of ulna and trochlea part of humerus (ulna-humeral joint) during elbow flexion (In vitro, mm^2)

Ahmad, Park & ElAttrache, 2004	**Load (N)**	**Contact area with intact MUCL(medial ulnar collateral ligament) (mm^2)**	**Contact area with partial tear of MUCL (mm^2)**	**Contact area with full tear of MUCL (mm^2)**
Elbow flexion angle (30^0)	1.25	35.3 ± 12.2	29.7 ± 14.4	20.1 ± 9.9
Elbow flexion angle (90^0)	1.25	37.5 ± 16.6	33.2 ± 14.8	26.0 ± 12.1
Elbow flexion angle (30^0)	2.00	43.2 ± 17.3	30.5 ± 15.3	21.1 ± 6.8
Elbow flexion angle (90^0)	2.00	42.1 ± 13.1	38.7 ± 11..6	31.6 ± 12.7

Table A3.8 Average contact pressure with standard deviations between olecranon part of ulna and trochlea part of humerus (ulna-humeral joint) during elbow flexion (In vitro, MPa)

Ahmad, Park & ElAttrache, 2004	**Load (N)**	**Contact pressure with intact MUCL(medial ulnar collateral**	**Contact pressure with partial tear of**	**Contact pressure with full tear of**

		ligament) (MPa)	MUCL (MPa)	MUCL (Mpa)
Elbow flexion angle (30^0)	1.25	0.37 ± 0.17	0.42 ±0.2	0.77 ± 0.19
Elbow flexion angle (90^0)	1.25	0.56 ± 0.17	0.64 ± 0.16	0.76 ± 0.23
Elbow flexion angle (30^0)	2.00	0.54 ± 0.22	0.56 ± 0.2	0.93 ± 0.15
Elbow flexion angle (90^0)	2.00	0.70 ± 0.20	0.77 ± 0.15	0.91 ± 0.06

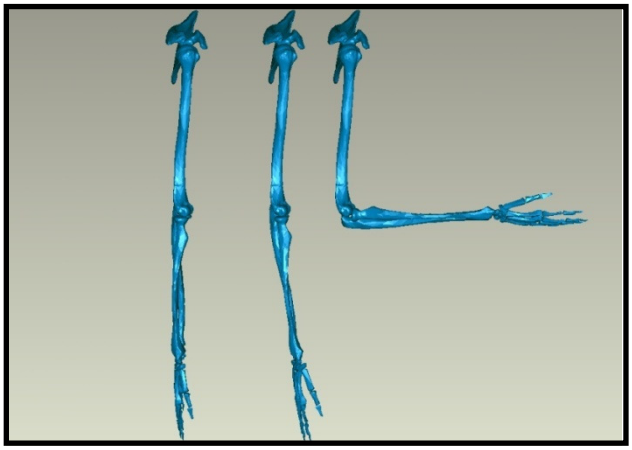


Figure A3.5 Elbow joint extension (Left), neutral (Middle) and flexion (Right) movements

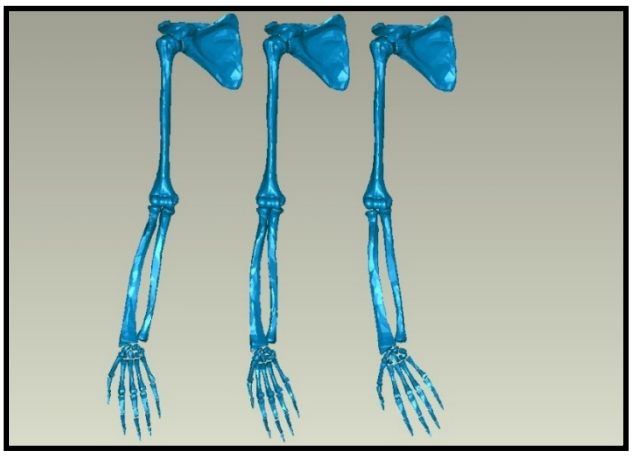


Figure A3.6 Elbow joint, valgus (Left), neutral (Middle) and varus (Right) movements

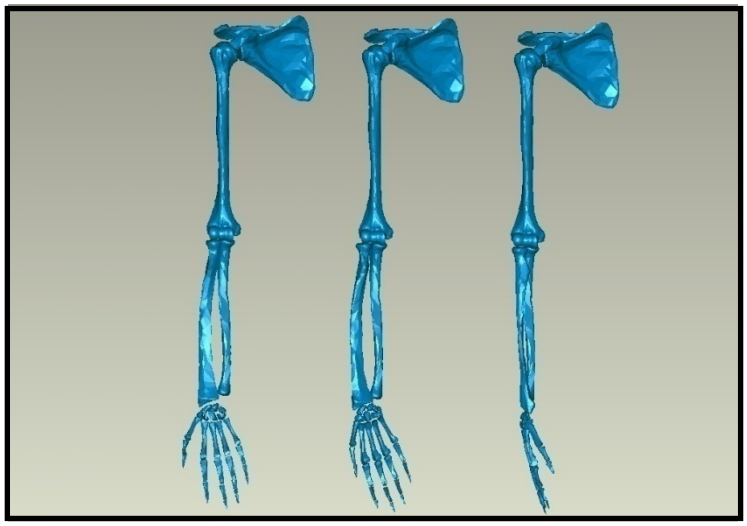


Figure A3.7 Elbow joint supination (Left), neutral (Middle) and pronation (Right) movements

A4 Euler angles and direction cosines

Euler angles are normally used to obtain the transformation matrix from one axes frame to the other. The Euler angles are popular because, they can be used to visualise how an object is oriented by means of a sequence of rotations. And they are useful also because in interactive graphics it is easier to use them in order to rotate an object. Therefore there is a need to transfer a given direction cosine matrix to the Euler angle definition. The Euler angle description of a rigid rotation is not unique and the user need to specify which particular rotation is selected. Three orthogonal transformation matrices are given as,

Rotation in z direction, $\boldsymbol{T}(\theta_z) = \begin{bmatrix} \cos(\theta_z) & -\sin(\theta_z) & 0 \\ \sin(\theta_z) & \cos(\theta_z) & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Rotation in y direction, $\boldsymbol{T}(\theta_y) = \begin{bmatrix} \cos(\theta_y) & 0 & \sin(\theta_y) \\ 0 & 1 & 0 \\ -\sin(\theta_y) & 0 & \cos(\theta_y) \end{bmatrix}$

And

Rotation in x direction, $\boldsymbol{T}(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_x) & -\sin(\theta_x) \\ 0 & \sin(\theta_x) & \cos(\theta_x) \end{bmatrix}$

Assuming that a local axes system, first rotated by the x axis then y and then z, the resulting transformation matrix can be written as, $\boldsymbol{X} = \boldsymbol{T}(\theta_z)\boldsymbol{T}(\theta_y)\boldsymbol{T}(\theta_x)\boldsymbol{x}$, where $\boldsymbol{X} = (X, Y, Z)$ and $\boldsymbol{x} = (x, y, z)$, The overall rotation matrix for this particular order of transformation is given by, $\boldsymbol{T} = \boldsymbol{T}(\theta_z)\boldsymbol{T}(\theta_y)\boldsymbol{T}(\theta_x)$

However if the overall rotation matrix is given and the angles are required then the calculation of angles will not be unique and they will depend on the order which the rotations are implemented. The motion modelling does not need Euler angles for the simulation. Only at the beginning if either mass or the mount configurations are described by the Euler angles then these are used to calculate the transformation matrix, but there onwards only direction cosines are needed. However if these angles change (in case of large rotations) then user may wish to know how they change. Therefore the analysis program must keep the information about the order of rotation in order to reverse the calculation from direction cosines to the Euler angles. The other reason is the way the KernelCAD graphics library displays the object motion. The library currently requires the orientation of an object to be described in term of its axes rotations. Therefore there is a need to describe the Euler angles from the direction cosines. It must be noted that there are six possible orders with which the reverse operation can be performed. Each is giving different Euler angles. These are (3,2,1), (3,1,2), (2,1,3), (2,3,1), (1,2,3) and (1,3,2). The following sections give all six solutions for the direction cosines matrix,

$$\boldsymbol{T} = \begin{bmatrix} l_{11} & l_{12} & l_{13} \\ l_{21} & l_{22} & l_{23} \\ l_{31} & l_{32} & l_{33} \end{bmatrix}$$

- **From local to global given by *x* first, *y* second and *z* third**

$$T = T(\theta_z)T(\theta_y)T(\theta_x) = \begin{bmatrix} \cos\theta_y \cos\theta_z & \sin\theta_x \sin\theta_y \cos\theta_z - \cos\theta_x \sin\theta_z & \cos\theta_x \sin\theta_y \cos\theta_z + \sin\theta_x \sin\theta_z \\ \cos\theta_y \sin\theta_z & \sin\theta_x \sin\theta_y \sin\theta_z + \cos\theta_x \cos\theta_z & \cos\theta_x \sin\theta_y \sin\theta_z - \sin\theta_x \cos\theta_z \\ -\sin\theta_y & \sin\theta_x \cos\theta_y & \cos\theta_x \cos\theta_y \end{bmatrix}$$

Solution is; $\tan(\theta_z) = \frac{l_{21}}{l_{11}}$, $\frac{-\tan(\theta_y)}{\cos\theta_z} = \frac{l_{31}}{l_{11}}$ or $\frac{-\tan(\theta_y)}{\cos\theta_x} = \frac{l_{31}}{l_{33}}$ and $\tan(\theta_x) = \frac{l_{32}}{l_{33}}$

- **From local to global given by *z* first, *x* second and *y* third**

$$T(\theta_y)T(\theta_x)T(\theta_z) = \begin{bmatrix} \cos\theta_z \cos\theta_y + \sin\theta_z \sin\theta_x \sin\theta_y & -\sin\theta_z \cos\theta_y + \cos\theta_z \sin\theta_x \sin\theta_y & (?)\cos\theta_x \sin\theta_y \\ \sin\theta_z \cos\theta_x & \cos\theta_z \cos\theta_x & -\sin\theta_x \\ -\cos\theta_z \sin\theta_y + \sin\theta_z \sin\theta_x \cos\theta_y & \sin\theta_z \sin\theta_y + \cos\theta_z \sin\theta_x \cos\theta_y & \cos\theta_x \cos\theta_y \end{bmatrix}$$

Solution is; $\tan(\theta_z) = \frac{l_{21}}{l_{22}}$, $\frac{-\tan(\theta_x)}{\cos\theta_z} = \frac{l_{23}}{l_{22}}$ or $\frac{-\tan(\theta_x)}{\cos\theta_y} = \frac{l_{23}}{l_{33}}$ and $\tan(\theta_y) = \frac{l_{13}}{l_{33}}$

- **From local to global given by *z* first, *y* second and *x* third**

$$T(\theta_x)T(\theta_y)T(\theta_z)=\begin{bmatrix}\cos\theta_z\cos\theta_y & -\sin\theta_z\cos\theta_y & \sin\theta_y\\ \sin\theta_z\cos\theta_x+\cos\theta_x\sin\theta_y\sin\theta_x & \cos\theta_z\cos\theta_x-\sin\theta_z\sin\theta_y\sin\theta_x & -\cos\theta_y\sin\theta_x\\ \sin\theta_z\sin\theta_x-\cos\theta_z\sin\theta_y\cos\theta_x & \cos\theta_z\sin\theta_x+\sin\theta_z\sin\theta_y\cos\theta_x & \cos\theta_x\cos\theta_y\end{bmatrix}$$

Solution is; $-\tan(\theta_z)=\frac{l_{12}}{l_{11}}$, $\frac{\tan(\theta_y)}{\cos\theta_z}=\frac{l_{13}}{l_{11}}$ or $\frac{\tan(\theta_y)}{\cos\theta_x}=\frac{l_{13}}{l_{33}}$ and $-\tan(\theta_x)=\frac{l_{23}}{l_{33}}$

- **From local to global given by, *y* first, *x* second and *z* third**

$$T(\theta_z)T(\theta_x)T(\theta_y)=\begin{bmatrix}\cos\theta_y\cos\theta_z-\sin\theta_y\sin\theta_x\sin\theta_z & -\cos\theta_x\sin\theta_z & \sin\theta_y\cos\theta_z+\cos\theta_y\sin\theta_x\sin\theta_z\\ \cos\theta_y\sin\theta_z+\sin\theta_y\sin\theta_x\cos\theta_z & \cos\theta_x\cos\theta_z & \sin\theta_y\sin\theta_z-\cos\theta_y\sin\theta_x\cos\theta_z\\ -\sin\theta_y\cos\theta_x & \sin\theta_x & \cos\theta_y\cos\theta_x\end{bmatrix}$$

Solution is; $-\tan(\theta_z)=\frac{l_{12}}{l_{22}}$, $\frac{\tan(\theta_x)}{\cos\theta_y}=\frac{l_{32}}{l_{33}}$ or $\frac{\tan(\theta_x)}{\cos\theta_z}=\frac{l_{32}}{l_{22}}$ and $-\tan(\theta_y)=\frac{l_{31}}{l_{33}}$

- **From local to global given by *x* first, *z* second and *y* third**

$$T(\theta_y)T(\theta_z)T(\theta_x)=\begin{bmatrix}\cos\theta_z\cos\theta_y & -\cos\theta_x\sin\theta_z\cos\theta_y+\sin\theta_x\sin\theta_y & \sin\theta_x\sin\theta_z\cos\theta_y+\sin\theta_y\cos\theta_x\\ \sin\theta_z & \cos\theta_x\cos\theta_z & -\sin\theta_x\cos\theta_z\\ -\cos\theta_z\sin\theta_y & \cos\theta_x\sin\theta_z\sin\theta_y+\sin\theta_x\cos\theta_y & -\sin\theta_x\sin\theta_z\sin\theta_y+\cos\theta_x\cos\theta_y\end{bmatrix}$$

Solution is; $-\tan(\theta_y)=\frac{l_{31}}{l_{11}}$, $\frac{\tan(\theta_z)}{\cos\theta_y}=\frac{l_{21}}{l_{11}}$ or $\frac{\tan(\theta_z)}{\cos\theta_x}=\frac{l_{21}}{l_{22}}$ and $-\tan(\theta_x)=\frac{l_{23}}{l_{22}}$

- **From local to global given by, *y* first, *z* second and *x* third**

$$T(\theta_x)T(\theta_z)T(\theta_y) = \begin{bmatrix} \cos\theta_y \cos\theta_z & -\sin\theta_z & \sin\theta_y \cos\theta_z \\ \cos\theta_y \sin\theta_z \cos\theta_x + \sin\theta_y \sin\theta_x & \cos\theta_z \cos\theta_x & \sin\theta_y \sin\theta_z \cos\theta_x - \cos\theta_y \sin\theta_x \\ \cos\theta_y \sin\theta_z \sin\theta_x - \sin\theta_y \cos\theta_x & \cos\theta_z \sin\theta_x & \sin\theta_y \sin\theta_z \sin\theta_x + \cos\theta_y \cos\theta_x \end{bmatrix}$$

Solution is; $\tan(\theta_x) = \frac{l_{32}}{l_{22}}$, $-\frac{\tan(\theta_z)}{\cos\theta_x} = \frac{l_{12}}{l_{22}}$ or $-\frac{\tan(\theta_z)}{\cos\theta_y} = \frac{l_{12}}{l_{11}}$ and $\tan(\theta_y) = \frac{l_{13}}{l_{11}}$

A5 Selected software code

A5.1 Connectivity for closed object (box, sphere, cone, cylinder and muscle)

```
For i = 1 To m
  For j = 1 To n
      NodeNO = j + n * (i - 1)
      eleNo2 = 2 * NodeNO
      eleNo1 = eleNo2 - 1
      triangleNodes(eleNo1, 1) = NodeNO + 1
      triangleNodes(eleNo1, 2) = NodeNO
      triangleNodes(eleNo1, 3) = NodeNO + n

      triangleNodes(eleNo2, 1) = NodeNO + 1
      triangleNodes(eleNo2, 2) = NodeNO + n
      triangleNodes(eleNo2, 3) = NodeNO + n + 1

      If j = n Then
          triangleNodes(eleNo1, 1) = (i - 1) * n + 1
          triangleNodes(eleNo2, 1) = (i - 1) * n + 1
          triangleNodes(eleNo2, 3) = i * n + 1
      End If

     Next
```

```
        Next
```

A5.2 Inertia matrix code

```
Sub computeInertia(ByVal m_iCurObject As ISection, ByRef mass As Double, ByRef COG() As
Double, ByRef Inertia(,) As Double)

        oneDiv6 = 1.0 / 6.0
        oneDiv24 = 1.0 / 24.0
        oneDiv60 = 1.0 / 60.0
        oneDiv120 = 1.0 / 120.0

        m_iMesh = CType(m_iCurObject, KernCADnet.IMesh)  ' extract mesh information

        For t = 1 To m_simplexCount (number triangular meshes)

            m_iMesh.GetSimplexVertices(m_curSimplex, Vertex(0), Vertex(1), Vertex(2))  '
mesh simplex vertex pointers
            r0, r1, r2 correspons to Vertex(0), Vertex(1) and Vertex(2) respectively
            d = (r1 - r0) X (r2 - r0)
            Subexpression(r0(1), r1(1), r2(1), f1x, f2x, f3x, g0x, g1x, g2x)
            Subexpression(r0(2), r1(2), r2(2), f1y, f2y, f3y, g0y, g1y, g2y)
            Subexpression(r0(3), r1(3), r2(3), f1z, f2z, f3z, g0z, g1z, g2z)

            integral(0) = integral(0) + d(1) * f1x
            integral(1) = integral(1) + d(1) * f2x
            integral(2) = integral(2) + d(2) * f2y
            integral(3) = integral(3) + d(3) * f2z
            integral(4) = integral(4) + d(1) * f3x
            integral(5) = integral(5) + d(2) * f3y
            integral(6) = integral(6) + d(3) * f3z
            integral(7) = integral(7) + d(1) * (r0(2) * g0x + r1(2) * g1x + r2(2) * g2x)
            integral(8) = integral(8) + d(2) * (r0(3) * g0y + r1(3) * g1y + r2(3) * g2y)
            integral(9) = integral(9) + d(3) * (r0(1) * g0z + r1(1) * g1z + r2(1) * g2z)
            m_iMesh.GetSimplexIterator.GetNext(m_curSimplex)
        Next t
        integral(0) = integral(0) * oneDiv6
        integral(1) = integral(1) * oneDiv24
        integral(2) = integral(2) * oneDiv24
        integral(3) = integral(3) * oneDiv24
        integral(4) = integral(4) * oneDiv60
        integral(5) = integral(5) * oneDiv60
        integral(6) = integral(6) * oneDiv60
        integral(7) = integral(7) * oneDiv120
        integral(8) = integral(8) * oneDiv120
        integral(9) = integral(9) * oneDiv120

        mass = integral(0)
        COG(1) = integral(1) / mass
        COG(2) = integral(2) / mass
        COG(3) = integral(3) / mass

        Inertia(1, 1) = integral(5) + integral(6)
        Inertia(2, 2) = integral(4) + integral(6)
        Inertia(3, 3) = integral(4) + integral(5)
        Inertia(1, 2) = -integral(7)
        Inertia(2, 3) = -integral(8)
        Inertia(1, 3) = -integral(9)


        Inertia(1, 1) = Inertia(1, 1) - mass * (COG(2) * COG(2) + COG(3) * COG(3))
        Inertia(2, 2) = Inertia(2, 2) - mass * (COG(3) * COG(3) + COG(1) * COG(1))
        Inertia(3, 3) = Inertia(3, 3) - mass * (COG(1) * COG(1) + COG(2) * COG(2))
        Inertia(1, 2) = Inertia(1, 2) + mass * COG(1) * COG(2)
        Inertia(2, 3) = Inertia(2, 3) + mass * COG(2) * COG(3)
        Inertia(1, 3) = Inertia(1, 3) + mass * COG(1) * COG(3)

        Inertia(2, 1) = Inertia(1, 2)
```

```
        Inertia(3, 1) = Inertia(1, 3)
        Inertia(3, 2) = Inertia(2, 3)

    End Sub

    Private Sub Subexpression(ByVal w0 As Double, ByVal w1 As Double, ByVal w2 As
Double, ByRef f1 As Double, ByRef f2 As Double, ByRef f3 As Double, ByRef g0 As Double,
ByRef g1 As Double, ByRef g2 As Double)

        temp0 = w0 + w1
        f1 = temp0 + w2
        temp1 = w0 * w0
        temp2 = temp1 + w1 * temp0
        f2 = temp2 + w2 * f1
        f3 = w0 * temp1 + w1 * temp2 + w2 * f2
        g0 = f2 + w0 * (f1 + w0)
        g1 = f2 + w1 * (f1 + w1)
        g2 = f2 + w2 * (f1 + w2)
    End Sub
```

A6 Geometric data preparations from Geomagic Studio 9

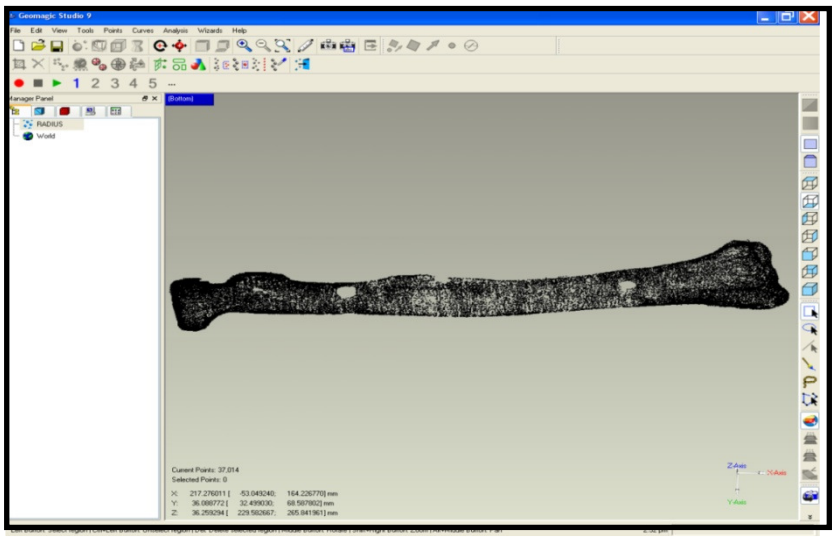


Figure A6.1 Screen shot of Geomagic Studio 9

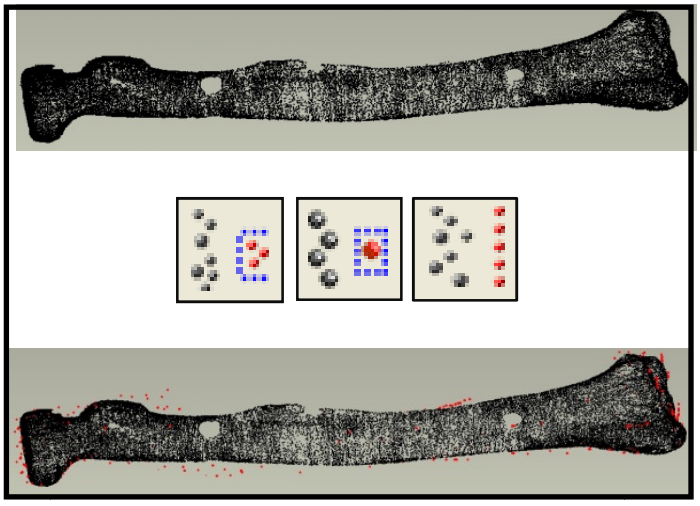


Figure A6.2 Disconnected surface point (red) elimination with Select Disconnected, Select Outliers and Reduce Noise functions

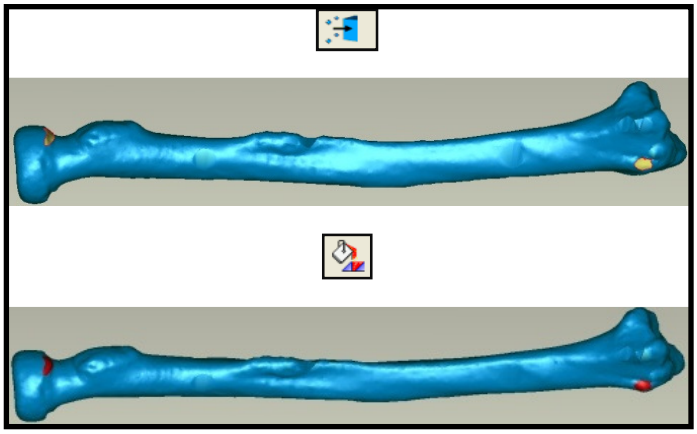


Figure A6.3 Wrapping function for triangulation and Fill Hole function for filling surface holes

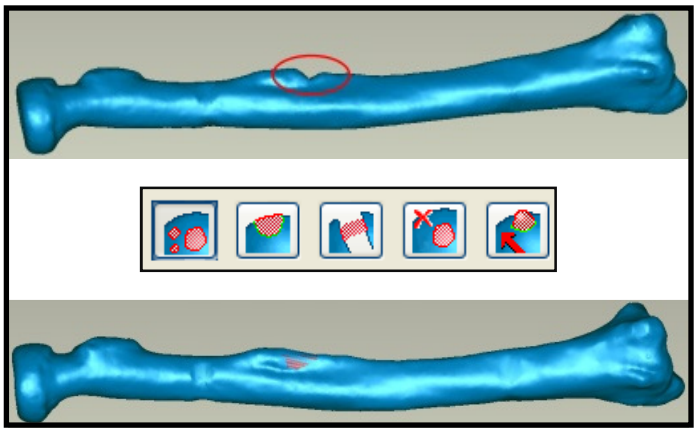


Figure A6.4 Geomagic Studio 9; Fill Hole, Fill Partial, Create Bridge, Clean Up and Move functions. Triangles have been deleted and Create Bridge has been used to fill the defected surface

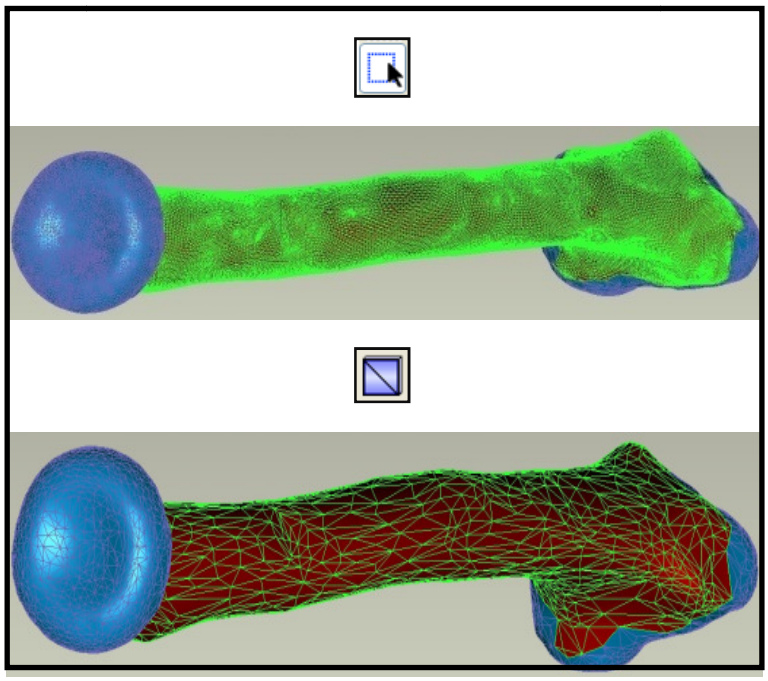


Figure A6.5 Triangle reduction is performed for selected surface by using Decline Triangle function. In this figure 75,000 triangles are reduced into 7000 triangles with 6000 triangles of contact surfaces and 1000 triangles for bone shaft

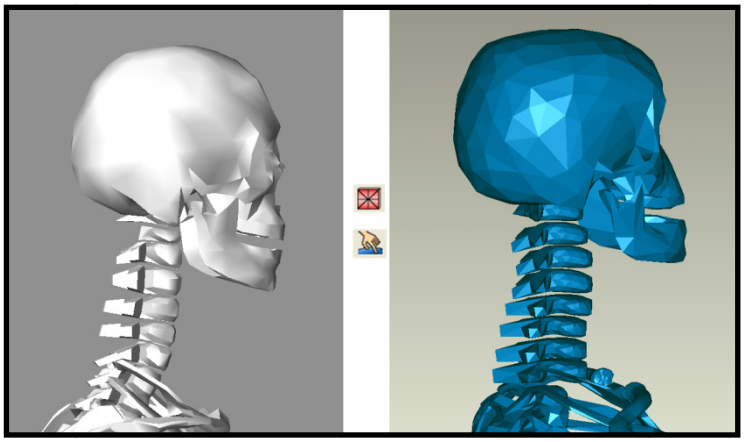


Figure A6.6 Cervical vertebrae joints from Lifemod software (Left). Triangulation is performed by Refine Polygons function (upper) and surface defects are smoothed by Sandpaper function (lower) through Geomagic Studio 9. For more realistic and anatomic cervical column, C1 and C2 needed to be shifted (Right)