

University of Memphis

University of Memphis Digital Commons

Electronic Theses and Dissertations

1-1-2019

A simulation-enhanced intraoperative planning tool for robotic-assisted total knee arthroplasty

Daniel Farley

Follow this and additional works at: <https://digitalcommons.memphis.edu/etd>

Recommended Citation

Farley, Daniel, "A simulation-enhanced intraoperative planning tool for robotic-assisted total knee arthroplasty" (2019). *Electronic Theses and Dissertations*. 2894.

<https://digitalcommons.memphis.edu/etd/2894>

This Dissertation is brought to you for free and open access by University of Memphis Digital Commons. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of University of Memphis Digital Commons. For more information, please contact khggerty@memphis.edu.

A SIMULATION-ENHANCED INTRAOPERATIVE SURGICAL PLANNING TOOL
FOR ROBOTIC-ASSISTED TOTAL KNEE ARTHROPLASTY

By

Daniel Farley

A Dissertation

Submitted in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

Major: Mechanical Engineering

The University of Memphis

August 2019

To my wife, Kelly,
and my parents, Nancy and David,
who have always
inspired me to
pursue knowledge

ACKNOWLEDGEMENTS

To my Major Advisor, Dr. Gladius Lewis, I express my sincerest appreciation for his assistance throughout each phase of this study. Without his supervision, patient guidance, and inspiration, this work would have not been possible. I would also like to thank members of my dissertation committee, Dr. Teong Tan, Dr. Ranganathan Gopalakrishnan, Mr. Ed Austin, and Dr. Si Janna, whose guidance and recommendations were invaluable.

In addition, I thank Liz Duxbury, Eric Fahlgren, and Shawn McGuan, of LifeModeler, Inc. (San Clemente, CA), for their generous donation of time and expertise. Investigation of various experimental datasets would not have been possible without the support provided by the LifeModeler team. Furthermore, I thank Smith & Nephew (Memphis, TN) for supporting this research project.

ABSTRACT

The purpose of the present study was to investigate current methods of surgical planning used in conjunction with robotics-assisted total knee arthroplasty (raTKA) to determine if improvements could be made using advanced computational techniques. Thus, through the use of musculoskeletal multi-body dynamic simulations, an enhanced surgical planning tool was developed, which provides insight on active postoperative joint mechanics. Development of the tool relied on patient-specific simulations using single-leg and full-body models. These simulations were constructed using two publicly-available datasets (Orthoload and SimTK); in particular, joint loading data obtained from subjects during various activities. Simulation parameters were optimized using a design-of experiments (DOE) methodology and validation of each of the models was conducted by calculating the root mean square error (RMSE) between joint loading calculated using the model and the corresponding results given in the appropriate dataset. Optimized and validated variants of each of the models were used in conjunction with the results of DOE studies that characterized the influence of a number of surgical planning variables on various biomechanical responses and linear regression analysis to derive knee performance equations (KPEs). In literature studies, some of the aforementioned responses have been strongly correlated with two outcomes commonly reported by dissatisfied TKA patients, namely, anterior knee pain and poor proprioception. In a proof-of-concept study, KPEs were used to calculate optimal positions and orientations of the femoral and tibial components in the case of one subject featured in the SimTK dataset. These results differed from corresponding ones reportedly achieved for the implant components in the subject. This trend suggests there is potential to improve robotic surgical planning for current-generation raTKA systems through the use of musculoskeletal simulation. Use of the proposed surgical planning tool does not require

computational resources beyond what are used with a specified current-generation raTKA system (Navio Surgical System). Furthermore, there are only minimal differences between the workflow involving the proposed planning tool and that when Navio Surgical System is used. A number of recommendations for future studies are made, such as larger scale simulation validation work and use of more complex regression techniques when deriving the KPEs.

TABLE OF CONTENTS

LIST OF FIGURES	iii
LIST OF TABLES	vii
CHAPTER 1 INTRODUCTION	1
CHAPTER 2 BACKGROUND	5
2.1. Osteoarthritis of the Knee and Management/Treatment Modalities	5
2.2. Knee Anatomy and Physiology	8
2.3. Total Knee Arthroplasty	12
2.3.1. Overview	12
2.3.2. Conventional TKA	12
2.3.3. Robotics-Assisted TKA	15
2.3.4. Elements of a Robotics-Assisted Surgical System	17
2.3.5. Types of Robotics-Assisted Surgical Systems	18
2.3.6. Features of Two Robotics-Assisted Total Knee Arthroplasty Systems	21
2.4. Patient Outcomes with Total Knee Arthroplasty	25
2.4.1. Conventional Total Knee Arthroplasty	25
2.4.2. Robotics-Assisted Total Knee Arthroplasty	26
2.5. Passive Joint Balancing	28
2.6. Musculoskeletal Simulation Modeling	30
2.7. Musculoskeletal Simulation Models and Surgical Planning	33

CHAPTER 3 DEVELOPMENT OF SURGICAL PLANNING TOOL.....	35
3.1. Overview	35
3.2. Simulation Development	36
3.3. Patient-Specific Model Development	39
3.4. Inverse Dynamic Simulation and Forward Dynamic Simulation	41
3.5. KneeSIM Lab Model: Validation and Optimization	43
3.6. Virtual Clinical Trial Knee (VCTK) Model: Validation and Optimization	46
3.7. Development of Knee Performance Equations.....	51
3.8. Proof-of-Concept Study	53
CHAPTER 4 RESULTS	56
4.1. Overview	56
4.2. KneeSIM Lab Model Validation	56
4.3. VCTK Model Validation	58
4.4. Knee Performance Equations.....	61
4.5. Proof-of-Concept Study	63
CHAPTER 5 DISCUSSION.....	67
5.1. Overview	67
5.2. Simulation Model Validation.....	68
5.3. Knee Performance Equations (KPEs).....	72
5.4. Proof-of-Concept Study	74

5.5. Feasibility of a Simulation-Enhanced Tool for Robotics-Assisted TKA	77
5.6 Study Limitations.....	81
5.7. Potential Clinical Impact.....	83
CHAPTER 6 CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE STUDY	85
6.1. Conclusions.....	85
6.2. Recommendations.....	88
References.....	89
Appendix A – DOE Experimental Design.....	106
Appendix B – Python Scripts.....	111
Appendix C – Matlab Script	117

LIST OF FIGURES

Figure 1. Incidence of osteoarthritis of the hand, hip, and knee grouped by age (Zhang and Jordan, 2010).	5
Figure 2. X-radiograph of the knee showing anteroposterior view (A) and lateral view (B). Cartilage thinning is indicated by (1) and osteophyte formation is highlighted by (2) (Sinusas, 2012).	6
Figure 3: A diagram showing differences between TKA (left) and UKA (right). In UKA, only a portion of the knee joint is resurfaced (Foran, 2016).	8
Figure 4. Structural knee anatomy showing bony structures and landmarks (left) and view of the meniscus (right) (Blackburn and Craig, 1980).	9
Figure 5. Muscular tissues of the lower limbs showing key structures for knee flexion and extension (“Diagram Of The Muscles In The Leg and Leg Muscle Charts,” 2019).	10
Figure 6. View of the capsular ligaments of the knee on the proximal tibia including the anterior cruciate ligament, posterior cruciate ligament, medial collateral ligament, and lateral collateral ligament. Also visible are the patellar ligament and the horns of the menisci (Flandry and Hommel, 2011).	11
Figure 7. Anterior-posterior view of the bone cuts for total knee arthroplasty (Parcells, 2017).	13
Figure 8. Schematic drawing of three parts of a total knee arthroplasty (TKA) (metallic femoral component, metallic tibial tray, and UHMWPE tibial insert) (left) and a photograph of a selection of the instrumentation used in TKA (alignment guides and trial instrumentation) (right) (Manner, 2016; Zimmer, 2014).	15
Figure 9. Schematic drawing of a typical workflow for robotics-assisted orthopedic surgery. ..	17

Figure 10. Photograph of the MAKO RiO System: haptic arm (left), the infra-red navigation camera (center), and the guidance module (right) (Stryker, 2017).....	22
Figure 11. Navio Surgical System: the operative workflow (left), the cart and the navigation camera (right, top), and the robot-controlled burr (right, bottom) (Lonner and Kerr, 2012).	23
Figure 12. Screen shot of Navio Surgical System workflow showing stress range of motion collection (left) and intraoperative gap balancing with flexion/extension joint laxity graph (Smith & Nephew, 2018).....	29
Figure 13. Trend in number of publications per year in musculoskeletal simulation research (Hicks et al., 2015).....	30
Figure 14. Musculoskeletal simulation model developed in OpenSim for modeling deep squatting motion (Schellenberg et al., 2018).	33
Figure 15. A schema of the steps in the development of a simulation-enhanced planning tool for robotics-assisted TKA.....	35
Figure 16. Schematic drawing of LifeModeler modeling strategies for active and passive types of simulation (LifeModeler Inc., 2019).....	38
Figure 17. Data flows for various modeling strategies of the simulation using Lifemodeler.	39
Figure 18. Muscles included in the patient-specific model.	41
Figure 19. Side-by-side comparison of Kansas Knee Simulator mechanical testing rig (left) and LifeModeler’s KneeSIM Lab model (right) (Clary et al., 2013).	43
Figure 20. Workflow for developing optimized simulation parameters including forward modeling control gain, soft tissue properties, and virtual implant positions.	46

Figure 21. A schematic drawing showing VCTK motion agents (yellow), which are driven by motion capture marker trajectories and spring elements (white), which, in turn, are attached to the bone model (red). 49

Figure 22. Registration process for the bone model and the implant component model using Geomagic Studio, Matlab, and Lifemodeler for the femur and the femoral component of the implant. This process is used to create a consistent coordinate system (CSYS) for implant position analysis. This process is identical to that when the tibia and the tibial component were used. 55

Figure 23. Photograph of Subject K1L performing deep knee bend activity (left) and animation of direction and magnitude of load vectors relative to the tibial component in the case of this subject (right)(Bergmann et al., 2014)...... 57

Figure 24. Tibiofemoral joint contact loading in Orthoload dataset (single-gait phase) (Bergmann et al., 2014) and in 2014 SimTK dataset (single-leg stance) (Subject PS) (Bergmann et al., 2014; Fregly et al., 2012)...... 59

Figure 25. Tibiofemoral joint contact force magnitude versus time for single-leg stance: optimized VCTK simulation model results versus 2014 SimTK dataset results (Subject PS)..... 60

Figure 26. Tibiofemoral joint contact moment magnitude versus time for single-leg stance: optimized VCTK simulation model results versus 2014 SimTK dataset results (Subject PS)..... 60

Figure 27. Maximum medial collateral ligament (MCL) and lateral collateral (LCL) strains for each of the simulation trials used for knee performance equation development with the Optimized KneeSIM Lab simulation model. 63

Figure 28. Maximum medial collateral ligament (MCL) and lateral collateral ligament (LCL) strains for each of the simulation trials used for knee performance equation development with the

VCTK model. Note that trial #16 was excluded from the figure as some strain values obtained were $\geq 50\%$.	64
Figure 29. Comparison of kinematics results from KneeSIM Lab simulation model (V/Knee) and Kansas Knee Simulator test rig (KU simulator) (Smith and Nephew, 2015).	68
Figure 30. Femur model comparison showing GEBOD-scaled model (grey) overlaid on CT-derived model (blue) from SimTK dataset. GEBOD-scaled model is undersized by 35 mm.	72
Figure 31. Schematic diagram of workflow for a current-generation robotics-assisted total knee arthroplasty with intraoperative surgical planning (NAVIO Surgical System Surgical Technique for Total Knee Arthroplasty, 2018).	78
Figure 32. Schematic diagram of workflow plan with the Navio Surgical System but with incorporation of the proposed simulation-enhanced intraoperative surgical planning tool. Steps unique to the incorporation of SEISPT are highlighted in red.	79
Figure 33. Example surface response maps for medial collateral ligament (MCL) and lateral collateral ligament (LCL) strains relative to anterior-posterior (AP) and medial-lateral (ML) positions of the femoral component, as obtained from Optimized KneeSIM Lab knee performance equations. As the position of the femoral component is adjusted during the robotic case planning, the surgeon is given visual feedback via the maps.	81
Figure 34. Rendering of Matlab graphical user interface for implant positioning using collateral ligament strain surface response maps. Positioning of the implant in the AP and ML directions is accomplished with the use of interactive sliders. The resulting ligament strains are projected on the maps and the implant model position is updated.	117

LIST OF TABLES

Table 1. Comparison of types of robotics-assisted orthopedic surgical systems (Davies, 2015; Joskowicz and Hazan, 2018; Roche, 2015).	19
Table 2. Comparison of image-based and imageless systems for robotics-assisted total knee arthroplasty (Davies, 2015; Jacofsky and Allen, 2016; Joskowicz and Hazan, 2018; Lang et al., 2011).	20
Table 3. DOE input factors ^a and their range of values for the knee performance equations.....	52
Table 4. Influences on the responses of the output outcomes ^a of the knee performance equations	52
Table 5. Errors for mean tibiofemoral joint contact loadings: Baseline KneeSIM Lab model versus Orthoload dataset.....	58
Table 6. Errors for mean tibiofemoral joint contact loading: Optimized KneeSiM Lab model versus Orthoload dataset.....	58
Table 7. RMS error for mean baseline and optimized VCTK model tibiofemoral joint contact loading compared to mean results in SimTK dataset.....	61
Table 8: Knee performance equations developed using optimized KneeSIM Lab and SimTK Subject PS data and implant.	62
Table 9: Knee performance equations developed using optimized VCTK and SimTK Subject PS data, kinematics, ground reaction force data and implants.	62
Table 10. Implant position and orientation data comparing results achieved for Subject PS (“actual passive surgical plan”) (gray) and results suggested for Subject PS using output from the conceptual active surgical plan (optimized KneeSIM Lab model) (blue). Implant positions and orientations are those that are postulated to minimize MCL and LCL strains. Results are shown	

relative to RMS equivalent error, which is defined as $|(RMSE\ conventional) - (RMSE\ robotic)|$ (Lonner et al., 2015; Jaramaz et al., 2018)..... 65

Table 11. Implant position and orientation data comparing results achieved for Subject PS (“actual passive surgical plan”) (gray) and results suggested for Subject PS using output from the conceptual active surgical plan (optimized VCTK model) (blue). Implant positions and orientations are those that are postulated to minimize MCL and LCL strains. Results are shown relative to RMS equivalent error, which is defined as $|(RMSE\ conventional) - (RMSE\ robotic)|$ (Lonner et al., 2015; Jaramaz et al., 2018)..... 66

Table 12. Previous winning results from the SimTK Grand Challenge to Predict *In Vivo* Knee Loads. Results given as RSME (expressed relative to body weight (BW) of subject)..... 70

Table 13. Metrics of improvement in regression fit of knee performance equations, with results from optimized VCTK model. Regression and transformation was carried out using Adams Insight software..... 82

Table 14: Fractional factorial design used to develop optimized KneeSIM Lab model. 107

Table 15: Fractional factorial design used to tune muscle gains for optimized VCTK model... 108

Table 16: Fractional factorial design (Latin Hypercube) used to create optimized VCTK model. 109

Table 17: Fractional factorial experimental design used to create Knee Performance Equations for KneeSIM Lab and VCTK 110

CHAPTER 1

INTRODUCTION

Total knee arthroplasty (TKA) is indicated for patients who suffer severe and persistent knee joint pain, most commonly due to osteoarthritis and other degenerative conditions that affect the articular tissues of the knee, and for whom conservative treatments or intake of pharmacologic agents have not been effective. TKA is one of the most widely performed orthopedic procedures. For example, in the United States, the number of TKAs doubled between 1999 and 2008 and, if current trends continue, it has been projected that, by 2030, the number of TKAs will be ~3.5 million per annum (Kurtz et al., 2007; Losina et al., 2012). While much advance has been made in features that improve patient outcomes with TKA, notably, introduction of ligament-sparing implant designs, there is room for improvement in other key features, in particular, intraoperative methods for sizing implant components, proper alignment of implant components, and ligament balancing. The three lattermost-mentioned features allow existing knee anatomy to be matched even if the joint tissue is degraded and knee function is sub-optimal. In current TKA practice, the surgeon relies on preoperative patient imaging, via computed tomography (CT) or magnetic resonance imaging (MRI), to assess the size and the orientation of implant components. Manual instrumentation, including jigs, guides, and saws, are used to implant the prosthesis components. Joint function is confirmed intraoperatively prior to finalizing implant component size and orientation but this decision is guided solely by surgeon preference and experience. Furthermore, intraoperative flexion and extension of the patient's knee joint, which is manually performed by the surgeon, captures a passive range of motion (ROM); that is, one that neglects the influence of muscle forces and gravity. Therefore, during

joint balancing, the true joint dynamics of the patient's knee are unknown and, as such, there is potential for joint instability and malalignment following TKA. Joint instability is strongly associated with poor patient outcome following TKA (Banks, 2017).

Computer-assisted surgical systems were introduced to the field of TKA in 2003. Two categories of the systems are now recognized, these being computer-assisted navigation systems (CANSs) and robotics-assisted surgical systems (RASSs) (Davies, 2015; Joskowicz and Hazan, 2018; Roche, 2015). A CANS helps/assists the surgeon to 1) create a preoperative plan; 2) track patient anatomy (notably, bone orientations and limb alignments) during surgery; and 3) track instrumentation and tools during the surgery. In contrast, RASS, involves 1) using a device to perform specific surgical task(s), such as hold a jig in a predetermined position while the surgeon uses a manually-controlled tool to prepare the bone surfaces (passive type); or 2) using a device to provide tactile feedback to the surgeon allowing him/her to cut a bone (semi-active type); or 3) using a robotic arm to cut a bone with no manipulation of the cutter by the surgeon (active type). The goal of a CANS and a RASS is to reduce the number of outlier TKA cases by allowing the surgeon to accurately position the implant components relative to a preoperative plan, as well as to provide objective measurements of soft tissue balancing (Zheng and Nolte, 2015). It is hoped that use of RASSs will help reduce dependence on surgeon experience in obtaining good patient outcomes following TKA (Song and Seon, 2018). However, despite demonstrating good operative precision and accuracy, there is no consensus on whether or not, when used for TKA, current-generation CANSs and RASSs lead to better long-term patient outcomes; specifically, reduction of anterior knee pain and improved poor proprioception (Karunaratne et al., 2018). Thus, the percentage of TKA patients who are dissatisfied when the procedure was carried out

using conventional surgical techniques (by one estimate, 20% of all cases (Dunbar et al., 2013)) may not be reduced when with a current-generation CASS or RASS is used for the surgery (Dunbar et al., 2013). It is suggested here that a reason for this phenomenon is that passive patient knee joint measurements are part of the input to these systems.

Physics-based computer models of the knee are widely used for *in silico* analysis of a knee joint that contains TKA component(s). There are many commercially-available software packages, open-source platforms, and research computer programs that may be used for this purpose. One example is KneeSIM Lab (LifeModeler, Inc., San Clemente, CA), which may be used 1) to model bones, muscles, ligaments, and tendons for analysis of TKA component designs; and 2) to integrate patient CT and MRI data into the simulation environment, allowing for patient-specific estimation of implant performance (LifeModeler Inc., 2019). While widely accepted as a tool for TKA design, the use of dynamic knee joint modeling as a means of clinical assessment is underutilized, a possible reason being a limited volume of literature on its efficacy as an operative planning tool. Combining patient-specific input data that include intraoperative active joint balancing measurements with a dynamic knee joint simulation model would lead to a novel patient-specific tool for TKA that may be executed using a RASS. Hereafter, this tool is referred to as a “simulation-enhanced intraoperative surgical planning tool” (SEISPT). Such a tool may then be used to size implant components and to position and orient them in the prepared joint space. Compared to both types of methods used for TKA (conventional and current-generation CASSs and RASSs), use of SEISPT potentially has a number of advantages, notably, higher percentage of restoration of proper knee joint kinematics and substantially better clinical outcomes (translating to a marked increase in patient satisfaction).

The purpose of the present work was to develop and evaluate SEISPT. The specific tasks were: 1) develop and validate two robust musculoskeletal simulation models that include the knee joint but of different fidelities; 2) incorporate outputs from each of these models into TKA planning algorithms, thus yielding SEISPT; 3) in a proof-of-concept study, determine if the calculated optimal position and orientation of TKA implant components when SEISPT was used was different from the corresponding values achieved when a conventional TKA technique was used; and 4) indicate how SEISPT could be deployed on a specified current-generation robotics-assisted TKA (raTKA) system with minimal disruption to the established workflow.

CHAPTER 2

BACKGROUND

2.1. Osteoarthritis of the Knee and Management/Treatment Modalities

Osteoarthritis (OA) is the most common joint disease, especially in persons aged 60 years and older. For example, in the United States, the incidence of OA in this age group is about 50% higher in women than in men (Figure 1). The disease is characterized by the thinning and erosion of the lubricious cartilage that covers the articular joints (Maclean, 2016; Zhang and Jordan, 2010). The etiology of OA is multifactorial, with some predisposing factors being old age, female gender, high body mass index, knee injury, repetitive joint use, low bone density, and poor joint laxity. Usually, symptomatic OA is manifest by the presence of pain, joint stiffness, and joint dysfunction, but radiographic diagnosis (Figure 2) is the gold standard for determining the proper management/treatment method (Sinusas, 2012; Zhang and Jordan, 2010).

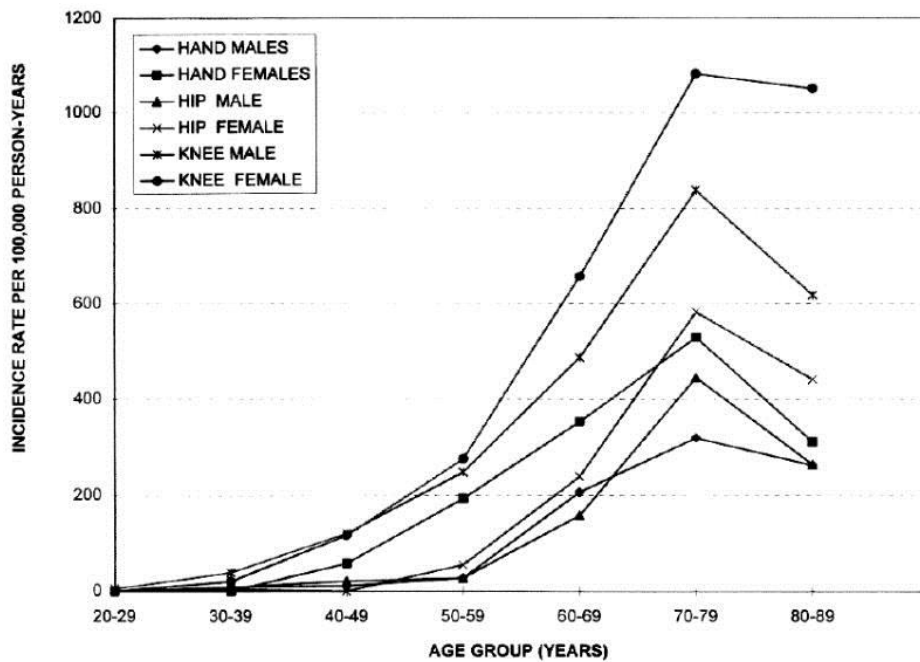


Figure 1. Incidence of osteoarthritis of the hand, hip, and knee grouped by age (Zhang and Jordan, 2010).



Figure 2. X-radiograph of the knee showing anteroposterior view (A) and lateral view (B). Cartilage thinning is indicated by (1) and osteophyte formation is highlighted by (2) (Sinusas, 2012).

Management or treatment modality for OA depends on the severity of the pain caused by the disease. For patients with mild OA, conservative methods include physical therapy and exercises to reduce body weight. Also, over-the-counter non-steroidal, anti-inflammatory drugs, such as acetaminophen, ibuprofen, and naproxen, may be prescribed (Sinusas, 2012). For patients with moderate OA who do not respond to conservative methods, naturally-occurring supplements, such as glucosamine and chondroitin sulfate, either in pill form or as an injection, may be prescribed (Clegg et al., 2006; Towheed et al., 2009). For patients with severe OA but with a low pain burden, commonly-used treatment options are prescription of an opioid, injection of a corticosteroid, or injection of hyaluronic acid (a natural component of synovial fluid) (Kon et al., 2011). For such patients, more recently, researchers have experimented with

injections of mesenchymal stem cells into the joint space to stimulate cartilage rejuvenation (Jo et al., 2014; Kon et al., 2011).

In cases where the pain due to severe OA is excessive, persistent, and resistant to the aforementioned treatment methods (so-called “end-stage OA” patients), the usual recourse is a surgical procedure, namely, partial replacement of the knee joint (most often, unicompartmental knee arthroplasty (UKA)) or TKA, with the latter being the more common procedure. In the United States, the annual number of TKAs has increased from 291,796 in 1999 to 615,050 in 2008 and, if this trend continues, it is expected the annual number of cases by 2030 would be ~3.5 million (Kurtz et al., 2007; Losina et al., 2012).

For cases in which end-stage OA is localized to one side of the knee and the patient’s soft tissues are healthy, a UKA may be implanted (Figure 3). UKA differs from TKA in that only one of the two tibiofemoral compartments is replaced; in most cases, on the medial side of the knee. UKA has several advantages compared to TKA; for example, 1) the soft tissues around the knee are preserved, which makes the procedure ideal for young patients who wish to return to an active lifestyle (Heyse et al., 2012); and 2) UKA is less invasive and is associated with shorter recovery time, less blood loss, and lower medical morbidity (Scott, 2003). However, UKA prostheses are difficult to implant in a consistent orientation, which may contribute to their lower long-term survival rates compared to TKA (Scott, 2003). Although robotics-assisted UKA provides improved implant component placement accuracy (Weber et al., 2013), as recently as 2017, only ~2% of all primary knee replacement procedures the United States were UKAs (American Joint Replacement Registry, 2018). For this reason, in the present study, the focus was on TKA.

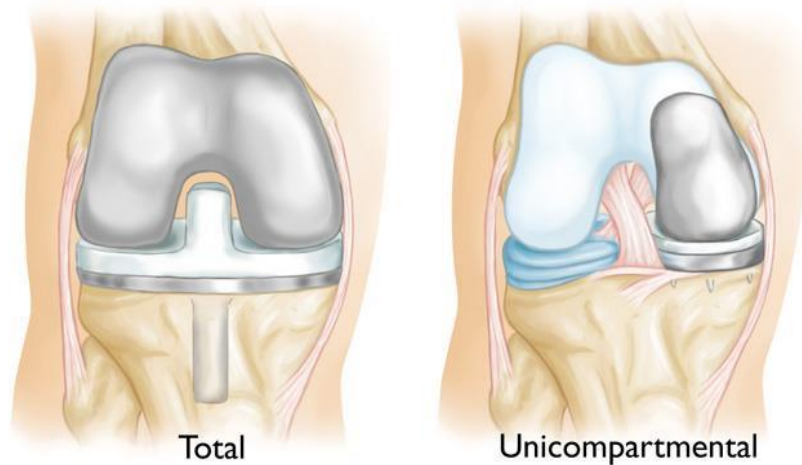


Figure 3: A diagram showing differences between TKA (left) and UKA (right). In UKA, only a portion of the knee joint is resurfaced (Foran, 2016).

2.2. Knee Anatomy and Physiology

The knee is a multi-compartment structure of three osseous tissues (the femur, the patella, and the tibia (Figure 4)) that interact between the patellofemoral joint and the tibiofemoral joint. The tibiofemoral joint is further divided into medial and lateral compartments. The distal femur has medial and lateral condyles, which interact with corresponding sides of the tibial plateau. The fibrous meniscus on the tibial plateau forms a deepened “seat” to receive the respective femoral condyles. The added depth from the meniscus is critical to knee function because the articular surfaces of the tibial plateau are convex. On the anterior aspect of the femur is the patellar groove, which is shaped to receive and guide the patella during joint articulation (Blackburn and Craig, 1980; Drake et al., 1989; Thompson and Netter, 2010).

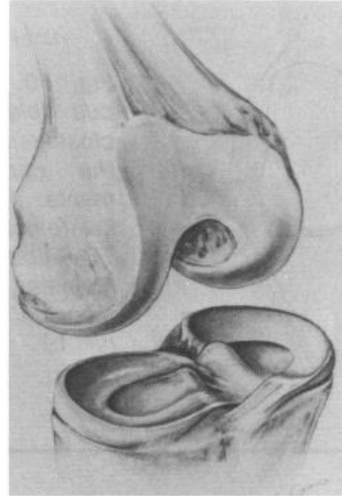
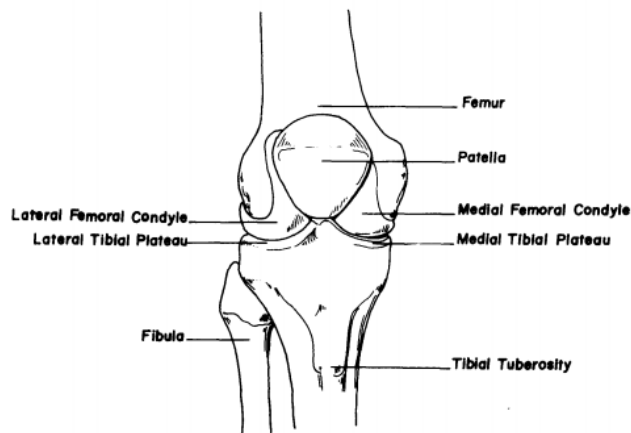


Figure 4. Structural knee anatomy showing bony structures and landmarks (left) and view of the meniscus (right) (Blackburn and Craig, 1980).

The extensor mechanism of the knee is composed of six muscles (rectus femoris, vastus intermedius, vastus lateralis, vastus medialis longus, vastus medialis obliquus, and articularis genu), the quadriceps femoris tendon (QFT), and the patellar ligament (Figure 5). The patella ligament provides critical mechanical advantage for extension of the knee by acting as the mechanical pulley for the quadriceps and changing the direction of the extension forces throughout knee flexion (Blackburn and Craig, 1980; Loudon, 2016). Other pertinent muscle structures include the semitendinosus, biceps femoris and semimembranosus

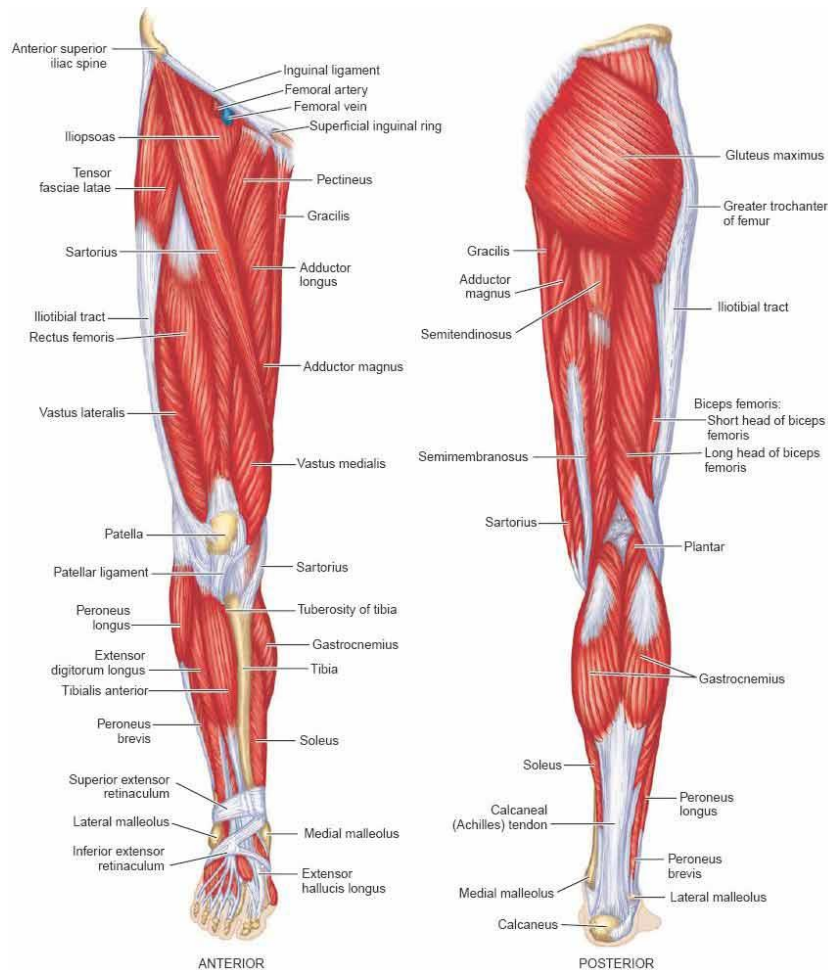


Figure 5. Muscular tissues of the lower limbs showing key structures for knee flexion and extension (“Diagram Of The Muscles In The Leg and Leg Muscle Charts,” 2019).

Knee flexion, extension, and stability are further supported by several soft tissues (Drake et al., 1989; Flandry and Hommel, 2011; Thompson and Netter, 2010) (Figure 6). On the medial and lateral aspects of the joint are the collateral ligaments. The medial collateral ligament (MCL) is attached proximally to the medial epicondyle of the femur and distally to the medial condyle of the tibia. Its primary function is to resist forces that would push the knee medially or into a valgus deformity. The lateral collateral ligament (LCL) is attached proximally to the lateral condyle of the femur and distally to the head of the fibula. Its primary function is to resist

forces that would move the knee laterally into a varus deformity. Other ligaments in the knee are cruciate ligaments. The anterior cruciate ligament (ACL) originates from the posterolateral surface of the intercondylar notch of the femur and is directed anteriorly and distally to insert on the anterior surface of the intercondylar eminence. The primary function of the ACL is to prevent the tibia from moving anteriorly off the femur. Additionally, the ACL prevents hyperextension of the joint. The posterior cruciate ligament (PCL) originates from the medial surface of the femoral intercondylar notch and is directed distally and posteriorly to insert on the proximal tibia. The primary function of the PCL is to prevent the tibia from displacing posteriorly relative to the femur. Other ligamentous tissues in the knee joint include the popliteal tendon, the capsular ligament, and the patellar ligament.

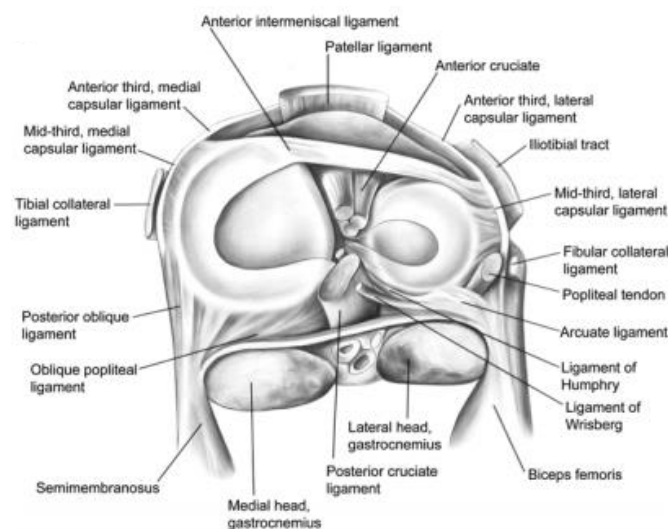


Figure 6. View of the capsular ligaments of the knee on the proximal tibia including the anterior cruciate ligament, posterior cruciate ligament, medial collateral ligament, and lateral collateral ligament. Also visible are the patellar ligament and the horns of the menisci (Flandry and Hommel, 2011).

2.3. Total Knee Arthroplasty

2.3.1. Overview

During TKA, the surgeon must resurface the articular structures of the joint, ensure that limb alignment is maintained, correct any deformity, replace the diseased cartilage surface with prostheses, and preserve as much of the osseous structures and soft tissues as possible. This ensures maintenance of function of the knee joint and allows biomechanical function to be performed easily, especially if the ACL and/or PCL are removed. However, given the complex anatomy of the knee joint, it is very difficult to achieve all the stated goals of TKA in a given patient, making TKA a very challenging procedure. In current clinical practice, TKA may be performed using a conventional technique or with the aid of a RASS.

2.3.2. Conventional TKA

Conventional techniques used in TKA utilize manual instrumentation and cutting guides to determine the location of the implant components. Often, the surgeon aligns the components relative to the anatomical and functional axes of the limbs of the patient and, for this purpose, utilizes prominent anatomical landmarks as reference. Usually, the surgeon resects the bony tissues at the distal end of the femur and the proximal end of the tibia (Figure 7). Optionally, the patella may be resurfaced. As these bone cuts dictate the location and orientation of the implant components, they are done with extreme care and precision to ensure that the intended implant component alignment is achieved. In conventional TKA, alignment guides are used to control the bone cut trajectory. These guides are pinned in place and are located with the use of various alignment instrumentation. The distal cut of the femur sets the varus-valgus orientation of joint restoration and, thus, the overall alignment of the limb. Typically, up to 5° of valgus is applied to

the distal femoral cut to achieve the desired cosmetic appearance and to prevent the thighs from rubbing together. Anterior and posterior cuts dictate the internal-external rotation of the implant components. In most TKAs, 3° of external rotation relative to the posterior condylar axis is normal. Chamfer cuts are made at an oblique angle to accept the femoral implant. The tibial cut removes the entire proximal tibia and, therefore, internal-external rotation is applied when the implant component is positioned. For most patients, a posterior slope of 2°-3° is applied (Brooks, 2009; Gromov et al., 2014; Schiraldi et al., 2016).

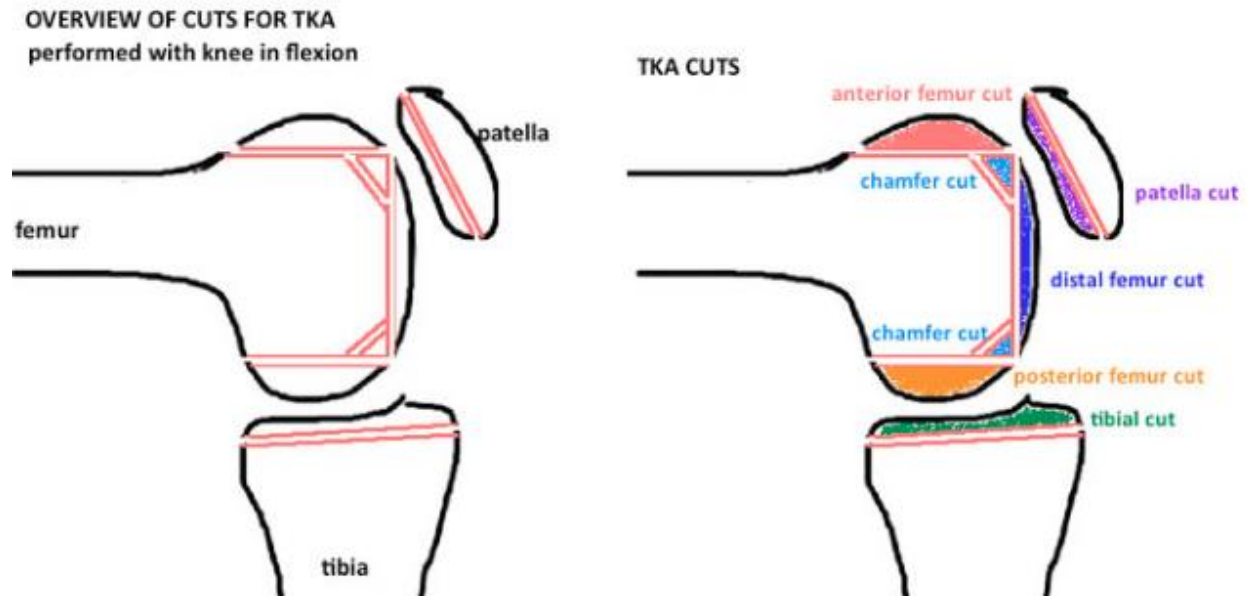


Figure 7. Anterior-posterior view of the bone cuts for total knee arthroplasty (Parcells, 2017).

There are a number of different methods for determining the proper location and orientation of the bone cuts, with two widely-used ones being measured resection (also known as mechanical alignment) and gap balancing. With the measured resection method, the bone cuts are made independent of initial soft tissue tension. In most cases, the cut planes are aligned

relative to three anatomical landmarks: the transepicondylar axis, the anteroposterior axis (or Whiteside's line), and the posterior condylar axis. The femoral component is placed parallel to the transepicondylar axis and its rotation is set relative to the Whiteside's line and the posterior condylar axis. Once the desired flexion and extension resections are achieved, the surgeon performs soft tissue release to complete the balancing of the joint (Daines and Dennis, 2014; Schiraldi et al., 2016; Schwartz et al., 2017; Sheth et al., 2017; Varacallo and Johanson, 2018). With the gap balancing method, soft tissue is released prior to making any bone cuts. This brings the limb into appropriate alignment before the rotation of the femoral component is set. During gap balancing, the knee is balanced in full extension and then in flexion or vice versa. The goal is to create a gap that is equal in both flexion and extension, ensuring that the tibial component is stable throughout the entire range of motion. For gap balancing, specialized instrumentation can be utilized for measuring soft tissue tensions in the joint (Churchill et al., 2018; Daines and Dennis, 2014; Sheth et al., 2017; Varacallo and Johanson, 2018).

With either the measured resection or gap balancing method, implant size and position are confirmed by temporary implantation of trial implant components. With the trial implant component in place, the surgeon exercises the knee through the range of motion to assess balance and stability. At this point, additional bone resection may be made or soft tissue release may be applied. The final implant components are anchored in the joint space using poly (methyl methacrylate) bone cement. The parts of a TKA are a metallic femoral component, a metallic tibial baseplate, an ultra-high-molecular weight polyethylene (UHMWPE) articular insert/tray (Figure 8), and an UHMWPE patellar cap. Often, the femoral component is fabricated from Co-Cr-Mo alloy and is highly polished on the articular surfaces and the tibial component is fabricated from Ti-6Al-4V alloy (Harris, 2019; Manner, 2016). Optionally, a

coating of a ceramic (such as titanium nitride) may be deposited on the articular faces of the femoral component and/or tibial baseplate to reduce friction and increase wear resistance. Additionally, recently, implant manufacturers have offered soft-tissue-sparing implant components, which allow the ACL and/or PCL not to be cut during surgery (Tsai et al., 2019).



Figure 8. Schematic drawing of three parts of a total knee arthroplasty (TKA) (metallic femoral component, metallic tibial tray, and UHMWPE tibial insert) (left) and a photograph of a selection of the instrumentation used in TKA (alignment guides and trial instrumentation) (right) (Manner, 2016; Zimmer, 2014).

2.3.3. Robotics-Assisted TKA

The alignment and balancing principles utilized in conventional TKA are the basis for those used in robotics-assisted TKA. The addition of navigation capabilities to such a system allows the surgeon to plan precise locations of bone cuts prior to removing bony tissue. While it is common practice in conventional TKA for a surgeon to utilize a template for sizing and positioning the implant components using preoperative radiographic images, there are limited means to transfer the preoperative plan into the operating room. A RASS facilitates this transfer and, as such, helps the surgeon to execute the planned bone resections with a high degree of accuracy.

The history of the use of CANSs and RASSs in orthopedic surgery is very short, having been introduced in 1992 with the development of ROBODOC surgical robot at the IBM T. J. Watson Research Center at the University of California, Davis. In 1995, this system was commercialized by Curexo Technologies (Joskowicz and Hazan, 2018). The ROBODOC system was indicated for total hip arthroplasty (THA) and was used to plan and size the acetabular cup as well as to prepare a cavity in the femur to accept the femoral stem. In 2000, the indication for the system was expanded to include TKA, with an application where the robot was used to prepare the femoral and tibial cuts (Joskowicz and Hazan, 2018). In the early 1990s, several surgical navigation systems were developed for pedicle screw insertion in spinal surgery. These systems pioneered real-time instrument position tracking with the use of infrared (IR) optical navigation (Merloz et al., 1998; Nolte et al., 1995) that, in later years, would be used in RASSs for TKA. Since 2008, there has been a surge in the number of CANSs and RASSs developed by commercial entities for use in TKA (Joskowicz and Hazan, 2018). Examples of these CANSs are KneeAlign2 (OrthoAlign, Inc., Aliso Viejo, CA, USA) and Vector Vision (Brain LAB, Munich, Germany) and examples of these RASSs are iBLOCK (OMNILife Science, East Taunton, MA, USA), MAKO RiO system (Stryker, Fort Lauderdale, FL, USA), Navio Precision Freehand Sculpting System (Smith & Nephew, Memphis, TN, USA), and Rosa Knee (Zimmer, Warsaw, IN, USA).

For many current-generation RASSs, the essential operative workflow is the same across many platforms. As such, a schematic drawing of a typical workflow is presented in the Figure 9. To begin with, specialized instrumentation that can be navigated by the RASS is required. Utilizing the navigated instruments, the surgeon characterizes and locates the anatomy of the patient and provides a coordinate system to develop the surgical plan and to execute the surgery.

An additional benefit of a RASS is that it provides for controlled bone resection, making the bone removal process semi-automated. Intraoperative navigation allows for more insightful implant trialing and joint balance assessment, with an option to assess the quality of the surgery after the final implant components have been placed.

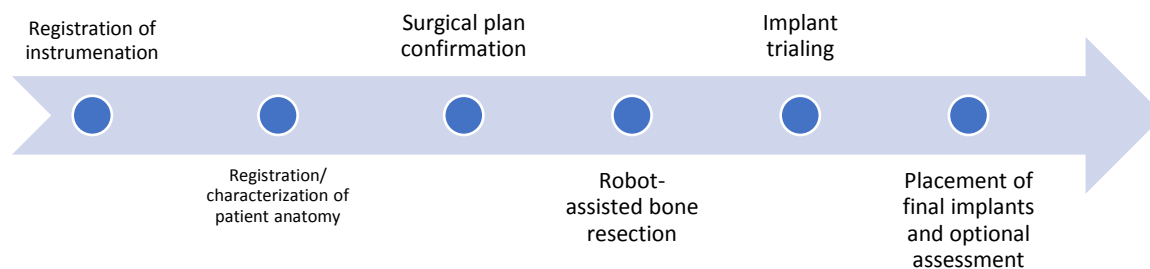


Figure 9. Schematic drawing of a typical workflow for robotics-assisted orthopedic surgery.

2.3.4. Elements of a Robotics-Assisted Surgical System

The main elements of a RASS are the virtual object, the navigation system, the user interface, and the robotic instrumentation. In orthopedic procedures, the virtual object is a three-dimensional (3D) representation of the patient anatomy. For TKA, this includes the femur and the tibia. The virtual object may be generated utilizing 3D medical imaging, such as CT or MRI or, alternatively obtained using a statistical modeling technique (Leung and Vyas, 2014; Pastides and Nathwani, 2017; Zheng and Nolte, 2015).

The navigation system allows the user to register the position of the patient anatomy to the virtual object. This provides a mutual coordinate system for the virtual object and the operating room. Changes in position of patient anatomy can be reflected in the coordinate reference frame of the virtual object as well as in updates to the bone shape made during the

surgery. Additionally, registration of the virtual object and the patient anatomy allows the surgeon to localize the position of tracked surgical instruments relative to the patient and, therefore, provides various levels of guidance and automation (Netravali et al., 2013; Zheng and Nolte, 2015). In the case of raTKA, navigated systems with advanced instrumentation are used to precisely control bone cuts. There are several navigation modalities that are in use, with the most common one being optical navigation with the use of reflective markers. Optical navigation is built on the principles of stereovision, in which a stereo infrared camera system is used to track an array of reflective markers, thereby providing the 3D coordinates of the array. This allows the surgeon to track the position of an instrument or a patient in 3D space and in real time (Ewurum et al., 2018; Zhou et al., 2017).




In the majority of RASSs, the user interface is a screen-based solution. Screen systems can easily be transported to the operating room and can be placed adjacent to or in the sterile field. Screen systems allow the user to interact with the surgical workflow and provide visual feedback of surgical progress and performance. The surgical plan is executed with the use of robotic-controlled instrumentation.

2.3.5. Types of Robotics-Assisted Surgical Systems

Design details of RASSs vary greatly between manufacturers but may be categorized into one of three types: passive, active, or semi-active. A passive system can be thought of simply as a navigation device. With a passive system, control of the surgical workflow is entirely in the hands of the surgeon. Such a system does not feature active control of instrumentation or possess any haptic guidance. A passive system allows for preoperative or intraoperative surgical planning and intraoperative navigation. An active system is one that features fully automated

execution of the surgical plan. In other words, the system allows the surgeon to generate and approve an operation plan but preparation of the bone to receive the implant components is out of his/her control. A semi-active system has features of both a passive system and an active system. That is, it allows for preoperative planning and instrument navigation but surgical plan execution is neither fully automated nor fully manual. For example, bone cuts are made by utilizing haptic guidance or with the use of smart instrumentation that allows the surgeon to precisely implement the surgical plan (Davies, 2015; Roche, 2015; van der List et al., 2016). Selected features of these three types of systems are presented in Table 1.



Table 1. Comparison of types of robotics-assisted orthopedic surgical systems (Davies, 2015; Joskowicz and Hazan, 2018; Roche, 2015).

	Passive System	Active System	Semi-Active System
Advantages	Navigation Joint balancing 25-year clinical history Low cost	Fully automated Precise and repetitive movements	Navigation and robotic assistance Joint balancing Sensor integration
Shortcomings	Surgeon-controlled, no assistance	High cost Regulatory barriers	Surgeon-controlled High cost
Example systems	ROBODOC DaVinci 	Nav3i Orthopilot 	Navio MAKO RiO OmniLife 

RASSs may also be classified on the basis of reliance or otherwise on medical image data. A system that relies on medical image data (CT or MRI) to generate the operative virtual object is classified as an “image-based” system. With such a system, in theory, the entire surgery

can be planned prior to entering the operating room. A system that does not require medical image data is classified as an “imageless system”. Such a system requires intraoperative planning. For such a system, the patient anatomy is characterized while the patient’s joint is exposed on the operating table. In such a system, it is common to use a statistical atlas model of the bony anatomy to generate the operative virtual object. Key landmarks and bone surface data, which are acquired intraoperatively, are used to deform the atlas model to fit the anatomy of the patient. While an imageless system requires an elaborate intraoperative workflow, patients avoid costly preoperative imaging and associated radiation exposure (Davies, 2015; Jacofsky and Allen, 2016; Joskowicz and Hazan, 2018; Lang et al., 2011). Selected features of image-based and imageless systems are summarized in Table 2.

Table 2. Comparison of image-based and imageless systems for robotics-assisted total knee arthroplasty (Davies, 2015; Jacofsky and Allen, 2016; Joskowicz and Hazan, 2018; Lang et al., 2011).

	Image-based System	Imageless System
Advantages	Preapproved operative planning Landmark registration Improved accuracy	Decreased cost of procedure Patient convenience Improved accuracy
Shortcomings	Preoperative images required Increased cost Increased radiation exposure	Relies on mapping No true preoperative plan
Example systems	MAKO RiO 	Navio 

2.3.6. Features of Two Robotics-Assisted Total Knee Arthroplasty Systems

The two systems that are widely used for raTKA are the MAKO RiO o and Navio Surgical Systems. Some key features of these are now presented.

MAKO RiO System

The key feature of the MAKO RiO System (Figure 10) is a robotic arm that provides haptic feedback to the surgeon. This arm is mounted to a large base that can be positioned next to the operating table. In addition to the robotic arm, the system features an IR optical navigation camera and a guidance module to interact with the system. During TKA, arrays of reflective markers are fixed to the patient's tibia and femur with bone pins. An array mounted to the base of the robotic arm allows the system to track the arm's position relative to the patient.

System planning relies on preoperative CT imaging to generate a surgical plan. Surgical planning may be done preoperatively with a CT-derived model or done intraoperatively. Image data are segmented by System imaging specialists, who work with the surgeon to develop the preoperative plan. During the surgery, the surgeon must confirm the operative plan before execution. To register the virtual bone model to the patient's anatomy, a tracked probe is used to locate 40 registration points on the femur and 40 registration points on the tibia. The planning software allows for intraoperative planning with both measured resection and gap balancing techniques. To execute the bone cuts, a sagittal saw mounted to the end of the robotic arm is utilized. The surgeon guides the saw into position utilizing haptic feedback from the robotic arm. Once in position, movement of the arms is constrained to the plane of the planned bone cuts. This provides a virtual cutting guide to the surgeon, only allowing translation of the saw in one direction (Stryker, 2017).

Use of the haptic robotic arms in the System results in implant position that, typically, is accurate within 2° of that specified in the operative plan (Bell et al., 2016; Ganko et al., 2017; Rauck et al., 2018). There is lack of consensus on the clinical impact of the System but some studies suggest that use of the System results in increased patient-reported outcome measures (PROMs), such as American Knee Society Scores (Jacofsky and Allen, 2016).



Figure 10. Photograph of the MAKO RiO System: haptic arm (left), the infra-red navigation camera (center), and the guidance module (right) (Stryker, 2017).

Navio Surgical System

The Navio Surgical System (Figure 11) is an imageless system. As such, with the System, the surgery is planned intraoperatively after the anatomical virtual object is created by mapping the bone surface with a tracked probe. Key anatomical landmarks and surface mapping data are used to deform a statistical atlas model of the femur and tibia. This atlas model is the basis for procedural planning, which may be carried out using either the measured resection or

gap balancing method. In addition, the System features soft tissue characterization by collecting ROM data in stressed (varus/valgus) and un-stressed states. Implant positioning is done virtually prior to making any bone cuts, and the effect of the implant position on ligament balancing is displayed as a visual read-out on the System screen.



Figure 11. Navio Surgical System: the operative workflow (left), the cart and the navigation camera (right, top), and the robot-controlled burr (right, bottom) (Lonner and Kerr, 2012).

The System allows for bony resection with the use of the robotic-controlled burr, or, else, the burr can be used to create positioning features for a conventional style cutting guide

(Jaramaz et al., 2013; Smith & Nephew, 2018). The System can be operated in speed- and exposure-control modalities, where the rotational speed or position of the burr is modulated depending on the tool tip position relative to the patient. Robotic control of the burr prevents the surgeon from cutting bone outside the envelope defined during surgical planning. The System features IR optical navigation. During TKA, reflective tracking arrays are secured to the patient's tibia and femur with bone pins. Corresponding arrays on the robotics-controlled burr and the instruments allow for navigated execution of the surgical plan (Smith & Nephew, 2018).

With the System, mean angular alignment, rotational, and translational errors are reported to be 1.46°, 3.2° and 1.18 mm, respectively, relative to the corresponding value stipulated in the operative plan (Lonner and Kerr, 2012). In addition, alignment of an implant component relative to the patient's mechanical axis has been reported to be within 1° (Jacofsky and Allen, 2016). The clinical impact of the System is a subject of much debate but some studies indicate that when the System is used, there is improvement in PROMs, such as increase in Oxford Knee Score (Jacofsky and Allen, 2016).

2.4. Patient Outcomes with Total Knee Arthroplasty

2.4.1. Conventional Total Knee Arthroplasty

Even though ~5 million Americans have undergone TKA, clinical outcomes of the procedure remain disputed (Kremers et al., 2014). Patient dissatisfaction rates have been commonly reported as high as 20% (Bryan et al., 2018; Choi and Ra, 2016; Dunbar et al., 2013; Kahlenberg et al., 2018; West et al., 2019). Usually, patient satisfaction is determined using one or more objective scoring systems. The majority of satisfaction measurement tools are patient surveys also known as PROMs, with the widely used ones being the Knee Society Scoring System (KSS), the 36-item Short Form Health Survey (SF-36), the 12-item Short Form Health Survey (SF-12), the Western Ontario and McMaster Universities Arthritis Index (WOMAC), the Knee injury and Osteoarthritis Outcome Score (KOOS), and the Oxford Knee Score (OKS) (Bellamy et al., 1988; Dawson et al., 1998; Roos et al., 1998; Scuderi et al., 2012; Stewart et al., 1988; Zavatsky, 1997). Usually, these surveys are administered both before and after TKA and, essentially, they are records of various characteristics of the patient, including pain level, mobility, physical function, and mental health.

Reduced patient satisfaction in TKA is a widely-discussed subject in the joint replacement literature. In a study by Parvizi et al. (2015) that examined 661 TKA recipients, 91% of patients were satisfied with pain relief and 89% of patients were satisfied with their ability to complete daily tasks. However, only 66% of patients reported that their knees felt normal (a characteristic referred to as “normal proprioception”). Additionally, 33% of patients had persistent pain, 41% exhibited joint stiffness, and 33% complained of noise, tightness or swelling. 38%, 31%, and 54% of patients reported dissatisfaction with getting in and out of an automobile, standing from a chair, and climbing stairs, respectively.

Patient satisfaction after TKA is multifactorial, with a number of relevant internal and external predictors. Internal predictors (that is, factors that are outside the control of the implant components manufacturer and the surgeon) include patient age, gender, personality, preoperative expectations, existing comorbidities, and severity of OA (Choi and Ra, 2016). External predictors (that is, factors that are within the control of the aforementioned actors) include implant component design, surgical technique, postoperative pain management method, and rehabilitation protocol (Choi and Ra, 2016).

2.4.2. Robotics-Assisted Total Knee Arthroplasty

There is lack of consensus on patient outcomes following raTKA. The foremost point of contention is whether improved accuracy obtained with raTKA translates to improved PROMs. Results of both short-term and long-term studies do not provide a definitive answer. Some workers reported significantly higher 6-month PROMs and less post-operative pain for the raTKA group compared to the conventional surgery group (Kayani et al., 2018; Marchand et al., 2017). In a study of 29 and 31 patients in the conventional TKA and raTKA groups, respectively, with mean follow-up of 2 years, Liow et al. (2017) reported no significant increases in KSS, OKS, or SF-36 for the patients in the latter group. In a study of 42 and 71 patients in the conventional TKA and raTKA groups, respectively, with mean follow-up of 10 years, Yang et al. (2017) found no significant difference in HSS score, WOMAC score, or postoperative ROM between the groups. Kim and Park (2019) reported that, in a 124-patient study, with mean follow-up of 13 years, there was no significant difference in KSS score, WOMAC score, knee ROM, UCLA activity score, and radiographic parameters of implant component alignment between raTKA-group patients and conventional TKA-group patients. Another study that

involved comparison of 196 conventional TKA cases and 155 raTKA cases, with a mean follow-up of 11 years, found no significant difference between the groups on the basis of KSS, HSS, WOMAC, and SF-12 PROM scores (Na et al., 2019). There are various opinions as to why raTKA does not consistently provide improved patient outcomes. These include complications associated with prolonged operative time and technical challenges that are experienced during cases that are performed early in the surgeon's learning curve.

It has been well demonstrated that prolonged operative time correlates with higher complication rate and increased risk for infection (Peersman et al., 2006; Pulido et al., 2008). One study showed that the risk for infection nearly doubles for surgeries lasting longer than 120 minutes (Pugely et al., 2015). However, various workers have demonstrated that there is no significant difference in operative time between raTKA and conventional TKA (Kayani et al., 2018; Song et al., 2011). In fact, Koulalis et al. (2011) reported that the time required for femoral component preparation in raTKA is less than half that required when a conventional technique is used.

The optimal learning curve associated with raTKA has been defined in several studies. Most research indicates that after novice surgeons perform 15-20 cases, surgical time reduces drastically; for example, Siebert et al. (2002) reported a reduction from 135 minutes to 90 minutes and Coon et al. (2016) reported a reduction from 120 minutes to 40 minutes, but, in neither of these reports was the number of cases required to achieve the time reduction stated. In a systematic review, Khlopas et al. (2018) reported that 15 is the mean number of cases required to achieve parity in operative time between raTKA and conventional TKA but Sodhi et al. (2018) reported that this number is likely to be closer to 40 cases.

2.5. Passive Joint Balancing

It has been demonstrated that there is a strong correlation between both knee stability and proprioception after TKA, on one hand and patient satisfaction on the other (Jaramaz et al., 2018; Moro-Oka et al., 2007; Stiehl et al., 2000). This correlation is even further enhanced for young patients who plan to return to an active lifestyle as soon as possible. Restoration of knee stability and patient proprioception rely on proper balance of the joint ligaments. In conventional TKA, biomechanical assessment and joint balancing are done by exercising the leg through a complete ROM. RASSs provide insights on soft tissue balancing; for example, with the Navio Surgical System, the soft tissues are characterized by recording ROM in both neutral and varus/valgus stressed states. This information is used to model the laxity of the joint, which is displayed in a graph showing gap balance information for medial and lateral condyles throughout the entire ROM (Figure 12). The graph display is updated in real-time as the surgeon adjusts the operation plan. Furthermore, if the surgeon chooses to do a soft tissue release to balance the joint, joint laxity can be re-measured during the procedure (Smith & Nephew, 2018). A systematic review of 11 studies examining gap balancing found that patients who underwent TKA using a CANS or a RASS demonstrated better outcomes than those for whom a conventional procedure was used (van der List et al., 2016).

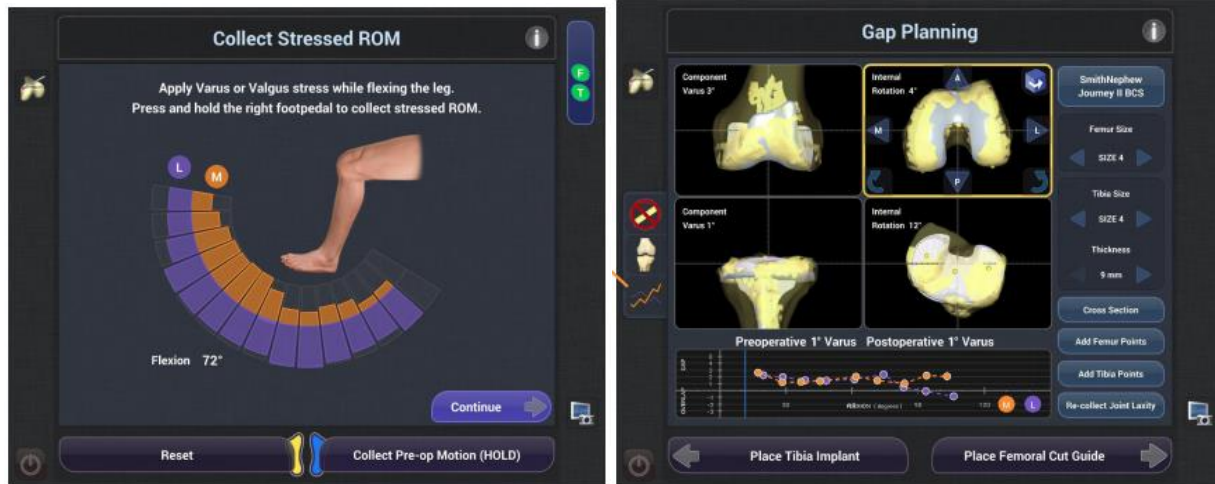


Figure 12. Screen shot of Navio Surgical System workflow showing stress range of motion collection (left) and intraoperative gap balancing with flexion/extension joint laxity graph (Smith & Nephew, 2018).

In both conventional and raTKA, gap balancing is done in a passive state. Joint laxity, ROM, and biomechanical assessment are done with the patient on the operating table under anesthesia. In other words, the influence of active soft tissue forces and, even, gravity are not considered. Furthermore, the intraoperative passive assessment is done prior to closing the wound and moving the patella into its postoperative position. As patellar position has been shown to significantly influence gap balancing in TKA, laxity assessment with the patella in a non-physiologic state may skew gap balancing results (Gejo et al., 2010). In light of the assumptions made during intraoperative gap balancing, it is not surprising that the correlation between intraoperative and postoperative joint kinematics for TKA is not significant (Ishida et al., 2015; Wada et al., 2017).

The disparity between intraoperative and postoperative joint laxity and biomechanics presents a particular challenge to the surgeon. It is possible to characterize active joint biomechanics using techniques such as mobile fluoroscopy and radiostereometric analysis. However, these imaging techniques are time-consuming, expensive, and expose the patient to

radiation. Furthermore, in the context of TKA, any preoperative assessment of biomechanics would be done on a pathologic patient, and, as such, would have limited clinical utility. Ideally, the surgeon should characterize the active behavior of the patient's joint before onset of pathology and, then, when the patient requires TKA, use a surgical plan that ensures that the joint is restored to its pre-pathology level.

2.6. Musculoskeletal Simulation Modeling

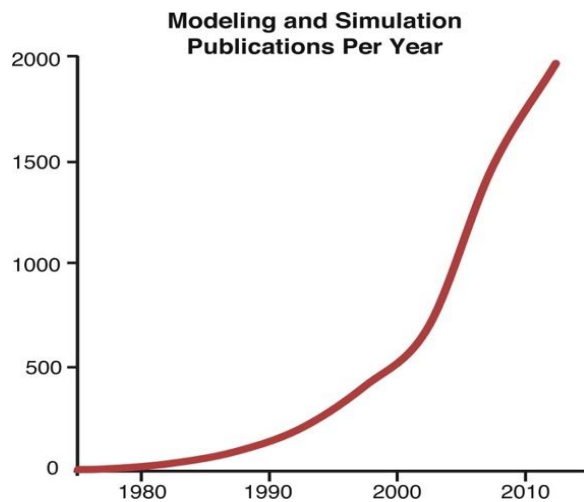


Figure 13. Trend in number of publications per year in musculoskeletal simulation research (Hicks et al., 2015).

An attractive means to characterize active joint biomechanics both before and after TKA is with the use of advanced simulation techniques. Research in simulating the musculoskeletal system has a very long history (> 40 years), but advances in computing and computational techniques in recent years have led to increased interest in this field. This is evidenced by the sharp rise in publications per year on this topic (Figure 13) (Hicks et al., 2015). It is pointed out

that the publications referred to in Figure 13 are those related to all biomechanical and musculoskeletal simulations, but it is reasonable to assume that, over the period considered, publications relating to simulation of the knee joint increased at about the same pace.

Various modeling techniques have been demonstrated to simulate the biomechanics of the knee in the presence of TKA (Gibbons et al., 2019; Kang et al., 2017; Knarr et al., 2016; Marra et al., 2017; Navacchia et al., 2019, 2016; Rasnick et al., 2016; Schellenberg et al., 2018,

2016; Shu et al., 2018; Smith et al., 2016). Popular techniques include modeling tissue mechanics with the finite element analysis method and multibody dynamic simulation. To understand the biomechanics associated with various TKA techniques, multibody dynamic simulation is particularly useful. Multibody dynamic simulation aims to characterize and analyze rigid and flexible bodies that move relative to one another. Analysis of these moving bodies must consider the forces that are applied externally as well as the forces that are generated internally. In multibody dynamic simulation of a physical system, a mathematical representation of the system is created through the equations of motion. The equations of motion allow for calculation of the position, velocity, and acceleration of a body (Hicks et al., 2015). The governing equations of motion or kinematics equations are represented by the following four equations:

$$v = v_0 + a \quad (1)$$

$$\Delta x = \left(\frac{v+v_0}{2} \right) t \quad (2)$$

$$\Delta x = v_0 t + \frac{1}{2} a t^2 \quad (3)$$

$$v^2 = v_0^2 + 2a\Delta x \quad (4)$$

where v is instantaneous velocity, v_0 is starting velocity, a is acceleration, t is time, and Δx is displacement. Depending on the desired output of the simulation, these equations can be used for an inverse kinematics analysis. Inverse kinematics simulations can be extended to understand the influence of internal and external forces and moments with the addition of the Newton-Euler equations for inverse dynamics analysis:

$$F = ma \quad (5)$$

$$M = Ia \quad (6)$$

where F is the sum of forces on the body, m is body mass, a is body acceleration, M is sum of moments on the body, and I is the body mass moment of inertia (Borese et al., 2002).

There are various commercially-available software packages that may be used for musculoskeletal simulation, examples being Anybody (Anybody Technology A/S, Aalborg, Denmark), OpenSim (NCSRR, Stanford, CA, USA) (Figure 14), LifeModeler (LifeModeler Inc., San Clemente, CA, USA), and Biomechanics of Bodies (BOB-Biomechanics, Coventry, UK). Key outputs from musculoskeletal simulation are related to the internal loading conditions of the joint, which cannot be measured easily *in vivo*. These include joint contact forces, ligament and tendon strains, and forces generated in the muscles. Force analysis can be used to optimize implant component design or to estimate implant component life. More importantly, inverse dynamic musculoskeletal simulation provides estimation of biomechanics in the presence of active muscle forces and patient body-weight. Validation of dynamic simulation models can be quite difficult and, furthermore, these efforts have been hampered by a lack of experimental data. In recent years, there has been an effort to quantify *in vivo* joint loading with the use of instrumented, force-measuring implant components (Fregly et al., 2012; Kutzner et al., 2010; Taylor et al., 2017; Torrão et al., 2015). Such components telemetrically communicate implant loading data while the subject is engaged in various normal activities of daily living. Some of datasets obtained from these studies have been made available to the public, examples being the Orthoload (Bergmann et al., 2014), SimTK Grand Challenge to Predict *In Vivo* Knee Loads (Fregly et al., 2012), and the CAMS Knee Set datasets (Taylor et al., 2017). These datasets are an invaluable resource in validation of various musculoskeletal simulation models.



Figure 14. Musculoskeletal simulation model developed in OpenSim for modeling deep squatting motion (Schellenberg et al., 2018).

2.7. Musculoskeletal Simulation Models and Surgical Planning

A musculoskeletal simulation model has the potential to provide estimation of active joint loading and mechanics and could be deployed on a RASS for operative planning. Some RASSs utilize preoperative planning whereas others utilize intraoperative planning. In a preoperative planning system, a patient-specific simulation could be constructed and then adjusted with the use of advanced 3D medical imaging and preoperative functional assessment from motion capture gait analysis and ground reaction force measurements. Thus, the influence of implant component size and alignment could be investigated ahead of the surgery.

Alternatively, intraoperative planning could be done while the patient is on the operating table, which leads to savings in total cost of the surgical procedure. However, here are two challenges in using a musculoskeletal simulation model in a RASS that utilizes intraoperative planning.

First, the time and computational burdens associated with solving musculoskeletal simulations are too high (for example, one simulation run could take hours) to feasibly deploy in the operating room. Second, simulations can crash, jeopardizing the surgery and, hence, the well-being (or, even, life) of the patient.

CHAPTER 3

DEVELOPMENT OF SURGICAL PLANNING TOOL

3.1. Overview

A generalized view of the surgical planning tool development strategy (that is, the key steps in the development of the tool) is given Figure 14. First, a validated simulation model of the knee joint after TKA was obtained. For this purpose, models with different fidelities were used; namely, single-joint (low fidelity) and full-body (high-fidelity) models. Second, utilizing the validated models, sensitivity analyses were conducted to determine the sensitivity of musculoskeletal model responses to changes in surgical technique. Third, those responses that were deemed significant were fed into a fractional factorial design simulation study to characterize the model behavior for various combinations of surgical inputs. Fourth, results of the factorial design study were analyzed, with these results, in conjunction with linear regression analysis, being used to create a series of equations (herein, referred to as “knee performance equations (KPEs)”). These KPEs are relationships between model inputs and outputs, and, more importantly, in the fifth step, these equations were used for multi-parameter optimization. The final results are the optimal position and orientation of an implant component, with the optimization being with respect to knee joint parameters that, in the literature, have been shown to be strongly and directly correlated with poor TKA outcomes.

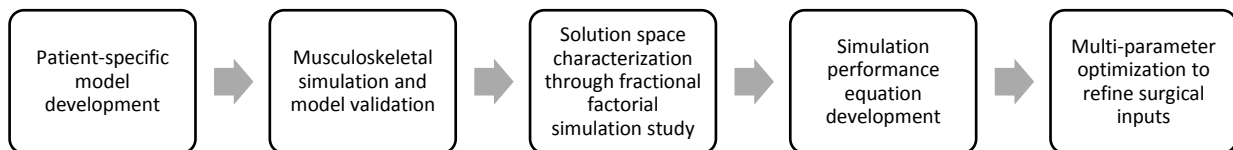


Figure 15. A schema of the steps in the development of a simulation-enhanced planning tool for robotics-assisted TKA.

3.2. Simulation Development

LifeModeler software package was used for the patient-specific multibody dynamic simulation of the knee joint. LifeModeler relies on the Adams/Solver physics engine (MSC Software, Newport Beach, CA, USA), which is a non-linear numerical solver that has a long history of use in the analysis of mechanical system behavior. LifeModeler is equipped with anatomy-specific plug-ins for analysis of various areas of the human body, examples being NeckSIM and LumbarSIM plug-ins for spinal analysis and KneeSIM Lab and Virtual Clinical Trial Knee (VCTK) for analysis of the knee joint. LifeModeler patient-specific simulations are comprised of various model components, including bony anatomy models, muscle tissue models, ligament and tendon models, and implant/prosthesis models. Bony anatomies are imported from 3D medical image databanks, such as CT or MRI results. Muscle and other soft tissue properties are prescribed based on patient health. Soft tissue generation and attachment sites are customized based on patient anatomy. CAD files of implant components, such as those provided in the SimTK Grand Challenge dataset, are imported into the modeling environment and are fixed to various body segments to simulate surgical intervention.

LifeModeler is capable of passive and active modeling modalities (Figure 16). Passive models can be generalized as those that solely react to the environment around them. In the context of biomechanical simulation, passive models run like physical tests on a dummy body. The body model does not drive any motion and simply reacts to outside stimuli. This type of model is useful for understanding, for example, man-machine interaction and ergonomics. Conversely, active models are those that cause reactions in the modeling environment. In the context of biomechanical simulation, these are models in which the joints of the body are actuated by muscle forces and the resulting kinematics, joint forces, and tissue forces are

obtained. As the purpose of the present study was to simulate the *in vivo* performance of a TKA prosthesis, an active modeling strategy was used.

The general modelling approach for LifeModeler active modeling is rooted in the sequential computation of inverse and forward dynamic simulations. Presented in the context of biomechanical simulation, inverse dynamic simulations utilize prescribed body kinematics to calculate the forces and moments acting on and in the body segments. In the case of LifeModeler simulations, kinematics data captured during gait analysis are used to drive inverse simulations. Results from inverse simulations are then used as input to subsequent forward dynamic simulations. In forward dynamic simulations, the forces imposed on the body are specified and the resulting joint kinetics are measured. In LifeModeler, muscle lengths and forces are recorded during inverse simulations and are used as training input for forward simulations. The recorded muscle lengths are used to drive the forward simulation, and the model kinematics, joint reaction forces, implant reaction forces, and tissue reaction forces are obtained.

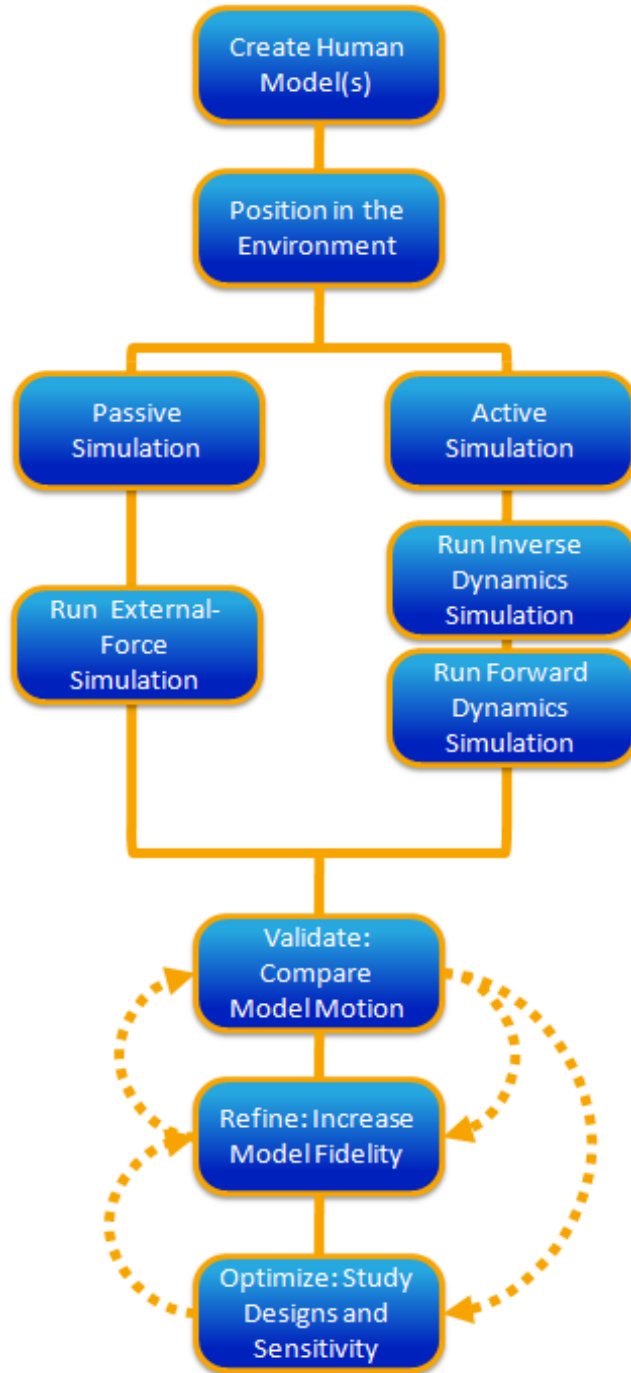


Figure 16. Schematic drawing of LifeModeler modeling strategies for active and passive types of simulation (LifeModeler Inc., 2019).

To summarize, then, in the present study, an active modeling strategy with LifeModeler was used to create patient-specific multibody dynamic models. The modeling strategy began with an inverse simulation to train the muscle model and a subsequent forward simulation to determine kinematics, joint reaction forces, implant forces, and soft tissue forces. The data required for each of the phase of the modeling strategy are shown in Figure 17.

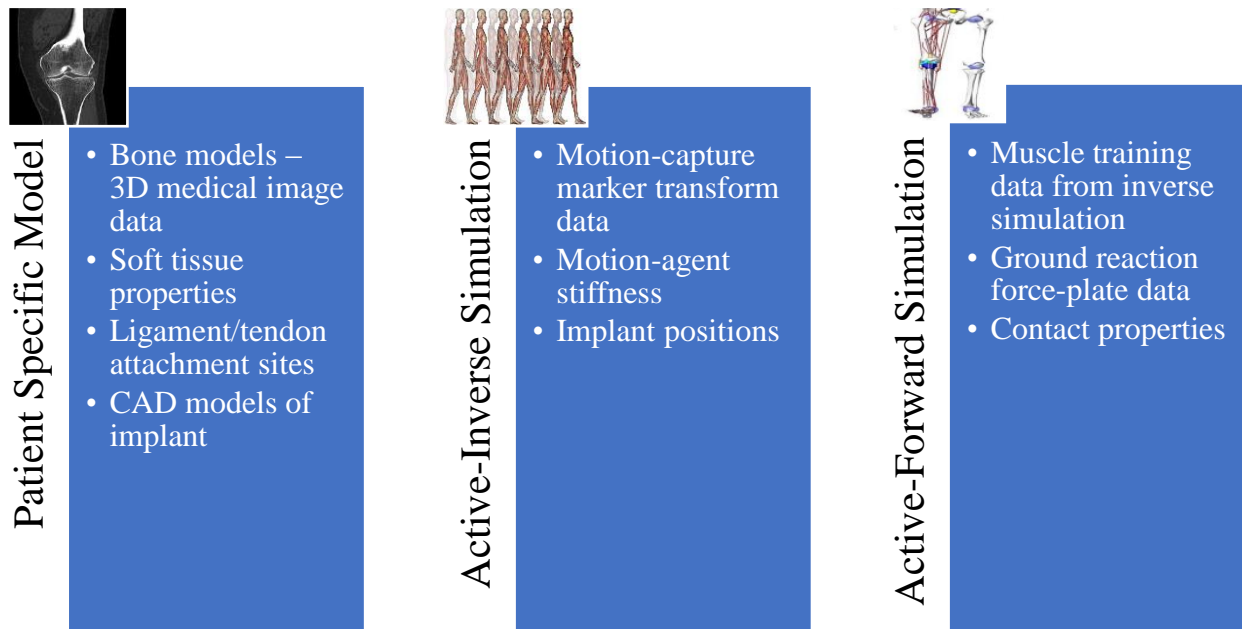


Figure 17. Data flows for various modeling strategies of the simulation using Lifemodeler.

3.3. Patient-Specific Model Development

The patient-specific model was created using biometric data for subjects, which are contained in two publicly-available datasets that included knee joint kinematics and kinetics results from subjects (who had undergone TKA) while they performed various normal activities of daily living. These databases were the Orthoload dataset and the SimTK Grand Challenge to Predict In Vivo Knee Loads Project dataset (Bergmann et al., 2014; Fregly et al., 2012)

(hereafter referred to as, SimTK dataset). LifeModeler allows full musculoskeletal characterization. Key modeling classes include body segments, anatomical landmarks, joints, muscles, ligaments, and implant components. Body segments are defined as an abstract representation of an articulating body part. Each body segment contains a bone or series of bones that act as reference body to which muscles and ligaments are attached. Bone geometries were imported from LifeModeler's proprietary anthropometric database. LifeModeler is capable of scaling bone geometries to match those of the test subject. Scaling in LifeModeler is sex-specific, with different scaling properties for female and male subjects. In the present study, the lower limbs of the subject were scaled using measurements derived from CT images, which were provided in the SimTK dataset. Landmarks on the bone models were defined for muscle attachment points. The list of muscles in the lower-body model is given in Figure 18.

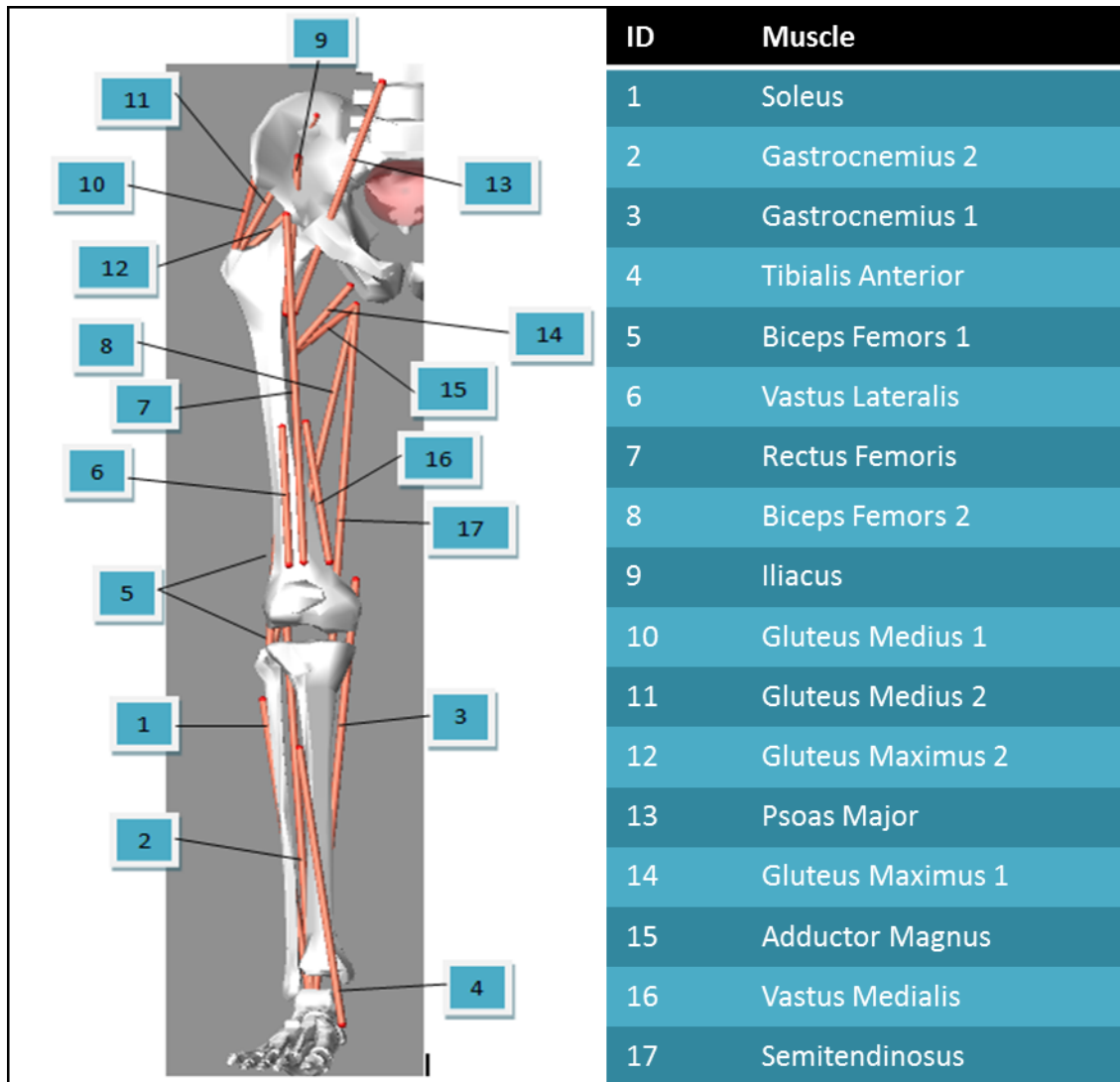


Figure 18. Muscles included in the patient-specific model.

3.4. Inverse Dynamic Simulation and Forward Dynamic Simulation

During an inverse dynamic simulation, muscle elements act as simple spring-damper elements but have no impact on the motion of the model. Their role is to record point-to-point muscle displacements, velocities, and spring forces. During a kinematics simulation, muscle elements were set to a recording state and were used to train muscle behavior for subsequent forward dynamic simulation. Based on the muscle length and force values recorded during the

inverse simulation, muscle activation was characterized with proportional, integral, and derivative (PID) control algorithms. Gain values for the PID control algorithms were tailored to achieve intended muscle behavior. Muscle forces were represented by the following equations for inverse and forward simulations, respectively.

$$\text{Force}_{\text{inverse}} = (\text{Stiffness} * \text{displacement} - \text{damping} * \text{velocity}) + \text{Preload} \quad (7)$$

$$\text{Force}_{\text{forward}} = P_{\text{gain}}(P_{\text{error}}) + I_{\text{gain}}(I_{\text{error}}) + D_{\text{gain}}(D_{\text{error}}), \quad (8)$$

where

$$P_{\text{error}} = (\text{target length} - \text{current length}) / (\text{range of motion}) \quad (9)$$

A ligament element was modeled as a point-to-point force. The force formulation is linear and the stiffness and damping of the ligament were specified. Ligament force was calculated using the expression:

$$\text{Force} = kD - cv \quad (10)$$

where k is the prescribed ligament stiffness, D is ligament displacement, c is the damping coefficient, and v is the displacement velocity. Ligament and tendon elements are wrap elements, meaning that they conform to the underlying bony anatomy and implant geometry, thus ensuring that there is proper force transmission between the tissues and the implant components. The geometrical details of the implant components were imported from various CAD applications. A virtual TKA was conducted by prescribing the positions and orientations of the implant components, relative to the skeletal elements. Contact mechanics relating to stick and slip friction, material stiffness, and damping characteristics were all tailored to optimize model performance.

3.5. KneeSIM Lab Model: Validation and Optimization

Preliminary exploration for TKA modeling was conducted using LifeModeler's KneeSIM Lab plug-in. This plug-in is a program designed to simulate a mechanical testing rig, namely, the Kansas Knee simulator (Figure 19) (DesJardins et al., 2000; Halloran et al., 2010; Zavatsky, 1997). The model itself was a lower left limb, with anatomically accurate bone models of the femur, tibia, and patella. The hip was modeled as a revolute joint parallel to the flexion and extension axis of the knee. Motion of the hip joint was constrained to translation in the vertical direction. The ankle joint was a combination joint model that allowed translation in the medial-lateral direction and rotation in flexion-extension, axial rotation, and varus-valgus rotation. The active muscles that drove flexion and extension of the knee model were the quadriceps and hamstring muscle groups. The model simulated a deep knee bend, of which the degree of rotation was customized. Outputs from KneeSIM Lab include patellofemoral and tibiofemoral kinematics, soft tissue forces and strains, and joint contact forces.

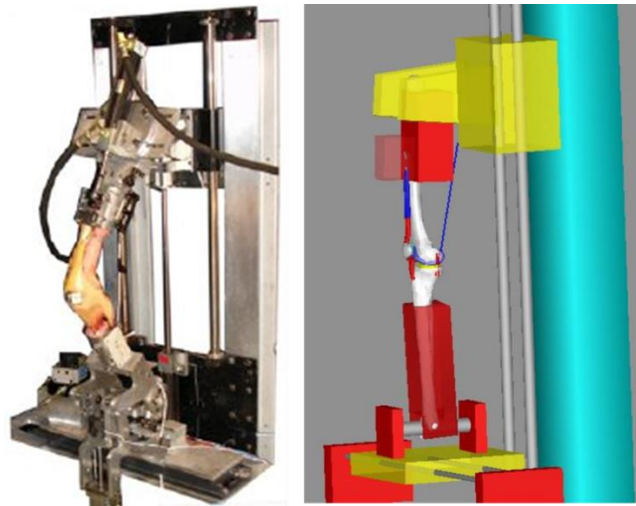


Figure 19. Side-by-side comparison of Kansas Knee Simulator mechanical testing rig (left) and LifeModeler's KneeSIM Lab model (right) (Clary et al., 2013).

Mechanical rigs are widely used for testing and evaluation of components of knee joint replacements. These rigs are popular because they accurately reproduce the kinematics of the knee after TKA in a highly controlled and measurable environment (De Coninck et al., 2016; Halloran et al., 2010; Maletsky and Hillberry, 2005; Varadarajan et al., 2009; Wunschel et al., 2010). In one study in which strain patterns in the iliotibial band, collateral ligaments, and patellofemoral ligament were determined, it was found that strain patterns predicted by KneeSIM Lab were highly correlated with those obtained from testing using cadaver knees (Evangelista et al., 2018). Furthermore, an examination of KneeSIM Lab kinematics compared to those generated by a Kansas Knee Simulator, showed that key measures, such as femoral internal-external rotation and anterior-posterior translation, were accurately predicted by the simulation model (Smith and Nephew, 2015).

KneeSIM Lab was selected for preliminary investigation for two reasons. First, it had been demonstrated that the model could accurately predict biomechanical parameters associated with various implant systems, and, therefore, could be used to optimize implant component selection for preoperative planning (Evangelista et al., 2018). The significant publication history supporting the use of mechanical testing frames for biomechanical research and evaluation of implant component design provided further justification for use of KneeSIM Lab to support the simulation-enhanced planning. Second, the reduced number of elements in the single-leg model provided a computationally efficient simulation, which allowed for rapid iteration. A drawback of the KneeSIM Lab is that model results are validated relative to *in vitro* rather than to *in vivo* joint data. In this regard, it is worth noting that a common feature of mechanical knee-joint simulator systems, such as the KneeSIM Lab, is that they utilize quadriceps force to drive flexion and extension of the joint. Knee kinematics are directly related to the magnitude of the forces

applied at the quadriceps. Thus, it is difficult for *in vitro* simulators to simultaneously reproduce accurate *in vivo* knee kinematics and relevant physiological loading (Verstraete and Victor, 2015).

To understand the simulated loading patterns produced by the baseline KneeSIM Lab model and their physiological relevance, a pilot study was conducted using the created patient-specific model and results given in the Orthoload database. In this database, there were *in vivo* knee loading data for 8 subjects (each of whom had undergone TKA) while performing various activities such as walking, stair ascend/descend, sit-to-stand, and deep knee bend. The loadings were measured using a customized and instrumented tibial tray; specifically, the tray was equipped with strain gauges and telemetry instrumentation that allowed for the measurement of knee loadings in all 6 degrees of freedom (DOF). In the present study, only the Orthoload results obtained during deep knee bend were used for evaluation of the KneeSIM Lab model.

With the baseline KneeSIM Lab model, the calculated tibiofemoral forces were higher than those given in the Orthoload dataset. It was hypothesized that the baseline KneeSIM Lab model could be optimized to produce loading patterns very similar to those given in the Orthoload dataset (Figure 20). To that end, the Design-of-Experiments (DOE) methodology was implemented to develop the ideal combination of implant component positioning, hip loading, and soft tissue properties. DOE is a statistical technique that can be applied to many types of experiments. DOE experimental design was carried out using Adams Insight software package (MSC Software, Newport Beach, CA, USA). In total, 15 input factors were identified that varied implant component alignment, ligament and tendon stiffness, origin and insertion sites for each ligament, and origin and insertion sites for each tendon. The study was designed as a fractional

factorial experiment with 16 runs. 5 functional output responses were obtained; namely, tibiofemoral forces and torques, patellofemoral forces and torques, and patellar kinematics.

A full description of the DOE plan is given in Appendix A.

A linear regression model was fitted to the DOE results. The resulting linear equations were used for simulation optimization using Adams Insight multi-objective optimization tool. The optimization goal was to minimize the overall root mean square error (RMSE) between the magnitude of the tibiofemoral force calculated using KneeSIM Lab model and the value given in the Orthoload dataset. The optimal implant component positions and tissue properties were re-loaded into LifeModeler KneeSIM Lab for subsequent modeling. This new model is designated the Optimized KneeSIM Lab model.

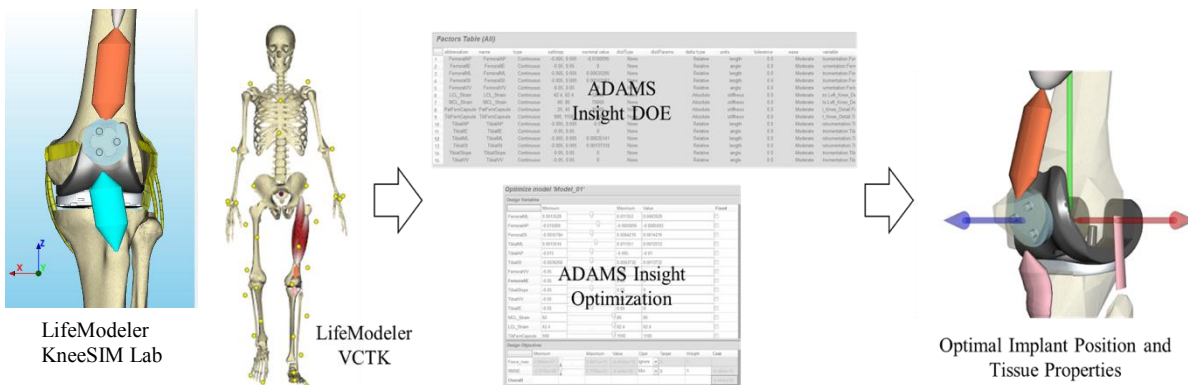


Figure 20. Workflow for developing optimized simulation parameters including forward modeling control gain, soft tissue properties, and virtual implant positions.

3.6. Virtual Clinical Trial Knee (VCTK) Model: Validation and Optimization

As previously discussed, it is difficult to re-create accurate physiological joint loading patterns with mechanical or cadaveric test rigs (Verstraete and Victor, 2015). Furthermore, these test rigs are designed to model a deep-knee bend or squatting activity. This motion pattern has

limited relevance for the majority of TKA patients, who are often elderly and find it difficult to complete a kneeling or squatting task. More relevant motion patterns for those patients are those that replicate normal activities of daily living, such as walking and climbing stairs. For this reason, a second simulation investigation based on a full-body gait analysis and implant loading data was performed. Thus, the study utilized LifeModeler VCTK plug-in, which is a full-body musculoskeletal simulation with a detailed soft tissue model surrounding the knee joint. The musculoskeletal simulation model can be tailored to patient-specific dimensions utilizing LifeModeler's custom scaling, which is based on the GEBOD database (Huaining Cheng et al., 1996).

To validate the musculoskeletal simulation model, the SimTK dataset was utilized. This dataset contained the results of a comprehensive biomechanical analysis of subjects who had received TKA. The subjects were outfitted with a force-sensing telemetric tibial implant component, similar to the case for the Orthoload subjects. This component was equipped with a device comprising four load-sensing elements capable of measuring axial load, anteroposterior and mediolateral moments, and shear forces (Kirking et al., 2006). Power to and communication with the device was achieved by wireless induction and data transmission. In addition to joint loading data, the dataset contained raw and synchronized motion-capture marker trajectories, force-plate ground reaction forces, skin surface electromyography data, CT scans, x-rays, and implant component geometries. The annual SimTK Grand Challenge Competition was first held in 2009 and the latest one was held in 2014. As VCTK was originally developed for the left knee, subject data from the 2014 Competition (hereafter, this dataset is referred to as, "2014 SimTK dataset") provided the most recently available data for a subject with a left knee TKA. As such, this body of data was used in the present study; specifically, data obtained from Subject

PS (180 cm tall, 75 kg; male) were used. (It is worth recalling that the subjects in the Orthoload dataset had also received left-knee TKA.)

The weight and height of Subject PS from the 2014 SimTK dataset was used to scale the patient specific VCTK musculoskeletal model. Musculoskeletal model anatomy was further refined by adjusting individual bone segment lengths based on measurements from the subject's CT data, which was provided with the SimTK dataset. Default ligament origin and insertion positions were maintained for the baseline model. Initial implant component position was determined by placing virtual prosthesis components relative to known bony landmarks per conventional surgical techniques, knowing that implant component position would be further refined in subsequent modeling steps.

As described in Section 3.4, inverse simulation kinematics were driven by gait laboratory marker data. Approximate marker locations were described in the competition brief. These locations were imported into the VCTK simulation modeling environment with a custom Python script (Appendix B), where the positions were further improved by modifying them relative to those in the virtual anatomy. LifeModeler VCTK allowed importation of the gait analysis marker data as motion agent elements, which are massless elements whose motions were controlled by the gait lab marker trajectories (Figure 21). To account for misalignment between real and virtual markers, motion agents were attached to body segments (bones) with virtual spring elements. Spring stiffness parameters were adjusted to achieve the desired kinematics.

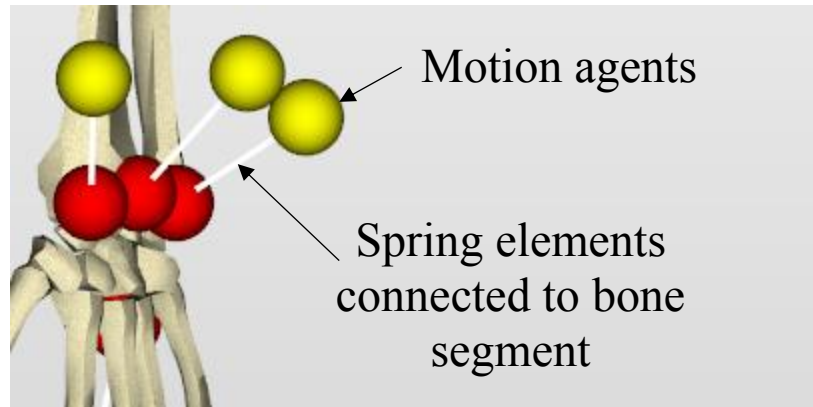


Figure 21. A schematic drawing showing VCTK motion agents (yellow), which are driven by motion capture marker trajectories and spring elements (white), which, in turn, are attached to the bone model (red).

Although results for a large number of joint parameters were provided to the SimTK competitors, in the present study, only joint contact forces were used for model calibrations. Movement trails with accompanying joint contact force data were examined to find the most relevant motion patterns for patients seeking TKA. After TKA, the goal of most patients is to walk without pain and/or to improve their walking endurance; as such, movement trails were filtered based on similarity to normal gait (Mancuso et al., 2009; Mannion et al., 2009).

It is of note that although the SimTK studies examined patient gait, instrumented knee data were not available to researchers who did not participate in the Competition. Therefore, in the present work, a sequential one-leg movement (hereafter referred to as “single-leg stance”) was selected for its similarity to normal-gait joint loading.

In accordance with the modeling and simulation processes described in sub-section 3.2, gait marker data were used to create an inverse simulation of Subject PS for a single-leg stance. As before, an inverse simulation was used to train muscle elements for subsequent forward simulation. In the case of VCTK, recorded muscle forces were used to drive movement of the

detailed left knee model. Movement of the other joints in the body was specified by PID control algorithms.

Preliminary analysis of the baseline VCTK model revealed that predicted tibiofemoral joint contact force magnitudes were markedly lower than those given for Patient PS in the 2014 SimTK dataset (maximum difference $2.5 \times BW$). Thus, it was decided to use the DOE methodology to optimize the model (Figure 20). Adams Insight was used to devise a fractional factorial experimental design to adjust the muscle gains for the detailed left knee model. A complete description of the experimental design is given in Appendix A. The measured output response from the gain-tuning DOE was the simulated tibiofemoral joint reaction force. The optimization tool in Adams Insight was used to determine the muscle gain parameters that produced the maximum tibiofemoral force magnitude that was closest to that for Subject PS. These muscle parameters were utilized in subsequent optimization studies. DOE was further used to determine the soft tissue parameters of the collateral ligaments and joint capsule as well as the implant alignments that produced joint contact forces that were closest to those for Subject PS. A separate fractional factorial experiment comprising 32 simulation runs was conducted (Appendix A). Optimal implant component position and soft tissue properties were selected using Adams Insight optimization tools, where the optimization goal was to minimize the RMSE between predicted and measured tibiofemoral contact force for Subject PS. The simulation model was adjusted to record force measurements at the same frequency as the SimTK measurements, and, as such, synchronizing the two sets of force magnitudes was straight forward. RMSE was chosen as the metric of interest for simulation optimization because it was reported by participants in the Grand Challenge Competitions. In other words, RMSE could be used to benchmark the present VCTK analysis results against those obtained by the

aforementioned participants. Note that, for each simulation run, RMSE was calculated using the following expression:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (P_i - O_i)^2}{n}}, \quad (11)$$

where P_i is the simulation model-predicted tibiofemoral joint contact force for Patient PS and O_i is the measured value of that force for Patient PS for n force measurements collected during testing (as reported in the 2014 SimTK dataset).

3.7. Development of Knee Performance Equations

A technique to characterize the many multibody dynamic simulations that were performed in the form of a set of closed-form linear functions was designed. Hereafter, these functions will be referred to as KPEs. Development of these equations was done using DOE techniques. The decision to use these techniques was guided by the need to use only input factors that could be incorporated into a planning tool for a raTKA system that is in current clinical use, such as the Navio Surgical System. In the present study, output responses were chosen based on review of the current literature. Patient dissatisfaction with TKA has been linked to a number of factors, with two of the most common ones being anterior knee pain and poor proprioception (Bryan et al., 2018; Choi and Ra, 2016; Kahlenberg et al., 2018; West et al., 2019). In many literature reports, it is suggested that anterior knee pain is associated with problems restoring patellar mechanics, which, for the most part, are ignored when current-generation raTKA is performed (Donell, 2018; Shervin et al., 2015). Proprioception, which is defined as the sensation of awareness of the position or movement of the body, is less well understood. However, some research suggests that mid-flexion instability contributes to poor proprioception in the joint (Clary et al., 2013). Thus, the KPEs developed in the present study focused on output responses

that could have a possible effect on anterior knee pain and proprioception (Tables 3 and 4). Note that implant position deviations are relative to the optimal implant position calculated during the simulation optimization DOE study. The ranges of positions and orientations were determined through various sweep studies to find a consistent and symmetric combination of values of these parameters that would allow simulation convergence to occur in a reasonable amount of time.

Table 3. DOE input factors^a and their range of values for the KneeSIM Lab and VCTK knee performance equations.

Knee Performance Equation Input Factors	Deviation from optimized implant pose
Femoral component ML position	± 5 mm
Femoral component AP position	± 5 mm
Femoral component SI position	± 5 mm
Tibial component ML position	± 5 mm
Tibial component AP position	± 5 mm
Tibial component SI position	± 5 mm
Femoral component varus/valgus rotation	± 0.05 radian
Femoral component internal/external rotation	± 0.05 radian
Tibial component varus/valgus rotation	± 0.05 radian
Tibial component internal/ external rotation	± 0.05 radian
Tibial component slope	± 0.05 radian

^aML: medial-lateral; AP: anteroposterior; SI: superior-inferior.

Table 4. Influences on the responses of the output outcomes^a of the VCTK knee performance equations.

Knee Performance Equation Output Responses	Target Influence
Anterior MCL strain	Mid-flexion stability (Evangelista et al., 2018; Kumar and Shahzad, 2019)
Posterior MCL strain	Mid-flexion stability (Evangelista et al., 2018; Kumar and Shahzad, 2019)
Anterior LCL strain	Mid-flexion stability (Evangelista et al., 2018; Kumar and Shahzad, 2019)
Posterior LCL strain	Mid-flexion stability (Evangelista et al., 2018; Kumar and Shahzad, 2019)
Quadriceps angle	Anterior knee pain (Almeida et al., 2016)
Quadriceps force	Anterior knee pain (Shervin et al., 2015)
Patellofemoral contact force	Anterior knee pain (Innocenti et al., 2011)
Tibial implant force	Implant longevity (Saikko, 2006)
Tibiofemoral joint contact force	Joint balance (Churchill et al., 2018)

^aMCL: medial collateral ligament; LCL: lateral collateral ligament.

A fractional factorial experimental design was devised for the KPE DOE (Appendix A). After solving for each of the simulation runs, linear regression was used to develop knee performance equations for each output response. For KneeSIM Lab and VCTK models, the simulation model output parameters relating to the KPE responses were slightly different. Therefore, the number of factors and responses for each model fidelity was different: for KneeSIM Lab, there were 11 responses relating to the 11 input factors and for VCTK there were 9 responses relating to the 11 input factors. KPEs were utilized for multi-objective optimization using Adams Insight to determine optimal implant component position and orientation for which both patellar mechanics parameters (which have been directly associated with anterior knee pain) and mid-flexion instability parameters (which have been directly associated with proprioception) are minimized.

3.8. Proof-of-Concept Study

To determine if use of the simulation-derived KPEs could have an impact on the surgical planning, a proof-of-concept study was designed. The essence of this study was to compare implant component position and orientation prescribed when the implant component was implanted using a conventional technique (“actual passive surgical plan”) to corresponding values calculated using the simulation-enhanced planning tool developed in the present study (SEISPT) (“conceptual active surgical plan”). As the final positions and orientations of the implant components were provided for Subject PS in the 2014 SimTK dataset, these measurements were used as the actual passive surgical plan results in the aforementioned comparison. Multi-objective optimization of the KPEs was used to determine the positions and orientations of implant components that minimize the MCL and LCL strain, which were then

used as the conceptual active surgical plan results. Additionally, an optimization objective to match the tibiofemoral joint contact force provided in the SimTK dataset was imposed. This additional optimization objective was enacted to ensure that the suggested surgical plan resulted in physiologically-relevant loading and clinically-acceptable implant component positions and orientations.

To compare final implant component positions and orientations, a shared coordinate reference frame was created for the bone and the implant component models utilizing Geomagic Studio (3D Systems, Rock Hill, SC, USA). Bone and implant component models for Patient PS (SimTK) were imported into Geomagic Studio. A custom Matlab script was used to determine the positional transform between the implant model and the bone model. Likewise, bone and implant component models were exported from LifeModeler. The KPE was used to position the implant component relative to the LifeModeler bone. To bring all bone and implant components elements into a single reference system, the registration tools in Geomagic Studio were utilized. Key anatomic landmarks were used to register the Subject PS (SimTK) and LifeModeler bone models, and, consequently, the implant components associated with each of these models. Geomagic Studio was further used to examine the difference in 1) the implant component position used per the actual passive surgical plan and the optimal implant component position calculated using the LifeModeler musculoskeletal simulation models (KneeSIM Lab and VCTK) and the KPEs (that is, SEISPT; “conceptual active surgical plan”); and 2) the implant component orientation used per the actual passive surgical plan and the optimal implant component orientation calculated using the “conceptual active surgical plan”. The steps in this process are summarized in Figure 22.

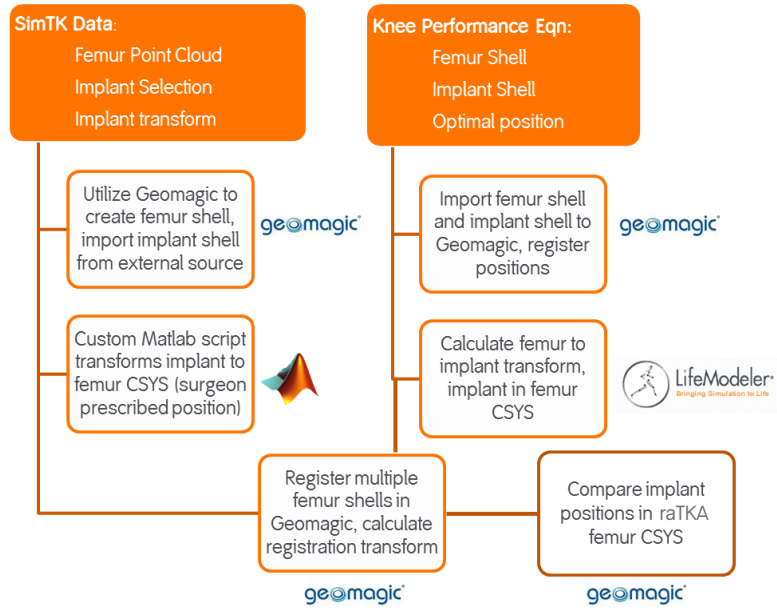


Figure 22. Registration process for the bone model and the implant component model using Geomagic Studio, Matlab, and Lifemodeler for the femur and the femoral component of the implant. This process is used to create a consistent coordinate system (CSYS) for implant position analysis. This process is identical to that when the tibia and the tibial component were used.

CHAPTER 4

RESULTS

4.1. Overview

The results presented herein describe simulation performance of both LifeModeler KneeSIM Lab and VCTK models relative to Orthoload and SimTK datasets, respectively. In each case, the optimized simulation model whose outputs were the closest to those in the respective dataset was used to create a set of equations which described the behavior of the joint (that is, KPEs). These KPEs were subsequently used to determine optimal implant positions and orientations as being position and orientation that minimized patellar mechanics parameters that have been strongly and directly correlated with anterior knee pain in TKA patients and mid-flexion instability parameters that have been strongly and directly correlated with poor proprioception in TKA patients. To determine if the simulation tool developed (that is, the combination of optimized musculoskeletal simulation model and KPEs) could alter surgical treatment strategy, optimal implant component position and orientation obtained from using the simulation tool were compared to corresponding values in the case of one subject for which TKA was performed using a conventional technique.

4.2. KneeSIM Lab Model Validation

In the Orthoload dataset, 3 load levels were reported for each subject: (mean loads in subjects with body weight of 75 kg (AVER-75 study group); high loads in subjects with body weight of 100 kg (HIGH-100 study group); and peak loads in subjects with body weight of 100 kg (PEAK-100 study group). As an example, some results obtained when a subject performed the deep knee bend activity are shown in Figure 23. Comparisons of the mean

tibiofemoral joint contact force and moment obtained using the baseline KneeSIM Lab model and the corresponding means reported in the Orthoload dataset are given in Table 5. The improvement in these results when the Optimized KneeSIM Lab model was used is small (Tables 5 and 6). It is of note that the researchers responsible for the Orthoload dataset suggest that all comparative studies should utilize data from the HIGH-100 and PEAK-100 studies (Bergmann et al., 2014). For this reason, RMSE for the optimized KneeSIM Lab model was only calculated for these two study groups.

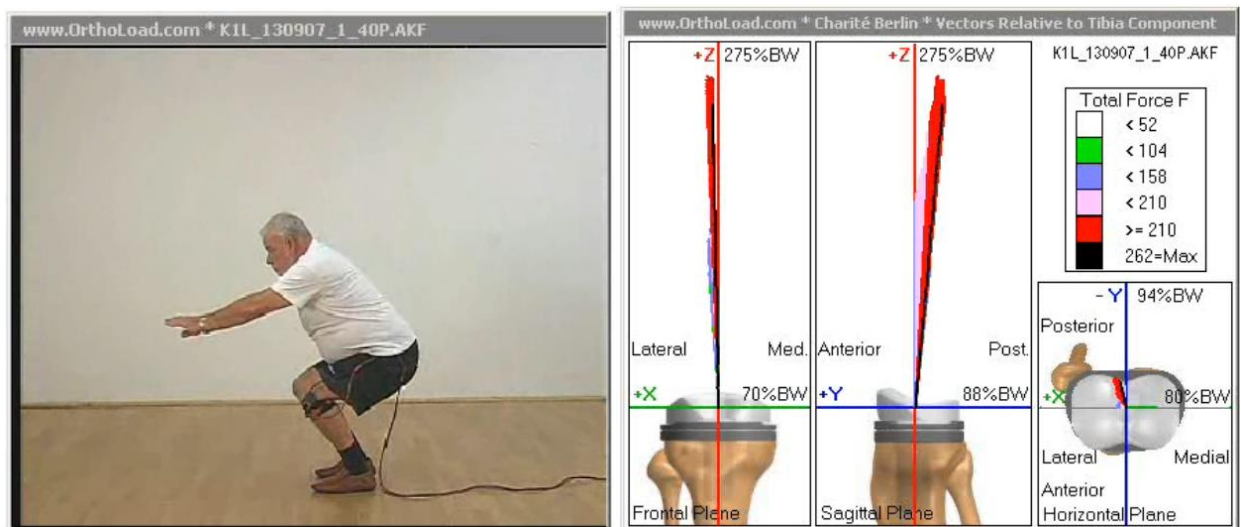


Figure 23. Photograph of Subject K1L performing deep knee bend activity (left) and animation of direction and magnitude of load vectors relative to the tibial component in the case of this subject (right)(Bergmann et al., 2014).

Table 5. Errors for mean tibiofemoral joint contact loadings: Baseline KneeSIM Lab model versus Orthoload dataset

Orthoload Dataset Group		RMSE
HIGH-100	Force	421 N
	Moment	10.86 Nm
PEAK-100	Force	446 N
	Moment	1.58 Nm

Table 6. Errors for mean tibiofemoral joint contact loading: Optimized KneeSiM Lab model versus Orthoload dataset

Orthoload dataset group		RMSE
HIGH-100	Force	413 N
	Moment	6.79 Nm
PEAK-100	Force	409 N
	Moment	7.15 Nm

4.3. VCTK Model Validation

As the SimTK Grand Challenge Competition was designed to be a blinded study, gait trials were not accompanied by joint contact force data. However, several calibration trials with joint contact force data and associated kinematics and ground reaction force data were provided. A comparison of all calibration movements was conducted with gait joint contact force data from the Orthoload dataset to identify loading pattern similarities. It was determined that the movement that most closely resembled the loading pattern of a single-gait phase was the single-leg stance movement. Therefore, simulation models constructed with VCTK utilized single-leg stance kinematics and ground force data. Figure 24 shows joint contact forces for 2014 SimTK single-leg stance (for Subject PS) as well as an Orthoload single-gait phase for walking. It is seen that although the time scales for the movements are different, the force magnitude and shape of the resulting load curves were qualitatively similar.

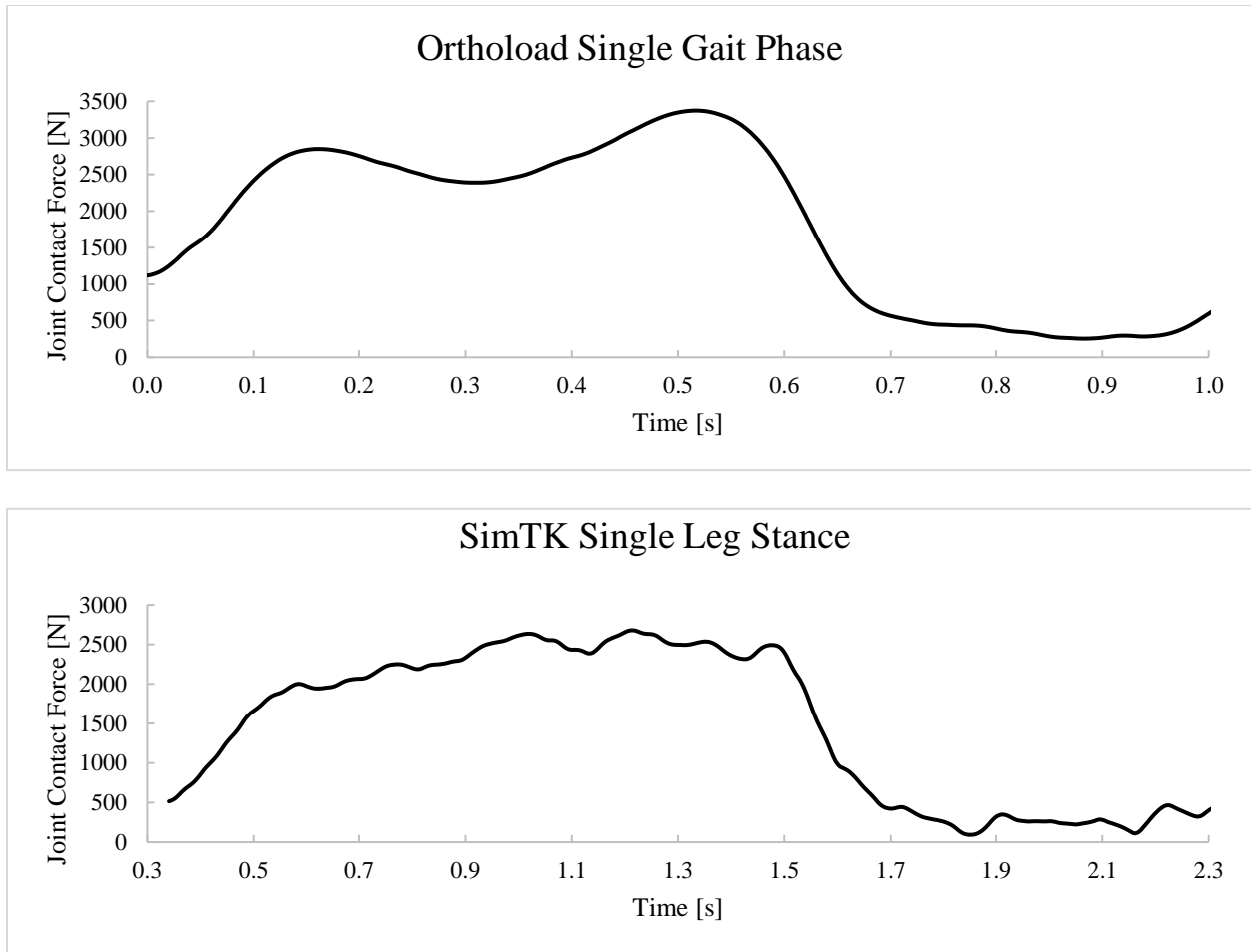


Figure 24. Tibiofemoral joint contact loading in Orthoload dataset (single-gait phase) (Bergmann et al., 2014) and in 2014 SimTK dataset (single-leg stance) (Subject PS) (Bergmann et al., 2014; Fregly et al., 2012).

Comparisons of the tibiofemoral joint force and moment obtained using the Optimized VCTK model and those reported in the SimTK study are given in Figures 25 and 26. It is seen that 1) during the early part of the cycle (0.3 s-0.8 s), the two sets of force results are very close but, after that, the model either underestimates or overestimates the experimentally-obtained results; and 2) throughout the cycle, the model either underestimates or overestimates the moment.

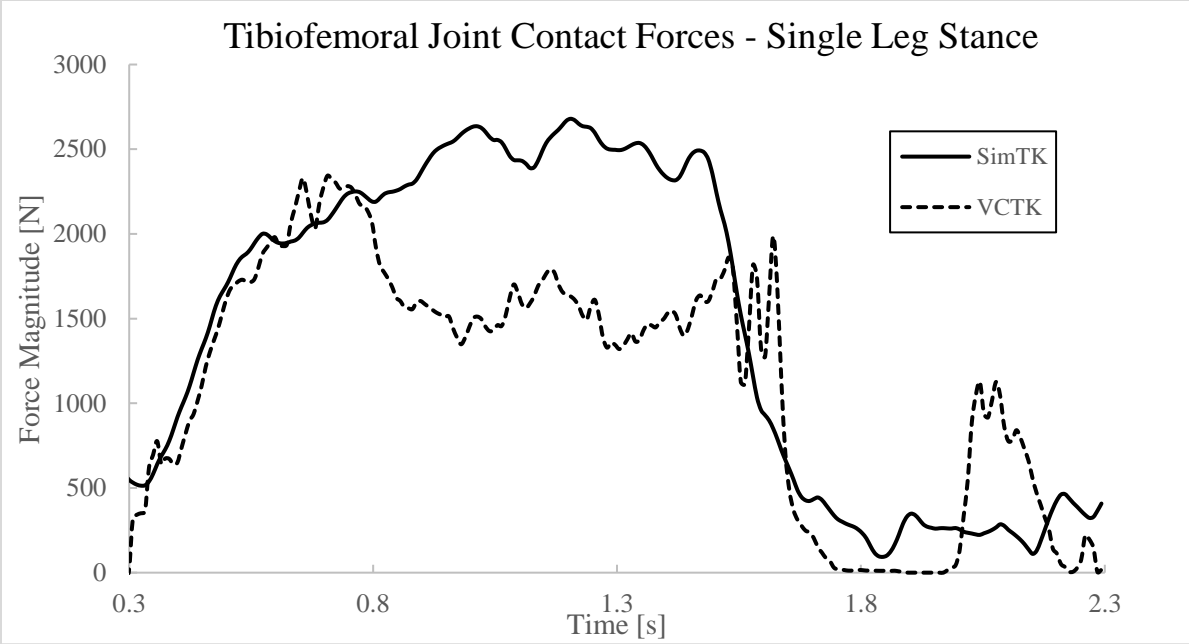


Figure 25. Tibiofemoral joint contact force magnitude versus time for single-leg stance: Optimized VCTK simulation model results versus 2014 SimTK dataset results (Subject PS).

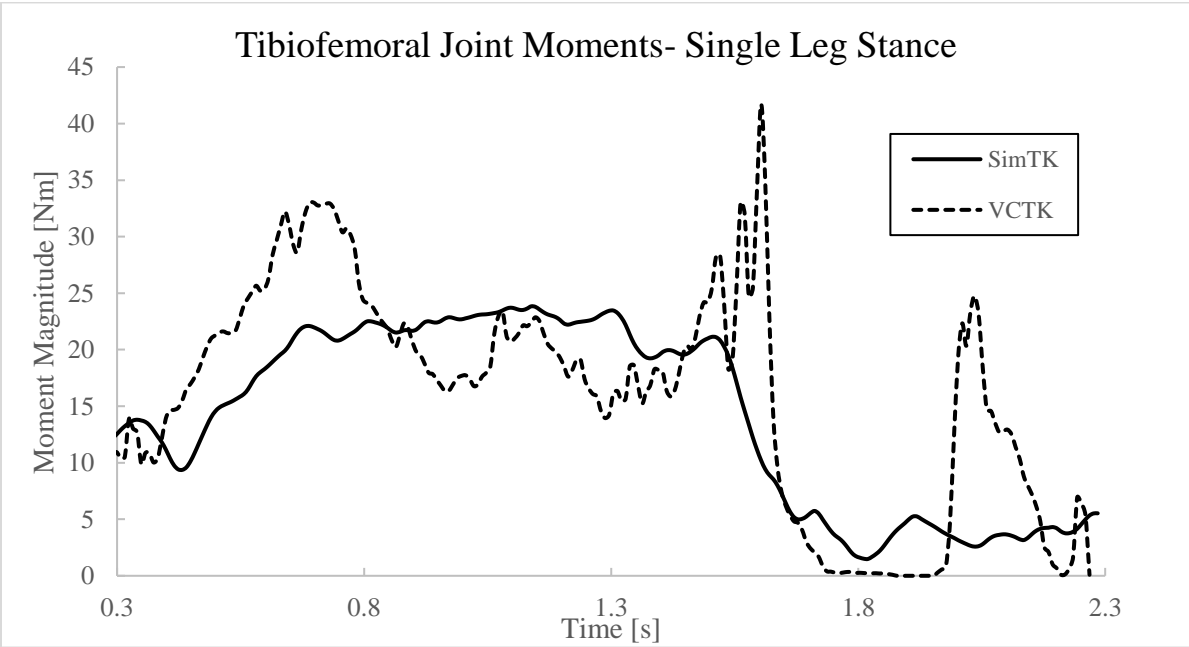


Figure 26. Tibiofemoral joint contact moment magnitude versus time for single-leg stance: Optimized VCTK simulation model results versus 2014 SimTK dataset results (Subject PS).

With respect to tibiofemoral joint contact loading reported in the SimTK dataset (Subject PS), RMSE was substantially reduced when Optimized VCTK model was used compared to when the baseline VCTK model was used (Table 7).

Table 7. RMS error for VCTK model tibiofemoral joint contact loading compared to mean results in SimTK dataset

	RMSE
Force magnitude (Baseline model result)	956 N
Moment magnitude (Baseline model result)	13.11 Nm
Force magnitude (Optimized model result)	618 N
Moment magnitude (Optimized model result)	7.63 Nm

4.4. Knee Performance Equations

The KPEs for the deep-knee bend using the Optimized KneeSIM Lab model are presented in the Table 8. In these tables, the simulation response names are given in the column headings and the names of the explanatory variables are given in the last column. The most highly correlated KPEs were those for tibiofemoral contact force (Pearson’s correlation coefficient ($R^2 = 0.92$), anterior LCL strain ($R^2 = 0.91$), anterior MCL strain ($R^2 = 0.89$), patellofemoral contact force ($R^2 = 0.88$), lateral compartment rollback ($R^2 = 0.84$), and posterior MCL strain ($R^2 = 0.81$).

The KPEs for single-leg stance movement using the Optimized VCTK model are presented in Table 10. The most highly correlated KPEs were those for quadriceps force ($R^2 = 0.85$), medial compartment rollback ($R^2 = 0.79$), lateral compartment rollback ($R^2 = 0.79$), Bundle 3 MCL strain ($R^2 = 0.76$), and Bundle 2 LCL strain ($R^2 = 0.79$).

Table 9: Knee performance equations developed using Optimized KneeSIM Lab and SimTK Subject PS data and implant.

Knee Performance Equation Output Responses											
IE Rotation (deg) ^a	Medial Rollback (mm)	Lateral Rollback (mm)	aMCL Strain	pMCL Strain	aLCL Strain	pLCL Strain	Q Angle (deg)	Quad Force (N)	Tibiofemoral Joint Contact Force (N)	Patellofemoral Joint Contact Force (N)	Variables
0.08	0.014	0.01	0.14	0.14	0.24	0.06	0.13	4533.5	3.30E+06	3.17E+06	1 (constant)
-2.47	-1.512	-1.81	-5.42	-16.57	-16.57	-4.89	0.35	-4.08E+05	4.88E+08	3.40E+09	Femoral ML position (mm)
-1.97	-1.247	-0.65	-10.01	-23.47	-23.47	-0.98	-32.50	-4.00E+05	-8.41E+08	1.87E+08	Femoral AP position (mm)
-13.93	0.053	-0.49	1.50	-4.41	-4.41	-7.12	-19.77	-4.63E+05	-3.23E+09	-2.15E+09	Femoral SI position (mm)
-1.99	0.087	0.01	4.52	9.60	9.60	2.25	-0.77	-1.00E+05	7.61E+08	-2.37E+08	Tibial ML position (mm)
-1.16	-0.751	-0.88	-5.82	9.97	-0.66	-0.66	22.97	5.75E+05	-1.06E+09	1.95E+09	Tibial AP position (mm)
-1.54	-0.423	-0.25	6.95	11.23	11.23	8.79	2.07	-1.14E+05	2.73E+09	3.77E+08	Tibial SI position (mm)
0.17	0.036	0.04	0.15	0.19	0.19	0.13	-0.30	-3262.9	1.71E+07	-4.74E+07	Femoral varus/valgus (deg)
-0.49	-0.059	-0.04	-0.49	-1.35	0.22	0.22	-1.75	-25749	-1.86E+07	-4.09E+07	Femoral IE rotation (deg)
-0.11	-0.032	-0.02	0.00	-0.13	-0.13	-0.15	-0.61	-9362.9	-6.84E+07	-4.98E+07	Tibial slope (deg)
-0.02	-0.004	-0.01	-0.02	-0.14	-0.14	0.06	0.05	-2385.9	2.40E+07	-4.44E+06	Tibial varus/valgus (deg)
-0.13	-0.025	0.00	-0.02	0.01	0.16	0.16	-0.46	-4452.9	-1.87E+07	-3.97E+07	Tibial IE rotation (deg)

^a IE rotation of femoral component relative to tibial component.

Table 8: Knee performance equations developed using Optimized VCTK and SimTK Subject PS data and implants.

Knee Performance Equation Output Responses										
IE Rotation (deg) ^a	Medial Rollback (mm)	Lateral Rollback (mm)	MCL Strain (Bundle 1)	MCL Strain (Bundle 2)	MCL Strain (Bundle 3)	LCL Strain (Bundle 1)	LCL Strain (Bundle 2)	Quad Force (N)	Variables	
-38.09	-0.463	-0.46	-0.44	-0.38	-0.33	-0.42	-0.35	1.26E+08	1 (constant)	
-88.12	-1.152	-1.15	-1.02	-0.97	-0.92	-0.81	-0.62	1.43E+08	Femoral ML position (mm)	
-77.32	-1.137	-1.14	-0.97	-0.85	-0.69	-0.76	-0.64	3.15E+08	Femoral AP position (mm)	
-65.23	-0.544	-0.54	-0.45	-0.44	-0.42	-0.37	-0.38	1.44E+08	Femoral SI position (mm)	
82.59	1.311	1.31	0.97	0.88	0.79	0.97	0.81	-3.71E+06	Tibial ML position (mm)	
-83.60	-1.435	-1.44	-0.82	-0.75	-0.71	-1.14	-1.01	1.35E+07	Tibial AP position (mm)	
60.34	0.617	0.62	0.66	0.53	0.41	0.60	0.31	-1.53E+08	Tibial SI position (mm)	
8.03	0.106	0.11	0.09	0.08	0.08	0.08	0.08	-2.66E+07	Femoral varus/valgus (deg)	
8.19	0.115	0.12	0.07	0.06	0.06	0.09	0.08	-1.19E+07	Femoral IE rotation (deg)	
-8.41	-0.122	-0.12	-0.06	-0.06	-0.05	-0.07	-0.08	-2.66E+07	Tibial slope (deg)	
7.21	0.086	0.09	0.06	0.05	0.04	0.06	0.05	1.42E+07	Tibial varus/valgus (deg)	
7.18	0.091	0.09	0.09	0.08	0.08	0.07	0.06	1.33E+06	Tibial IE rotation (deg)	

^a IE rotation of femoral component relative to tibial component.

4.5. Proof-of-Concept Study

A key step in the proof-of-concept study was determination of the optimization scheme for the analysis of the KPEs. As KPEs using both KneeSIM Lab and VCTK models produced acceptable fits with respect to MCL and LCL strains, ligament response parameters were considered for surgical plan optimization. The ligament strains obtained in the various simulation trials are presented in Figures 27 and 28.

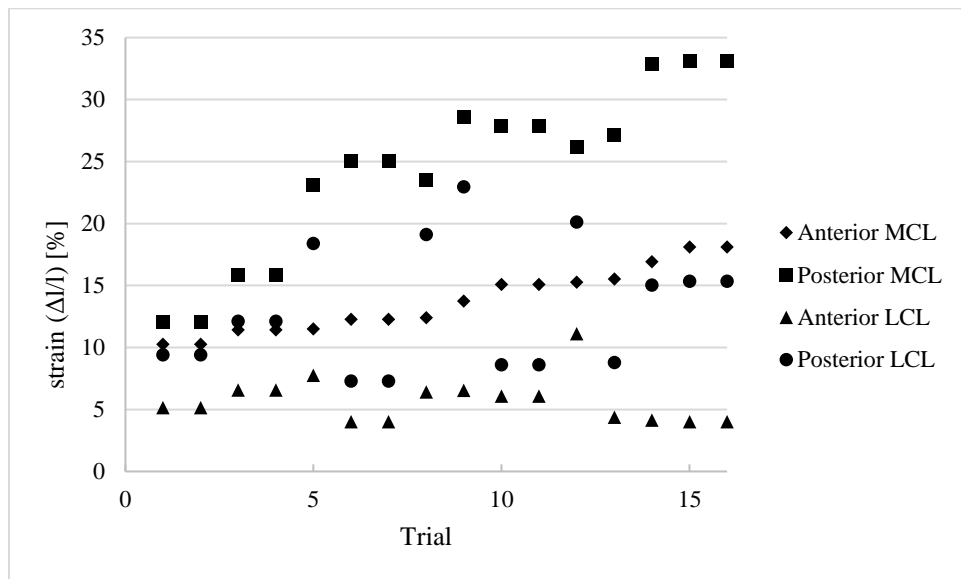


Figure 27. Maximum medial collateral ligament (MCL) and lateral collateral (LCL) strains for each of the simulation trials used for knee performance equation development with the Optimized KneeSIM Lab simulation model.

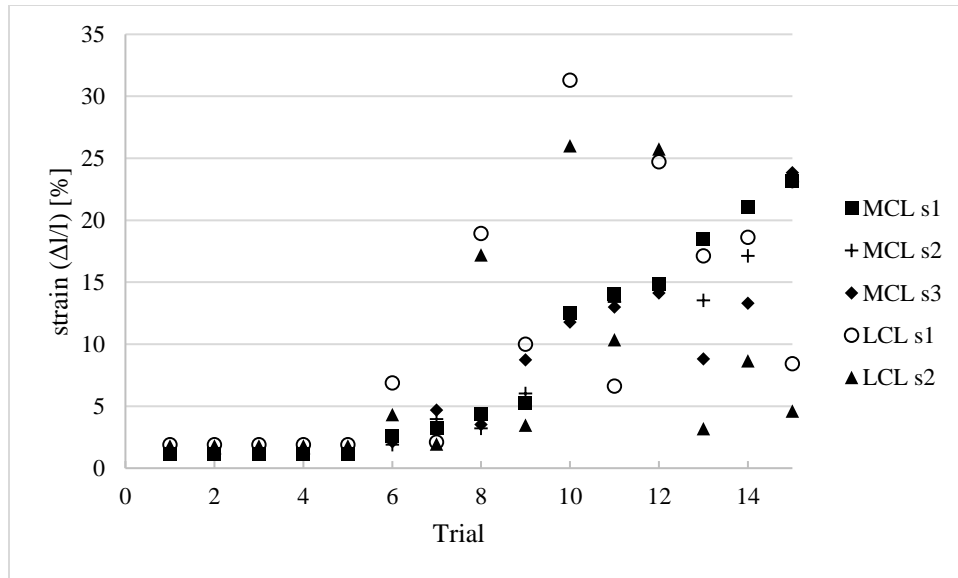


Figure 28. Maximum medial collateral ligament (MCL) and lateral collateral ligament (LCL) strains for each of the simulation trials used for knee performance equation development with the Optimized VCTK model. Note that trial #16 was excluded from the figure as some strain values obtained were $\geq 50\%$.

With both models (optimized KneeSIM Lab and optimized VCTK), the full collection of results on optimal locations and orientations of both the femoral and tibial components are presented in Tables 10 and 11. One point that should be kept in mind when interpreting these results is the accuracy of current-generation raTKA systems, specifically, the Navio Surgical System (Jaramaz et al., 2018) relative to the theoretical accuracy of a RASS that includes the SEISPT (Figure 29).

Table 10. Implant position and orientation data comparing results achieved for Subject PS (“actual passive surgical plan”) (gray) and results suggested for Subject PS using output from the conceptual active surgical plan (optimized KneeSIM Lab model) (blue). Implant positions and orientations are those that are postulated to minimize MCL and LCL strains. Results are shown relative to RMS equivalent error, which is defined as $|(RMSE\ conventional) - (RMSE\ robotic)|$ (Lonner et al., 2015; Jaramaz et al., 2018).

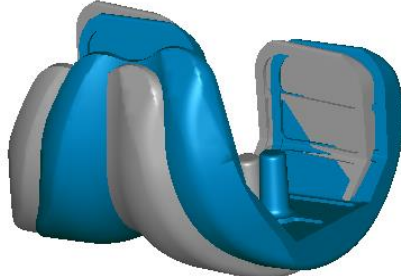



Femoral component position and orientation	Actual passive surgical plan vs. conceptual active surgical plan (Mean result)	RMS Equivalent Error	
Medial-lateral	4.0 mm	1.31 mm	
Anterior-posterior	1.3 mm	1.26 mm	
Superior-inferior	6.7 mm	0.83 mm	
Flexion-extension	1.09°	4.5°	
Varus-valgus	4.65°	3.4°	
Internal-external	6.34°	4.9°	
Tibial component position and orientation	Actual passive surgical plan vs. conceptual active surgical plan (Mean result)	RMS Equivalent Error	
Medial-lateral	3.3 mm	1.31 mm	
Anterior-posterior	2.0 mm	1.26 mm	
Superior-inferior	5.8 mm	0.83 mm	
Flexion-extension	0.89°	4.5°	
Varus-valgus	0.32°	3.4°	
Internal-external	1.49°	4.9°	

Table 11. Implant position and orientation data comparing results achieved for Subject PS (“actual passive surgical plan”) (gray) and results suggested for Subject PS using output from the conceptual active surgical plan (optimized VCTK model) (blue). Implant positions and orientations are those that are postulated to minimize MCL and LCL strains. Results are shown relative to RMS equivalent error, which is defined as $|(RMSE\ conventional) - (RMSE\ robotic)|$ (Lonner et al., 2015; Jaramaz et al., 2018).

Femoral component position and orientation	Actual passive surgical plan vs. conceptual active surgical plan (Mean result)	RMS Equivalent Error	
Medial-lateral	4.23 mm	1.31 mm	
Anterior-posterior	4.1 mm	1.26 mm	
Superior-inferior	6.73 mm	0.83 mm	
Flexion-extension	0.18°	4.5°	
Varus-valgus	3.73°	3.4°	
Internal-external	3.98°	4.9°	
Tibial component position and orientation	Actual passive surgical plan vs. conceptual active surgical plan (Mean result)	RMS Equivalent Error	
Medial-lateral	3.68 mm	1.31 mm	
Anterior-posterior	1.26 mm	1.26 mm	
Superior-inferior	7.6 mm	0.83 mm	
Flexion-extension	1.58°	4.5°	
Varus-valgus	2.45°	3.4°	
Internal-external	3.20°	4.9°	

CHAPTER 5

DISCUSSION

5.1. Overview

A RASS provides a means to accurately characterize the patient anatomy and to develop a surgical plan relative to a patient-specific model. Robotic surgical planning, if done intraoperatively, can utilize information about the patient's soft tissues to achieve a properly balanced joint replacement. Systems model the behavior of a patient's ligaments by intraoperatively flexing the knee joint and using the kinematics data to create a model of the ligaments and tendons. While these models provide the surgeon with patient-specific soft tissue data with which to plan the surgery, the influence of active tissue mechanics, specifically, the influence of active muscle forces and patellar mechanics, is, largely, ignored. The purpose of the present study was to investigate the use of a patient-specific musculoskeletal simulation tool that can provide the surgeon intraoperative feedback relating to the influence of active soft tissue behaviors on the operative plan. The first part of the study involved validation of two musculoskeletal simulation models that included the knee joint but of different fidelities with respect to data obtained from subjects who had instrumented tibial component and participated in various normal activities of daily living. These data are contained in publicly-available reports. In the second part of the study, a technique to characterize simulation model behavior for intraoperative analysis was designed and executed. This culminated in a novel conceptual simulation-enhanced intraoperative surgical planning tool (SEISPT). In the third part of the study, a proof-of-concept test of SEISPT was conducted, which comprised a comparison of

positions and orientations of implant components calculated using SEISPT to those achieved when a conventional TKA technique was used.

5.2. Simulation Model Validation

Development of the simulation-enhanced planning algorithms proposed in this study was conducted with two musculoskeletal simulation models that included the knee joint, namely, the KneeSIM Lab single-leg model (a low-fidelity model) and the VCTK full-body model (a high-fidelity model).

KneeSIM Lab accurately replicates the kinematics and ligament strain patterns of a mechanical test frame (Evangelista et al., 2018; Innocenti et al., 2011; Smith and Nephew, 2015) (Figure 30). As such, it was justified to carry out a validation study using KneeSIM Lab and data obtained from subjects (who had received instrumented TKA) while they performed deep knee bend (Orthoload dataset). The baseline KneeSIM Lab model performed reasonably well (RMSE = 421 N; 0.49 x body weight (BW)) but that performance was not improved by much when an optimized variant of the model was used (RMSE = 409 N; 0.47 BW).

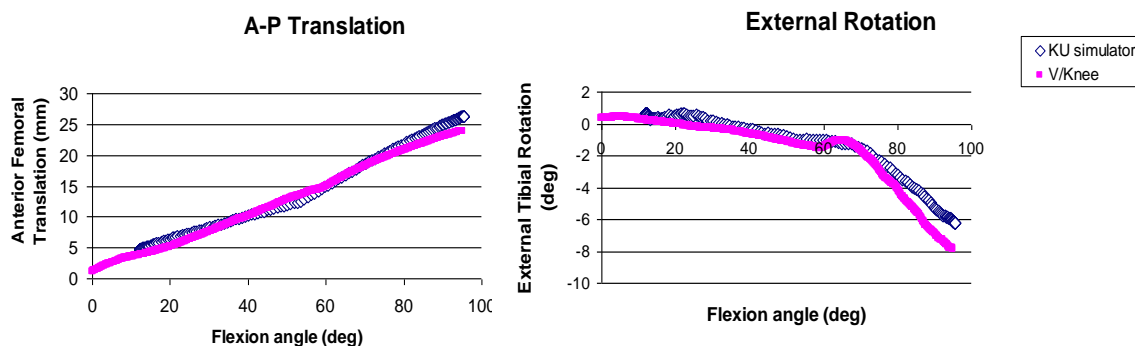


Figure 29. Comparison of kinematics results from KneeSIM Lab simulation model (V/Knee) and Kansas Knee Simulator test rig (KU simulator) (Smith and Nephew, 2015).

LifeModeler's VCTK plug-in is a fully customizable modeling solution for musculoskeletal simulation. It was hypothesized that by using a full-body simulation that was trained with kinematics and force data from a human subject would lead to lower RMSE. The data used to construct, tune, and validate the patient-specific model were taken from another publicly-available dataset (SimTK Grand Challenge to Predict *In Vivo* Knee Loads dataset (Fregly et al., 2012)). This provided the added benefit of creating a standard to which the VCTK simulations could be compared. As previously discussed, this dataset did not include instrumented implant component loading data from gait analysis, which was identified as a movement of interest for the majority of TKA patients (Mancuso et al., 2009; Mannion et al., 2009). A qualitative analysis of all the loading results in the 2014 SimTK dataset was considered relative to a gait sample in the Orthoload dataset. A single-leg stance provided a very similar loading pattern; thus, this movement was selected for development of the simulation model. Baseline VCTK simulation produced an RMSE of 956 N (1.3 BW) and DOE-optimized VCTK simulation produced a RMSE of 618 N (0.84 BW), each of which was higher than RMSEs reported by winners of the 2014 Grand Challenge Competition (Table 12).

Table 12. Previous winning results from the SimTK Grand Challenge to Predict *In Vivo* Knee Loads. Results given as RSME (expressed relative to body weight (BW) of subject).

Authors	Year winner announced	Subject identifier	Mass (kg)	RMSE Medial (xBW)	RMSE Lateral (xBW)	RMSE Total (xBW)
Kim et al. (2013)	2010	JW	64.6	0.36	0.51	0.69
Hast and Piazza (2013)	2011	DM	67.0	0.49	0.47	0.68
Manal and Buchanan (2013)	2012 ^a	SC	78.4	0.46	0.38	0.69
Knowlton et al., (2013)	2012 ^b	SC	78.4	0.33	0.36	0.62
Marra et al. (2015)	2014	PS	75.0	0.20	0.40	0.30
Jung et al. (2015)	2015	DM	70.0	0.23	0.28	0.36 ^c

^{a,b}Joint winners.

^cTotal RMSE calculated by present worker from medial and lateral results reported by Jung et al. (2015).

With KneeSIM Lab, the RMSE was comparable to that reported by previous winners of the SimTK Grand Challenge Competition (Table 12), albeit for a deep knee bend movement compared to the single-gait phase analyzed in the Challenge Competition. As KneeSIM Lab simulations were scaled to match the 110° of flexion measured in the Orthoload dataset, it is possible that high error during deep flexion accounted for the high RMSE results. It is relevant to point out that for movements of deep flexion (> 80°) (comparable to deep knee bend), the error in joint contact forces calculated using musculoskeletal simulation models reported in the literature is high (up to 1.9BW) (Schellenberg et al., 2018). Results, such as the present ones, for KneeSIM Lab and for other models presented in reports in the literature (for example, Schellenberg et al., 2018) reinforce the point that, for musculoskeletal simulation model

validation, acceptable subject data should be those obtained from subjects (who have received TKA) engaged in movements that are physiologically relevant to the majority of TKA patients, such as level walking, chair rise, and stair climbing.

With VCTK simulation development, the RMSE obtained was much higher than that reported by Grand Challenge Competition winners. However, it is important to note that the VCTK model was developed utilizing a scaled skeletal model, instead of the exact bone anatomies provided in the 2014 SimTK dataset. The use of the scalable anatomic model was desirable for this application so that body measurements (height, weight, limb length) could be parameterized and included in embodiments of the KPEs. The GEBOD scaling algorithms used in VCTK model development utilized subject age, weight, and height to determine body segment length and mass. Scaling of the anatomy, which relied entirely on the GEBOD algorithms, produced undersized models of the femur and the tibia compared to models of these bones when CT data were used. Specifically, compared to CT-derived models, GEBOD-scaled femur and tibia models were 35 mm and 19 mm undersized, respectively (Figure 31). Furthermore, when musculoskeletal simulation models are used, marker position accuracy exerts a strong influence on the accuracy of the simulations (Lund et al., 2015). Improper scaling of the bone models could have resulted in inaccurate placement of the gait marker relative to the patient bone model. Although LifeModeler's use of motion agents offsets the effect of gait marker inaccuracies, the magnitude of this impact was not quantified. Only after further refinement of the VCTK bone models utilizing measurements from CT data was the optimized RMSE of 0.83 x body weight achieved.



Figure 30. Femur model comparison showing GEBOD-scaled model (grey) overlaid on CT-derived model (blue) from SimTK dataset. GEBOD-scaled model is undersized by 35 mm.

While RMSE for tibiofemoral joint contact forces calculated using the Optimized VCTK model were higher than those reported by past winners of the Grand Challenge competition, the joint contact force profiles were qualitatively very similar to those in the SimTK dataset (R^2 for force magnitude and moment magnitude = 0.84 and 0.69, respectively). This strong correlation allowed the use of the Optimized VCTK model simulations for development of the KPEs, which led to calculation of optimal position and orientation of implant components.

5.3. Knee Performance Equations (KPEs)

In the development of the KPEs, the criterion used to select the output response was that each response is related to factors that have been cited in the literature as those that are strongly correlated with TKA patient satisfaction scores. To provide an advantage over surgical planning tools used in current-generation RASSs, the output factor response list was further scrutinized to include only factors that could be influenced by active tissue loading.

For the proof-of-concept study, the first set of output responses used were those that are associated with patellar mechanics and mid-flexion instability. Femoral component alignment and rotation have a major impact on patellar tracking and rotation (Donell, 2018; Keshmiri et al., 2016; Merican et al., 2011). In spite of this knowledge, current-generation raTKAs do not consider the impact of femoral component alignment on patellar kinematics, which can vary greatly among component designs (Donell, 2018). Therefore, any information provided to the surgeon relating implant orientation to patellar function could have major value to the patient. The second set of output responses are those related to ligament strains, as improper balance of the ligaments has been associated with poor joint stability and poor proprioception (Evangelista et al., 2018; Kumar and Shahzad, 2019).

When TKA is performed using current-generation RASSs, although soft tissue balance is considered, characterization of the tissues is done by passive ROM assessment in most cases. The KPEs could be used to provide insight into, for example, ligament behaviors with the added effect of active muscle forces and gravity and the influence of muscle forces on ligament strain during normal activities, such as level walking.

Goodness-of-fit for the derived KPEs when KneeSIM Lab model (deep-knee-bend simulation) and VCTK model (single-leg stance simulation) were used were determined using R^2 values for each regression. With KneeSIM Lab, the equations with the highest R^2 were those related to ligament strain and the tibiofemoral joint contact force. For VCTK, the equations with the highest R^2 were those related to quadriceps force, knee kinematics, and ligament strain.

5.4. Proof-of-Concept Study

For this study, the SimTK dataset was used. For the subjects in this dataset, the instrumented prosthesis (tibial insert) was implanted using a conventional technique (standard mid-vastus approach (D’Lima et al., 2006)). However, 1) the implantation was based on a 3D preoperative plan developed with the use of a patient CT scan; and 2) specialized instrumentation was developed to accommodate the instrumented tibia tray. Thus, even though a RASS was not used, the aforementioned features of the implantation method are such that it could be argued that the method could be considered a plausible proxy for a RASS. In other words, the data obtained from the subjects could be regarded as clinical data.

It was understood that providing the ideal implant component position is a multi-factorial problem that requires a patient-specific solution. For example, a patient who is at a higher risk of developing post-operative instability may have a different relative optimal implant component position than one who is at risk for anterior knee pain. With this in mind, the proof-of-concept study was designed to optimize those responses associated with instability (MCL and LCL strain), as the KPEs associated with those responses had good correlation for both optimized KneeSIM Lab and optimized VCTK models.

In many of the simulation trials, the ligament strains determined were higher than *in vitro* ligament strains in the native knee (2%-8% for the MCL and 1.5%-5% for the LCL compared to results in Figure 27 and 28 (Delpont et al., 2015, 2013)). Thus, optimization criterion for the proof-of concept study was configured to keep MCL and LCL strains to levels consistent with those in the native knee while maintaining a maximum tibiofemoral joint contact force consistent with the loading data given in the SimTK dataset, thus providing a clinically-acceptable implant component position.

As demonstrated, both Optimized KneeSIM Lab and Optimized VCTK models suggested a different position and orientation for both femoral and tibial components, with the largest difference being in the superior-inferior position (gap length) (Tables 10 and 11). In both cases, suggested implant component position and orientation resulted in a larger implant component gap; that is, the femoral component moved superiorly and the tibial component moved inferiorly with more bone removed from both the tibia and the femur. Intuitively, this result was expected as the optimization goal was to reduce strains in the ligaments. With a larger planned gap between the tibia and the femur, a smaller deformation of the ligament structures was required during movement. Additionally, this result underscores the usefulness of the KPEs as far as prediction of behavior of a simulation model was concerned.

The KPEs also suggested an achievably different mediolateral implant position. For the Optimized KneeSIM Lab and VCTK models, suggested femoral component movement was consistent with that of the tibial component. In other words, both implant components were moved in the same direction by approximately the same amount (with KneeSIM Lab: minimum difference in medial-lateral (ML) movements of the femoral and tibial components of 1.4 mm and 2.1 mm, respectively; with VCTK: minimum difference in ML movements of the femoral and tibial components of 1.68 mm and 2.48 mm, respectively). However, KneeSIM Lab results suggested movement in the lateral direction whereas VCTK results suggested movement in the medial direction. It is worth recalling that the KPE sets for KneeSIM Lab and SimTK were developed using different movement patterns (deep knee bend for the former vs. single-leg stance for the latter). It is well documented that different movements cause different loading conditions in the knee, and this difference translates to difference in strain behavior in the collateral ligaments (Delpport et al., 2015, 2013).

The actual passive surgical plan was presented relative to the conceptual active surgical plan considering the expected errors for both conventional and robotic techniques (Jaramaz et al., 2018; Lonner et al., 2015). That is, with known expected implant component position and orientation error for conventional (actual passive) and robotic (conceptual active) TKA, it could be determined if SEISPT would suggest an achievably different implant component position and orientation. If the RMS equivalent error (Equation (12)) between conventional and robotic techniques is less than the difference between the actual passive and conceptual active implant pose, the position and orientation suggested by SEISPT would be markedly different for patient PS than what was planned by surgeons participating in the SimTK study.

$$|\text{RMS}(\text{conventional planned} - \text{conventional achieved}) - \text{RMS}(\text{robotic planned} - \text{robotic achieved})| = \text{RMSE Equivalent} \quad (12)$$

Utilizing the KPEs from optimized KneeSIM Lab model, achievably different implant positions were suggested in the ML, AP and SI directions for both femoral and tibial components; while achievably different implant orientations were suggested for varus-valgus alignment and internal-external rotation. For KPEs from optimized VCTK model, achievable implant positions were demonstrated in the ML, AP, and SI directions for the femoral component and in the ML and SI directions for the tibial component. Only orientation in varus-valgus alignment was achievably different for the femoral component.

The purpose of the proof-of-concept study was to determine if use of the simulation-enhanced tool developed in the present study (SEISPT) could result in position and orientation of an implant component that were different from those that were achieved on a subject who had received a TKA, as reported in an *in vivo* study that could be considered a proxy for a clinical RASS case. The results show that there is a difference and it is noticeable, thereby hinting at the

potential for SEISPT to be part of the surgical plan for a next-generation raTKA system.

However, it should be noted that development of a surgical plan is multi-factorial. This means that ligament behavior and patella tracking are only two of myriad parameters that should be considered, some others being restoration of tibiofemoral alignment, restoration of proper kinematic function, and proper balancing of the soft tissue envelope.

5.5. Feasibility of a Simulation-Enhanced Tool for Robotics-Assisted TKA

RASSs feature either preoperative planning or intraoperative planning. With intraoperative planning, the surgeon characterizes the anatomy of the patient while he/she is on the operating table. This imposes a time constraint on the planning because the risk for surgery-acquired infection increases with increase of time on the operating table (Kapadia et al., 2016; Pugely et al., 2015; Song et al., 2013). Thus, if a new simulation-based planning tool is to be considered for use in intraoperative surgery planning it must be time-efficient; that is, the time to deploy it should be comparable to or, better still, shorter than time spent when current-generation intraoperative planning methods are used. The key to the time-efficiency of SEISPT is the optimization procedure that culminates in the generation of the KPEs, which, in turn, are used to compute a series of model output responses. DOE and linear regression allowed for the characterization of hundreds of simulation runs in a single linear equation. Optimization of the KPEs does not require sophisticated computer hardware or software and can be achieved in a very short time (on the order of seconds). Thus, SEISPT could easily be deployed on any current-generation raTKA system. Schematic drawings of the workflow for a specific current-generation raTKA system that utilizes intraoperative surgical planning (namely, Navio Surgical System) and the proposed workflow for that same system but with SEIST incorporated are

presented in Figures 32 and 33, respectively. As shown (with highlighted items in Figure 33), there are only very small differences between the proposed and existing workflows and, as such, utilization of the proposed workflow plan could be achieved with minimal disruption and, hence, minimal adverse effect on surgery time and cost.

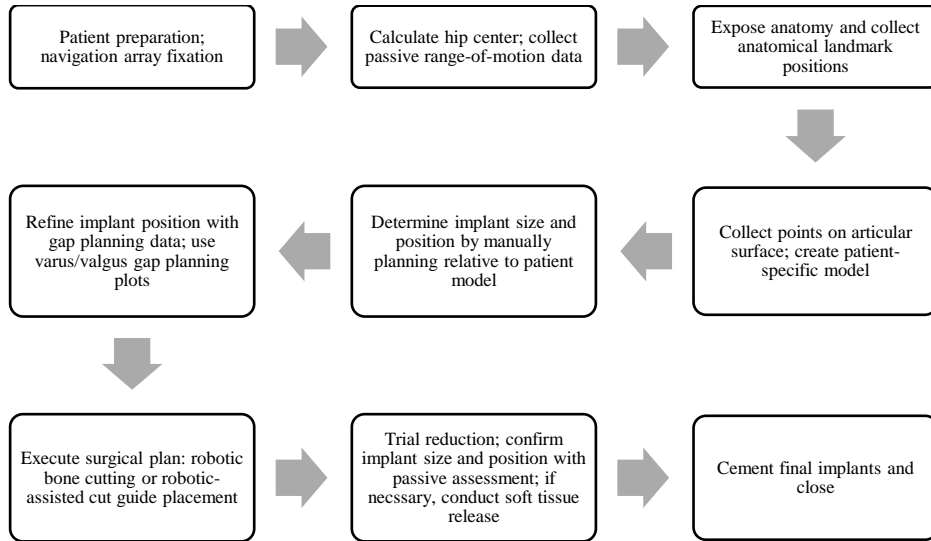


Figure 31. Schematic diagram of workflow for a current-generation robotics-assisted total knee arthroplasty with intraoperative surgical planning (NAVIO Surgical System Surgical Technique for Total Knee Arthroplasty, 2018).

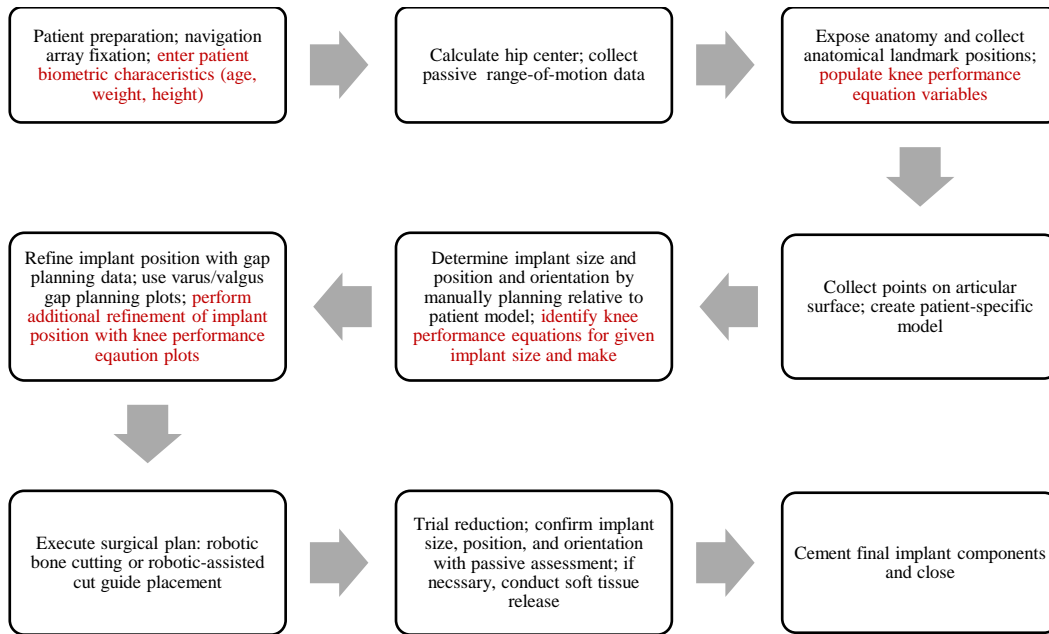


Figure 32. Schematic diagram of workflow plan with the Navio Surgical System but with incorporation of the proposed simulation-enhanced intraoperative surgical planning tool. Steps unique to the incorporation of SEISPT are highlighted in red.

It is envisaged that a large library of KPEs could be stored in the surgical robotics system computer. Each implant component size and design would have its own set of KPEs derived from DOE-designed simulation studies. A common set of patient-specific input factors would be utilized for each set of KPEs. These variables are populated in the patient characterization steps and the only additional information provided to the system would be patient biometric characteristics, such as age, height, and weight. Other patient-specific information is already being recorded to deform the patient-specific anatomical model. Implant component sizing would be done by examining the size of the bone model, as is done in current raTKA surgery practice. Once an implant component size is selected, the corresponding KPE set

would be selected from the library and used to generate visual informatics displays that guide final implant component position and orientation.

One example of visual informatics display that could be deployed on the surgical robotic system planning screen is a multi-objective surface response map. An example surface response map for MCL and LCL strains relative to the anterior-posterior (AP) and medial-lateral (ML) positions of the femoral component was created in Matlab (See Appendix C) and is shown in detail in Figure 34. In one rendering of the visual informatics display, the current AP and ML positions are shown on the surface response map and updated as the proposed AP and ML positions are changed during planning. The use of visual informatics displays to communicate KPE results to the surgeon is especially attractive because it allows the surgeon to choose which parameters are given priority. For example, for a particular patient, restoration of proper patellar kinematics may be more important than ligament balancing. In that case, the surgeon can focus on the patellar surface response map to do initial positioning of the implant component, and, then, modify the position using the ligament balancing surface response map.

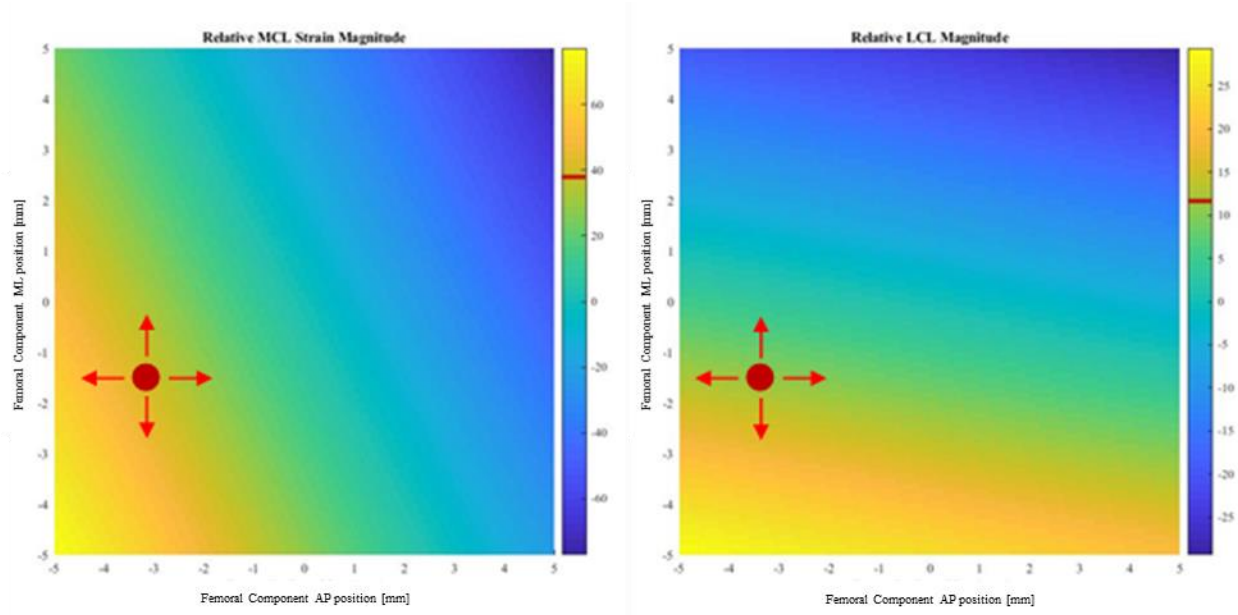


Figure 33. Example surface response maps for medial collateral ligament (MCL) and lateral collateral ligament (LCL) strains relative to anterior-posterior (AP) and medial-lateral (ML) positions of the femoral component, as obtained from Optimized KneeSIM Lab knee performance equations. As the position of the femoral component is adjusted during the robotic case planning, the surgeon is given visual feedback via the maps.

5.6 Study Limitations

The study has five limitations. First, only two datasets were used in the validation of the musculoskeletal simulation models, these being the only ones that, to the author's knowledge, were publicly-available during the course of the study. Additionally, the two datasets utilized to develop KPEs were not consistent for the different modeling fidelities (Orthoload dataset for KneeSIM Lab and SimTK dataset for VCTK). Ideally, a consistent set of data would have been used to highlight the differences in recommended implant position between the two models, but the SimTK dataset did not provide loading data for a deep knee bend movement. Thus the Orthoload dataset was utilized instead.

Second, in the development of the KPEs, more runs could have been added to the experimental design or a higher-order regression could have been utilized. However, each of these approaches has its shortcoming: longer processing time (former approach) and increased computational burden (latter approach). As the ultimate use of the KPEs was to calculate the optimal position and orientation of implant components, linear equations were used because this approach provided the best balance between processing time and computational burden.

Third, the KPE fits could be improved through the use of various transform methodologies. For example, to achieve comparable fit to the KPEs generated using optimized KneeSIM Lab model, the results from optimized VCTK model were augmented with an inverse response transform (Table 13). While, in this case, an inverse response transform improved fit for those KPEs relating to MCL and LCL strain, different transform variants would be required to maximize fit across all KPEs. Thus, the complexity and computational burden associated with improving fit through data manipulation would be impractical to implement on a large scale.

Table 13. Metrics of improvement in regression fit of knee performance equations, with results from optimized VCTK model. Regression and transformation was carried out using Adams Insight software.

Model	MCL ^a Strain (Bundle 1 ^b)	MCL Strain (Bundle 2 ^b)	MCL Strain (Bundle 3 ^b)	LCL ^a Strain (Bundle 1 ^b)	LCL Strain (Bundle 2 ^b)
R ² (Fit using linear regression)	0.744	0.750	0.755	0.746	0.790
R ² (Fit using inverse response transform method)	0.871	0.884	0.910	0.838	0.867

^aMCL: medial collateral ligament; LCL: lateral collateral ligament

^bMCL and LCL elements in LifeModeler are divided into several parallel elastic elements (that is, bundles). There are 3 bundles in the MCL model and 2 bundles in the LCL model.

Fourth, it was difficult to define the RMS equivalent error, which was used to determine the expected difference in implant orientation and position between conventional and SEISPT suggested surgical plans. Data from Lonner et al. (2015) and Jaramaz et al. (2018) were utilized. However, the Lonner study did not provide the RMSE in expected position and orientation for both femoral and tibial components individually. Thus RMSE measures from Jaramaz et al. (2018) were averaged for femoral and tibial components and this average was used to calculate the RMS equivalent error shown in Equation 12. Furthermore, Lonner et al. (2015) study related to accuracy associated with UKA implants. It was assumed that accuracy levels for TKA procedures would be equivalent.

Finally, there were a number of patient-specific simulation model parameters that were not prescribed by patient data included in the Orthoload or SimTK datasets. These include, for example, tissue properties, attachment sites, joint contact properties, and implant material properties. To avoid bias in the model validation and optimization, unspecified model parameters were estimated using data available in current literature. Furthermore, unspecified parameters relating to the properties and locations of the soft tissues of the knee were optimized during validation and were maintained for development of the KPE's. Therefore implant locations suggested by the KPE's were derived from models that most closely replicate the joint loading patterns demonstrated in the empirical data sets.

5.7. Potential Clinical Impact

High dissatisfaction rates with primary TKA are strongly associated with high incidences of anterior knee pain and poor proprioception (Almeida et al., 2016; Innocenti et al., 2011; Kumar and Shahzad, 2019; Shervin et al., 2015). For the former group of patients, the most

common treatment option is revision joint replacement. Compared to primary TKA, revision TKA is more complex and puts the patient at higher risk for postoperative complications, such as pulmonary embolism, deep vein thrombosis, and acute myocardial infarction (Werner et al., 2015). Furthermore, the incidence of infection in revision TKA is much higher than that for primary TKA (5.5% versus 0.7% of cases) (Werner et al., 2015). Also, revision TKA is more expensive than primary TKA; for example, in the United States, \$16,243 per case versus \$49,000 per case (Bhandari et al., 2012; Shankar et al., 2016). A simulation-enhanced intraoperative surgical planning tool that utilizes active joint kinematics as part of the input (in this case, SEISPT), when used in a raTKA system, could improve patient satisfaction and, hence, reduce the need for revision TKA.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE STUDY

6.1. Conclusions

The following are the conclusions reached:

- (1) A single-leg LifeModeler plug-in (KneeSIM Lab) was utilized for the low-fidelity musculoskeletal simulation model that included the knee joint. Validation of the model was conducted with tibiofemoral joint contact force data obtained from 8 subjects while each engaged in deep knee bend. Each subject had received an instrumented tibial tray in the left knee and the data were contained in a publicly available dataset (Orthoload dataset). RMSE between an optimized KneeSIM Lab model and the Orthoload data was 409 N (equivalent to 0.48 x mean body weight (BW) of the 8 subjects), which was within the range of errors reported for simulation modeling activities of this type. Furthermore, it was determined that simulation of deep knee bend, like the one modeled with KneeSIM Lab, is less reliable than simulation of other activities, such as gait. LifeModeler plug-in (VCTK) was used in conjunction with various results from tests on one subject (PS) contained in another publically-available dataset (2014 SimTK Grand Challenge Competition dataset) to create a complex full-body simulation model that included the knee joint (high-fidelity model). The subject had received an instrumented tibial component and data were collected while the subject performed a series of normal daily activities. As joint contact force data were not available for gait activities, gait data from a single-leg stance was utilized to train the full-body model. RMSE between an optimized VCTK model and the data reported for Subject PS while he/she performed a single-leg stance (618 N or 0.83 x subject body weight) was larger than errors reported by the

winner of the 2014 Competition. It is to be noted, however, that the winner (as did all winners of the Competition in the different years) used exact models created from CT data whereas, in the present work, a scaled skeletal model was used in the VCTK simulation. In musculoskeletal simulations, it has been shown that improper scaling of the skeletal model has a strong effect on the simulated joint contact forces, perhaps related to inaccuracies of gait marker placement relative to the anatomy. While the VCTK model consistently underestimated the joint contact forces during single-leg stance, a strong correlation was found between simulation results and experimentally-obtained data ($R^2 = 0.84$). When the whole collection of results is considered, there is indication that VCTK model could be used for trend analysis to optimize implant component position and orientation when the surgical plan includes active tissue balancing results.

- (2) A number of input parameters and outputs from runs of KneeSIM Lab and VCTK simulations were used to generate KPEs. These input factors were specifically selected to correspond with those used to develop a patient-specific model for anatomical planning. KPEs could be used as a key element of the surgical planning tool for a raTKA system; for example, a library of KPEs could be developed and deployed on a surgical system that populates variable values during intraoperative anatomical characterization. Optimization of KPEs is not computationally burdensome, and, therefore, could be done in real time. Thus, optimization results for implant component position and orientation could be communicated to the surgeon intraoperatively with the use of visual informatics

displays, which are currently used for soft tissue balancing across several raTKA systems.

(3) A proof-of-concept study was conducted in which achieved positions and orientations of components for Subject PS were compared to corresponding results computed from optimization of KPEs. The optimization objectives were set to position and orient femoral and tibial components to minimize collateral ligament strains while maintaining a physiologically relevant patellofemoral joint contact force. The aforementioned comparison showed that, for each of the implant components, the simulation tool provided a different position and orientation. Furthermore, and, more importantly, the implant component positions and orientations obtained through use of the simulation tool were deemed to be clinically achievable. Thus, the present results suggest that there is potential to improve surgical treatment with insights related to active joint mechanics.

(4) With respect to a specific current-generation raTKA system (Navio Surgical System), schematic drawings are presented of the surgical workflow that includes SEISPT as well as of the established surgical workflow (that is, one does not include SEISPT). It could be seen that there are only small differences between these two diagrams, suggesting that deployment of the Navio Surgical System that includes SEISPT could be accomplished with minimal disruption and, hence, minimal increase in surgery time and cost.

6.2. Recommendations

The following recommendations are made for future study:

- 1) Essentially, derivation of the KPEs and their optimization could be considered to be the core of the intraoperative planning tool for a raTKA system presented in the present study (designated, SEISPT). To increase the efficiency with which these equations are obtained (and, hence, shorten the time of execution of the surgical plan) requires that the response equations of the musculoskeletal simulation model used be refined with no associated cost in computational burden (and, hence, increase in simulation run time). Way(s) to do so should be explored.
- 2) Patient-specific characteristics that were not utilized in the generation of the KPEs should be added. Examples are age, height, and weight, length of femur, the length of tibia, native knee varus or valgus limb alignment, and native knee rotational alignment. This will allow for use of KPEs across populations of diverse patients. However, more involved simulation and validation work is associated with this task as it is unclear how biomechanical outputs scale with anthropometric properties.
- 3) The question as to whether a raTKA system that includes SEISPT as the intraoperative planning tool leads to improved clinical and patient outcomes for TKA patients should be investigated in well-planned randomized controlled trials. Such a trial should involve a large number of patients (≥ 100), surgeons (≥ 10) and sites (≥ 3) and have a long follow-up (≥ 5 years).

References

- Almeida, G.P.L., Silva, A.P. de M.C.C.E., França, F.J.R., Magalhães, M.O., Burke, T.N., Marques, A.P., 2016. Q-angle in patellofemoral pain: relationship with dynamic knee valgus, hip abductor torque, pain and function. *Rev. Bras. Ortop.* 51, 181–6.
- American Joint Replacement Registry, 2018. Fifth AJRR Annual Report on Hip and Knee Arthroplasty Data 2018. *Am. Jt. Replace. Regist.*
- Banks, S.A., 2017. What postoperative outcome measures link joint stability to patient satisfaction?, in: *Journal of the American Academy of Orthopaedic Surgeons*. pp. S40–S43.
- Bell, S.W., Anthony, I., Jones, B., MacLean, A., Rowe, P., Blyth, M., 2016. Improved accuracy of component positioning with robotic-assisted unicompartmental knee arthroplasty: data from a prospective, randomized controlled study. *J. Bone Joint Surg. Am.* 98, 627–35.
- Bellamy, N., Buchanan, W.W., Goldsmith, C.H., Campbell, J., Stitt, L.W., 1988. Validation study of WOMAC: a health status instrument for measuring clinically important patient relevant outcomes to antirheumatic drug therapy in patients with osteoarthritis of the hip or knee. *J. Rheumatol.* 15, 1833–40.
- Bergmann, G., Bender, A., Graichen, F., Dymke, J., Rohlmann, A., Trepczynski, A., Heller, M.O., Kutzner, I., 2014. Standardized loads acting in knee implants. *PLoS One* 9, e86035.
- Bhandari, M., Smith, J., Miller, L.E., Block, J.E., 2012. Clinical and economic burden of revision knee arthroplasty. *Clin. Med. Insights. Arthritis Musculoskelet. Disord.* 5, 89–94.
- Blackburn, T., Craig, E., 1980. Knee anatomy: a brief review. *Phys. Ther.* 60, 1556–1560.
- Boresi, A., Schmidt, R., Mei, F., 2002. *Engineering Mechanics: Dynamics, Applied Mechanics Reviews*. CL-Engineering.
- Brooks, P., 2009. Seven cuts to the perfect total knee. *Orthopedics* 32, 680–682.

- Bryan, S., Goldsmith, L.J., Davis, J.C., Hejazi, S., MacDonald, V., McAllister, P., Randall, E., Suryaprakash, N., Wu, A.D., Sawatzky, R., 2018. Revisiting patient satisfaction following total knee arthroplasty: A longitudinal observational study. *BMC Musculoskelet. Disord.* 19, 423–431.
- Choi, Y.-J., Ra, H.J., 2016. Patient satisfaction after total knee arthroplasty. *Knee Surg. Relat. Res.* 28, 1–15.
- Churchill, J.L., Khlopas, A., Sultan, A.A., Harwin, S.F., Mont, M.A., 2018. Gap-balancing versus measured resection technique in total knee arthroplasty: a comparison study. *J. Knee Surg.* 31, 13–16.
- Clary, C.W., Fitzpatrick, C.K., Maletsky, L.P., Rullkoetter, P.J., 2013. The influence of total knee arthroplasty geometry on mid-flexion stability: An experimental and finite element study. *J. Biomech.* 46, 1351–1357.
- Clegg, D.O., Reda, D.J., Harris, C.L., Klein, M.A., O'Dell, J.R., Hooper, M.M., Bradley, J.D., Bingham, C.O., Weisman, M.H., Jackson, C.G., Lane, N.E., Cush, J.J., Moreland, L.W., Schumacher, H.R., Oddis, C. V., Wolfe, F., Molitor, J.A., Yocum, D.E., Schnitzer, T.J., Furst, D.E., Sawitzke, A.D., Shi, H., Brandt, K.D., Moskowitz, R.W., Williams, H.J., 2006. Glucosamine, chondroitin sulfate, and the two in combination for painful knee osteoarthritis. *N. Engl. J. Med.* 354, 795–808.
- Coon, T., Hernandez, A., Conditt, M., 2016. Short term survivorship and outcomes of robotically assisted bi-compartmental arthroplasty. *Bone Jt. J.* 98-B, 8–9.
- D'Lima, D.D., Patil, S., Steklov, N., Slamin, J.E., Colwell, C.W., 2006. Tibial forces measured in vivo after total knee arthroplasty. *J. Arthroplasty* 21, 255–262.
- Daines, B.K., Dennis, D.A., 2014. Gap balancing vs. measured resection technique in total knee

- arthroplasty. *Clin. Orthop. Surg.* 6, 1–8.
- Davies, B., 2015. Robotic surgery - A personal view of the past, present and future. *Int. J. Adv. Robot. Syst.* 12, 54.
- Dawson, J., Fitzpatrick, R., Murray, D., Carr, A., 1998. Questionnaire on the perceptions of patients about total knee replacement. *J. Bone Joint Surg. Br.* 80, 63–9.
- De Coninck, S., Smis, J., Victor, J., De Baets, P., Verstraete, M., 2016. Computational modeling of a dynamic knee simulator. *Int. J. Sustain. Constr. Des.* 6, 8–16.
- Delpont, H., Labey, L., De Corte, R., Innocenti, B., Vander Sloten, J., Bellemans, J., 2013. Collateral ligament strains during knee joint laxity evaluation before and after TKA. *Clin. Biomech.* 28, 777–782.
- Delpont, H., Labey, L., Innocenti, B., De Corte, R., Vander Sloten, J., Bellemans, J., 2015. Restoration of constitutional alignment in TKA leads to more physiological strains in the collateral ligaments. *Knee Surgery, Sport. Traumatol. Arthrosc.* 23, 2159–2169.
- DesJardins, J.D., Walker, P.S., Haider, H., Perry, J., 2000. The use of a force-controlled dynamic knee simulator to quantify the mechanical performance of total knee replacement designs during functional activity. *J. Biomech.* 33, 1231–1242.
- Diagram Of The Muscles In The Leg and Leg Muscle Charts [WWW Document], 2019. . *Hum. Anat. Libr.* URL <https://wirindiagrampedia.com/> (accessed 5.19.19).
- Donell, S., 2018. Patellar tracking in primary total knee arthroplasty. *EFORT open Rev.* 3, 106–113.
- Drake, R.L., Vogl, Wayne, A., Mitchell, A.W.M., 1989. *Gray’s Anatomy for Students*, 2nd E, Journal of Chemical Information and Modeling. Churchill Livingstone.
- Dunbar, M., Richardson, G., Robertsson, O., 2013. I can’t get no satisfaction after my total knee

- replacement: rhymes and reasons. *Bone Joint J.* 95-B, 148–152.
- Evangelista, P.J., Laster, S.K., Lenz, N.M., Sheth, N., Schwarzkopf, R., 2018. A computer model of mid-flexion instability in a balanced total knee arthroplasty. *J. Arthroplasty* 33, S265–S269.
- Ewurum, C.H., Guo, Y., Pagnha, S., Feng, Z., Luo, X., 2018. Surgical navigation in orthopedics: Workflow and system review, in: *Advances in Experimental Medicine and Biology*. pp. 47–63.
- Flandry, F., Hommel, G., 2011. Normal anatomy and biomechanics of the knee. *Sport. Med. Arthrosc. Rev.* 19, 82–92.
- Foran, J., 2016. Unicompartamental Knee Replacement [WWW Document]. OrthoInfo - AAOS. URL <https://orthoinfo.aaos.org/en/treatment/unicompartamental-knee-replacement/> (accessed 5.20.19).
- Fregly, B.J., Besier, T.F., Lloyd, D.G., Delp, S.L., Banks, S.A., Pandy, M.G., D’Lima, D.D., 2012. Grand challenge competition to predict in vivo knee loads. *J. Orthop. Res.* 30, 503–513.
- Ganko, A., Donnelly, W., Brennan, S., English, H., Halliday, B., Crawford, R., 2017. Early results & accuracy of the MAKO robotic unicompartamental knee replacement. *Orthop. J. Sport. Med.* 5, 5.
- Gejo, R., McGarry, M.H., Jun, B.J., Hofer, J.K., Kimura, T., Lee, T.Q., 2010. Biomechanical effects of patellar positioning on intraoperative knee joint gap measurement in total knee arthroplasty. *Clin. Biomech.*
- Gibbons, K.D., Clary, C.W., Rullkoetter, P.J., Fitzpatrick, C.K., 2019. Development of a statistical shape-function model of the implanted knee for real-time prediction of joint

- mechanics. *J. Biomech.* 88, 55–63.
- Gromov, K., Korchi, M., Thomsen, M.G., Husted, H., Troelsen, A., 2014. What is the optimal alignment of the tibial and femoral components in knee arthroplasty? *Acta Orthop.* 85, 480–7.
- Halloran, J.P., Clary, C.W., Maletsky, L.P., Taylor, M., Petrella, A.J., Rullkoetter, P.J., 2010. Verification of predicted knee replacement kinematics during simulated gait in the kansas knee simulator. *J. Biomech. Eng.* 132, 081010.
- Harris, B., 2019. Total Knee Replacement [WWW Document]. Hip Knee Replace. URL <http://www.bartonharrismd.com/total-knee-replacement.html>
- Hast, M.W., Piazza, S.J., 2013. Dual-joint modeling for estimation of total knee replacement contact forces during locomotion. *J. Biomech. Eng.* 135, 021013.
- Heyse, T.J., Khefacha, A., Peersman, G., Cartier, P., 2012. Survivorship of UKA in the middle-aged. *Knee* 19, 585–591.
- Hicks, J.L., Uchida, T.K., Seth, A., Rajagopal, A., Delp, S.L., 2015. Is my model good enough? Best practices for verification and validation of musculoskeletal models and simulations of movement. *J. Biomech. Eng.* 137, 020905.
- Huaining Cheng, Obergefell, L., Rizer, A., 1996. The development of the GEBOD program, in: *Proceedings of the 1996 Fifteenth Southern Biomedical Engineering Conference.* IEEE, Dayton, OH, pp. 251–254.
- Innocenti, B., Pianigiani, S., Labey, L., Victor, J., Bellemans, J., 2011. Contact forces in several TKA designs during squatting: A numerical sensitivity analysis. *J. Biomech.* 44, 1573–1581.
- Ishida, K., Shibamura, N., Matsumoto, T., Sasaki, H., Takayama, K., Toda, A., Kuroda, R.,

- Kurosaka, M., 2015. Factors affecting intraoperative kinematic patterns and flexion angles in navigated total knee arthroplasty. *Knee Surgery, Sport. Traumatol. Arthrosc.* 23, 1741–1747.
- Jacofsky, D.J., Allen, M., 2016. Robotics in arthroplasty: a comprehensive review. *J. Arthroplasty* 31, 2353–2363.
- Jaramaz, B., Mitra, R., Nikou, C., Kung, C., 2018. Technique and accuracy assessment of a novel image-free handheld robot for knee arthroplasty in bi-cruciate retaining total knee replacement, in: *The 18th Annual Meeting of the International Society for Computer Assisted Orthopaedic Surgery*. pp. 98–101.
- Jaramaz, B., Picard, F., Gregori, A., 2013. Evaluating joint laxity in UKR using Navio-PFS. *Orthop. Proc.* 95-B.
- Jo, C.H., Lee, Y.G., Shin, W.H., Kim, H., Chai, J.W., Jeong, E.C., Kim, J.E., Shim, H., Shin, J.S., Shin, I.S., Ra, J.C., Oh, S., Yoon, K.S., 2014. Intra-articular injection of mesenchymal stem cells for the treatment of osteoarthritis of the knee: A proof-of-concept clinical trial. *Stem Cells* 32, 1254–1266.
- Joskowicz, L., Hazan, E.J., 2018. Computer-aided orthopedic surgery: Incremental shift or paradigm change?, in: *Silva Sequeira, J. (Ed.), Advances in Experimental Medicine and Biology*. Springer, pp. 21–30.
- Jung, Y., Phan, C.-B., Koo, S., 2015. Intra-articular knee contact force estimation during walking using force-reaction elements and subject-specific joint model. *J. Biomech. Eng.* 138, 021016.
- Kahlenberg, C.A., Nwachukwu, B.U., McLawhorn, A.S., Cross, M.B., Cornell, C.N., Padgett, D.E., 2018. Patient satisfaction after total knee replacement: A systematic review. *HSS J.*

14, 192–201.

- Kang, K.T., Koh, Y.G., Son, J., Kwon, O.R., Lee, J.S., Kwon, S.K., 2017. Influence of increased posterior tibial slope in total knee arthroplasty on knee joint biomechanics: a computational simulation study. *J. Arthroplasty* 33, 572–579.
- Kapadia, B.H., Berg, R.A., Daley, J.A., Fritz, J., Bhave, A., Mont, M.A., 2016. Periprosthetic joint infection. *Lancet* 387, 386–94.
- Karunaratne, S., Duan, M., Pappas, E., Fritsch, B., Boyle, R., Gupta, S., Stalley, P., Horsley, M., Steffens, D., 2018. The effectiveness of robotic hip and knee arthroplasty on patient-reported outcomes: A systematic review and meta-analysis. *Int. Orthop.* 1–13.
- Kayani, B., Konan, S., Tahmassebi, J., Pietrzak, J.R.T., Haddad, F.S., 2018. Robotic-arm assisted total knee arthroplasty is associated with improved early functional recovery and reduced time to hospital discharge compared with conventional jig-based total knee arthroplasty. *Bone Joint J.* 100-B, 930–937.
- Keshmiri, A., Maderbacher, G., Baier, C., Zeman, F., Grifka, J., Springorum, H.R., 2016. Significant influence of rotational limb alignment parameters on patellar kinematics: an in vitro study. *Knee Surgery, Sport. Traumatol. Arthrosc.* 24, 2407–2414.
- Khlopas, A., Sodhi, N., Sultan, A.A., Chughtai, M., Molloy, R.M., Mont, M.A., 2018. Robotic arm-assisted total knee arthroplasty. *J. Arthroplasty* 33, 2002–2006.
- Kim, Y.-H., Park, J., 2019. Does a Robotic-Assisted Total Knee Arthroplasty Improve Clinical Outcome in Young Patients?, in: *American Academy of Orthopedic Surgeons Annual Meeting*. AAOS, Las Vegas, NV.
- Kim, Y.-H., Park, W.-M., Phuong, B.T.T., 2013. Effect of Joint Center Location on In-Vivo Joint Contact Forces During Walking, in: *ASME 2010 Summer Bioengineering Conference*,

- Parts A and B. ASME, Naples, FL, p. 267.
- Kirking, B., Krevolin, J., Townsend, C., Colwell, C.W., D'Lima, D.D., 2006. A multiaxial force-sensing implantable tibial prosthesis. *J. Biomech.* 39, 1744–1751.
- Knarr, B.A., Higginson, J.S., Zeni, J.A., 2016. Change in knee contact force with simulated change in body weight. *Comput. Methods Biomech. Biomed. Engin.* 19, 320–323.
- Knowlton, C.B., Wimmer, M.A., Lundberg, H.J., 2013. Grand challenge competition: A parametric numerical model to predict in vivo medial and lateral knee forces in walking gaits, in: *ASME 2012 Summer Bioengineering Conference*. ASME, Fajardo, Puerto Rico, p. 199.
- Kon, E., Filardo, G., Delcogliano, M., Marcacci, M., Mandelbaum, B., Buda, R., Timoncini, A., Giannini, S., Fornasari, P.M., 2011. Platelet-rich plasma intra-articular injection versus hyaluronic acid viscosupplementation as treatments for cartilage pathology: From early degeneration to osteoarthritis. *J. Arthrosc. Relat. Surg.* 27, 1490–1501.
- Koulalis, D., O'Loughlin, P.F., Plaskos, C., Kendoff, D., Cross, M.B., Pearle, A.D., 2011. Sequential versus automated cutting guides in computer-assisted total knee arthroplasty. *Knee* 18, 436–442.
- Kremers, H.M., Larson, D.R., Crowson, C.S., Kremers, W.K., Washington, R.E., Steiner, C.A., Jiranek, W.A., Berry, D.J., 2014. Prevalence of total hip and knee replacement in the United States. *J. Bone Jt. Surg. - Am. Vol.* 97, 1386–1397.
- Kumar, R., Shahzad, R., 2019. Intraoperative assessment of mid-flexion instability in primary total knee replacement. *J. Pakistan Orthop. Assoc.* 30, 93–97.
- Kurtz, S., Ong, K., Lau, E., Mowat, F., Halpern, M., 2007. Projections of primary and revision hip and knee arthroplasty in the United States from 2005 to 2030. *J. Bone Jt. Surg. - Ser. A*

89, 780–785.

- Kutzner, I., Heinlein, B., Graichen, F., Bender, A., Rohlmann, A., Halder, A., Beier, A., Bergmann, G., 2010. Loading of the knee joint during activities of daily living measured in vivo in five subjects. *J. Biomech.* 43, 2164–2173.
- Lang, J.E., Mannava, S., Floyd, A.J., Goddard, M.S., Smith, B.P., Mofidi, A., Seyler, T.M., Jinnah, R.H., 2011. Robotic systems in orthopaedic surgery. *J Bone Jt. Surg Br* 93, 1296–1299.
- Leung, T., Vyas, D., 2014. Robotic Surgery: Applications. *Am. J. Robot. Surg.* 1, 1–64.
- LifeModeler Inc., 2019. LifeMOD Framework Manual.
- Liow, M.H.L., Goh, G.S.-H., Wong, M.K., Chin, P.L., Tay, D.K.-J., Yeo, S.-J., 2017. Robotic-assisted total knee arthroplasty may lead to improvement in quality-of-life measures: a 2-year follow-up of a prospective randomized trial. *Knee Surgery, Sport. Traumatol. Arthrosc.* 25, 2942–2951.
- Lonner, J.H., Kerr, G.J., 2012. Robotically assisted unicompartmental knee arthroplasty. *Oper. Tech. Orthop.* 22, 182–188.
- Lonner, J.H., Smith, J.R., Picard, F., Hamlin, B., Rowe, P.J., Riches, P.E., 2015. High degree of accuracy of a novel image-free handheld robot for unicondylar knee arthroplasty in a cadaveric study. *Clin. Orthop. Relat. Res.* 473, 206–212.
- Losina, E., Thornhill, T.S., Rome, B.N., Wright, J., Katz, J.N., 2012. The dramatic increase in total knee replacement utilization rates in the United States cannot be fully explained by growth in population size and the obesity epidemic. *J. Bone Jt. Surg. - Ser. A* 94, 201.
- Loudon, J.K., 2016. Biomechanics and pathomechanics of the patellofemoral joint. *Int. J. Sports Phys. Ther.* 11, 820–830.

- Lund, M.E., Andersen, M.S., de Zee, M., Rasmussen, J., 2015. Scaling of musculoskeletal models from static and dynamic trials. *Int. Biomech.* 2, 1–11.
- Maclean, F.M., 2016. Arthritis and other proliferative joint diseases. *Diagnostic Histopathol.*
- Maletsky, L.P., Hillberry, B.M., 2005. Simulating dynamic activities using a five-axis knee simulator. *J. Biomech. Eng.* 127, 123.
- Manal, K., Buchanan, T.S., 2012. Predictions of condylar contact during normal and medial thrust gait, in: *ASME 2012 Summer Bioengineering Conference*. ASME, Fajardo, Puerto Rico, pp. 197–198.
- Mancuso, C.A., Jout, J., Salvati, E.A., Sculco, T.P., 2009. Fulfillment of patients' expectations for total hip arthroplasty. *J. Bone Jt. Surg. - Ser. A* 91, 2073–2078.
- Manner, P.W., 2016. Knee Replacement Implants - OrthoInfo - AAOS [WWW Document]. OrthoInfo. URL <https://orthoinfo.aaos.org/en/treatment/knee-replacement-implants/> (accessed 3.23.19).
- Mannion, A.F., Kämpfen, S., Munzinger, U., Kramers-de Quervain, I., 2009. The role of patient expectations in predicting outcome after total knee arthroplasty. *Arthritis Res. Ther.* 11, R139.
- Marchand, R.C., Sodhi, N., Khlopa, A., Sultan, A.A., Harwin, S.F., Malkani, A.L., Mont, M.A., 2017. Patient satisfaction outcomes after robotic arm-assisted total knee arthroplasty: A short-term evaluation. *J. Knee Surg.* 30, 849–853.
- Marra, M.A., Strzelczak, M., Heesterbeek, P.J.C., Van De Groes, S., Janssen, D., Koopman, B.F.J.M., Wymenga, A.B., Verdonschot, N., 2017. The effect of tibial slope on the biomechanics of cruciate-retaining TKA: A musculoskeletal simulation study, in: *European Knee Society Conference 2017*.

- Merican, A.M., Ghosh, K.M., Iranpour, F., Deehan, D.J., Amis, A.A., 2011. The effect of femoral component rotation on the kinematics of the tibiofemoral and patellofemoral joints after total knee arthroplasty. *Knee Surgery, Sport. Traumatol. Arthrosc.* 19, 1479–1487.
- Merloz, P., Tonetti, J., Pittet, L., Coulomb, M., Lavallée, S., Troccaz, J., Cinquin, P., Sautot, P., 1998. Computer-assisted spine surgery. *Comput. Aided Surg.* 337, 86–96.
- Moro-Oka, T.A., Muenchinger, M., Canciani, J.P., Banks, S.A., 2007. Comparing in vivo kinematics of anterior cruciate-retaining and posterior cruciate-retaining total knee arthroplasty. *Knee Surgery, Sport. Traumatol. Arthrosc.* 15, 93–99.
- Na, S.M., Cho, K., Seon, J.-K., Song, E.K., 2019. Robotic vs. Conventional Primary Total Knee Arthroplasty: Clinical and Radiological Long-Term Results with a Minimum Follow Up of 10 Years, in: American Academy of Orthopedic Surgeons Annual Meeting. AAOS, Las Vegas, NV.
- Navacchia, A., Hume, D.R., Rullkoetter, P.J., Shelburne, K.B., 2019. A computationally efficient strategy to estimate muscle forces in a finite element musculoskeletal model of the lower limb. *J. Biomech.* 84, 94–102.
- Navacchia, A., Rullkoetter, P.J., Schütz, P., List, R., Fitzpatrick, C.K., Shelburne, K.B., 2016. Subject-specific modeling of muscle force and knee contact in total knee arthroplasty. *J. Orthop. Res.* 34, 1576–1587.
- Netravali, N.A., Shen, F., Park, Y., Bargar, W.L., 2013. A perspective on robotic assistance for knee arthroplasty. *Adv. Orthop.* 2013, 1–9.
- Nolte, L.P., Visarius, H., Arm, E., Langlotz, F., Schwarzenbach, O., Zamorano, L., 1995. Computer-aided fixation of spinal implants. *Comput. Aided Surg.* 1, 88–93.
- Parcells, B., 2017. TKA bone cuts [WWW Document]. Hip Knee B. URL

- <https://hipandkneebook.com/tja-publication-blog/2017/3/14/tka-bone-cuts> (accessed 5.1.19).
- Pastides, P., Nathwani, D., 2017. The role of newer technologies in knee arthroplasty. *Orthop. Trauma* 31, 47–52.
- Peersman, G., Laskin, R., Davis, J., Peterson, M.G.E., Richart, T., 2006. Prolonged operative time correlates with increased infection rate after total knee arthroplasty. *HSS J.* 2, 70–72.
- Pugely, A.J., Martin, C.T., Gao, Y., Schweizer, M.L., Callaghan, J.J., 2015. The incidence of and risk factors for 30-day surgical site infections following primary and revision total joint arthroplasty. *J. Arthroplasty* 30, 47–50.
- Pulido, L., Ghanem, E., Joshi, A., Purtill, J.J., Parvizi, J., 2008. Periprosthetic joint infection: the incidence, timing, and predisposing factors. *Clin. Orthop. Relat. Res.* 466, 1710–1715.
- Rasnack, R., Standifird, T., Reinbolt, J.A., Cates, H.E., Zhang, S., 2016. Knee joint loads and surrounding muscle forces during stair ascent in patients with total knee replacement. *PLoS One* 11, e0156282.
- Rauck, R.C., Blevins, J.L., Cross, M.B., 2018. Component placement accuracy in unicompartmental knee arthroplasty is improved with robotic-assisted surgery: will it have an effect on outcomes? *HSS J.* 14, 211–213.
- Roche, M.W., 2015. Robotic and sensor-assisted technologies in knee arthroplasty. *Oper. Tech. Orthop.* 25, 127–149.
- Roos, E.M., Roos, H.P., Lohmander, L.S., Ekdahl, C., Beynnon, B.D., 1998. Knee injury and osteoarthritis outcome score (KOOS)--development of a self-administered outcome measure. *J. Orthop. Sports Phys. Ther.* 28, 88–96.
- Saikko, V., 2006. Effect of contact pressure on wear and friction of ultra-high molecular weight

- polyethylene in multidirectional sliding. *Proc. Inst. Mech. Eng. Part H J. Eng. Med.* 220, 723–731.
- Schellenberg, F., List, R., Schütz, P., Kutzner, I., Schwachmeyer, V., Taylor, W.R., Lorenzetti, S., 2016. Musculoskeletal squat simulation evaluation by means of an instrumented total knee arthroplasty, in: 34th International Conference on Biomechanics in Sports, Tsukuba, Japan. pp. 347–350.
- Schellenberg, F., Taylor, W.R., Trepczynski, A., List, R., Kutzner, I., Schütz, P., Duda, G.N., Lorenzetti, S., 2018. Evaluation of the accuracy of musculoskeletal simulation during squats by means of instrumented knee prostheses. *Med. Eng. Phys.* 61, 95–99.
- Schiraldi, M., Bonzanini, G., Chirillo, D., de Tullio, V., 2016. Mechanical and kinematic alignment in total knee arthroplasty. *Ann. Transl. Med.* 4, 130–130.
- Schwartz, A.J., Ravi, B., Kransdorf, M.J., Clarke, H.D., 2017. Does tibial slope affect perception of coronal alignment on a standing anteroposterior radiograph? *J. Arthroplasty* 32, 2285–2288.
- Scott, R.D., 2003. Unicondylar arthroplasty: redefining itself. *Orthopedics* 26, 951–952.
- Scuderi, G.R., Bourne, R.B., Noble, P.C., Benjamin, J.B., Lonner, J.H., Scott, W.N., 2012. The new Knee Society Knee Scoring System. *Clin. Orthop. Relat. Res.* 470, 3–19.
- Shankar, S., Tetreault, M.W., Jegier, B.J., Andersson, G.B., Della Valle, C.J., 2016. A cost comparison of unicompartmental and total knee arthroplasty. *Knee* 23, 1016–1019.
- Shervin, D., Pratt, K., Healey, T., Nguyen, S., Mihalko, W.M., El-Othmani, M.M., Saleh, K.J., 2015. Anterior knee pain following primary total knee arthroplasty. *World J. Orthop.* 6, 795.
- Sheth, N.P., Husain, A., Nelson, C.L., 2017. Surgical techniques for total knee arthroplasty. *J.*

- Am. Acad. Orthop. Surg. 25, 499–508.
- Shu, L., Yamamoto, K., Yao, J., Saraswat, P., Liu, Y., Mitsuishi, M., Sugita, N., 2018. A subject-specific finite element musculoskeletal framework for mechanics analysis of a total knee replacement. *J. Biomech.* 77, 146–154.
- Siebert, W., Mai, S., Kober, R., Heeckt, P.F., 2002. Technique and first clinical results of robot-assisted total knee replacement. *Knee* 9, 173–180.
- Sinusas, K., 2012. Osteoarthritis: diagnosis and treatment. *Am. Fam. Physician* 85, 49–56.
- Smith & Nephew, 2018. NAVIO Surgical System Surgical Technique for Total Knee Arthroplasty.
- Smith and Nephew, 2015. LifeModeler KneeSIM Lab Kansas Knee Simulator Study, Smith and Nephew Internal Report.
- Smith, C.R., Vignos, M.F., Lenhart, R.L., Kaiser, J., Thelen, D.G., 2016. The influence of component alignment and ligament properties on tibiofemoral contact forces in total knee replacement. *J. Biomech. Eng.* 138, 021017.
- Sodhi, N., Khlopas, A., Piuze, N.S., Sultan, A.A., Marchand, R.C., Malkani, A.L., Mont, M.A., 2018. The learning curve associated with robotic total knee arthroplasty. *J. Knee Surg.*
- Song, E.-K., Seon, J.-K., 2018. Robotic Total Knee Arthroplasty, in: *Computer Assisted Orthopaedic Surgery for Hip and Knee*. Springer Singapore, Singapore, pp. 27–40.
- Song, E.-K., Seon, J.-K., Park, S.-J., Jung, W. Bin, Park, H.-W., Lee, G.W., 2011. Simultaneous bilateral total knee arthroplasty with robotic and conventional techniques: a prospective, randomized study. *Knee Surgery, Sport. Traumatol. Arthrosc.* 19, 1069–1076.
- Song, Z., Borgwardt, L., Høiby, N., Wu, H., Sørensen, T.S., Borgwardt, A., 2013. Prosthesis infections after orthopedic joint replacement: the possible role of bacterial biofilms. *Orthop.*

- Rev. (Pavia). 5, 65–71.
- Stewart, A.L., Hays, R.D., Ware, J.E., 1988. The MOS short-form general health survey. Reliability and validity in a patient population. *Med. Care* 26, 724–35.
- Stiehl, J.B., Komistek, R.D., Cloutier, J.-M., Dennis, D.A., 2000. The cruciate ligaments in total knee arthroplasty. *J. Arthroplasty* 15, 545–550.
- Stryker, 2017. Surgical Guide Mako TKA Surgical Guide.
- Taylor, W.R., Schütz, P., Bergmann, G., List, R., Postolka, B., Hitz, M., Dymke, J., Damm, P., Duda, G., Gerber, H., Schwachmeyer, V., Hosseini Nasab, S.H., Trepczynski, A., Kutzner, I., 2017. A comprehensive assessment of the musculoskeletal system: The CAMS-Knee data set. *J. Biomech.* 65, 32–39.
- Thompson, J.C., Netter, F.H., 2010. *Netter’s Concise Orthopaedic Anatomy*, Netter clinical science. Elsevier.
- Torrão, J.N., dos Santos, M.P.S., Ferreira, J.A., 2015. Instrumented knee joint implants: innovations and promising concepts. *Expert Rev. Med. Devices* 12, 571–584.
- Towheed, T., Maxwell, L., Anastassiades, T.P., Shea, B., Houpt, J.B., Welch, V., Hochberg, M.C., Wells, G.A., 2009. Glucosamine therapy for treating osteoarthritis. *Cochrane Database Syst. Rev.*
- Tsai, T., Liow, M.H.L., Li, G., Arauz, P., Peng, Y., Klemm, C., Kwon, Y., 2019. Bi-cruciate retaining total knee arthroplasty does not restore native tibiofemoral articular contact kinematics during gait. *J. Orthop. Res.* jor.24333.
- van der List, J.P., Chawla, H., Joskowicz, L., Pearle, A.D., 2016. Current state of computer navigation and robotics in unicompartmental and total knee arthroplasty: a systematic review with meta-analysis. *Knee Surgery, Sport. Traumatol. Arthrosc.* 24, 3482–3495.

- Varacallo, M., Johanson, N.A., 2018. Total knee arthroplasty (TKA) techniques, StatPearls. StatPearls Publishing, Treasure Island, FL.
- Varadarajan, K.M., Harry, R.E., Johnson, T., Li, G., 2009. Can in vitro systems capture the characteristic differences between the flexion–extension kinematics of the healthy and TKA knee? *Med. Eng. Phys.* 31, 899–906.
- Verstraete, M.A., Victor, J., 2015. Possibilities and limitations of novel in-vitro knee simulator. *J. Biomech.* 48, 3377–3382.
- Wada, K., Mikami, H., Hamada, D., Yamazaki, T., Tomita, T., Sairyō, K., 2017. Can intraoperative kinematic analysis predict postoperative kinematics following total knee arthroplasty? A preliminary. *J. Med. Investig.* 65, 21–26.
- Weber, P., Crispin, A., Schmidutz, F., Utzschneider, S., Pietschmann, M.F., Jansson, V., Müller, P.E., 2013. Improved accuracy in computer-assisted unicondylar knee arthroplasty: A meta-analysis. *Knee Surgery, Sport. Traumatol. Arthrosc.* 21, 2453–2461.
- Werner, B.C., Evans, C.L., Carothers, J.T., Browne, J.A., 2015. Primary total knee arthroplasty in super-obese Patients: dramatically higher postoperative complication rates even compared to revision surgery. *J. Arthroplasty* 30, 849–853.
- West, J.A., Scudday, T., Anderson, S., Amin, N.H., 2019. Clinical outcomes and patient satisfaction after total knee arthroplasty: a follow-up of the first 50 cases by a single surgeon. *J. Int. Med. Res.* 47, 1667–1676.
- Wunschel, M., Mller, O., Lo, J., Obloh, C., Wlker, N., 2010. The anterior cruciate ligament provides resistance to externally applied anterior tibial force but not to internal rotational torque during simulated weight-bearing flexion. *Arthrosc. - J. Arthrosc. Relat. Surg.* 26, 1520–1527.

- Yang, H.Y., Seon, J.K., Shin, Y.J., Lim, H.A., Song, E.K., 2017. Robotic total knee arthroplasty with a cruciate-retaining implant: a 10-year follow-up study. *Clin. Orthop. Surg.* 9, 169.
- Zavatsky, A.B., 1997. A kinematic-freedom analysis of a flexed-knee-stance testing rig. *J. Biomech.* 30, 277–80.
- Zhang, Y., Jordan, J.M., 2010. Epidemiology of osteoarthritis. *Clin. Geriatr. Med.* 26, 355–69.
- Zheng, G., Nolte, L.P., 2015. Computer-assisted orthopedic surgery: current state and future perspective. *Front. Surg.* 2, 66.
- Zhou, Z., Wu, B., Duan, J., Zhang, X., Zhang, N., Liang, Z., 2017. Optical surgical instrument tracking system based on the principle of stereo vision. *J. Biomed. Opt.* 22, 065005.
- Zimmer, 2014. Zimmer PERSONA surgical instruments for total knee arthroplasty [WWW Document]. *Appl. Des. Mag.* URL <https://www.appliancedesign.com/articles/94067-zimmer-persona-surgical-instruments-for-total-knee-arthroplasty> (accessed 3.23.19).

Appendix A – DOE Experimental Design

The following tables highlight the experimental design employed for the various DOE studies conducted during this investigation. The experimental designs are presented in the following order:

- 1) Fractional factorial experiment to optimize implant position and soft tissue properties for KneeSIM Lab baseline model. Resulting model is optimized KneeSIM Lab.
- 2) Fractional factorial experiment to tune muscle gain parameters for VCTK baseline model. Resulting model is a refined version of the VCTK baseline.
- 3) Fractional factorial experiment (Latin Hypercube design) to optimize implant position and soft tissue properties for VCTK baseline model. Resulting model is optimized VCTK.
- 4) Fractional factorial experimental design to develop Knee Performance equations used for both optimized KneeSIM Lab and optimized VCTK

Table 14: Fractional factorial design used to develop optimized KneeSIM Lab model.

Trial	MCL_insertion_anterior	MCL_insertion_supe	LCL_insertion_supe	LCL_insertion_anterior	MCL_stiffness	LCL_stiffness	PatTen_origin_anterior	PatTen_origin_medial	PatTen_origin_medial	QuadTen_medial_r	QuadTen_anterior_r	PatTen_insertion_anterior	PatTen_insertion_medial	PatTen_insertion_superior	Femoral_IE	Tibial_IE
Trial 1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
Trial 2	-1	-1	-1	-1	1	1	-1	-1	-1	-1	-1	1	1	-1	1	1
Trial 3	-1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	1	-1	1
Trial 4	-1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	1	1	-1
Trial 5	-1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	1	-1	1
Trial 6	-1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	1	1	-1
Trial 7	-1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	-1	1	-1
Trial 8	-1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	-1	-1	1
Trial 9	-1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	1	1	-1
Trial 10	-1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	-1	-1	1
Trial 11	-1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	1	-1	1
Trial 12	-1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	-1	1	-1
Trial 13	-1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	1	-1	1
Trial 14	-1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	-1	1	-1
Trial 15	-1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	1	-1	-1
Trial 16	-1	-1	-1	-1	1	1	-1	-1	-1	-1	-1	-1	-1	1	1	1
Trial 17	-1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	-1	-1	-1
Trial 18	1	-1	-1	-1	1	1	-1	-1	1	1	-1	-1	-1	1	1	1
Trial 19	1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	-1	1	1
Trial 20	1	-1	-1	-1	1	1	-1	-1	-1	-1	-1	1	1	1	-1	-1
Trial 21	1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	-1	-1	-1	-1	-1
Trial 22	1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	1	1	-1
Trial 23	1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	-1	-1	-1	-1	-1
Trial 24	1	-1	-1	-1	1	1	-1	-1	-1	-1	-1	1	1	-1	-1	1
Trial 25	1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	1	1	-1
Trial 26	1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	1	-1	1
Trial 27	1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	-1	-1	-1	-1	1
Trial 28	1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	-1	-1	-1
Trial 29	1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	-1	1	1
Trial 30	1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	1	-1	-1
Trial 31	1	-1	-1	-1	-1	-1	1	1	-1	-1	-1	1	1	-1	-1	-1
Trial 32	1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	-1	1	1	1

Table 15: Fractional factorial design used to tune muscle gains for optimized VCTK model.

Trial	Biceps_femoris_gain1	Biceps_femoris_gain2	Gastrocnemius_gain1	Gastrocnemius_gain2	Semitendinosus_gain	Gluteus_Maximus_gain1
Trial 1	-1	-1	-1	-1	-1	-1
Trial 2	-1	-1	-1	-1	1	1
Trial 3	-1	-1	-1	1	-1	1
Trial 4	-1	-1	-1	1	1	-1
Trial 5	-1	-1	1	-1	-1	1
Trial 6	-1	-1	1	-1	1	-1
Trial 7	-1	-1	1	1	-1	-1
Trial 8	-1	-1	1	1	1	1
Trial 9	-1	1	-1	-1	-1	1
Trial 10	-1	1	-1	-1	1	-1
Trial 11	-1	1	-1	1	-1	-1
Trial 12	-1	1	-1	1	1	1
Trial 13	-1	1	1	-1	-1	-1
Trial 14	-1	1	1	-1	1	1
Trial 15	-1	1	1	1	-1	1
Trial 16	-1	1	1	1	1	-1
Trial 17	1	-1	-1	-1	-1	1
Trial 18	1	-1	-1	-1	1	-1
Trial 19	1	-1	-1	1	-1	-1
Trial 20	1	-1	-1	1	1	1
Trial 21	1	-1	1	-1	-1	-1
Trial 22	1	-1	1	-1	1	1
Trial 23	1	-1	1	1	-1	1
Trial 24	1	-1	1	1	1	-1
Trial 25	1	1	-1	-1	-1	-1
Trial 26	1	1	-1	-1	1	1
Trial 27	1	1	-1	1	-1	1
Trial 28	1	1	-1	1	1	-1
Trial 29	1	1	1	-1	-1	1
Trial 30	1	1	1	-1	1	-1
Trial 31	1	1	1	1	-1	-1
Trial 32	1	1	1	1	1	1

Table 16: Fractional factorial design (Latin Hypercube) used to create optimized VCTK model.

Trial	FemoralML	FemoralAP	FemoralSI	TibialML	TibialAP	TibialSI	FemoralVV	FemoralIE	TibialSlope	TibialVV	TibialIE	MCL_Strain	LCL_Strain	TibFemCapsule	PatFemCapsule	
Trial 1	-11	5	-10	-10	-9	-13	-8	-11	9	-1	-6	-6	1	2	-7	6
Trial 2	4	-7	8	12	-5	1	-9	-12	7	2	2	-13	-4	-8	-9	15
Trial 3	-15	-9	-11	-9	-13	13	7	-3	11	7	7	-11	-11	-12	10	-14
Trial 4	-1	3	-12	-15	10	-2	9	1	2	13	8	8	-1	13	4	5
Trial 5	-6	7	-6	-1	-14	4	14	-13	10	-5	-5	-1	3	10	3	4
Trial 6	-12	-12	12	-2	11	5	-3	11	-9	5	10	-15	11	11	-13	10
Trial 7	-8	-5	-5	-11	1	12	-16	-10	-16	15	15	-9	16	6	-11	16
Trial 8	2	14	-3	-16	8	7	2	13	13	8	8	-5	6	-7	-15	-16
Trial 9	8	13	3	11	2	6	-15	-6	12	-9	-9	-7	-3	-2	11	-5
Trial 10	-10	-14	-14	14	-12	-11	13	14	-4	-13	-2	-2	10	16	-4	-10
Trial 11	13	-16	-16	-12	-4	3	6	-1	-14	1	-4	-6	-6	-6	6	9
Trial 12	10	-3	-8	-6	-10	-1	-11	-8	-13	14	-15	-15	9	-15	-2	13
Trial 13	3	-6	7	-4	14	-5	-4	-4	-15	-11	16	12	14	14	-3	-8
Trial 14	5	-11	-15	16	-7	-10	-7	10	4	-2	-2	3	5	-14	15	7
Trial 15	-4	2	14	8	13	-3	-1	2	16	10	-10	5	7	5	-8	14
Trial 16	15	9	9	1	-16	10	-6	5	15	15	10	-3	-5	-1	-16	-4
Trial 17	12	11	11	-3	-2	-14	-10	3	14	9	13	13	-13	15	13	12
Trial 18	-5	-15	-2	3	-6	15	15	4	6	4	-11	-11	-14	7	5	-12
Trial 19	11	4	2	4	-8	-9	16	12	5	16	1	1	15	-10	8	-13
Trial 20	-3	6	4	6	12	9	-2	7	8	-14	11	11	8	-9	9	-6
Trial 21	-2	-2	-4	10	7	-7	8	16	-2	3	3	-8	-8	3	1	1
Trial 22	-7	-10	-1	-8	6	-4	12	-5	-7	-12	-10	-10	-10	1	-12	-7
Trial 23	16	10	6	-14	4	11	-5	-2	-6	-4	12	11	11	-11	14	3
Trial 24	-9	15	1	-5	-1	-12	3	6	1	-3	4	4	-9	-4	-10	-1
Trial 25	-16	1	16	-7	-3	-16	4	-14	-12	12	6	6	-7	4	-1	-11
Trial 26	7	-1	-13	15	5	-15	-13	-15	-8	6	15	2	2	-16	12	-3
Trial 27	9	12	5	5	9	14	-12	-16	-10	-6	14	-2	12	-6	-2	-2
Trial 28	14	16	-7	2	15	-8	11	-9	-5	11	7	-16	8	8	2	2
Trial 29	1	-8	10	9	-15	2	-14	15	-11	-8	-14	-12	-12	9	16	8
Trial 30	6	-13	-9	-13	-11	-6	10	8	-1	-15	-12	-12	4	-5	7	-15
Trial 31	-13	8	13	7	3	16	1	-7	-3	-16	-16	9	13	-3	-14	-9
Trial 32	-14	-4	15	13	16	8	5	9	3	-7	-7	-16	14	-13	-5	11

Table 17: Fractional factorial experimental design used to create Knee Performance Equations for KneesIM Lab and VCTK

Trial	FemoralML	FemoralAP	FemoralSI	TibialML	TibialAP	TibialSI	FemoralV	FemoralIE	TibialSIop	TibialVV	TibialIE
Trial 1	-1	-1	-1	-1	-1	-1	-1	-1	1	1	1
Trial 2	-1	-1	-1	1	-1	-1	1	1	-1	-1	1
Trial 3	-1	-1	1	-1	1	-1	-1	1	1	-1	-1
Trial 4	-1	-1	1	1	1	1	-1	-1	1	1	-1
Trial 5	-1	1	-1	-1	-1	1	-1	1	-1	-1	1
Trial 6	-1	1	-1	1	1	-1	1	-1	1	-1	1
Trial 7	-1	1	1	-1	-1	1	1	-1	1	-1	-1
Trial 8	-1	1	1	1	-1	-1	-1	1	-1	-1	-1
Trial 9	1	-1	-1	-1	1	1	1	-1	-1	-1	-1
Trial 10	1	-1	-1	1	1	-1	-1	1	1	-1	-1
Trial 11	1	-1	1	-1	-1	1	-1	1	1	-1	1
Trial 12	1	-1	1	1	-1	-1	1	-1	-1	-1	1
Trial 13	1	1	-1	-1	-1	-1	1	1	1	1	-1
Trial 14	1	1	-1	1	-1	1	-1	-1	-1	1	-1
Trial 15	1	1	1	-1	1	-1	-1	-1	-1	1	1
Trial 16	1	1	1	1	1	1	1	1	1	1	1

Appendix B – Python Scripts

The attached text is a sample of the custom Python scripts developed to import and scale skeletal, muscular, ligament, and implant model from SimTK data to LifeModeler VCTK.

Additionally, the scripts import motion capture data and force plate data from the SimTK dataset.

```
from lm.modeling      import base, adams, anatomy, geometry
from lm.lib           import path
from lm.io.csvReader import CsvReader
from lm.modeling.multiRun.data import ComparativeResponse

model = ROOT.World
dude  = model.Isaac
ExecuteTask('VCTK_ANTHROPOMETRIC',
    body    = dude,
    gender  = 0, # gender needs a key, 0 vs 1??
    height  = 1820,
    weight  = 75,
    age     = 70*12,
    Upper_Neck    = 64.50,
    Lower_Neck    = 58.67,
    Thoracic      = 44.95,
    Lumbar        = 41.90,
    Right_Shoulder = 56.09,
    Right_Shoulder_From_Body_Center = 7.70,
    Right_Elbow   = 45.55,
    Right_Wrist   = 34.64,
    Right_Hip     = 39.2,
    Right_Hip_From_Body_Center = 3.28,
    Right_Knee    = 20.6,
    Right_Ankle   = 4.13,
    Right_Metatarsal = 0.00,
    Left_Shoulder = 56.09,
    Left_Shoulder_From_Body_Center = 7.70,
    Left_Elbow    = 45.55,
    Left_Wrist    = 34.64,
    Left_Hip     = 39.2,
    Left_Hip_From_Body_Center = 3.28,
    Left_Knee    = 20.6,
    Left_Ankle   = 4.13,
    Left_Metatarsal = 0.00
)

# File location for formatted gait trials
# Overground Gait Trials\Video Motion Data\Header_PS_ngait_og_ssl_trajectories.csv
dataloc = r'C:\Users\farleyd\Documents\SimNav\grandchallenge_data\PS\Synchronized Motion Data\Calibration Trials\Video Motion Data'
mocapfile = 'HeaderPS_llegstandl_trajectories.csv'
geoloc = 'C:/Users/farleyd/DOCUME~1/SimNav/GRANDC~1/PS/GEOMET~1/'
grffile = 'PS_llegstandl_grf.csv'

if not path.isdir(dataloc):
    # swap the path for LD machine.
    geoloc = dataloc = r"C:\Users\lduxbury\Desktop\danny_simtk"

# Run the VCTK Mocap/GRF task with type = Unknown first.
ExecuteTask('Plugins.VirtualClinicalTrial_KneeSIM.BDL Dual X-ray Tools.Read CSV of Mocap and GRF',
    body          = dude,
    csvFile       = path.join(dataloc,mocapfile),
    grfDataStart  = 0,
    grfDataEnd    = 0,
    mocapSampleRate = 120,
    mocapDataStart = 4,
    mocapDataEnd  = 341,
    maType        = 'Unknown',
    interpolate    = True,
)
agentSet = dude.MotionAgents.Unknown
```

```

# This is the orientation for the static trials
#dude.MotionAgents.location = UNITS('mm', 596, -972, 308)
#dude.MotionAgents.orientation = UNITS('degree', 180, 90, 180)

# This is the location for gait trials
#dude.MotionAgents.location = UNITS('mm', 114, -1000, 840)
#dude.MotionAgents.orientation = UNITS('degree', 180, 90, 270)

# Then run this to update the Unknown set to this customized set.
My_New_Plugin_Set = (
  # SimTK marker system.
  # Segment      Name      Index  Location      Annotation
  ("Upper_Torso", "Sternum", 0, ( 95.0, 0.0, 100.0), "Sternal notch"),
  ("Neck", "Neck", 1, (-74.0, 0.0, 9.0), "Spinous process of C7"),
  ("Right_Scapula", "R_Shoulder", 2, (-8.0, -59.0, 60.0), "Right acromiom"),
  ("Left_Scapula", "L_Shoulder", 3, ( 8.0, 59.0, 60.0), "Left acromiom"),
  ("Upper_Torso", "Xiphoid", 4, ( 95.0, 0.0, -40.0), "Xiphoid process"),
  ("Upper_Torso", "Thoracic", 5, (-100.0, 0.0, -71.0), "Upper back at T10 level"),
  ("Lower_Torso", "R_Asis", 6, ( 44.0, -139.0, 43.0), "Right ASIS protrusion"),
  ("Lower_Torso", "L_Asis", 7, ( 44.0, 139.0, 43.0), "Left ASIS protrusion"),
  ("Lower_Torso", "R_Psis", 8, (-83.0, -27.0, 81.0), "Right PSIS protrusion"),
  ("Lower_Torso", "L_Psis", 9, (-83.0, 27.0, 81.0), "Left PSIS protrusion"),
  ("Central_Torso", "Lumbar", 10, (-74.0, 0.0, -2.0), "Small of the back L4-L5
level"),

  ("Right_Upper_Arm", "R_Elbow", 11, ( 18, -56.0, -149.0), "Lateral epicondyle of the
right elbow"),
  ("Right_Upper_Arm", "R_ElbowMedial", 12, (-14, 25, -162), "Medial epicondyle of the right
elbow"),
  ("Right_Scapula", "R_ShoulderAnterior", 13, ( 37, -59.0, 10.0), "Right shoulder"),
  ("Right_Scapula", "R_ShoulderPosterior", 14, (-52.0, -59.0, 14.0), "Right shoulder"),
  ("Right_Lower_Arm", "R_Wrist", 15, (13.0, -23.0, -138.0), "Dorsal side of right hand
center of wrist"),
  ("Right_Lower_Arm", "R_Radius", 16, (33.0, -21.0, -141.0), "Dorsal side of right hand
on radius"),
  ("Right_Lower_Arm", "R_Ulna", 17, (-4.0, -23.0, -141.0), "Dorsal side of right hand
on ulna"),

  ("Left_Upper_Arm", "L_Elbow", 18, ( 18, 56.0, -149.0), "Lateral epicondyle of the
left elbow"),
  ("Left_Upper_Arm", "L_ElbowMedial", 19, (-14, -25, -162), "Medial epicondyle of the left
elbow"),
  ("Left_Scapula", "L_ShoulderAnterior", 20, ( 37, 59.0, 10.0), "Left Shoulder"),
  ("Left_Scapula", "L_ShoulderPosterior", 21, (-52.0, 59.0, 14.0), "Left Shoulder"),
  ("Left_Lower_Arm", "L_Wrist", 22, ( 13.0, 23.0, -138.0), "Dorsal side of the left
hand center of wrist"),
  ("Left_Lower_Arm", "L_Ulna", 23, (-4.0, 23.0, -141.0), "Dorsal side of the left
hand on ulna"),
  ("Left_Lower_Arm", "L_Radius", 24, ( 33.0, 21.0, -141.0), "Dorsal side of the left
hand on radius"),

  ("Left_Upper_Leg", "L_Thigh_Superior", 25, (-20, 95, 100), "Mid-anterior surface of
thigh"),
  ("Left_Upper_Leg", "L_Thigh_Inferior", 26, ( 1.0, 65.0, -130.0), "Disal anterior surface of
the thigh"),
  ("Left_Upper_Leg", "L_Thigh_Lateral", 27, ( 48.0, 14.0, -30.0), "Lateral aspect of thigh
midway between superior and inferior markers"),

  ("Left_Upper_Leg", "L_Knee_Lateral", 28, ( 0.0, 59.0, -215.0), "Lateral epicondyle of left
femur"),
  ("Left_Upper_Leg", "L_Knee_Medial", 29, ( 0, -65.0, -215.0), "Medial epicondyle of left
femur"),
  ("Left_Patella", "L_Patella", 30, ( 43, 3, -2.0), "Center of left patella"),

  ("Left_Lower_Leg", "L_Shank_Superior", 31, ( 4.0, 62.0, 10.0), "Superior anterior bony
surface of left tibia"),
  ("Left_Lower_Leg", "L_Shank_Inferior", 32, (-12.0, 46.0, -195.0), "Inferior anterior
surface of the left tibia"),
  ("Left_Lower_Leg", "L_Shank_Lateral", 33, ( 28.0, 3.0, -98.0), "Lateral aspect of shank
midway between superior and inferior markers"),

  ("Left_Lower_Leg", "L_Ankle_Lateral", 34, ( 3, 42.0, -248.0), "Lateral malleolus of left
ankle"),
  ("Left_Lower_Leg", "L_Ankle_Medial", 35, ( 3.0, -46.0, -234.0), "Medial malleolus of left
ankle"),

  ("Right_Upper_Leg", "R_Thigh_Superior", 36, (-20, -95, 100), "Mid-anterior surface of
thigh"),
  ("Right_Upper_Leg", "R_Thigh_Inferior", 37, ( 1.0, -65.0, -130.0), "Disal anterior surface
of the thigh"),

```

```

("Right_Upper_Leg", "R_Thigh_Lateral", 38, ( 48, -14.0, -30.0), "Lateral aspect of thigh
midway between superior and inferior markers"),
("Right_Upper_Leg", "R_Knee_Lateral", 39, ( 0.0, -59.0, -215.0), "Lateral epicondyle of the
right femur"),
("Right_Upper_Leg", "R_Knee_Medial", 40, ( 0, 65.0, -215.0), "Medial epicondyle of the right
femur"),
("Right_Lower_Leg", "R_Patella", 41, ( 4.0, 4.0, 194.0), "Center of the right
patella"),

("Right_Lower_Leg", "R_Shank_Superior", 42, ( 4.0, -62.0, 10.0), "Superior anterior bony
surface of right tibia"),
("Right_Lower_Leg", "R_Shank_Inferior", 43, ( -12.0, -46.0, -195.0), "Inferior anterior
surface of the right tibia"),
("Right_Lower_Leg", "R_Shank_Lateral", 44, ( 28.0, -3.0, -98.0), "Lateral aspect of shank
midway between superior and inferior markers"),

("Right_Lower_Leg", "R_Ankle_Lateral", 45, ( 3, -42.0, -248.0), "Lateral malleolus of the
right ankle"),
("Right_Lower_Leg", "R_Ankle_Medial", 46, ( 3.0, 46.0, -234.0), "Medial malleolus of the
right ankle"),

("Right_Foot", "R_Heel", 47, ( 30.0, 4.0, -164.0), "Back of shoe toe
height"),
("Right_Foot", "R_Toe", 48, ( 23.0, -12.0, 156.0), "Front edge shoetop above
second toe"),
("Right_Foot", "R_Midfoot_Medial", 49, (68.0, 26.0, -62.0), "?"),
("Right_Foot", "R_Midfoot_Lateral", 50, ( 60.0, -69.0, 21.0), "?"),
("Right_Foot", "R_Hindfoot", 51, ( 3.0, -41.0, -157.0), "?"),
("Right_Foot", "R_Midfoot_Superior", 52, ( -8.0, -13.0, 0.0), "Right foot on top of
shoe laces"),
("Right_Foot", "R_ToeMedial", 53, (57, 25.0, 99.0), "?"),
("Right_Foot", "R_ToeLateral", 54, ( 52.0, -66.0, 35.0), "?"),

("Left_Foot", "L_Heel", 55, ( 30.0, -4.0, -164.0), "?"),
("Left_Foot", "L_Toe", 56, ( 23.0, 12.0, 156.0), "?"),
("Left_Foot", "L_Midfoot_Medial", 57, ( 68.0, -26.0, -62.0), "?"),
("Left_Foot", "L_Midfoot_Lateral", 58, ( 60.0, 69.0, 21.0), "?"),
("Left_Foot", "L_Hindfoot", 59, ( 3.0, 41.0, -157.0), "?"),
("Left_Foot", "L_Midfoot_Superior", 60, ( -8.0, 13.0, 0.0), "?"),
("Left_Foot", "L_ToeMedial", 61, ( 57, -25.0, 99.0), "?"),
("Left_Foot", "L_ToeLateral", 62, ( 52.0, 66.0, 35.0), "?"),

("Left_Foot", "Oglobal", 63, ( 0.0, 0.0, 0.0), "?"),
("Lower_Torso", "Sacral", 64, ( -113.0, 0.0, 97.0), "?"),
("Left_Foot", "Xglobal", 65, ( 0.0, 0.0, 0.0), "?"),
("Left_Foot", "Yglobal", 66, ( 0.0, 0.0, 0.0), "?"),
("Left_Foot", "Zglobal", 67, ( 0.0, 0.0, 0.0), "?")
)

```

```

# Turn off O,X,Y,Z, and undefined markers
deadAgents = list((
  agentSet.Oglobal,
  agentSet.Xglobal,
  agentSet.Yglobal,
  agentSet.Zglobal,
  agentSet.L_Hindfoot,
  agentSet.R_Hindfoot,
  agentSet.L_Midfoot_Medial,
  agentSet.R_Midfoot_Medial,
  agentSet.L_ToeLateral,
  agentSet.R_ToeLateral,
  agentSet.L_ToeMedial,
  agentSet.R_ToeMedial,
))
for agent in deadAgents:
  agent.active = False
  agent.visible = False
dude.updateMotionAgents(agentSet, "My_New_Set", agentInfo=My_New_Plugin_Set)

agentSet.preAlignToBody(dude,
  rasi = agentSet.R_Asis,
  rpsi = agentSet.R_Psis,
  lasi = agentSet.L_Asis,
  lpsi = agentSet.L_Psis,
)

# ----- GRF -----
# The coordinate of the ground reaction data matches that of the mocap
ref = agentSet.getOne('grfRef', 'geometry.Landmark')
csvFile = path.join(dataloc, grffile)

```



```

with CsvReader(csvFile) as reader:
    header, data = reader.readTable(nLeftColumns=0, nDataColumns=22, nHeaders=1, dataStart=2, lastData=2641)

time = data[0]
# The time in the file starts at 4.325. The time in the Mocap file also starts at 4.325, but it is being
# ignored and starting at zero.
# so for now, just start this at zero too. If thats a problem, can change all times to start at 4.325
startsAt = time[0]
time      = [t-startsAt for t in time]
zeros    = [0]*len(time)

# The plates seem to be ordered 2-1-3 according to when the footstrikes happen
# Matching to the gait mocap (or examining the COP positions), this is R-L-R
for i,side in zip(("2", "1", "3"), ('Right', 'Left', 'Right')):
    dx = data[header.index('COPx%s'%i)]
    dy = data[header.index('COPy%s'%i)]
    dz = zeros
    fx = data[header.index('Fxs%s'%i)]
    fy = data[header.index('Fys%s'%i)]
    fz = data[header.index('Fzs%s'%i)]
    tx = zeros
    ty = zeros
    tz = data[header.index('Tzs%s'%i)]

    # Curiously, and unlike the other 3 mocap files I've just examined,
    # this one seems to present the forces that are applied to the foot, not measured by the plate.
    # So, flip all force components over to describe (equal and opposite) force component on plate.
    fx = [-1*r for r in fx]
    fy = [-1*r for r in fy]
    fz = [-1*r for r in fz]
    tx = [-1*r for r in tx]
    ty = [-1*r for r in ty]
    tz = [-1*r for r in tz]

    grx = agentSet.getOne("GRF_PLATE%s"%i, anatomy.GroundReactionForce,
        drivenPoint    = dude.getNamedSegment("%s_Foot"%side).getCM(),
        referencePoint = ref,
        resolvedToCOP  = True,
    )

    with base.UnitsSetTo("mmks", angle="radian"):
        grx.set_curves(data=(time, dx, dy, dz, fx, fy, fz, tx, ty, tz))

grxs = (agentSet.GRF_PLATE1, agentSet.GRF_PLATE2, agentSet.GRF_PLATE3)
fringe = base.FringeData(
    parent      = agentSet,
    name        = 'torqueScale',
    forceGraphicScale = 0.01, # Nmm are very small, and these arrows are too large
    range       = (0.0, 6000.0),
)
for g in grxs:
    g.torqueGraphic.fringeData = fringe
    g.active = False

#Optimized position

dude.Instrumentation.Femoral.Solid.location= UNITS('mm',-0.04796,0.12363,0.033586)
dude.Instrumentation.Tibial.Solid.location = UNITS('mm', 0.144455,0.176812,0.087841 )

dude.Instrumentation.Femoral.orientation = UNITS('degree', 0.125477757,0.583500219,0.974028252)
dude.Instrumentation.Tibial.orientation = UNITS('degree', 0.177616916, 0.077349302,0.968298674)

#Import new component set - USE THIS SET
dude.Instrumentation.Femoral.Solid.shell = path.join(geoloc, 'Femoral Component RO.stl')
dude.Instrumentation.Femoral.Solid.units = 'mm' # from None
dude.Instrumentation.Femoral.location = UNITS('mm', 6.352949164144653,-10.00948405677323,1.4216120680508593)

dude.Instrumentation.Patellar.Solid.shell = path.join(geoloc, 'Patellar Button.stl')
dude.Instrumentation.Patellar.Solid.units = 'mm' # from None
dude.Instrumentation.Patellar.location = UNITS('mm', -1.92584729184328,35.506,40.577103782493296)
dude.Instrumentation.Patellar.orientation = UNITS('degree', 0,180,180)

dude.Instrumentation.Tibial.Solid.shell = path.join(geoloc, 'Tibial Tray RO.stl')
dude.Instrumentation.Tibial.Solid.units = 'mm' # from None
dude.Instrumentation.Tibial.location = UNITS('mm', 6.351407245745511,-10.00000000000001,1.3731766445490567)

#Update Reference positions

```

```

dude.Instrumentation.References.FlexionFacetCenters.Medial.location = UNITS('mm',-17.42000000000005,-
8.46999999999997,23.60999999999964)
dude.Instrumentation.References.FlexionFacetCenters.Medial.orientation = UNITS('degree',270.0,90.0,0.0)

dude.Instrumentation.References.FlexionFacetCenters.Lateral.location = UNITS('mm',-17.42000000000005,-
8.46999999999997,23.60999999999964)

dude.Instrumentation.References.BanksPoint.location = UNITS('mm',0.0,6.060000000000085,28.6299999999999)
dude.Instrumentation.References.BanksPoint.orientation = UNITS('degree',270.0,90.0,0.0)

dude.Instrumentation.References.CondylarLowPoints.MedialLowPoint.boundingBoxCorner.location = UNITS('mm',
0.0,-38.000000000000036,-20.000000000000018)
dude.Instrumentation.References.CondylarLowPoints.LateralLowPoint.boundingBoxCorner.location = UNITS('mm',-
99.9999999999999,-39.00000000000001,-20.000000000000018)

#Move ligaments over

dude.Segments.Left_Upper_Leg.Attachments.Ligament.MCL.location = UNITS('mm', -4.595906870551257,-
51.120728901535735,-221.05807002640498)
dude.Segments.Left_Lower_Leg.Attachments.Ligament.MCL.location = UNITS('mm', -2.3686628065403506,-
40.833616816892494,136.79533832301837)
dude.Segments.Left_Lower_Leg.Attachments.Ligament.LCL.location = UNITS('mm', -
12.956788001588244,37.1095610051618,145.05529067403089)
dude.Segments.Left_Upper_Leg.Attachments.Ligament.LCL.location = UNITS('mm', -
11.009974700881036,37.26215289275141,-218.28032411563225)
dude.Segments.Left_Lower_Leg.Attachments.Ligament.PatellarTendon.location = UNITS('mm',
34.362865328076204,5.750121942373856,117.4094982363309)
dude.Segments.Left_Lower_Leg.Attachments.Ligament.PatellarTendon.orientation = UNITS('degree',
91.459890832141,8.280596438954415,268.4901080298243)

# Run settle run
patella_agent = agentSet.L_Patella
patella_agent.translationalStiffness = UNITS('newton/mm', 10.0) # from 10000.0 newton/mm
patella_agent.translationalDamping = UNITS('newton/mm/sec', 1.0) # from 10.0 newton/mm/sec

ExecuteTask('VCTK_SETTLING_RUN',
  deactivateMuscles = False,
  deactivateGravity = True,
  minimumSettlingTime = 0.5,
  translationalThreshold = 1,
  rotationalThreshold = 1,
  endTime = 5,
  timeSteps = 500,
  relocateMotionAgents = True,
  retainFinalVelocities = False,
)

#Optimized position

dude.Instrumentation.Femoral.Solid.location= UNITS('mm',-0.0544,0.13018,0.02428)
dude.Instrumentation.Tibial.Solid.location = UNITS('mm', -.055837,0.11434,-.00583 )

dude.Instrumentation.Femoral.Solid.orientation = UNITS('degree', 0.14755,0.12855,6.1032)
dude.Instrumentation.Tibial.Solid.orientation = UNITS('degree', 2.9822, 0.16096,3.2145)

# set and run inverse
trackGroup = dude.getOne('trackerAgents', base.Group)
attach = dude.Segments.Lower_Torso
tracker = ExecuteTask('Model.Motion Tracking.TrackerAgent'+_(' create...'),
  parent = trackGroup,
  name = attach.name,
  type = 'Recording',
  drivenPoint = attach.Attachments.CM,
  translationalStiffness = 10,
  translationalDamping = 1,
  rotationalStiffness = 1e5,
  rotationalDamping = 1e4,
  showMotionPath = True,
)
ExecuteTask('Analysis.Train',
  model = model,
  agentType = 'Driven',
  jointType = 'noChange',
  muscleType = 'noChange',
)
simSpec = model.SimSpec
simSpec.Simulation.end = 2.8
simSpec.Simulation.steps = 337
simSpec.Integrator.hmin = 1e-10

```

```

inverse = ExecuteTask('VirtualClinicalTrial_KneeSIM.Run tests.Dynamic Simulation',
    systemModel      = model,
    simulationModel  = simSpec,
    runName          = 'Inverse',
)

# set and run forward
ExecuteTask('VCTK_TRAIN',
    model           = model,
    run             = inverse,
    agentType      = 'Free',
    jointType      = 'Servo',
    muscleType     = 'noChange',
    trackType      = 'Driving',
)

for ma in agentSet.findAll(type=anatomy.MotionAgent):
    ma.visible = False

for muscle in dude.Muscles.Left_Leg.findAll(type=anatomy.Muscle):
    muscle.train(inverse, toType='closedSimple')

for muscle in dude.Muscles.Left_Leg.findAll(type=anatomy.Muscle):
    muscle.pGain = 1e6 #1e10
    muscle.iGain = 1e4 #1e11
    muscle.dGain = 1e5 #1e6
for j in dude.Joints.findAll(type='anatomy.JointControl'):
    j.pGain = 1e8 #1e3
    j.dGain = 1e6 #1e1
tracker.dy.type = "Driven"
# tracker.ax.type = "Free"
# tracker.ay.type = "Free"
# tracker.az.type = "Free"
for g in grxs:
    g.active = True

forward = ExecuteTask('VirtualClinicalTrial_KneeSIM.Run tests.Dynamic Simulation',
    runName      = 'Forward', )

ActivateTask('Plugins.VirtualClinicalTrial_KneeSIM.Results Exploration.Animation Controls')
view=ROOT.WorldAnimation
view.left()
view.fit()

```

Appendix C – Matlab Script

The attached text is a Matlab script that creates a graphical user interface showing the knee performance equation surface response maps. Interactive slider bars allow the user to position the femoral implant relative to a bone model in the anterior-posterior and medial-lateral directions. The corresponding ligament strains are projected the MCL Strain and LCL Strain surface response maps and the display of the implant and bone model fit is updated (Figure 35).

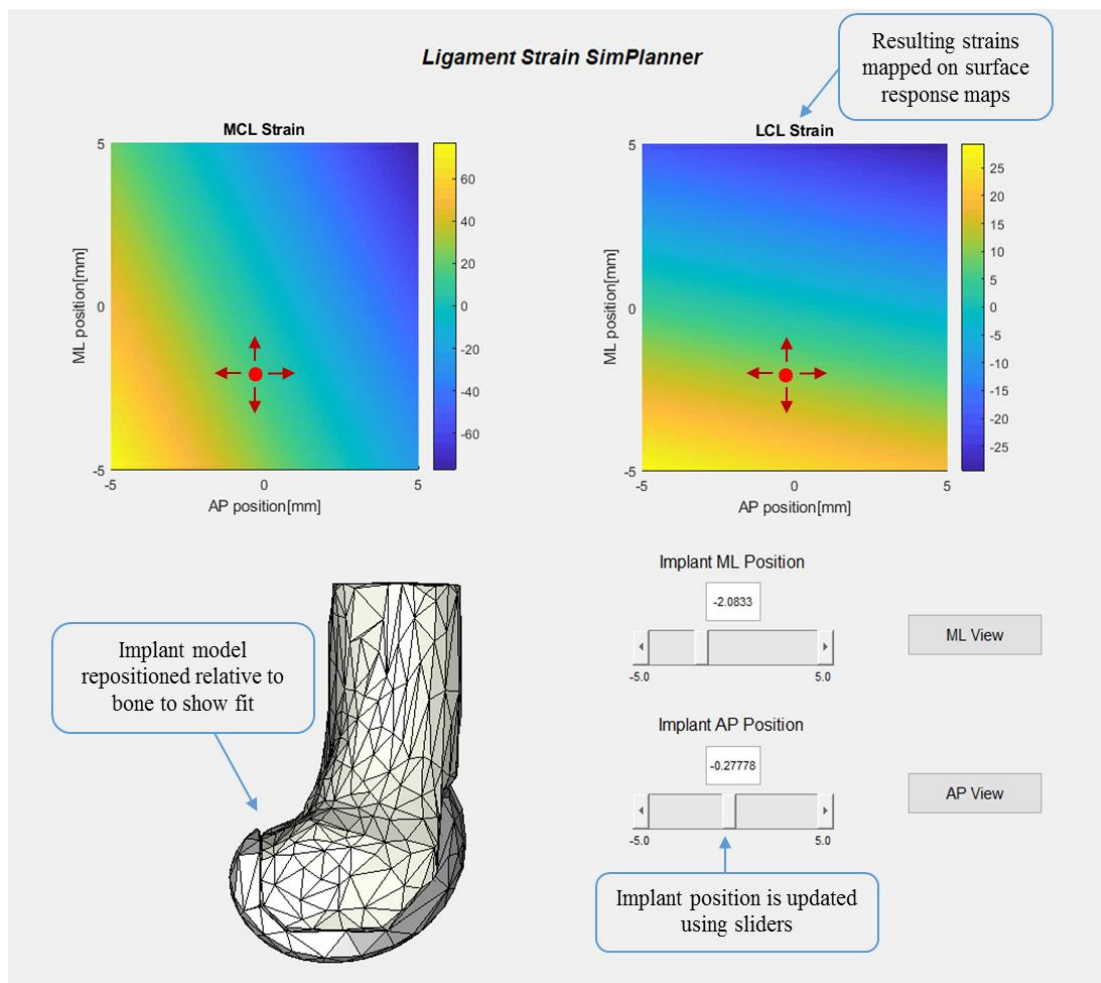


Figure 34. Rendering of Matlab graphical user interface for implant positioning using collateral ligament strain surface response maps. Positioning of the implant in the AP and ML directions is accomplished with the use of interactive sliders. The resulting ligament strains are projected on the maps and the implant model position is updated.

```

function varargout = TestingFigure(varargin)

% TESTINGFIGURE MATLAB code for TestingFigure.fig
% TESTINGFIGURE, by itself, creates a new TESTINGFIGURE or raises the existing
% singleton*.
%
% H = TESTINGFIGURE returns the handle to a new TESTINGFIGURE or the handle to
% the existing singleton*.
%
% TESTINGFIGURE('CALLBACK',hObject,eventData,handles,...) calls the local
% function named CALLBACK in TESTINGFIGURE.M with the given input arguments.
%
% TESTINGFIGURE('Property','Value',...) creates a new TESTINGFIGURE or raises the
% existing singleton*. Starting from the left, property value pairs are
% applied to the GUI before TestingFigure_OpeningFcn gets called. An
% unrecognized property name or invalid value makes property application
% stop. All inputs are passed to TestingFigure_OpeningFcn via varargin.
%
% *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
% instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help TestingFigure

% Last Modified by GUIDE v2.5 13-May-2019 11:55:33

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @TestingFigure_OpeningFcn, ...
                  'gui_OutputFcn',  @TestingFigure_OutputFcn, ...
                  'gui_LayoutFcn',   [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before TestingFigure is made visible.
function TestingFigure_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to TestingFigure (see VARARGIN)

% Choose default command line output for TestingFigure
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes TestingFigure wait for user response (see UIRESUME)
% uiwait(handles.figure1);

%insert bone images
% axes(handles.axes3D); % Use actual variable names from your program!
% [V,D,F] = stl2matlab('FemurSmall.stl');
% plotSTL(V,D);
%
set(0,'defaultfigurecolor',[1 1 1])

```

```

% Setup heat plots
AP = -5:0.1:5;
ML = -5:0.1:5;
aMCL = zeros(101,101);
aLCL = zeros (101,101);
for i=1:101
    for j = 1:101

        aMCL(i,j) = -5.4199*AP(1,i)-10.014*ML(1,j);
    end

end

for i=1:101
    for j = 1:101

        aLCL(i,j) = -4.89*AP(1,i)-0.98*ML(1,j);
    end

end

%Plot MCL heat map
axes(handles.axesMCL);
surf(AP,ML,aMCL);
view(0,90);
shading interp;
colorbar; xlabel('AP position[mm]'); ylabel('ML position[mm]');title('MCL Strain');

%Plot LCL heat map
axes(handles.axesLCL);
surf(AP,ML,aLCL);
view(0,90);
shading interp;
colorbar; xlabel('AP position[mm]'); ylabel('ML position[mm]');title('LCL Strain');

%plot 3d bone and implant
axes(handles.axes3D);
[x1,y1,z1,c1] = stlread('FemurSmall_dec.stl');
[x2,y2,z2,c2] = stlread('FemImp.stl');
hold on
bone = patch(x1*1000,y1*1000,z1*1000,[1 1 0.94], 'FaceLighting', 'gouraud','AmbientStrength',
0.35);
material('dull')
camlight('headlight')
imp = patch(x2,y2,z2,[0.7 0.7 0.7]);
axis off;
camorbit(180,0);
camroll(180);

rotate3d on;

% --- Outputs from this function are returned to the command line.
function varargout = TestingFigure_OutputFcn(hObject, eventdata, handles)
% varargout cell array for returning output args (see VARARGOUT);
% hObject handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on slider movement.
function slider1_Callback(hObject, eventdata, handles)
% hObject handle to slider1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
% get(hObject,'Min') and get(hObject,'Max') to determine range of slider

```

```

%obtains the slider value from the slider component
sliderValue = get(handles.slider1, 'Value');
%puts the slider value into the edit text component
set(handles.slider_editText, 'String', num2str(sliderValue));
% Update handles structure
guidata(hObject, handles);

%Create tracker that updates with slider

% Do first for MCL plot
axes(handles.axesMCL);
hold on %// Important, otherwise it erases the current data plotted.
%// Get position of both sliders
xval = (get(handles.slider1, 'value'));
yval = (get(handles.slider2, 'value'));
%Delete old marker
if length(handles.axesMCL.Children)>1
    child = get(gca, 'Children');
    delete(child(1));
end
%// Plot new marker.You can customize its properties as you want.
scatter3(handles.axesMCL, xval, yval, 2000, 100, 'r', 'filled');

% Do second for LCL plot
axes(handles.axesLCL);
hold on %// Important, otherwise it erases the current data plotted.
%// Get position of both sliders
xval = (get(handles.slider1, 'value'));
yval = (get(handles.slider2, 'value'));
%Delete old marker
if length(handles.axesLCL.Children)>1
    child = get(gca, 'Children');
    delete(child(1));
end
%// Plot new marker.You can customize its properties as you want.
scatter3(handles.axesLCL, xval, yval, 2000, 100, 'r', 'filled');

%Update implant position
axes(handles.axes3D);
hold on
%remove implant in previous position
child = get(gca, 'Children');
delete(child(1));
%load implant new position
[x2, y2, z2, c2] = stlread('FemImp.stl');
x2 = x2+(xval/1000);
z2 = z2+(yval/1000);
imp = patch(x2, y2, z2, [0.7 0.7 0.7]);

% --- Executes during object creation, after setting all properties.
function slider1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject, 'BackgroundColor'), get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', [.9 .9 .9]);
end

function slider_editText_Callback(hObject, eventdata, handles)
% hObject    handle to slider_editText (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject, 'String') returns contents of slider_editText as text
%        str2double(get(hObject, 'String')) returns contents of slider_editText as a double
%get the string for the editText component
sliderValue = get(handles.slider_editText, 'String');
%convert from string to number if possible, otherwise returns empty

```

```

sliderValue = str2num(sliderValue);
%if user inputs something is not a number, or if the input is less than 0
%or greater than 100, then the slider value defaults to 0
if (isempty(sliderValue) || sliderValue < 0 || sliderValue > 100)
    set(handles.slider1,'Value',0);
    set(handles.slider_editText,'String','0');
else
    set
end
% --- Executes during object creation, after setting all properties.
function slider_editText_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider_editText (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on slider movement.
function slider2_Callback(hObject, eventdata, handles)
% hObject    handle to slider2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%         get(hObject,'Min') and get(hObject,'Max') to determine range of slider

%obtains the slider value from the slider component
sliderValue = get(handles.slider2,'Value');
%puts the slider value into the edit text component
set(handles.slider_editText2,'String', num2str(sliderValue));
% Update handles structure
guidata(hObject, handles);

%Create tracker that updates with slider

% Do first for MCL plot
axes(handles.axesMCL);
hold on %// Important, otherwise it erases the current data plotted.
%// Get position of both sliders
xval = (get(handles.slider1,'value'));
yval = (get(handles.slider2,'value'));
%Delete old marker
if length(handles.axesMCL.Children)>1
    child = get(gca,'Children');
    delete(child(1));
end
%// Plot new marker.You can customize its properties as you want.
scatter3(handles.axesMCL,xval,yval,2000,100,'r','filled');

% Do second for LCL plot
axes(handles.axesLCL);
hold on %// Important, otherwise it erases the current data plotted.
%// Get position of both sliders
xval = (get(handles.slider1,'value'));
yval = (get(handles.slider2,'value'));
%Delete old marker
if length(handles.axesLCL.Children)>1
    child = get(gca,'Children');
    delete(child(1));
end
%// Plot new marker.You can customize its properties as you want.
scatter3(handles.axesLCL,xval,yval,2000,100,'r','filled');
%Update implant position
axes(handles.axes3D);
hold on
%remove implant in previous position

```



```

child = get(gca,'Children');
delete(child(1));
%load implant new position
[x2,y2,z2,c2] = stlread('FemImp.stl');
x2 = x2+(xval/1000);
z2 = z2+(yval/1000);
imp = patch(x2,y2,z2,[0.7 0.7 0.7]);

% --- Executes during object creation, after setting all properties.
function slider2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider_editText2_Callback(~, eventdata, handles)
% hObject    handle to slider_editText2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of slider_editText2 as text
%        str2double(get(hObject,'String')) returns contents of slider_editText2 as a double

%get the string for the editText component
sliderValue = get(handles.slider_editText2,'String');
%convert from string to number if possible, otherwise returns empty
sliderValue = str2num(sliderValue);
%if user inputs something is not a number, or if the input is less than 0
%or greater than 100, then the slider value defaults to 0
if (isempty(sliderValue) || sliderValue < 0 || sliderValue > 100)
    set(handles.slider2,'Value',0);
    set(handles.slider_editText,'String','0');
else
    set(handles.slider2,'Value',sliderValue);
end

% --- Executes during object creation, after setting all properties.
function slider_editText2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider_editText2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
axes(handles.axes3D);
view(90,0);
%camorbit(180,0);
camroll(90);

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
axes(handles.axes3D);
view(0,90);

```