Spring 5-2023

OpenSim-Based Musculoskeletal Modeling: Foundation for Interactive Obstetric Simulator

Bahador Dodge
Old Dominion University, bdodge63@gmail.com

Recommended Citation

Dodge, Bahador. "OpenSim-Based Musculoskeletal Modeling: Foundation for Interactive Obstetric Simulator" (2023). Master of Science (MS), Thesis, Electrical & Computer Engineering, Old Dominion University, DOI: 10.25777/fjya-hq06
https://digitalcommons.odu.edu/ece_etds/253

OPENSIM-BASED MUSCULOSKELETAL MODELING: FOUNDATION FOR INTERACTIVE OBSTETRIC SIMULATOR

by

Bahador Dodge
B.S. February 2008, Azad University, Iran

A Thesis Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

MODELING AND SIMULATION ENGINEERING

OLD DOMINION UNIVERSITY
May 2023

Approved by:

Michel Audette (Director)

Yuzhong Shen (Member)

Hunter Jared Bennett (Member)

ABSTRACT

OPENSIM-BASED MUSCULOSKELETAL MODELING: FOUNDATION FOR INTERACTIVE OBSTETRIC SIMULATOR

Bahador Dodge
Old Dominion University, 2023
Director: Dr. Michel Audette

The use of mathematical and computational models to understand complex biological systems, such as the human birth process, is a rapidly growing field in medicine. These models can be used to optimize and personalize medical treatments for individual patients, enhance training, and aid in educational efforts. While recent advancements in healthcare, particularly in obstetrics, have improved care for mothers and babies, studies and government reports indicate a rising rate of maternal mortality in the United States.

Despite this rising trend, there is a lack of detailed studies concerning the use of modeling and simulation to develop an interactive obstetrics simulator that can aid both practitioners and patients. This research builds upon a novel template for developing an interactive obstetric simulator and aims to replicate an onerous finite element vaginal delivery simulation with an interactive, patient-specific simulator that emphasizes musculoskeletal dynamics. The study utilizes the open-source platform of OpenSim and inverse-kinematic solutions to develop fetal and maternal musculoskeletal models and simulate birth on the musculoskeletal level.

I dedicate this thesis to my beloved wife, Sheida, and to my parents, Roya and Mehran. Their unwavering love, enduring patience, and support have been my pillars of strength throughout my master's journey.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my committee members for their invaluable support, guidance and expertise throughout my research and the writing of this thesis. Their patience and dedication have been instrumental in the completion of this work.

I am particularly grateful to my advisor, Dr. Michel Audette, for his unwavering support, encouragement and for providing me with the tools and resources to succeed. His guidance and expertise have been invaluable in shaping my research, and I am honored to have had the opportunity to work under his mentorship.

TABLE OF CONTENTS

Page

Chapter

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

CHAPTER 1

INTRODUCTION

A detailed study conducted in 2020 by Tikkanen et al. [1] reveals that the U.S. has a high maternal mortality rate among developed countries with 17 percent of deaths occurring on the day of delivery, due to too few midwifes and maternity care providers. Moreover, the Centers for Disease Control and Prevention CDC reports a nearly 40 percent increase in maternal mortality rate in 2021 compared to 2020 with 1,205 deaths in total in the U.S. [2].

Although lack of midwifes or access to maternity care impacts maternal health, there are other factors contributing to labor outcome that affect both maternal and neonatal health including obstetrics training [3] and delivery complications [4]. Results of a study conducted by Birch et al. show improved obstetrics performance with obstetrics training [3]. She and her team conclude that obstetrics is a medical specialty associated with a high level of risk in which, to some degree, complications are unavoidable, highlighting the importance of training [3].

In 2008, some form of pregnancy complication was reported in 94.1 percent of the 4.2 million deliveries that occurred that year, Elixhauser and Wier reports [4]. A study in 2014 found that major obstetrical complications vary significantly among U.S. hospitals, depending on their performance [5]. The combination of these recent studies emphasizes the significance of obstetrics training and its correlation with obstetrics performance in managing labor complications.

Obstetric complications can range from mild to severe and encompass a variety of issues. Some of the more frequently encountered complications during labor include lack of progression, preterm delivery, premature rupture of membranes, infections, stillbirth, and high blood pressure, while more serious complications include shoulder dystocia, preeclampsia, and prolapsed umbilical cord, among others [6].

To put it in perspective, preterm delivery, which is the birth of a neonatal before 37 weeks of gestation, occurs in approximately 10% of all pregnancies. Normally, a pregnancy lasts about 40 weeks from the first day of the last menstrual period. Preterm delivery can lead to various health problems for the baby, including respiratory distress syndrome, low birth weight, and developmental delays [6].

Shoulder dystocia is a medical issue that occurs in about 1-2% of pregnancies and refers to difficulty delivering the fetal shoulder after the head has been delivered as depicted in figure 1. It is often caused by a large fetus trying to pass through a typical sized pelvis. The incidence of shoulder dystocia has increased in recent decades, and it is a serious emergency for the baby that can result in breathing difficulties, death, brain damage, or brachial plexus injury. The latter is caused by the efforts to resolve the dystocia [7].

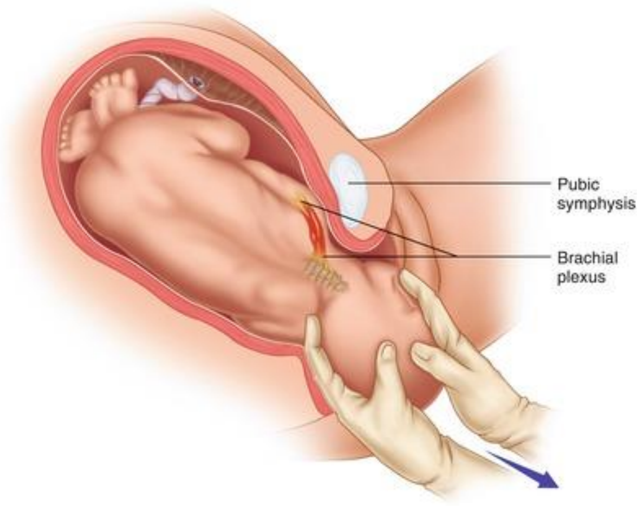


Figure 1. Shoulder dystocia, reproduced from Obgynkey.com [8].

During a lack of progression, as in cases of shoulder dystocia, various interventions can be applied, such as performing an episiotomy, executing the McRoberts maneuver by flexing and abducting the mother's hips and positioning her thighs onto her abdomen, applying suprapubic pressure to the fetal anterior shoulder, and delivering the posterior arm. If these techniques are unsuccessful in resolving the lack of progression, more drastic measures like deliberately breaking the clavicle or performing a cesarean may be considered [8].

Obstetrician training for delivery complications is done using either low-fidelity or high-fidelity modalities, mainly through mannequins [9]. By definition, a high-fidelity training modality is more realistic with more features than a low-fidelity modality that lacks some of components or features, making the training scenario less realistic.

A study conducted in 2006 by Crofts et al. shows a 94% overall delivery success rate in using high-fidelity mannequins for obstetrics training during a shoulder dystocia event compared to 72% when training with a low-fidelity mannequin [10]. However, there is conflicting research on this subject.

Figure 2. High-fidelity obstetrics NOELLE mannequin form Gaumard.com [11].

In 2019, Massoth et al. reported achieving better results in training with low-fidelity mannequins and recorded low-performance in groups trained with high-fidelity mannequins [11]. Prior to this, in 2006, Scerbo et al. reached the same conclusion that training with low-fidelity mannequins shows better results in overall performance [12]. In contrast, Matsumoto et al. found no notable distinction between training with low-fidelity or high-fidelity models [13]. A study by Lee et al. in 2009 also found no significant performance margin between training with low-fidelity or high-fidelity mannequins, finding no difference between the two for training scenarios [14].

Figure 3. Low-fidelity obstetrics PROMPT Flex simulator, reproduced from Laerdal.com [15].

An extensive study in 2017 conducted by Krishnan et al. reports multiple disadvantages of training with such modalities in general. Identifying a broad range of limitations such as incomplete mimicking of human systems, defective learning by poor mannequin design, expensive cost of increased fidelity and lack of correct physical representation, etc. [16].

Despite the contradictory research on the benefits of using low and high-fidelity modalities in medical training, the aim of this review is not to undermine the importance of mannequin-based training. However, with recent advancements in technology, there has been a growing trend towards utilizing computer-based simulations and virtual reality in medical training.

In 2023, Mahling et al. studied use of computer-based simulation in medicine showing a significant positive attitude of trainees towards computer-based training [17]. In 2002, Ravert studied the same topic and found a 75% increase in positive effects of computer-based simulation in training [18]. In 2014, Karakus et al. found a significant improvement in medical training could be achieved through computer-based simulation resulting in increased performance in multi-step diagnostic approaches [19].

Modeling and Simulation (M&S) is a computational discipline that involves creating a model of a system as an abstraction and using simulations to observe model behavior and draw conclusions. In the field of medicine, M&S has become increasingly important as it allows for the study of complex biological systems [20]. The use of mathematical and computational models in medicine has become an essential tool for understanding the underlying processes in human labor and delivery [21]. Additionally, M&S can be used to optimize and personalize medical treatments for individual patients and act as an educational tool for medical students and practitioners [22]. Finally, M&S can improve the efficiency of medical intervention techniques by using virtual reality (VR) and augmented reality (AR) technology for training sessions [23].

M&S has been used in predictive analyses such as predicting human birth rates and fetal death at a population scale [24], as well as in medical surgical applications [25]. However, there have been no detailed studies using M&S to model and simulate the human birth process at interactive rates, particularly in cases of delivery complications, for the purpose of training practitioners, until recently.

In 2021, Audette et al. proposed a template of a novel obstetrics simulator pipeline using open-source platforms [22]. Their research outlined the architecture and foundation of the simulator, with the core components of the simulator including real-time birthing simulation, a pipeline that utilized fetal MRI to register skin surface segmentation, and the open-source musculoskeletal dynamics simulation platform OpenSim [26] for representing fetal movement

[22]. These coupled with VR and haptic devices, enable obstetricians to practice and refine their skills in managing complex delivery scenarios, such as shoulder dystocia, in a controlled and safe environment. The foundation for this interactive simulation project is based on the work of a Porto-based research group.

The Porto-based research group, Parente, Natal Jorge and Oliveira, conducted a series of predictive studies to examine the deformation of the pelvic floor during delivery using Finite Element Simulation (FE) [27-29]. The focus of their studies is on mechanisms behind pelvic floor dysfunction related to vaginal delivery. A central element of these studies is the development of a fetus anatomy consisting of a tetrahedral surface mesh devoid of bones or ligaments as shown in figure 4.

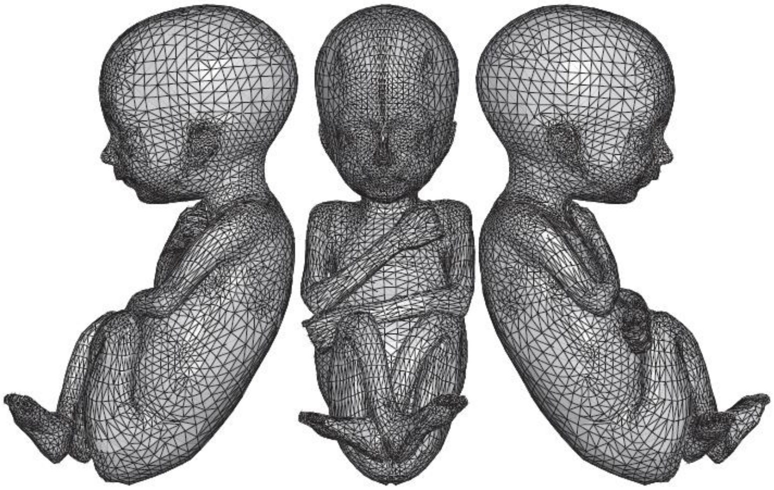


Figure 4. Fetal model used in the predictive simulations of Porto-based group from Parente [27].

Finite element modeling and simulation is a mathematical technique used to analyze complex engineering and scientific problems. It is based on the concept of dividing a complex system or structure into smaller, more manageable parts called finite elements. Each FE is analyzed individually, and the results are combined to obtain a comprehensive understanding of the entire system. FE modeling is widely used in various fields such as mechanical engineering or biomechanics [30].

The mathematical aspect of FE modeling involves using partial differential equations to represent the physical behavior of the system. These equations describe how the system responds to various external stimuli such as forces and pressure. The equations are then discretized into smaller, simpler equations that can be solved using numerical methods. These smaller equations are typically solved iteratively, and the results are combined to obtain the final solution. FE modeling and simulation can be performed using specialized software, such as Abaqus [31], which automates the entire process.

The accuracy of FE modeling and simulation is dependent on the accuracy of the mathematical model used. Therefore, the selection of appropriate material models and boundary conditions is crucial in obtaining accurate results. In addition, the mesh size, or the size of each finite element, plays a significant role in the accuracy of the results. Smaller mesh sizes result in more accurate results but require more computational resources and time [30].

The FE simulation of the Porto-based research group represented the maternal anatomy with a static collection of pelvis bones. By defining a series of finite elements and assigning four rigid points to their fetal model, the movement of the fetus through the vaginal canal is controlled. The location of these points is in the center mass of the head (P1), collar (P2), torso (P3) and hip (P4) as depicted in figure 5. The pelvis, representing the maternal anatomy, is depicted in figure 6.

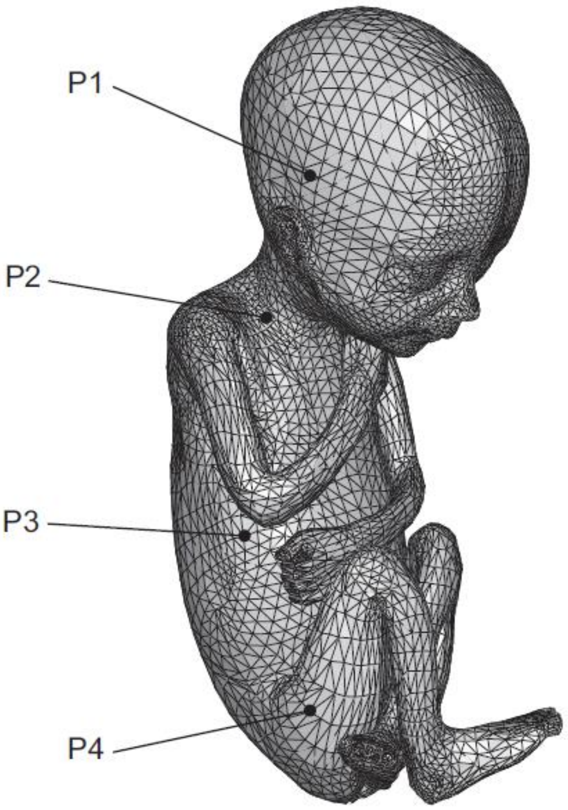


Figure 5. Location of four rigid body points on FE fetal anatomy from Parente [27].

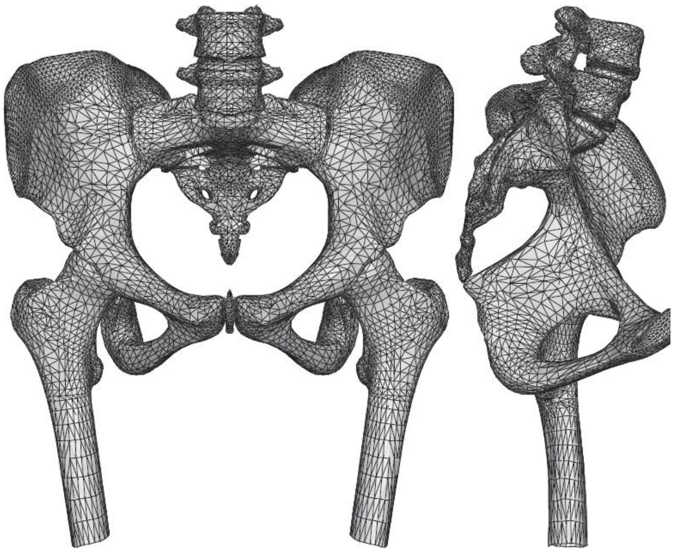


Figure 6. Maternal anatomy representation of the FE simulation from Parente [27].

The FE fetal anatomy is created with anthropometric measurements similar to those of a full-term fetus in typical obstetric practice. The predictive simulation took about twenty hours to complete, culminating in the passage of the FE fetal head through the pelvic floor. A sequence of images, depicted in figure 7, captured the progression of the FE fetal movement through the birth canal.

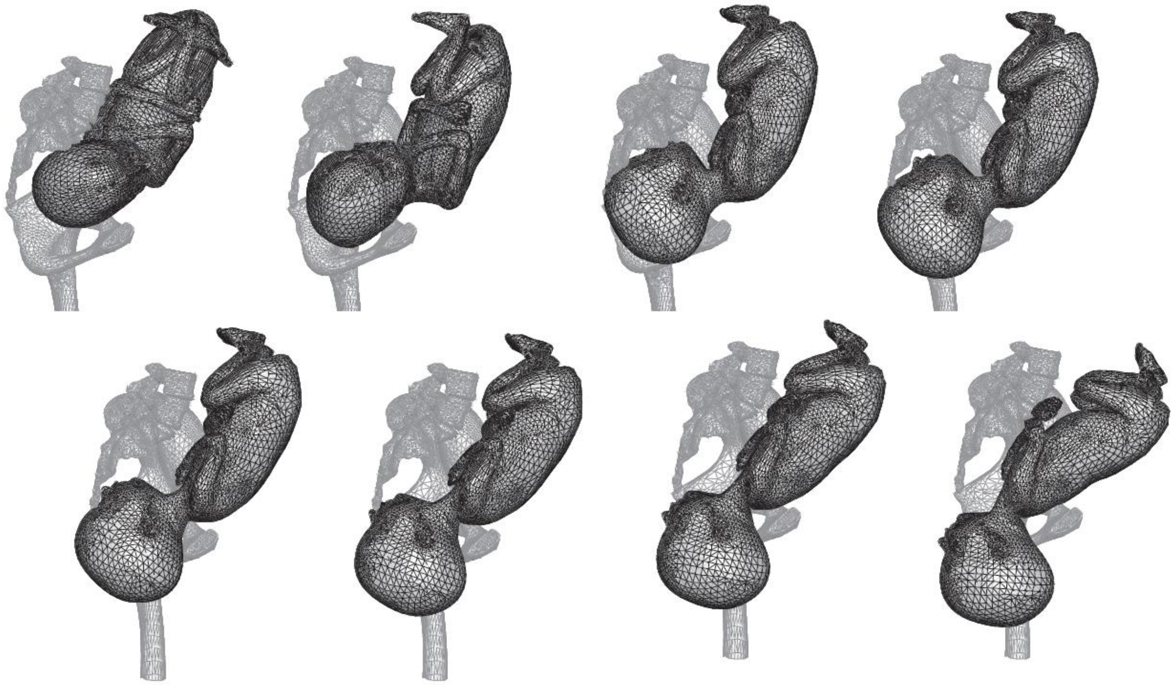


Figure 7. FE fetal movement through maternal pelvis from Parente [27].

While studies by Parente, Natal Jorge, and Oliveira show promise in calculating the impacts of vaginal delivery on the pelvis, their work has some limitations. First, their simulations run for several hours [28], making them unsuitable for training purposes where scenarios need to be fast and on-demand. Second, their central figure lacks structural anatomies [28], which are crucial when recreating complicated scenarios like shoulder dystocia that require shoulder bones to begin with. Third, the maternal anatomy is only represented by a static pelvis [28].

The limitations of the Porto-based group’s work, and the novelty of Audette's proposed obstetrics simulator pipeline [22] prompted this research to incorporate both studies. By using the finite element fetal anatomy and its movement [27] and building upon the proposed project [22], the aim of this research is to develop a musculoskeletal model of the FE fetus by using the open-source platform OpenSim [26] and transforming the onerous finite element study into a patient-specific, real-time, virtual simulation that incorporates a full maternal musculoskeletal structure.

1.1 Problem Statement

Maternal mortality rates in the United States are increasing, and several contributing factors have been identified, including a shortage of obstetricians and midwives, training, and complications during labor and delivery. There are many types of delivery complications, each with varying rates of occurrence, underscoring the importance of training to prepare for any situation.

Traditionally, obstetrics training has relied on either low-fidelity or high-fidelity mannequin-based modalities, which have proven useful but also have limitations, according to conflicting studies. To address these issues, there is a growing trend toward computer-based simulation for obstetric training, which has the potential to be more effective than traditional methods.

While simulation is already used in predictive modeling in obstetrical care, there is a lack of detailed research dedicated to musculoskeletal dynamic incorporated interactive simulations for delivery training when faced with complicated scenarios such as lack of progression or, in rare cases, shoulder dystocia.

1.2 Research Objective

This research aims to develop a patient-specific, real-time, virtual obstetrics simulator that will subsequently use haptic devices to facilitate delivery training for obstetricians, particularly for complicated deliveries such as lack of progression or, in more rare cases, shoulder dystocia. To accomplish this, the study uses data from an extensive finite element study of vaginal delivery, which represents maternal anatomy with a static pelvis bone and a hollow tetrahedral fetus mesh and builds upon a novel template of obstetric simulator which is proposed recently.

The study models the musculoskeletal structure of the fetus in terms of its FE anthropometric data using OpenSim and incorporates a fully modeled musculoskeletal maternal body to simulate the musculoskeletal passage of the fetus through the vaginal canal. By converting the lengthy finite element predictive simulation to an interactive, patient-specific virtual simulation, this study represents preliminary work toward developing a real-time obstetrics simulator. Additionally, this study presents two approaches to visually enhance the musculoskeletal representation of the fetal anatomy by skinning.

1.3 Thesis Content

Chapter 2 focuses on the background study of fetal musculoskeletal modeling, as the goal of this research is to develop a musculoskeletal model of the FE fetus and simulate birth though use of OpenSim [26]. The background section reviews prior works and methods.

Chapter 3 delves into the implementation of OpenSim [26] for skeletal modeling of fetal and maternal structures and presents two skinning approaches for enhancing the representation of fetal anatomy. The chapter begins with data acquisition and data validation, then explains the rescaling of the female adult model to fetal size and marker placement for motion tracking with implementation of inverse kinematics (IK) to simulate fetal movement, which then is validated through multiple steps. Then two different approaches to skinning are presented. One, Biomechanical Animated Skinned Human (BASH) and two, manual skinning using a generic full-term fetal anatomy are discussed.

Chapter 4 presents the results of the work carried out in Chapter 3. The chapter provides a comprehensive explanation of the main objective of this study and analysis of the results.

Chapter 5 provides concluding remarks and the future work of the thesis.

CHAPTER 2

BACKGROUND

Obstetrics, the branch of medicine that deals with pregnancy, childbirth, and postpartum care, goes back to ancient times. However, obstetrics as a medical specialty began to develop in the 16th century. In the 19th and 20th centuries, significant advances were made in obstetrics, including the development of forceps and other tools to assist delivery, and the introduction of modern techniques to reduce the risk associated with labor [32].

While the history of obstetrics dates back a long time, the focus of this research is on converting an onerous FE vaginal delivery simulation to an interactive, patient-specific simulation that centers on OpenSim [26] fetal musculoskeletal modeling based on anthropometric data of the FE fetus. Therefore, this section aims to review relevant literature on modeling of the fetal or neonatal musculoskeletal system in OpenSim [26] and introduce OpenSim [26] and its capabilities in more detail.

Literature on the complete neonatal or fetal musculoskeletal system is very limited, with few studies focusing on this topic. Lim et al. conducted research in 2022 that explores the difficulties associated with creating subject-specific musculoskeletal models of neonates. This study highlights the importance of these models in understanding the mechanics of movement but also notes the unique challenges that arise when creating such models [33].

One of the key challenges highlighted in the article is the lack of data available on neonatal movement. Unlike adults, neonates are unable to perform complex movements or follow instructions, which makes it difficult to capture the necessary data to create accurate models. The study proposes using various techniques such as motion capture, MRI, and ultrasound to obtain the required data [33]. Another challenge is the difficulty of accurately representing the complex interactions between the various components of the musculoskeletal system in neonates, like muscle interactions, which the study suggests could be addressed by incorporating physiological data and modeling the system at a higher level of abstraction [33].

Despite the challenges highlighted by Lim et al. in modeling the musculoskeletal system of neonates, their study is able to develop a model of a newborn infant by utilizing motion capture data by placing markers on the newborn and transporting the marker data from the experiment onto a generic GAIT 2392 OpenSim model [26], simulating the movement of lower extremities in a neonatal model.

Verbruggen et al. [34] presented a paper that focuses on modeling the biomechanics of fetal movements. The study used a combination of experimental data and finite element modeling to investigate the forces involved in fetal movements during pregnancy. The methodology consists of three steps which are illustrated in figure 8.

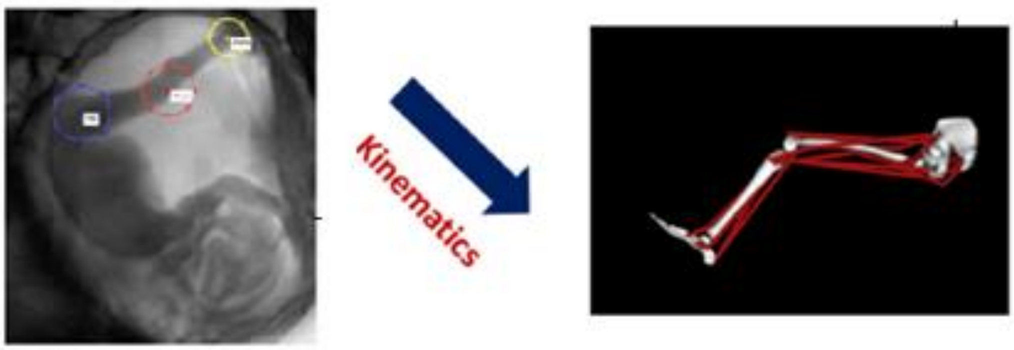


Figure 8. Reproduced from Verbruggen et al. study [34].

Step one tracks the displacement of the fetal kicks observed in MRI images taken from three fetuses aged 20 – 22 weeks, using custom software developed in MATLAB. Step two calculates the forces generated by fetal leg muscles producing the observed kicks. This step is modeled using a finite element study. Step three consists of creating a partial musculoskeletal model of the right leg of a fetal body using OpenSim [26] and is based on the MRI measurements. The partial musculoskeletal model is the result of a modified 3DgaitModel2354 [26] model and consists of the right pelvis, femur, tibia, talus, calcaneus and toes [34].

The study used OpenSim [26] to describe fetal movements and to calculate intramuscular forces necessary for generating kicks. It found that the fetal movements were generated by the excitation of skeletal muscles and the movement of joints. Their results suggest that their model can provide insight into the development of the fetal musculoskeletal system and may have clinical applications, such as in the diagnosis of fetal abnormalities [34].

A study by Kim et al. in 2022 discussed the importance of investigating the fundamental principles of cognitive development and developmental disorders from the prenatal to postnatal stages. Determining the necessary sensorimotor experiences in fetuses and infants by directly evaluating the role of the cortex in cognitive development is challenging due to technical and ethical difficulties. To address this, the study proposes a simulation-based approach for explaining the mechanism of cognitive development using a musculoskeletal and uterine model to simulate fetal movements [35].

The study consists of multiple stages. First, the study models a full musculoskeletal structure of a fetus by rescaling a generic model in OpenSim [26] using a scaling tool. Then, the study wrapped the skeletal system using a generic high-quality skin mesh. Second, a soft uterine model was used to contain the fetal anatomy inside it.

The resulting wrapping and simulation is generated using MuJoCo software [36]. Lastly, the simulation is conducted using a MuJoCo and OpenSim [26] Python wrapper. Figure 9 shows the high level sequence of Kim et al.'s study [35].

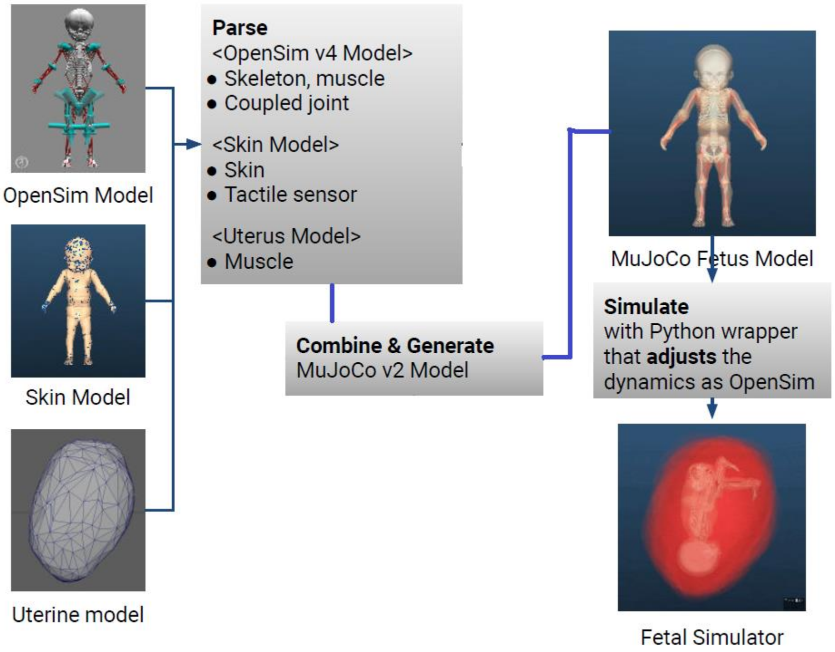


Figure 9. Modeling sequence from Kim et al. study [35].

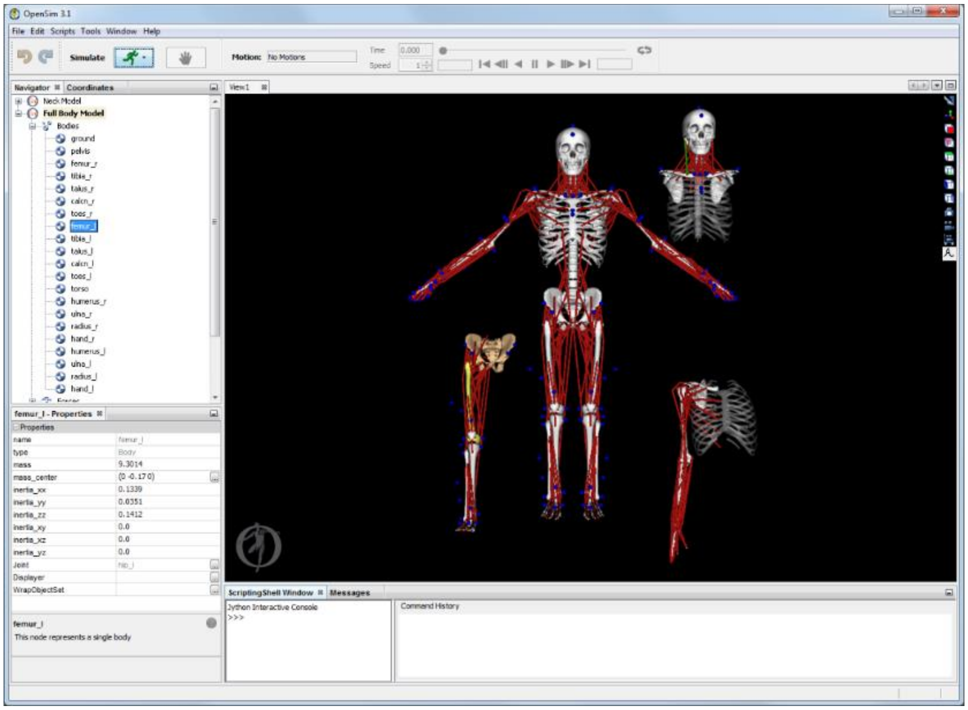
Despite the limited number of studies on fetal musculoskeletal modeling, a common thread among them is the utilization of OpenSim [26] software for modeling, analyzing, and simulating the musculoskeletal dynamics of fetal and neonatal structures. Therefore, it is crucial to conduct a thorough review of OpenSim [26] software and gain an understanding of its capabilities, as it is a powerful tool in the field of biomechanics and holds significant relevance to this research.

2.1 What is OpenSim?

OpenSim [26] is a free, open-source software package used to model and simulate musculoskeletal structures and analyze their movement. It is commonly used in biomechanics research to investigate human movement and can be applied to a wide range of fields. OpenSim

[26] allows users to create and analyze models of various anatomical structures, including bones, joints, and muscles, and simulate their movement in different scenarios. Figure 10 shows OpenSim graphical user interface (GUI).

OpenSim [26] has many tools for model development, including the ability to create full-featured musculoskeletal models with rigid body parts, joints, and muscles. In addition to building custom models, OpenSim [26] also provides a library of musculoskeletal models that are validated based on specific parts under study and can be used as a starting point. These models have been extensively tested and can be modified to suit the specific needs of the research project. In this way, OpenSim [26] provides a powerful tool for investigating the biomechanics of the human body and developing simulations of movement and function.



femur_l - Properties

Properties

name	femur_l
type	Body
mass	9.3014
mass_center	(0 -0.17 0)
inertia_xx	0.1339
inertia_yy	0.0351
inertia_zz	0.1412
inertia_xy	0.0
inertia_xz	0.0
inertia_yz	0.0
Joint	hip_l
Displayer	
WrapObjectSet	

femur_l

This node represents a single body

Jython Interactive Console
>>>

Figure 10. The GUI of OpenSim from Delp et al. [37].

In terms of model analyses and simulation, OpenSim [26] is capable of model scaling, importing experimental data such as marker trajectories (in motion capture), inverse dynamics, static optimization, and forward dynamics. Moreover, the ability to calculate muscle-driven forward simulations as well as plotting markers and muscle activity is also supported in OpenSim.

The OpenSim [26] model is a structured representation of the musculoskeletal system and its dynamics, composed of interconnected modules that depict biological structures or devices [38]. Its purpose is to simulate the physical system and consists of two elements: a system of equations with fixed physical parameters and a state vector that contains variables that alter over time, such as joint angles. Figure 11 explains this abstraction process.

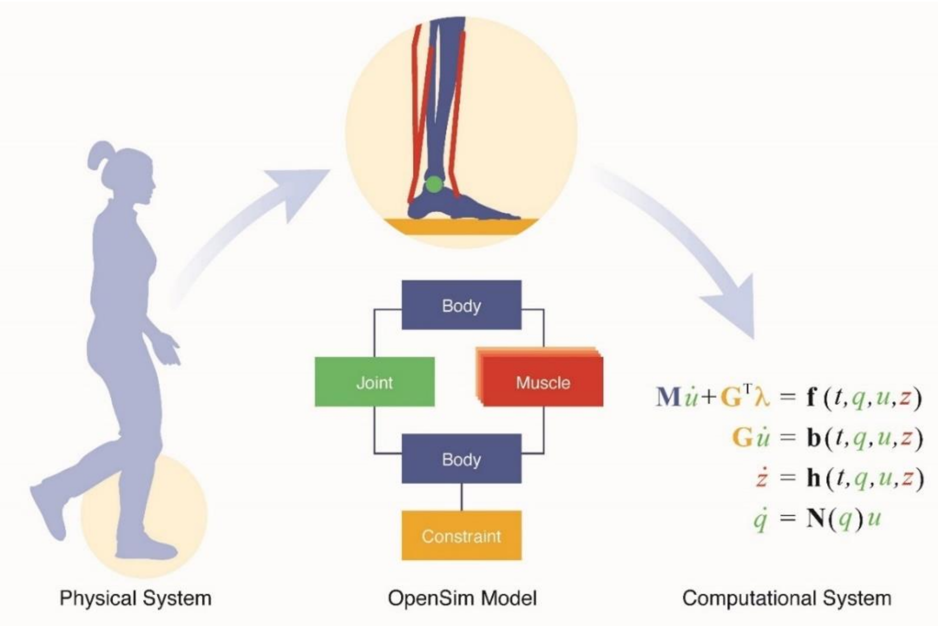


Figure 11: OpenSim model system [39].

OpenSim [26] stands out as a valuable tool for modeling and simulating movement due to its capacity to generate subject-specific musculoskeletal models and its analytical tools such as forward and inverse dynamics and kinematics [26].

CHAPTER 3

METHODOLOGY

The methodology of this thesis outlines the steps taken to create the foundation of a virtual obstetrics simulator to improve the training of obstetricians during delivery processes with complications, such as shoulder dystocia. The methodology begins with the acquisition and processing of data, the implementation of fetal and maternal models in OpenSim [26], the application of an inverse kinematic solution, and the replication of the delivery simulation. Additionally, this study discusses the initial steps taken toward visual enhancement of the fetal musculoskeletal structure with skinning by replicating the BASH model [40] and employment of Blender [41]. The results and analysis of each step taken are discussed here.

3.1 Data Acquisition

The FE vaginal delivery simulation of Parente, Natal Jorge and Oliveira [27] is conducted using Abaqus [31], a powerful simulation tool widely used for FE studies. The FE output database is a binary file that stores information related to the maternal pelvis and fetal tetrahedral mesh as well as fetal displacement and orientation in 3D space. The data has the standard engineering coordinate system which is called the global coordinate system. The output data then is separated into two categories.

The FE simulation output file contains displacement and some rotational data of the four rigid body points introduced in the introduction section. The location of these four rigid body points is described in table 1. These rigid points also were placed in the center of the mass as depicted in figure 12.

Table 1. Rigid body points location

Rigid Points	Location
P1	Head
P2	Collar
P3	Torso
P4	Hip

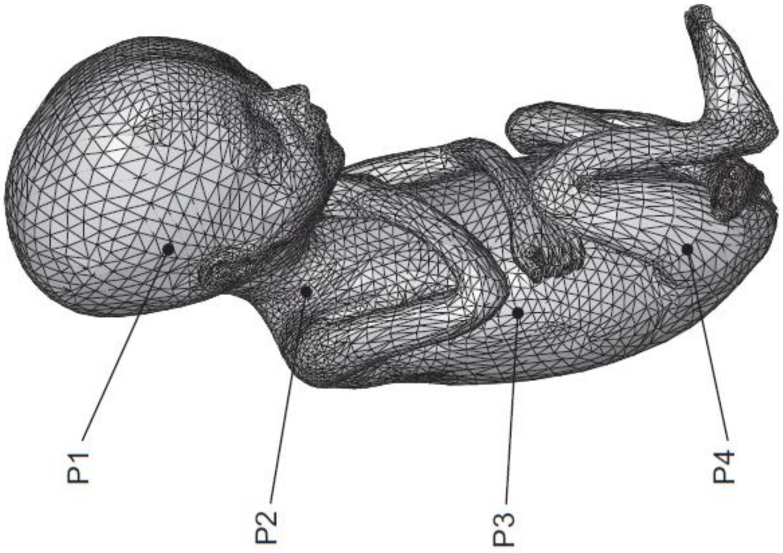


Figure 12. Rigid body points locations [27].

After extracting the data from the Abaqus [31] output database, it is separated into two categories. The first category includes the displacement coordinates of four points in X, Y, and Z coordinates throughout 1112 frames spanning 7 seconds. These coordinates are in standard engineering format. These data points are then compiled into an Excel file for further processing.

The resulting Excel file provides information on the motion of the four points through 3D space, over time. Table 2 presents the first five rows of the data in this category.

Table 2. First 5 rows of displacement data extracted from Abaqus output file

Time	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z
	Head			Collar			Torso			Hip		
0	-63.5434	44.63574	-9.3221	-23.9283	64.09013	-83.8562	18.38773	146.9963	-67.3236	47.82998	197.733	-45.4331
0.02	-63.5434	44.63574	-9.3021	-23.0598	63.45011	-82.9962	19.11267	146.447	-66.5587	48.31806	197.3206	-44.6689
0.04	-63.5434	44.63574	-9.2821	-22.1956	62.80027	-82.1362	19.83465	145.8877	-65.7986	48.8036	196.8992	-43.9155
0.07	-63.5434	44.63574	-9.2521	-20.9094	61.80796	-80.8462	20.91032	145.0308	-64.6678	49.52565	196.2507	-42.8063
0.115	-63.5434	44.63574	-9.2071	-19.009	60.28326	-78.9112	22.50206	143.7081	-62.9931	50.59014	195.2439	-41.1903

The second category, however, included the initial coordinates of the four rigid points at time zero, as well as the sum of their displacement and rotation at each given second of the simulation. Table 3 describes the second data category.

Table 3. Sum of displacement and rotation per rigid point for each second

Second Steps		**0 to 1**		**1 to 2**		**2 to 3**		**3 to 4**		**4 to 5**		**5 to 6**		**6 to 7**		Legend
Rigid points		**Displacement**	**Rotarion**	**Displacement**	**Rotarion**	**Displacement**	**Rotarion**	**Displacement**	**Rotarion**	**Displacement**	**Rotarion**	**Displacement**	**Rotarion**	**Displacement**	**Rotarion**	Displacement Unit = Unknown Rotation Unit = Rad / s
Head	**X**	-10	[]	-20	-0.58	-27.5	0	-35	0	-40	0	[]	0	-40	0	[] = No boundry given
	Y	-15	[]	-30	-0.7	-46.5	0	-57	0	-70	0	-80	0	-110	0	
	Z	4	0.4	8	0.4	8	0.125	8	0.125	8	0.2	8	0.2	8	0.2	
Collar	**X**	-0.4	-0.58	[]	-0.4	[]	0	[]	0	[]	0	-26	0	-54	0	
	Y	-0.6	-0.7	[]	-0.6	-39.4	0	-45	0	[]	0	-50	0	-62	0	
	Z	43	-0.04	86	-0.04	86	-0.1	86	-0.1	86	-0.15	[]	-0.05	[]	0.3	0 = No movement
Torso	**X**	[]	[]	[]	[]	[]	[]	[]	[]	[]	0	[]	0	[]	0	
	Y	[]	[]	[]	[]	[]	[]	[]	[]	[]	0	[]	0	[]	0	
	Z	[]	[]	[]	[]	[]	[]	[]	[]	[]	[]	[]	[]	[]	-0.75	
Hip	**X**	[]	[]	[]	[]	[]	[]	[]	[]	[]	0	[]	0	[]	0	
	Y	[]	[]	[]	[]	[]	[]	[]	[]	[]	0	[]	0	[]	0	
	Z	[]	[]	[]	[]	[]	[]	[]	[]	[]	[]	[]	[]	[]	-0.75	

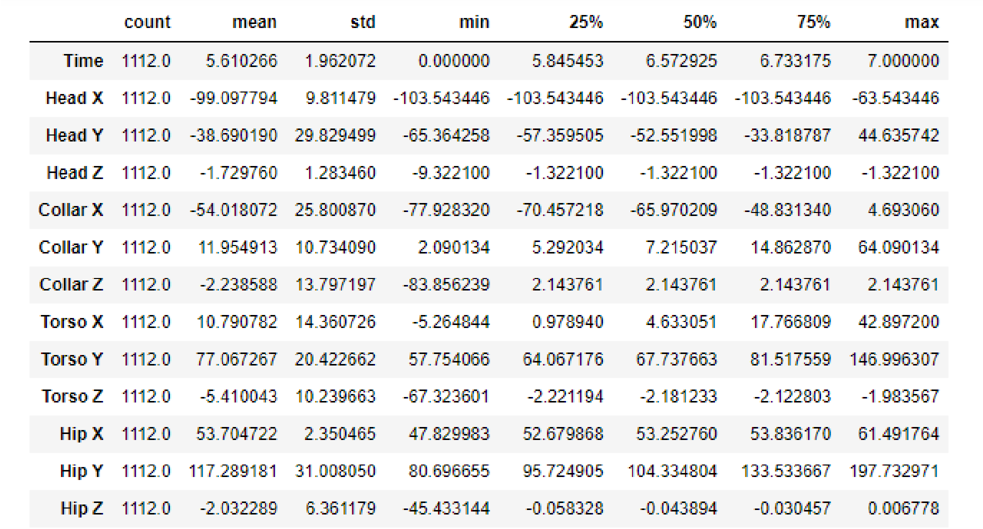
3.2 Data Validation

Data validation is a crucial step in any research process to ensure the accuracy and reliability of the collected data. The validation process involves checking the completeness and correctness of the data, identifying outliers, missing values, and resolving any inconsistencies or discrepancies. This study performs a series of data validation processes to ensure the accuracy and consistency of the collected data. These steps involve using Python to check the statistical properties of each axis, performing a missing-value check and identifying any outliers using scatter plots.

3.3 Validating Data – Step One

To ensure the completeness of the collected data, a thorough data validation process is conducted for each rigid body displacement data in 3D space. This step finds no missing values in the dataset using the Pandas library in Python. Additionally, the statistical properties of the dataset such as count, min, max, etc. are presented in table 4.

Table 4. Statistical analysis of the four rigid body displacement data



	count	mean	std	min	25%	50%	75%	max
Time	1112.0	5.610266	1.962072	0.000000	5.845453	6.572925	6.733175	7.000000
Head X	1112.0	-99.097794	9.811479	-103.543446	-103.543446	-103.543446	-103.543446	-63.543446
Head Y	1112.0	-38.690190	29.829499	-65.364258	-57.359505	-52.551998	-33.818787	44.635742
Head Z	1112.0	-1.729760	1.283460	-9.322100	-1.322100	-1.322100	-1.322100	-1.322100
Collar X	1112.0	-54.018072	25.800870	-77.928320	-70.457218	-65.970209	-48.831340	4.693060
Collar Y	1112.0	11.954913	10.734090	2.090134	5.292034	7.215037	14.862870	64.090134
Collar Z	1112.0	-2.238588	13.797197	-83.856239	2.143761	2.143761	2.143761	2.143761
Torso X	1112.0	10.790782	14.360726	-5.264844	0.978940	4.633051	17.766809	42.897200
Torso Y	1112.0	77.067267	20.422662	57.754066	64.067176	67.737663	81.517559	146.996307
Torso Z	1112.0	-5.410043	10.239663	-67.323601	-2.221194	-2.181233	-2.122803	-1.983567
Hip X	1112.0	53.704722	2.350465	47.829983	52.679868	53.252760	53.836170	61.491764
Hip Y	1112.0	117.289181	31.008050	80.696655	95.724905	104.334804	133.533667	197.732971
Hip Z	1112.0	-2.032289	6.361179	-45.433144	-0.058328	-0.043894	-0.030457	0.006778

3.4 Validating Data – Step Two

Although the statistical properties of the dataset presented in table 4 indicate that there are no outliers in each rigid body displacement data, scatter plots are generated as a visual aid to ensure the completeness of the collected data. Using this technique makes it easier to identify any outliers. Then, each scatter plot is visually inspected to identify any irregularities in the dataset. This process enables the identification and correction of any potential errors or inconsistencies in the data, ensuring the reliability of the results obtained from the Abaqus simulation.

The scatter plot representation of the fetus head movement for each axis through 1112 frames of time equal to 7 seconds is shown in figure 13.

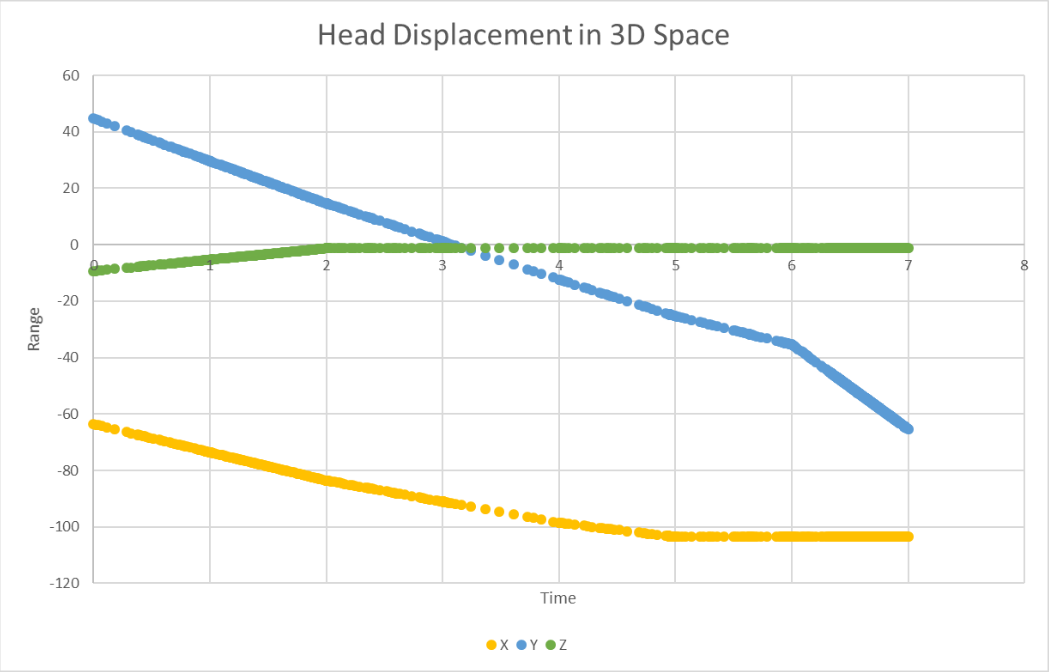


Figure 13. Scatter plot of fetus head movement.

Moving on to other points, figures 14, 15 and 16 illustrate scatter plots of the collar, torso and hip respectively.

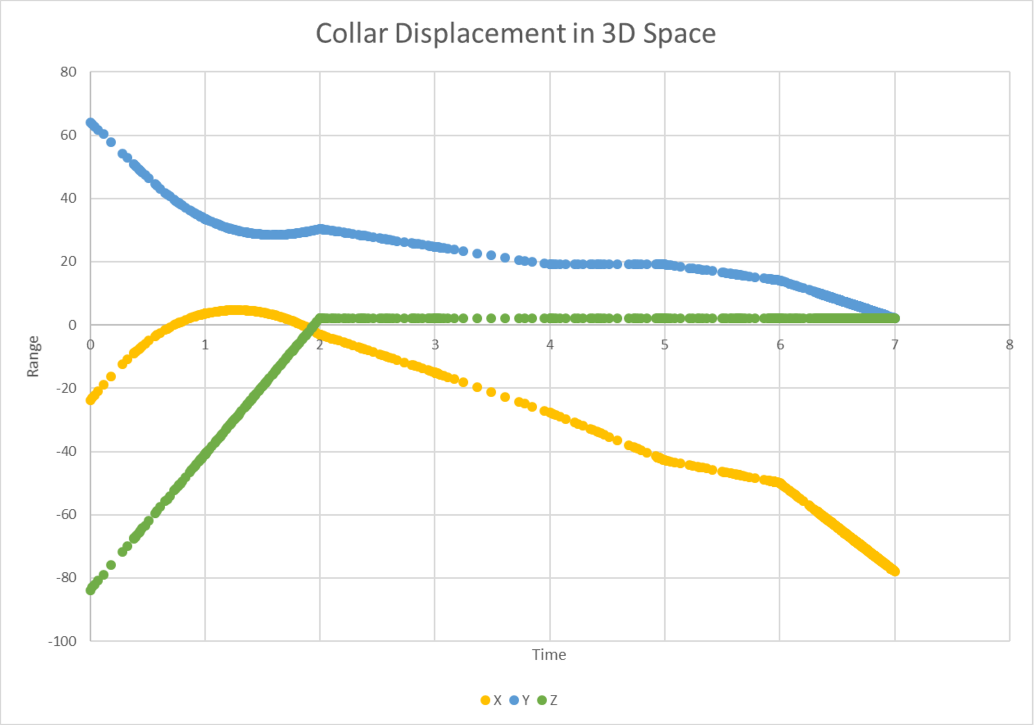


Figure 14. Scatter plot of fetus collar movement based on Abaqus data.

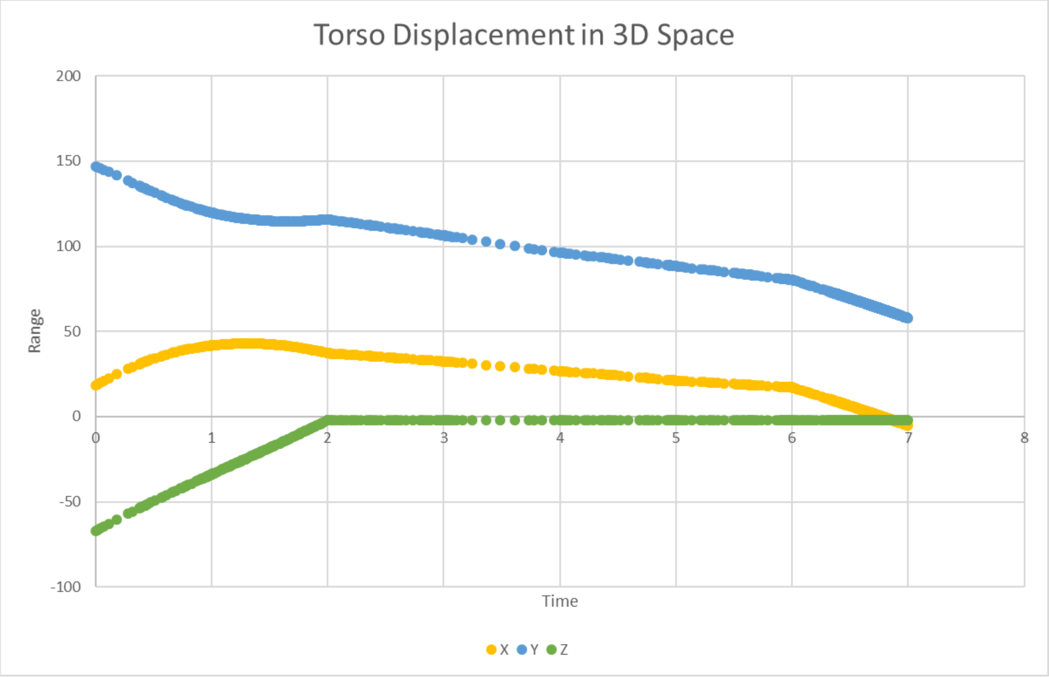


Figure 15. Scatter plot of torso movement based on Abaqus data.

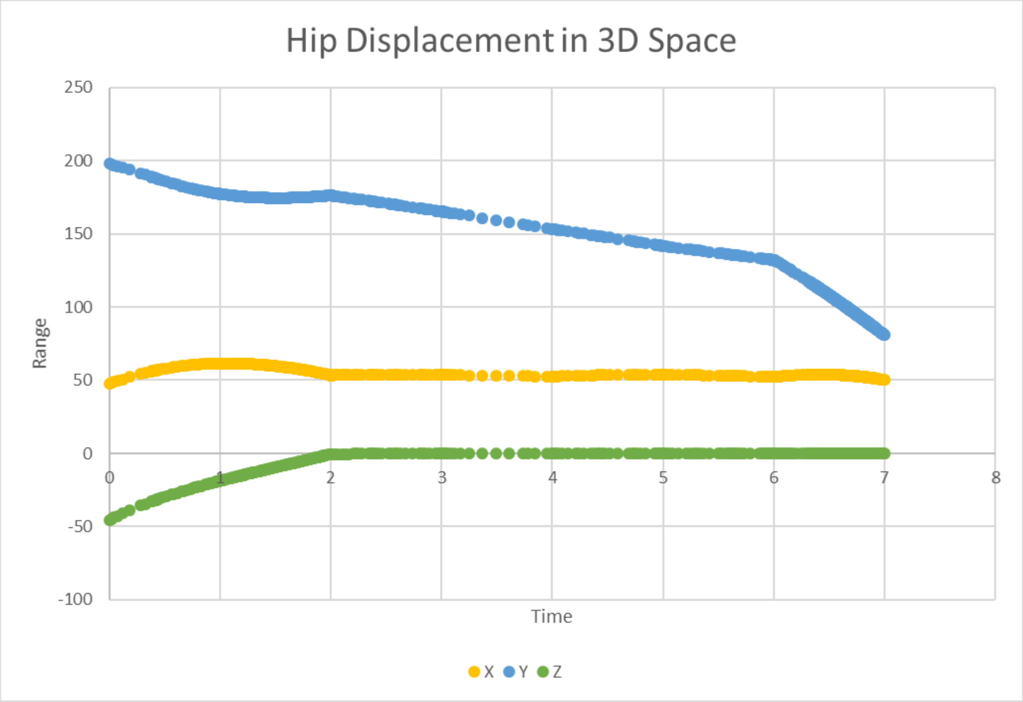


Figure 16. Scatter plot of fetus torso movement based on Abaqus data.

When inspected visually, these scatter plots show no outliers in each of their axes respectively. The continuity of each point of each axis throughout the entire timeframe is highlighted. The scatter plots for each axis are so dense that they almost form a line. However, there are some minor discrete movements between the second 3 and 4, which may have been caused by change of speed or changes in direction by the fetus during the simulation.

OpenSim-Based Skeletal Dynamic Simulation

After obtaining the data related to the displacement of the FE vaginal delivery simulation, the next step is to create a fetal musculoskeletal model and incorporate the displacement data to re-create the FE delivery dynamics. This method converts the onerous FE delivery simulation which used a hollow fetal mesh for an interactive delivery simulation emphasizing the musculoskeletal dynamics. Thus, based on the capabilities of OpenSim [26] introduced in the background chapter, this platform is selected for the purpose of creating the fetal and maternal musculoskeletal anatomies.

In this research, a specific set of OpenSim [26] tools are utilized including the scaling tool, marker placement, and inverse kinematic solutions. The scaling tool is used to scale the size of the fetal skeletal model based on the anthropometric data of the FE fetus. Markers are then generated to govern the motion of musculoskeletal dynamics. Additionally, an inverse kinematic solution is utilized to calculate the joint angles of the musculoskeletal fetal model based on the motion data. By using these tools, an accurate interactive re-creation of FE simulation is achieved.

3.5 Modeling the Fetal and Maternal Anatomies

For the fetal musculoskeletal model, this research utilized an existing published female adult model by Burkhart [42], as shown in figure 17, with minimal modification. The model serves as the maternal anatomy. Key characteristics of this model include both lower and upper extremities modeled. It comes with 108 degrees of freedom in total, 64.17 inches in height, 134.5 lbs. and 114 markers. Additionally, this model did not include any constraints or controllers. These key characteristics, combined with its publication as a female model, make it an ideal candidate for adaptation as a maternal body.

The OpenSim rescaling tool shown in figure 18 is utilized to adjust the size and proportions to create a more precise representation of the fetal skeletal anatomy. Table 5 presents the dimensions utilized in the rescaling process of OpenSim to develop the fetal skeletal model by adjusting the maternal model. These measurements for rescaling are obtained from the Parente, Natal Jorge and Oliveira study [28]. Although the specific measurement presented in their publications gives the dimension of the fetal head at 9.0 cm for Biparietal, due to its significance in the birth process, the measurements from the rest of the FE fetus concerning measurements for the torso, neck and limbs are obtained from the Abaqus output data file.

Table 5. Fetal scaling dimensions based on Parente, Natal Jorge, Oliveira study [28]

Body-Part	**Measurements (cm)**
Biparietal	9.0
Neck	3.15
Humerus	7.91
Radius	6.42
Femur	7.84
Tibia	8.66
Torso	15.9

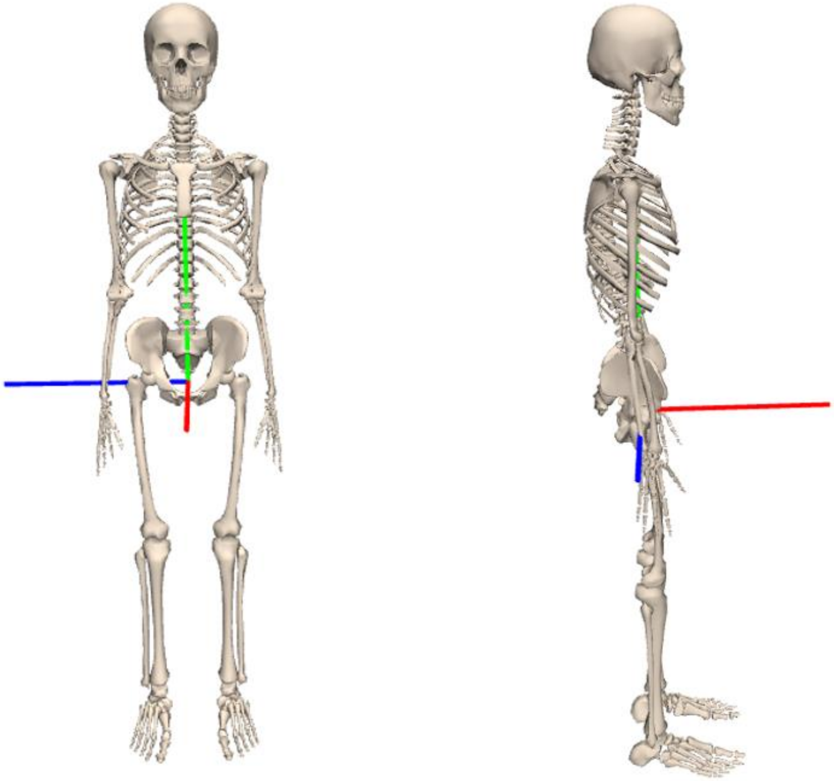
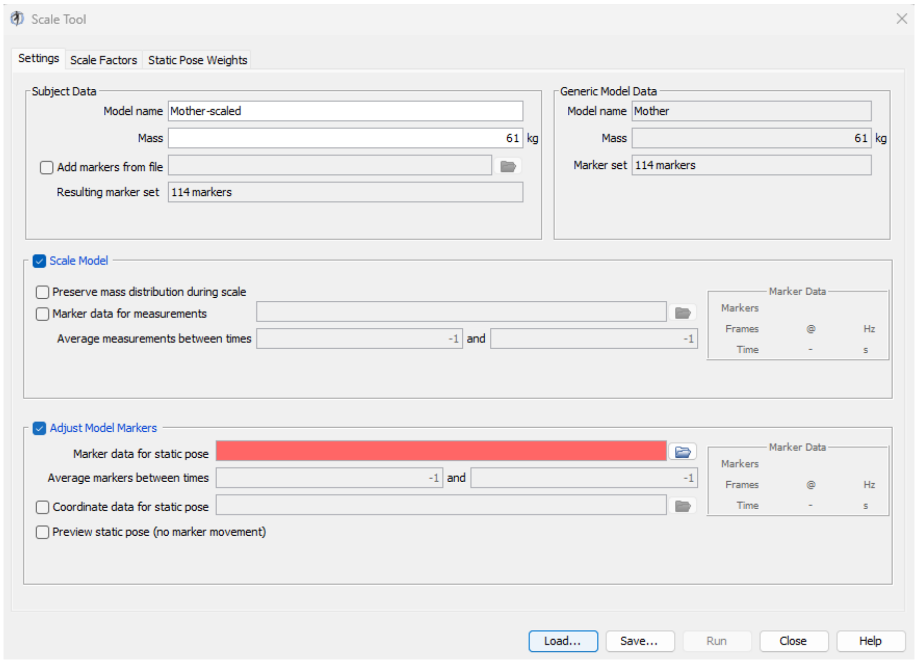


Figure 17. Maternal model.



Subject Data
Model name Mother-scaled
Mass 61 kg
☐ Add markers from file
Resulting marker set 114 markers

Generic Model Data
Model name Mother
Mass 61 kg
Marker set 114 markers

☑ Scale Model
☐ Preserve mass distribution during scale
☐ Marker data for measurements
Average measurements between times -1 and -1
Marker Data
Markers
Frames @ Hz
Time - s

☑ Adjust Model Markers
Marker data for static pose
Average markers between times -1 and -1
☐ Coordinate data for static pose
☐ Preview static pose (no marker movement)
Marker Data
Markers
Frames @ Hz
Time - s

Figure 18. The scale tool of OpenSim used to rescale maternal model to get fetal structure.

The anthropometric measurements obtained in table 5 are consistent with values cited in the literature for the average length of newborns [43]. Using a rescaled version of an already published musculoskeletal anatomy for the purpose of obtaining the fetal structure is a simpler approach compared to building it from scratch.

The process of rescaling the musculoskeletal model of the mother to obtain the fetal structure is a two-stage process. In the first stage, the maternal model is rescaled based on the overall length of the fetus. This is achieved by adding up the length of the skull, neck, torso, femur, and tibia, resulting in an overall length of approximately 45 cm or 17.7 inches. The maternal model has an overall length of 163 cm or 64.17 inches, which resulted in a scaling ratio of approximately 27% to obtain the fetal model. The outcome of stage one rescaling can be seen in figure 19.

Figure 19. Rescaled fetal model compared to maternal anatomy.

Although the resulting fetal anatomy is consistent with the anthropometric length of the FE fetus, there are discrepancies in the actual measurements of limbs and skull due to the maternal model's specific body proportions. Therefore, scaling down with a constant ratio does not give the desired measurements of table 5. Additionally, the resulting fetus has teeth, which is not accurate. To address these issues, a second stage of rescaling is necessary, which involves fixing the skull representation and modifying the limb proportions.

To achieve the second stage of rescaling, the actual length of each limb and the skull of the rescaled fetus is recorded. Then new ratios are calculated for each body part using the measurements in table 5. The second stage is achieved by accessing the OpenSim source file of the rescaled fetal model, allowing direct modification of the ratios and changing the source file used to represent each body part. By fixing the ratios and changing the skull source file to a generic fetal skull, the second stage of rescaling was complete. The resulting model, shown in figure 20, reflects the desired limb and skull measurements.

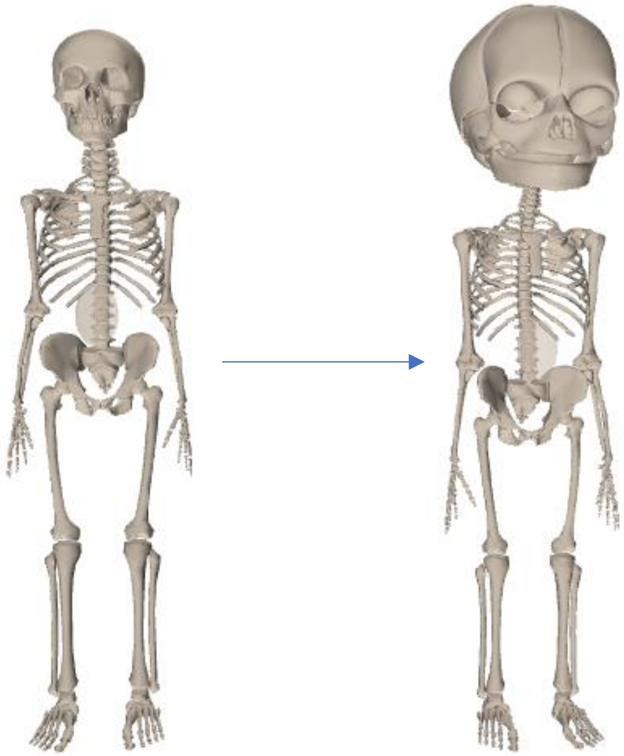


Figure 20. Rescaled fetal anatomy based on table 5 anthropometric data.

The maternal model, along with the rescaled fetal structure shown in figure 21, is used to investigate the mechanics of the labor process and accelerate the FE vaginal delivery simulation.

This approach is a precursor to the development of an interactive, patient-specific obstetric simulator proposed by Audette et al. [22]. The accurate representation of the fetal musculoskeletal system is achieved using OpenSim's rescaling tool, which is critical in understanding the dynamics of birth in terms of the musculoskeletal system.

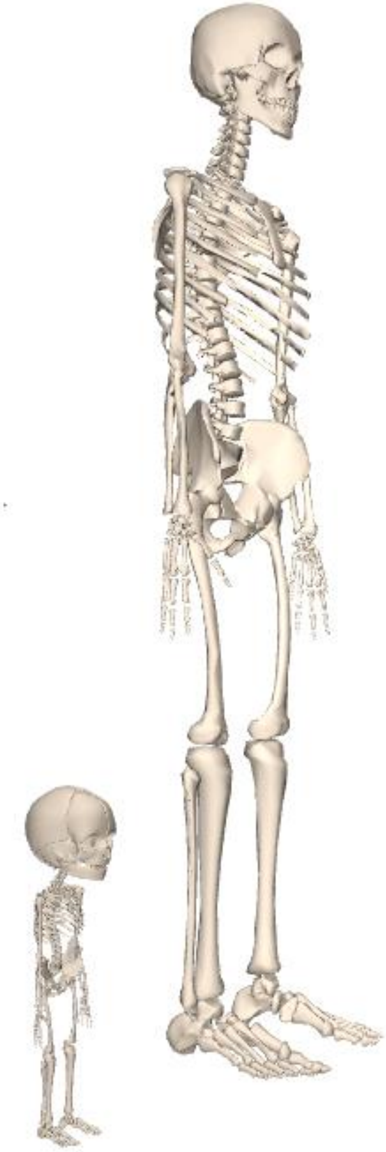


Figure 21. Final fetal and maternal anatomies.

3.6 Marker Placement

Before final adjustments of the data obtained from the FE simulation, it is necessary to define the four rigid body points on the fetal model in terms of markers. The four rigid body points defined in the FE simulation are placed in the center mass of each respective body part mentioned in table 1. A set of four markers are created using the OpenSim marker tool. These markers are placed at the location corresponding to figure 12. Although the skull obscures the head marker, the result of this implementation is depicted in figure 22.

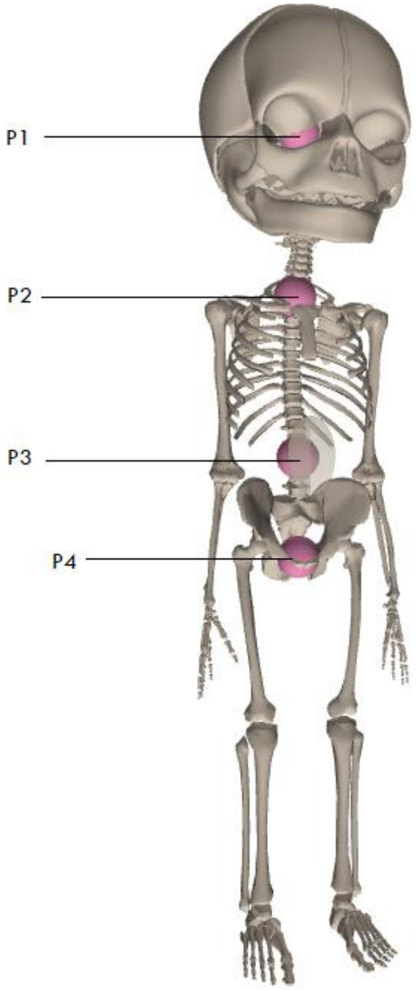


Figure 22. Marker placement position on fetal skeletal model.

3.7 Putting the Model in Fetal Position

The fetal position refers to the characteristic pose of a developing fetus in the uterus, which is typically compact with the back curved, head bowed, and limbs drawn up to the torso [44] as shown in figure 23. However, the position used in the FE simulation differs slightly from this typical fetal position. Parente [27] positioned the fetal model in a more upright pose, with only the limbs bent inward.

Figure 23. Leonardo Da Vinci depiction of fetal position [44].

To replicate this position for the musculoskeletal fetal anatomy a slight bend is introduced to the back portion, while the limbs were fully bent inward, as shown in figure 24.

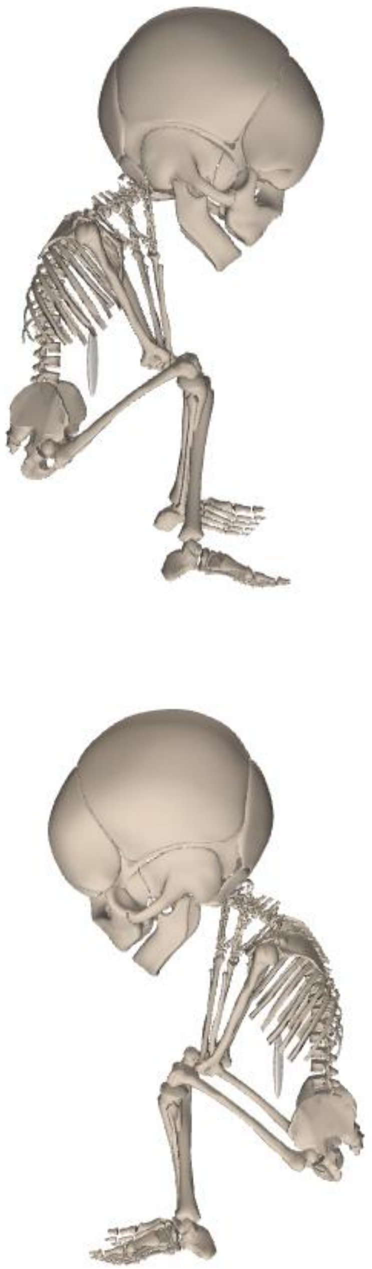


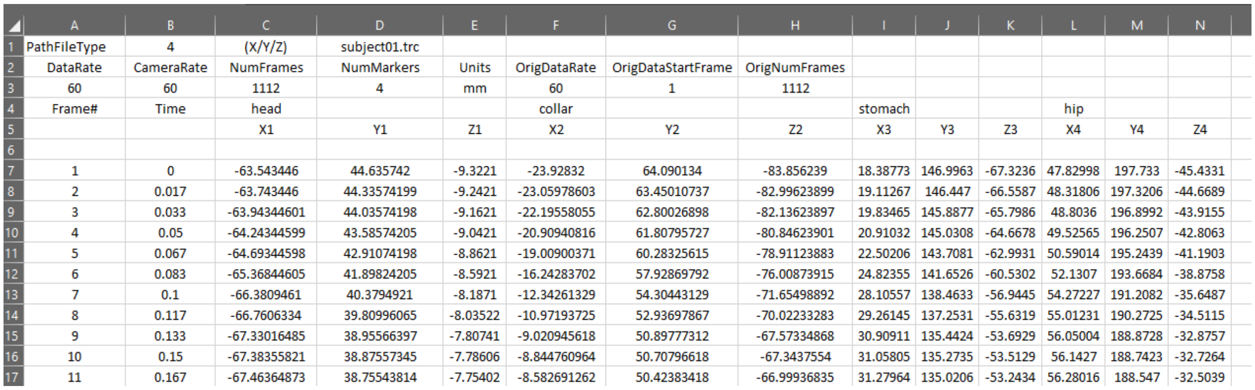
Figure 24. Fetal position of the musculoskeletal anatomy.

3.8 Preparing for Inverse Kinematic Solution

Inverse kinematics is a technique that involves using a set of displacement data to calculate the set of joint angles and rotational data required to achieve that displacement. In biomechanics, inverse kinematics is used to determine the movements of joints and limbs during a specific motion. By using inverse kinematics, it is possible to change displacement data to a set of orientations and rotations, which enables a model to replicate a certain motion [45].

Before using the OpenSim inverse kinematic solution tool, it is crucial to ensure that the excel file containing the displacement data of the four rigid body points is properly formatted and contains the required content. OpenSim requires a specific file format, denoted by the extension ".trc". This file format includes a header that is used by OpenSim to identify the key components of the displacement data. The proper ".trc" file format with a unique identifier header for the converted Excel file is presented in table 6.

Table 6. Proper “.trc” file format of converted displacement data.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	PathFileType	4	(X/Y/Z)	subject01.trc										
2	DataRate	CameraRate	NumFrames	NumMarkers	Units	OrigDataRate	OrigDataStartFrame	OrigNumFrames						
3	60	60	1112	4	mm	60	1	1112						
4	Frame#	Time	head			collar			stomach			hip		
5			X1	Y1	Z1	X2	Y2	Z2	X3	Y3	Z3	X4	Y4	Z4
6														
7	1	0	-63.543446	44.635742	-9.3221	-23.92832	64.090134	-83.856239	18.38773	146.9963	-67.3236	47.82998	197.733	-45.4331
8	2	0.017	-63.743446	44.33574199	-9.2421	-23.05978603	63.45010737	-82.99623899	19.11267	146.447	-66.5587	48.31806	197.3206	-44.6689
9	3	0.033	-63.94344601	44.03574198	-9.1621	-22.19558055	62.80026898	-82.13623897	19.83465	145.8877	-65.7986	48.8036	196.8992	-43.9155
10	4	0.05	-64.24344599	43.58574205	-9.0421	-20.90940816	61.80795727	-80.84623901	20.91032	145.0308	-64.6678	49.52565	196.2507	-42.8063
11	5	0.067	-64.69344598	42.91074198	-8.8621	-19.00900371	60.28325615	-78.91123883	22.50206	143.7081	-62.9931	50.59014	195.2439	-41.1903
12	6	0.083	-65.36844605	41.89824205	-8.5921	-16.24283702	57.92869792	-76.00873915	24.82355	141.6526	-60.5302	52.1307	193.6684	-38.8758
13	7	0.1	-66.3809461	40.3794921	-8.1871	-12.34261329	54.30443129	-71.65498892	28.10557	138.4633	-56.9445	54.27227	191.2082	-35.6487
14	8	0.117	-66.7606334	39.80996065	-8.03522	-10.97193725	52.93697867	-70.02233283	29.26145	137.2531	-55.6319	55.01231	190.2725	-34.5115
15	9	0.133	-67.33016485	38.95566397	-7.80741	-9.020945618	50.89777312	-67.57334868	30.90911	135.4424	-53.6929	56.05004	188.8728	-32.8757
16	10	0.15	-67.38355821	38.87557345	-7.78606	-8.844760964	50.70796618	-67.3437554	31.05805	135.2735	-53.5129	56.1427	188.7423	-32.7264
17	11	0.167	-67.46364873	38.75543814	-7.75402	-8.582691262	50.42383418	-66.99936835	31.27964	135.0206	-53.2434	56.28016	188.547	-32.5039

The markers introduced to the fetal model are responsible for governing the fetal motion based on the input data. As these markers are set to specific body parts, most joints and limbs do not have an active role in replicating the FE delivery motion. Thus, it is crucial to prevent unrelated joints and limbs from affecting the inverse kinematic outcome, as they can result in unnatural movements.

To ensure accurate calculations, only specific joints were designated to move in the fetal position, while other body parts are locked in place to only move in union with the motion-driven portion of the anatomy. This is achieved by configuring the relevant joints in the OpenSim coordinates section to move respective to their degree of freedom, while the rest of the body is fixed in the pose. Table 7 provides the list of moving joints in the fetal skeletal model.

Note: The joint names and their corresponding rotational degrees of movement remain unchanged in reference to the maternal model.

Table 7. List of moving joints

Body Part	**Range of Motion (degree)**
Pelvis Rotation	360
Pelvis Tilt	360
Pelvis List	360
L5 - FE	180
L4 - FE	180
L3 - FE	180
Head	180
Neck	180

3.9 Inverse Kinematic Solution

"Kinematics is the study of motion without considering the forces and moments that produce that motion"[45]. Inverse kinematics is a technique used to calculate the joint angles

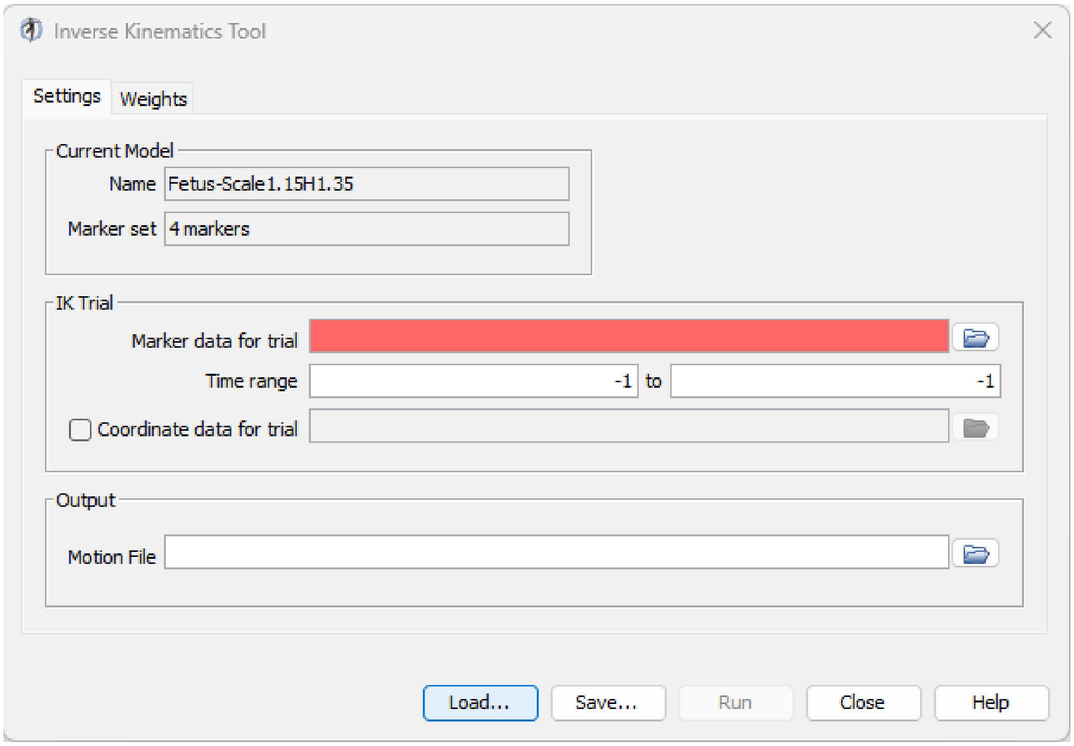
needed for a model to achieve a desired motion or position. It is commonly used in animation to control the movement of limbs and joints. In the context of biomechanics, inverse kinematics can be used to calculate the joint angles of a human model based on motion capture data. In this way, inverse kinematics can help researchers understand how different movements are generated by the musculoskeletal system. It should be noted that to use inverse-kinematic at least three non-linear marker data are necessary. In the case of this study, 4 markers are placed on the body.

OpenSim's IK solution solves for the weighted least squares equation shown in equation 1. q is the vector of coordinates being solved, x_i^{exp} is the experimental position of marker i, $x_i(q)$ is the position of the corresponding model marker, q_j^{exp} is the experimental value for coordinate j [46].

$$Min\left[\sum w_i || x_i^{exp} - x_i(q)||^2 + \sum w_i\left(q_j^{exp} - q_j\right)^2 \right]$$

Equation 1. Weighted Least Squares Equation [46].

The IK tool in OpenSim can be seen in figure 25. It requires a specific input file, which is the ".trc" data file that is now generated and properly formatted. Because the OpenSim IK solution solves for weighted least-squares, in this tool users can put any value for the weights of each marker. These weights make sure that the emphasis of rescaling goes to the marker with the highest weight and downward to the least weighted marker. In this study, however, all weights for four markers are set to 1. this ensures that no marker gets higher priority and emphasis when rescaling in performing.



Name Fetus-Scale1.15H1.35

Marker set 4 markers

Time range -1 to -1

Figure 25. IK tool of OpenSim.

The input data file is then fed into the IK tool, and the IK is initiated in OpenSim. During the IK calculation, it is evident that the model is starting to move in 3D space. After finalizing the IK calculations which lasted about 5 minutes and may vary depending on the hardware used, the fetal model now has a motion associated with IK calculations. The motion can be saved as a separate output file. OpenSim uses a “.mot” extension to save such output. This file contains the actual motion of the fetal skeletal model based on the input data in terms of orientation of each of the four markers in 3D space. This motion can be viewed on the OpenSim GUI.

Although the three non-linear marker conditions are met with four markers on the fetal model, they tend to become almost linear in certain poses during the simulation, which causes errors in the inverse kinematics calculation and results in incorrect fetal position and orientation in space. To address this issue, two methods have been implemented. First, the markers are slightly offset from each other, reducing their linearity in problematic frames. Second, after the IK calculation is complete, the motion file is analyzed, and frames with incorrect data are manually corrected. This later correction method is done comparing the sum of the orientations calculated for each second of the simulation to the data presented in table 2.

The results of the steps taken so far are aimed to re-create the FE vaginal delivery with emphasis on musculoskeletal dynamics. The final fetal musculoskeletal motion associated with this step is discussed in detail in the results section.

3.10 Visual Enhancement of the Simulation

So far, this study has focused on modeling and simulating the fetal and maternal structures using the open-source platform OpenSim based on the FE delivery simulation, with the ultimate goal of creating an interactive VR, patient-specific, haptic driven obstetrics simulator. While validity in terms of skeletal structure is important, it is equally crucial that the models look realistic for obstetricians to train on, enhancing their skills. Therefore, in this part of the methodology, the steps taken to enhance the visualization of the fetal musculoskeletal model with skinning are presented to deliver a natural, human-looking fetal model. The skinning process is conducted using two separate approaches.

First, the study provides an overview and application of the BASH model (Biomechanical Animated Skinned Model), which is a unique tool that uses the SCAPE framework and converts the OpenSim musculoskeletal model to a skinned human model.

Second, the study utilizes a manual skinning technique with Blender using a generic model to enhance the realism of the fetal skeletal system. These two approaches are reviewed in detail, and their respective results are discussed in the result section.

3.11 Biomechanical Animated Skinned Model – BASH

The Biomechanical Animated Skinned Human (BASH) model is proposed as a method to animate 3D human surface models for biomechanical analysis [40]. It provides an animated skinned visualization of a musculoskeletal model defined in the OpenSim format without requiring any additional data. The BASH model uses the SCAPE model to yield a natural human appearance and realistic soft tissue deformations. Furthermore, muscle activity is highlighted on the surface which enables dynamic analysis. Figure 26 illustrates the BASH model methodology.

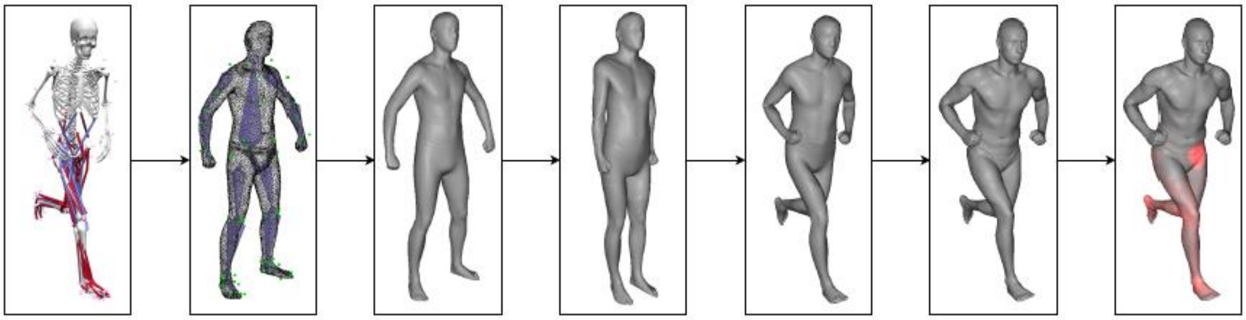


Figure 26. BASH model methodology from Schleicher et al. [40].

SCAPE (Shape Completion and Animation of People), the platform that the BASH model is built upon is a data-driven method for building a unified model of human shape that can produce dense full-body meshes and capture details such as muscle deformations of the body in different poses [47]. It learns separate models of body deformation for changes in pose and differences in body shape between humans, and it allows proper deformation scaling. SCAPE can be used for partial view completion and producing a full 3D animation of a moving person from marker motion capture data, and it allows for variation of the individual body shape, making it possible to create realistic shape completions and dense 3D animations for people whose exact body shape is not included in any of the available data sources [47]. Figure 27 shows the SCAPE method.

Figure 27. SCAPE, Shape Completion and Animation of People from Anguelov et al. [47].

The BASH model presumably has a unique position in this research for two main reasons. First, it utilizes the OpenSim model as input, which is significant because it eliminates the need for other methods of integrating OpenSim models into different platforms for skin enhancement purposes. Second, it uses the IK output file “.mot”, which is the marker movements data, to animate the skinned model. This approach is highly valuable as it ensures realistic and measured skin presentation in the resulting 3D animation with visual enhancement of the skeletal model [40]. The combination of these two unique attributes of the BASH model makes it a highly interesting and efficient tool for generating realistic animations of human movement in biomechanical simulations.

3.12 Steps Taken to Implement Bash Model

The BASH model's source code can be easily accessed through GitHub, but it does not include the OpenSim model or any other required input files such as the IK motion file. The BASH model is published with the OpenSim model "runMaD," which is also freely available on the author's OpenSim page. To use the BASH model, one must first build the repository and obtain the executable and then incorporate the runMaD model and its corresponding motion file as arguments to run the program successfully. The outcome of this experiment is demonstrated in figure 28.

Figure 28. runMaD model experiment with BASH model.

After confirming that the BASH model is working correctly with the runMaD model, the fetal skeletal model and its IK motion file are introduced into the BASH model. However, the resulting output visualization differed from what is expected. Despite trying various configurations of the fetal skeletal model ranging from changing the markers, using a re-scaled runMaD model or changing the parameters of the BASH script, no satisfactory result is achieved. These results are presented in the next chapter.

The first attempt to enhance the visualization of the fetal skeletal model using the BASH model resulted in no satisfactory output. However, a different approach is chosen to enhance the visualization of the fetal model using Blender.

The Blender software is an open-source 3D modeling and animation software package [41]. Blender is chosen for the second attempt.

3.13 Manual Enhancement via Blender

A complete fetal model as shown in figure 29 with all anatomical features, including the skeleton, organs, veins, placenta, umbilical cord, etc., is chosen to enhance the visualization of the interactive simulation.

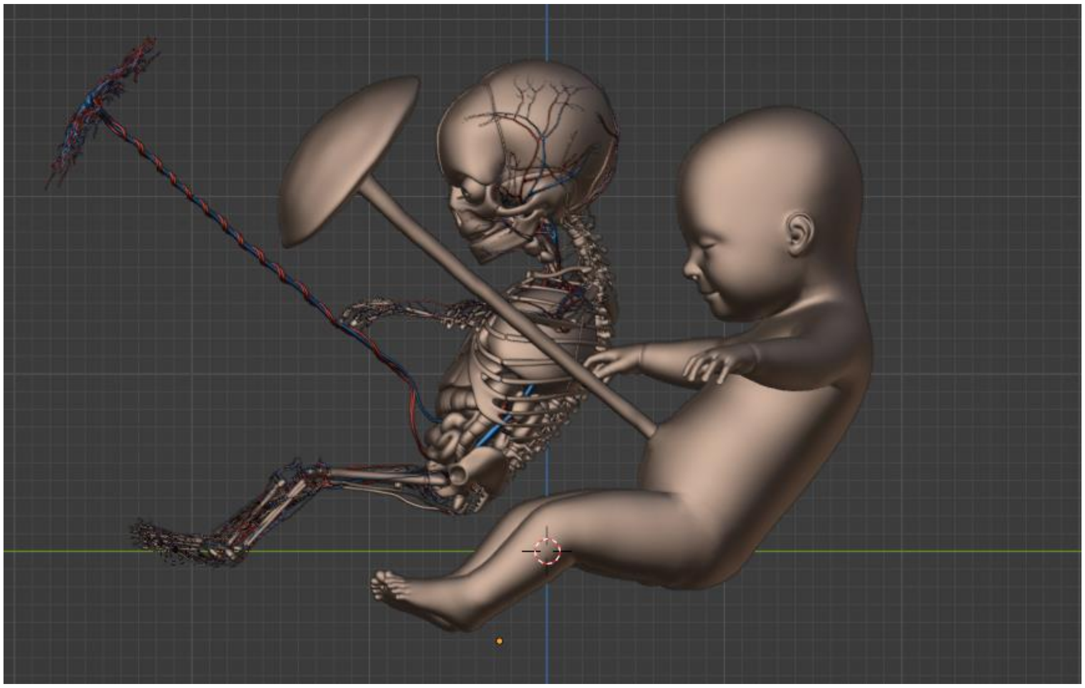


Figure 29. Generic full-term fetus, chosen for skinning.

The following steps are taken to re-create this process with fetal skin.

- Create a rig to act as the animation foundation governing the fetal motion in the Blender environment.
- Assign fetal mesh to the rig, with proper weighting for unified movement.
- Re-create the FE delivery simulation in Blender via animation with one-to-one projection.

In the first step, the Blender rigging tool is utilized to create a suitable rig acting as the structure for the fetal mesh. Then the rig is put into the fetal position. Figure 30 illustrates the pose.

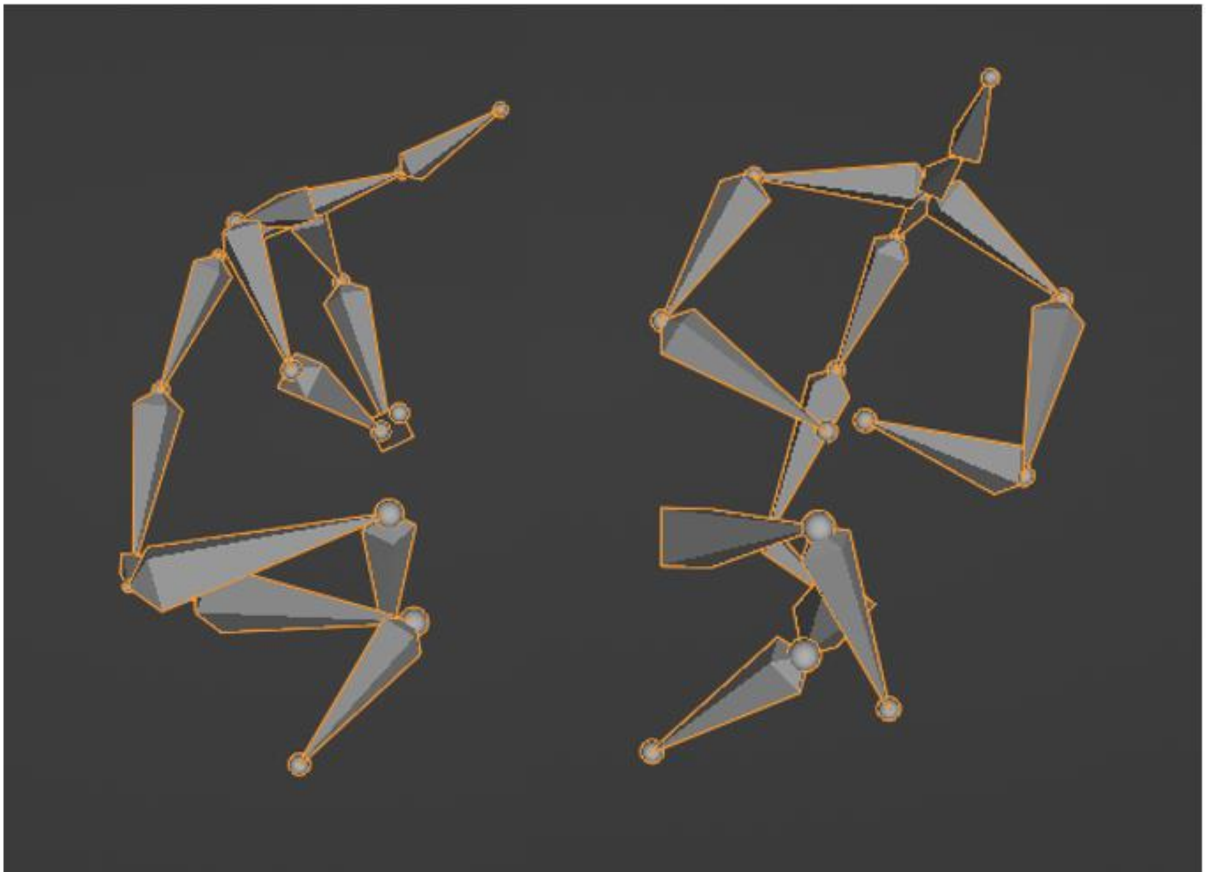


Figure 30. Blender rigging structure in fetal position.

Then, the surface mesh of the fetal skin is added to the rig structure. Figure 31 shows the result of this coupling. By assigning the fetal mesh, the rig and the mesh both are posed in the fetal position.

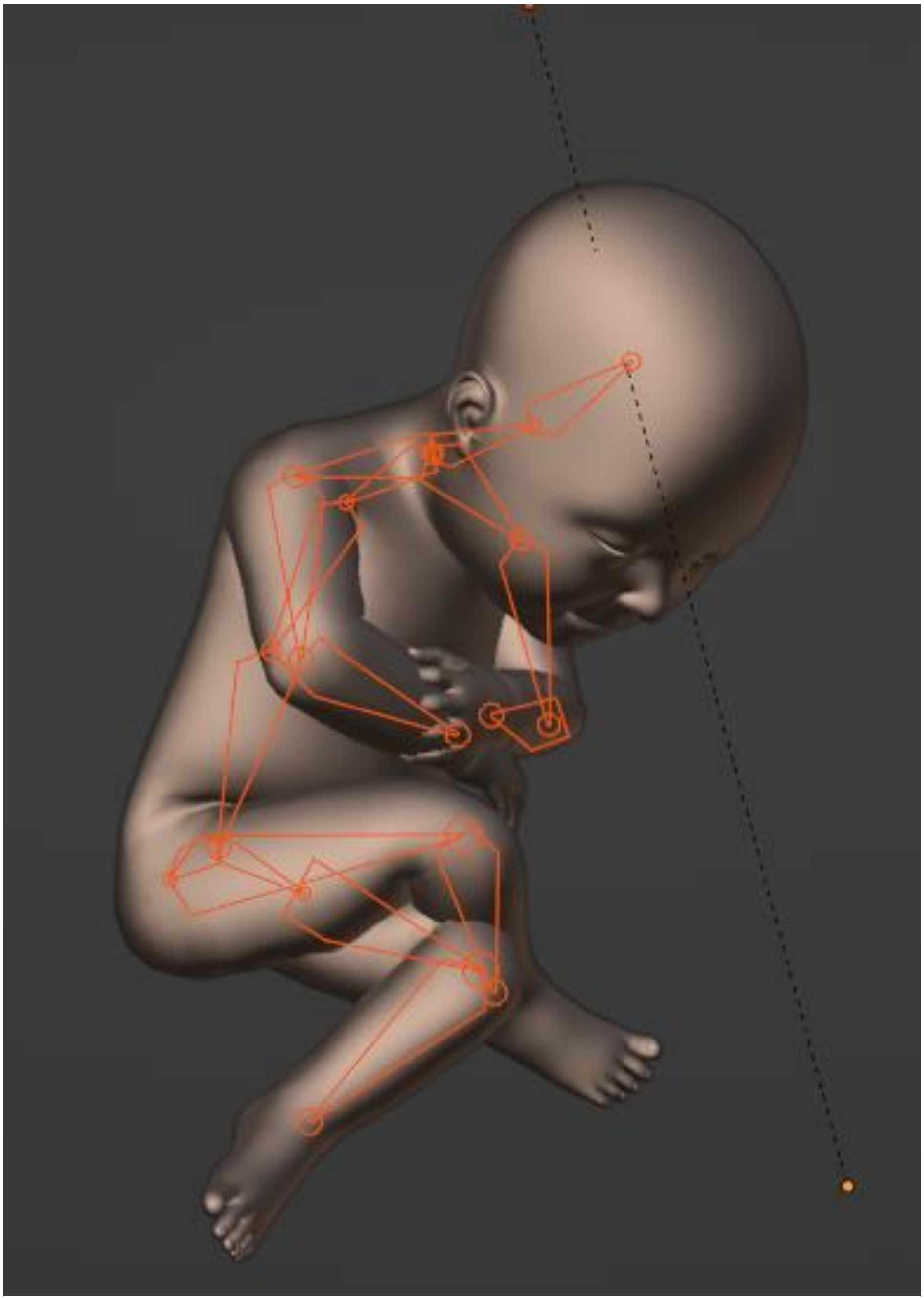


Figure 31. Addition of fetal mesh to rig structure in fetal position.

Finally, the motion profile of the OpenSim musculoskeletal dynamic simulation is put in the background to create a one-to-one animation using the new fetal skin and its rig. A sequence of movements based on the background movement are introduced to the rig and accompanied fetal mesh to reconstruct the interactive simulation of the OpenSim; the resulting render is a one-to-one animation of the musculoskeletal dynamic simulation emphasizing enhanced visualization of the fetal model using Blender. The result of this approach is presented in the results section.

3.14 Conclusion

In conclusion, the methodology section discussed the steps taken to convert an onerous finite element vaginal delivery simulation to an interactive, patient-specific musculoskeletal simulation using OpenSim. The FE delivery simulation used a static pelvis bone to represent the maternal model and a hollow tetrahedral fetal mesh with its anthropometric measurements representing the fetus.

The ultimate goal of this research is to build upon a novel template proposed by Audette et al. [22] to develop an interactive, patient-specific, VR-based, haptic driven obstetrics simulator. The core component of this proposed pipeline is the development of a musculoskeletal model of Parente, Natal Jorge, Oliveira fetal anatomy. By utilizing the steps taken in the methodology section and using OpenSim [26], this study builds upon this core component.

The steps taken in the methodology are presented in the order in which the study utilized them. The result of each step is discussed; graphs and tables are provided to better describe the process. The outcome of the OpenSim implementation is discussed in the result section, being the goal of this study. The methodology section also presented the efforts to utilize the BASH model for enhancing the visualization of fetal movements during childbirth, which then led to the use of Blender to manually skin the fetal skeletal anatomy to enhance its visualization. The resulting render in Blender and the result of the BASH model are discussed in the next chapter.

CHAPTER 4

RESULTS

In this section, a review of the steps taken in the methodology section is presented. The result and limitation of each step is discussed. This section covers the steps taken to convert an onerous finite element birthing simulation to an interactive, patient-specific delivery simulation. Moreover, this section reviews and presents the result of two approaches aimed at enhancing the visual representation of the musculoskeletal model developed for fetal anatomy.

Revisiting the steps taken in the methodology section, first, relevant data is extracted from the Abaqus [31] output file. Second, a validation process is performed on the dataset to ensure that it does not contain missing values. Additionally, scatter plots are generated to check for outliers. The validation process showed that the dataset has no missing values nor any outliers. Moreover, a published female adult musculoskeletal anatomy is selected to represent the maternal anatomy.

Third, the OpenSim [26] scaling tool is utilized to rescale the maternal model based on a constant ratio of 27% to develop the fetal anatomy. The OpenSim fetal source file is then configured to change the representation of the skull and measure the limbs and skull to meet the anthropometric data provided by table 1. Finally, by placing four markers on the OpenSim fetal skeletal structure, representing the four rigid body points of FE fetal, and invoking the inverse kinematics tool of OpenSim, necessary calculations are made to re-create the musculoskeletal dynamics of the FE vaginal delivery simulation.

Additionally, the BASH model is utilized for skinning the fetal anatomy developed in OpenSim [26]. A key feature of the BASH model is that it uses the OpenSim model as input and uses the motion file associated with IK calculations to generate a skinned animated model. However, all the attempts made to incorporate the fetal skeletal model into the BASH model did not produce satisfactory results. These results are discussed later in this chapter. Despite this, another approach is made to generate skinning using a generic full-term model in Blender, which produced satisfactory results.

The primary focus of the result section is the key objective of this research: the Inverse Kinematic (IK) results of the musculoskeletal modeling via OpenSim. The IK results calculated the necessary rotational data required for the four markers to govern the movement of the fetus, allowing for the conversion of FE birthing to an interactive simulation, while the visual enhancement aimed to provide a more realistic representation of fetal movements during childbirth.

4.1 Data and Validation Results

The data is extracted from the Abaqus [31] output file which is generated as the outcome of the FE birthing simulation. It contained displacement data of 4 rigid body points that are placed on the FE fetus and in the center mass of the head, collar, torso and hip respectively. These 4 points governed the movement of the FE fetus passing through the birth canal. These displacement data are then moved to an Excel file which now has the displacement of each of the 4 rigid points in 3D space through 7 seconds or 1112 frames. Then the data acquired is validated in terms of missing data, outliers and their statistical properties. No inconsistency or missing value is found. Also, no outlier data is detected. Scatter plots are generated to present the validity results as shown in figure 32.

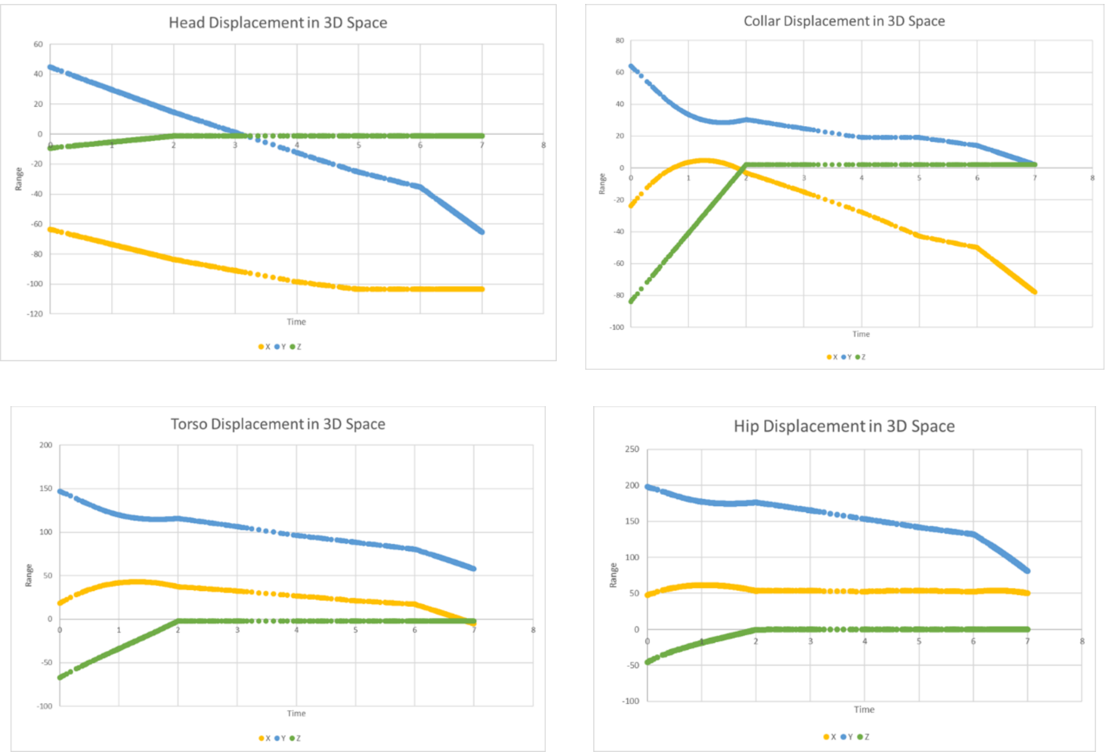


Figure 32. Scatter plots generated to show validity in terms of missing values, outliers.

However, a limitation in this process is in terms of accessibility to the Abaqus [31] software. The freely available version of Abaqus is the learning edition which is capable of opening or simulating up to 1000 nodes. The FE simulation output data file had more nodes than the limited package could open. Thus, to get access to imbedded data, the Mechanical Engineering Department of Old Dominion University provided access to the commercial edition of Abaqus.

4.2 OpenSim Implementation Results

The implementation of fetal and maternal skeletal modeling using OpenSim consisted of multiple steps. The use of a verified full adult model allowed for the detailed representation of the maternal model compared to use of a static pelvis. Additionally, rescaling the maternal model in two steps resulted in development of the fetal anatomy measured based on the anthropometric data of the FE fetus, and with necessary modification, correct representation of fetal anatomy is ensured. Four markers are introduced to the rescaled fetus to govern the fetal movement based on the IK calculations.

The OpenSim IK tool is utilized to re-create fetal movement based on the input displacement data, animating the fetal model through space. The IK minimum marker condition is met but due to the specific pose of the fetal anatomy in some frames the 4 attached markers are put into a linear pattern forcing the IK calculation to produce an error requiring a two-step intervention to get a clean simulation. The results of the OpenSim implementation are presented in this section, including the accuracy of the rescaled model, the validity of the motion tracking markers, and the effectiveness of the IK technique in producing realistic simulations of the birthing process.

The IK calculations took about 5 minutes, and during these calculations the fetal anatomy started to move in the background accordingly. The final motion file came with some erroneous data points due to linear marker data. After the incorrect data points were manually corrected the final result provided a satisfactory outcome. The fetal movement visually matched the movement of the FE simulation. Figure 33 provides snapshots of this step, comparing the OpenSim IK results and the initial FE birthing simulation.

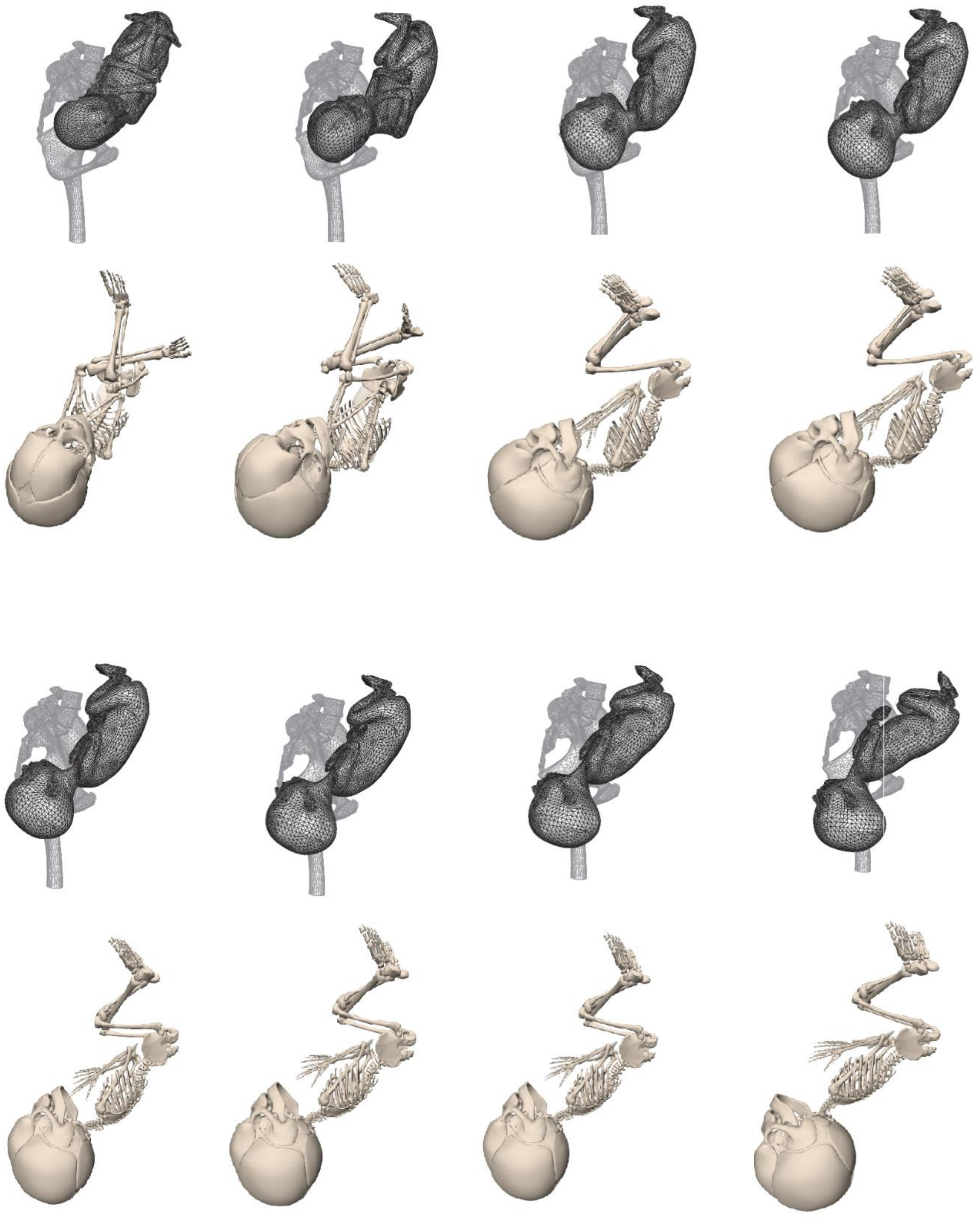


Figure 33. IK Result snapshots compared to FE simulation.

The comparison between the Inverse Kinematic (IK) results and the FE delivery fetal movements showed matching between the movements of the OpenSim model and the FE model. Visually matching the movements is one way of evaluating the outcome. However, further statistical analysis was necessary to make sure the results were within the acceptable margins.

4.3 Evaluating the OpenSim IK Results

To make sure the accuracy of the IK results matches with the FE simulation outcome, the following steps are taken:

- Compare FE displacement plot with IK output marker movement,
- Calculate the correlation coefficient between the two output axes,
- Calculate the root-mean-square deviation.

OpenSim emphasizes the pelvis as the central element in biomechanical analysis of any human movement. The way OpenSim creates its anatomies is around the pelvis. The body is structured around the pelvic bone, and wherever the pelvis goes the rest of the body follows accordingly. On the contrary, the FE study is centered around the head. The pre-defined displacements are introduced for the head to follow a specific path and then the rest of the body follows the same movement as the head does.

Although it seems contrary, the FE study has a rigid body point in the hip area, where a marker is put and tracked in the same place for the OpenSim fetal representation. Thus, the pelvis movement of both bodies are compared in this analytical study.

Figure 34 provides two plots, one from the displacement data related to the pelvis rigid body point of the FE study and the other represents the IK hip marker displacement data.

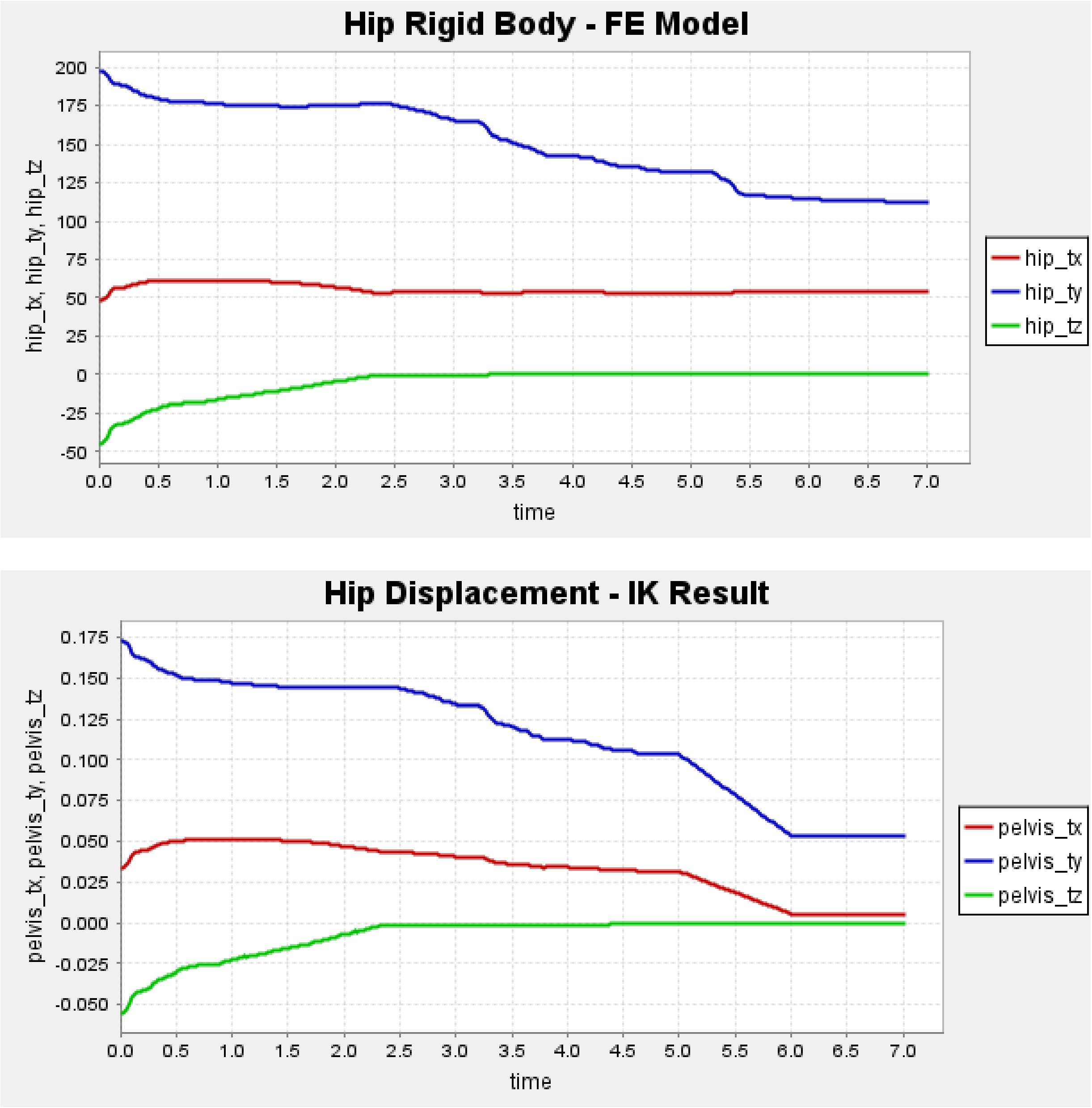


Figure 34. Displacement comparison between FE pelvis and OpenSim hip IK movements through 3D space.

The X, Y and Z axes represent fetal movement on the standard engineering coordinate system with X axis forward, Y axis up and Z axis on the right. The comparison between the two graphs represents very similar pelvis movements in 3D space. Although the FE pelvis graph shows a dip in seconds 5 to 6, the same dip is emphasized in the IK result. To make sure this emphasized dip is not a concern, the correlation coefficient of the 3D axes is also calculated and compared to show the degree of relationship between each axis.

Equation 2 is used to calculate the correlation coefficient between each axis of the pelvis rigid body point displacement and IK hip displacement data.

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2(y_i - \bar{y})^2}}$$

r = correlation coefficient

x_i = values of the x-axis

$\bar{x}$ = mean of values of the x-axis

y_i = values of the y-axis

$\bar{y}$ = mean of values of the y-axis

Equation 2. Correlation coefficient formula.

The correlation coefficients of each axis between the two displacements of pelvis and hip are 58%, 97% and 99% for X, Y and Z axes, respectively. Although the correlation coefficient shows very similar values between the Y and Z axes, the Root-Mean-Squared Deviation (RMSD) of the output displacement data of the pelvis rigid body and hip marker for each axis is also calculated. The RMSD formula is presented in equation 3.

$$RMSD = \sqrt{\frac{\sum_1^n (x_i - \bar{x})^2}{n}}$$

Equation 3. Root-Mean-Squared Error (RMSD) formula.

The RMSD results are 0.06, 0.15 and 0.01 for the X, Y and Z axes, respectively. Although the correlation coefficient of the Y axis showed a 97% positive relationship, the RMSD showed the Y axis has more error. Further investigation showed the emphasized dip between seconds 5 and 6 is the reason for this increased error. The RMSD shows that the result of IK calculations and the FE delivery simulation are indeed very close; thus, the inverse-kinematic solution results are validated. This in-turn validated the OpenSim interactive simulation.

In the FE delivery simulation, a static pelvis represents the maternal anatomy and simulates the passage of the fetus through the birth canal. To expand upon this study and make it more interactive, a full anatomical model of the mother is now incorporated into the simulation.

Although the maternal model has no modification regarding its anthropometric measurements, its combination with the fetal model allows visualization of the birthing process in more detail and helps practitioners investigate the effects of different maternal poses on the delivery process.

By varying the maternal model's pose, practitioners can gain a better understanding of how maternal positioning affects the birthing process or help train on maneuvers needed in complicated scenarios. These modifications are possible by manually adjusting and rotating the maternal model over the X, Y and Z axes respectively. The amount of rotation over each axis is dependent on the target pose. The fetal model also needs to incorporate the same rotations to make sure its relative position with the maternal body stays the same. Figure 35 details this incorporation.

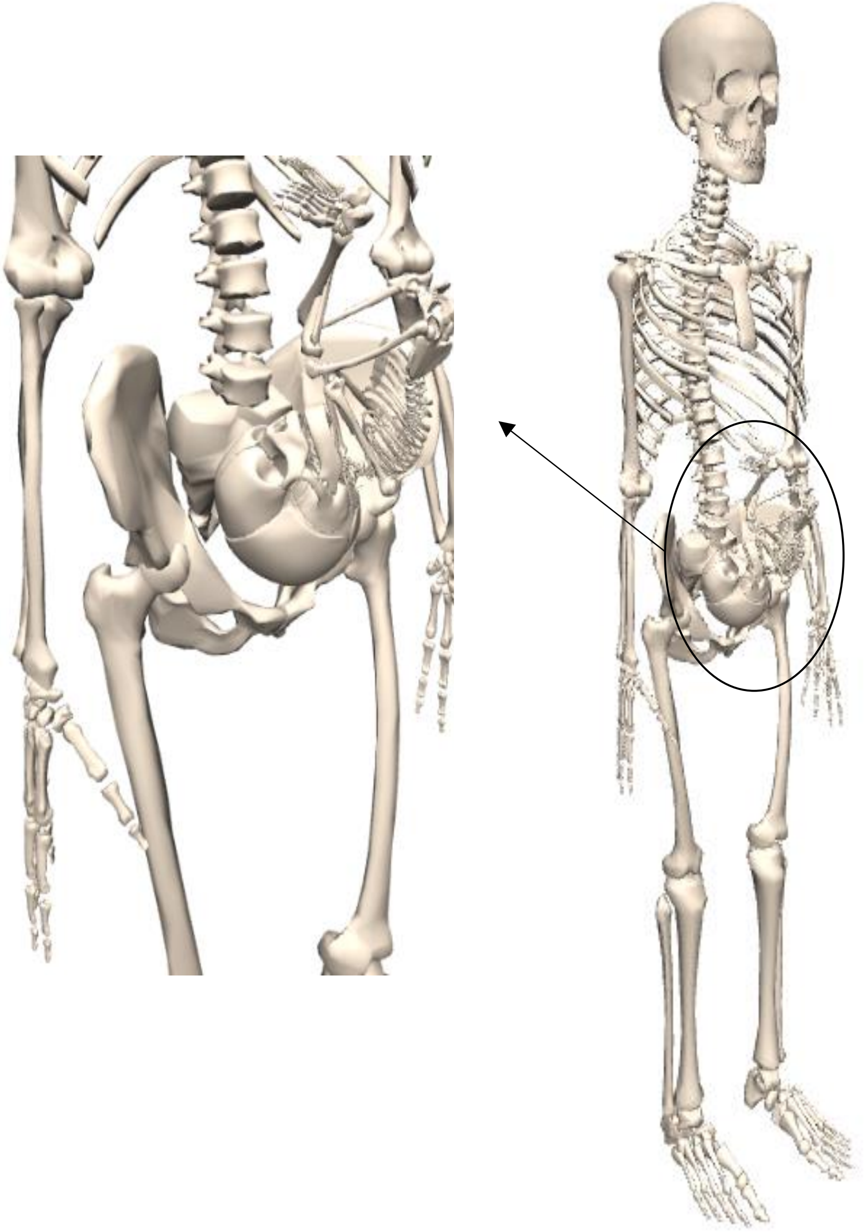


Figure 35. Fetal and maternal model in the same scene.

The integration of the maternal and fetal models resulted in a comprehensive simulation that accurately depicts the female anatomy during pregnancy. The realistic fetal motion and its interaction with the maternal anatomy created a detailed birthing scenario, providing a strong foundation for the development of a patient-specific training platform in the future. By utilizing this platform, practitioners will be able to better understand the complexities of childbirth and improve their skills in a safe and controlled environment. Figure 36 illustrates the birth of a fetus accompanied by a maternal model.

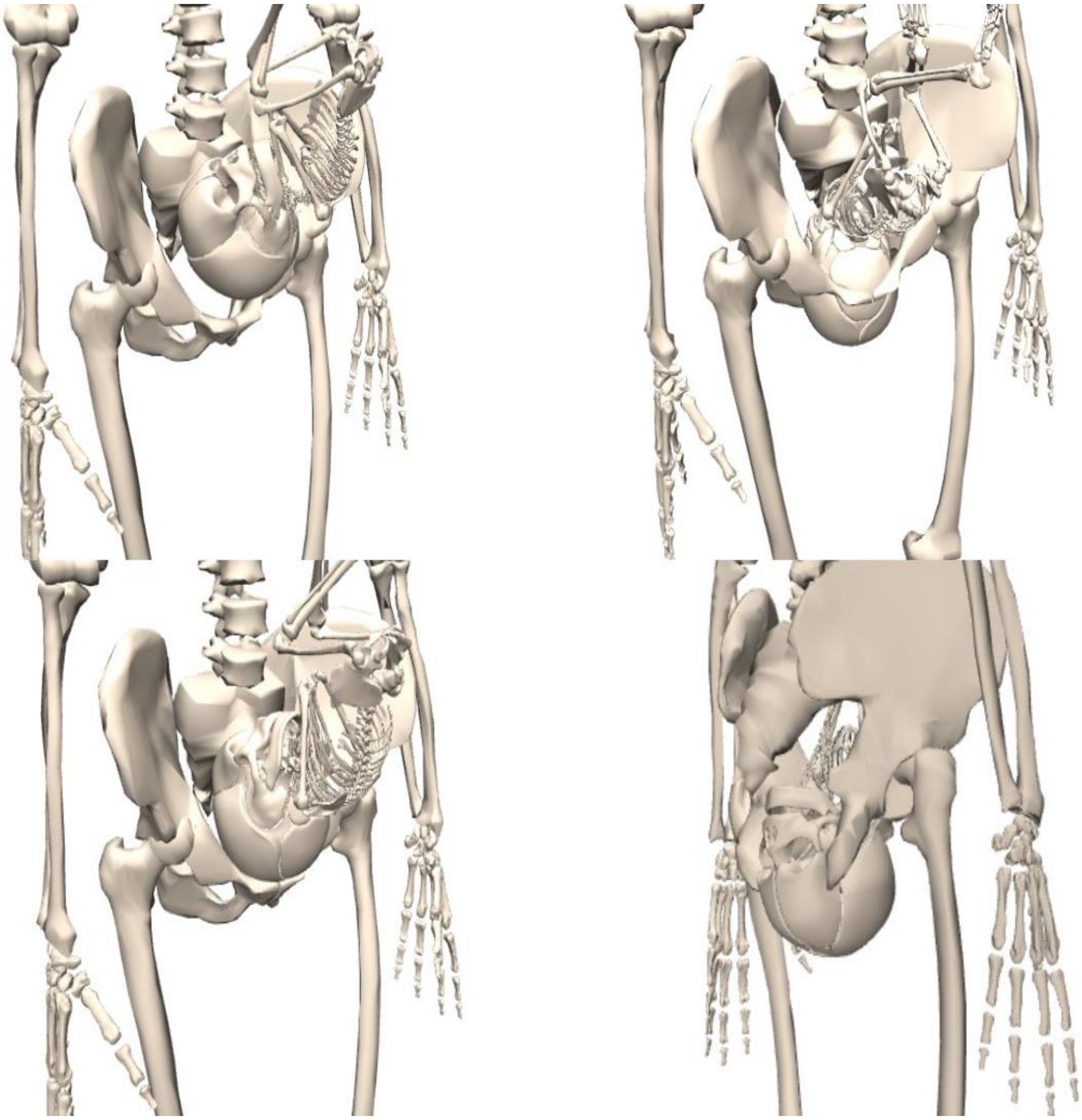


Figure 36. Birth with fetal and maternal models.

One significant advantage of having both maternal and fetal models present in the simulation is the flexibility to manipulate the maternal model into different poses. This allows practitioners to adjust any necessary variables in maternal pose to find the optimal solution during delivery. The ability to experiment with various maternal positions and scenarios enables a better understanding of the effects of maternal positioning on the birthing process. Thus, this comprehensive model provides a platform for patient-specific training and simulation, aiding in the development of new birthing techniques and enhancing the overall delivery experience for both the mother and the practitioner. Figure 37 shows the maternal model on her back.

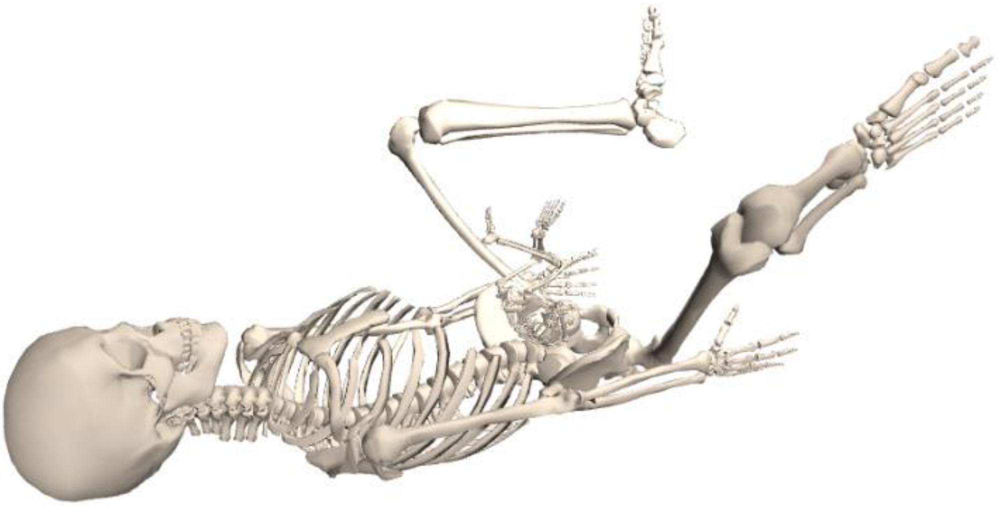


Figure 37. Maternal model put on her back.

However, there are limitations in modeling the fetal and maternal models in OpenSim. First, the fetal model is based on a re-scaled version of an adult model. The re-scaling had to be done in two steps to ensure the accuracy of the final fetus. This also was not possible without accessing the source-file and changing some of the parameters directly.

The second limitation is related to the FE birthing simulation data. The Porto-based research group introduced a set of pre-defined movements for the fetal model which is not the real case in most delivery scenarios as by the time of the delivery, the fetus head is already in position near the vaginal canal.

Third, the interaction between fetal and maternal models in this study is not simulated. The models are separate from each other and manually adjusted in the scene on top of each other to present a better visualization of birthing process.

Fourth, the IK result which enabled the simulation of birth is calculated based on a set of fixed fetal coordinates, meaning if the posture of the mother changes, the fetal simulation cannot cope with the new setting and new displacement data should be calculated or adjusted based on the rotation of the maternal structure.

Fifth, although the OpenSim software is capable of handling Biomechanical studies, due to its limited output and input file recognition any model created or used inside it cannot be used in different open-source or commercial software for future expansions.

Finally, the maternal model's anthropometric data was not modified. Thus, to be able to make this simulator more patient-specific, the maternal model's body measurements should be accounted for and implemented.

4.4 Visual Enhancement Results

This study tried to enhance the visualization of the interactive birth process which helps make the training experience more realistic for practitioners. Although musculoskeletal dynamics can help in better understanding labor dynamics, in real life what physicians see visually when examining their patients is not in terms of their skeletal structure. This final step of the study aimed at providing a better approach in visualizing the interactive simulation. The result of the two approaches mentioned in the methodology section are as follows.

The BASH model which stands for Biomechanical Animated Skinned Human is chosen as the first approach. This software package uses the OpenSim model file with its respective motion file to generate a subject specific mesh skin using marker data provided by the model. This platform was built upon the SCAPE project which introduced this idea back in 2005.

When the fetal model input to exercise using the BASH model, the output is not what was expected and not useful as the model behaved erratically. Figure 38 shows the result of this trial.

Figure 38. BASH model implementation of fetal model.

The model shows distorted mesh in most parts of the body. One reason for this result may be that the BASH model used the SCAPE model which was created for adult humans. The SCAPE model was trained only with adult size models; thus, having very small fetal anatomy did not cope well. When markers on the fetal body get close to each other in terms of their relative distance, the matrix calculation introduced in the SCAPE model cannot cope with small distances and generates errors.

As this result is dissatisfactory, another method is implemented to scale up the models, meaning keeping the fetal model size equal to actual adult size and re-scaling the maternal model to 4 times that of the fetus. If this approach is successful, future adjustments can be applied to fix the scaleup in terms of training scenarios, but this method also resulted in dissatisfaction. Figure 39 shows the result of the scaling up method.

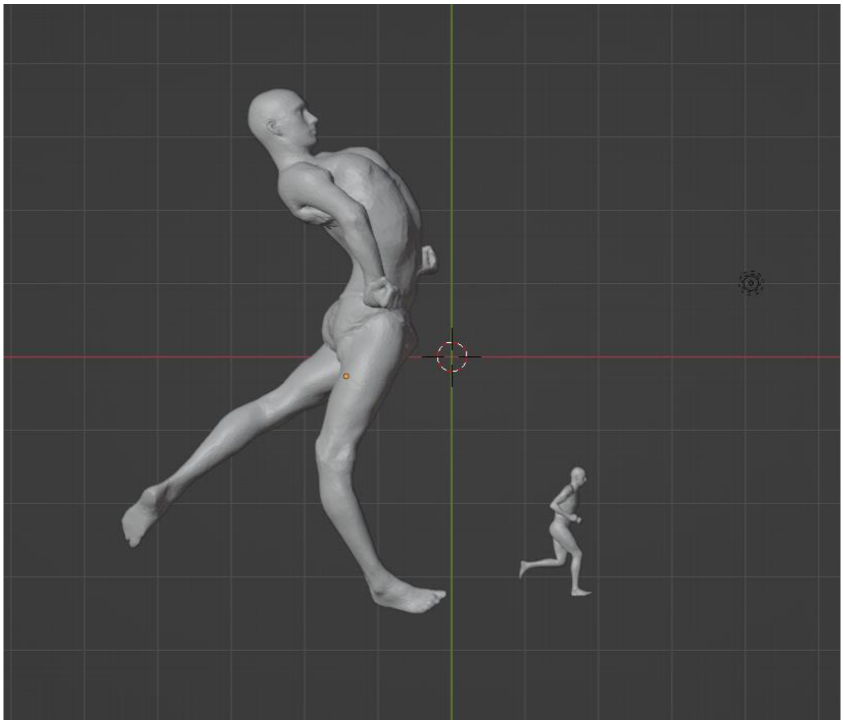


Figure 39. Scale up maternal model with adult size fetal.

Although the adult size fetal model is showing correct skin formation, the output does not resemble any fetal anatomy and discourages the user from using this model. Moreover, the maternal model also shows excessive deformity and demonstrates that scaling up models is also not going to work.

Besides the limitations mentioned for the BASH model, another issue is the lack of maternal anatomy in the publicly available SCAPE dataset. Therefore, no attempts could be made to use a proper female mesh to acquire enhanced visualization of the maternal model.

These BASH manipulation methods show that even though the BASH model is capable of presenting enhanced visualization for most OpenSim models, the extreme height difference between a fetal structure and adult model cannot be covered in BASH. This result forced this study to seek another approach to visualizing the fetal structure with enhanced visualization.

The second attempt consisted of using the Blender software and a full term generic fetal 3D model. The aim is to re-create an enhanced visualization of the online birthing process using this model. Using a rigging tool to act like a structure for the fetal mesh and providing the set of predefined movements based on the IK results created an acceptable result in terms of animating the IK simulation with fetal mesh. Figure 40 shows the result of this approach.

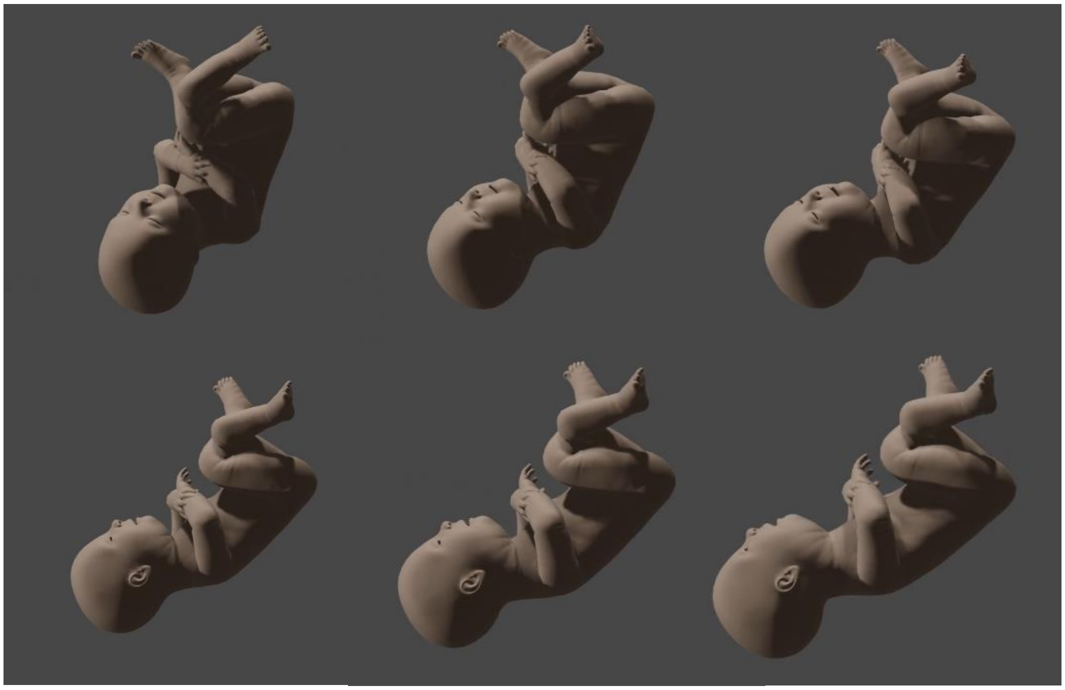


Figure 40. IK result with enhanced visualization of fetal model.

CHAPTER 4

CONCLUSION AND FUTURE WORK

This study aimed to build upon a novel template described in the methodology section which set a roadmap to build a patient-specific, VR-based, Haptic driven obstetrics simulator by expanding upon this framework and simulating an interactive birthing process through the use of OpenSim and utilization of Inverse Kinematics (IK) techniques. OpenSim is selected due to its ease of use. It would be easy to manipulate, change the measurements, add or delete body segments in the OpenSim model compared to finite element models. Additionally, OpenSim IK calculations are so fast that they cannot be compared to finite element calculations.

The primary objective of this study is to develop an interactive, patient-specific labor simulation that can adapt to different fetal and maternal dimensions and provide a patient-specific and realistic representation of the birth process emphasizing musculoskeletal dynamics.

The results of this study demonstrate the implementation of the OpenSim and IK techniques for fetal skeletal modeling and simulation. The conversion of an onerous FE birthing process to an interactive, patient-specific simulation of birth is achieved with accurate results. Furthermore, the inclusion of the maternal anatomical model in the simulation provided a more comprehensive representation of the birthing process, allowing for the manipulation of maternal pose that can help optimize the training scenarios.

Another aspect of this study focused on the visual enhancement of the musculoskeletal representation. Two approaches are explored, including the use of the BASH model and manual visualization in Blender. While the BASH model approach did not yield satisfactory results, the manual visualization technique generated a skinned animated model of the fetus that provides a more realistic representation of fetal movement during childbirth.

However, for future work, this study only used a specific set of data as the means for fetal movement in 3D space. The fetal rotation and orientation are defined based on this data. Thus, it is necessary for future developments to use additional data to justify fetal movement in 3D space. Although, as of now such datasets are not available, such datasets can be obtained through use of mannequins by simulating birth with motion capture.

Additionally, the 4-markers used in developing this study are not enough as the inverse-kinematic trial showed. For future developments it is necessary to model the fetal anatomy with more markers in different locations such as major joints. These markers should be placed at the center of each joint respectively, and data regarding their displacement should be acquired.

The novel obstetrics simulator architecture uses the 3D segmentation of fetal MRI and its coupling with the musculoskeletal anatomy as the basis of its simulator. This study, due to lack of MRI datasets followed a different path for skinning. However, in the future this study should emphasize fetal MRI segmentation and its role in visual enhancement of fetal structure.

Additionally, a vital step toward an interactive obstetrics simulator is to couple virtual-reality goggles and haptic driven gloves to provide the necessary tools for its intended users to practice. This can be achieved through different scenarios. One, either in a future version of OpenSim this capability will be incorporated, or two, the OpenSim musculoskeletal anatomy created with anthropometric data relevant to a particular study needs to be exported to a suitable space capable of handling VR and haptics. As of publication of this thesis, only one plug-in is developed by Abella, J. and Demircan, E. [48] that is capable of importing OpenSim models through a Unity [49] environment which is capable of such a task.

This study did not incorporate any forces generated by the uterus during labor. The Porto-based research group recently published a paper that simulated uterine contractions during labor using a chemo-mechanical constitutive model [50]. To accurately model labor, the forces involved with this process should be carefully modeled and implemented in the simulation.

Lastly, as this simulation's patient-specificity is important for the fetus, this idea should be considered for the mother as well. For example, a pregnant woman's pelvis becomes larger during pregnancy; thus, this increase in size should be incorporated in future maternal models as well as incorporation of her anthropometric data.

Overall, this study has significant implications for the medical community by providing a more comprehensive and realistic representation of the birth process. This interactive musculoskeletal delivery simulation can be used as a stepping stone to develop comprehensive training tools for medical practitioners to improve their understanding of birth and to optimize their delivery outcomes. The visual enhancement techniques explored in this study can also be further developed to improve the realism of the simulation and enhance its overall effectiveness as a training tool.

REFERENCES

[1] R. Tikkanen, M. Z. Gunja, M. FitzGerald, and L. Zephyrin, "Maternal mortality and maternity care in the United States compared to 10 other developed countries," *The Commonwealth Fund*, vol. 10, 2020.

[2] D. L. Hoyert, "Maternal mortality rates in the United States," *NCHS Health E-Stats*, 2023, doi: https://dx.doi.org/10.15620/cdc:124678.

[3] L. Birch *et al.*, "Obstetric skills drills: evaluation of teaching methods," *Nurse education today*, vol. 27, no. 8, pp. 915-922, 2007.

[4] A. Elixhauser and L. M. Wier, "Complicating conditions of pregnancy and childbirth, 2008: statistical brief# 113," 2011.

[5] L. G. Glance *et al.*, "Rates of major obstetrical complications vary almost fivefold among US hospitals," *Health Affairs*, vol. 33, no. 8, pp. 1330-1336, 2014.

[6] D. M. Avery, "Obstetric emergencies," *Am J Clin Med*, vol. 6, no. 2, pp. 42-47, 2009.

[7] S. Menticoglou, "Shoulder dystocia: incidence, mechanisms, and management strategies," *International Journal of Women's Health*, pp. 723-732, 2018.

[8] E. G. Baxley and R. W. Gobbo, "Shoulder dystocia," *American Family Physician*, vol. 69, no. 7, pp. 1707-1714, 2004.

[9] K. R. Jeanne M. Carey. "The How When Why of High Fidelity Simulation." https://pubmed.ncbi.nlm.nih.gov/32644739/ (accessed.

[10] J. F. Crofts, C. Bartlett, D. Ellis, L. P. Hunt, R. Fox, and T. J. Draycott, "Training for shoulder dystocia: a trial of simulation using low-fidelity and high-fidelity mannequins," *Obstetrics & Gynecology*, vol. 108, no. 6, pp. 1477-1485, 2006.

[11] C. Massoth *et al.*, "High-fidelity is not superior to low-fidelity simulation but leads to overconfidence in medical students," *BMC medical education*, vol. 19, pp. 1-8, 2019.

[12] M. W. Scerbo, E. A. Schmidt, and J. P. Bliss, "Comparison of a virtual reality simulator and simulated limbs for phlebotomy training," *Journal of Infusion Nursing*, vol. 29, no. 4, pp. 214-224, 2006.

[13] E. D. Matsumoto, S. J. Hamstra, S. B. Radomski, and M. D. Cusimano, "The effect of bench model fidelity on endourological skills: a randomized controlled study," *The Journal of urology*, vol. 167, no. 3, pp. 1243-1247, 2002.

[14] K. H. K. Lee, H. Grantham, and R. Boyd, "Comparison of high-and low-fidelity mannequins for clinical performance assessment," *Emergency Medicine Australasia,* vol. 20, no. 6, pp. 508-514, 2008.

[15] Laerdal. "PROMPT Flex Birthing Simulator." https://laerdal.com/us/products/simulation-training/obstetrics-pediatrics/prompt-flex/ (accessed.

[16] D. G. Krishnan, A. V. Keloth, and S. Ubedulla, "Pros and cons of simulation in medical education: A review," *Education,* vol. 3, no. 6, pp. 84-87, 2017.

[17] M. Mahling, R. Wunderlich, D. Steiner, E. Gorgati, T. Festl-Wietek, and A. Herrmann-Werner, "Virtual Reality for Emergency Medicine Training in Medical School: Prospective, Large-Cohort Implementation Study," *Journal of Medical Internet Research,* vol. 25, p. e43649, 2023.

[18] P. Ravert, "An integrative review of computer-based simulation in the education process," *CIN: Computers, Informatics, Nursing,* vol. 20, no. 5, pp. 203-208, 2002.

[19] A. Karakuş, L. Duran, Y. Yavuz, L. Altintop, and F. Çalişkan, "Computer-based simulation training in emergency medicine designed in the light of malpractice cases," *BMC medical education,* vol. 14, pp. 1-5, 2014.

[20] S. Motta and F. Pappalardo, "Mathematical modeling of biological systems," *Briefings in Bioinformatics,* vol. 14, no. 4, pp. 411-422, 2013.

[21] S. Pinto *et al.*, "Application of virtual reality techniques to a birth simulation," in *2017 4th Experiment@ International Conference (exp. at'17)*, 2017: IEEE, pp. 125-126.

[22] M. Audette *et al.*, "Toward an Interactive, Patient-specific, VR-based Obstetrics Simulator," 2021.

[23] R. J. Scalese, V. T. Obeso, and S. B. Issenberg, "Simulation technology for skills training and competency assessment in medical education," *Journal of general internal medicine,* vol. 23, no. 1, pp. 46-49, 2008.

[24] J. C. Barrett, "A Monte Carlo simulation of human reproduction," *Genus,* pp. 1-22, 1969.

[25] M. A. Audette, S. P. Bordas, and J. E. Blatt, "Robotically steered needles: A survey of neurosurgical applications and technical innovations," *Robotic Surgery: Research and Reviews,* pp. 1-23, 2020.

[26] S. L. Delp *et al.*, "OpenSim: open-source software to create and analyze dynamic simulations of movement," *IEEE transactions on biomedical engineering*, vol. 54, no. 11, pp. 1940-1950, 2007.

[27] M. P. L. Parente, "Biomechanics of the Pelvic Floor during Vaginal delivery," Universidade do Porto (Portugal), 2008.

[28] M. Parente, R. N. Jorge, T. Mascarenhas, A. Fernandes, and J. Martins, "The influence of an occipito-posterior malposition on the biomechanical behavior of the pelvic floor," *European Journal of Obstetrics & Gynecology and Reproductive Biology*, vol. 144, pp. S166-S169, 2009.

[29] M. Silva *et al.*, "Study on the influence of the fetus head molding on the biomechanical behavior of the pelvic floor muscles, during vaginal delivery," *Journal of Biomechanics*, vol. 48, no. 9, pp. 1600-1605, 2015.

[30] C. C. Spyrakos, *Finite Element Modeling*. West Virginia Univ. Press Morgantown, WV, USA, 1994.

[31] G. Abaqus, "Abaqus 6.11," *Dassault Systemes Simulia Corporation, Providence, RI, USA*, 2011.

[32] T. Clanerani, *A short history of obstetrics and gynecology*. Charles C Thomas, 1960.

[33] Y. Lim, T. Chambers, C. Walck, S. Siddicky, E. Mannen, and V. Huayamave, "Challenges in Kinetic-Kinematic Driven Musculoskeletal Subject-Specific Infant Modeling," *Mathematical and Computational Applications*, vol. 27, no. 3, p. 36, 2022.

[34] S. W. Verbruggen *et al.*, "Modeling the biomechanics of fetal movements," *Biomechanics and modeling in mechanobiology*, vol. 15, pp. 995-1004, 2016.

[35] D. Kim, H. Kanazawa, and Y. Kuniyoshi, "Simulating a Human Fetus in Soft Uterus," in *2022 IEEE International Conference on Development and Learning (ICDL)*, 2022: IEEE, pp. 135-141.

[36] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ international conference on intelligent robots and systems*, 2012: IEEE, pp. 5026-5033.

[37] SimTK.org. "The OpenSim GUI." https://simtk-confluence.stanford.edu:8443/display/OpenSim/Welcome+to+OpenSim (accessed.

[38] A. Seth *et al.*, "OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement," *PLoS computational biology,* vol. 14, no. 7, p. e1006223, 2018.

[39] SimTK.org. "OpenSim API Guide." https://simtk.org/api_docs/opensim/api_docs/md_doc_APIGuide.html (accessed.

[40] R. Schleicher *et al.*, "BASH: Biomechanical Animated Skinned Human for Visualization of Kinematics and Muscle Activity," in *VISIGRAPP (1: GRAPP)*, 2021, pp. 25-36.

[41] O. Blender, "Blender—A 3D modelling and rendering package," *Retrieved. represents the sequence of Constructs1 to,* vol. 4, 2018.

[42] K. Burkhart, D. Grindle, M. L. Bouxsein, and D. E. Anderson, "Between-session reliability of subject-specific musculoskeletal models of the spine derived from optoelectronic motion capture data," *Journal of biomechanics,* vol. 112, p. 110044, 2020.

[43] J. M. JUDD, "Assessing the newborn from head to toe," *Nursing2021,* vol. 15, no. 12, pp. 34-41, 1985.

[44] Wikipedia. "Fetal Position." https://en.wikipedia.org/wiki/Fetal_position (accessed.

[45] SimTK.org. "Tutorial 3 - Scaling, Inverse Kinematics, and Inverse Dynamics." https://simtk-confluence.stanford.edu:8443/display/OpenSim/Tutorial+3+-+Scaling%2C+Inverse+Kinematics%2C+and+Inverse+Dynamics (accessed.

[46] SimTK.org. "How Inverse Kinematics Works." https://simtk-confluence.stanford.edu:8443/display/OpenSim/How+Inverse+Kinematics+Works (accessed.

[47] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "Scape: shape completion and animation of people," in *ACM SIGGRAPH 2005 Papers*, 2005, pp. 408-416.

[48] J. Abella and E. Demircan, "A multi-body simulation framework for live motion tracking and analysis within the unity environment," in *2019 16th International Conference on Ubiquitous Robots (UR)*, 2019: IEEE, pp. 654-659.

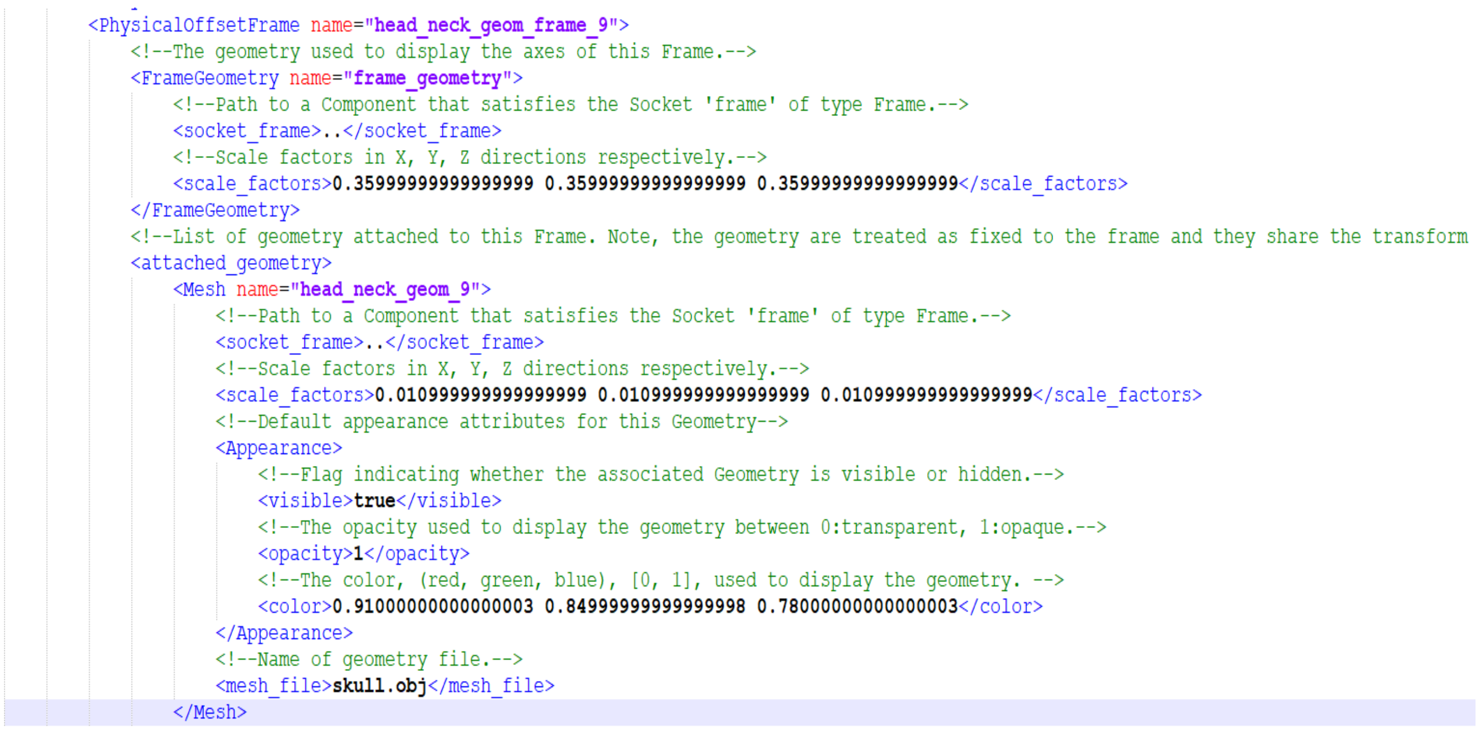
[49] J. K. Haas, "A history of the unity game engine," *Diss. Worcester Polytechnic Institute,* vol. 483, no. 2014, p. 484, 2014.

[50] M. Vila Pouca, J. Ferreira, D. Oliveira, M. Parente, M. Mascarenhas, and R. Natal Jorge, "Simulation of the uterine contractions and foetus expulsion using a chemo-mechanical

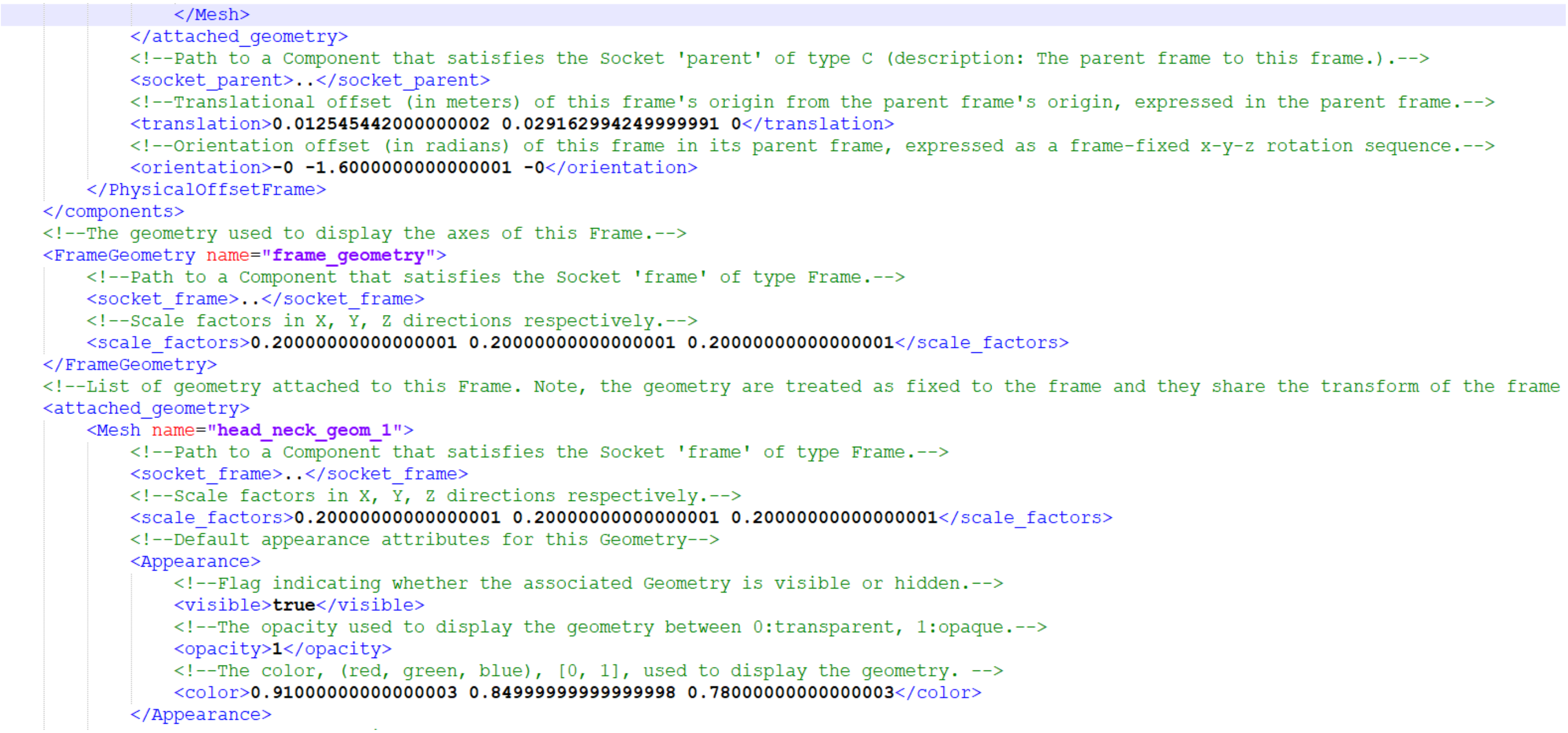
constitutive model," *Biomechanics and Modeling in Mechanobiology*, vol. 18, no. 3, pp. 829-843, 2019.

APPENDICES

A. Fetal XML Code – Skull Replacement Info

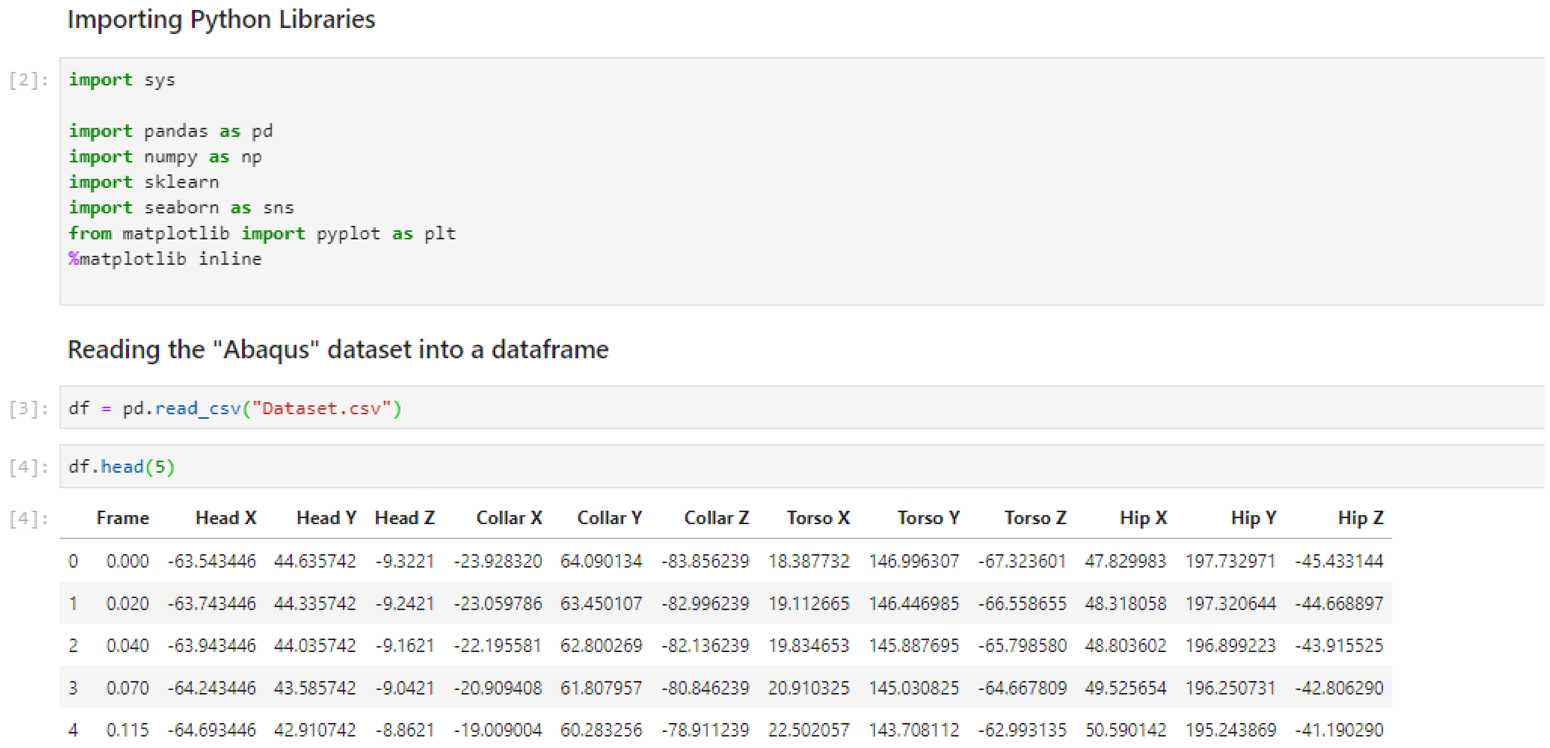


```
<PhysicalOffsetFrame name="head_neck_geom_frame_9">
    <!--The geometry used to display the axes of this Frame.-->
    <FrameGeometry name="frame_geometry">
        <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
        <socket_frame>..</socket_frame>
        <!--Scale factors in X, Y, Z directions respectively.-->
        <scale_factors>0.35999999999999999 0.35999999999999999 0.35999999999999999</scale_factors>
    </FrameGeometry>
    <!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform
    <attached_geometry>
        <Mesh name="head_neck_geom_9">
            <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
            <socket_frame>..</socket_frame>
            <!--Scale factors in X, Y, Z directions respectively.-->
            <scale_factors>0.010999999999999999 0.010999999999999999 0.010999999999999999</scale_factors>
            <!--Default appearance attributes for this Geometry-->
            <Appearance>
                <!--Flag indicating whether the associated Geometry is visible or hidden.-->
                <visible>true</visible>
                <!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->
                <opacity>1</opacity>
                <!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
                <color>0.91000000000000003 0.84999999999999998 0.78000000000000003</color>
            </Appearance>
            <!--Name of geometry file.-->
            <mesh_file>skull.obj</mesh_file>
        </Mesh>
```



```xml
            </Mesh>
        </attached_geometry>
        <!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->
        <socket_parent>..</socket_parent>
        <!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->
        <translation>0.012545442000000002 0.029162994249999991 0</translation>
        <!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->
        <orientation>-0 -1.6000000000000001 -0</orientation>
    </PhysicalOffsetFrame>
</components>
<!--The geometry used to display the axes of this Frame.-->
<FrameGeometry name="frame_geometry">
    <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
    <socket_frame>..</socket_frame>
    <!--Scale factors in X, Y, Z directions respectively.-->
    <scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
</FrameGeometry>
<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame
<attached_geometry>
    <Mesh name="head_neck_geom_1">
        <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
        <socket_frame>..</socket_frame>
        <!--Scale factors in X, Y, Z directions respectively.-->
        <scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
        <!--Default appearance attributes for this Geometry-->
        <Appearance>
            <!--Flag indicating whether the associated Geometry is visible or hidden.-->
            <visible>true</visible>
            <!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->
            <opacity>1</opacity>
            <!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
            <color>0.91000000000000003 0.84999999999999998 0.78000000000000003</color>
        </Appearance>
```

B. Python Pipeline for Data Validation



Importing Python Libraries

```
[2]: import sys

     import pandas as pd
     import numpy as np
     import sklearn
     import seaborn as sns
     from matplotlib import pyplot as plt
     %matplotlib inline
```

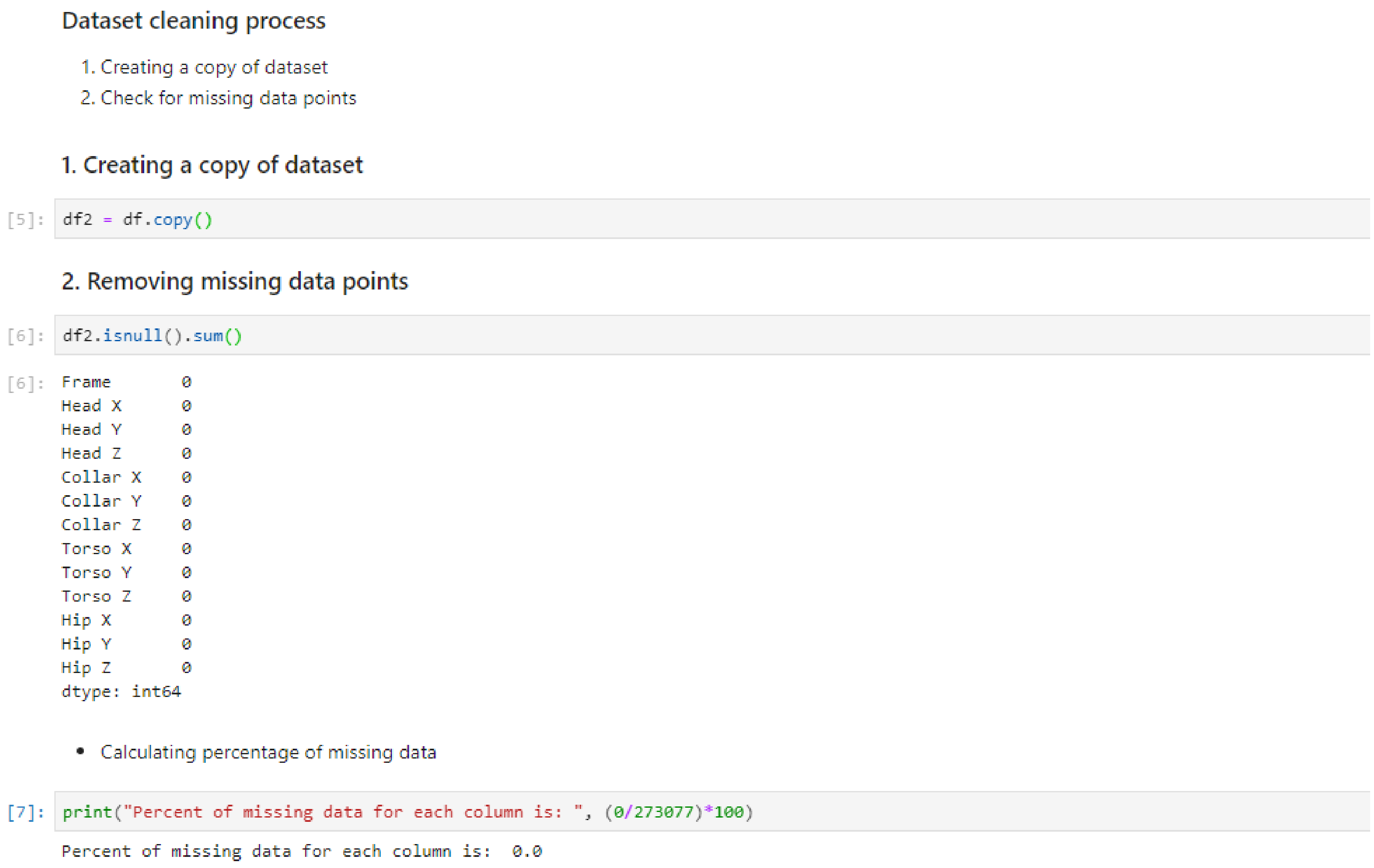
Reading the "Abaqus" dataset into a dataframe

```
[3]: df = pd.read_csv("Dataset.csv")
```

```
[4]: df.head(5)
```

[4]:

	Frame	Head X	Head Y	Head Z	Collar X	Collar Y	Collar Z	Torso X	Torso Y	Torso Z	Hip X	Hip Y	Hip Z
0	0.000	-63.543446	44.635742	-9.3221	-23.928320	64.090134	-83.856239	18.387732	146.996307	-67.323601	47.829983	197.732971	-45.433144
1	0.020	-63.743446	44.335742	-9.2421	-23.059786	63.450107	-82.996239	19.112665	146.446985	-66.558655	48.318058	197.320644	-44.668897
2	0.040	-63.943446	44.035742	-9.1621	-22.195581	62.800269	-82.136239	19.834653	145.887695	-65.798580	48.803602	196.899223	-43.915525
3	0.070	-64.243446	43.585742	-9.0421	-20.909408	61.807957	-80.846239	20.910325	145.030825	-64.667809	49.525654	196.250731	-42.806290
4	0.115	-64.693446	42.910742	-8.8621	-19.009004	60.283256	-78.911239	22.502057	143.708112	-62.993135	50.590142	195.243869	-41.190290



Dataset cleaning process

1. Creating a copy of dataset
2. Check for missing data points

1. Creating a copy of dataset

```
[5]: df2 = df.copy()
```

2. Removing missing data points

```
[6]: df2.isnull().sum()
```

```
[6]: Frame       0
     Head X      0
     Head Y      0
     Head Z      0
     Collar X    0
     Collar Y    0
     Collar Z    0
     Torso X     0
     Torso Y     0
     Torso Z     0
     Hip X       0
     Hip Y       0
     Hip Z       0
     dtype: int64
```

- Calculating percentage of missing data

```
[7]: print("Percent of missing data for each column is: ", (0/273077)*100)

     Percent of missing data for each column is:  0.0
```

VITA

Bahador Dodge

EDUCATION

Old Dominion University, Norfolk, VA

Master of Science in Modeling and Simulation Engineering	May 2023
Bachelor of Science in Electrical Engineering	February 2008

RESEARCH AND PUBLICATIONS

- Purwanto, Wirawan, **Bahador Dodge**, et al. "DeapSECURE Computational Training for Cybersecurity: Progress Toward Widespread Community Adoption." Journal of computational science education (2023).
- **Dodge, Bahador**, et al. "DeapSECURE Computational Training for Cybersecurity: Third-Year Improvements and Impacts." (2022).