Full-body Inverse Dynamics using Inertial Measurement Units

by

Mohsen Diraneyya

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
System Design Engineering

Waterloo, Ontario, Canada, 2019

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my department.

I understand that my thesis may be made electronically available to the public.

Abstract

Estimating the loads on the human body is crucial in ergonomics, where it is of use in workplace design, task-load assessment, and safety limits establishment. It is also relevant to rehabilitation studies, where it can be used to design programs, activities, and instruments. Estimating these loads requires the collection of data on motion kinematics and external forces during the task or exercise of interest. Traditionally, Optical Motion Capture (OMC) systems and Force Plates (FPs) were commonly used to collect kinematic data and measure Ground Reaction Forces (GRFs). However, this experimental set-up is limited to laboratory settings and small, confined spaces. It also imposes significant instrumentation costs.

The availability of wearable Inertial Measurement Units (IMUs) and better signal processing techniques have allowed for the development of effective whole body Inertial Motion Capture (IMC) systems. Inverse dynamic models that use motion kinematics collected from these systems are also being developed. A challenging aspect in this endeavor is the need to estimate GRFs from kinematics, without recourse to FPs, in order to take full advantage of the IMC systems' portability. To overcome this challenge, some models include upper body segments only and solve for joints loads using a top-down approach. Other models consider gait motions and apply a smooth transition assumption relevant only to the gait cycle. The aim of this current research is to continue along this latter line of development

by introducing a general purpose full-body inverse dynamics model based on IMC kinematics. This model allows for true system portability, dispensing with the use of FPs and any other equipment confined to in-lab use.

This study has developed a whole-body model that determines the net forces and moments in body joints during general motions captured using an IMC system. Further, an anatomical lower-spine model has also been used to estimate the disk contact forces in the lower back and thereby assess the critical loads on the lower back. Using inverse dynamics, the model also estimates total GRF from the kinematic data and breaks it down into right and left GRFs using an optimization approach that minimizes energy expenditure. The model predictions were validated by comparing them to values measured during an experimental pilot study. The results show an excellent prediction of the total vertical GRF, with relative Root-Mean-Square-Error (rRMSE) of less than 2.4 %. The predictions were less accurate for the horizontal components, ranging from 22.5 % to 39.4% for the anterior-posterior direction, mainly because of their smaller amplitudes. The optimization approach for predicting the right and left vertical GRFs performed well for standing and walking tasks, with rRMSE less than 13.0 %.

The model was then used to analyze the forces experienced by masons during bricklaying. Static and dynamic estimates of joint loads were compared to understand how movement affects joint loads.

Acknowledgements

First, I would like to thank my supervisors, Professor Eihab M. Abdel-Rahman and Professor Carl T. Haas for their guidance throughout my studies, this work could not be completed without their supervision. I would also like to thank my colleagues, Lichen Zhang and Juhyeong Ryu for their help and encouragement.

I would like to extend my thanks to the thesis reviewers; Professor Steven Fischer and Professor Matthew Borland for their thorough feedback to improve and enhance this thesis. I would like thank professor Steven Fischer, Aleks Budarick, and Amanda Calford for their help with the validation study data collection and allowing me to use the occupational biomechanics and ergonomics lab. I would like to extend my gratitude to the Canadian Masonry Design Center for funding this project and helping with the data collection on bricklayers.

I would like to thank Professor Ziyad Masoud for his support and encouragement, without his help I would not be where I am today.

Last but not least I thank my dear siblings: Asmaa, Israa, Muslem, Mones, and Salma for their love and support.

إهداء

إلى أمي وأبي,

عجزت دوما عن التعبير عن حبي وامتناني لكما لحبكما لي, اهتمامكما بي, وتعبكما علي منذ أن جلبتماني للحياة. شكرا لكما وأرجو أن أنال دومًا رضاكما.

Dedication

To my father and mother; Mutasem and Rehab. I have always failed to show my love and appreciation for your love, concern, and support since I was a kid. Thank you and I hope that I always meet your expectations.

Table of Contents

List of Tables

List of Figures

Acronyms

3DSSPP 3D Static Strength Prediction Program.

AJC Ankle Joint Center.

CJC Cervical Joint Center.

CMUs Concrete Masonry Units.

CS Coordinate System.

DOF Degrees of Freedom.

EJC Elbow Joint Center.

EO External Oblique.

ES Erector Spinae.

FPs Force Plates.

GRFs Ground Reaction Forces.

GUI Graphic User Interface.

HJC Hip Joint Center.

HV Head Vertex.

IMC Inertial Motion Capture.

IMUs Inertial Measurement Units.

IO Internal Oblique.

ISB International Society of Biomechanics.

KJC Knee Joint Center.

LD Latissimus Dorsi.

LJC Lumbar Joint Center.

MH Metacarpal Heads.

MRI Magnetic Resonance Imaging.

MSDs Musculoskeletal Disorders.

NIOSH National Institute for Occupational Safety and Health.

OMC Optical Motion Capture.

PCSA Physiological Cross-Sectional Area.

RA Rectus Abdominis.

RMSE Root-Mean-Square Error.

rRMSE relative Root-Mean-Square Error.

SJC Shoulder Joint Center.

WJC Wrist Joint Center.

List of Symbols

m^i Segment i mass.

$\boldsymbol{r}^i$ Segment i center of mass with respect to the local frame origin.

$\boldsymbol{\mathfrak{F}}^i$ Segment i local frame.

$[\boldsymbol{J}^i]$ Segment i inertia tensor.

$\boldsymbol{\zeta}^1$ The root segment (Pelvis) local frame origin with respect to the reference frame origin.

$\boldsymbol{\mathfrak{F}}^0$ The IMC reference frame.

$\boldsymbol{\zeta}^i$ Segment i local frame origin with respect to the parent segment frame origin.

$\boldsymbol{\mathfrak{F}}^k$ Segment k local frame.

$[{}^0\boldsymbol{R}^i]$ Rotational matrix from segment i local frame to the reference frame.

$[{}^k\boldsymbol{R}^i]$ Rotational matrix from segment i local frame to the parent segment local frame.

$\boldsymbol{\alpha}^i$ Segment i angular acceleration.

$\boldsymbol{\omega}^i$ Segment i angular velocity.

$\boldsymbol{a}^i$ Segment i center of mass acceleration.

$\boldsymbol{a}_s^i$ Segment i IMU acceleration.

$\boldsymbol{F}_{ex}^i$ Total external force on segment i.

$\boldsymbol{M}_{ex}^i$ Total external moment on segment i.

$\boldsymbol{F}^{ik}$ Net reaction force between segments i and k (i.e. joint ik net force).

$\boldsymbol{F}^{jni}$ Net reaction force between segments jn and i (i.e. joint jni net force).

$\boldsymbol{W}^i$ Weight force of segment i.

$\boldsymbol{M}^{ik}$ Net reaction moment between segments i and k (i.e. joint ik net moment).

$\boldsymbol{M}^{jni}$ Net reaction moment between segments jn and i (i.e. joint jni net moment).

$\boldsymbol{\zeta}^{jn}$ Segment jn local frame origin with respect to the parent segment frame origin..

$\boldsymbol{r}_{ex}^i$ Total external force on segment i point of action location with respect to the local frame origin.

$\boldsymbol{M}^{i\star}$ Net inertial moment of segment i.

$\mathfrak{F}^{jn}$ Segment jn local frame.

I_n Muscle n intensity; force per unit physiological cross-sectional area.

A_n The physiological cross-sectional area of muscle n.

I_{max} Maximum muscle intensity.

F_{res} Tension force in the right Erector Spinae muscle.

F_{les} Tension force in the left Erector Spinae muscle.

F_{rld} Tension force in the right Latissimus Dorsi muscle.

F_{lld} Tension force in the left Latissimus Dorsi muscle.

F_{rra} Tension force in the right Rectus Abdominis muscle.

F_{lra} Tension force in the left Rectus Abdominis muscle.

F_{rio} Tension force in the right Internal Oblique muscle.

F_{lio} Tension force in the left Internal Oblique muscle.

F_{reo} Tension force in the right External Oblique muscle.

F_{leo} Tension force in the left External Oblique muscle.

F_c Compression contact force on the L5/S1 disk.

F_{sl} Lateral shear contact force on the L5/S1 disk.

F_{sa} Anterior shear contact force on the L5/S1 disk.

F_{ap} The force of the abdominal pressure exerted on the thoracic diaphragm.

θ_{es} The angle between the Erector Spinae muscle line of action and the normal to the L5/S1 disk in the sagittal plane.

θ_{ra} The angle between the Rectus Abdominis muscle line of action and the normal to the L5/S1 disk in the sagittal plane.

θ_{io} The angle between the Internal Oblique muscle line of action and the normal to the L5/S1 disk in the sagittal plane.

θ_{eo} The angle between the External Oblique muscle line of action and the normal to the L5/S1 disk in the sagittal plane.

θ_{ld} The angle between the Latissimus Dorsi muscle line of action and the normal to the L5/S1 disk in the coronal plane.

r^s_{es} The moment arm of the Erector Spinae muscle in the sagittal plane.

r^s_{ld} The moment arm of the Latissimus Dorsi muscle in the sagittal plane.

r^s_{ra} The moment arm of the Rectus Abdominis muscle in the sagittal plane.

r^s_{io} The moment arm of the Internal Oblique muscle in the sagittal plane.

r^s_{eo} The moment arm of the External Oblique muscle in the sagittal plane.

r^c_{es} The moment arm of the Erector Spinae muscle in the coronal plane.

r^c_{ld} The moment arm of the Latissimus Dorsi muscle in the coronal plane.

r^c_{ra} The moment arm of the Rectus Abdominis muscle in the coronal plane.

r^c_{io} The moment arm of the Internal Oblique muscle in the coronal plane.

r^c_{eo} The moment arm of the External Oblique muscle in the coronal plane.

r^s_{ap} The force of the abdominal pressure exerted on the thoracic diaphragm.

A_{es} The physiological cross-sectional area of the Erector Spinae muscle.

A_{ld} The physiological cross-sectional area of the Latissimus Dorsi muscle.

A_{ra} The physiological cross-sectional area of the Rectus Abdominis muscle.

A_{io} The physiological cross-sectional area of the Internal Oblique muscle.

A_{eo} The physiological cross-sectional area of the External Oblique muscle.

$\boldsymbol{F}_g$ Total ground reaction force vector.

$\boldsymbol{M}_g$ Total ground reaction moment vector about the pelvic frame origin.

$\boldsymbol{l}^i$ Segment i center of mass with respect to the pelvic frame origin.

$\boldsymbol{l}^i_{ex}$ Total external force on segment i point of action location with respect to the pelvic frame origin.

$\boldsymbol{F}_{rg}$ Right ground reaction force vector.

$\boldsymbol{F}_{lg}$ Left ground reaction force vector.

$\boldsymbol{M}_{rg}$ Right ground reaction moment vector about the pelvic frame origin.

$\boldsymbol{M}_{lg}$ Left ground reaction moment vector about the pelvic frame origin.

$\boldsymbol{F}^{i\star}$ Net inertial force of segment i.

$\boldsymbol{\mathfrak{F}}^L$ The laboratory reference frame.

Chapter 1

Introduction

1.1 Introduction

In biomechanics, inverse dynamics analysis is used to predict the forces and moments exerted by the body's muscles and joints when performing a specific motion. This analysis requires a range of data, including the subject's anthropometric measures, the motion's kinematics, and the external forces applied on the body.

The anthropometric data consists of the subject's height and weight, his/her segments' lengths, masses, and mass moments of inertia. While the subject's height and weight can be measured, the segment's properties differ for different choices of body models. For instance, the torso is modeled as one rigid segment in Dumas et al.'s [2] model and as two rigid segments, the thoracic and the abdominal, in Young et al.'s [3] model. Therefore, the segment's length can be measured as the distance between two bony landmarks as defined in the model in use. Alternatively, the segment's length along with the other inertial properties of the segment may be estimated from generic anthropometric data assuming a 50^{th} percentile subject.

Several motion capture systems can be used to obtain kinematic data. Optical Motion Capture (OMC) systems are the one most used to collect human body kinematics. However, optical-based tracking requires a line of sight between the cameras and the markers attached to the body segments. Because of this need and because of the required set-up procedure for these systems, they are constrained to in-lab use within a confined area. This renders them impractical for on-site studies or tasks that require open spaces. An alternative is the Inertial Motion Capture (IMC) systems, which have been a popular choice in recent years because of the advancement in accelerometer technologies. Such systems are a composite of several Inertial Measurement Units (IMUs) that are integrated with a human model. Each IMU unit contains a 3-axis accelerometer, a gyroscope, and a magnetometer, which is usually used to correct the drift in acceleration readings [4]. IMC systems can be used in any open space, they don't require line of sight, and have a minuscule effect on the normal human motion because of their small size.

The external forces acting on the body usually consist of the Ground Reaction Forces (GRFs) and hand loads. Although hand loads are usually known from the experiment's design, GRFs are challenging to collect. Usually, Force Plates (FPs) are used for that purpose. However, FPs suffer from a number of limitations that make them unsuitable for out-of-lab experiments: they are expensive, have a limited capture area, and have to be ground mounted. Alternatively, GRFs can be measured by wearable devices such as pressure insoles. Although GRFs are reaction forces that theoretically could be calculated from the inverse dynamics of the kinematic data, this is not the case when both of the subject's feet are in contact with the floor, as the problem becomes indeterminate. Several studies have attempted to solve the problem, by estimating GRFs based on empirical data for gait cycles (e.g. [5]), by using neural networks(e.g. [6]), or by using an optimization approach.

Inverse dynamics analysis is used to estimate joint net forces and moments. In many cases, this is sufficient to assess body loads and risk levels in a given task. In other cases, the net forces and moments must be broken down to individual loads experienced by the constituent components of joint forces and/or the joint contact forces. In this case, once joint net forces and moments were calculated by applying inverse dynamics to an 'external' model, a detailed 'internal' joint model is deployed to determine the force breakdown. Internal models typically suffer from indeterminacy, as most human joints have more muscles, ligaments, and contact forces than Degrees of Freedom (DOF). Therefore, these models have to be either simplified to a single equivalent muscle model, in which all the acting muscles are replaced by a single muscle, or define an optimization problem to solve the indeterminacy and estimating the muscle forces subject to an objective function.

Using the external full body model followed by internal joint models, joint net forces and moments, muscle forces, and joint contact forces can be evaluated. These loads can be used as risk indicators to assess the workers' motions patterns during the execution of a task under study.

Working in construction is one of the most physically demanding occupations with a high rate of Musculoskeletal Disorders (MSDs). Construction workers suffer an average of 70 back injuries that result in days away from work for every 10,000 full-time workers each year, and bricklayers have one of the highest rates, at 100 injuries per 10,000 workers [7]. In fact, over 37 percent of all work-related injuries are in construction [8], with overexertion and back injuries accounting for more than 40 percent of construction MSDs [9]. Furthermore, construction workers' injuries can result in permanent damage and early retirement [10].

1.2 Scope

Although IMC systems have been used extensively to measure body kinematics, only a few studies used them in kinetic analysis. Further, the scope of those studies has been limited to the upper-body joints only [11], the gait cycle only [12], or have required the use of FPs to estimate GRFs [13].

This study aims to develop a full body inverse dynamics model free of those limitations. The model will apply to any motion pattern that can be captured by an IMC system and will not require the use of FPs or other specialized equipment. The model will provide for assessment of loads experienced by workers during work tasks. The model had been used to build an on-site ergonomic assessment suite, built with Graphic User Interface (GUI), that uses movement files and apply inverse dynamic analysis to estimate the joint net moments and lumber disk contact force.

Several human modeling software is available for ergonomics assessment. However, all these tools are designed to address the ergonomic aspect when designing and evaluating a product or the workplace [14], not for assessing the worker's motion. In that aspect, these tools aid the designer, not the worker. They use worker anthropometric measures and motion data from an existing database. Alternatively, specific movement can be imported to the software, this may allow for the assessment of the worker's motion, but this is a lengthy process that requires the use of multiple software to format the kinematic data as these tools are not designed for motion assessment. Some of those tools (e.g. Jack [15]) can import the human motion directly from OMC systems, but not from the IMC system. Furthermore, those tools do not consider the inertial forces of the motion. As they are meant for the workplace and product design, their concern is the posture taken by the worker during interaction with the product and workplace. This posture affects only the static loads on the human body and does not affect the inertial forces.

The on-site ergonomic assessment suite developed in this study is meant for training and post-training assessment of workers. It can import motion kinematics directly as exported from two IMC systems. It can be used easily with little training, and it requires low computational cost so that it can perform a near real-time analysis. The tool is designed to visualize and interpret the results on the interface, to be used by trainers and trainees. The tool is built on the model developed in this study, therefore it is capable of performing both static and dynamic analysis of the motion in hand.

1.3 Objectives

The objectives of this research can be summarized as follows:

- Develop a full body inverse dynamics model for general motions captured by IMC systems

- Build an on-site biomechanics-based ergonomic assessment suite

The full body model will need to estimate GRFs from measured motion patterns without recourse to FPs. This problem becomes indeterminate when both feet are in contact with the ground, therefore, a contact detection algorithm is developed to predict foot contact and an optimization approach [16] is adopted for GRFs estimation during double stance.

Moreover, an internal model of the lumber joint will be integrated with the full body model to estimate the lower back muscle forces and the lumber disk contact force. Finally, the model and ergonomic assessment suite will be validated experimentally and deployed to study loads and risk levels experienced by masons during bricklaying.

Chapter 2

Literature Review

2.1 Kinematics and Dynamics from IMUs

To predict joints' kinetics and tissue forces in the human body during some movement, inverse dynamics should be applied. In order to do this, task kinematics should be captured to be used as an input to the model. The OMC system is the gold standard for collecting human kinematics. However, the OMC systems require laboratory settings, they are not portable, and they only cover a limited space which restricts the subjects' behavior [13]. A laboratory-free portable system allow for *in situ* motion capture. This is advantageous because it reduces the complexity of collecting task kinematics and allows for more natural movements compared to a lab environment [17, 18].

An alternative to OMC systems is the use of IMUs. Each IMU consists of an accelerometer, a gyroscope, and a magnetometer. Recent developments have increased the accuracy and reduced the size and cost of these unites [12]. IMUs can be combined with a model of the human body to get a complete IMC system that measures segments orientation, angular velocity and acceleration for all body segments [19, 4]. The kinematics of different joints obtained from the IMC system were validated in previous studies [20, 21, 22, 23, 24], but using these kinematics to predict joints kinetics had only been investigated in a few studies.

Kim and Nussbaum [13] compared the shoulders, knees, hips, and L5/S1 joint angles and moments calculated by an inverse dynamics models based on kinematic data obtained from an OMC system to those based on kinematic data obtained from an IMC system. In both cases they measured GRFs using FPs. The mean absolute error (MAE) between the two sets of results across various manual material handling tasks was less than 5.85° for joint angles and less than 16.4 $N \cdot m$ for joint moments. One limitation to this approach is its dependence on FPs for GRFs measurements, which undermines the portability of IMUs systems.

Faber et al. [11] compared the L5/S1 joint moment estimated by a top-down model based on motions obtained from an IMC system to a top-down model based on motions obtained from an OMC system, and a bottom-up model based on motions obtained from the OMC system and the measured GRFs during trunk bending. They also compared the total GRFs estimated by the IMC-based model to those measured directly by FPs. All results were in close agreement for the dominant (vertical) component of the GRFs and the dominant (flexion-extension) component of the L5/S1 moment with Root-Mean-Square Error (RMSE) less than 5% of peak values. The other components of the GRFs and L5/S1 moment had larger relative errors. The model didn't breakdown the total GRFs into right-foot and left-foot contributions, therefore, it cannot be used to study the lower extremities' kinetics.

Karatsidis et al. [12] presented the only attempt to predict the breakdown of the total GRF into right and left GRFs using IMC-based inverse dynamics. They broke down the GRFs during gait motions into right-foot and left-foot contributions using a 'smooth transition assumption'. For normal walking speed, the relative Root-Mean-Square Error (rRMSE) of their estimate compared to FPs measurements was 5.3%, 9.4%, and 12.4% for the vertical, anterior, and lateral directions, respectively.

2.2 Decomposition of Ground Reaction Forces

Estimating GRFs during activities that don't involve double support phase, such as running, can be solved analytically [25, 26, 27]. The problem becomes statically indeterminate during double stance. Therefore, the current practice is to measure them directly using two FPs. But the use of IMC systems in the last few years opened up the possibilities for ambulatory studies of human motion. Since FPs are limited to in-lab use, require a dedicated space [28], and can alter normal motion pattern they are not suitable for tasks that require large spaces or occur outside lab environments. Two alternatives to overcome this challenge have been proposed and used: using insoles-embedded wearable pressure sensors for GRFs measurement [29] and estimating GRFs using kinematic data only.

An early attempt to tackle this problem [16] had proposed a general procedure to solve indeterminate closed loop problems by postulating that the human neuromuscular system acts to minimize muscular effort. Therefore, they proposed an optimization solution where the objective function minimizes the sum of the squares of the net moment for all joints in the closed loop. For problems involving the GRFs, this would include right and left ankles, knees and hips. Using this optimization procedure to solve for the GRFs on the right and left feet, they found that error ranged from 5 % to 25 % from the value of the vertical GRF and from 3 % to 46 % for the value of the horizontal GRF. The advantage of their approach is that it is not task-specific, rather it can be used for any generic motion.

Ren et al. [5] proposed the use of a 'smooth transition functions'; semi-empirical functions that predict the ratio of the two feet GRFs during the double support phase of the gait cycle. They assumed that the tailing foot vertical (y) and lateral (z) components of the GRF drop smoothly to zero as it leaves ground according to the function

$$\frac{F_{yz}}{F_{yz\circ}} = e^{(t/T_{dS})^3} \tag{2.1}$$

where F_y and F_z are the predicted forces at time t, $F_{y\circ}$ and $F_{z\circ}$ are the total forces. They also assumed that the anterior-posterior component (x) of the GRF drop smoothly to zero according to the function

$$\frac{F_x}{F_{x0}} = k_1 e^{-[(t/T_{ds})-(2/3)]^2} - k_2 \frac{t}{T_{ds}} \tag{2.2}$$

where F_x is the predicted force, F_{x0} is the total force, T_{ds} is half the period of the double support phase, and k_1 and k_2 are constants determined by the boundary condition at heel strike and toe off.

The function results agreed well with the measured values from FPs in the sagittal plane. The estimations were less accurate in the other two planes. The rRMSE was less than 1% for the vertical force (best) and 9.4% for the transverse plane moment (worst). Later study [6] used larger dataset and showed that the smooth transition function was less accurate, with the rRMSE ranging from 6.9% for the vertical forces to 35.5% for the frontal plane moment.

Oh et al. [6] used a neural network to predict GRFs from motion kinematics only during gait. They collected kinematic data along with GRFs measurements to train and test the algorithm. The rRMSE of the predicted vertical force was 4.7% and the rRMSE of the predicted frontal plane moment was 29.4%. Both of these studies used the FPs measurements to detect the start and end of the double support phase which undermines the value of their approaches as alternatives to FPs.

Karatsidis et al. [12] proposed a gait event detection algorithm to overcome this shortcoming. It predicts foot contact with the ground based on the norms of the heel velocity $||V_{heel}||$ and the toe velocity $||V_{toe}||$ compared to a threshold velocity V_{th}. They used this algorithm in combination with a modified smooth transition function to calculate the right and left GRFs and found that the rRMSE of their predictions ranged from 5.3% for the vertical forces to 29.6% for the sagittal plane moment.

2.3 Internal Forces in the Lumbar Spine

The models described so far are external, they don't account for the breakdown of forces among the internal tissues at each joint. While external models can predict the joint's net moment and force, it is necessary to build anatomically detailed internal models to estimate muscles and ligament forces as well as joint contact forces. Since the joint's net moment is directly related to muscles activities required to meet the load demands, it is a good indicator of the loading on the joint internal tissue. However, it is insufficient in situations that require further insight into the loading sharing among the joint support structures. For example, studying ligament rupture requires estimates of ligament loads and strains and studying joint arthroplasty requires estimates the joint contact forces.

The lumbar spine joints, especially the lower back L4/L5 and L5/S1 joints, are subject to large external moments during heavy material handling which causes large compression forces on the lumber disks as the lower back muscles contract to balance it. Those compression forces were first measured in 1970 [30] by pressure transducers inserted at the center of the disc. Later studies using Magnetic Resonance Imaging (MRI) have shown that degeneration of L4/L5 and L5/S1 disks are a serious risk for occupations that involve heavy lifting tasks [31, 32, 33]. Therefore, it is important to study the lower back disk's compression and shear forces.

The first attempt to make an anatomical model of lower back disks was a single equivalent muscle model [34], which has the advantage of being statically determinant. Early models were 2-dimensional, in the sagittal plane, and assumed that the Erector Spinae (ES) muscle is the only active muscle delivering the net moment in the sagittal plane. These models considered the abdominal pressure to act on a fixed diaphragm area along a line of action parallel to the disk compression force. The ES muscle line of action was also assumed parallel to the disk compression force. The moment arm was assumed to be 5 cm in earlier models and increased to 5.3 to 7.0 cm in more recent models [35]. However, studies on the strength of lumber disks showed that the compression forces predicted by those older models were unrealistic, leading to micro-fractures in the disks' cartilage [35]. Another shortcoming of these model is that they do not predict the lateral and anterior shear forces of the disk and do not include other muscles with lines of action not parallel to the disk normal.

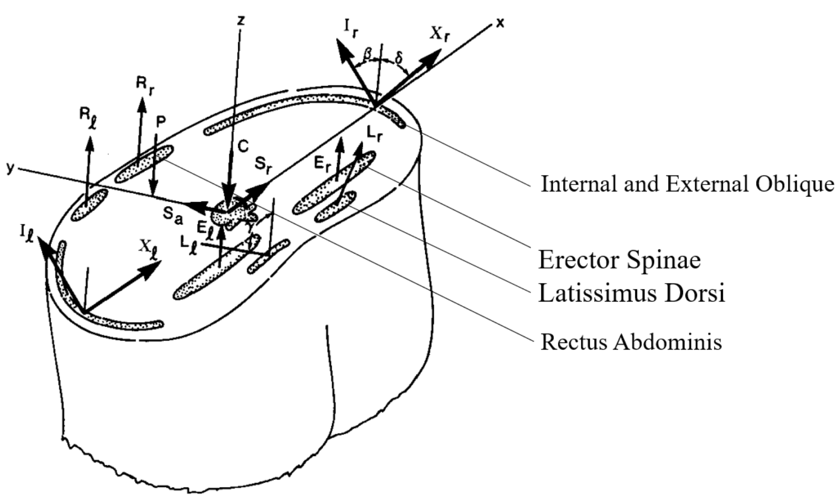


Figure 2.1: Schultz and Andersson's 10-muscle model [1]. Used with permission

Schultz and Andersson introduced a more complex 3-dimensional model [36] that accounts for the right and left Erector Spinae (ES), Rectus Abdominis (RA), Latissimus Dorsi (LD), Internal Oblique (IO), and External Oblique (EO) muscles, Figure 2.1, as well as the abdominal pressure. The model solves for 13 unknowns: 10 muscles forces, the disk compression force, lateral shear, and anterior-posterior shear. With only 6 equations of motion, the problem is statically indeterminate.

Two consecutive optimization problems were proposed [37] and refined [38] as a method to solve this indeterminate problem. This first optimization problem minimizes the maximum muscle intensity (muscle force per unit area) across all ten muscles. Since the solution of this problem is not unique, the second problem minimizes the summation of the absolute muscles forces subject to a constraint maintaining the maximum muscle intensities found in the first optimization problem.

These model can be applied to either L3/L4 [36], L4/L5 [39], or L5/S1 joints. For instance, 3DSSPP uses the 10-muscle model [36] to evaluate the contact forces in the L4/L5 joint and a single equivalent muscle model [34] to evaluate the contact forces in the L5/S1 joint. As the joint contact forces 3DSSPP finds for these joints are evaluated using different models, they can not be compared quantitatively.

2.4 Occupational Risk Assessment

Various metrics can be implemented to assess the biomechanical risks encountered by workers during their tasks. These metrics may be derived from body kinematic parameters, kinetic parameters, external loads, and/or the duration and frequency of exposure to those loads [40]. Body kinematic parameters, namely joint angular positions, velocities, and accelerations and segment center of mass accelerations, can be measured directly by motion capture systems. Gross kinetic parameters, namely net joint forces and moments, can be evaluated from full body external models using inverse dynamics. Detailed kinetic parameters derived from internal joint models, namely muscle forces and joint contact forces, can also be used to evaluate risk levels within critical joints. Cumulative metrics, that represent an integral of some of the preceding parameters, can also be used to evaluate risk levels where the body is exposed to loads frequently or for extended periods of time.

The state-of-the-art in task risk assessment are composite metrics, such as RULA [41], REBA [42], and OWAS [43], designed to qualitatively gauge risk levels via combinations of body posture (joint angles), external loads, and their frequencies (repetitions). While these measures have a direct relationship to the kinetic loads, they do not consider velocities or accelerations, which were found to be good indicators of risk when combined with kinetic data [44]. Moreover, these metrics were shown to saturate for manual handling tasks such as bricklaying [45] indicating high-risk levels regardless of the actual loads developing in the lower spine. Therefore, they do not appear to accurately represent the loads acting on the body.

Kinetic parameters can be used as a direct (quantitative) indicator of risk. Peak values, average values, or integrated values have been used for that purpose. However, Norman et al. [46] showed that estimating risk levels in the lumbar spine requires to require a measure of peak static loads in spine joints, a measure of dynamic loads, represented by the trunk velocity, and an integral measure of the duration and frequency of exposure to those loads.

Traditionally, kinetic parameters were derived from static analysis, without considering the inertial forces caused by motion. Studies have found that those parameters are not adequate detectors of risk levels on their own and must be augmented with measures of dynamic loads, such as joint velocities [44]. The need to account for kinematic parameters in conjugation with static kinetic parameters to predict MSDs represent a deficiency in the static estimates of body loads and suggest a need for a new approach to risk assessment that explicitly evaluates the full dynamic loads experienced by the body during work tasks.

Chapter 3

Full Body Model

To determine the biomechanical loads on different body parts during motion, the human body can be modeled as a multibody system; where each body segment is modeled as a rigid body with the joints connecting different segments. This chapter describes model developed in this study and the method used to solve the inverse dynamics problem of this model.

A 15-segment body model, Figure 3.1, is adopted to make use of all the accelerations and joint angles obtained from the IMC system. The model is composed of the pelvis, the torso, the right and left upper arms, the right and left lower arms, the right and left hands, the right and left thighs, the right and left legs, and the right and left feet. The pelvis is taken as the root segment. The segments are numbered going from the pelvis to the neck

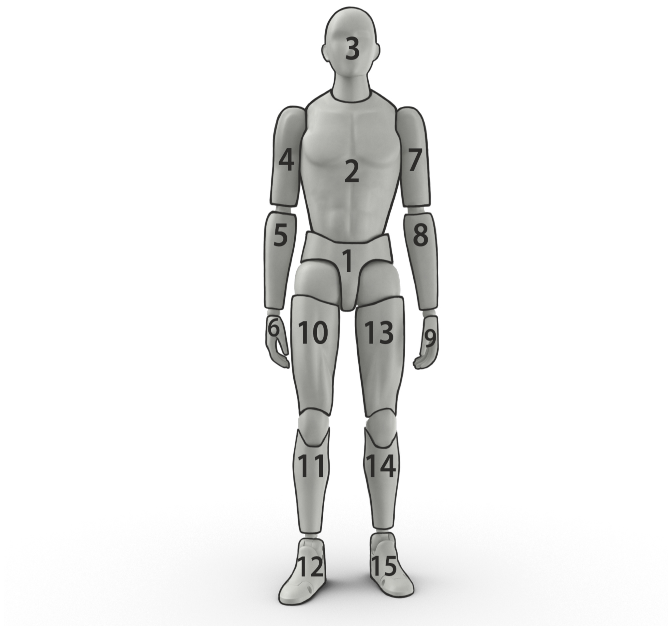


Figure 3.1: The 15-segment full body model

and head segment and from the pelvis towards the distal segment of each limb.

The higher-order lower body arrays of the model are constructed to relate each segment to its proximal segment. This chain is used to solve the inverse dynamics problem using a top-down approach for the upper limbs and a bottom-up approach for the lower limbs.

3.1 Inertial Properties

The inertial parameters of each segment, namely segment length, mass, the center of mass, and the mass moment of inertia tensor, are obtained from the literature [2] as functions

Table 3.1: The higher-order lower body arrays for the full body model

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$L(i)$	0	1	2	2	4	5	2	7	8	1	10	11	1	13	14
$L^2(i)$	0	0	1	1	2	4	1	2	7	0	1	10	0	1	13
$L^3(i)$	0	0	0	0	1	2	0	1	2	0	0	1	0	0	1
$L^4(i)$	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0
$L^5(i)$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

of the subject height and mass. To use those parameters consistently, the model follows Dumas et al. [2] definitions of the segment and their anatomical landmarks as shown in Figure 3.2. The segment mass m^i is defined as a percentage of the body mass, the segment length is defined as a percentage of height, the center of mass location $\boldsymbol{r}^i$ with respect to the local frame origin, expressed in the local frame $\boldsymbol{\mathcal{F}}^i$ is defined as a percentage of the segment length, and the inertia tensor $[\boldsymbol{J}^i]$ in the local frame (Coordinate System (CS)) $\boldsymbol{\mathcal{F}}^i$ is defined as a function of the segment mass and length.

The origins of the local frames are placed at the centers of the segment proximal joint except for the root segment frame, pelvis, whose origin is placed at the Lumbar Joint Center (LJC). Table 3.2 defines the length of each segment in terms of its anatomical landmarks and the origin of its local frame. The axes of the local frames are labeled following the recommendations of the International Society of Biomechanics (ISB) [47, 48] with the Y-axis running through the length of the segments pointing cranially, the Z-axis

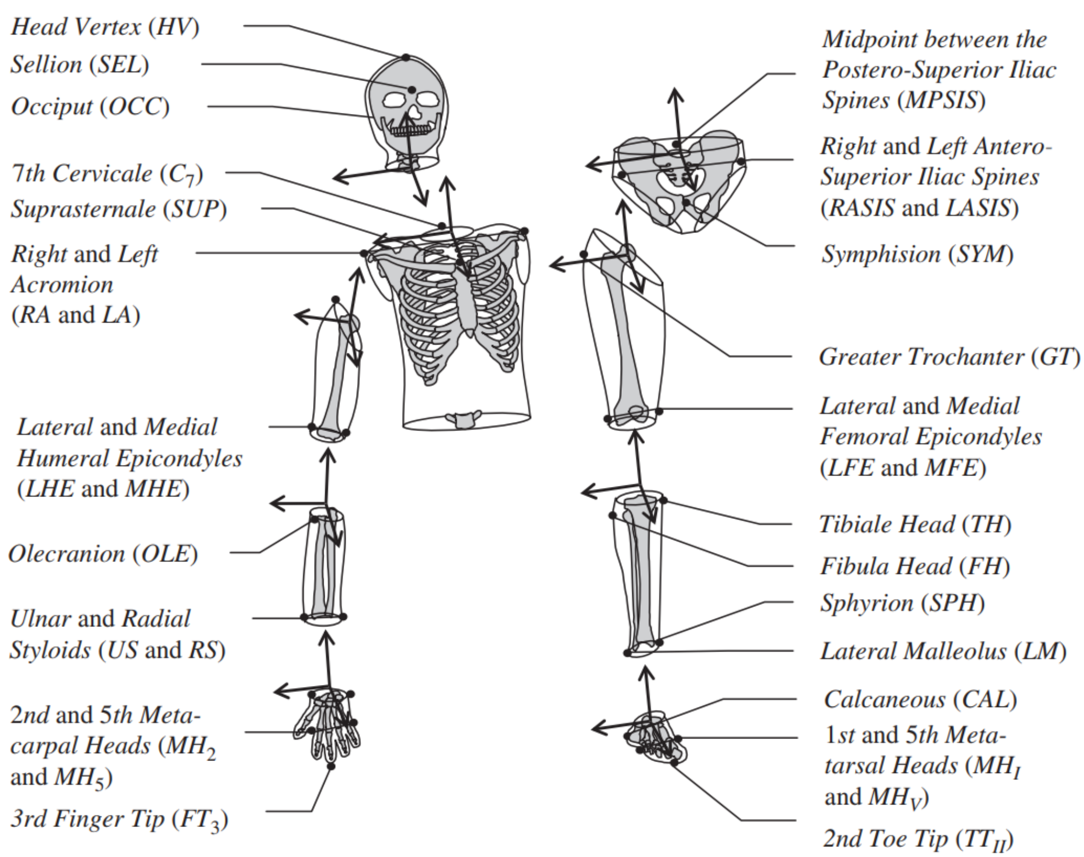


Figure 3.2: Locations of the anatomical landmarks [2]. Used with permission

Table 3.2: Body segments' length and local frame origins

Segment	Length	Origin
Pelvis	LJC to HJC projection in sagittal plane	Lumbar Joint Center (LJC)
Torso	CJC to LJC	Lumbar Joint Center (LJC)
Head & Neck	CJC to HV	Cervical Joint Center (CJC)
Arm	SJC to EJC	Shoulder Joint Center (SJC)
Forearm	EJC to WJC	Elbow Joint Center (EJC)
Hand	WJC to midpoint between MH2 and MH5	Wrist Joint Center (WJC)
Thigh	HJC to KJC	Hip Joint Center (HJC)
Leg	KJC to AJC	Knee Joint Center (KJC)
Foot	AJC to midpoint between MH1 and MH5	Ankle Joint Center (AJC)

running laterally and pointing to the right, and the X-axis formed by the cross product of the Y and Z-axes. The definitions of the local axes are shown in Table 3.3.

3.2 Model Kinematics

In order to write the multibody system's equations of motion, we defined the kinematic relationships among the body segments in terms of the kinematic data obtained from the IMC system. The location of the pelvic frame origin $\boldsymbol{\zeta}^1(t)$ with respect to the reference frame $\mathcal{F}^0$ origin is obtained directly from the IMC system. The origins of all other body frames $\boldsymbol{\zeta}^i$ are expressed with respect to the origin of the parent (proximal) segment frame

Table 3.3: Definitions of segments body frame directions

Segment	X-Axis
Pelvis	cross product between Y- & Z-axes
Torso	cross product between Y- & Z-axes
Head and Neck	cross product between Y- & Z-axes
Arm	points anteriorly normal to the plane containing SJC, LHE, and MHE
Forearm	points anteriorly normal to the plane containing EJC, US, and RS
Hand	points anteriorly normal to the plane containing WJC, MH2, and MH5
Thigh	points anteriorly normal to the plane containing HJC, LFE, and MFE
Leg	points anteriorly normal to the plane containing KJC, the AJC and the Fibula Head (FH)
Foot	the vector from the Calcaneous (CAL) to the midpoint between MHI and MHV

Segment	Y-Axis
Pelvis	from the LASIS to the RASIS
Torso	the vector from the LJC to the CJC
Head and Neck	the vector from the CJC to the HV
Arm	the vector from the EJC to the SJC
Forearm	the vector from the WJC to the EJC
Hand	the vector from the midpoint between MH2 and MH5 to the WJC
Thigh	the vector from the KJC to the HJC
Leg	the vector from the AJC to the KJC
Foot	points cranially normal to the plane containing CAL, MHI, and MHV

Segment	Z-Axis
Pelvis	points cranially normal to the plane containing LASIS, RASIS, and MPSIS
Torso	points laterally normal to the plane containing LJC, CJC, and SUP
Head and Neck	points laterally normal to the plane containing HV, CJC, and SEL
Arm	cross product between X- & Y-axes
Forearm	cross product between X- & Y-axes
Hand	cross product between X- & Y-axes
Thigh	cross product between X- & Y-axes
Leg	cross product between X- & Y-axes
Foot	cross product between X- & Y-axes

$\boldsymbol{\mathcal{F}}^k$. They are estimated from the anthropometric parameters of the subject.

The rotation matrix $[{}^0\boldsymbol{R}^i](t)$ from segment i frame $\boldsymbol{\mathcal{F}}^i$ to the reference frame $\boldsymbol{\mathcal{F}}^0$ is evaluated as [49] :

$$[{}^0\boldsymbol{R}^i](t) = \begin{bmatrix} {\epsilon_1^i}^2 + {\epsilon_2^i}^2 - {\epsilon_3^i}^2 - {\epsilon_4^i}^2 & 2(\epsilon_2^i\epsilon_3^i - \epsilon_1^i\epsilon_4^i) & 2(\epsilon_2^i\epsilon_4^i + \epsilon_1^i\epsilon_3^i) \\ 2(\epsilon_2^i\epsilon_3^i + \epsilon_1^i\epsilon_4^i) & {\epsilon_1^i}^2 - {\epsilon_2^i}^2 + {\epsilon_3^i}^2 - {\epsilon_4^i}^2 & 2(\epsilon_3^i\epsilon_4^i - \epsilon_1^i\epsilon_2^i) \\ 2(\epsilon_2^i\epsilon_4^i - \epsilon_1^i\epsilon_3^i) & 2(\epsilon_3^i\epsilon_4^i + \epsilon_1^i\epsilon_2^i) & {\epsilon_1^i}^2 - {\epsilon_2^i}^2 - {\epsilon_3^i}^2 + {\epsilon_4^i}^2 \end{bmatrix} \tag{3.1}$$

where the segment attitude is obtained from the IMC system as the components of the unit quaternion $\epsilon_1^i(t), \epsilon_2^i(t), \epsilon_3^i(t)$ and $\epsilon_4^i(t)$. Similarly, the rotation matrix $[{}^k\boldsymbol{R}^i](t)$ from segment i frame $\boldsymbol{\mathcal{F}}^i$ to its parent segment k frame $\boldsymbol{\mathcal{F}}^k$ is evaluated as

$$[{}^k\boldsymbol{R}^i](t) = [{}^k\boldsymbol{R}^0] \cdot [{}^0\boldsymbol{R}^i] \tag{3.2}$$

where $[{}^k\boldsymbol{R}^0] = [{}^0\boldsymbol{R}^k]^T$.

These rotation matrices are used in conjunction with segment lengths (Table 3.2) to define the location of the segment frame origin with respect to the parent segment frame origin $\boldsymbol{\zeta}^i$. We collocate the origin of each segment frame with that of its proximal joint center as listed in Table 3.2.

The segment angular acceleration $\boldsymbol{\alpha}^i(t)$ is evaluated by numerically differentiating its angular velocity vector $\boldsymbol{\omega}^i(t)$, obtained from the IMC system, using a single step forward difference

$$\boldsymbol{\alpha}^i(t) = \frac{d\boldsymbol{\omega}^i(t)}{dt} \quad \Rightarrow \quad \boldsymbol{\alpha}^i(t_n) \approx \frac{\boldsymbol{\omega}^i(t_{n+1}) - \boldsymbol{\omega}^i(t_n)}{t_{n+1} - t_n} \tag{3.3}$$

The acceleration of the segment's center of mass $\boldsymbol{a}^i(t)$ is evaluated from the measured segment-fixed IMU acceleration $\boldsymbol{a}_s^i$(t), the location of the IMU with respect to the center of mass $\boldsymbol{r}_s^i(t)$, and the segment angular velocity $\boldsymbol{\omega}^i(t)$ and acceleration $\boldsymbol{\alpha}^i(t)$ as:

$$\boldsymbol{a}^i(t) = \boldsymbol{a}_s^i + \boldsymbol{\alpha}^i \times \boldsymbol{r}_s^i + \boldsymbol{\omega}^i \times (\boldsymbol{\omega}^i \times \boldsymbol{r}_s^i) \tag{3.4}$$

3.3 The Inverse Dynamics Problem

The forces applied to body segments are either external to the body or internal to it. External forces $\boldsymbol{F}_{ex}^i(t)$ and moments $\boldsymbol{M}_{ex}^i(t)$, such as hand loads and GRFs, are not measured by IMC systems. Instead, they are determined based on the specifications of the task under analysis, measured using additional force sensors, or estimated from body kinematics. The internal forces and moments are: the net joint forces and moments, segment weight, and the inertial forces and moments.

The goal of the inverse dynamics problem is to determine the net joint forces and moments given the segments' inertial properties, body motions, and the external forces and moments. Each segment's equations of motion can be used to evaluate the net force and moment of in its proximal joint. Starting from the most distal segment, where the only joint is proximal, and propagating inward toward the root segment (pelvis), we can find for all joint forces and moments. Therefore, a top-down approach is taken for upper body segments and a bottom-up approach is taken for the lower body segments.

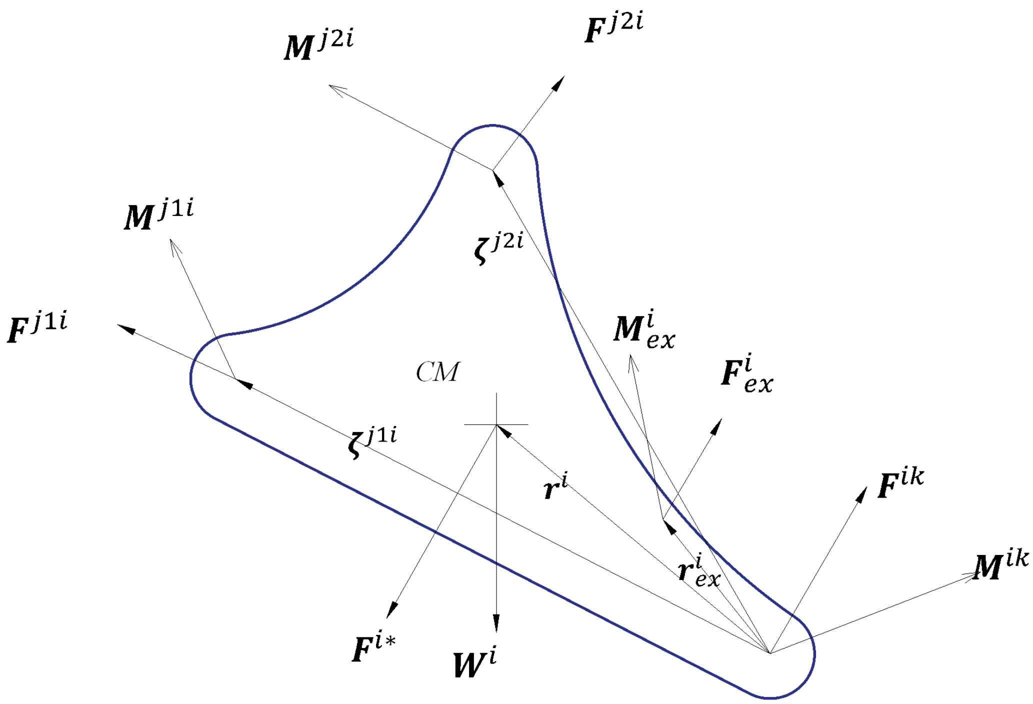


Figure 3.3: The free body diagram of segment i

Let i be the segment of interest, k be the proximal segment to i (i.e. $L(i) = k$), $j1$ & $j2$ be two distal segments to i (i.e. $L(j1) = L(j2) = i$). Note that more than one segment can be distal to the same segment, where multiple branches originate from it. For instance, the right upper arm, left upper arm, and head segments are all distal to the torso.

The free body diagram for segment i is shown in Figure 3.3. The forces acting on the segment are the net force in the proximal joint $\boldsymbol{F}^{ik}$, the sum of the net forces in the distal joints $\sum_n \boldsymbol{F}^{jni}$, the total external force acting on the segment $\boldsymbol{F}^{i}_{ex}$, the segment weight $\boldsymbol{W}^{i}$, and the inertial force of the segment:

$$\boldsymbol{F}^{i\star} = -m^i \boldsymbol{a}^i \tag{3.5}$$

Using D'Alembert's principle, Newton's Law for this segment can be written as:

$$\boldsymbol{F}^{ik} + \sum_n \boldsymbol{F}^{jni} + \boldsymbol{F}^{i}_{ex} + \boldsymbol{W}^{i} + \boldsymbol{F}^{i\star} = 0 \tag{3.6}$$

Similarly, the moments acting on the center of mass are the net moment $\boldsymbol{M}^{ik}$ of the proximal joint k, the moment produced by the net force of the proximal joint about the center of mass $-\boldsymbol{r}^i \times \boldsymbol{F}^{ik}$, the sum of the net moments of the distal joints $\sum_n \boldsymbol{M}^{jni}$, the moments produced by the net forces of the distal joints about the center of mass $\sum_n [(\boldsymbol{\zeta}^{jn} - \boldsymbol{r}^i) \times \boldsymbol{F}^{jni}]$, the total external moment $\boldsymbol{M}^{i}_{ex}$ acting on the segment, the moment produced by the total external force acting on the segment about its center of mass $[(\boldsymbol{r}^{i}_{ex} - \boldsymbol{r}^i) \times \boldsymbol{F}^{i}_{ex}]$, and the inertial moment $\boldsymbol{M}^{i\star}$ of the segment around its center of mass

$$\boldsymbol{M}^{i\star} = -[\boldsymbol{J}^i] \cdot \boldsymbol{\alpha}^i - \boldsymbol{\omega}^i \times ([\boldsymbol{J}^i] \cdot \boldsymbol{\omega}^i) \tag{3.7}$$

Euler's law can, therefore, be expressed around the segment center of mass as:

$$\begin{aligned} &\boldsymbol{M}^{ik} + (-\boldsymbol{r}^i \times \boldsymbol{F}^{ik}) + \sum_n \boldsymbol{M}^{jni} + \sum_n ((\boldsymbol{\zeta}^{jn} - \boldsymbol{r}^i) \times \boldsymbol{F}^{jni}) \\ &+ \boldsymbol{M}^i_{ex} + ((\boldsymbol{r}^i_{ex} - \boldsymbol{r}^i) \times \boldsymbol{F}^i_{ex}) + \boldsymbol{M}^{i\star} = 0 \end{aligned} \tag{3.8}$$

where $\boldsymbol{\zeta}^{jn}$ is the location of frame $\mathfrak{F}^{jn}$ origin with respect to frame $\mathfrak{F}^i$ origin and $\boldsymbol{r}^i_{ex}$ is the location of the point of action of the external force with respect to frame $\mathfrak{F}^i$ origin.

The two equations of motion (3.6) and (3.8) can be solved for the proximal joint's net force $\boldsymbol{F}^{ik}$ and moment $\boldsymbol{M}^{ik}$, since all other variables in the equations are body parameters, measured motions, known external forces, or have been determined via the distal segment's equations of motion. First, each component of the vector equation (3.6) is solved for the corresponding component of the proximal joint's net force. These forces are then substituted into the vector equation (3.8) and each of its components is solved for the corresponding proximal joint's net moment.

3.4 Lower-Back Disk Contact Forces

Given the net joint force and moment, it is possible to determine the internal forces within a joint, including muscle, ligament, and contact forces. This is an indeterminate problem for most joints since the number of forces involved typically exceeds the number of equations of motion for a single segment. In our case, we are interested in labor tasks involving manual handling. The internal forces within the lower back joints are of critical importance for this class of tasks. Specifically, Alwasel et al. [50] found that the L5/S1 joint was the most heavily loaded joint during bricklaying tasks. Therefore, we are interested in developing an anatomically valid model that can determine the individual muscle forces and disk contact forces, namely the disk anterior-posterior shear, lateral shear, and compression forces, of the L5/S1 joint.

Towards this end, we adopt the 3-dimensional, 10-muscle model developed by Schultz and Andersson [36]. The model includes the right and left Erector Spinae (ES), Rectus Abdominis (RA), Latissimus Dorsi (LD), Internal Oblique (IO), and External Oblique (EO) muscles, Figure 2.1, and the abdominal pressure exerted on the thoracic diaphragm. The muscle forces are defined in terms of the muscle intensity I_n and Physiological Cross-Sectional Area (PCSA) A_n as:

$$F_n = A_n I_n \tag{3.9}$$

The muscles' moment arms, PCSA, and their line-of-action in the joint frame are obtained from [51, 52, 38, 53].

We followed Bean et al. [38] successive linear optimization approach to solve the indeterminate force distribution problem. Given the net joint force and moment and the abdominal pressure, evaluated as a function of the torso flexion angle, we solved a first linear optimization problem to find the maximum muscle intensity required to balance the exerted moment and a second linear optimization problem to find the muscle and contact forces subject to the maximum muscle intensity found.

Linear optimization finds optimal values for a vector of variables $\boldsymbol{x}$ that satisfy an objective function $\boldsymbol{f}(\boldsymbol{x})$, a set of equality constraints:

$$[\boldsymbol{C}_{eq}]\boldsymbol{x} = \boldsymbol{b}_{eq} \tag{3.10}$$

where $[\boldsymbol{C}_{eq}]$ is a linear matrix and $\boldsymbol{b}_{eq}$ is an equality constraint vector, and a set of inequality constraints:

$$[\boldsymbol{C}_{in}]\boldsymbol{x} \leq \boldsymbol{b}_{in} \tag{3.11}$$

where $[\boldsymbol{C}_{in}]$ is a linear matrix and $\boldsymbol{b}_{in}$ is an inequality constraint vector, within boundaries:

$$\boldsymbol{lb} \leq \boldsymbol{x} \leq \boldsymbol{ub} \tag{3.12}$$

where $\boldsymbol{lb}$ and $\boldsymbol{ub}$ are the lower and upper bounds of the solution in parameter space.

First Linear Optimization Problem

The first optimization problem finds optimal values for the magnitudes of the muscle forces, the components of the contact force, and the maximum muscle intensity I_{max}. The vector of variables is, thus, expressed as:

$$\boldsymbol{x} = \begin{bmatrix} F_{res} \\ F_{les} \\ F_{rld} \\ F_{lld} \\ F_{rra} \\ F_{lra} \\ F_{rio} \\ F_{lio} \\ F_{reo} \\ F_{leo} \\ F_c \\ F_{sl} \\ F_{sa} \\ I_{max} \end{bmatrix} \tag{3.13}$$

where F_{res} and F_{les} are the right and left ES muscle forces, F_{rld} and F_{lld} are the right and left LD muscle forces, F_{rra} and F_{lra} are the right and left RA muscle forces,, F_{rio} and F_{lio} are the right and left IO muscle forces, F_{reo} and F_{leo} are the right and left EO muscle forces, F_c is the disk compression force, F_{sl} is the disk lateral shear, and F_{sa} is the disk anterior shear. This vector is optimized subject to an objective function that minimizes the maximum muscle intensity

$$\min\{I_{max}\} \tag{3.14}$$

The equality constraints impose force balance among the muscle forces, the net joint force $\boldsymbol{F}_{net}$, the contact force, and the abdominal pressure force F_{ap}:

$$-(F_{rld} + F_{lld})\sin\theta_{ld} + F_{sl} = F_{net_x} \tag{3.15}$$

$$\begin{aligned}(F_{res} + F_{les})\sin\theta_{es} + (F_{rra} + F_{lra})\sin\theta_{ra} + (F_{rio} + F_{lio})\sin\theta_{io} \\ + (F_{reo} + F_{leo})\sin\theta_{eo} + F_{sa} = F_{net_y}\end{aligned} \tag{3.16}$$

$$\begin{aligned}-(F_{res} + F_{les})\cos\theta_{es} - (F_{rld} + F_{lld})\cos\theta_{ld} - (F_{rra} + F_{lra})\cos\theta_{ra} \\ - (F_{rio} + F_{lio})\cos\theta_{io} - (F_{reo} + F_{leo})\cos\theta_{eo} + F_c = F_{net_z} - F_{ap}\end{aligned} \tag{3.17}$$

where θ_{es}, θ_{ra}, θ_{io}, and θ_{eo} are the angles between the ES, RA, IO, and EO muscles line-of-action and the normal to the L5/S1 disk surface in the sagittal plane, θ_{ld} is the angle between the LD muscle line-of-action and the normal to the L5/S1 disk surface in the coronal plane and F_{net_x}, F_{net_y}, and F_{net_z} are the net joint force components.

The equations are expressed in a joint coordinate system, Figure 2.1. The origin of the coordinate system is located at the joint center, the x-axis points to the right along a mediolateral direction, the z-axis points along the direction of the normal to the disk surface, and the y-axis points in the direction of their cross-product to form a right-handed frame.

The equality constraints also impose moment balance among the net joint moment $\boldsymbol{M}_{net}$ and the moments of muscle forces and the abdominal pressure force taken around the joint center and written in the joint coordinate system:

$$\begin{aligned} & r_{es}^s(F_{res} + F_{les})\cos\theta_{es} + r_{ld}^s(F_{rld} + F_{lld})\cos\theta_{ld} + r_{ra}^s(F_{rra} + F_{lra})\cos\theta_{ra} \\ & + r_{io}^s(F_{rio} + F_{lio})\cos\theta_{io} + r_{eo}^s(F_{reo} + F_{leo})\cos\theta_{eo} = M_{net_x} - r_{ap}^s F_{ap} \end{aligned} \tag{3.18}$$

$$\begin{aligned} & r_{es}^c(F_{res} - F_{les})\cos\theta_{es} + r_{ld}^c(F_{rld} - F_{lld})\cos\theta_{ld} + r_{ra}^c(F_{rra} - F_{lra})\cos\theta_{ra} \\ & + r_{io}^c(F_{rio} - F_{lio})\cos\theta_{io} + r_{eo}^c(F_{reo} - F_{leo})\cos\theta_{eo} = M_{net_y} \end{aligned} \tag{3.19}$$

$$\begin{aligned} & r_{es}^c(F_{res} - F_{les})\sin\theta_{es} + r_{ld}^c(F_{rld} - F_{lld})\sin\theta_{ld} + r_{ra}^c(F_{rra} - F_{lra})\sin\theta_{ra} \\ & + r_{io}^c(F_{rio} - F_{lio})\sin\theta_{io} + r_{eo}^c(F_{reo} - F_{leo})\sin\theta_{eo} = M_{net_z} \end{aligned} \tag{3.20}$$

where M_{net_x}, M_{net_y}, and M_{net_z} are the components of the net joint moment, r_{es}^s, r_{ld}^s, r_{ra}^s, r_{io}^s, and r_{eo}^s are the ES, LD, RA, IO, and EO muscles moment arms in the sagittal plane, r_{es}^c, r_{ld}^c, r_{ra}^c, r_{io}^c, and r_{eo}^c are the ES, LD, RA, IO, and EO muscles moment arms in the coronal plane, and r_{ap}^s is the abdominal pressure force moment arm in the sagittal plane. The abdominal pressure is assumed to act parallel to the disk normal. Equations (3.15)–(3.20) are used to obtained the linear equality matrix $[\boldsymbol{C}_{eq}]$ and vector $\boldsymbol{b}_{eq}$.

The inequality constraints are defined to constrain the muscle forces so that they do not exceed the maximum muscle intensity I_{max}

$$\begin{aligned}
F_{res} &\leq A_{es} I_{max} & F_{les} &\leq A_{es} I_{max} \\
F_{rld} &\leq A_{ld} I_{max} & F_{lld} &\leq A_{ld} I_{max} \\
F_{rra} &\leq A_{ra} I_{max} & F_{lra} &\leq A_{ra} I_{max} \\
F_{rio} &\leq A_{io} I_{max} & F_{lio} &\leq A_{io} I_{max} \\
F_{reo} &\leq A_{eo} I_{max} & F_{leo} &\leq A_{eo} I_{max}
\end{aligned} \tag{3.21}$$

where A_{es}, A_{ld}, A_{ra}, A_{io}, and A_{eo}, the PCSAs of the ES, LD, RA, IO, and EO muscles, are held constant. These constrains bound the maximum muscle forces by I_{max}, which is minimized by the objective function. Equation (3.21) is used to obtain the linear inequality matrix $[\boldsymbol{C}_{in}]$ and vector $\boldsymbol{b}_{in}$.

The muscles forces act only in tension, therefore, only their lower bound is set to zero. The compression force and muscle intensity can only take positive values, whereas the shear forces may take positive or negative values and, therefore, are unbounded. The lower and upper boundaries vectors are, thus, defined as:

$$
\boldsymbol{lb} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\infty \\ -\infty \\ 0 \end{bmatrix} \qquad \boldsymbol{ub} = \begin{bmatrix} \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \end{bmatrix} \tag{3.22}
$$

The objective function $\boldsymbol{f}(\boldsymbol{x})$, matrices $[\boldsymbol{C}_{eq}]$ and $[\boldsymbol{C}_{in}]$, the vectors $\boldsymbol{b}_{eq}$ and $\boldsymbol{b}_{in}$, and the boundaries of variables (3.22) are defined in Matlab function **linprog**, which is used to solve for the vector of variables $\boldsymbol{x}$. The solution vector is not unique, since the forces in those muscles with intensity less than I_{max} can be redistributed to satisfy the force and moment balance. Therefore, a second optimization is defined subject to the maximum muscle intensity I_{max}.

Second Linear Programming Optimization

This optimization problem solves only for the muscle forces and the components of the joint contact force

$$\boldsymbol{x} = \begin{bmatrix} F_{res} \\ F_{les} \\ F_{rld} \\ F_{lld} \\ F_{rra} \\ F_{lra} \\ F_{rio} \\ F_{lio} \\ F_{reo} \\ F_{leo} \\ F_{c} \\ F_{sl} \\ F_{sa} \end{bmatrix} \tag{3.23}$$

The objective function is to minimize the summation of the muscle forces

$$\min\{\Sigma_{n=1}^{10} x_n\} \tag{3.24}$$

The equality constraints are identical to those of the first problem. The lower boundaries are defined to ensure that the muscles only act in tension. On the other hand, the upper boundaries are defined to ensure that the maximum muscle forces do not exceed the limit defined by the maximum muscle intensity I_{max}.

$$\boldsymbol{lb} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\infty \\ -\infty \end{bmatrix}, \qquad \boldsymbol{ub} = I_{max} \begin{bmatrix} A_{es} \\ A_{es} \\ A_{ld} \\ A_{ld} \\ A_{ra} \\ A_{ra} \\ A_{io} \\ A_{io} \\ A_{eo} \\ A_{eo} \\ \infty \\ \infty \\ \infty \end{bmatrix} \tag{3.25}$$

No inequality constraints are imposed, since the boundaries maintain all intensities less than or equal to I_{max}.

The objective function (3.24), matrix $[\boldsymbol{C}_{eq}]$, vector $\boldsymbol{b}_{eq}$, and the boundaries of variables (3.25) are defined in Matlab function **linprog**, which is used to solve for the vector of variables $\boldsymbol{x}$. The solution vector defines unique values for the forces in the ten muscles and the components of the joint contact force.

Chapter 4

Estimation of Ground Reaction Forces

Ground Reaction Forces (GRFs) are often the largest external force acting on the body. Although GRFs can be measured using FPs, this is limited to small areas within laboratory environments. Alternatively, GRFs can be estimated from inverse dynamic analysis as the total reaction force at the ground. This is a determinate problem when solving for one force vector and its point of action. However, double stance postures require the evaluation of two distinct Ground Reaction Forces (GRFs), which is an indeterminate problem. We followed the approach proposed by Vaughan et al. [16] to estimate Ground Reaction Forces (GRFs) during the double stance. First, the total GRFs is calculated using inverse dynamics.

Second, a contact detection algorithm predicts each foot contact with the ground. Finally, an optimization problem is solved to estimate the breakdown of GRFs between the right and left feet when both are in contact with the floor.

4.1 Total Ground Reaction Force

We can rewrite Newton's second law for all body segments in terms of the total ground reaction force:

$$\boldsymbol{F_g} = \sum_{i=1}^{15} (\boldsymbol{F}^{i\star} - \boldsymbol{W}^i - \boldsymbol{F^i_{ex}}) \tag{4.1}$$

where $\boldsymbol{F}_g$ has been isolated from all other external forces acting on the body $\Sigma_i \boldsymbol{F}^i_{ex}$.

Similarly, we can rewrite Euler's equation describing the moment balance around the lumbar joint center (pelvic frame origin) in terms of the total ground reaction moment:

$$\boldsymbol{M_g} = \sum_{i=1}^{15} (\boldsymbol{M}^{i\star} + \boldsymbol{l}^i \times \boldsymbol{F}^{i\star} + \boldsymbol{l}^i \times \boldsymbol{W}^i - \boldsymbol{M^i_{ex}} - \boldsymbol{l}^i_{ex} \times \boldsymbol{F^i_{ex}}) \tag{4.2}$$

where $\boldsymbol{M}_g$ as been isolated from all other external moments $\Sigma_i \boldsymbol{M}^i_{ex}$. The location of segment i center of mass $\boldsymbol{l}^i$ with respect to the pelvic frame origin is written as:

$$\boldsymbol{l}^i = \boldsymbol{r}^i + \sum_{n \in S} \boldsymbol{\zeta}^n \tag{4.3}$$

where S is the set of all parent segments to segment i. The point of action of the net external forces $\boldsymbol{l}_{ex}^{i}$ on segment i with respect to the pelvic frame origin is:

$$\boldsymbol{l}_{ex}^{i} = \boldsymbol{r}_{ex}^{i} + \sum_{n \in S} \boldsymbol{\zeta}^{n} \tag{4.4}$$

For example, Figure 4.1 shows vector $\boldsymbol{l}^{12}$ describing the right leg center of mass.

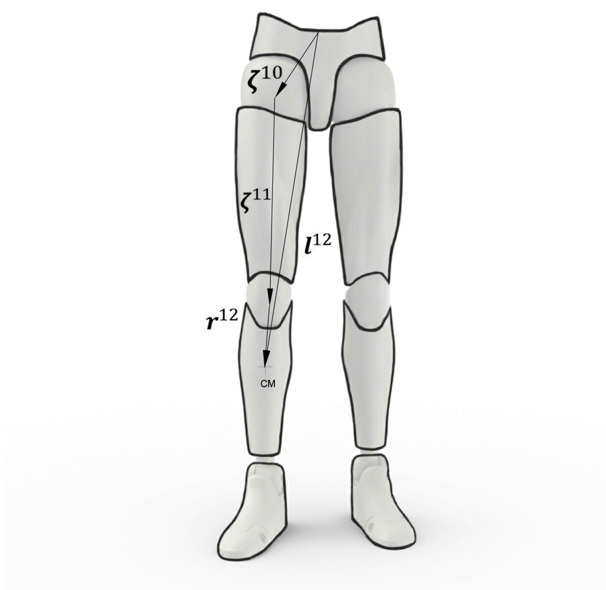


Figure 4.1: Right lower leg center of mass with respect to the pelvic frame origin

The total ground reaction moment represents the location of the center of pressure, where the total ground reaction forces is applied, with respect to the lumbar joint center. The ground reaction moment can also be expressed about other points, such as the projection of the ankle joint center on the ground [5, 11].

4.2 Foot Contact Detection

Following Karatsidis et al.[12], foot velocity is used to detect foot contact with the ground. To establish a foot contact with the ground, the contact condition requires that the toe speed v_{toe} drops below a threshold v_{th}. A foot loses contact with the ground, when the toe speed is larger than that threshold *and* the heel acceleration is negative. The velocity threshold protects against noise in the measured toe speed. The condition on heel acceleration enforces an assumption that the heel reaches maximum velocity at toe-off. The flow-chart in Figure 4.2 summarizes this algorithm.

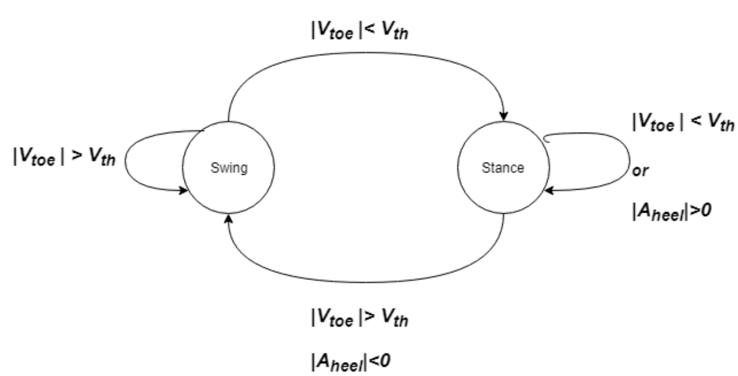


Figure 4.2: Contact detection algorithm

4.3 Breakdown of Ground Reaction Forces

During double-stance stage, an optimization problem is solved to determine the vector:

$$\boldsymbol{x} = \begin{bmatrix} F_{rgx} \\ F_{rgy} \\ F_{rgz} \\ M_{rgx} \\ M_{rgy} \\ M_{rgz} \\ F_{lgx} \\ F_{lgy} \\ F_{lgz} \\ M_{lgx} \\ M_{lgy} \\ M_{lgz} \end{bmatrix} \tag{4.5}$$

where F_{rgx}, F_{rgy}, and F_{rgz} are the components of the right foot reaction force $\boldsymbol{F}_{rg}$, F_{lgx}, F_{lgy}, and F_{lgz} are the components of the left foot reaction force $\boldsymbol{F}_{lg}$, M_{rgx}, M_{rgy}, and M_{rgz} are the components of moment $\boldsymbol{M}_{rg}$ produced by $\boldsymbol{F}_{rg}$ around the pelvic frame origin, and M_{lgx}, M_{lgy}, and M_{lgz} are the components of the moment $\boldsymbol{M}_{lg}$ produced by $\boldsymbol{F}_{lg}$ about the pelvic frame origin.

The objective function $\boldsymbol{f}(\boldsymbol{x})$ minimizes the squares of the net joint moment magnitudes for the joints along the closed loop from the right to the left foot, namely the right ankle $\boldsymbol{M}_{ra}$, right knee $\boldsymbol{M}_{rk}$, right hip $\boldsymbol{M}_{rh}$, left hip $\boldsymbol{M}_{lh}$, left knee $\boldsymbol{M}_{lk}$, and left ankle $\boldsymbol{M}_{la}$:

$$\min\{|\boldsymbol{M}_{rh}|^2 + |\boldsymbol{M}_{lh}|^2 + |\boldsymbol{M}_{rk}|^2 + |\boldsymbol{M}_{lk}|^2 + |\boldsymbol{M}_{ra}|^2 + |\boldsymbol{M}_{la}|^2\} \tag{4.6}$$

The joint moments are obtained from the inverse dynamics model described in Chapter 3.

Equality constraints are enforced to maintain the total ground reaction force $\boldsymbol{F}_g$ and moment $\boldsymbol{M}_g$ found from the equations of motion (4.1) and (4.2)

$$\boldsymbol{F}_{rg} + \boldsymbol{F}_{lg} = \boldsymbol{F}_g \tag{4.7}$$

$$\boldsymbol{M}_{rg} + \boldsymbol{M}_{lg} = \boldsymbol{M}_g \tag{4.8}$$

They form a 12×12 equality constraint matrix $[\boldsymbol{C}_{eq}]$ and the corresponding constraint vector $\boldsymbol{b}_{eq}$.

The vertical component y of the ground reaction forces can take only positive values. Whereas the the horizontal components x and z and all components of the ground reaction moment can act in either direction. Thus, the lower and upper bounds of the solution vector are defined as:

$$\boldsymbol{lb} = \begin{bmatrix} -\infty \\ 0 \\ -\infty \\ -\infty \\ -\infty \\ -\infty \\ -\infty \\ 0 \\ -\infty \\ -\infty \\ -\infty \\ -\infty \end{bmatrix}, \qquad \boldsymbol{ub} = \begin{bmatrix} \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \\ \infty \end{bmatrix} \tag{4.9}$$

Matlab function **fmincon** is employed to solve for $\boldsymbol{x}$ subject to the objective function (4.6), matrix $[\boldsymbol{C}_{eq}]$, vector $\boldsymbol{b}_{eq}$, and the boundaries (4.9).

Chapter 5

Biomechanics-based Ergonomic Assessment

IMC systems allow us to capture human motion on-site. We exploited this feature to develop a complete suite of ergonomic assessment tools based on biomechanical analysis of human motions and loads, thereby enabling on-site task analysis. It solves the inverse dynamics of the full-body model described in Chapter 3 to estimate the net joint forces and moments in major body joints from motions captured by an IMC system. The calculated forces and moments are then used to asses the risk levels experienced by body segments during a given task. The suite is able to use two common IMC systems; MVN [54] and Perception Neuron [55]. It also includes two Graphic User Interfaces (GUIs) for each of those two IMC systems. This chapter describes the structure, functionality, and interface of the ergonomic assessment suite.

5.1 Structure

The flow-chart of the inverse dynamics solver is shown in Figure 5.1. Parallelograms indicate the input to the model. Rectangles represent Matlab code developed to carry out a given process. The input to the model when using Perception Neuron IMC system are:

1) 'BVH' file containing the location of the pelvic joint center in the reference frame $\mathcal{F}^0$ and the rotation matrices between each segment frame $\mathcal{F}^i$ and its parent segment frame $\mathcal{F}^k$ as per Table 3.1,
2) 'CALC' file containing the locations, accelerations, and angular velocities of the IMUs in the reference frame $\mathcal{F}^0$,
3) the subject height and mass,
4) the external forces, and
5) the GRFs.

When using MVN IMC system, the data listed under items (1) and (2) are contained in an

'MVNX' file. The net force and moment in major body joints are evaluated by the Inverse Dynamic Solver.

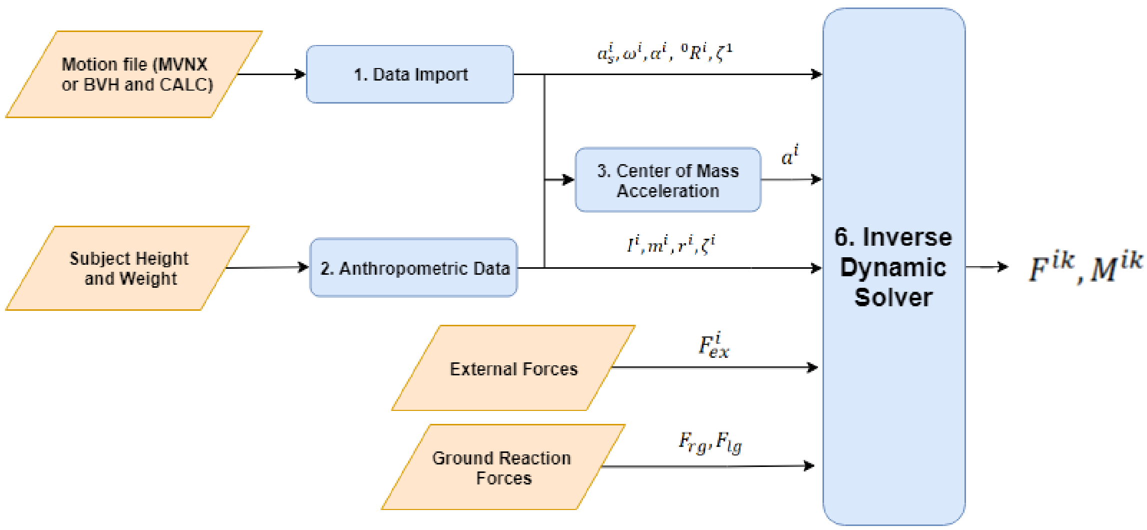


Figure 5.1: Flow chart of the Inverse Dynamics problem

5.1.1 Data Import

First, the motion data captured during the task under investigation is imported in its native format, as recorded by the corresponding software, into Matlab. For the perception Neuron IMC system, the BVH file is imported using an open source code [56] and CALC file is imported using a custom code, Appendix A.1. For the MVN IMC system, the MVNX file is imported using 'load_mvnx.m' code provided by MVN Studio Developer Toolkit [57].

These imported files are then converted by Matlab codes, Appendix A.1, into a matrix form where each captured motion frame is represented by a row. The columns represent the sensors' location $\boldsymbol{r}_s^i$, acceleration $\boldsymbol{a}_s^i$, angular velocity $\boldsymbol{\omega}^i$, and angular acceleration $\boldsymbol{\alpha}^i$, the segments' orientation $[{}^0\boldsymbol{R}^i]$ in the global frame $\boldsymbol{\mathcal{F}}^i$, and the origin of the pelvic frame in the global frame $\boldsymbol{\zeta}^1$.

5.1.2 Data Processing

The regression formulas of Dumas et al. [2] and subject's height and mass are used to calculate each segment mass m^i, its inertia dyadic $[\boldsymbol{J}^i]$, the location of its center of mass $\boldsymbol{r}^i$ in its local frame $\boldsymbol{\mathcal{F}}^i$, and the local frame origin in the the parent frame $\boldsymbol{\zeta}^i$. The code is provided in Appendix A.2.

The center of mass acceleration $\boldsymbol{a}^i$ is calculated as per equation (3.4). Since the MVN system, reports the acceleration of the joint center rather than the IMU, the vector $\boldsymbol{r}_s^i(t)$ in the equation is taken as the location of the joint center relative to the center of mass instead of the location of the IMU with respect to the center of mass. The codes for calculating the center of mass acceleration for Perception Neuron and MVN systems are provided in Appendix A.3.

5.1.3 Ground Reaction Forces

The velocity and acceleration of each foot are calculated and used to determine the toe velocity and heel acceleration. The Foot Contact Detection algorithm, section 4.2, is then used to predict foot contact with the ground. The code implementing the algorithm is provided in Appendix A.4.

The components of the total ground reaction force are calculated from (4.1) and the force location is found from (4.2). The code for GRFs estimation is provided in Appendix A.5. Where the contact detection algorithm has determined that one foot is in contact with the ground, the total GRF is assigned to that foot. Where the algorithm has determined that both feet are in contact with the ground, the optimization problem in section 4.3 is solved to breakdown the total ground reaction force between the right and left feet. The flow chart of the optimization problem is shown in Figure 5.2 and code implementing it is provided in Appendix A.5.

5.1.4 Inverse Dynamics

The inverse dynamic problem, equations (3.6) and (3.8), are solved for the net joint forces $\boldsymbol{F}^{ik}$ and moments $\boldsymbol{M}^{ik}$, respectively. The code is provided in Appendix A.6

Finally, the net force and moment in the L5/S1 joint, the lower back anatomical model,

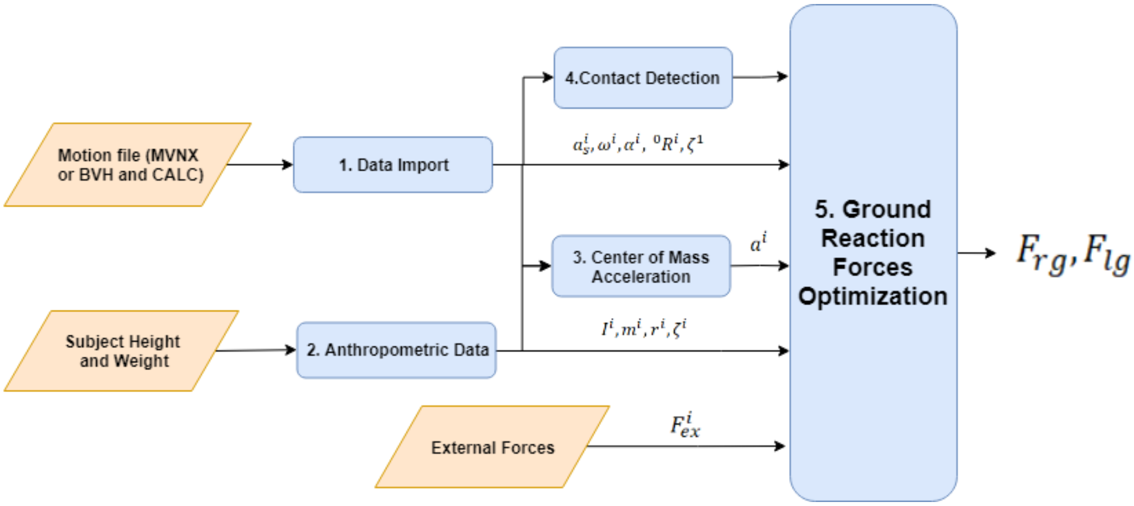


Figure 5.2: Breakdown of GRFs

and the double linear optimization approach described in section 3.4 are used to estimate the components of the disk contact force and the individual lower back muscle forces. The flow chart of this process is shown in figure 5.3 and the code implementing it is provided in Appendix A.7.

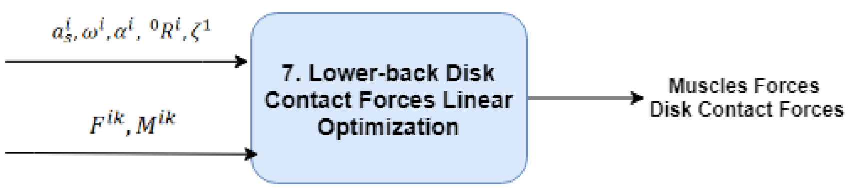


Figure 5.3: Estimation of lower back muscle and disk contact forces

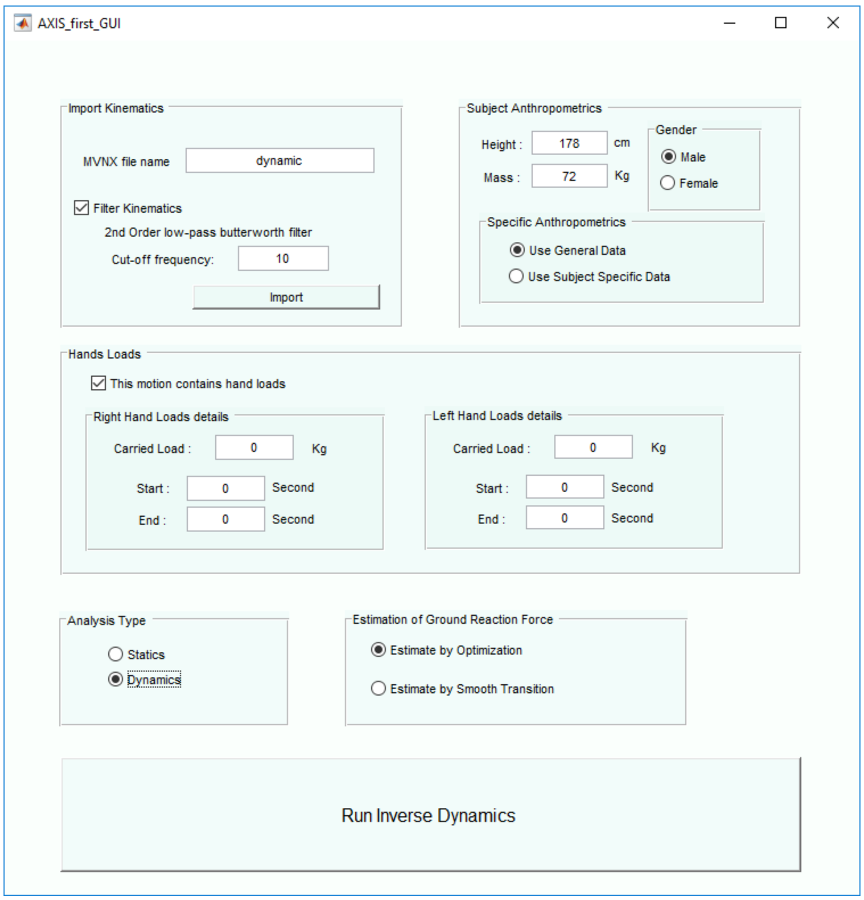


Figure 5.4: Input GUI window

5.2 Tool Functionality and Interface

The input GUI window of the ergonomic assessment suite, for MVN IMC system, is shown in Figure 5.4. The user imports the captured body kinematics by specifying

- the path and name of the MVNX file exported from mvn studio [57] or
- the paths and names of the BVH and CALC files exported by Axis Neuron [58], for Perception Neuron IMC.

A second-order zero-phase low-pass Butterworth filter is applied to the kinematic data with the desired cut-off frequency. The subject's gender, height, and mass are specified. The mass and length of each body segment can be automatically estimated assuming a 50^{th} percentile subject [2] or defined specifically in a segment length and segment mass lists, Figure 5.5, invoked by toggling a box in the GUI window. If the task includes hand load, the user needs to specify the load carried in the right and left hands and the time at which the load was carried and released. The user can also choose the preferred method of GRFs estimation as either the double-linear optimization problem described above or the smooth transition method [12] where the task is strictly composed of gait cycles. Finally, the GUI window allows the user to initiate the solution of the inverse dynamics problem.

The user can specify whether the suite should carry out the static or dynamic analysis of the task. In static analysis, accounts for the segments' weights, external forces, and the body posture, but ignores the segments' inertial forces. This can be done by setting the inertial force $\boldsymbol{F}^{i\star}$ and inertial moment $\boldsymbol{M}^{i\star}$ to zero in the equations of motion (3.6), (3.8), (4.1), and (4.2). The static analysis can be carried to distinguish between the loads due to movement and momentum generated by body segment and loads due to posture, segment weight, and hand loads.

Upon completion of the solution process, a post-processing GUI window, Figure 5.6, is invoked. It allows the user visualize and save the results. Specifically, it shows the motion on a stick figure, where segments are colored corresponding to the load on their proximal joints. The color red is chosen to represent high load, while the color green is chosen to represent a safe load. The 90^{th} percentile load point from the loads on masons was chosen as the reddest, and a zero load is chosen as the most green. The color changes linearly with the increase of load. This motion can be visualized from different orthogonal and isometric view angles. The tool also plots joint loads vs time graph for each of the model body joints. The loads are the norms of the joint net moments, the components of the L5/S1 joint contact forces, and the ground reaction forces and moments.

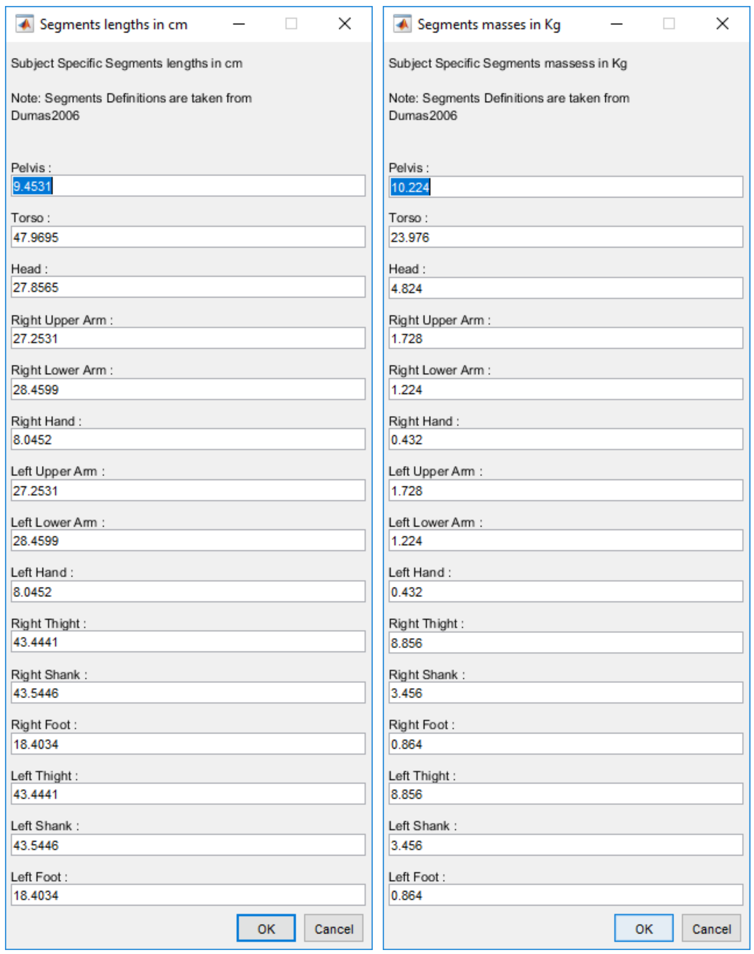


Figure 5.5: Subject-specific anthropometric data entry

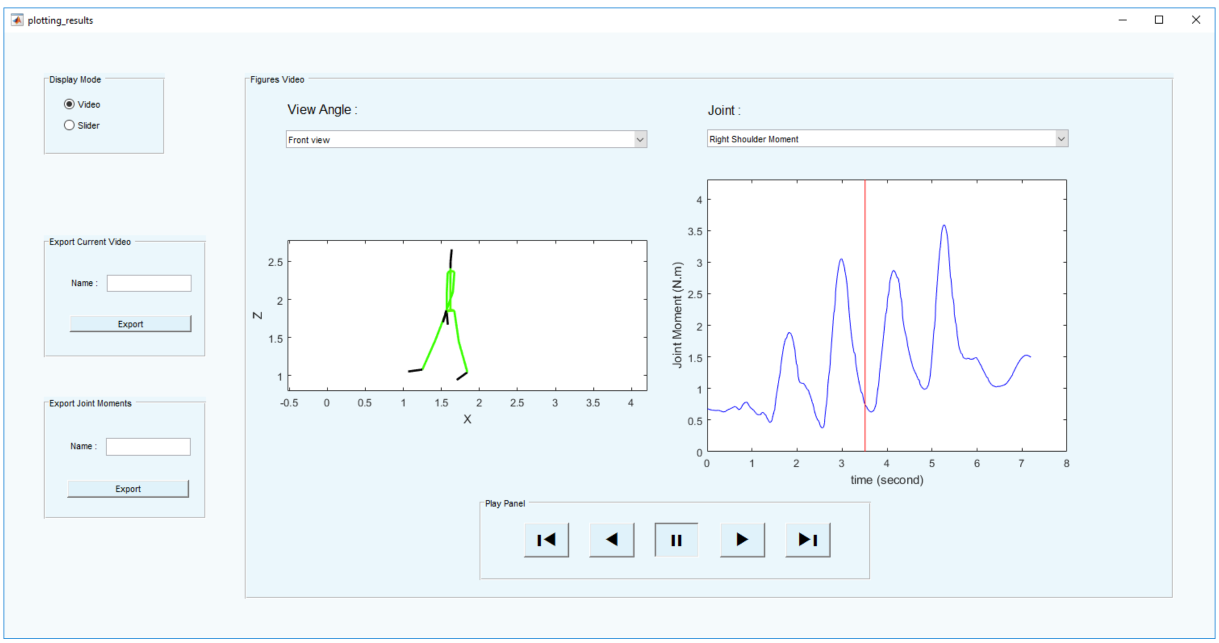


Figure 5.6: Post-processing GUI window

Chapter 6

Model Validation

A pilot study was conducted to validate the full-body model against measured GRFs. The model was also validated by comparing its estimates of the net joint forces and moments as well as L5/S1 contact force, during a standard bricklaying task, to those obtained from an established software for static biomechanical analysis, 3D Static Strength Prediction Program (3DSSPP) [59], developed by the Centre for Ergonomics at the University of Michigan. Unlike the software developed in Chapter 5, 3DSSPP is intended as a simulation tool for workplace movement, not as a motion assessment tool for training. Furthermore, it does not consider the inertial forces caused by accelerations of the body segment. The motion data can be imported to 3DSSPP as ".loc" files, which contain the body joint center at each frame. Therefore, the IMC system kinematics had been used to opting the joint center in the ".loc" format.

6.1 Pilot experiment

6.1.1 Methods

In this experiment, one subject (height: 178 cm, weight: 72 kg) performed three tasks, each with one trial, while the right and left GRFs were being measured using two FPs (FP4060-05-PT-1000, Bertec Corporation [60]) at a sampling rate of 960 Hz and body motions were captured using IMC system (Perception Neuron [55]) at a sampling rate of 120 Hz.

Seventeen IMUs were attached to the body segments as required by the IMC systems; on the hands, lower arms, upper arms, shoulders, head, torso, pelvis, upper legs, lower legs, and feet. After attaching the 17 IMUs constituting the system to the corresponding body segments, a calibration session was carried out where the subject took three postures: A pose, T pose, and S pose, Figure 6.2. The IMC system uses these poses to identify the orientations of the IMUs within its inertial frame $\mathcal{F}^0$.

The FPs measure the GRFs in a laboratory inertial frame $\mathcal{F}^L$ while the IMC system measures the segment accelerations in its own inertial frame $\mathcal{F}^0$. To align the two frames,

Figure 6.1: The lifting task within the pilot study

(a) A pose (b) T pose (c) S pose

Figure 6.2: The calibration postures

an OMC system (Vicon [61]) tracked the position of a pelvic anterior-superior iliac spine marker throughout the experiment. The two frames were then aligned by comparing the position of this bony landmark in both frame.

During the first task, the subject was standing stationary for 3 seconds with both feet in contact with the FPs. In the second task, the subject performed one gait cycle such that each foot landed on the corresponding force plate during the cycle. The third task involved material handling, where the subject moved a 17 kg box from a position immediately to his front right side to his front left side, while squatting, Figure 6.1, by lifting it from the ground to knee height level and depositing it again on the ground. Throughout this task, the subject feet were stationary and in contact with the FPs.

The measured forces (FPs) and motions (IMC system) were filtered using a second-order zero-phase Butterworth low-pass filter with a cut-off frequency of 10 Hz. The GRFs were then sampled down to 120 Hz to match the sampling frequency of the accelerations and angular velocities. The FPs measurements where synchronized with the IMC system kinematics by detecting distinguished features in the motion of the pelvic anterior-superior iliac as recorded by both motion capture systems.

To evaluate the total GRF, the center of mass acceleration $\boldsymbol{a}^i$ of each segment was evaluated as by equation (3.4). Then, the inertial force $\boldsymbol{F}^{i\star}$, inertial moments $\boldsymbol{M}^{i\star}$, and weight $\boldsymbol{W}^i$ or each segment was obtained. These values had been substituted in equations

(4.1) and (4.2) to calculate the total GRF components. If both feet are predicted to have contact with the ground as by the prediction algorithm in Section 4.2, an optimization problem were defined, Section 4.3, to breakdown the total GRF and GRM to their right and left contributions. The total GRF and GRM were used to constrain the right and left GRFs to maintain the equations of motion.

We examined the efficacy of our approach in breaking down the total GRF by comparing the predicted GRFs to those measured by the FPs. The relative Root-Mean-Square Error (rRMSE) was used as a metric to evaluate agreement between the two sets of results over a given task. The Root-Mean-Square Error (RMSE) is defined as:

$$RMSE = \sqrt{\frac{1}{T}\int_0^T \big(x_1(t) - x_2(t)\big)^2 dt} \tag{6.1}$$

where x_1 is a measured GRF or GRM component, x_2 is the corresponding model predicted component, and T is the total task time. The rRMSE is defined as:

$$rRMSE = \frac{RMSE}{max(|x_1(t)|)} \times 100\% \tag{6.2}$$

6.1.2 Results and Discussion

The predicted and measured vertical component of the total GRF during the standing task are shown in Figure 6.3. The estimated and measured vertical components are in excellent agreement with $rRMSE = 2.39\%$.

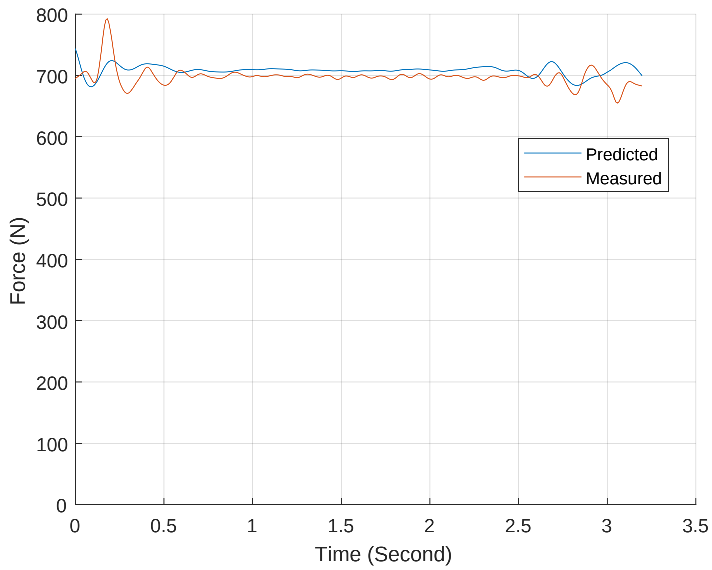
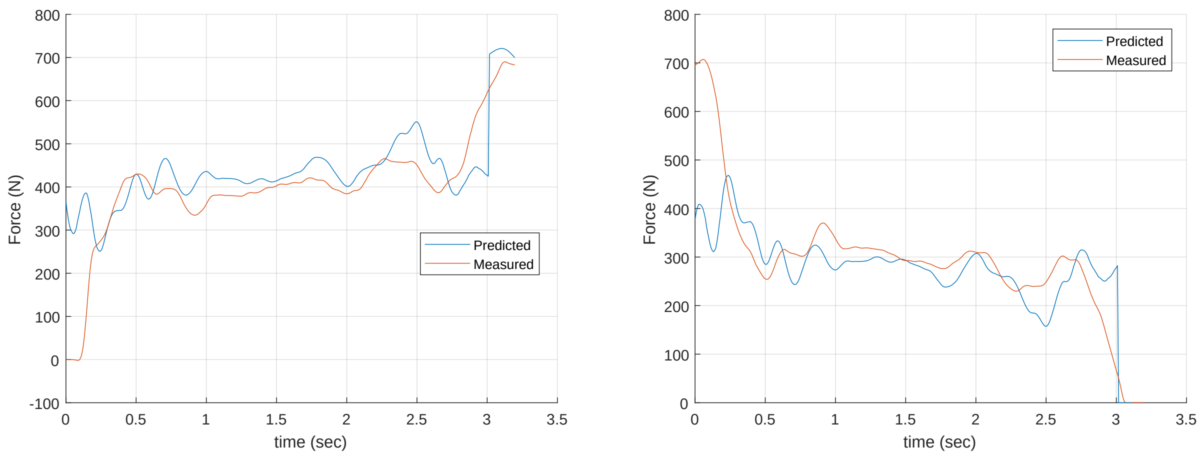


Figure 6.3: The predicted (blue line) and measured (red line) vertical components of the total GRF during standing

The predicted and measured breakdown of the GRF into right and left vertical components for the same task are shown in Figure 6.4. The predicted forces follow the same pattern as the measured forces as the subject shifts his weight from the left to the right foot. The estimated and measured components are in good agreement with $rRMSE = 13.0\%$ for the right foot and $rRMSE = 12.7\%$ for the left foot.



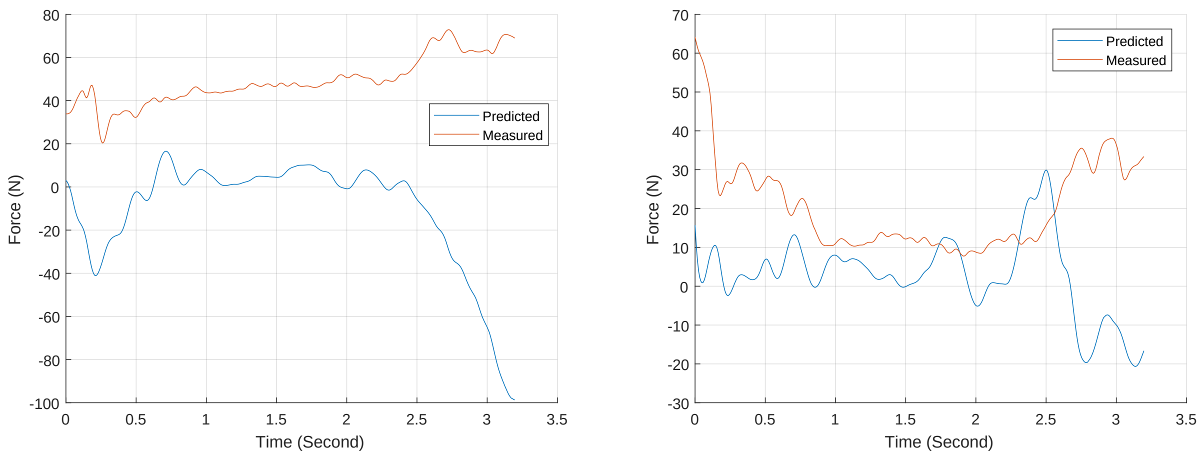
(a) Right foot (b) Left foot

Figure 6.4: The predicted (blue line) and measured (red line) vertical components of the right and left GRFs during standing

Both anterior-posterior and mediolateral components, Figure 6.5, were not predicted as accurately. A very important observation is that the magnitude of these two components is very small compared to vertical components, as the standing tasks is mainly static, and the only force is the body weight acting vertically downward. The $rRMSE$ was 39.4% and 93.7% for the anterior-posterior and mediolateral components respectively.

For one gait cycle during the walking task, predicted and measured vertical component of the total GRF are shown in Figure 6.6. The agreement was excellent with $rRMSE =$ 2.35% . The peak around mid-cycle corresponds to the heal strike motion, as the impact increase the vertical component of the total GRF.

The model predictions of the right and left vertical components of the GRFs, Figure 6.7,



(a) Anterior-posterior component (b) Mediolateral component

Figure 6.5: The predicted (blue line) and measured (red line) horizontal components of the total GRF during standing

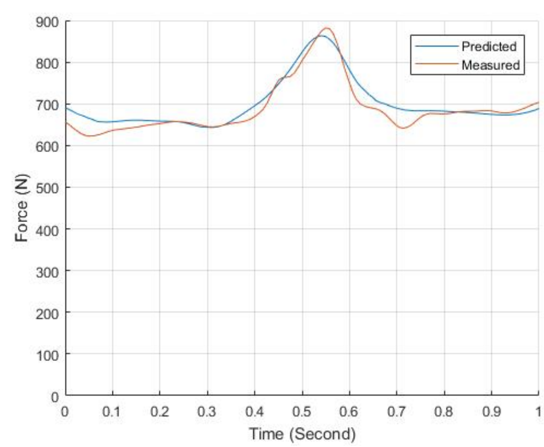
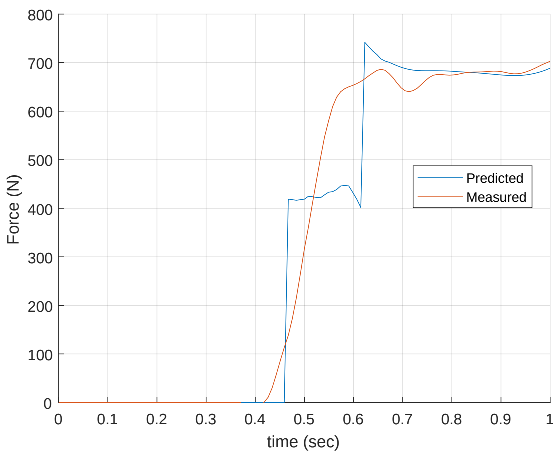
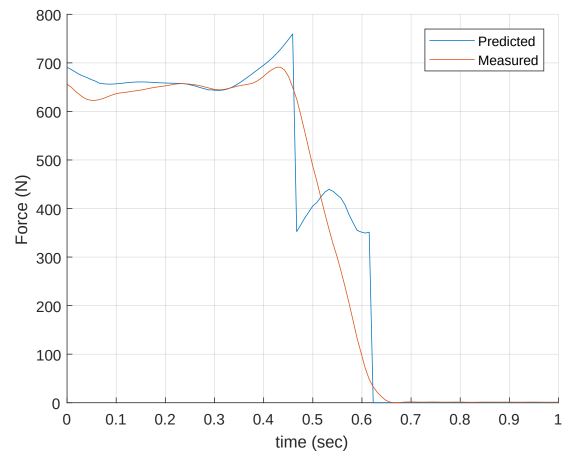


Figure 6.6: The predicted (blue line) and measured (red line) vertical components of the total GRF for one gait cycle

were excellent during single stance, but not as accurate during double stance. Throughout the gait cycle, the $rRMSE = 10.5\%$ was for the right foot and $rRMSE = 10.7\%$ for the left foot.



(a) Right foot

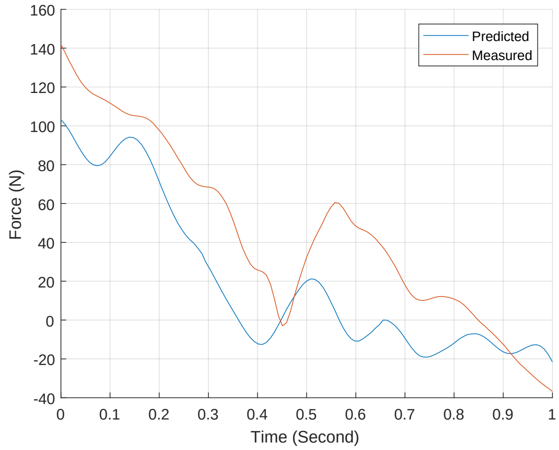


(b) Left foot

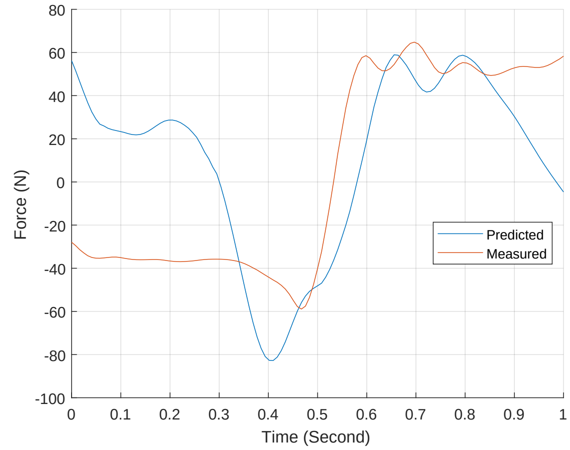
Figure 6.7: The predicted (blue line) and measured (red line) vertical components of the right and left GRFs for one gait cycle

The estimated anterior-posterior component for the gait cycle, Figure 6.8, showed the same pattern as the FPs measurements. However, the loading pattern for the mediolateral component was not predicted correctly for the first swinging phase. The anterior-posterior component was larger in magnitude for the walking task as compared to the other two tasks. This is a result of the body accelerating forward in the direction of motion. The $rRMSE$ was 22.5% and 63.6% for the anterior-posterior and mediolateral components respectively.

For the Lifting motion, the vertical component of the total GRF was more fluctuated as the subject lifts the load, swing it, and dispense it. The predicted and measured vertical component of the total GRF for this task is shown in Figure 6.9. The agreement was



(a) Anterior-posterior component



(b) Mediolateral component

Figure 6.8: The predicted (blue line) and measured (red line) horizontal components of the total GRF for one gait cycle

excellent with $rRMSE = 1.52\%$.

For this task, the vertical components or the right and left GRFs predicted from optimization, was not in good agreement with measured components, Figure 6.10. Although the predictions had the same patterns as the measurements, they were different in values. the $rRMSE = 22.1\%$ was for the right foot and $rRMSE = 46.9\%$ for the left foot.

The predicted and measured anterior-posterior and mediolateral directions of the total GRF for the lifting task are shown in Figure 6.11.The $rRMSE$ was 36.6% and 97.0% for the anterior-posterior and mediolateral components respectively.

The rRMSE was quantified for the total GRF components to validate the full body model and the inverse dynamics. While the rRMSE of the vertical components of right

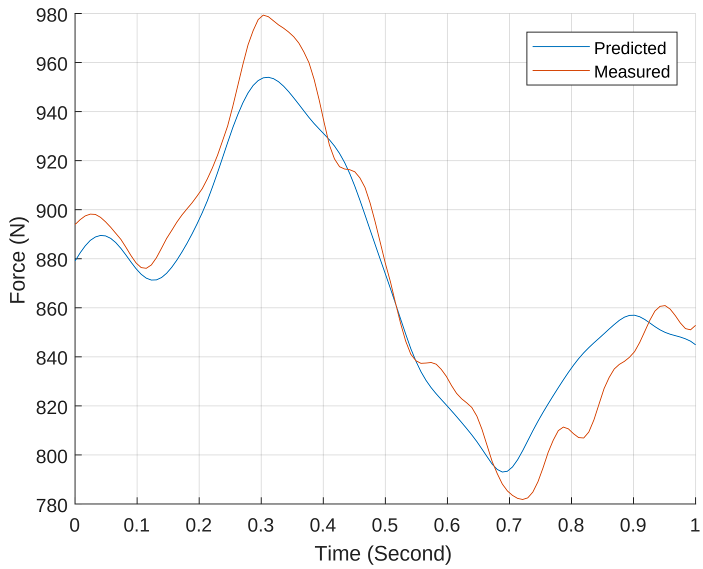
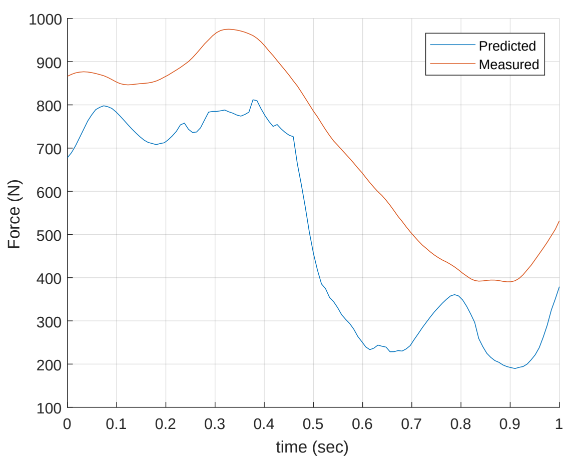
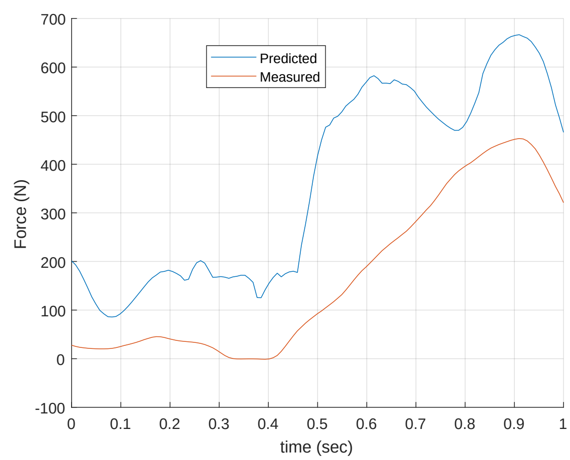


Figure 6.9: The predicted (blue line) and measured (red line) vertical components of the total GRF during lifting

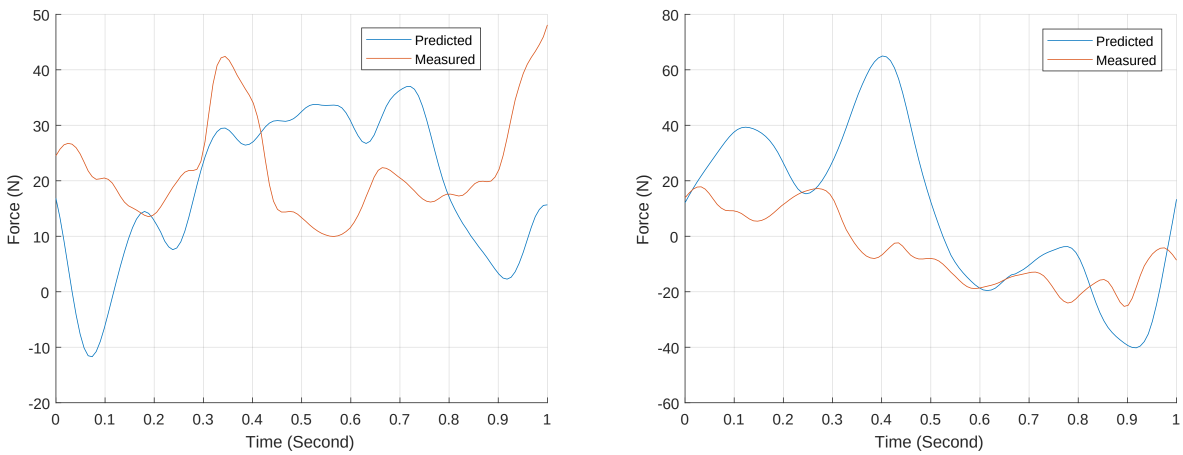


(a) Right foot



(b) Left foot

Figure 6.10: The predicted (blue line) and measured (red line) vertical components of the right and left GRFs during lifting



(a) Anterior-posterior component (b) Mediolateral component

Figure 6.11: The predicted (blue line) and measured (red line) horizontal components of the total GRF during lifting

and left GRFs were used to assess the applicability of the optimization regime for the different tasks.

The rRMSE for the total GRF components of the static standing, walking and lifting trials are listed in Table 6.1. The estimated vertical component of the total GRF showed excellent agreement with measurements, with the rRMSE was less than 2.4 % of the maximum value for all three tasks. However, the evaluated anterior-posterior and the mediolateral components did not agree with measurements, where the rRMSE was (22.5-39.4%) and (63.6-97.0%) respectively. This can be attributed to the lower absolute values of these two components, especially for the mediolateral component. Therefore, the error in the magnitude of the total GRF vector remains acceptable, as it is mainly affected by the vertical component. The error may be a result of the noise in the IMU signal of the acceleration and the orientation or to the difference between the actual and estimated segment inertial properties. The larger relative error in the horizontal components of the GRFs is observed in previous studies [13, 23, 12].

Table 6.1: The relative RMSE (%) for the total GRF components

Task	Vertical	Anterior-Posterior	Mediolateral
Standing	2.39	39.4	93.7
Walking	2.35	22.5	63.6
Lifting	1.52	36.6	97

After estimating the total GRFs, the feet contact with the ground were predicted, and the right and left GRFs had been estimated. The optimization estimation error is quantified in Table 6.2 for all three tasks. The optimization approach for predicting the right and left GRFs performed well for the standing and walking tasks, with rRMSE of (10.5-13.0) percent for the vertical component. However, it was less successful for the lifting task, with the rRMSE as high as 46.9 percent. This error might be because of the large flexion angle in both knees during the lifting, Figure 6.1, which means that the net joint moment for the right and left hips and knees is relatively large. This suggests that the optimization approach might only be applicable for tasks that are performed in standing posture.

Although the estimation error is greater than that reported by Ren et al. using the smooth transition assumption [5], the latter is only applicable for the double stance period of the gait cycle and it depends on the motion pattern as it is based on empirical data of healthy subjects performing normal gait.

Table 6.2: The relative RMSE (%) for the vertical components of the right and left GRFs

Task	Right GRF	Left GRF
Standing	13.0	12.7
Walking	10.5	10.7
Lifting	22.1	46.9

6.2 Bricklaying

6.2.1 Methods

First, a database of bricklaying motion kinematics [62] was expanded by recruiting thirty-two additional healthy male subjects, thereby increasing the number of participants to 53. The subjects were recruited from trainees within the training program of the Canada Masonry Design Centre, where the experiment was conducted. The study was approved by the University of Waterloo Office of Research Ethics and the participants provided informed consent, Appendix B, to all of the experiment procedure. The participants' height, weight, and level of experience are summarized in Table 6.3.

Each subject completed a lead wall, Figure 6.12, by laying 45 block in six courses while

Table 6.3: Participants height, weight and level of experience

#	Experience Level	experience	Participants	Height (cm)	Mass (kg)
1	Novices	< 1 year	17	182 ± 6.9	86.1 ± 13.8
2	Level 1 Apprentices	1-3 years	9	180.6 ± 5.6	89.4 ± 17.2
3	Level 2 Apprentices	3-5 years	13	181.4 ± 4.7	91.7 ± 16.0
4	Journeymen	>5 years	14	178.1 ± 6.1	87.3 ± 10.3

instrumented with the IMC system. A video camera was used to record the motion of the bricklayers, those videos were used to segment the motion of the workers such that lifting of each block is separated as a single file. Concrete Masonry Units (CMUs) blocks were used, the unit measures 390 × 190 × 100 mm in dimensions and it weights 16.6 kg. These loads are modeled as masses on the middle finger metacarpophalangeal joint.

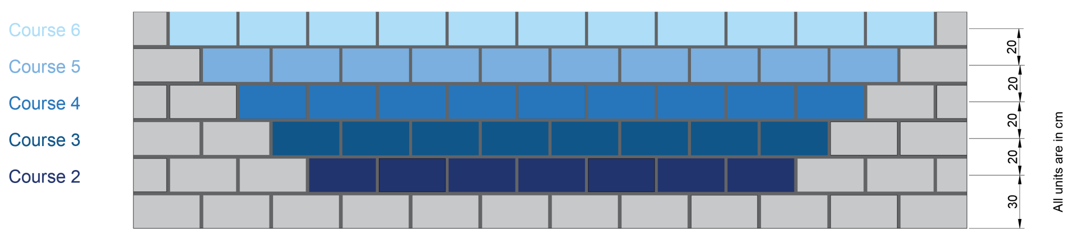


Figure 6.12: Layout of the lead wall completed by the subjects

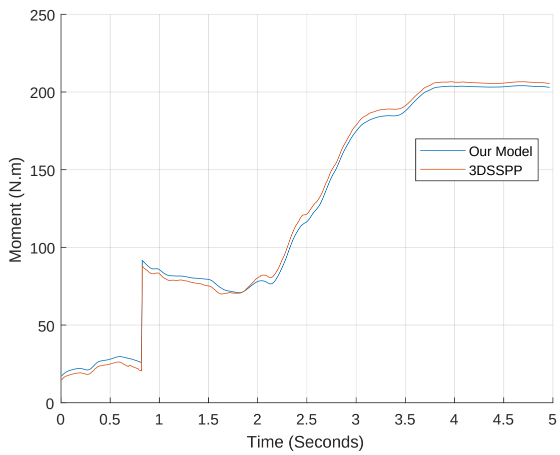
Only kinematic data, sgements' orientation, acceleration, and angular velocity, had been collected during this experiment. The movements of 13 of those subjects were captured using MVN IMC system [54], while the movements of the other 40 subjects were collected using Perception Neuron IMC system [55].

Using these kinematic data, the inverse dynamics problem was solved twice, for static and dynamic loads. The static problem took into account body posture, body segment weight, and static external forces, the CMU weight in our case, but did not account for the inertial forces acting on body segments. The dynamic problem accounted also for the inertial forces. The net joint moments obtained from the static analysis are good risk indicators for postures taken during tasks that do not involve significant motions. But they are not adequate for tasks involving significant motions, such as bricklaying. For those tasks, the net joint moments obtained from the dynamic analysis are better indicators of risk levels.

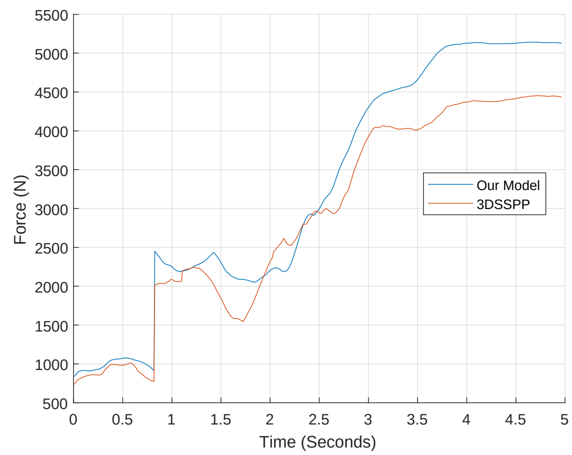
We estimated the static loads to compare our model results directly to those of 3DSSPP, which does not consider inertial forces. In addition, comparing the static and dynamic estimates of joint loads allows us to elicit the additional loads imposed by motion patterns undertaken by a given worker. Estimating those loads, in turn, offers an opportunity to investigate whether workers gain skill at better motion planning and coordination and manage to lower their energy expenditure with experience.

6.2.2 Results and Discussion

The L5/S1 joint net moment and L5/S1 disk compression force are graphed versus time, Figures 6.13, for one of the lifting motions. The jump at time 0.8 seconds corresponds to the block pick-up, as it was assumed to happen instantaneously.



(a) Net joint moment



(b) Disk compression force

Figure 6.13: L5/S1 net moment and compression force as estimated from this model against 3DSSPP

The motion data from 10 randomly selected bricklayers had been used with the 3DSSPP software [59]. The software evaluates the static biomechanical loads resulting from the body weight and external forces, without considering the inertial forces. Using the inverse dynamics model described in Chapter 3, the static loads had been estimated by setting the accelerations to zero. The net joint moments of the right shoulder, left shoulder, and the L5/S1 joint and the components of the L5/S1 disk contact force were estimated. Each trial contains one lifting motion, that is, the motion during lifting one block from the stack to the wall. There were 45 trails for each participant and a total of 450 trials. The average of the RMSE and rRMSE for all 450 lifting motions are listed in Table 6.4.

The rRMSE was less than 10 percent for the right and left shoulders net moment and

Table 6.4: Relative RMSE between the developed model and 3DSSPP software

Joint	rRMSE
Right Shoulder Net Moment	8.95
Left Shoulder Net Moment	10.2
L5/S1 Net Moment	3.97
L4/L5 Compression Force	13.6

less than 4 percent for the L5/S1 joint net moment. This error can be attributed to the difference in the model segment inertial properties, as the error was larger for the shoulder; where the moment is more dependent on the segment proportional masses, than the lower back, where the moment is more dependent on the whole body height and weight. The rRMSE of the L5/S1 disk compression as predicted by this model as compared to 3DSSPP was 13.6 percent. Although this shows less agreement than that of the joint moment, the solution for the lower back disk compression is not an analytical solution, but an optimization problem as described in section 3.4. As a result, different models may lead to different estimations.

Chapter 7

Demonstration of the Ergonomic Assessment Suite

7.1 The Ergonomic Assessment Suite

The biomechanics-based ergonomic assessment suite described in Chapter 5 is demonstrated in this section. The task under analysis is a lift in which a journeyman lays a CMU at the second course. After collection the motion kinematics, the motion files were imported via the input GUI, Figure 5.4, where the subject anthropometric data, block weight, and lift time were also specified. The execution button at the bottom of the GUI is then used to invoke inverse dynamic analysis.

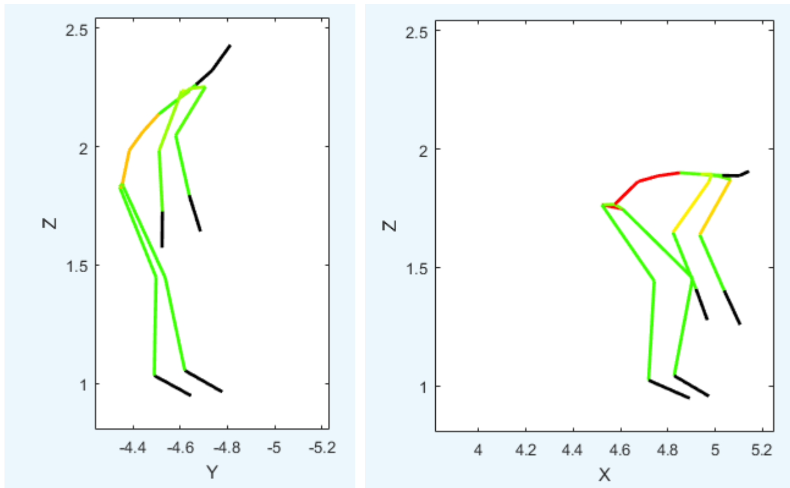
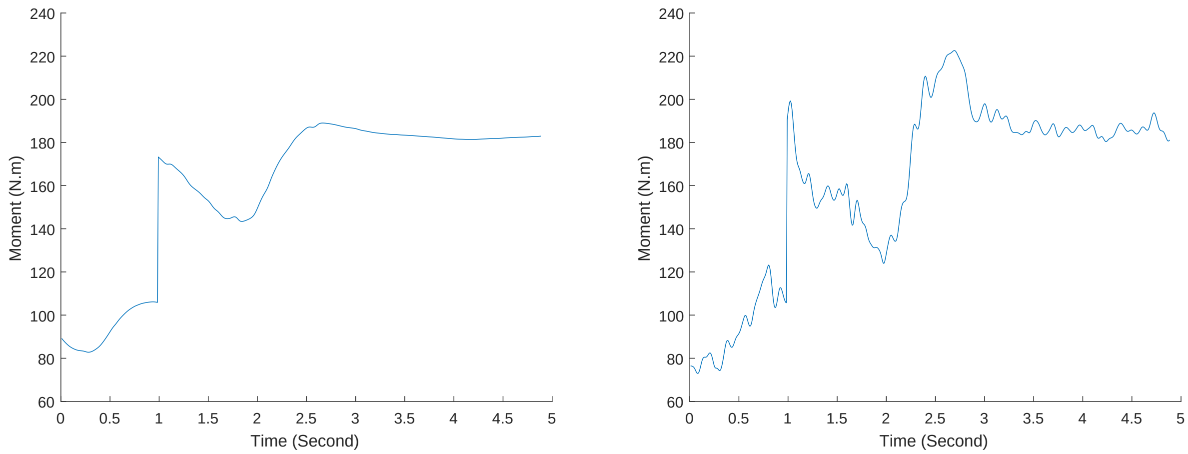


Figure 7.1: Stick figure of the subject at the beginning (left panel) and end (right panel) of the lift

The post-processing GUI, Figure 5.6, displays a video stick figure to demonstrate the load (net moment) of different joints during the lift. Each segment is colored according to the load level in its proximal joint. The rid color indicates a larger moment, while the green color indicates a smaller moment. Figure 7.1 shows the stick figure at the beginning (left panel) and end (right panel) of the lift. While the load on the lumbar joint, indicated by the color of the torso, is moderate during CMU pickup, it is higher during CMU lay down. The posture of the two stick figures suggests that the reason for that difference is a higher flexion angle during laying than pick up. Different view angles can be selected from a drop-down menu to visualize this video, including the 6 orthogonal views and 4 isometric views.

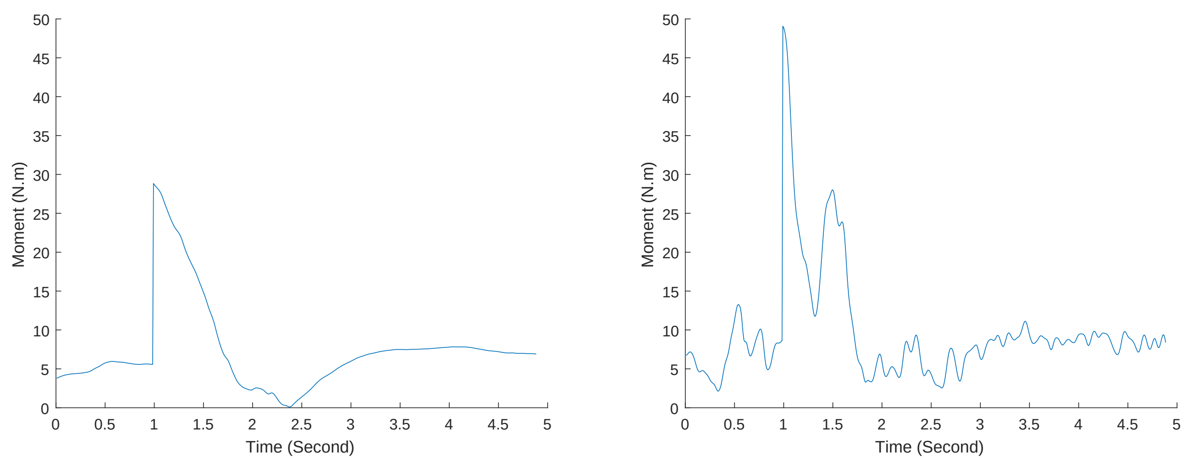
The net moment in major body joints and the contact force components in the L5/S1 disk are evaluated as functions of time and can be visualized and saved. A drop-down menu is used to select the joint load to graph. Figure 7.2 compares the net moment in the lumbar (L5/S1) joint evaluated using static and dynamic analysis for the lift under study. The jump in the net moment in both figures at t = 1 second corresponds to the time at which the external load (CMU weight) is applied to the hands. The participant started the motion in a standing posture, where the net lumbar joint moment was minimal. As he bent his trunk to pick up the CMU the lumber joint net moment increased. It then decreased as the subject straightened his back to walk towards the wall and increased significantly as he laid down the CMU. The 'dynamic' net joint moment followed the same pattern as its 'static' counterpart, although it had more fluctuations, due to body and blocks acceleration, and larger peaks.

In contrast, Figure 7.3, compares the net moment in the dominant shoulder joint obtained from static and dynamic analysis. The moment peaked at pickup and decreased gradually as the subject lowered his hands to lay-down the CMU. This pattern is observable in lower courses lifts. On the other hand, the net moment increases in higher courses as the worker reaches higher while lifting the CMU. The dynamic to static moment ratio is much larger for the shoulder joint compared to the lumber joint. This suggests that inertial forces are more significant to the shoulder joint than they are to the lumbar joint.



(a) Static (b) Dynamic

Figure 7.2: Net moment in the lumbar (L5/S1) joint during a lift



(a) Static (b) Dynamic

Figure 7.3: Net moment in the dominant shoulder joint during a lift

7.2 Loads Experienced by Bricklayers

The ergonomic suite was used to evaluate the static and dynamic biomechanical loads experienced by bricklayers during the CMU lifts required to complete the lead wall shown in Figure 6.12. The maximum load in each joint was found for each lift. These maxima were averaged for all lifts and participants based on

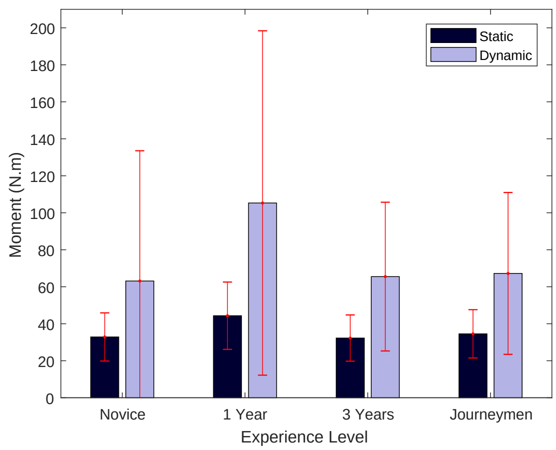
- the experience level
- course height

These averages were compared to investigate the relationships among experience level, course heights, and biomechanical loads.

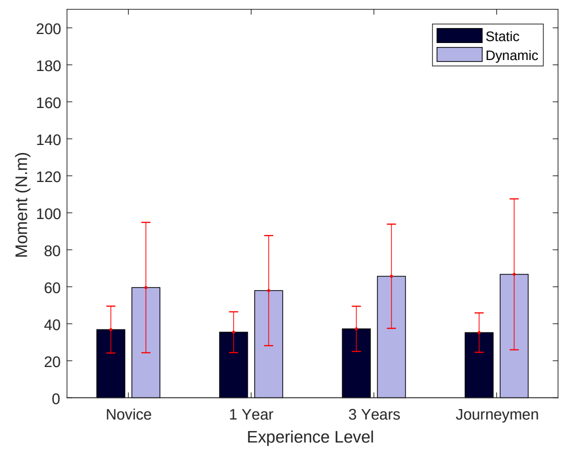
7.2.1 Experience Level

The results for the dominant and non-dominant shoulder joints, dominant and non-dominant knee joints and L5/S1 disk-averaged by experience level are presented and discussed below in more details.

The static net moment in the dominant and non-dominant shoulder joints, Figure 7.4, were similar and that was true across all four experience levels. However, the increase in the net joint moment was significantly higher for the dominant should joint than it was



(a) Dominant shoulder

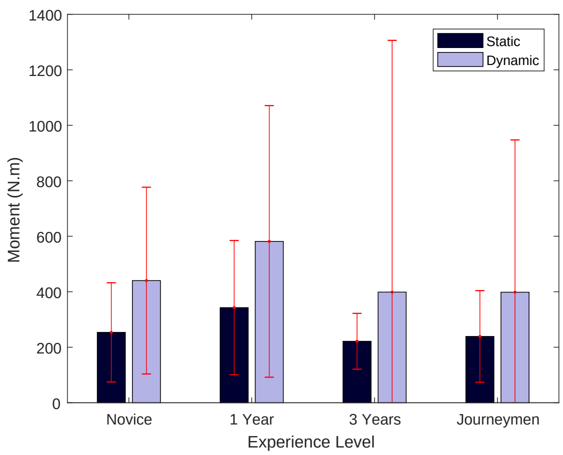


(b) Non-dominant shoulder

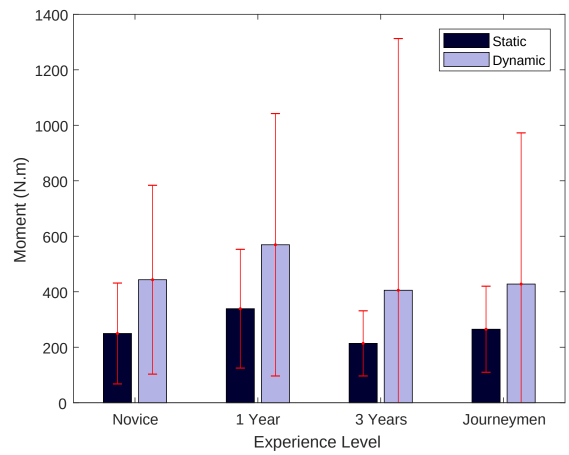
Figure 7.4: Net moment in the shoulder joints averaged by experience level

for the non-dominant joint. This indicates that masons tended to pivot around the non-dominant side resulting in the dominant arm experiencing higher inertial loads than the non-dominant arm, even though the posture and static loads on the contralateral sides were similar. This shows a deficiency in static analysis of tasks involving significant rotational motions. This deficiency is even more serious for tasks involving uncoordinated motions of the hands, such as one-handed CMU lifts and tool manipulation. In this case, the differences between arm loads will also arise due to differences between their transnational accelerations.

The knee net moments, Figure 7.5, show identical results in terms of differentiating the experience levels. There were no significant differences between the dominant and non-dominant sides, which is expected as the lower body is less susceptible to dominant hand



(a) Dominant knee



(b) Non-dominant knee

Figure 7.5: Net moment in the knee joints averaged by experience level

differences.

Figure 7.6 shows the lumber joint disk compression force for different experience levels. Both the novices and the journeymen had safer working postures (lower static disk compression force) as compared to the apprentices. This result had been observed on previous study [62] and had been attributed to the novices working more cautiously, which results in smaller loads on their bodies but a slower working pace and less productivity. On the other hand, the journeymen had gained the experience to work more productively while keeping a safer posture. The dynamic to static load ratio was almost identical across all experience groups, with an increase of about 25% in the dynamic loads compared to the static loads. This suggests that there is no significant difference in inertial forces experienced by different experience groups.

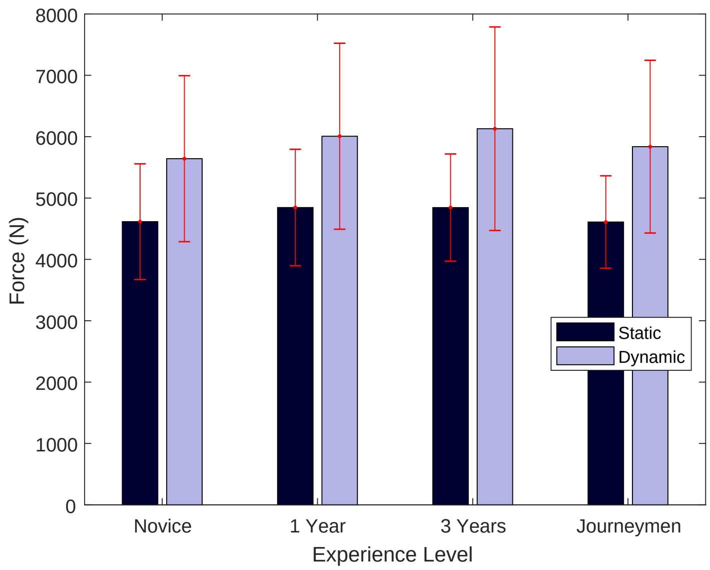


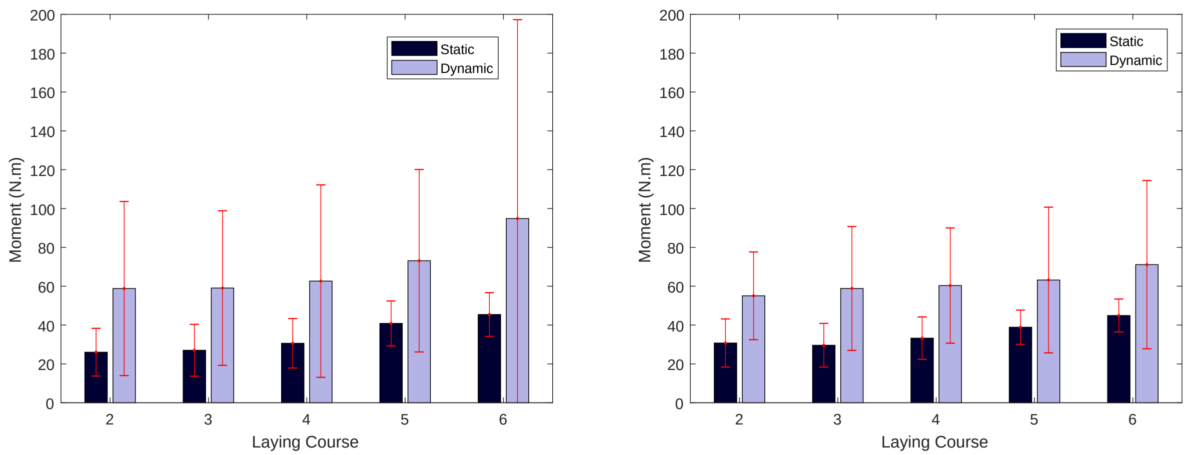
Figure 7.6: L5/S1 disk compression force by experience level

7.2.2 Course Height

Similarly, The results for the dominant and non-dominant shoulder joints, dominant and non-dominant knee joints, and L5/S1 disk were averaged by course are presented and discussed below in more details.

The shoulder net moments, Figure 7.7, were significantly larger for the higher courses. This is caused by the larger flexion angles of the shoulders as the hands have to reach higher, which impose a larger moment on the shoulders.

Figure 7.8 shows the static and dynamic knee net moments. Although the static moments were identical across all courses, the dynamic moments were larger for the higher



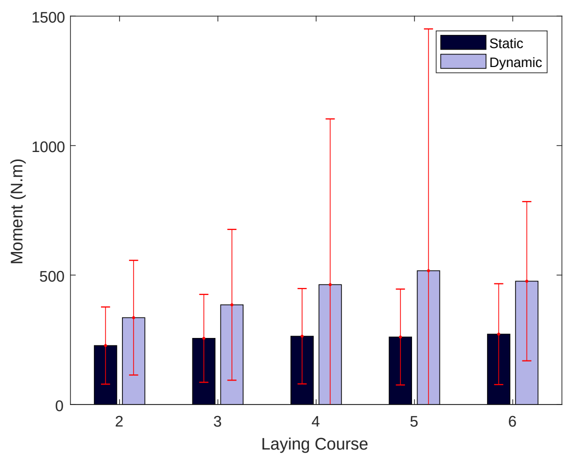
(a) Dominant shoulder (b) Non-dominant shoulder

Figure 7.7: Net moment in the shoulder joints averaged by course

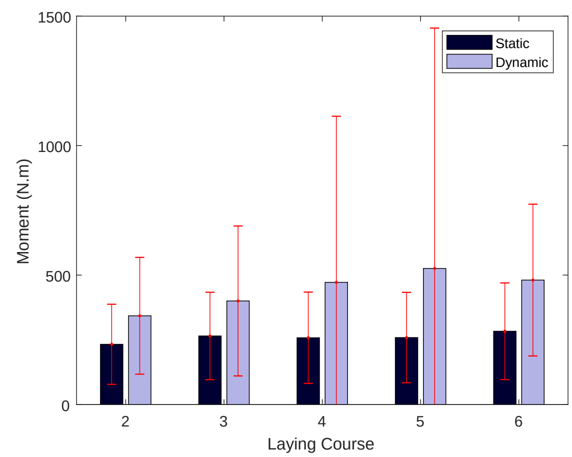
courses. This means that the inertial forces increase when laying-down the CMU on a higher course, possibly to accelerate the CMU and generate momentum so it can reaches higher.

The lumber disk compression forces, Figure 7.9, were larger for the lower courses, as the workers had to bend to lay-down on these courses. However, the dynamic to static load ratio increased with the higher course, where it was 1.16 for the lowest course and 1.35 for the highest courses. This again may be explained by the participant accelerating the CMU when he needs to lay it down higher, resulting in larger inertial forces throughout the body.

Moreover, the dynamic to static load ratio was much larger for the shoulder than it was for the lower back. This is a result of faster arm movement compared to the torso,



(a) Dominant knee



(b) Non-dominant knee

Figure 7.8: Net moment in the knee joints averaged by course

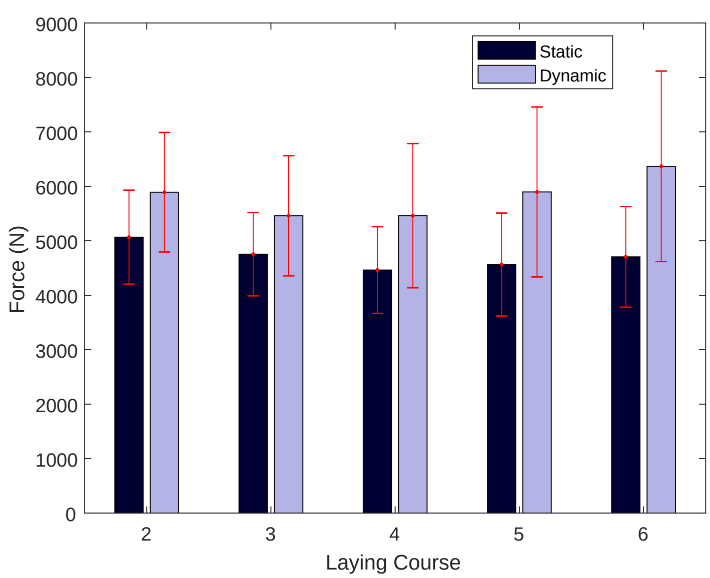


Figure 7.9: L5/S1 disk compression force by course

but it may also be a result of the segments dynamic balance; i.e. the dynamic loads from the right arm are countered by the dynamic loads from the left arm.

Finally, the lower back disk compression did not exceed the maximum permissible limit set by National Institute for Occupational Safety and Health (NIOSH) (6376 N). However, in certain cases, the dynamic disk compression did exceed this limit. This represents the importance of re-evaluating these limits based on dynamic loads, as it may be crucial to determine if a certain task is biomechanically safe or not.

Chapter 8

Conclusions and Future Work

8.1 Full Body Model for IMC systems

Traditionally, estimating the biomechanical loads on the human body was a long process that requires lab equipment and a dedicated space to capture the body motion and the GRFs. However, the use of IMC systems allowed for biomechanical task evaluation out of the lab, in open spaces, or on-site.

This thesis developed a human body model to apply inverse dynamics from motion kinematics captured using the IMC system, which allows the estimation of the net moments and forces on all major body joints. Although this problem is a determinate problem that can be solved analytically for the upper-body segments, or for the whole body when only one foot is in contact with the ground, it becomes indeterminate when during double stance. To overcome this, Ground Reaction Forces (GRFs) had been estimated using an optimization approach during the indeterminate phases.

A pilot experiment had been conducted to validate the predicted GRFs against measured values. The total GRFs predictions were excellent for the vertical component, but they were less accurate for the anterior-posterior and lateral components. This comparison validates the used inertial properties, the captured motion kinematics, and the developed inverse dynamics for the full body model. The estimated right and left GRFs showed an acceptable agreement with the measured values for the standing and lifting tasks, which suggest that the optimization approach developed and used in this study can be used to GRFs estimation in a range of tasks.

For the joints loads, and the lower back disk compression forces, the model had been validated against an existing model (3DSSPP). Predictions of the net joints' forces and moments and the lower back disk contact forces obtained from this new model, were in close agreement with those obtained from 3DSSPP.

8.2 Biomechanical-based Ergonomics Assessment Tool

The model had been used to build an On-site Biomechanics-based Ergonomics Assessment Software. This software is meant to be used on-site to quickly and easily assess the worker's motion by estimating the loads exerted by the major body joints. The model had then been used to estimate the biomechanical loads on bricklayers during their lifting tasks. The was achieved by extending and using an existing database of motion kinematics of bricklayers during building a lead-wall. The loads had been compared for participants with different experience level, for courses with different laying height, and when including the static loads only against dynamic loads.

The model and biomechanics-based ergonomics assessment tool developed in this study made applying inverse dynamic analysis on human motion more reachable. Using an IMC system with this tool can make the biomechanical assessment faster, cheaper and applicable to a larger range of tasks (e.g. on-site tasks or tasks that requires large open spaces). Unlike previous models, this model is not designed for a specific task; usually, gait; but it is suitable for any task that is dominated by the inertial and weight forces. The GRFs predicted on this model are used on the bottom-up approach in estimating the joint net moments in the lower limbs. Therefore, this prediction did not affect upper-body kinetics.

Furthermore, as the total GRF was estimated from the whole body inverse dynamics, it can be used to estimate the lumber joint net moment using the bottom-up approach, as it will yield the same results as the top-down approach. The total GRF estimation showed excellent agreement with the measurements, which is used to validate the full body model. On the other hand, the right and left GRFs estimations were not successful for the lifting task, suggesting that the optimization approach is not applicable to all tasks.

8.3 Future Work

The model prediction of the total, right, and left GRFs were validated using the motion of one subject only, performing three tasks. This is a major limitation of this work. Thus, more validation is needed to assess the full body model and the optimization approach used.

All the loads estimated using this model, except for the lumber joint, are joints' net forces and moments. Nonetheless, estimating the tissues internal forces such as muscles contraction forces, ligaments stress, and strain, and joints contact forces are also important for many tasks. Achieving this requires developing an anatomically detailed internal model for each joint. Some tools (such as OpenSim [63]) had been developed to such internal models with the physiological and physical properties of each element with the full body model. This software usually applies the inverse dynamics analysis on the kinematics obtained from OMC systems. The tool proposed in this research may be developed further to be integrated with such software for more detailed studies. This can be achieved by calculating joints angles, net forces, and net moments as defined in the model used on the targeted software.

Furthermore, other risk factor metric may be implemented to the ergonomics assessment tools. This may include cumulative loading, such as the integrated joint moment over the whole task time. It may also include kinematic variables, such as joint flexion speed. Using these risk factors, along with the average and maximum joint net moments, can help to predict MSDs [46].

Future work could also integrate the tool with the native IMC system software. This can allow for real-time data transfer from the IMUs to the Matlab tool. This data can then be processed and used for inverse dynamics analysis. Such a real-time assessment tool may have the potential to be used for safety training programs or as an overexertion warning system.

Another possible improvement on the current work is to use a Force-Sensing Glove System for automatic detection of hand loads during material handling tasks. This will make executing the assessment easier, as the user does not have to specify the hand load details, and more accurate. Force shoes may also be used to overcome the inaccuracy in predicting the right and left GRFs. They may also help to predict the hand loads without the need for Force-Sensing Glove [64].

References

[1] A. Schultz, G. Andersson, K. Haderspeck, R. Örtengren, M. Nordin, and R. Björk, "Analysis and measurement of lumbar trunk loads in tasks involving bends and twists," *Journal of biomechanics*, vol. 15, no. 9, pp. 669–675, 1982.

[2] R. Dumas, L. Cheze, and J.-P. Verriest, "Adjustments to mcconville et al. and young et al. body segment inertial parameters," *Journal of biomechanics*, vol. 40, no. 3, pp. 543–553, 2007.

[3] J. W. Young, R. F. Chandler, C. C. Snow, K. M. Robinette, G. F. Zehner, M. S. Loftberg *et al.*, "Anthropometric and mass distribution characteristics of the adult female." Civil Aerospace Medical Institute, Tech. Rep., 1983.

[4] D. Roetenberg, H. Luinge, and P. Slycke, "Xsens mvn: full 6dof human motion tracking using miniature inertial sensors," *Xsens Motion Technologies BV, Tech. Rep*, vol. 1, 2009.

[5] L. Ren, R. K. Jones, and D. Howard, "Whole body inverse dynamics over a complete gait cycle based only on measured kinematics," *Journal of biomechanics*, vol. 41, no. 12, pp. 2750–2759, 2008.

[6] S. E. Oh, A. Choi, and J. H. Mun, "Prediction of ground reaction forces during gait based on kinematics and a neural network model," *Journal of biomechanics*, vol. 46, no. 14, pp. 2372–2380, 2013.

[7] P. Entzel, J. Albers, and L. Welch, "Best practices for preventing musculoskeletal disorders in masonry: Stakeholder perspectives," *Applied Ergonomics*, vol. 38, no. 5, pp. 557–566, 2007.

[8] S. P. Schneider, "Musculoskeletal injuries in construction: a review of the literature," *Applied occupational and environmental hygiene*, vol. 16, no. 11, pp. 1056–1064, 2001.

[9] X. Wang, X. S. Dong, S. D. Choi, and J. Dement, "Work-related musculoskeletal disorders among construction workers in the united states from 1992 to 2014," *Occup Environ Med*, vol. 74, no. 5, pp. 374–380, 2017.

[10] H. Brenner and W. Ahern, "Sickness absence and early retirement on health grounds in the construction industry in ireland," *Occupational and environmental medicine*, vol. 57, no. 9, pp. 615–620, 2000.

[11] G. Faber, C. Chang, I. Kingma, J. Dennerlein, and J. van Dieën, "Estimating 3d l5/s1 moments and ground reaction forces during trunk bending using a full-body ambulatory inertial motion capture system," *Journal of biomechanics*, vol. 49, no. 6, pp. 904–912, 2016.

[12] A. Karatsidis, G. Bellusci, H. M. Schepers, M. de Zee, M. S. Andersen, and P. H. Veltink, "Estimation of ground reaction forces and moments during gait using only inertial motion capture," *Sensors*, vol. 17, no. 1, p. 75, 2016.

[13] S. Kim and M. A. Nussbaum, "Performance evaluation of a wearable inertial motion capture system for capturing physical exposures during manual material handling tasks," *Ergonomics*, vol. 56, no. 2, pp. 314–326, 2013.

[14] E. Poirson and M. Delangle, "Comparative analysis of human modeling tools," in *International Digital Human Modeling Symposium*, 2013.

[15] Siemens PLM Software, "Jack human simulation software," https://www.dex.siemens.com/plm/jack, accessed: 2019-04-16.

[16] C. L. Vaughan, J. G. Hay, and J. G. Andrews, "Closed loop problems in biomechanics. part iian optimization approach," *Journal of Biomechanics*, vol. 15, no. 3, pp. 201–210, 1982.

[17] T. R. Waters, R. B. Dick, J. Davis-Barkley, and E. F. Krieg, “A cross-sectional study of risk factors for musculoskeletal symptoms in the workplace using data from the general social survey (gss),” *Journal of Occupational and Environmental Medicine*, vol. 49, no. 2, pp. 172–184, 2007.

[18] A. Garg and J. M. Kapellusch, “Applications of biomechanics for prevention of work-related musculoskeletal disorders,” *Ergonomics*, vol. 52, no. 1, pp. 36–59, 2009.

[19] H. J. Luinge and P. H. Veltink, “Measuring orientation of human body segments using miniature gyroscopes and accelerometers,” *Medical and Biological Engineering and computing*, vol. 43, no. 2, pp. 273–282, 2005.

[20] A. G. Cutti, A. Giovanardi, L. Rocchi, A. Davalli, and R. Sacchetti, “Ambulatory measurement of shoulder and elbow kinematics through inertial and magnetic sensors,” *Medical & biological engineering & computing*, vol. 46, no. 2, pp. 169–178, 2008.

[21] T. Cloete and C. Scheffer, “Benchmarking of a full-body inertial motion capture system for clinical gait analysis,” in *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE.* IEEE, 2008, pp. 4579–4582.

[22] A. Godwin, M. Agnew, and J. Stevenson, "Accuracy of inertial motion sensors in static, quasistatic, and complex dynamic motion," *Journal of biomechanical engineering*, vol. 131, no. 11, p. 114501, 2009.

[23] G. S. Faber, C.-C. Chang, P. Rizun, and J. T. Dennerlein, "A novel method for assessing the 3-d orientation accuracy of inertial/magnetic sensors," *Journal of biomechanics*, vol. 46, no. 15, pp. 2745–2751, 2013.

[24] J.-T. Zhang, A. C. Novak, B. Brouwer, and Q. Li, "Concurrent validation of xsens mvn measurement of lower limb joint angular kinematics," *Physiological measurement*, vol. 34, no. 8, p. N63, 2013.

[25] D. W. Wundersitz, K. J. Netto, B. Aisbett, and P. B. Gastin, "Validity of an upper-body-mounted accelerometer to measure peak vertical and resultant force during running and change-of-direction tasks," *Sports Biomechanics*, vol. 12, no. 4, pp. 403–412, 2013.

[26] Y. Ohtaki, K. Sagawa, and H. Inooka, "A method for gait analysis in a daily living environment by body-mounted instruments," *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, vol. 44, no. 4, pp. 1125–1132, 2001.

[27] E. Charry, W. Hu, M. Umer, A. Ronchi, and S. Taylor, "Study on estimation of peak ground reaction forces using tibial accelerations in running," in *Intelligent Sensors, Sensor Networks and Information Processing, 2013 IEEE Eighth International Conference on*. IEEE, 2013, pp. 288–293.

[28] A. Ancillao, S. Tedesco, J. Barton, and B. OFlynn, "Indirect measurement of ground reaction forces and moments by means of wearable inertial sensors: a systematic review," *Sensors*, vol. 18, no. 8, p. 2564, 2018.

[29] Y. Jung, M. Jung, K. Lee, and S. Koo, "Ground reaction force estimation using an insole-type pressure mat and joint kinematics during walking," *Journal of biomechanics*, vol. 47, no. 11, pp. 2693–2699, 2014.

[30] A. Nachemson and G. Elfstrom, "Intravital dynamic pressure measurements in lumbar discs," *Scand J Rehabil Med*, vol. 2, no. suppl 1, pp. 1–40, 1970.

[31] M. C. Battié, T. Videman, L. E. Gibbons, L. D. Fisher, H. Manninen, and K. Gill, "Determinants of lumbar disc degeneration: a study relating lifetime exposures and magnetic resonance imaging findings in identical twins," *Spine*, vol. 20, no. 24, pp. 2601–2612, 1995.

[32] K. Luoma, H. Riihimäki, R. Raininko, R. Luukkonen, A. Lamminen, and E. Viikari-Juntura, "Lumbar disc degeneration in relation to occupation," *Scandinavian journal of work, environment & health*, pp. 358–366, 1998.

[33] K. Luoma, H. Riihimäki, R. Luukkonen, R. Raininko, E. Viikari-Juntura, and A. Lamminen, "Low back pain in relation to lumbar disc degeneration," *Spine*, vol. 25, no. 4, pp. 487–492, 2000.

[34] J. M. Morris, D. B. Lucas, and B. Bresler, "Role of the trunk in stability of the spine," *JBJS*, vol. 43, no. 3, pp. 327–351, 1961.

[35] D. B. Chaffin, G. Andersson, B. J. Martin *et al.*, *Occupational biomechanics*. Wiley New York, 1999.

[36] A. B. Schultz and G. B. Andersson, "Analysis of loads on the lumbar spine," *Spine*, vol. 6, no. 1, pp. 76–82, 1981.

[37] A. Schultz, K. Haderspeck, D. Warwick, and D. Portillo, "Use of lumbar trunk muscles in isometric performance of mechanically complex standing tasks," *Journal of Orthopaedic Research*, vol. 1, no. 1, pp. 77–91, 1983.

[38] J. C. Bean, D. B. Chaffin, and A. B. Schultz, “Biomechanical model calculation of muscle contraction forces: a double linear programming method,” *Journal of biomechanics*, vol. 21, no. 1, pp. 59–66, 1988.

[39] center for ergonomics university of michigan, “3D Static Strength Prediction Program,” https://c4e.engin.umich.edu/tools-services/3dsspp-software/, accessed: 2019-04-22.

[40] G. David, “Ergonomic methods for assessing exposure to risk factors for work-related musculoskeletal disorders,” *Occupational medicine*, vol. 55, no. 3, pp. 190–199, 2005.

[41] L. McAtamney and E. N. Corlett, “Rula: a survey method for the investigation of work-related upper limb disorders,” *Applied ergonomics*, vol. 24, no. 2, pp. 91–99, 1993.

[42] S. Hignett and L. McAtamney, “Rapid entire body assessment (reba),” *Applied ergonomics*, vol. 31, no. 2, pp. 201–205, 2000.

[43] O. Karhu, R. Härkönen, P. Sorvali, and P. Vepsäläinen, “Observing working postures in industry: Examples of owas application,” *Applied Ergonomics*, vol. 12, no. 1, pp. 13–17, 1981.

[44] W. S. Marras, S. A. Lavender, S. E. Leurgans, F. A. Fathallah, S. A. Ferguson, W. Gary Allread, and S. L. Rajulu, "Biomechanical risk factors for occupationally related low back disorders," *Ergonomics*, vol. 38, no. 2, pp. 377–410, 1995.

[45] J. Ryu, M. Diraneyya, C. Haas, and E. Abdel-Rahman, "Automated rule-based assessment of bricklaying," *Medical Engineering & Physics*, 2019, submitted.

[46] R. Norman, R. Wells, P. Neumann, J. Frank, H. Shannon, M. Kerr, T. O. U. B. P. Study *et al.*, "A comparison of peak vs cumulative physical work exposure risk factors for the reporting of low back pain in the automotive industry," *Clinical biomechanics*, vol. 13, no. 8, pp. 561–573, 1998.

[47] G. Wu, S. Siegler, P. Allard, C. Kirtley, A. Leardini, D. Rosenbaum, M. Whittle, D. D DLima, L. Cristofolini, H. Witte *et al.*, "Isb recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motionpart i: ankle, hip, and spine," *Journal of biomechanics*, vol. 35, no. 4, pp. 543–548, 2002.

[48] G. Wu, F. C. Van der Helm, H. D. Veeger, M. Makhsous, P. Van Roy, C. Anglin, J. Nagels, A. R. Karduna, K. McQuade, X. Wang *et al.*, "Isb recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motionpart ii: shoulder, elbow, wrist and hand," *Journal of biomechanics*, vol. 38, no. 5, pp. 981–992, 2005.

[49] R. L. Huston, *Fundamentals of Biomechanics.* CRC Press, 2013.

[50] A. Alwasel, E. M. Abdel-Rahman, C. T. Haas, and S. Lee, "Experience, productivity, and musculoskeletal injury among masonry workers," *Journal of Construction Engineering and Management*, vol. 143, no. 6, p. 05017003, 2017.

[51] S. Kumar, "Moment arms of spinal musculature determined from ct scans," *Clinical Biomechanics*, vol. 3, no. 3, pp. 137–144, 1988.

[52] S. McGill, N. Patt, and R. Norman, "Measurement of the trunk musculature of active males using ct scan radiography: implications for force and moment generating capacity about the l4l5 joint," *Journal of biomechanics*, vol. 21, no. 4, pp. 329–341, 1988.

[53] D. B. Chaffin, M. S. Redfern, M. Erig, and S. A. Goldstein, "Lumbar muscle size and locations from ct scans of 96 women of age 40 to 63 years," *Clinical Biomechanics*, vol. 5, no. 1, pp. 9–16, 1990.

[54] Xsens, "MVN inertial motion capture system," https://www.xsens.com/products/xsens-mvn-animate, accessed: 2019-03-31.

[55] Noitom, "Perception Neuron inertial motion capture system," https://neuronmocap.com/store?field_category_value=PN, accessed: 2019-03-31.

[56] W. S. Robertson, "Biovision (bvh) in matlab," https://github.com/wspr/bvh-matlab, 2012.

[57] Xsens, "MVN Studio," https://www.xsens.com/mvn-studio-download-responsive, accessed: 2019-04-02.

[58] Noitom, "Axis Neuron," http://neuronmocap.com/content/axis-neuron-software, accessed: 2019-04-02.

[59] D. B. Chaffin, "Development of computerized human static strength simulation model for job design," *Human Factors and Ergonomics in Manufacturing & Service Industries*, vol. 7, no. 4, pp. 305–322, 1997.

[60] Bertec, "Force Plate 4060-05-PT-1000," www.bertec.com/products/force-plates, accessed: 2019-04-02.

[61] Vicon, "Vicon Optical Motion Capture System," https://www.vicon.com/products/camera-systems/vantage, accessed: 2019-04-02.

[62] A. Alwasel, "Use of kinematics to minimize construction workers' risk of musculoskeletal injury," PhD dissertation, University of Waterloo, 2017.

[63] S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman, and D. G. Thelen, "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.

[64] G. S. Faber, A. Koopman, I. Kingma, C. Chang, J. Dennerlein, and J. van Dieën, "Continuous ambulatory hand force monitoring during manual materials handling using instrumented force shoes and an inertial motion capture suit," *Journal of biomechanics*, vol. 70, pp. 235–241, 2018.

[65] J. S. Boschman, H. F. van der Molen, J. K. Sluiter, and M. H. Frings-Dresen, "Musculoskeletal disorders among construction workers: a one-year follow-up study," *BMC musculoskeletal disorders*, vol. 13, no. 1, p. 196, 2012.

[66] E. P. Hanavan Jr, "A mathematical model of the human body," AIR FORCE AEROSPACE MEDICAL RESEARCH LAB WRIGHT-PATTERSON AFB OH, Tech. Rep., 1964.

[67] D. Bartelink, "The role of abdominal pressure in relieving the pressure on the lumbar intervertebral discs," *The Journal of bone and joint surgery. British volume*, vol. 39, no. 4, pp. 718–725, 1957.

[68] S. M. McGill and R. W. Norman, "Effects of an anatomically detailed erector spinae model on l4l5 disc compression and shear," *Journal of biomechanics*, vol. 20, no. 6, pp. 591–600, 1987.

[69] J. T. McConville, C. E. Clauser, T. D. Churchill, J. Cuzzi, and I. Kaleps, "Anthropometric relationships of body and body segment moments of inertia," ANTHROPOLOGY RESEARCH PROJECT INC YELLOW SPRINGS OH, Tech. Rep., 1980.

APPENDICES

Appendix A

Matlab Code

A.1 Kinematic Data from Motion File

A.1.1 Read CALC file

```
% This funciton reads calculation file (.calc) exported from
% axis neuron software and perception neuron suite, for Noitom company.

% The function reads the columnt of interest that contains:
%  sensors locations, velocities, quanterinions, accelerations and
% angualr velocities for all 21 sensors in the calculation file.
% Note that some of them may be copies of each other, depends actual number of sensors used.


function [Calc_Data] = read_calc_file(calc_filename)

% add a file extension if necessary:
if ~strncmpi(fliplr(calc_filename),'clac.',5)
  calc_filename = [calc_filename,'.calc'];
end

%% counter for number of lines

fid=fopen(calc_filename,'r');

file_data_cell = textscan(fid, '%s',10000000,'delimiter', '\n');
file_data = file_data_cell{1,1};

no_of_raws=size(file_data_cell{1},1);

fclose(fid);

%% read line by line
```

```
Data=zeros(no_of_raws,338);

for i = 7 : no_of_raws
    tap_locations=strfind(file_data{i},'	');
%     if size(tap_locations,1)==0
%         continue
%     end
    Data(i,1)=str2double(file_data{i}(1:tap_locations(1)-1));
    %endcol=size(tap_locations,2);
    for j=2:338
        Data(i,j)=str2double(file_data{i}(tap_locations(j-1)+1:tap_locations(j)-1));
    end

end



Calc_Data=Data(7:end,:);


% the coloumn are in the order:


% 'pelvic_position_x','pelvic_position_y','pelvic_position_z','pelvic_velocity_x','pelvic_velocity_y','pelvic_velocity_z',
% 'pelvic_quaternion_1','pelvic_quaternion_2','pelvic_quaternion_3','pelvic_quaternion_4','pelvic_accelration_x','pelvic_accelration_y','pelvic_accelration_z',
% 'pelvic_angular_velocity_x','pelvic_angular_velocity_y','pelvic_angular_velocity_z',
%
% 'Right_Thigh_position_x','Right_Thigh_position_y','Right_Thigh_position_z','Right_Thigh_velocity_x','Right_Thigh_velocity_y','Right_Thigh_velocity_z',
% 'Right_Thigh_quaternion_1','Right_Thigh_quaternion_2','Right_Thigh_quaternion_3','Right_Thigh_quaternion_4','Right_Thigh_accelration_x','Right_Thigh_accelration_y','Right_Thigh_accelration_z',
% 'Right_Thigh_angular_velocity_x','Right_Thigh_angular_velocity_y','Right_Thigh_angular_velocity_z',
%
% 'Right_Shank_position_x','Right_Shank_position_y','Right_Shank_position_z','Right_Shank_velocity_x','Right_Shank_velocity_y','Right_Shank_velocity_z',
% 'Right_Shank_quaternion_1','Right_Shank_quaternion_2','Right_Shank_quaternion_3','Right_Shank_quaternion_4','Right_Shank_accelration_x','Right_Shank_accelration_y','Right_Shank_accelration_z',
% 'Right_Shank_angular_velocity_x','Right_Shank_angular_velocity_y','Right_Shank_angular_velocity_z',
%
% 'Right_Foot_position_x','Right_Foot_position_y','Right_Foot_position_z','Right_Foot_velocity_x','Right_Foot_velocity_y','Right_Foot_velocity_z',
% 'Right_Foot_quaternion_1','Right_Foot_quaternion_2','Right_Foot_quaternion_3','Right_Foot_quaternion_4','Right_Foot_accelration_x','Right_Foot_accelration_y','Right_Foot_accelration_z',
% 'Right_Foot_angular_velocity_x','Right_Foot_angular_velocity_y','Right_Foot_angular_velocity_z',
%
% 'Left_Thigh_position_x','Left_Thigh_position_y','Left_Thigh_position_z','Left_Thigh_velocity_x','Left_Thigh_velocity_y','Left_Thigh_velocity_z',
% 'Left_Thigh_quaternion_1','Left_Thigh_quaternion_2','Left_Thigh_quaternion_3','Left_Thigh_quaternion_4','Left_Thigh_accelration_x','Left_Thigh_accelration_y','Left_Thigh_accelration_z',
% 'Left_Thigh_angular_velocity_x','Left_Thigh_angular_velocity_y','Left_Thigh_angular_velocity_z',
%
% 'Left_Shank_position_x','Left_Shank_position_y','Left_Shank_position_z','Left_Shank_velocity_x','Left_Shank_velocity_y','Left_Shank_velocity_z',
% 'Left_Shank_quaternion_1','Left_Shank_quaternion_2','Left_Shank_quaternion_3','Left_Shank_quaternion_4','Left_Shank_accelration_x','Left_Shank_accelration_y','Left_Shank_accelration_z',
% 'Left_Shank_angular_velocity_x','Left_Shank_angular_velocity_y','Left_Shank_angular_velocity_z',
%
% 'Left_Foot_position_x','Left_Foot_position_y','Left_Foot_position_z','Left_Foot_velocity_x','Left_Foot_velocity_y','Left_Foot_velocity_z',
% 'Left_Foot_quaternion_1','Left_Foot_quaternion_2','Left_Foot_quaternion_3','Left_Foot_quaternion_4','Left_Foot_accelration_x','Left_Foot_accelration_y','Left_Foot_accelration_z',
% 'Left_Foot_angular_velocity_x','Left_Foot_angular_velocity_y','Left_Foot_angular_velocity_z',
%
% 'Right_Shoulder_position_x','Right_Shoulder_position_y','Right_Shoulder_position_z','Right_Shoulder_velocity_x','Right_Shoulder_velocity_y','Right_Shoulder_velocity_z',
% 'Right_Shoulder_quaternion_1','Right_Shoulder_quaternion_2','Right_Shoulder_quaternion_3','Right_Shoulder_quaternion_4','Right_Shoulder_accelration_x','Right_Shoulder_accelration_y','Right_Shoulder_accelration_z',
% 'Right_Shoulder_angular_velocity_x','Right_Shoulder_angular_velocity_y','Right_Shoulder_angular_velocity_z',
%
% 'Right_UpperArm_position_x','Right_UpperArm_position_y','Right_UpperArm_position_z','Right_UpperArm_velocity_x','Right_UpperArm_velocity_y','Right_UpperArm_velocity_z',
```

```
% 'Right_UpperArm_quaternion_1','Right_UpperArm_quaternion_2','Right_UpperArm_quaternion_3','Right_UpperArm_quaternion_4','Right_UpperArm_accelration_x','Right_UpperArm_accelration_y','Right_UpperArm_accelration_z',
% 'Right_UpperArm_angular_velocity_x','Right_UpperArm_angular_velocity_y','Right_UpperArm_angular_velocity_z',
%
% 'Right_ForeArm_position_x','Right_ForeArm_position_y','Right_ForeArm_position_z','Right_ForeArm_velocity_x','Right_ForeArm_velocity_y','Right_ForeArm_velocity_z',
% 'Right_ForeArm_quaternion_1','Right_ForeArm_quaternion_2','Right_ForeArm_quaternion_3','Right_ForeArm_quaternion_4','Right_ForeArm_accelration_x','Right_ForeArm_accelration_y','Right_ForeArm_accelration_z',
% 'Right_ForeArm_angular_velocity_x','Right_ForeArm_angular_velocity_y','Right_ForeArm_angular_velocity_z',
%
% 'Right_Hand_position_x','Right_Hand_position_y','Right_Hand_position_z','Right_Hand_velocity_x','Right_Hand_velocity_y','Right_Hand_velocity_z',
% 'Right_Hand_quaternion_1','Right_Hand_quaternion_2','Right_Hand_quaternion_3','Right_Hand_quaternion_4','Right_Hand_accelration_x','Right_Hand_accelration_y','Right_Hand_accelration_z',
% 'Right_Hand_angular_velocity_x','Right_Hand_angular_velocity_y','Right_Hand_angular_velocity_z',
%
% 'Left_Shoulder_position_x','Left_Shoulder_position_y','Left_Shoulder_position_z','Left_Shoulder_velocity_x','Left_Shoulder_velocity_y','Left_Shoulder_velocity_z',
% 'Left_Shoulder_quaternion_1','Left_Shoulder_quaternion_2','Left_Shoulder_quaternion_3','Left_Shoulder_quaternion_4','Left_Shoulder_accelration_x','Left_Shoulder_accelration_y','Left_Shoulder_accelration_z',
% 'Left_Shoulder_angular_velocity_x','Left_Shoulder_angular_velocity_y','Left_Shoulder_angular_velocity_z',
%
% 'Left_UpperArm_position_x','Left_UpperArm_position_y','Left_UpperArm_position_z','Left_UpperArm_velocity_x','Left_UpperArm_velocity_y','Left_UpperArm_velocity_z',
% 'Left_UpperArm_quaternion_1','Left_UpperArm_quaternion_2','Left_UpperArm_quaternion_3','Left_UpperArm_quaternion_4','Left_UpperArm_accelration_x','Left_UpperArm_accelration_y','Left_UpperArm_accelration_z',
% 'Left_UpperArm_angular_velocity_x','Left_UpperArm_angular_velocity_y','Left_UpperArm_angular_velocity_z',
%
% 'Left_ForeArm_position_x','Left_ForeArm_position_y','Left_ForeArm_position_z','Left_ForeArm_velocity_x','Left_ForeArm_velocity_y','Left_ForeArm_velocity_z',
% 'Left_ForeArm_quaternion_1','Left_ForeArm_quaternion_2','Left_ForeArm_quaternion_3','Left_ForeArm_quaternion_4','Left_ForeArm_accelration_x','Left_ForeArm_accelration_y','Left_ForeArm_accelration_z',
% 'Left_ForeArm_angular_velocity_x','Left_ForeArm_angular_velocity_y','Left_ForeArm_angular_velocity_z',
%
% 'Left_Hand_position_x','Left_Hand_position_y','Left_Hand_position_z','Left_Hand_velocity_x','Left_Hand_velocity_y','Left_Hand_velocity_z',
% 'Left_Hand_quaternion_1','Left_Hand_quaternion_2','Left_Hand_quaternion_3','Left_Hand_quaternion_4','Left_Hand_accelration_x','Left_Hand_accelration_y','Left_Hand_accelration_z',
% 'Left_Hand_angular_velocity_x','Left_Hand_angular_velocity_y','Left_Hand_angular_velocity_z',
%
% 'Head_position_x','Head_position_y','Head_position_z','Head_velocity_x','Head_velocity_y','Head_velocity_z',
% 'Head_quaternion_1','Head_quaternion_2','Head_quaternion_3','Head_quaternion_4','Head_accelration_x','Head_accelration_y','Head_accelration_z',
% 'Head_angular_velocity_x','Head_angular_velocity_y','Head_angular_velocity_z',
%
% 'Neck_position_x','Neck_position_y','Neck_position_z','Neck_velocity_x','Neck_velocity_y','Neck_velocity_z',
% 'Neck_quaternion_1','Neck_quaternion_2','Neck_quaternion_3','Neck_quaternion_4','Neck_accelration_x','Neck_accelration_y','Neck_accelration_z',
% 'Neck_angular_velocity_x','Neck_angular_velocity_y','Neck_angular_velocity_z',
%
% 'Spine3_position_x','Spine3_position_y','Spine3_position_z','Spine3_velocity_x','Spine3_velocity_y','Spine3_velocity_z',
% 'Spine3_quaternion_1','Spine3_quaternion_2','Spine3_quaternion_3','Spine3_quaternion_4','Spine3_accelration_x','Spine3_accelration_y','Spine3_accelration_z',
% 'Spine3_angular_velocity_x','Spine3_angular_velocity_y','Spine3_angular_velocity_z',
%
% 'Spine2_position_x','Spine2_position_y','Spine2_position_z','Spine2_velocity_x','Spine2_velocity_y','Spine2_velocity_z',
% 'Spine2_quaternion_1','Spine2_quaternion_2','Spine2_quaternion_3','Spine2_quaternion_4','Spine2_accelration_x','Spine2_accelration_y','Spine2_accelration_z',
% 'Spine2_angular_velocity_x','Spine2_angular_velocity_y','Spine2_angular_velocity_z',
%
% 'Spine1_position_x','Spine1_position_y','Spine1_position_z','Spine1_velocity_x','Spine1_velocity_y','Spine1_velocity_z',
% 'Spine1_quaternion_1','Spine1_quaternion_2','Spine1_quaternion_3','Spine1_quaternion_4','Spine1_accelration_x','Spine1_accelration_y','Spine1_accelration_z',
```

```
% 'Spine1_angular_velocity_x','Spine1_angular_velocity_y','Spine1_angular_velocity_z',
%
% 'Spine_position_x','Spine_position_y','Spine_position_z','Spine_velocity_x','Spine_velocity_y','
    Spine_velocity_z',
% 'Spine_quaternion_1','Spine_quaternion_2','Spine_quaternion_3','Spine_quaternion_4','
    Spine_accelration_x','Spine_accelration_y','Spine_accelration_z',
% 'Spine_angular_velocity_x','Spine_angular_velocity_y','Spine_angular_velocity_z','Contact_Left','
    Contact_Right'
```

A.1.2 Kinematics from MVNX

```
%% load MVNX

% FileName='dynamic';
% MVNX_Tree=load_mvnx(FileName);

%% Kinematics from MVNX

% This funtion finds joint centers' position, accelration and segemnts'
% angular velocties, angular accelerations, rotational matrix from segment CS to global CS,
% and pelvic origin locaiton in global CS from mvnx file


% The funtion outputs are for each segment of 15 segment body model, The segments are:
% 1: Pelvic, 2: Torso, 3: Head&Neck, 4: Right upper arm, 5: Right Forearm, 6: Right hand
% 7: Left upper arm, 8: Left Forearm, 9: Left hand, 10: Right Thigh, 11: Right Shank,
% 12: Right Foot, 13: Left Thigh, 14: Left Shank and 15: Left Foot

% all inputs from mvnx are in global frame

function [A,omega,alpha,zeta_pelvis,R_L_G,Vel] = Kinematics_from_mvnx(MVNX_Tree,Wc)

% Input:
% MVNX_Tree, contains the data as read by "load_mvnx.m" function provided by xsens
% Wc: desired cut-off frequency for filtering the data with 2nd order butterworth low pass filter


% Outputs:
% A : acceleration of each segment origin. 3xnx15 matrix, 3d vector for each frame for each segment
% omega : angular velocity of each segment origin. 3xnx15 matrix, 3d vector for each frame for each
    segment
% alpha : angular acceleration of each segment origin. 3xnx15 matrix, 3d vector for each frame for each
    segment
% zeta_pelvis : Pelvic CS origin for each frame (framesx3) matrix, 3D vector for each frame
% R_L_G: Rotational matrix from local segment CS to global CS (referecne frame)
%       (3x3x15xframes), 3X3 matrix for each segment for each frame (15 segments)


%% reading kinematics

MVNX_Data=MVNX_Tree.subject.frames.frame; %data of intrest from mvnx
frame_rate=MVNX_Tree.subject.frameRate; % frame rate of data

frames_time = [MVNX_Data.time];
experiment_frames=(frames_time>0);
MVNX_Data=MVNX_Data(experiment_frames); %remove t-pose and n-pose frames

R_L_G=zeros(3,3,15,size(MVNX_Data,2)); %Rotational Matrix from Local to Global (3x3) for each body
    segment for each time frame
zeta_pelvis=zeros(size(MVNX_Data,2),3); %pelvic origin in grloba CS

A=zeros(3,size(MVNX_Data,2),15); %segments' origin acceleration (joints' acceleration)
omega=zeros(3,size(MVNX_Data,2),15); %segments' angular velocity
alpha=zeros(3,size(MVNX_Data,2),15); %segments' angular acceleration

% this rotational matrix is used to rotate the coordinate system from the
% one used in mvnx (z is the vertical direction) to the one used in
% dumas2006 paper and that's recommended by ISB (WU2005)

R=[1 0 0;0 0 1;0 -1 0]; %rotational matrix from mvnx global CS to dumas2006 global CS (ISB
    recommendation, WU2008)
inv_R=R';

%%

```

```
Orientation=zeros(size(MVNX_Data,2),92); %segments orientation
position=zeros(size(MVNX_Data,2),69); %segments origins' velocity
velocity=zeros(size(MVNX_Data,2),69); %segments origins' position
acceleration=zeros(size(MVNX_Data,2),69); %segments origins' acceleration
angularVelocity=zeros(size(MVNX_Data,2),69); %segments angular velocity
angularAcceleration=zeros(size(MVNX_Data,2),69); %segments angular acceleration


for i=1:size(MVNX_Data,2)

    Orientation(i,:) = MVNX_Data(i).orientation;
    position(i,:) = MVNX_Data(i).position;
    velocity(i,:) = MVNX_Data(i).velocity;
    acceleration(i,:) = MVNX_Data(i).acceleration;
    angularVelocity(i,:) = MVNX_Data(i).angularVelocity;
    angularAcceleration(i,:) = MVNX_Data(i).angularAcceleration;

end

%% filtering with 2nd order low pass butterworth filter

Wn=Wc/(frame_rate/2); %cut-off frequency over half the original frequency
[b,a]=butter(2,Wn,'low'); %butter filter factors

acceleration=filtfilt(b,a,acceleration); %Applying butter filter on acceleration
angularVelocity=filtfilt(b,a,angularVelocity); %Applying butter filter on angular velocity
angularAcceleration=filtfilt(b,a,angularAcceleration); %Applying butter filter on angular acceleration
position=filtfilt(b,a,position); %Applying butter filter on position
Orientation=filtfilt(b,a,Orientation); %Applying butter filter on orientation

%% Assigning to output variables

for i=1:size(MVNX_Data,2)

    %the rotaional matrix obtained from orientation matrix is from mvnx
    %local to mvnx global, so we multiplied it by R and inv_R to make
    %it from our defined local to our defined global

    R_L_G(:,:,1,i) = R*quat2rotm(Orientation(i,1:4))*inv_R; %#ok<*MINV> %rotational matrix from pelvic to global
    R_L_G(:,:,2,i) = R*quat2rotm(Orientation(i,5:8))*inv_R; %rotational matrix from torso to global
    R_L_G(:,:,3,i) = R*quat2rotm(Orientation(i,25:28))*inv_R; %rotational matrix from head to global
    R_L_G(:,:,4,i) = R*quat2rotm(Orientation(i,33:36))*inv_R; %rotational matrix from right upperArm to global
    R_L_G(:,:,5,i) = R*quat2rotm(Orientation(i,37:40))*inv_R; %rotational matrix from right forearm to global
    R_L_G(:,:,6,i) = R*quat2rotm(Orientation(i,41:44))*inv_R; %rotational matrix from right hand to global
    R_L_G(:,:,7,i) = R*quat2rotm(Orientation(i,49:52))*inv_R; %rotational matrix from left upperArm to global
    R_L_G(:,:,8,i) = R*quat2rotm(Orientation(i,53:56))*inv_R; %rotational matrix from left forearm to global
    R_L_G(:,:,9,i) = R*quat2rotm(Orientation(i,57:60))*inv_R; %rotational matrix from left hand to global
    R_L_G(:,:,10,i) = R*quat2rotm(Orientation(i,61:64))*inv_R; %rotational matrix from right thight to global
    R_L_G(:,:,11,i) = R*quat2rotm(Orientation(i,65:68))*inv_R; %rotational matrix from right Shank to global
    R_L_G(:,:,12,i) = R*quat2rotm(Orientation(i,69:72))*inv_R; %rotational matrix from right foot to global
    R_L_G(:,:,13,i) = R*quat2rotm(Orientation(i,77:80))*inv_R; %rotational matrix from left thight to global
    R_L_G(:,:,14,i) = R*quat2rotm(Orientation(i,81:84))*inv_R; %rotational matrix from left Shank to global
    R_L_G(:,:,15,i) = R*quat2rotm(Orientation(i,85:88))*inv_R; %rotational matrix from left foot to global

end



%pelvic origin position, but shifted as the origin in mvnx is centeral hip, but in this model it is L5/S1
% zeta_pelvis(:,:) = position(:,1:3)+(MVNX_Tree.subject.segments.segment(1).points.point(2).pos_s);
zeta_pelvis(:,:) = position(:,4:6);
```

```
%Rotated by rotational matrix from mvn CS to our model's CS.
%The same rotation is applied to all variables
zeta_pelvis=R*zeta_pelvis';

A(:,:,1) = R*acceleration(:,1:3)'; % pelvic origin acceleration
A(:,:,2) = R*acceleration(:,4:6)';  % torso origin acceleration
A(:,:,3) = R*acceleration(:,16:18)';  % head&neck origin acceleration
A(:,:,4) = R*acceleration(:,25:27)';  % right upperarm origin acceleration
A(:,:,5) = R*acceleration(:,28:30)';  % right forearm acceleration
A(:,:,6) = R*acceleration(:,31:33)';  % right hand acceleration
A(:,:,7) = R*acceleration(:,37:39)';  % left upperarm origin acceleration
A(:,:,8) = R*acceleration(:,40:42)';  % left forearm acceleration
A(:,:,9) = R*acceleration(:,43:45)';  % left hand acceleration
A(:,:,10) = R*acceleration(:,46:48)';  % right thight origin acceleration
A(:,:,11) = R*acceleration(:,49:51)';  % right shank acceleration
A(:,:,12) = R*acceleration(:,52:54)';  % right foot acceleration
A(:,:,13) = R*acceleration(:,58:60)';  % left thight origin acceleration
A(:,:,14) = R*acceleration(:,61:63)';  % left shank acceleration
A(:,:,15) = R*acceleration(:,64:66)';  % left foot acceleration

A(2,:,:)=A(2,:,:)+9.807; %add gravity acceleration

omega(:,:,1) = R*angularVelocity(:,1:3)'; % pelvic origin angularVelocity
omega(:,:,2) = R*angularVelocity(:,4:6)';  % torso origin angularVelocity
omega(:,:,3) = R*angularVelocity(:,16:18)';  % head&neck origin angularVelocity
omega(:,:,4) = R*angularVelocity(:,25:27)';  % right upperarm origin angularVelocity
omega(:,:,5) = R*angularVelocity(:,28:30)';  % right forearm angularVelocity
omega(:,:,6) = R*angularVelocity(:,31:33)';  % right hand angularVelocity
omega(:,:,7) = R*angularVelocity(:,37:39)';  % left upperarm origin angularVelocity
omega(:,:,8) = R*angularVelocity(:,40:42)';  % left forearm angularVelocity
omega(:,:,9) = R*angularVelocity(:,43:45)';  % left hand angularVelocity
omega(:,:,10) = R*angularVelocity(:,46:48)';  % right thight origin angularVelocity
omega(:,:,11) = R*angularVelocity(:,49:51)';  % right shank angularVelocity
omega(:,:,12) = R*angularVelocity(:,52:54)';  % right foot angularVelocity
omega(:,:,13) = R*angularVelocity(:,58:60)';  % left thight origin angularVelocity
omega(:,:,14) = R*angularVelocity(:,61:63)';  % left shank angularVelocity
omega(:,:,15) = R*angularVelocity(:,64:66)';  % left foot angularVelocity

alpha(:,:,1) = R*angularAcceleration(:,1:3)'; % pelvic origin angularAcceleration
alpha(:,:,2) = R*angularAcceleration(:,4:6)';  % torso origin angularAcceleration
alpha(:,:,3) = R*angularAcceleration(:,16:18)';  % head&neck origin angularAcceleration
alpha(:,:,4) = R*angularAcceleration(:,25:27)';  % right upperarm origin angularAcceleration
alpha(:,:,5) = R*angularAcceleration(:,28:30)';  % right forearm angularAcceleration
alpha(:,:,6) = R*angularAcceleration(:,31:33)';  % right hand angularAcceleration
alpha(:,:,7) = R*angularAcceleration(:,37:39)';  % left upperarm origin angularAcceleration
alpha(:,:,8) = R*angularAcceleration(:,40:42)';  % left forearm angularAcceleration
alpha(:,:,9) = R*angularAcceleration(:,43:45)';  % left hand angularAcceleration
alpha(:,:,10) = R*angularAcceleration(:,46:48)';  % right thight origin angularAcceleration
alpha(:,:,11) = R*angularAcceleration(:,49:51)';  % right shank angularAcceleration
alpha(:,:,12) = R*angularAcceleration(:,52:54)';  % right foot angularAcceleration
alpha(:,:,13) = R*angularAcceleration(:,58:60)';  % left thight origin angularAcceleration
alpha(:,:,14) = R*angularAcceleration(:,61:63)';  % left shank angularAcceleration
alpha(:,:,15) = R*angularAcceleration(:,64:66)';  % left foot angularAcceleration

Vel(:,:,1) = R*velocity(:,1:3)'; % pelvic origin velocity
Vel(:,:,2) = R*velocity(:,4:6)';  % torso origin velocity
Vel(:,:,3) = R*velocity(:,16:18)';  % head&neck origin velocity
Vel(:,:,4) = R*velocity(:,25:27)';  % right upperarm origin velocity
Vel(:,:,5) = R*velocity(:,28:30)';  % right forearm velocity
Vel(:,:,6) = R*velocity(:,31:33)';  % right hand velocity
Vel(:,:,7) = R*velocity(:,37:39)';  % left upperarm origin velocity
Vel(:,:,8) = R*velocity(:,40:42)';  % left forearm velocity
Vel(:,:,9) = R*velocity(:,43:45)';  % left hand velocity
Vel(:,:,10) = R*velocity(:,46:48)';  % right thight origin velocity
Vel(:,:,11) = R*velocity(:,49:51)';  % right shank velocity
Vel(:,:,12) = R*velocity(:,52:54)';  % right foot velocity
Vel(:,:,13) = R*velocity(:,58:60)';  % left thight origin velocity
Vel(:,:,14) = R*velocity(:,61:63)';  % left shank velocity
Vel(:,:,15) = R*velocity(:,64:66)';  % left foot velocity
```

A.1.3 Kinematics from BVH

```
% This function finds the kinematics from the BVH file

function  [zeta_pelvis,R_L_G] = Kinematics_from_BVH(BVH_file)
```

```
% Inputs:
% BVH_file is the input which is the bvh data as imported using "loadbvh"
% funciton from university of adelaide "https://github.com/wspr/bvh-matlab"

% Outputs:
% zeta_pelvis : Pelvic CS origin for each frame (framesx3) matrix, 3D vector for each frame
% R_L_G: Rotational matrix from local segment CS to global CS (referecne frame)
%       (3x3x15xframes), 3X3 matrix for each segment for each frame (15 segments)
% L: The length of each segment, as the JC specifis the distance between joints and thus the length of each segment.
%       (15X1) vector, length of each segment


%% extract  rotational matrix and pelvis origion

pelvic_origin_translation(:,:)=BVH_file(1).Dxyz; %the translational part of the translation and rotation matrix
zeta_pelvis=pelvic_origin_translation'/100; %pelvic zeta (origin of pelvic location in global CS) in meters

R_L_G=zeros(3,3,15,size(zeta_pelvis,1));

R_L_G(:,:,1,:)=BVH_file(1).trans(1:3,1:3,:); %rotational matrix from pelvic to global, for all frames
R_L_G(:,:,2,:)=BVH_file(11).trans(1:3,1:3,:); %rotational matrix from torso to global, for all frames
R_L_G(:,:,3,:)=BVH_file(15).trans(1:3,1:3,:); %rotational matrix from head&neck to global, for all frames
R_L_G(:,:,4,:)=BVH_file(18).trans(1:3,1:3,:); %rotational matrix from right upperarm to global, for all frames
R_L_G(:,:,5,:)=BVH_file(19).trans(1:3,1:3,:); %rotational matrix from right forearm to global, for all frames
R_L_G(:,:,6,:)=BVH_file(20).trans(1:3,1:3,:); %rotational matrix from right hand to global, for all frames
R_L_G(:,:,7,:)=BVH_file(46).trans(1:3,1:3,:); %rotational matrix from left upperarm to global, for all frames
R_L_G(:,:,8,:)=BVH_file(47).trans(1:3,1:3,:); %rotational matrix from left forearm to global, for all frames
R_L_G(:,:,9,:)=BVH_file(48).trans(1:3,1:3,:); %rotational matrix from left hand to global, for all frames
R_L_G(:,:,10,:)=BVH_file(2).trans(1:3,1:3,:); %rotational matrix from right thigh to global, for all frames
R_L_G(:,:,11,:)=BVH_file(3).trans(1:3,1:3,:); %rotational matrix from right shank to global, for all frames
R_L_G(:,:,12,:)=BVH_file(4).trans(1:3,1:3,:); %rotational matrix from right foot to global, for all frames
R_L_G(:,:,13,:)=BVH_file(6).trans(1:3,1:3,:); %rotational matrix from left thigh to global, for all frames
R_L_G(:,:,14,:)=BVH_file(7).trans(1:3,1:3,:); %rotational matrix from left shank to global, for all frames
R_L_G(:,:,15,:)=BVH_file(8).trans(1:3,1:3,:); %rotational matrix from left foot to global, for all frames


%% rotate from BVH CS to the model's CS

% rotate the local coordinate system from bvh local to our model's local
% (as defined by Dumas2006 and Wu2008 in ISB recommmendations)

R1=[0 1 0;
    -1 0 0;
    0 0 1]; % rotation matrix from upper extermities bvh local to our model's local

R2=[0 0 -1;
    0 1 0;
    1 0 0]; % rotation matrix from other segments' bvh local to our model's local

for frame =1:size(R_L_G,4)

    R_L_G(:,:,1,frame)=R_L_G(:,:,1,frame)*R2;
    R_L_G(:,:,2,frame)=R_L_G(:,:,2,frame)*R2;
    R_L_G(:,:,3,frame)=R_L_G(:,:,3,frame)*R2;
    R_L_G(:,:,4,frame)=R_L_G(:,:,4,frame)*R1;
    R_L_G(:,:,5,frame)=R_L_G(:,:,5,frame)*R1;
    R_L_G(:,:,6,frame)=R_L_G(:,:,6,frame)*R1;
    R_L_G(:,:,7,frame)=R_L_G(:,:,7,frame)*R1;
    R_L_G(:,:,8,frame)=R_L_G(:,:,8,frame)*R1;
    R_L_G(:,:,9,frame)=R_L_G(:,:,9,frame)*R1;
```

```
    R_L_G(:,:,10,frame)=R_L_G(:,:,10,frame)*R2;
    R_L_G(:,:,11,frame)=R_L_G(:,:,11,frame)*R2;
    R_L_G(:,:,12,frame)=R_L_G(:,:,12,frame)*R2;
    R_L_G(:,:,13,frame)=R_L_G(:,:,13,frame)*R2;
    R_L_G(:,:,14,frame)=R_L_G(:,:,14,frame)*R2;
    R_L_G(:,:,15,frame)=R_L_G(:,:,15,frame)*R2;

end
```

A.1.4 Kinematics from CALC

```
% This funtion finds sensor's location, linear acceleration,
% angular velocity and angular acceleration from .calc file

% The funtion outputs are for the sensor on each segment of 15 segment body model, The segments are:
% 1: Pelvic, 2: Torso, 3: Head&Neck, 4: Right upper arm, 5: Right Forearm, 6: Right hand
% 7: Left upper arm, 8: Left Forearm, 9: Left hand, 10: Right Thigh, 11: Right Shank,
% 12: Right Foot, 13: Left Thigh, 14: Left Shank and 15: Left Foot

% all inputs from calc are in "sensor global" CS except for location, the
% outputs are in the same CS except for location
% the loction input is in calc BVH CS and location output is in 3D BVH CS

function [A,omega,alpha,sensor_location] = Kinematics_from_calc(calc_data)

% Input: calc_data, contains the data as expressed in calc file exported in sensor global CS,
% only numeric data starting from the first frame to the last frame, size: nx336
% wehre n is the number of frames, 336 is 16 data element for each of the
% 21 segments in .calc file. the segments are, in order:

% 1:pelvic, 2:Right thigh, 3:right shank, 4:right foot, 5:left thigh, 6:left shank
% 7:Left foot, 8:right shoulder, 9:right upperarm, 10:right forearm, 11:right hand
% 12:left shoulder, 13:left upperarm, 14:left forearm, 15:left hand, 16:head, 17:neck
% 18:spine3 ,19:spine2, 20:spine1, 21:spine

% note: 16 and 17 are exactly the same (copies) for quanterion, acceleration and angular velocity
% note: 18,19 and 20 are exactly the same (copies) for quanterion, acceleration and angular velocity
% note: 1 and 21 are exactly the same (copies) for quanterion, acceleration and angular velocity

% 16 data elemetns are in arranged in the following order:
% position x,y&z, velocity x,y&z, quanterion r,i,j&k, acceleration x,y&z and angular velocity x,y&z

% Outputs:
% A : sensor acceleration of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% omega : angular velocity of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% alpha : angular acceleration of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% sensor_location : coordinates of each sensor's location with respect to pelvis origion in bvh CS, 3xnx15 matrix, 3d vector for each frame for each segment

%% read from calc_data matrix, and permute to fit the desired format

sensor_location_from_calc=zeros(size(calc_data,1),3,21);
A_from_calc=zeros(size(calc_data,1),3,21);
omega_from_calc=zeros(size(calc_data,1),3,21);
alpha_from_calc=zeros(size(calc_data,1)-1,3,21);

for i=0:20
    sensor_location_from_calc(:,1:3,i+1)=calc_data(:,16*i+1:16*i+3); %segment i+1 sensor location
    A_from_calc(:,1:3,i+1)=calc_data(:,16*i+11:16*i+13)*9.81; %segment i+1 acceleration in m/sec^2
    omega_from_calc(:,1:3,i+1)=calc_data(:,16*i+14:16*i+16); %segment i+1 angular velocity in rad/sec
    alpha_from_calc(:,1:3,i+1)=diff(omega_from_calc(:,1:3,i+1))*121; %segment i+1 angular acceleration in rad/sec^2 (differentiation of angular velocity)
end

sensor_location_from_calc = permute(sensor_location_from_calc,[2,1,3]);
A_from_calc = permute(A_from_calc,[2,1,3]);
omega_from_calc = permute(omega_from_calc,[2,1,3]);
alpha_from_calc = permute(alpha_from_calc,[2,1,3]);

%% take only the segments we are interested in, in the order we are interested in

R_sensor_BVH=[1 0 0;0 0 1;0 1 0]; %change reference frame from sensor global to BVH
R_bvhcalc_BVH=[1 0 0; 0 0 -1; 0 1 0]; %change reference frome for sesnor location from bvh of .calc, to bvh of .3d

```

```
%1pelvic
sensor_location(:,:,1)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,1);
A(:,:,1)=R_sensor_BVH*A_from_calc(:,:,1);
omega(:,:,1)=R_sensor_BVH*omega_from_calc(:,:,1);
alpha(:,:,1)=R_sensor_BVH*alpha_from_calc(:,:,1);

%2Torso
sensor_location(:,:,2)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,18);
A(:,:,2)=R_sensor_BVH*A_from_calc(:,:,18);
omega(:,:,2)=R_sensor_BVH*omega_from_calc(:,:,18);
alpha(:,:,2)=R_sensor_BVH*alpha_from_calc(:,:,18);

%3head&neck
sensor_location(:,:,3)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,16);
A(:,:,3)=R_sensor_BVH*A_from_calc(:,:,16);
omega(:,:,3)=R_sensor_BVH*omega_from_calc(:,:,16);
alpha(:,:,3)=R_sensor_BVH*alpha_from_calc(:,:,16);

%4Right upperarm
sensor_location(:,:,4)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,9);
A(:,:,4)=R_sensor_BVH*A_from_calc(:,:,9);
omega(:,:,4)=R_sensor_BVH*omega_from_calc(:,:,9);
alpha(:,:,4)=R_sensor_BVH*alpha_from_calc(:,:,9);

%5Right forearm
sensor_location(:,:,5)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,10);
A(:,:,5)=R_sensor_BVH*A_from_calc(:,:,10);
omega(:,:,5)=R_sensor_BVH*omega_from_calc(:,:,10);
alpha(:,:,5)=R_sensor_BVH*alpha_from_calc(:,:,10);

%6Right hand
sensor_location(:,:,6)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,11);
A(:,:,6)=R_sensor_BVH*A_from_calc(:,:,11);
omega(:,:,6)=R_sensor_BVH*omega_from_calc(:,:,11);
alpha(:,:,6)=R_sensor_BVH*alpha_from_calc(:,:,11);

%7Left upperarm
sensor_location(:,:,7)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,13);
A(:,:,7)=R_sensor_BVH*A_from_calc(:,:,13);
omega(:,:,7)=R_sensor_BVH*omega_from_calc(:,:,13);
alpha(:,:,7)=R_sensor_BVH*alpha_from_calc(:,:,13);

%8Left forearm
sensor_location(:,:,8)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,14);
A(:,:,8)=R_sensor_BVH*A_from_calc(:,:,14);
omega(:,:,8)=R_sensor_BVH*omega_from_calc(:,:,14);
alpha(:,:,8)=R_sensor_BVH*alpha_from_calc(:,:,14);

%9Left hand
sensor_location(:,:,9)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,15);
A(:,:,9)=R_sensor_BVH*A_from_calc(:,:,15);
omega(:,:,9)=R_sensor_BVH*omega_from_calc(:,:,15);
alpha(:,:,9)=R_sensor_BVH*alpha_from_calc(:,:,15);

%10Right Thigh
sensor_location(:,:,10)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,2);
A(:,:,10)=R_sensor_BVH*A_from_calc(:,:,2);
omega(:,:,10)=R_sensor_BVH*omega_from_calc(:,:,2);
alpha(:,:,10)=R_sensor_BVH*alpha_from_calc(:,:,2);

%11Right Shank
sensor_location(:,:,11)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,3);
A(:,:,11)=R_sensor_BVH*A_from_calc(:,:,3);
omega(:,:,11)=R_sensor_BVH*omega_from_calc(:,:,3);
alpha(:,:,11)=R_sensor_BVH*alpha_from_calc(:,:,3);

%12Right foot
sensor_location(:,:,12)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,4);
A(:,:,12)=R_sensor_BVH*A_from_calc(:,:,4);
omega(:,:,12)=R_sensor_BVH*omega_from_calc(:,:,4);
alpha(:,:,12)=R_sensor_BVH*alpha_from_calc(:,:,4);

%13Left Thigh
sensor_location(:,:,13)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,5);
A(:,:,13)=R_sensor_BVH*A_from_calc(:,:,5);
omega(:,:,13)=R_sensor_BVH*omega_from_calc(:,:,5);
alpha(:,:,13)=R_sensor_BVH*alpha_from_calc(:,:,5);
```

```
%14Left Shank
sensor_location(:,:,14)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,6);
A(:,:,14)=R_sensor_BVH*A_from_calc(:,:,6);
omega(:,:,14)=R_sensor_BVH*omega_from_calc(:,:,6);
alpha(:,:,14)=R_sensor_BVH*alpha_from_calc(:,:,6);

%15Left foot
sensor_location(:,:,15)=R_bvhcalc_BVH*sensor_location_from_calc(:,:,7);
A(:,:,15)=R_sensor_BVH*A_from_calc(:,:,7);
omega(:,:,15)=R_sensor_BVH*omega_from_calc(:,:,7);
alpha(:,:,15)=R_sensor_BVH*alpha_from_calc(:,:,7);

%% shift global origin to pelvis origion (assume same as sensor) for sensor location
% This way, we express the sensor's location with respect o pelvis origion
for i=1:size(sensor_location,2)
%     for j=1:size(sensor_location,3)
          sensor_location(:,i,:)=sensor_location(:,i,:)-sensor_location(:,i,1);
%     end
end
```

A.2 Segments Anthropometric Data

```
%% Anthropometric data for average Male of specified weight

% This Anthropometric data for males only, at this point, I may add a new
% input for the gender and adjust for the females later

% segment's geometry and interitial property (subject anthropometric)
% are based on the following paper and its Corrigendum :

% Adjustments to McConcille et al. and Young et al. body segment intertial parameters
% R. Dumas, L. Cheze, J. P. Verriest, Journal of biomechanics,  2006

% The function gives the intertia and geometry property of each segment in
% 15 segment body model, The segments are:
% 1: Pelvic, 2:Torso, 3: Head&Neck, 4: Right upper arm, 5: Right Forearm, 6: Right hand
% 7: Left upper arm, 8: Left Forearm, 9: Left hand, 10: Right Thigh, 11: Right Shank,
% 12: Right Foot, 13: Left Thigh, 14: Left Shank and 15: Left Foot

% One change from Dumas2006 definition of segments' CS, is that the Torso
% origin is defined as Lumber Joint Center (LJC) instead of Cervical joint
% center (CJC), the properties are adjusted accordingle (Torso cetner of
% mass and CS origin)


function  [r,zeta,I,M,L] = Anthropometric(Mass,Height,L,M)

% Inputs:

% Mass: the total mass of the subject (weight in kg)
% Height: the upstanding height of the subject (in m)
% L:  an optional input contains segment "length", if not reported, it well
% be considered average and taken from Dumas2006 (in m) adjusted for the given height


%Outputs:

% L: The "Length" of the segment as reported in Dumas2006. if it is measured, it can be taken as input. vector of 15 elemets.
% r: The location of each segment's Center of mass in segment's CS. 3x15 matrix. 3D vector for each segment.
% zeta: The location of each segment's CS origin in preceeding segment's CS. 3x15 matrix. 3D vector for each segment.
%          for the pelvic, zeta is variable for each frame, and thus it is set to zero in this code
% M: The mass of each segment, vector of 15 elements.
% I: The inertia tensor of the segment about it's center of mass, in segment's CS. 3x3x15 matrix. 3x3 dyadic for each segment.


%% Mass
% unit : kg
if ~exist('M','var')
    M(1) = Mass*0.142 ; %Pelvis mass
```

```
    M(2) = Mass*0.333 ; %Torso mass
    M(3) = Mass*0.067 ; %Head & Neck mass
    M(4) = Mass*0.024 ; %Right UpperArm mass
    M(5) = Mass*0.017 ; %Right ForeArm mass
    M(6) = Mass*0.006 ; %Right Hand mass
    M(7) = Mass*0.024 ; %Left UpperArm mass
    M(8) = Mass*0.017 ; %Left ForeArm mass
    M(9) = Mass*0.006 ; %Left Hand mass
    M(10) = Mass*0.123 ; %Right Thigh mass
    M(11) = Mass*0.048 ; %Right Shank mass
    M(12) = Mass*0.012 ; %Right Foot mass
    M(13) = Mass*0.123 ; %Left Thigh mass
    M(14) = Mass*0.048 ; %Left Shank mass
    M(15) = Mass*0.012 ; %Left Foot mass
end
%% Length
% unit : m

if ~exist('L','var') % if user doesn't input segments' lengths , the defauls are
    L(1) = 0.094; %Pelvis length
    L(2) = 0.477; %Torso length
    L(3) = 0.244; %Head & Neck length
    L(4) = 0.271; %Right UpperArm length
    L(5) = 0.283; %Right ForeArm lenght
    L(6) = 0.080; %Right Hand lenght
    L(7) = 0.271; %Left UpperArm length
    L(8) = 0.283; %Left ForeArm length
    L(9) = 0.080; %Left Hand length
    L(10) = 0.432; %Right Thigh length
    L(11) = 0.433; %Right Shank length
    L(12) = 0.183; %Right Foot length
    L(13) = 0.432; %Left Thigh length
    L(14) = 0.433; %Left Shank length
    L(15) = 0.183; %Left Foot length

    L=L*Height/1.77;
end

%% Center of mass
% unit : m


    r(:,1)=L(1)*[2.8 -28.0 -0.6]/100; % Center of Mass in local coordinates for Pelvis
    r(:,2)=L(2)*[-3.6 58 -0.2]/100; % Center of Mass in local coordinates for Torso **
    r(:,3)=L(3)*[2.0 53.6 0.1]/100; % Center of Mass in local coordinates for Head & Neck
    r(:,4)=L(4)*[1.7 -45.2 -2.6]/100; % Center of Mass in local coordinates for Right UpperArm
    r(:,5)=L(5)*[1 -41.7 1.4]/100; % Center of Mass in local coordinates for Right ForeArm
    r(:,6)=L(6)*[8.2 -83.9 7.4]/100; % Center of Mass in local coordinates for Right Hand
    r(:,7)=L(7)*[1.7 -45.2 -2.6]/100; % Center of Mass in local coordinates for Left UpperArm
    r(:,8)=L(8)*[1 -41.7 1.4]/100; % Center of Mass in local coordinates for Left ForeArm
    r(:,9)=L(9)*[8.2 -83.9 7.4]/100; % Center of Mass in local coordinates for Left Hand
    r(:,10)=L(10)*[-4.1 -42.9 3.3]/100; % Center of Mass in local coordinates for Right Thigh
    r(:,11)=L(11)*[-4.8 -41.0 0.7]/100; % Center of Mass in local coordinates for Right Leg
    r(:,12)=L(12)*[38.2 -15.1 2.6]/100; % Center of Mass in local coordinates for Right Foot
    r(:,13)=L(13)*[-4.1 -42.9 3.3]/100; % Center of Mass in local coordinates for Left Thigh
    r(:,14)=L(14)*[-4.8 -41.0 0.7]/100; % Center of Mass in local coordinates for Left Leg
    r(:,15)=L(15)*[38.2 -15.1 2.6]/100; % Center of Mass in local coordinates for Left Foot

%** note: for Torso (body 2), the cetner of mass is expressed in Torso CS
%assuming the origin is LJC, in Dumas2006, the origin is CJC

%% Segment CS origin
% unit : m

    zeta(:,1)=[0 0 0]; % Pelvic origin in the referece frame
    zeta(:,2)=[0 0 0]; % Torso origin in Pelvic Coordinate system
    zeta(:,3)=[0 L(2) 0]; % Head&Neck origin in Torso Coordinate system
    zeta(:,4)=[0.021 L(2)-0.073 0.209]; % Right UpperArm origin in Torso Coordinate system
    zeta(:,5)=[0 -L(4) 0]; % Right ForeArm origin in Right UpperArm Coordinate system
    zeta(:,6)=[0 -L(5) 0]; % Right Hand origin in Right ForeArm Coordinate system
    zeta(:,7)=[0.021 L(2)-0.073 -0.209]; % Left UpperArm origin in Torso Coordinate system
    zeta(:,8)=[0 -L(7) 0]; % Left ForeArm origin in Left UpperArm Coordinate system
    zeta(:,9)=[0 -L(8) 0]; % Left Hand origin in Left ForeArm Coordinate system
    zeta(:,10)=[0.056 -0.075 0.081]; % Right Thigh origin in Pelvic Coordinate system
    zeta(:,11)=[0 -L(10) 0]; % Right Leg origin in Right Thigh Coordinate system
    zeta(:,12)=[0 -L(11) 0]; % Right Foot origin in Right Leg Coordinate system
    zeta(:,13)=[0.056 -0.075 -0.081]; % Left Thigh origin in Pelvic Coordinate system
```

```
    zeta(:,14)=[0 -L(13) 0]; % Left Leg origin in Left Thigh Coordinate system
    zeta(:,15)=[0 -L(14) 0]; % Left Foot origin in Left Leg Coordinate system


%note: for pelvic, the actual origin in the reference frame is variable, but
%for the sake of consistency, it is expressed here as zero vector.

%% Inertia Tensor
% Unit : kg*m^2

    I(:,:,1)=(L(1)*[101 25i 12i;25i 106 8i;12i 8i 95]/100).^2*M(1); % Tensor of Inertia in local coordinates for Pelvis
    I(:,:,2)=(L(2)*[27 18 2;18 25 4i;2 4i 28]/100).^2*M(2); % Tensor of Inertia in local coordinates for Torso
    I(:,:,3)=(L(3)*[28 7i 2i;7i 21 3;2i 3 30]/100).^2*M(3); % Tensor of Inertia in local coordinates for Head & Neck
    I(:,:,4)=(L(4)*[31 6 5;6 14 2;5 2 32]/100).^2*M(4); % Tensor of Inertia in local coordinates for Right UpperArm
    I(:,:,5)=(L(5)*[28 3 2;3 11 8i;2 8i 27]/100).^2*M(5); % Tensor of Inertia in local coordinates for Right ForeArm
    I(:,:,6)=(L(6)*[61 22 15;22 38 20i;15 20i 56]/100).^2*M(6); % Tensor of Inertia in local coordinates for Right Hand
    I(:,:,7)=(L(7)*[31 6 5;6 14 2;5 2 32]/100).^2*M(7); % Tensor of Inertia in local coordinates for Left UpperArm
    I(:,:,8)=(L(8)*[28 3 2;3 11 8i;2 8i 27]/100).^2*M(8); % Tensor of Inertia in local coordinates for Left ForeArm
    I(:,:,9)=(L(9)*[61 22 15;22 38 20i;15 20i 56]/100).^2*M(9); % Tensor of Inertia in local coordinates for Left Hand
    I(:,:,10)=(L(10)*[29 7 2i;7 15 7i;2i 7i 30]/100).^2*M(10); % Tensor of Inertia in local coordinates for Right Thigh
    I(:,:,11)=(L(11)*[28 4i 2i;4i 10 5;2i 5 28]/100).^2*M(11); % Tensor of Inertia in local coordinates for Right Leg
    I(:,:,12)=(L(12)*[17 13 8i;13 37 0;8i 0 36]/100).^2*M(12); % Tensor of Inertia in local coordinates for Right Foot
    I(:,:,13)=(L(13)*[29 7 2i;7 15 7i;2i 7i 30]/100).^2*M(13); % Tensor of Inertia in local coordinates for Left Thigh
    I(:,:,14)=(L(14)*[28 4i 2i;4i 10 5;2i 5 28]/100).^2*M(14); % Tensor of Inertia in local coordinates for Left Leg
    I(:,:,15)=(L(15)*[17 13 8i;13 37 0;8i 0 36]/100).^2*M(15); % Tensor of Inertia in local coordinates for Left Foot
```

A.3 Center of Mass Acceleration

A.3.1 Center of Mass Acceleration for MVNX

```
% This funtion finds ceter of mass acceleration, given the joints
% accelration, angular velocity and angular acceleration, along with center of mass location in local
% CS and the rotational matrix from local CS to global CS

% The funtion output is the cetner of mass accelration for each segment of 15 segment body model, The segments are:
% 1: Pelvic, 2: Torso, 3: Head&Neck, 4: Right upper arm, 5: Right Forearm, 6: Right hand
% 7: Left upper arm, 8: Left Forearm, 9: Left hand, 10: Right Thigh, 11: Right Shank,
% 12: Right Foot, 13: Left Thigh, 14: Left Shank and 15: Left Foot

%while the iinput is the joint's acceleration in the same order, where the
%joint is the segment's proximal joint (local CS origin)
% 1: L5S1, 2:L5S1, 3: C7T1, 4: Right shoulder, 5: Right elbow, and so on

function [CM_acceleration] = center_of_mass_acceleration_fromJC(JC_acceleration,omega,alpha,r,R_L_G)

% Inputs:
%joint_acceleration: joint's acceleration of each segment proximal joint. 3xnx15 matrix, 3d vector for each frame for each segment
% omega : angular velocity of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% alpha : angular acceleration of each segment. 3xnx15 matrix, 3d vector for each frame for each segment for each segment
% r: The location of each segment's Center of mass in segment's CS. 3x15 matrix. 3D vector for each segment.
% R_L_G: Rotational matrix from local segment CS to global CS (referecne frame)
%       (3x3x15xframes), 3X3 matrix for each segment for each frame (15 segments)

```

```
% Output: CM_accelration is the acceleration of each segment's ceter of
% mass. 3xnx15 matrix, 3d vector for each frame for each segment

%%

CM_acceleration = zeros(size(JC_acceleration,1),size(JC_acceleration,2),size(JC_acceleration,3));

for frame = 1: size(JC_acceleration,2)-1

    for segment = 1 : 15

        r_cm = R_L_G(:,:,segment,frame)*r(:,segment); %center of mass location away from segment's origin (proximal joint) in global CS

        % center of mass acceleration
        CM_acceleration(:,frame,segment) = JC_acceleration(:,frame,segment) + cross(alpha(:,frame,segment),r_cm) + cross(omega(:,frame,segment),cross(omega(:,frame,segment),r_cm));

    end
end
```

A.3.2 Center of Mass Acceleration for Perception Neuron

```
% This funtion finds ceter of mass acceleration, given the sensors
% accelration, angular velocity, angular acceleration and location, along with center of mass location in local
% CS and the rotational matrix from local CS to global CS

% The funtion output is the cetner of mass accelration for each segment of 15 segment body model, The segments are:
% 1: Pelvic, 2: Torso, 3: Head&Neck, 4: Right upper arm, 5: Right Forearm, 6: Right hand
% 7: Left upper arm, 8: Left Forearm, 9: Left hand, 10: Right Thigh, 11: Right Shank,
% 12: Right Foot, 13: Left Thigh, 14: Left Shank and 15: Left Foot


function [CM_acceleration] = center_of_mass_acceleration(sensor_acceleration,sensor_location,omega,alpha,r,zeta,R_L_G)

% Inputs:
%sensor_accelration: sensor's acceleration of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% omega : angular velocity of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% alpha : angular acceleration of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% sensor_location : coordinates of each sensor's location with resepct to pelvis origion in bvh CS, 3xnx15 matrix, 3d vector for each frame for each segment
% r: The location of each segment's Center of mass in segment's CS. 3x15 matrix. 3D vector for each segment.
% zeta: The location of each segment's CS origin in preceeding segment's CS. 3x15 matrix. 3D vector for each segment.
%          for the pelvic, zeta is variable for each frame and given in zeta_pelvic
% zeta_pelvis : Pelvic CS origin for each frame (framesx3) matrix, 3D vector for each frame
% R_L_G: Rotational matrix from local segment CS to global CS (referecne frame)
%       (3x3x15xframes), 3X3 matrix for each segment for each frame (15 segments)

% Output: CM_accelration is the acceleration of each segment's ceter of
% mass. 3xnx15 matrix, 3d vector for each frame for each segment


%% Lower body array (preceeding segment)

L_B_A=[0 1 2 2 4 5 2 7 8 1 10 11 1 13 14]; % Lower body array (preceeding segment)

%%

CM_acceleration = zeros(size(sensor_acceleration,1),size(sensor_acceleration,2),size(sensor_acceleration,3));

for frame = 1: size(sensor_acceleration,2)-1

    for segment = 1 : 15

        %% find the displacement vector from center of mass location to sensor's location for each segment for each frame

        % find the vector from segment origin to pelvic origin expressed in the reference frame
```

```
        segment_origin=zeros(3,1);
        proximal_segment=L_B_A(segment);
        current_segment=segment;
        while proximal_segment >= 1
            R_PS_G = R_L_G(:,:,proximal_segment,frame); % rotational matrix from proximal segment CS to Global CS (reference frame)
            zeta_current=zeta(:,current_segment); %the origin locaion of current segment away from proximal segment in proximal segment CS
            segment_origin=segment_origin+R_PS_G*zeta_current; % the segment origin location away from pelvic origin is the summation of all vectors from pelvic origin to segment origin in reference frame
            current_segment=proximal_segment;proximal_segment=L_B_A(proximal_segment);
        end


        % segment center of mass location
        segment_CM = segment_origin + R_L_G(:,:,segment,frame)*r(:,segment);

        % displacement vector from center of mass location to sensor's location
        r_sensor = sensor_location(:,frame,segment) - segment_CM;

        %% find center of mass acceleration

        CM_acceleration(:,frame,segment) = sensor_acceleration(:,frame,segment) + cross(alpha(:,frame,segment),r_sensor) + cross(omega(:,frame,segment),cross(omega(:,frame,segment),r_sensor));

    end
end
```

A.4 Contact Detection Algorithm

```

%% This function Predict the foot contact with the gorund for both feet using Contact detection algorithem described in the thesis

function [contact] = Contact_Detection(r_toe_vel,l_toe_vel,r_heel_vel,l_heel_vel,TH)
%%

% Inputs:
% r_toe_vel : Right Toe velocity norm; nx1 vecotr, velocity norm at each frame
% l_toe_vel : Left Toe velocity norm; nx1 vecotr, velocity norm at each frame
% r_heel_vel : Right Heel velocity norm; nx1 vecotr, velocity norm at each frame
% l_heel_vel : Left Heel velocity norm; nx1 vecotr, velocity norm at each frame
% TH : Velocity threshold

% outputs:
%contact : nx2 matrix, predicted contact (1:contact, 0: no contact) for each frame for each foot

%% Contact Detection

% Derivative of heel velocity
r_heel_acc=diff(r_heel_vel);
l_heel_acc=diff(l_heel_vel);

% right and left contact
contact_r=zeros(size(r_toe_vel,1),1);
contact_l=zeros(size(r_toe_vel,1),1);


%first frame contact detection
if r_toe_vel(1) <= TH %if toe velocity is smaller than threshold, there is contact
    contact_r(1)=1;
else
    contact_r(1)=0;
end

if l_toe_vel(1) <= TH %if toe velocity is smaller than threshold, there is contact
    contact_l(1)=1;
else
    contact_l(1)=0;
end


for n=2:size(r_heel_acc,1)
```

```
    if contact_r(n-1)==1 % if the foot is already in contact
        if r_toe_vel(n)>TH && r_heel_acc(n)<=0 % it loses contact if the toe velocity is larger than threshold, and heel velocity reaches maximum (derivative becomes negative)
            contact_r(n)=0;
        else %otherwise, it stays in contact
            contact_r(n)=1;
        end
    else % if the foot is not in contact
        if r_toe_vel(n)>TH
            contact_r(n)=0;
        else % it becomes at contact if the toe velocity is smaller than the threshold
           contact_r(n)=1;
        end
    end

    if contact_l(n-1)==1% if the foot is already in contact
        if l_toe_vel(n)>TH && l_heel_acc(n)<=0% it loses contact if the toe velocity is larger than threshold, and heel velocity reaches maximum (derivative becomes negative)
            contact_l(n)=0;
        else %otherwise, it stays in contact
            contact_l(n)=1;
        end
    else % if the foot is not in contact
        if l_toe_vel(n)>TH
            contact_l(n)=0;
        else % it becomes at contact if the toe velocity is smaller than the threshold
           contact_l(n)=1;
        end
    end
    if contact_r(n)==0 && contact_l(n)==0 %if niether foot is detected to have contact, assume contact in right foot to avoid unbalanced force when applying ID
        contact_r(n)=1;
    end
end

%last frame contact
contact_r(end)=contact_r(end-1);
contact_l(end)=contact_l(end-1);

% Contact matrix contains the right and left contact
contact=[contact_r contact_l];
```

A.5 Contact Detection Algorithm

A.5.1 GRFs Optimization Cost Function

```
% This funciton calculate the net joint moment and force of the joints in the lower limbs closed loop
%(RFoot/Ground-R_Ankle-RKnee-RHip-LHip-LKnee-LAnkle-LFoot/Ground-RFoot/Ground)

% This funciton is used as a cost function, for an optimization method to
% find Ground Reaciton Froces and Ground Reaction Moments on right and left feet

%The joints that are considered in this summation of moments are:
% right hip, left hip, right knee, left knee, right ankle and left ankle

% the cost funciton is set to summation of the squared net moment of these joints

function sum_of_moments = cost_function(r_l_GRF_M, variables_struc)

%% extrct variables from variables_struc
frame=variables_struc.frame;

L_B_A=variables_struc.L_B_A;
r=variables_struc.r;zeta=variables_struc.zeta;I=variables_struc.I;M=variables_struc.M;

R_L_G=variables_struc.R_L_G;A=variables_struc.A;omega=variables_struc.omega;alphaa=variables_struc.alphaa;

External_Forces=variables_struc.External_Forces;External_Moments=variables_struc.External_Moments;
External_point_of_action=variables_struc.External_point_of_action;

%% add GRF and GRM to external forces and moments
```

```
External_Forces(1:3,frame,12)=External_Forces(1:3,frame,12)+r_l_GRF_M(1:3);  %add GRF to right foot external force
External_Moments(1:3,frame,12)=External_Moments(1:3,frame,12)+r_l_GRF_M(4:6);  %add GRM to right foot external Moment

External_Forces(1:3,frame,15)=External_Forces(1:3,frame,15)+r_l_GRF_M(7:9);  %add GRF to Left foot external force
External_Moments(1:3,frame,15)=External_Moments(1:3,frame,15)+r_l_GRF_M(10:12);  %add GRM to Left foot external Moment



%% find joint moment and joint force for lower limbs' joints
J_F=zeros(3,14);J_M=zeros(3,14); %joint force and joint moment for this frame only, for joints from 9:14 only (segment 10:15)

    for segment = 15 : -1 : 10 % go through lower limbs segments from distal to proximal

        R_S_G = R_L_G(:,:,segment,frame); % rotational matrix from segment CS to Global CS (reference frame) for this segment and this frame
        I_segment = R_S_G'*I(3,3,segment)*R_S_G; % segment intertia tensor in the refernce frame for this frame
        r_CM = R_S_G*r(1:3,segment); % center of mass location away from segment CS origin, but expressed in Global CS
        r_external = R_S_G*External_point_of_action(1:3,frame,segment); % external force location away from segment CS origin, but expressed in Global CS

        F_inertia = - M(segment)*A(1:3,frame,segment); %intertia force including the wight, acceleration should include g, or else add (+M(segment)*(0 g 0)), or depends on your CS
        M_intertia = - ( I_segment*alphaa(1:3,frame,segment) + cross( omega(1:3,frame,segment) , I_segment*omega(1:3,frame,segment))); % intertia moment

        F_external = External_Forces(1:3,frame,segment); % net external forces on the segment at this frame
        M_external = External_Moments(1:3,frame,segment) + cross((r_external-r_CM),F_external); % net external moments on the segment center of mass at this frame

        % find the distal segments attached to this segment to consider the reaction forces and moments
        % from the joints of these segments with the current segment in this segment's equations of motion

        % note that most segments have only one distal segments attached to them, but the most distal segments (feet and hands)
        % don't have any, while torso have 3 (2 shoulders and neck)

        distal_segments = find (L_B_A==segment);
        F_distal_joints = zeros(3,1);  M_distal_joints = zeros(3,1);
        distal=1; %starting from the first distal joint

            while distal <= size(distal_segments,2) %going through all distal joints from 1 to the number of distal joints to that segment

                r_distal_joint = R_S_G*zeta(:,distal_segments(distal)); % joint force location away from segment CS origin, but expressed in Global CS
                F_distal_joints = F_distal_joints - J_F(1:3,distal_segments(distal)-1); %summation of all distal joints forces on this segment.
                M_distal_joints = M_distal_joints - J_M(1:3,distal_segments(distal)-1) + cross((r_distal_joint-r_CM),-J_F(1:3,distal_segments(distal)-1));
               %summation of all distal joints moments produced around this segment's cetner of mass.
                distal=distal+1;
            end

        J_F(1:3,segment-1) = - F_inertia - F_external - F_distal_joints; % net joint force of the joint between current segment and preceeding segment
        J_M(1:3,segment-1) = - M_intertia - M_external - M_distal_joints  - cross( -r_CM , J_F(1:3,segment-1) ) ; % net joint moment of the joint between current segment and preceeding segment

    end

    %% Calculate the anatomical component of knee and ankel moments (flextion/extension, lateral bending, and Flexion
%         l_knee_moment(1:3)=R_L_G(:,:,13,frame)*J_M(:,13);
          l_ankle_moment(1:3)=R_L_G(:,:,14,frame)*J_M(:,14);
%         r_knee_moment(1:3)=R_L_G(:,:,10,frame)*J_M(:,10);
          r_ankle_moment(1:3)=R_L_G(:,:,11,frame)*J_M(:,11);
```

```
        %% Calculate the value of the cost function


    joints_moments_squared=(sum((J_M(:,:).^2),1)); % net joints moments squared

    sum_of_moments=sum(joints_moments_squared(9:14))+l_ankle_moment(1).^4+r_ankle_moment(1).^4+l_ankle_moment(2).^4+r_ankle_moment(2).^4; %summation of joints moment for closed loop joints (including GRM around ankle)
```

A.5.2 Right and Left GRFs Estimation

```
%This functions finds the ground reaction forces and moments

% External forces are composite of  to Ground Reaction forces, which are solved
% for, and other external forces that is inputed to this solver

% The problem can be in two forms, either determinate, with only one foot touching the ground or no foot touchs the ground,
% or indeterminate, where both feet touch the ground making a closed loop and indeterminate problem.

% The total GRF is solved for by solving the equations of motion of the whole body. That is total GRF equals the summation of m*a for all
% segments. while the Ground reaction moment is solved for around the pelvic CS origin, by solving the moment equations of motion of the whole
% body. That is, the moment euals sumamtion of (I*alpha+ omega x (I*omega)+ r_cm x (m*a) for all segments.
% Total GRF and GRM is equal to the summation of right and left GRF and GRM
% if only one foot touches the ground, the problem is solvable. However, if
% both feet are in contact with the ground, the problem of finding each foot's ground reaction forces and moments becomes indertminate

% This function solves the indeterminate problem by an optimization method. That is,If the problem is indeterminate,
% it is solved for by an optimization funciton that minimze the summation of joint moment in the loop as a cost funtion
% with maintaining the total GRF and moment to balance the equations of motion and finds ground reaction force and moment on each foot.
% The indeterminate problem is solved by follwing the method proposed in (VAUGHAN1982)
% with some modifications describen on the thesis .

% VAUGHAN, Christopher L.; HAY, James G.; ANDREWS, James G. Closed loop problems in biomechanics.
% Part II An optimization approach. Journal of Biomechanics, 1982, 15.3: 201-210.?

%The funtion inputs are the kinematics of the motion, the intertia properties of the subject, all external forces
%during the motion, except for ground reaction force which is solved for in this code, And a variable that specify either
%both feet touch the gorund or not at each frame. that is, is the problem at that frame determinate or indeterminate.


function [GRF_r,GRF_l,GRM_r,GRM_l,Net_GRF,Net_GRM] = GRF_Optimization(r,zeta,I,M,R_L_G,A,omega,alphaa,contact,External_Forces,External_Moments,External_point_of_action)


% Inputs:

% r: The location of each segment's Center of mass in segment's CS. 3x15 matrix. 3D vector for each segment.
% zeta: The location of each segment's CS origin in preceeding segment's CS. 3x15 matrix. 3D vector for each segment.
% M: The mass of each segment, vectory of 15 elements.
% I: The inertia tensor of the segment about it's center of mass, in segment's CS. 3x3x15 matrix. 3x3 dyadic for each segment.
% R_L_G: Rotational matrix from local segment CS to global CS (referecne frame)
%       (3x3x15xframes), 3X3 matrix for each segment for each frame (15 segments)
% A : center of mass acceleration of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% omega : angular velocity of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% alpha : angular acceleration of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% contact: a value that specify if each feet in contact with the floor (1) or not (0)
%       (framsx2) a vector of size frames, one value for each frame for each foot
% External_Forces: the total external force on each segment for each frame in the reference frame (3xnx15) 3D vector for each frame for each segment
```

```
% External_Moments: the total external moment around the point of action on each segment for each frame in the reference frame (3xnx15) 3D vector for each frame for each segment
% External_point_of_action: the location of point of action of external force on each segment for each frame in segment's CS
%       (3xnx15) 3D vector for each frame for each segment

% Outputs:

% GRF_r : Ground Reaction force vector on the right foot. 3xn matrix, 3d vector for each frame
% GRF_l : Ground Reaction force vector on the left foot. 3xn matrix, 3d vector for each frame
% GRM_r : Ground Reaction moment vector on the right foot. This is the moment around right foot origin ( right Ankle Joint Center). 3xn matrix, 3d vector for each frame
% GRM_l : Ground Reaction moment vector on the left foot. This is the moment around left foot origin ( left Ankle Joint Center). 3xn matrix, 3d vector for each frame

%% Ready? Begin

%% Lower body array (preceeding segment)

L_B_A=[0 1 2 2 4 5 2 7 8 1 10 11 1 13 14]; % Lower body array (preceeding segment)

%% find GRF and GRM


Net_GRF=zeros(3,size(A,2)-1);
Net_GRM=zeros(3,size(A,2)-1);

GRF_r=zeros(3,size(A,2)-1);GRF_l=zeros(3,size(A,2)-1);GRM_r=zeros(3,size(A,2)-1);GRM_l=zeros(3,size(A,2)-1);
 R_Foot_origin=zeros(3,size(A,2)-1);L_Foot_origin=zeros(3,size(A,2)-1);

% save several variables to one structure variable to use in cost function

    variables_struc.r=r;variables_struc.zeta=zeta;variables_struc.R_L_G=R_L_G;variables_struc.omega=omega;
    variables_struc.alphaa=alphaa;variables_struc.External_Forces=External_Forces;
    variables_struc.External_Moments=External_Moments;variables_struc.External_point_of_action=External_point_of_action;
    variables_struc.I=I;variables_struc.M=M;variables_struc.A=A;
    variables_struc.L_B_A=L_B_A;


for frame = 1: size(A,2)-1 % go through all frames from first to last
    %% Net Ground Reaction Force and moment

    Net_GRF(:,frame)=0; % Summation of forces equals Total Ground Reaction Froce
    Net_GRM(:,frame)=0; % Summation of moments around pelvic origin equals Total Ground Reaction Moment around pelvic origin

    right_lower_limb_net_force=zeros(3,1);
    right_lower_limb_net_moment=zeros(3,1);
    left_lower_limb_net_force=zeros(3,1);
    left_lower_limb_net_moment=zeros(3,1);

    for segment = 15 : -1 : 1 % go through all segments from distal to proximal

        R_S_G = R_L_G(:,:,segment,frame); % rotational matrix from segment CS to Global CS (reference frame) for this segment and this frame
        I_segment = R_S_G'*I(3,3,segment)*R_S_G; % segment intertia tensor in the refernce frame for this frame
        r_CM = R_S_G*r(1:3,segment); % center of mass location away from segment CS origin, but expressed in Global CS
        r_external = R_S_G*External_point_of_action(1:3,frame,segment); % external force location away from segment CS origin, but expressed in Global CS

        F_inertia = - M(segment)*A(1:3,frame,segment); %intertia force including the wight, acceleration should include g, or else add (+M(segment)*[0; g; 0]), or depends on your CS
        M_intertia = - ( I_segment*alphaa(1:3,frame,segment) + cross( omega(1:3,frame,segment) , I_segment*omega(1:3,frame,segment))); % intertia moment

        F_external = External_Forces(1:3,frame,segment); % net external forces on the segment at this frame
        M_external = External_Moments(1:3,frame,segment); % direct external moment applied on the segment

       %find the vector from segment origin to pelvic origin expressed in the reference frame
        segment_origin=zeros(3,1);
```

```
        proximal_segment=L_B_A(segment);
        current_segment=segment;
        while proximal_segment >= 1
            R_PS_G = R_L_G(:,:,proximal_segment,frame); % rotational matrix from proximal segment CS to Global CS (reference frame)
            zeta_current=zeta(:,current_segment); %the origin locaion of current segment away from proximal segment in proximal segment CS
            segment_origin=segment_origin+R_PS_G*zeta_current; % the segment origin location away from pelvic origin is the summation of all vectors from pelvic origin to segment origin in reference frame
            current_segment=proximal_segment;proximal_segment=L_B_A(proximal_segment);
        end

        Net_GRF(:,frame)=Net_GRF(:,frame)-F_inertia-F_external; %Ground reaction froce equals the summation of all intertia forces and external forces on all segments
        Net_GRM(:,frame)=Net_GRM(:,frame)-M_intertia-cross((r_CM+segment_origin),F_inertia)-M_external-cross((r_external+segment_origin),F_external);
        % Ground reaction moments around pelvic CS origin equals the summation of applied external moments, intertia moments,
        % and cross product of inertia forces by its location away from pelvic CS and external forces by its location away from pelvic CS

        if segment == 12
            R_Foot_origin(:,frame) = segment_origin; % save right foot origin in the memory
        elseif segment == 15
            L_Foot_origin(:,frame) = segment_origin; % save left foot origin in the memory
        end

        if segment >= 10 && segment <= 12
            right_lower_limb_net_force=right_lower_limb_net_force-F_inertia-F_external; % right_lower_limb_net_force equals the summation of all intertia forces and external forces on segments 10 11 12
            right_lower_limb_net_moment=right_lower_limb_net_moment-M_intertia-cross((r_CM+segment_origin),F_inertia)-M_external-cross((r_external+segment_origin),F_external);
        elseif segment > 12
            left_lower_limb_net_force=left_lower_limb_net_force-F_inertia-F_external;  % left_lower_limb_net_force equals the summation of all intertia forces and external forces on segments 13 14 15
            left_lower_limb_net_moment=left_lower_limb_net_moment-M_intertia-cross((r_CM+segment_origin),F_inertia)-M_external-cross((r_external+segment_origin),F_external);
        end

    end

    %% right and left GRF and GRM

    %check which feet are in contact with the ground
    % if neither is, then GRF_r GRF_l, GRM_r and GRM_l are zeros, regardless of the ballancing net GRF
    %       In theory the net GRF and GRM should be zero if neither foot is in contact.
    % if only one is, then net GRF is equal to that foot GRF, and net GRM is euqal to that foot GRM but shifted
    %       from pelvic origin to foot origin
    % if both are, solve by optimization similar to that described in VAUGHAN1982

    if contact(frame,1)==0 && contact(frame,2)==0

        GRF_r(:,frame)=0;GRF_l(:,frame)=0;GRM_r(:,frame)=0;GRM_l(:,frame)=0;

    elseif contact(frame,1)==1 && contact(frame,2)==0

        GRF_r(:,frame)=Net_GRF(:,frame);
        GRM_r(:,frame)=Net_GRM(:,frame)+cross(-R_Foot_origin(:,frame),Net_GRF(:,frame));

    elseif contact(frame,1)==0 && contact(frame,2)==1

        GRF_l(:,frame)=Net_GRF(:,frame);
        GRM_l(:,frame)=Net_GRM(:,frame)+cross(-L_Foot_origin(:,frame),Net_GRF(:,frame));

    else
        %% optimization problem


%           solve for the vector of 12 unknowns:
%           GRF_right_x
%           GRF_right_y
%           GRF_right_z
%           GRM_right_x
```

```
%           GRM_right_y
%           GRM_right_z
%           GRF_left_x
%           GRF_left_y
%           GRF_left_z
%           GRM_left_x
%           GRM_left_y
%           GRM_left_z



        % first iteration
        if frame ~= 1
            GRFM_first_itiration=[GRF_r(:,frame-1);GRM_r(:,frame-1);Net_GRF(:,frame-1)-GRF_r(:,frame-1);GRM_l(:,frame-1)];
        else
            right_GRM_firstiteration=right_lower_limb_net_moment+(Net_GRM(:,frame)-left_lower_limb_net_moment-right_lower_limb_net_moment)./2+cross(-R_Foot_origin(:,frame),Net_GRF(:,frame)/2);
            left_GRM_firstiteration=left_lower_limb_net_moment+(Net_GRM(:,frame)-left_lower_limb_net_moment-right_lower_limb_net_moment)./2+cross(-L_Foot_origin(:,frame),Net_GRF(:,frame)/2);

            GRFM_first_itiration=[Net_GRF(:,frame)./2;right_GRM_firstiteration;Net_GRF(:,frame)./2;left_GRM_firstiteration];
        end

        %constraints

        % Equality Constraints
        Aeq=zeros(6,12);beq=zeros(6,1);

        %Net GRF vector is equal to the summation of right and left GRFs vecotrs
        %expand to 3 scalar equations in 3 dimensions

        Aeq(1,1)=1;Aeq(1,7)=1;beq(1)=Net_GRF(1,frame); % x direction
        Aeq(2,2)=1;Aeq(2,8)=1;beq(2)=Net_GRF(2,frame); % y direction
        Aeq(3,3)=1;Aeq(3,9)=1;beq(3)=Net_GRF(3,frame); % z direction

        %Net GRM vecotr is the net moment around the pelvis origin but right and
        %left GRM vectors are expressed as the moment around the ankles. Therefore,
        %net GRM vector is equal to the summation of right and left GRM vectors plus the
        %summation of the cross products between ankle position vector and
        %GRF vectors for both feet
        %expand to 3 scalar equations in 3 dimensions

        % x direction
        Aeq(4,2)=-R_Foot_origin(3,frame);Aeq(4,3)=R_Foot_origin(2,frame);
        Aeq(4,8)=-L_Foot_origin(3,frame);Aeq(4,9)=L_Foot_origin(2,frame);
        Aeq(4,4)=1;Aeq(4,10)=1;beq(4)=Net_GRM(1,frame);
        % y direction
        Aeq(5,1)=R_Foot_origin(3,frame);Aeq(5,3)=-R_Foot_origin(1,frame);
        Aeq(5,7)=L_Foot_origin(3,frame);Aeq(5,9)=-L_Foot_origin(1,frame);
        Aeq(5,5)=1;Aeq(5,11)=1;beq(5)=Net_GRM(2,frame);
        % z direction
        Aeq(6,1)=-R_Foot_origin(2,frame);Aeq(6,2)=R_Foot_origin(1,frame);
        Aeq(6,7)=-L_Foot_origin(2,frame);Aeq(6,8)=L_Foot_origin(1,frame);
        Aeq(6,6)=1;Aeq(6,12)=1;beq(6)=Net_GRM(3,frame);

        %upper and lower bound

        % Forces in y-direction are bigger than zero for both feet
        %right foot y-direction moment (constrainted between -25 and 25 N.m)
        %left foot y-direction moment (constrainted between -25 and 25 N.m)
        %other forces are unbounded (10000 for vertical Force, and 1000 for all other componentes)

        lb = [ -1000 ; 0     ; -1000 ; -1000 ; -25 ; -1000 ; -1000 ; 0     ; -1000 ; -1000 ; -25 ; -1000 ];
        ub = [  1000 ; 10000 ;  1000 ;  1000 ;  25 ;  1000 ;  1000 ; 10000 ;  1000 ;  1000 ;  25 ;  1000 ];

        Aineq=[];
        bineq=[];


        % add "frame" variable to the structure to use in cost function

```

```
        variables_struc.frame=frame;

        % optimization problem, looking for GRFs and moments in right and left feet,
        % with the objective fuction as the summation of moments in the lower limbs
        % That is, the cloosed loop (RFoot/Ground-R_Ankle-RKnee-RHip-LHip-LKnee-LAnkle-LFoot/Ground-RFoot/Ground)

        options = optimset('Display','none','LargeScale','off');

        [optimized_GRF, ~, ~] = fmincon(@(x)cost_function(x,variables_struc), GRFM_first_itiration,Aineq,bineq,Aeq,beq,lb,ub,[],options);

        % save optimization result for this frame to corresponding variables
        GRF_r(:,frame)=optimized_GRF(1:3);
        GRM_r(:,frame)=optimized_GRF(4:6);

        GRF_l(:,frame)=optimized_GRF(7:9);
        GRM_l(:,frame)=optimized_GRF(10:12);

    end

end
```

A.6 Inverse Dynamic Solver

```
%% ( this is the function is the most important part, the dynamics analysis solver)
% This funtion performs the inverse dynamics analysis of 15 segment body model

% The funtion inputs are the kinematics of the motion, the intertia properties of the subject, and all external forces during the motion
%  The user should also input the external forces and moments for each segment along
% with its point of action location in segment's SC.

% The function outputs are net joint forces and net joint moment for each joint for each frame.
%
% The joints are: (in order) (14 joints)
% L5/S1, C7/T1, Right Shoulder, Right Elbow, Right Wrest, Left Shoulder, Left Elbow, Left Wrest,
% Right Hip, Right Knee, Right Ankle, Left Hip, Left Knee, Left Ankle

% The segments are:
% 1: Pelvic, 2: Torso, 3: Head&Neck, 4: Right upper arm, 5: Right Forearm, 6: Right hand
% 7: Left upper arm, 8: Left Forearm, 9: Left hand, 10: Right Thigh, 11: Right Shank,
% 12: Right Foot, 13: Left Thigh, 14: Left Shank and 15: Left Foot

% Each joint i corrsponding to the origin of segment i+1, as the only
% proximal joint to that segment. Thus, for each joint i, the net force and
% moment can be found be solvign the equations of motion of segment i.
% going from distal to proximal. from segment 15 to segment 2.

% The equations of motion of segment 1 should give be redundant and ballanced,
% given that the Ground Reaction forces and moments are
% solved for or measured correctely and assuming that this model is valid

function [J_F,J_M] = Dynamics_solver(r,zeta,I,M,R_L_G,A,omega,alphaa,External_Forces,External_Moments,External_point_of_action)

% Inputs:
% r: The location of each segment's Center of mass in segment's CS. 3x15 matrix. 3D vector for each segment.
% zeta: The location of each segment's CS origin in preceeding segment's CS. 3x15 matrix. 3D vector for each segment.
% M: The mass of each segment, vectory of 15 elements.
% I: The inertia tensor of the segment about it's center of mass, in segment's CS. 3x3x15 matrix. 3x3 dyadic for each segment.
% R_L_G: Rotational matrix from local segment CS to global CS (referecne frame)
%       (3x3x15xframes), 3X3 matrix for each segment for each frame (15 segments)
% A : center of mass acceleration of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% omega : angular velocity of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% alpha : angular acceleration of each segment. 3xnx15 matrix, 3d vector for each frame for each segment
% External_Forces: the force on each segment for each frame in the reference frame (3xnx15) 3D vector for each frame for each segment
% External_Moments: the moment on each segment for each frame in the reference frame (3xnx15) 3D vector for each frame for each segment
```

```matlab
% External_point_of_action: the location of point of action of external force on each segment for each frame in segment's CS
%       (3xnx15) 3D vector for each frame for each segment

% Outputs:
% J_F : Joints net forces for each joint for each frame (3x14xframes) 3D vector for each joint for each frame
% J_M : Joints net moments for each joint for each frame (3x14xframes) 3D vector for each joint for each frame
%           the joints order specified above

%% Ready? Begin

%% Lower body array (preceeding segment)

L_B_A=[0 1 2 2 4 5 2 7 8 1 10 11 1 13 14]; % Lower body array (preceeding segment)

    %% Joints net forces and moments

for frame = 1: size(A,2)-1 % go through all frames from first to last

    for segment = 15 : -1 : 1 % go through all segments from distal to proximal

        R_S_G = R_L_G(:,:,segment,frame); % rotational matrix from segment CS to Global CS (reference frame) for this segment and this frame
        I_segment = R_S_G'*I(3,3,segment)*R_S_G; % sVCXegment intertia tensor in the refernce frame for this frame
        r_CM = R_S_G*r(1:3,segment); % center of mass location away from segment CS origin, but expressed in Global CS
        r_external = R_S_G*External_point_of_action(1:3,frame,segment); % external force location away from segment CS origin, but expressed in Global CS

        F_inertia = - M(segment)*A(1:3,frame,segment); %intertia force including the wight, acceleration should include g, or else add (+M(segment)*(0 g 0)), or depends on your CS
        M_intertia = - ( I_segment*alphaa(1:3,frame,segment) + cross( omega(1:3,frame,segment) , I_segment*omega(1:3,frame,segment))); % intertia moment

        F_external = External_Forces(1:3,frame,segment); % net external forces on the segment at this frame
        M_external = External_Moments(1:3,frame,segment) + cross((r_external-r_CM),F_external); % net external moments on the segment center of mass at this frame

        % find the distal segments attached to this segment to consider the reaction forces and moments
        % from the joints of these segments with the current segment in this segment's equations of motion

        % note that most segments have only one distal segments attached to them, but the most distal segments (feet and hands)
        % don't have any, while torso have 3 (2 shoulders and the neck)

        distal_segments = find (L_B_A==segment);
        F_distal_joints = zeros(3,1); M_distal_joints = zeros(3,1);
        distal=1; %starting from the first distal joint

            while distal <= size(distal_segments,2) %going through all distal joints from 1 to the number of distal joints to that segment
                r_distal_joint = R_S_G*zeta(:,distal_segments(distal)); % joint force location away from segment CS origin, but expressed in Global CS

                F_distal_joints = F_distal_joints - J_F(1:3,distal_segments(distal)-1,frame); %summation of all distal joints forces on this segment.
                M_distal_joints = M_distal_joints - J_M(1:3,distal_segments(distal)-1,frame) + cross((r_distal_joint-r_CM),-J_F(1:3,distal_segments(distal)-1,frame));
               %summation of all distal joints moments produced around this segment's cetner of mass.
                distal=distal+1;
            end

            %for the first segment, the equations of motion are redundant and the caculated force and momnet are the unballanced force and moment,
            %due to wrong entry in external forces
            %(e.g. gorund reaction forces don't ballance the net interitial and external forces)

        if segment~=1
            J_F(1:3,segment-1,frame) = - F_inertia - F_external - F_distal_joints; % net joint force of the joint between current segment and preceeding segment
            J_M(1:3,segment-1,frame) = - M_intertia - M_external - M_distal_joints  - cross( -r_CM , J_F(1:3,segment-1,frame) ) ; % net joint moment of the joint between current segment and
```

```
                preceeding segment
        else
            J_F(1:3,15,frame) = - F_inertia - F_external - F_distal_joints; % net joint force of the joint between current segment and preceeding segment
            J_M(1:3,15,frame) = - M_intertia - M_external - M_distal_joints  - cross( -r_CM , J_F(1:3,15,frame) ) ; % net joint moment of the joint between current segment and preceeding segment
        end
    end
end
```

A.7 Lower-Back Disk Contact Forces Optimization

```
function [L4L5Compression,L4L5LateralShear,L4L5AnteriorShear,muscles_forces] = L4L5_Linear_Optimization_Bean_Schultz(L5S1_Moment,L5S1_Force,R_Pelvic_Global,Theta_H)

%% find moment components
L4L5_Moment_local=zeros(3,length(L5S1_Moment));
L5S1_Force_local=zeros(3,length(L5S1_Moment));

for i=1:length(L5S1_Moment)
    L4L5_Moment_local(:,i)=R_Pelvic_Global(:,:,i)'*L5S1_Moment(:,i);
    %find L4L5 moment in local frame, assume L4L5 moment the same as L5S1 since the model assumes the torso as one rigid body
    L5S1_Force_local(:,i)=R_Pelvic_Global(:,:,i)'*L5S1_Force(:,i);
end

L4L5_Coronal_Moment=L4L5_Moment_local(1,:);
L4L5_Torque=L4L5_Moment_local(2,:);
L4L5_Saggital_Moment=L4L5_Moment_local(3,:);

L4L5_Anterior_force=L5S1_Force_local(1,:);
L4L5_Normal_force=L5S1_Force_local(2,:);
L4L5_Lateral_force=L5S1_Force_local(3,:);

%% muscle properties
gender=1; %male

if gender==1 %if male
    %Physiological cross-sectional area in cm^2
    A_ES=31; % Erector Spinae
    A_LD=3; % Latissimus Dorsi
    A_RA=13; % Rectus Abdominus
    A_IO=5; % Internal Oblique
    A_EO=5; % External Oblique

    %Coronal Moment Arm in m
    r_Coronal_ES=5.4*10^-2; % Erector Spinae
    r_Coronal_LD=6.3*10^-2; % Latissimus Dorsi
    r_Coronal_RA=3.6*10^-2; % Rectus Abdominis
    r_Coronal_IO=13.5*10^-2; % Internal Oblique
    r_Coronal_EO=13.5*10^-2; % External Oblique

    %Saggital Moment Arm in m
    r_Saggital_ES=4.4*10^-2; % Erector Spinae
    r_Saggital_LD=5.6*10^-2; % Latissimus Dorsi
    r_Saggital_RA=-10.8*10^-2; % Rectus Abdominis
    r_Saggital_IO=-3.8*10^-2; % Internal Oblique
    r_Saggital_EO=-3.8*10^-2; % External Oblique

    %Line of action angle to the desk normal (in degrees)

    Theta_ES=0;% Erector Spinae in the Saggital plane
    Theta_LD=45;% Latissimus Dorsi in the Coronal Plane
    Theta_RA=0;% Rectus Abdominis in the Saggital Plane
    Theta_IO=45;% Internal Oblique in the Saggital Plane
    Theta_EO=-45;% External Oblique in the Saggital Plane
    Theta_AP=0; %Abdominal Pressure in teh Saggital Plane

    % Diaphram area affected by the abdominal presure in m^2

    A_ab=465*10^-4;

else %female
```

```
    %Physiological cross-sectional area in cm^2
    A_ES=31; % Erector Spinae
    A_LD=3; % Latissimus Dorsi
    A_RA=13; % Rectus Abdominus
    A_IO=5; % Internal Oblique
    A_EO=5; % External Oblique

    %Coronal Moment Arm in m
    r_Coronal_ES=5.4*10^-2; % Erector Spinae
    r_Coronal_LD=6.3*10^-2; % Latissimus Dorsi
    r_Coronal_RA=3.6*10^-2; % Rectus Abdominis
    r_Coronal_IO=13.5*10^-2; % Internal Oblique
    r_Coronal_EO=13.5*10^-2; % External Oblique

    %Saggital Moment Arm in m
    r_Saggital_ES=4.4*10^-2; % Erector Spinae
    r_Saggital_LD=5.6*10^-2; % Latissimus Dorsi
    r_Saggital_RA=-10.8*10^-2; % Rectus Abdominis
    r_Saggital_IO=-3.8*10^-2; % Internal Oblique
    r_Saggital_EO=-3.8*10^-2; % External Oblique

    %Line of action angle to the desk normal (in degrees)

    Theta_ES=0;% Erector Spinae in the Saggital plane
    Theta_LD=45;% Latissimus Dorsi in the Coronal Plane
    Theta_RA=0;% Rectus Abdominis in the Saggital Plane
    Theta_IO=45;% Internal Oblique in the Saggital Plane
    Theta_EO=-45;% External Oblique in the Saggital Plane
    Theta_AP=0; %Abdominal Pressure in teh Saggital Plane

    % Diaphram area affected by the abdominal presure in m^2

    A_ab=465*10^-4;
end

%% Abdominal Pressure force and moment arm

% from Morris1961 and Chaffin's book (occupational biomechanics)
P_A_mmHg=zeros(length(L5S1_Moment),1);
P_A=zeros(length(L5S1_Moment),1);
F_AP=zeros(length(L5S1_Moment),1);
r_Saggital_AP=zeros(length(L5S1_Moment),1);

for i=1:length(L5S1_Moment)
    P_A_mmHg(i)=(43-0.36*Theta_H(i))*norm(L4L5_Saggital_Moment(i)^1.8)/10000; %Abdominal Pressure in mmHg
    P_A(i)=P_A_mmHg(i)*133.322368; %Abdominal Pressure in Pa
    F_AP(i)=P_A(i)*A_ab; %Abdominal Pressure force in N
    r_Saggital_AP(i)=(7+8*sind(Theta_H(i)))*10^-2; %Abdominal Pressure's force moment arm in the saggital plane
end

%% First linear programming, the objective function is the maximmum muscle intensity

L4L5Compression=zeros(1,length(L5S1_Moment));
L4L5AnteriorShear=zeros(1,length(L5S1_Moment));
L4L5LateralShear=zeros(1,length(L5S1_Moment));
I=zeros(1,length(L5S1_Moment));
muscles_forces=zeros(13,length(L5S1_Moment));

options = optimoptions('linprog','Display','off');

% optimization by linear programming at each frame,
% optimizing for the vector contains the 10 muscles forces, the desk compression, lateral shear,
% Anterior Shearm and the maximum muscle intinsity I (14 variables to optimize)

% The vecor to optimize is
%[F_ES_r; F_ES_l;F_LD_r;F_LD_l;F_RA_r;F_RA_l;F_IO_r;F_IO_l;F_EO_r;F_EO_l;C;S_l;S_a;I_max];

% forces of (Erector Spinae right and left, Latissimus Dorsi right and
% left, Rectus Abdominus right and left, Internal and External Oblique,
% right and left), desk compression, desk lateral shear, desk Anterior
% shear, and teh maximum muscle intinsity

% this optimization doesn't have a unique solution, refer to 'Bean1988'
% paper that this optimization is based on

```

```
for i=1:length(L5S1_Moment)
    %Equality constraints ,sum of forces and moments contibutions of all muscles equal net
    %joint force and moment (equilibrium) (see Schultz1981 for original, this is a generalized form)

    Aeq=[0 0 -sind(Theta_LD) sind(Theta_LD) 0 0 0 0 0 0 0 1 0 0; ...
         sind(Theta_ES) sind(Theta_ES) 0 0 sind(Theta_RA) sind(Theta_RA) sind(Theta_IO) sind(Theta_IO) sind(Theta_EO) sind(Theta_EO) 0 0 1 0; ...
        -cosd(Theta_ES) -cosd(Theta_ES) -cosd(Theta_LD) -cosd(Theta_LD) -cosd(Theta_RA) -cosd(Theta_RA) -cosd(Theta_IO) -cosd(Theta_IO) -cosd(Theta_EO) -cosd(Theta_EO) 1 0 0 0; ...
         r_Saggital_ES*cosd(Theta_ES) r_Saggital_ES*cosd(Theta_ES) r_Saggital_LD*cosd(Theta_LD) r_Saggital_LD*cosd(Theta_LD) r_Saggital_RA*cosd(Theta_RA) r_Saggital_RA*cosd(Theta_RA) r_Saggital_IO*cosd(Theta_IO) r_Saggital_IO*cosd(Theta_IO) r_Saggital_EO*cosd(Theta_EO) r_Saggital_EO*cosd(Theta_EO) 0 0 0 0; ...
         r_Coronal_ES*cosd(Theta_ES) -r_Coronal_ES*cosd(Theta_ES) r_Coronal_LD*cosd(Theta_LD) -r_Coronal_LD*cosd(Theta_LD) r_Coronal_RA*cosd(Theta_RA) -r_Coronal_RA*cosd(Theta_RA) r_Coronal_IO*cosd(Theta_IO) -r_Coronal_IO*cosd(Theta_IO) r_Coronal_EO*cosd(Theta_EO) -r_Coronal_EO*cosd(Theta_EO) 0 0 0 0; ...
         r_Coronal_ES*sind(Theta_ES) -r_Coronal_ES*sind(Theta_ES) r_Saggital_LD*sind(Theta_LD) -r_Saggital_LD*sind(Theta_LD) r_Coronal_RA*sind(Theta_RA) -r_Coronal_RA*sind(Theta_RA) r_Coronal_IO*sind(Theta_IO) -r_Coronal_IO*sind(Theta_IO) r_Coronal_EO*sind(Theta_EO) -r_Coronal_EO*sind(Theta_EO) 0 0 0 0];

    beq=[L4L5_Lateral_force(i);L4L5_Anterior_force(i)-F_AP(i)*sind(Theta_AP);L4L5_Normal_force(i)-F_AP(i)*cosd(Theta_AP);L4L5_Saggital_Moment(i)-F_AP(i)*cosd(Theta_AP)*r_Saggital_AP(i);L4L5_Coronal_Moment(i);L4L5_Torque(i)];


    %Inequlaity constraints, intensity of each muscle doesn't exceed maximmum intensity
    Aineq=[1 0 0 0 0 0 0 0 0 0 0 0 0 -A_ES;0 1 0 0 0 0 0 0 0 0 0 0 0 -A_ES; 0 0 1 0 0 0 0 0 0 0 0 0 0 -A_LD; 0 0 0 1 0 0 0 0 0 0 0 0 0 -A_LD; ...
           0 0 0 0 1 0 0 0 0 0 0 0 0 -A_RA;0 0 0 0 0 1 0 0 0 0 0 0 0 -A_RA; 0 0 0 0 0 0 1 0 0 0 0 0 0 0 -A_IO; 0 0 0 0 0 0 0 1 0 0 0 0 0 -A_IO; ...
           0 0 0 0 0 0 0 0 1 0 0 0 0 -A_EO;0 0 0 0 0 0 0 0 0 1 0 0 0 -A_EO];
    bineq=[0;0;0;0;0;0;0;0;0;0];

    %lower bound = 0
    lb=[0 0 0 0 0 0 0 0 0 0 -50000 -50000 -50000 0];
    % upper bound is very big (the actual limit is from maximum intinsity)
    ub=[50000 50000 50000 50000 50000 50000 50000 50000 50000 50000 50000 50000 50000 50000];


    f=[0 0 0 0 0 0 0 0 0 0 0 0 0 1];

    x = linprog(f,Aineq,bineq,Aeq,beq,lb,ub,options);

    I(i)=x(end); %maximmum intinsity is the last elemnt in the optimized vector

end

%% Second linear programming, the objective function is summation of muscles forces

% optimization by linear programming at each frame,
% optimizing for the vector contains all muscles forces (4 variables to optimize)

% this optimization have a unique solution, refer to 'Bean1988'
% paper that this optimization is based on


for i=1:length(L5S1_Moment)
    %Equality constraints ,sum of moment contibutions of all muscles equal net
    %joint moment (equilibrium)

    Aeq=[0 0 -sind(Theta_LD) sind(Theta_LD) 0 0 0 0 0 0 0 1 0; ...
         sind(Theta_ES) sind(Theta_ES) 0 0 sind(Theta_RA) sind(Theta_RA) sind(Theta_IO) sind(Theta_IO) sind(Theta_EO) sind(Theta_EO) 0 0 1; ...
        -cosd(Theta_ES) -cosd(Theta_ES) -cosd(Theta_LD) -cosd(Theta_LD) -cosd(Theta_RA) -cosd(Theta_RA) -cosd(Theta_IO) -cosd(Theta_IO) -cosd(Theta_EO) -cosd(Theta_EO) 1 0 0; ...
         r_Saggital_ES*cosd(Theta_ES) r_Saggital_ES*cosd(Theta_ES) r_Saggital_LD*cosd(Theta_LD) r_Saggital_LD*cosd(Theta_LD) r_Saggital_RA*cosd(Theta_RA) r_Saggital_RA*cosd(Theta_RA) r_Saggital_IO*cosd(Theta_IO) r_Saggital_IO*cosd(Theta_IO) r_Saggital_EO*cosd(Theta_EO) r_Saggital_EO*cosd(Theta_EO) 0 0 0; ...
         r_Coronal_ES*cosd(Theta_ES) -r_Coronal_ES*cosd(Theta_ES) r_Coronal_LD*cosd(Theta_LD) -r_Coronal_LD*cosd(Theta_LD) r_Coronal_RA*cosd(Theta_RA) -r_Coronal_RA*cosd(Theta_RA) r_Coronal_IO*cosd(Theta_IO) -r_Coronal_IO*cosd(Theta_IO) r_Coronal_EO*cosd(Theta_EO) -r_Coronal_EO*cosd(Theta_EO) 0 0 0; ...
         r_Coronal_ES*sind(Theta_ES) -r_Coronal_ES*sind(Theta_ES) r_Saggital_LD*sind(Theta_LD) -r_Saggital_LD*sind(Theta_LD) r_Coronal_RA*sind(Theta_RA) -r_Coronal_RA*sind(Theta_RA)
```

```
        r_Coronal_IO*sind(Theta_IO) -r_Coronal_IO*sind(Theta_IO) r_Coronal_EO*sind(Theta_EO) -
        r_Coronal_EO*sind(Theta_EO) 0 0 0];

    beq=[L4L5_Lateral_force(i);L4L5_Anterior_force(i)-F_AP(i)*sind(Theta_AP);L4L5_Normal_force(i)-F_AP(
        i)*cosd(Theta_AP);L4L5_Saggital_Moment(i)-F_AP(i)*cosd(Theta_AP)*r_Saggital_AP(i);
        L4L5_Coronal_Moment(i);L4L5_Torque(i)];

    %Inequlaity constraints, intensity of each muscle doesn't exceed maximmum intensity
    Aineq=[];
    bineq=[];

    %lower bound = 0
    lb=[0 0 0 0 0 0 0 0 0 0 -50000 -50000 -50000];
    % upper bound is very big (the actual limit is from maximum intinsity)
    ub=I(i)*[A_ES A_ES A_LD A_LD A_RA A_RA A_IO A_IO A_EO A_EO 50000 50000 50000];


      f=[1 1 1 1 1 1 1 1 1 1 0 0 0]; %summation of muscles forces objective function
%       f=[0 0 0 0 0 0 0 0 0 0 1 0 0]; %desk compression objective function

    x = linprog(f,Aineq,bineq,Aeq,beq,lb,ub,options);

    L4L5Compression(i)=x(11);
    L4L5LateralShear(i)=x(12);
    L4L5AnteriorShear(i)=x(13);

    muscles_forces(1:13,i)=x;

end
```

Appendix B

Consent Form

Department of Systems Design Engineering

Date: November 07, 2017

Title of Project: ***Identification of Good Form among Construction Trade Workers***

Principal Investigator: ***Eihab Abdel-Rahman***
University of Waterloo, Department of ***Systems Design Engineering***
519-888-4567 Ext. ***37737***
eihab@uwaterloo.ca

Principal Investigator: ***Carl Haas***
University of Waterloo, Department of ***Civil and Environmental Engineering***
519-888-4567 Ext. ***35492***
chaas@uwaterloo.ca

Co Investigator: ***SangHyun Lee***
University of Michigan, USA, Department of ***Civil and Environmental Engineering***
+1 734-764-9420
shdpm@umich.edu

Co Investigator: ***Abdullatif Alwasel***
University of Waterloo, Department of ***Systems Design Engineering***
519-888-4567 Ext. ***33693***
aalwasel@uwaterloo.ca

Student investigator: **JuHyeong Ryu**
University of Waterloo, Department of ***Civil and Environmental Engineering***
519-884-4567 Ext. ***33929***
j4ryu@uwaterloo.ca

Student investigator: **Lichen Zhang**
University of Waterloo, Department of ***Civil and Environmental Engineering***
519-884-4567 Ext. ***30479***
l272zhan@uwaterloo.ca

Student investigator: **Mohsen Mutasem Diraneyya**
University of Waterloo, Department of ***System Design Engineering***
519-884-4567 Ext. ***38095***
mmdirane@uwaterloo.ca

Purpose of this Study

This study is being carried out as part of the PhD program requirements of Mr. JuHyeong Ryu's and the MASc. programs requirements for Ms. Lichen Zhang and Mr. Mohsen Diraneyya. Injury is one of the reasons that remove workers off the workforce early in their careers. These injuries are usually a result of working in dangerous postures that overtime lead to injury. Workers often use these posture as part of their daily work without realizing their long-term implications.

This study hypothesizes that experienced workers, those with 5 or more years of experience in construction trades, have adopted a healthy way of work. Thus, we aim at extracting this 'way of work' to deduce methods that novice workers can use to gain expertise while avoiding injuries and early retirement.

We are seeking to recruit trainees in construction apprenticeship programs to participate in this study. The study will recruit 150 trainees or more from all four of experience levels in the program (no experience, first year, second year, and third year trainees). We will also recruit 50 expert workers, with 5 or more years of experience, in construction trades and no apparent health issues.

Procedures Involved in this Study

The project consists of a one-hour session. This session will be utilised to collect data about how you move your body during your daily work tasks. Five sensors will be used in this study. Four sensors will be placed at each of the four joints: shoulder, hip, knee, and elbow. The fifth sensor is a wearable commercial motion-tracking suit that will be mounted on each body segment of upper and lower body. Specifically, upper arms, forearms, trunk, thighs, and legs. These are commercially available sensors that measure the joint angles. More information on the suit is available at (http://perceptionmocap.com/).

The researcher will also ask you to wear an off-the-shelf knee and elbow brace instrumented with sensors to measure the angle of rotation. The researcher will help you put on the braces. You will also be asked to wear a shoulder and hip sensor that will be attached to the skin on your shoulder using double sided tape provided by the manufacturer. The hip sensor will be attached to your clothing. The researcher will attach these sensors to your skin and clothing to ensure they are located in the right place and measuring your joint angles.

The motion tracking suit units, also known as IMUs, will be strapped to the body segment using a Velcro tape provided by the company. No adhesive material is used. The suit sends the measured motions wirelessly to a nearby computer.

The placement of the five sensors will not need removing of clothing articles. However, you will need to lift your shirt sleeve to expose your upper shoulder for the placement of the sensor on the skin of your shoulder area. After the placement of the sensor your sleeve can be lowered down again.

The sensors and braces will only be placed on the dominant side of your body, however, the motion suit will be placed on both sides to provide full motion tracking. In addition, two video cameras will record how you complete the task. One camera will record a side view while the other will record a back view of the scene. The sensors translate the joint rotation into change in resistance that can be read by the computer.

Prior to the task, you will be asked about your height, age, and weight. You will be asked to participate in one of the following tasks:

- Complete a wall starting from a lead wall. Mortar and blocks will be brought to the site of building. The wall is 6 blocks high and 12 blocks wide.
- Lay one or two courses of blocks in an existing wall.
- Complete tool and material handling tasks, such as rebar tying of reinforcement walls, drilling of reinforcement walls, and grinding or welds. Material will be laid out before hand for your task.

The sessions can be scheduled outside class time to suite your availability. As an appreciation of your time, a $10 Tim Card will be given for each participant. The amount received is taxable. It is your responsibility to report this amount for income tax purposes.

Risks to Participation and Associated Safeguards

- There is always a risk of muscle, joint or other injury in any physical work. However, the risks in this study are not anticipated to be greater than those required for your daily work tasks.
- If you are allergic to alcohol swabs used to sanitize the equipment and/or adhesive material used in double-sided tapes you are not be eligible to participate in this study as both materials will be used in this study.
- Sensors are not disposable and will be used for all participants in this study. The sensors will be sanitized using alcohol swabs between uses. The double-sided tape is disposable. Redness or a rash may occur when removing the tape from your skin. This should be temporary and disappear in one or two days.

Time Commitment

Participation in this study will require approximately 1 hour of your time. All sessions will be scheduled outside of class time.

Changing Your Mind about Participation

You may withdraw from this study at any time without penalty. To do so, indicate this to the researcher or one of the research assistants by saying, "I no longer wish to participate in this study".

Personal Benefits of Participation

There are no direct benefits for participating in the study. However, this study will provide researchers with knowledge about how workers move in their daily tasks thus allowing researchers to design work tasks more safe and efficient.

Confidentiality

To ensure the confidentiality of individuals' data, each participant will be identified by a participant identification code known only to the principal investigators and student investigators. Videotapes will be stored for 7 years, from the day of study anticipated completion (Aug 2021), in a secure area for further research purposes in the future e.g. alerting the worker using video data. No face blurring will be used as the video recording will not be facing the participant, hence, mostly no face recording is done. A separate consent will be requested in order to use the videotapes and/or photographs for teaching, for scientific presentations, or in publications of this work.

Data related to your participation will be submitted to an online data repository. It will be completely anonymized/de-identified by removing names and video recordings before submission. This process is integral to the research process as it allows other researchers to verify results and avoid duplicating research. Other individuals may access this data by downloading data spreadsheets. Should you choose, you may review all data that will be submitted before it is entered into data repository.

Participant Feedback

After the study is completed, you will be provided with an appreciation letter from the research team.

Concerns about Your Participation

This study has been reviewed and received ethics clearance through a University of Waterloo Research Ethics Committee (ORE#20023). If you have questions for the Committee contact the Chief Ethics Officer, Office of Research Ethics, at 1-519-888-4567 ext. 36005 or ore-ceo@uwaterloo.ca.

For all other questions contact Eihab Abdel-Rahman, Carl Haas, JuHyeong Ryu at 519-888-4567 Ext. 37737, 35492, and 33929 respectively.

Questions about the Study

If you have additional questions later or want any other information regarding this study, please contact (Eihab Abdel-Rahman, Carl Haas, JuHyeong Ryu) at 519-888-4567 Ext. 37737, 35492, and 33929 respectively.

CONSENT TO PARTICIPATE

By signing this consent form, you are not waiving your legal rights or releasing the investigator(s) or involved institution(s) from their legal and professional responsibilities.

I agree to take part in a research study being conducted by Dr. Eihab Abdel-Rahman, Dr. Carl Haas, and Juhyeong Ryu of the Department of Systems Design Engineering and Civil and Environmental Engineering, University of Waterloo.

I have made this decision based on the information I have read in the Information letter. All the procedures, any risks and benefits have been explained to me. I have had the opportunity to ask any questions and to receive any additional details I wanted about the study. If I have questions later about the study, I can ask one of the researchers (Eihab Abdel-Rahman, Department of Systems Design Engineering, Carl Haas, JuHyeong Ryu, Department of Civil and Environmental Engineering at 519-888-4567 exts. 33737, 35492, 33929 respectively).

I understand that I may withdraw from the study at any time without penalty by telling the researcher.

This study has been reviewed and received ethics clearance through a University of Waterloo Research Ethics Committee (ORE#20023). If you have questions for the Committee contact the Chief Ethics Officer, Office of Research Ethics, at 1-519-888-4567 ext. 36005 or ore-ceo@uwaterloo.ca.

Do you want to review data before it is stored in data repository?

Yes ☐ No ☐

______________________________	______________________________
Printed Name of Participant	Signature of Participant
______________________________	______________________________
Dated at Waterloo, Ontario	Witnessed

Consent to Use Video and/or Photographs

Sometimes a certain photograph and/or part of a video-tape clearly shows a particular feature or detail that would be helpful in teaching or when presenting the study results in a scientific presentation or publication. If you grant permission for photographs or videotapes in which you appear to be used in this manner, please complete the following section.

I agree to allow video and/or photographs to be used in teaching or scientific presentations, or published in scientific journals or professional publications of this work without identifying me by name.

Printed Name of Participant	Signature of Participant
Dated at Waterloo, Ontario	Witnessed

ELSEVIER LICENSE
TERMS AND CONDITIONS

Mar 24, 2019

This Agreement between University of Waterloo -- Mohsen Diraneyya ("You") and Elsevier ("Elsevier") consists of your license details and the terms and conditions provided by Elsevier and Copyright Clearance Center.

License Number	4554491380706
License date	Mar 22, 2019
Licensed Content Publisher	Elsevier
Licensed Content Publication	Journal of Biomechanics
Licensed Content Title	Adjustments to McConville et al. and Young et al. body segment inertial parameters
Licensed Content Author	R. Dumas,L. Chèze,J.-P. Verriest
Licensed Content Date	Jan 1, 2007
Licensed Content Volume	40
Licensed Content Issue	3
Licensed Content Pages	11
Start Page	543
End Page	553
Type of Use	reuse in a thesis/dissertation
Intended publisher of new work	other
Portion	figures/tables/illustrations
Number of figures/tables/illustrations	1
Format	both print and electronic
Are you the author of this Elsevier article?	No
Will you be translating?	No
Original figure numbers	Fig. 1
Title of your thesis/dissertation	Full-body Inverse Dynamics for Generic Motion Captured using Inertial Measurement Units
Expected completion date	Apr 2019
Estimated size (number of pages)	70

ELSEVIER LICENSE
TERMS AND CONDITIONS

Mar 24, 2019

This Agreement between University of Waterloo -- Mohsen Diraneyya ("You") and Elsevier ("Elsevier") consists of your license details and the terms and conditions provided by Elsevier and Copyright Clearance Center.

License Number	4554491086744
License date	Mar 22, 2019
Licensed Content Publisher	Elsevier
Licensed Content Publication	Journal of Biomechanics
Licensed Content Title	Analysis and measurement of lumbar trunk loads in tasks involving bends and twists
Licensed Content Author	A.B. Schultz,G.B.J. Andersson,K. Haderspeck,R. Örtengren,M. Nordin,R. Björk
Licensed Content Date	Jan 1, 1982
Licensed Content Volume	15
Licensed Content Issue	9
Licensed Content Pages	7
Start Page	669
End Page	675
Type of Use	reuse in a thesis/dissertation
Portion	figures/tables/illustrations
Number of figures/tables/illustrations	1
Format	both print and electronic
Are you the author of this Elsevier article?	No
Will you be translating?	No
Original figure numbers	Fig. 1
Title of your thesis/dissertation	Full-body Inverse Dynamics for Generic Motion Captured using Inertial Measurement Units
Expected completion date	Apr 2019
Estimated size (number of pages)	70