Influence of Motion Profile on Estimating Anatomical Elbow Joint Axes using Inertial Measurement Units

by

Timothy M. McGrath

B.S., Mississippi State University (2015)

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2017

Signature redacted

Author ....
Department of Aeronautics and Astronautics
May 25, 2017

Signature redacted

Certified by ....
Leia Stirling
Charles Stark Draper Professor of Aeronautics and Astronautics
Thesis Supervisor

Signature redacted

Accepted by ....
Professor Youssef M. Marzouk
Associate Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

Influence of Motion Profile on Estimating Anatomical Elbow Joint Axes using Inertial Measurement Units

by

Timothy M. McGrath

Submitted to the Department of Aeronautics and Astronautics on May 25, 2017, in partial
fulfillment of the requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

Current human motion measurement systems using inertial measurement units (IMUs) typically rely on precise alignment, static calibration poses, or dynamic calibration motions. Müller et al. recently proposed a method for online calibration of the human elbow anatomical joint axes via decomposition of angular velocity measurements. This thesis evaluated this calibration-free method in the context of two motion types. First, the method is evaluated with human-generated motions common to occupational rehabilitation, to investigate activities of daily living (ADLs) as online calibration motions. Second, the method is evaluated with parameterized sinusoidal motion, to investigate amplitudes and frequencies of motions that yield robust axis estimations. It was found for the axis of interest, high on-axis motion and low off-axis motion lead to precise axis estimation and high accuracy estimation of the pronation/supination elbow axis. Further, high off-axis motion and low on-axis motion yielded imprecise axis estimation and inaccurate estimation of both the flexion/extension and pronation/supination elbow axes. A comparative study of different filtering methodologies to the estimation of upper extremity human motion was also performed. Compared to a motion capture truth, the Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) were found to perform similarly. An implemented particle filter (PF) was found to perform better than both the EKF and UKF, and on the order of accuracy of a manufacturers black-box algorithm. This work is the first to evaluate a particle filter in the estimation of human motion by inertial sensors. The particle filter was then subject to a sensitivity analysis of the error of its estimated 3D orientation to its underlying algorithm inputs, namely, the accelerometer and magnetometer uncertainties, and number of particles. Recommended operational levels for these parameters are reported. Future work will combine the Müller auto-calibration method, robust IMU orientation filters, and knowledge of appropriate online motions to develop an IMU bias correction method for long periods of measurement.

Thesis Supervisor: Leia Stirling
Title: Charles Stark Draper Professor of Aeronautics and Astronautics

Acknowledgments

My work and education would not have been attainable without others who have offered immense support. First, to my advisor, Professor Leia Stirling: your dedication to the advancement of your students is unique and incredible. I cannot thank you enough for the time you have invested in me. You've been firm but supportive in pushing me outside of my comfort zone, and patient in my struggles. I'm a better engineer and problem solver than I was two years ago, and your advising is to thank for that. Thank you for giving me this opportunity, thank you for your patience, thank you for your cheerful words of encouragement in times of success. I look forward to working with you on a doctorate over the coming years.

My parents, Tim and Cheryl, have sacrificed everything to see their son make something of himself. I will never truly be able to repay either of you for the love and support you've given me. Throughout my life, you both have worked tirelessly to provide for our family. I can only hope to show others, or to eventually show my children, a mere fraction of the love you've given me. To my sister, Amber, whose drive and perseverance in the face of adversity is a shining light in my life. And to my grandmother, Faye, who cared for me as her own son. This is also for my Aunt Rosalyn, who valued education, and my Uncle David, who inspired me to take up engineering; to play loud music; to love nature. I love you all, and would not be who I am today without any of you.

To my beautiful and talented girlfriend, Paromita. In a very trying two years, you have been the source of constancy in my life. From one thousand three hundred and seventeen miles away, I never felt alone. I've grown as a human with you, and wouldn't know what to do without you. Your heart, mind, and sense of humor make me fall for you every day–I love you. I could fill another thesis with how much you've done for me: thank you.

I also couldn't have done any of this without the wisdom of other faculty within the MVL and the camaraderie of the other students within this lab. I wouldn't want to be anywhere else. And to the friends I've drifted from, the new friends I've made, and the ones I have yet to meet: thanks for making this life a good one.

And last but probably not least, to Mike and the Muddy Charles Pub. That place has made this whole experience a lot more bearable. In a place that needs plenty of mental health support, you guys do a damn good job. A lot of blood, sweat, and beers have gone into this work, so thank you for the latter.

This work was supported in funding through the National Science Foundation, through the Graduate Research Fellowships Program (GRFP) and NSF CAREER Award No. IIS-1453141. The motion capture data from the Carnegie-Melon motion capture database (`mocap.cs.cmu.edu`) was created with funding from NSF EIA-0196217.

Table of Contents

List of Figures

List of Tables

Nomenclature

$\mathbf{A}$	Quaternion Derivative Update Matrix
$\vec{\mathbf{a}}_k$	1x3 Accelerometer measurement at discrete time step k, m/s^2
$\mathbf{B}$	Active control matrix
$\mathbf{D}_n^b$	DCM from $\mathcal{B}$ to $\mathcal{N}$
Δt	Discrete time interval, seconds
$\mathbf{f}(\vec{\mathbf{x}}_k)$	State dynamics/transitional model at discrete time step k
$\vec{g}_c$	Acceleration due to gravity
$\vec{\omega}_k$	1x3 Gyroscope measurement at discrete time step k, rad/s^2
$\vec{h}_c$	Local magnetic field vector
$\mathbf{h}(\vec{\mathbf{x}}_k)$	Observation model at discrete time step k
$\mathbf{I}_{mxm}$	The m-by-m identity matrix
$\vec{\mathbf{m}}_k$	1x3 Magnetometer measurement at discrete time step k, G
$\mathbf{Q}$	Process noise covariance matrix
$\mathbf{q}_k$	1x4 Quaternion at discrete time step k, scalar element first
$\mathbf{q}$	State orientation quaternion

$\mathbf{R}$	Measurement noise covariance matrix
$^{\vec{a}}\mathbb{S}$	Accelerometer scale factor matrix
$^{\vec{g}}\mathbb{S}$	Gyroscope scale factor matrix
$^{\vec{m}}\mathbb{S}$	Magnetometer scale factor matrix
$\vec{\mathbf{u}}$	Input control vector
$\vec{\sigma}_a^2$	Accelerometer measurement uncertainty 3-vector
$\vec{\sigma}_m^2$	Magnetometer measurement uncertainty 3-vector
$\vec{\mathbf{v}}$	Gaussian noise $\vec{\mathbf{v}} \sim N(0, \mathbf{R})$
$\vec{\mathbf{w}}$	Gaussian noise $\vec{\mathbf{w}} \sim N(0, \mathbf{Q})$
$\hat{\mathbf{x}}_k$	Predicted state at discrete time step k
$\vec{\mathbf{x}}_k$	Estimated state at discrete time step k
$\mathbf{y}_k$	1x6 Observed measurement vector at discrete time step k
$\mathbf{z}_k$	1x6 Predicted measurement vector at discrete time step k
$\mathbf{0}_{mxm}$	The m-by-m zero matrix

Chapter 1

Introduction, Motivation, Previous Work, and Literature Review

1.1 Motivation

The study of human biomechanics is important to the advancement of rehabilitation therapy, sports medicine, human performance, and human-robot interaction. Better biomechanical models and biomechanical measurement systems enable more robust systems for interacting with and understanding the human. For example, in rehabilitation, understanding human motion informs the plan of care. For sports applications, understanding motion can lead to improved strategy development. To this end, an accurate measurement of human joint angles is desired. This measurement is complicated, as human motion is characteristically nonlinear, uncorrelated, and non-smooth [4]. While other computer vision approaches to human measurement exist [5], the current *gold standard* for measurement of human motion is optical motion capture [6]. Passive optical motion capture uses infrared light emitted from mounted cameras to localize reflective spheres (*markers*) in three-dimensional space. Active motion capture includes methods such as markers which are infrared light sources. Optical motion capture has the advantage of being very accurate in the triangulation of the markers, but the interpretation of the data may not be obvious. To translate 3D position of markers in space to human joint angles, models of the human body must be assumed. An often-used model is OpenSim [7]. From markers set on major anatomical landmarks of the body, a least-squares optimization to fit a model may be performed to estimate the joint angles.

This approach, of course, is financially expensive due to the equipment required to collect the data, and is cumbersome to post-process. The subject must be fitted in non-reflective clothing with markers placed precisely at anatomical landmarks. Any misplacement would manifest itself as an error in the estimation of joint angles. The infrared cameras are expensive, must be calibrated and oriented properly to view a necessarily-large capture volume, and suffer from marker dropout when the markers are occluded by another object. This approach can be inappropriate for the clinician, sports performance expert, or engineer when measurements need to be made at low cost and in the environment of interest.

Inertial sensors, in contrast, offer an inexpensive way to make robust measurements. Decades of research in state estimation and filtering have enabled the appropriate mathematical models and techniques to estimate orientation in three-dimensional space. The modern inertial measurement unit (IMU), containing accelerometers, gyroscopes, and magnetometers, is a common tool of motion measurement in non-deforming bodies. Unfortunately, usage of IMUs to measure human motion is dependent on a kinematic model of motion, which can be sensitive to misalignments of the IMUs on the subject. We desire a method which can circumvent these tedious calibration and alignment methods. The Müller method [3] can estimate joint axes in a calibration-free manner, but it is unclear what motions are required to obtain good estimates of the axes. Furthermore, IMUs can drift over time, which confounds their usage in long-term measurement applications.

While methods exist in the literature to estimate IMU orientation, these methods have not been compared on human-generated data. This work examines three types of filters, and performs a sensitivity analysis of the best performing filter to the underlying parameters. To assess how the choice of motion affects the calibration-free axis estimation methodology, a robotic experiment was performed using a set of occupational human motion data and parameterized sinusoidal motion.

1.2 The Filtering Problem

The Bayesian filtering problem presents a classic conundrum in state estimation–it is desired to uncover the true state of a process, yet that process can only be measured with discrete, noisy samples. This problem has been well studied, beginning with the seminal work of Norbert Weiner [8]. Algorithmic implementations for reduction in noise to help unmask a state soon followed [9]. A leap forward occurred in 1960 when R. E. Kalman [10] presented an optimal, recursive estimator of state with minimal covariance for linear dynamical systems under white noise. As opposed to previous works, this work created a provably-optimal method for state estimation of linear processes. Now, engineers had a filter with optimal performance for practical calculation of a hidden state. Its potential was famously uncovered by the work of Stanley F. Schmidt, who first applied the Kalman filter method to the C-5A Inertial Navigation System (INS) [11]. Immediately, the Kalman filter was used in the actually-nonlinear systems of interest in the aerospace community. Alongside developments in the mathematics of nonlinear filtering [12], Schmidt and his group would implement a simple technique to extend the Kalman filter to nonlinear systems. The aptly-named *Extended Kalman Filter* would linearize the dynamics of a nonlinear system by the well-known Taylor Series expansion. From this point forward, a simple, relatively inexpensive method existed to estimate the state of more general noisy stochastic processes. In the decades following, filtering methodologies of many flavors would spring up, each with their own strengths and weaknesses. The most popular are the *Unscented* Kalman Filter [13] and collection of Sequential Monte Carlo methods know as the *Particle Filter* [14].

1.2.1 Inertial Sensor Fusion

The aforementioned filtering methods are fairly generic, in that they are applicable to any process which can be characterized by an observation and state transition model under additive white noise. We must contextualize them for the IMU problem at hand.

The inertial sensor fusion problem can be viewed as a subset of the larger, general filtering problem. With this more specific problem come specific intricacies. For example, sensors can return faulty information if damaged. Magnetometers specifically can return bad measurements in the presence of a changing or unknown magnetic field. Furthermore, selection of the state representation can also create complexity. For example, Euler angles are prone to gimbal lock, and quaternion representations of orientation are not mathematically independent amongst its four components (they must have unity norm). This violates a basic assumption of the particle filter, that the state is a mathematically independent quantity. These nuances present specific hurdles which must be considered in our implementations of the classic filtering techniques.

In this work, our inertial measurement units are 9 degrees-of-freedom (DOF)–a 3-axis accelerometer, a 3-axis gyroscope, and a 3-axis magnetometer. An example of these small devices is shown in Figure 1.1. These measurements are fused into an orientation estimate represented by the *quaternion*, a 4-dimensional vector of unity norm which avoids the common gimbal-locking phenomenon of the Euler angle representation [15].

In general, there will be a common structure within the algorithms which describe the state filtering. Namely, two models (multivariate functions) are required. First, a (possibly nonlinear) state dynamics model $\mathbf{f}(\vec{\mathbf{x}}_k)$. The dynamics model (otherwise called the *state transition model*) maps the previous state and the discrete time to a prediction of the new state. Practically, in the inertial sensor fusion problem, this will map the previous state quaternion $\mathbf{q}_{k-1}$ and gyroscope measurements $\vec{\omega}_k$ to a new quaternion $\mathbf{q}_k$. Second, an observation model $\mathbf{h}(\vec{\mathbf{x}}_k)$ is required. This model will map the predicted state to a set of predicted measurements. Conceptually, the models are represented as:

$$\hat{\mathbf{x}}_k \leftarrow \mathbf{f}(\vec{\mathbf{x}}_k, \Delta t, \vec{\omega}_k) \tag{1.1a}$$

$$\mathbf{z}_k \leftarrow \mathbf{h}(\vec{\mathbf{x}}_k) \tag{1.1b}$$

The 6-element array $\mathbf{z}_k$ of predicted measurements has the exact same structure as the

6-element array of *observed* measurements $\mathbf{y}_k$: 3 accelerometer measurements and 3 magnetometer measurements, in that order.

$$\mathbf{y}_k \leftarrow \begin{bmatrix} \vec{\mathbf{a}}_k & \vec{\mathbf{m}}_k \end{bmatrix} \tag{1.2}$$

1.2.2 System Model

The classical discrete-time system can be described through the dynamics presented below. The Bayesian approach, as implemented by the particle filter, assigns probabilities to these models, rather than deterministic quantities. However, the models are similar, and it is of use to understand the system model in its deterministic form before extrapolating to the probabilistic model.

Figure 1.1 APDM Opal v2 IMU

The deterministic model describes the rotation of a rigid body in space. This rigid-body rotating system is observed by the measurement of the IMU's three sensors: accelerometer, gyroscope, and magnetometer. The filtered state will be an estimate of the IMU's orientation in space, represented by a quaternion.

State Definition

The state vector is defined as an orientation quaternion, $\mathbf{q} = [q_0, q_1, q_2, q_3]$ where q_0 is the scalar quaternion component and the other three components $\hat{q} = [q_1, q_2, q_3]$ are the vector components of the quaternion.

Explicitly written, the state vector $\vec{\mathbf{x}}_k$ is:

$$\vec{\mathbf{x}} = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix} \tag{1.3}$$

Naturally, this quaternion can easily be converted to Euler angles. Quaternions are computationally desirable because they are not subject to the singularity/ambiguity issues that are common to Euler angle definitions of orientation.

State Transition Model

In the traditional Kalman Filter algorithm, it is assumed that the system can be modeled as:

$$\vec{\mathbf{x}}_k = \mathbf{A}\vec{\mathbf{x}}_{k-1} + \mathbf{B}\vec{\mathbf{u}}_{k-1} + \vec{\mathbf{w}}_{k-1} \tag{1.4}$$

where $\vec{\mathbf{w}}$ is normal Gaussian noise, uncorrelated with other system noises. $\mathbf{B}$ is the matrix that defines control dynamics while $\vec{\mathbf{u}}$ is the control input vector, all variables that influence the state vector $\vec{\mathbf{x}}$.

In the case of the linear, basic Kalman Filter, $\mathbf{A}$ is a constant matrix. However, in general dynamic motion, $\mathbf{A}_{k|k+1}$ is a nonlinear, time-dependent matrix. It serves to map the previous state at $k-1$ to the current state at k.

Note that k is the iterator in the algorithm. It represents time, where k is the current time and $k-1$ is the previous time step. In this document, the notation in Eq (1.4) will be used consistently, where subscripts k and $k-1$ represent the values at times k and $k-1$, respectively. This is the discrete, deterministic state model, as employed by algorithms like the Extended Kalman Filter and Unscented Kalman Filter. The Particle filter is a stochastic method, which will be elaborated on in Section 1.2.6.

In the nonlinear, time-varying case of the dynamics function $\mathbf{f}(\vec{\mathbf{x}})$, the aforementioned nonlinear matrix $\mathbf{A}_k$ will use the previous state $\vec{\mathbf{x}}_{k-1}$ to predict the current state $\hat{\mathbf{x}}_k$ by:

$$\hat{\mathbf{x}}_k = \mathbf{f}(\vec{\mathbf{x}}_{k-1}, \vec{\omega}, \Delta t) = \mathbf{A}_k \vec{\mathbf{x}}_{k-1} \tag{1.5}$$

where $\mathbf{A}_k$ is:

$$\mathbf{A}_k = \begin{bmatrix} \cos(|\vec{\omega}|\Delta t/2) & -\vec{\omega}_x/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) & -\vec{\omega}_y/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) & -\vec{\omega}_z/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) \\ \vec{\omega}_x/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) & \cos(|\vec{\omega}|\Delta t/2) & \vec{\omega}_z/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) & -\vec{\omega}_y/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) \\ \vec{\omega}_y/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) & -\vec{\omega}_z/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) & \cos(|\vec{\omega}|\Delta t/2) & \vec{\omega}_x/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) \\ -\vec{\omega}_z/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) & \vec{\omega}_y/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) & -\vec{\omega}_x/|\vec{\omega}|\sin(|\vec{\omega}|\Delta t/2) & \cos(|\vec{\omega}|\Delta t/2) \end{bmatrix} \tag{1.6}$$

Note that $|\vec{\omega}|$ is simply the 2-norm of the gyro rate vector. $\mathbf{A}_k$ simply represents the dynamics of a rotating body over a small time interval Δt. It is more intuitively represented in 3-axis coordinate form, but Equation 1.6 represents it for quaternion dynamics.

Measurement Model

The IMU sensors consist of a tri-axis accelerometer, tri-axis gyroscope, and tri-axis magnetometer. These measured vector quantities are $\vec{\mathbf{a}}$. $\vec{\omega}$, and $\vec{\mathbf{m}}$, respectively. The IMU is subject to total acceleration (sum of gravity $\vec{g}_c$ and body acceleration $\vec{\mathbf{a}}_{body}$), the angular rotation rates $\vec{\mathbf{g}}_{true}$, and the local magnetic field $\vec{h}_c$. Likewise, the measurements are subject to biases $\vec{\mathbf{b}_\mathbf{a}}$, $\vec{\mathbf{b}_\mathbf{g}}$, and $\vec{\mathbf{b}_\mathbf{m}}$ and uncorrelated Gaussian white noises $\vec{\mathbf{v}}_a$, $\vec{\mathbf{v}}_g$, and $\vec{\mathbf{v}}_m$.

$$\vec{\mathbf{a}} = {}^{\vec{a}}\mathbb{S}[\mathbf{D}_n^b(\mathbf{q})(\vec{g}_c + \vec{\mathbf{a}}_{body})] + \vec{\mathbf{b}_\mathbf{a}} + \vec{\mathbf{v}}_a \tag{1.7a}$$

$$\vec{\omega} = {}^{\vec{g}}\mathbb{S}\vec{\mathbf{g}}_{true} + \vec{\mathbf{b}_\mathbf{g}} + \vec{\mathbf{v}}_g \tag{1.7b}$$

$$\vec{\mathbf{m}} = {}^{\vec{m}}\mathbb{S}\mathbf{D}_n^b(\mathbf{q})\vec{h}_c + \vec{\mathbf{b}_\mathbf{m}} + \vec{\mathbf{v}}_m \tag{1.7c}$$

In (1.7), $\mathbf{D}_n^b$ is the direction cosine matrix (DCM) that represents the rotation of body frame $\mathcal{B}$ into the navigation frame $\mathcal{N}$. Finally, ${}^{\vec{a}}\mathbb{S}$, ${}^{\vec{g}}\mathbb{S}$, and ${}^{\vec{m}}\mathbb{S}$ are the scale factor matrices of the three measurement sensors. These matrices are used to correct any axis sensitivity differences by a sensor. In the case of a perfect sensor, these scale factor matrices are simply the 3-by-3 identity matrix.

The measured vector is the concatenation of the accelerometer and magnetometer readings $\mathbf{y}_k = \begin{bmatrix} \vec{\mathbf{a}}_k & \vec{\mathbf{m}}_k \end{bmatrix}^T$. As we have established, the motion is nonlinear, so the aforementioned relation could be more aptly written as:

$$\mathbf{y} = \begin{bmatrix} \vec{\mathbf{a}}_k \\ \vec{\mathbf{m}}_k \end{bmatrix} = \mathbf{h}(\vec{\mathbf{x}}_k) + \begin{bmatrix} \vec{\mathbf{b}}_{\mathbf{a}} \\ \vec{\mathbf{b}}_{\mathbf{m}} \end{bmatrix} + \begin{bmatrix} \dot{\vec{\mathbf{b}}}_{\mathbf{a}} \\ \dot{\vec{\mathbf{b}}}_{\mathbf{m}} \end{bmatrix} \Delta t + \begin{bmatrix} \vec{\mathbf{v}}_a \\ \vec{\mathbf{v}}_m \end{bmatrix} \tag{1.8}$$

This function $h(\vec{\mathbf{x}}_k)$ is quite important. It defines the rotational quaternion dynamics. It is succinctly defined as:

$$\mathbf{h}(\vec{\mathbf{x}}_k) = \begin{bmatrix} \mathbf{D}_n^b(\mathbf{q}_k)^{-1} & \mathbf{0}_{3x3} \\ \mathbf{0}_{3x3} & \mathbf{D}_n^b(\mathbf{q}_k)^{-1} \end{bmatrix} \begin{bmatrix} \vec{g}_c \\ \vec{h}_c \end{bmatrix} \tag{1.9}$$

where $\mathbf{D}_n^b(\mathbf{q}_k)$ is the 3x3 rotation matrix defined by:

$$\mathbf{D}_n^b(\mathbf{q}_k) = (q_0^2 - \hat{q}^T\hat{q})\mathbf{I}_{3x3} + 2\hat{q}\hat{q}^T - 2\mathbf{q}_0[\hat{q}\times] \tag{1.10}$$

with $\hat{q} = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}^T$ and $[\hat{q}\times] = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix}$. Fully expanded, it would read:

$$\mathbf{D}_n^b(\mathbf{q}_k) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_0q_3 + q_1q_2) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_0q_1 + q_2q_3) \\ 2(q_0q_2 + q_1q_3) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{1.11}$$

Next, it becomes important to remember that this rotation is not linear. We are not assuming linearity in our dynamics. In the EKF, a local linearization of the nonlinear dynamics is taken to numerically compute orientation at each time step. In the case of the Unscented Kalman Filter, discrete probability points are described on a sigmoid [16]. Recall Equation (1.5), the function that defines the nonlinear dynamics. In the case of the particle filter, a Bayesian inference technique will be used to discretely approximate the posterior probability density of the state.

Process Noise Covariance Matrix Q

The process noise covariance matrix represents the error due to unmodeled dynamics such as wind gusts or random vibration, and also uncertainty due to gyroscope noise. Typically, small numbers are chosen and then tuned until the estimated state agrees with some external truth measurement.

Measurement noise covariance matrix R

The noise of a static sensor measurement may be measured. This simple process consists of setting the IMU on a motionless surfaces and recording data. Take multiple points, then compute the variance of the sample set for both the accelerometer and magnetometer. In real systems, there is noise, which will make this quantity nonzero.

Furthermore, in real systems, there is uncertainty in measurement due to more than just the physical noise of the sensor. Unmodeled dynamics in acceleration and magnetic field are relevant examples. In acceleration, our measurement model presented in Section 1.2.2 only models static acceleration due to tilt of the IMU in Earth's gravitational field. Dynamic body accelerations introduce errors to this model. Likewise, the measurement model of the magnetometer assumes that ambient magnetic field is static in direction and magnitude, which is not always the case in laboratory environments with electrical equipment.

If the uncertainties of the accelerometer and magnetometer measurement are written as $\vec{\sigma}_a^2$ and $\vec{\sigma}_m^2$ (note that these are 3-vectors, there is a separate uncertainty about each axis), then the matrix $\mathbf{R}$ is computed as:

$$\mathbf{R} = \begin{bmatrix} \vec{\sigma}_a^2 & \vec{\sigma}_m^2 \end{bmatrix}^T \mathbf{I}_{6x6} \tag{1.12}$$

1.2.3 The Kalman Filter

While the traditional, linear Kalman filter is not applicable in the nonlinear dynamics domain of this work, the filtering algorithm is presented below, along with a discussion. The algorithm will be expanded into its nonlinear forms in the following sections.

$$\hat{\mathbf{x}}_k = \mathbf{A}\vec{\mathbf{x}}_{k-1} + \mathbf{B}\vec{\mathbf{u}}_k \tag{1.13a}$$

$$\hat{\mathbf{P}}_k = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} \tag{1.13b}$$

$$\mathbf{K}_k = \hat{\mathbf{P}}_k\mathbf{H}^T\left(\mathbf{H}\hat{\mathbf{P}}_k\mathbf{H}^T + \mathbf{R}\right)^{-1} \tag{1.13c}$$

$$\vec{\mathbf{x}}_k = \hat{\mathbf{x}}_k + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{H}\hat{\mathbf{x}}_k\right) \tag{1.13d}$$

$$\mathbf{P}_k = \left(\mathbf{I}_{4x4} - \mathbf{K}_k\mathbf{H}\right)\hat{\mathbf{P}}_k \tag{1.13e}$$

Note that the matrix $\mathbf{H}$ is simply the linear matrix analog of the measurement model $\mathbf{h}(\vec{\mathbf{x}}_k)$. Recall that under linear dynamics, used here in the Kalman filter, that the predicted measurements can always be represented by the linear transformation $\mathbf{z}_k = \mathbf{H}\vec{\mathbf{x}}_k$. Yet unintroduced are the matrices $\mathbf{P}$ and $\mathbf{K}$.

State Estimate Covariance Matrix P

The state covariance matrix $\mathbf{P}$ is a direct measure of the belief in the estimated state. If the state is a 4-vector quaternion, then $\mathbf{P}$ is a 4x4 matrix. The diagonal of the matrix can be thought of as the direct variance of the estimated state parameters, while the off-diagonal elements are measures of the covariance of the state with other elements of the same state. The proof of optimality of the Kalman filter for linear dynamics, in fact, minimized the mean square error (MMSE) of this state covariance.

A common measure of total state belief is the trace of the covariance matrix. The operation $trace(\mathbf{P})$ sums the diagonal elements of the matrix–yielding an overall scalar metric for state belief. This measure is commonly reported alongside time plots of the state. Along with general state variance, this time series gives a concept of how long it took for the filter to converge to a stable estimate. If initialized to $\mathbf{P}_1 = \mathbf{I}_{4x4}$, the filter will generally begin to rapidly reduce the values of these diagonal elements. Likewise, the value of $trace(\mathbf{P})$ will increase in regions where data are noisy or less believable.

Kalman Gain Matrix K

The Kalman gain matrix $\mathbf{K}$ is a function of this aforementioned state covariance. In equation 1.13d, the Kalman matrix multiplies the *difference* between the predicted and measured sensor data. If this difference is very small, the prediction of the state was modeled well by $\mathbf{A}$ when compared to what is measured. However, if there is a large discrepancy between what we believe we will observe and what we do observe, a correction is necessary. In lies the power of the Kalman filter. The Kalman gain matrix represents trust in our measurements. If the Kalman gain matrix $\mathbf{K}$ is small, we trust our sensors very little. In this case, the updated state $\vec{\mathbf{x}}_k$ would be very close to the predicted state $\hat{\mathbf{x}}_k$. The Kalman gain matrix is the arbiter between these two values. If the Kalman gain matrix is large, we trust our sensors a lot, and the discrepancy $\mathbf{y}_k - \mathbf{H}\hat{\mathbf{x}}_k$ weighs heavily into the computation of the final state estimate $\vec{\mathbf{x}}_k$.

1.2.4 Extended Kalman Filter

Almost immediately after the development of Kalman's filtering algorithm, Schmidt began to investigate nonlinear extensions of the algorithm [11]. The most obvious formulation was to utilize the Taylor series expansion to linearize these nonlinear dynamics and measurement models. To do so, we define the Jacobians of the system transition and measurement models as:

$$\widetilde{\mathbf{F}}_{k-1} = \left.\frac{\partial \mathbf{f}(\vec{\mathbf{x}})}{\partial \vec{\mathbf{x}}}\right|_{\hat{\mathbf{x}}_{k-1}} \tag{1.14}$$

$$\mathbf{H}_{k-1} = \left.\frac{\partial \mathbf{h}(\vec{\mathbf{x}})}{\partial \vec{\mathbf{x}}}\right|_{\hat{\mathbf{x}}_{k-1}} \tag{1.15}$$

These now-linearized state transition and measurement models can now be used in a familiar algorithm.

$$\hat{\mathbf{x}}_k = \mathbf{f}(\vec{\mathbf{x}}_{k-1}) \tag{1.16a}$$

$$\hat{\mathbf{P}}_k = \widetilde{\mathbf{F}}_{k-1}\mathbf{P}_{k-1} + \widetilde{\mathbf{F}}_{k-1}^T + \mathbf{Q}_{k-1} \tag{1.16b}$$

$$\mathbf{K}_k = \hat{\mathbf{P}}_k\widetilde{\mathbf{H}}_k^T \left(\widetilde{\mathbf{H}}_k\hat{\mathbf{P}}_k\widetilde{\mathbf{H}}_k^T + \mathbf{R}\right)^{-1} \tag{1.16c}$$

$$\vec{\mathbf{x}}_k = \hat{\mathbf{x}}_k + \mathbf{K}_k(\mathbf{y}_k - \mathbf{h}(\hat{\mathbf{x}}_k)) \tag{1.16d}$$

$$\mathbf{P}_k = \left(\mathbf{I}_{4x4} - \mathbf{K}_k\widetilde{\mathbf{H}}_k\right)\hat{\mathbf{P}}_k \tag{1.16e}$$

Now that the dynamics are nonlinear, there is no guarantee of optimality as Kalman proved for his original filter (recall that the Kalman filter was optimal in a MMSE sense). The accuracy of the filter, intuitively, would depend on how well a first-order Taylor series expansion can approximate the nonlinear dynamics.

The first order Taylor series expansion about a point x would simply be equivalent to finding the slope of the function $f(x)$ at point x and using this linear approximation for $f(x)$ at x. If this is a good approximation, i.e., the function $f(x)$ is not highly nonlinear at x over the sample interval Δt, then the EKF will have high accuracy. In scenarios of high nonlinearity, this first-order approximation may be unsuitable.

Formulations of the EKF do exist with higher-order Taylor series approximation of the nonlinear dynamics model $f(x)$ [17]. However, higher-order Taylor series approximations are in general computationally expensive. Furthermore, every order of approximation beyond first order yields quickly diminishing returns in filter accuracy. While higher-order EKFs may be used in specific applications, they are not common for general nonlinear filter schemes.

1.2.5 Unscented Kalman Filter

While the EKF offered the *de facto* nonlinear filtering method for decades, it has a specific set of disadvantages. Namely, as previously discussed, highly nonlinear functions can doom first-order Taylor series approximations, and higher-order EKF formulations can be undesirable for computational complexity and diminishing returns in filter accuracy. Furthermore, the Jacobians of nonlinear functions can very quickly become intractable to calculate for large functions of many variables. In these scenarios, new schema are desired. Jeffrey Uhlmann [16] formulated a new approach to utilizing the available information within a nonlinear function: the unscented transform.

The Unscented Transform

Uhlmann conjectured: rather than approximating a nonlinear function as linear, why not approximate a nonlinear function as a distribution which would still capture the mean and covariance of the underlying nonlinear function? After all, Kalman's algorithm is a direct operation with the mean and covariance of that nonlinear function.

The unscented transform, explained fully in [18], transforms an n-vector of state and

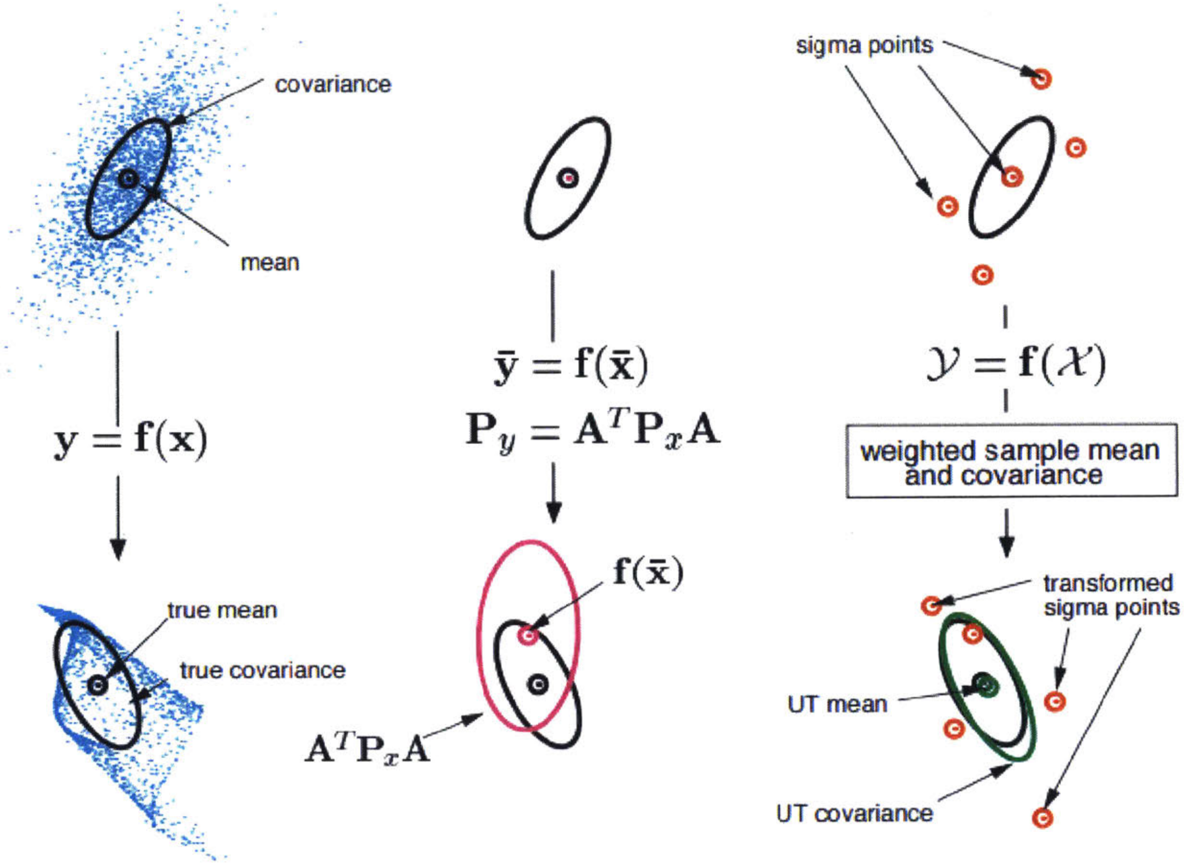


Figure 1.2 Illustration of mean and covariance propagation in the Kalman filter framework: true state (left); linearized EKF (middle); unscented transform (right) [1]

n-by-n matrix of covariance into an n+1 x n+1 matrix of *sigma points*, which is a symmetric matrix that represents the statistical distribution of the underlying state and covariance. This distribution retains the first moment (mean), the second moment (covariance), and under certain assumptions, the third moment (skew) of the state and covariance. The unscented transform is illustrated conceptually in Figure 1.2.

Once this set of sigma points has been computed, these individual points can be transformed by the nonlinear function $f(x)$ *exactly*, rather than an approximation being necessary. These transformed sigma points can then be decoded to find the new state mean and covariance. This method can be employed in the Kalman filter to propagate nonlinear dynamics and measurement models.

The UKF has become a cornerstone algorithm in state estimation when the dynamics and measurement models are either highly nonlinear, non-differentiable, or too burdensome to compute the Jacobian. Furthermore, the unscented transform has been employed with weights on the means and covariance of the state to aide performance of the filter when the functions being transformed are well-studied. The UKF shows notable improvements over the canonical EKF [13], but is more computationally complex than the EKF. In cases where the dynamics of a system can be modeled accurately as a Taylor series linearization, then the EKF is more appropriate. Also, while the UKF can capture skew, the unscented transform cannot propagate general non-Gaussian distributions. In these cases, a particle filter may be preferred [19].

1.2.6 Particle Filter

The particle filter method (or *Sequential Monte Carlo method*) is a stochastic approach to solving the filtering problem. The filter is a computational implementation of Bayes' Rule. Given the classical equation,

$$P(\theta|Y) = \frac{P(Y|\theta)P(\theta)}{P(Y)} \tag{1.17a}$$

$$Y = \begin{bmatrix} \vec{\mathbf{a}}_k \\ \\ \vec{\mathbf{m}}_k \end{bmatrix} \tag{1.17b}$$

$$\theta = \mathbf{q} \tag{1.17c}$$

we can recursively estimate $P(\theta|Y)_k$, or the state posterior probability distribution at time k. $P(Y|\theta)$ represents the likelihood model, and represents the "likelihood" of our current predicted state, $\hat{\mathbf{x}}_k$ being correct given our measurement $\mathbf{y}_k$. The prior probability $P(\theta)$ represents the statistical extrapolation of our nonlinear dynamics update model, a simple integration of the gyroscope rates. It could be interpreted as the gyroscope prediction, which is then multiplied by the conditional likelihood probability. Finally, $P(Y)$ is the marginal likelihood of the measurements, which is constant for every prediction/hypothesis proposed by $P(\theta)$. As such, it is often thought of as simply a normalizing constant, which becomes unimportant as the particle weights are normalized to sum to one for obvious reasons.

The Particle Filter Algorithm

The particle filter [20] represents a commonly-used stochastic sampling method, as opposed to the deterministic sampling manner of the UKF. The PF is initialized by sampling a number of *particles* randomly from a predefined initial probability distribution about the initial mean. These particles are propagated through the nonlinear dynamics model $\mathbf{f}(\vec{\mathbf{x}}_k)$. Then, the weights $\mathbf{w}$ of these particles are sampled from a second, not necessarily identical, distribution. Finally, if these particles are found to diverge or converge, they may be resampled using a number of different algorithms. Equation 1.18 suggests a basic particle filter for IMU tracking purposes [20].

The particle filter is a powerful statistical technique because a finite system of discrete particles can capture complex distributions. Accordingly, more computational time must be spent to compute the dynamics of these particle systems. The computational burden of the particle filter can be a significant problem for use in online analysis or in systems with low-cost on-board hardware.

$$\mathbf{xp}_{1,i} \leftarrow \mathcal{N}(\vec{\mathbf{x}}_1, \sigma) \tag{1.18a}$$

$$\mathbf{w}_{1,i} \leftarrow \frac{1}{N_p} \tag{1.18b}$$

$$\mathbf{xp}_{k,i} \leftarrow \mathbf{f}(\mathbf{xp}_{k-1,i}) \tag{1.18c}$$

$$\mathbf{y}_k \leftarrow \text{current measurement} \tag{1.18d}$$

$$\mathbf{z}_{k,i} \leftarrow \mathbf{h}(\mathbf{xp}_{k,i}) \tag{1.18e}$$

$$\mathbf{w}_{k,i} \leftarrow \frac{\exp(-\frac{1}{2}(\mathbf{y}_k - \mathbf{z}_{k,i})^T \mathbf{R}^{-1}(\mathbf{y}_k - \mathbf{z}_{k,i}))}{\sum_{i=1}^{N_p} \exp(-\frac{1}{2}(\mathbf{y}_k - \mathbf{z}_{k,i})^T \mathbf{R}^{-1}(\mathbf{y}_k - \mathbf{z}_{k,i}))} \mathbf{w}_{k-1,i} \tag{1.18f}$$

$$\mathbf{w}_{k,i} \leftarrow \text{resample weights as necessary} \tag{1.18g}$$

$$\vec{\mathbf{x}}_k \leftarrow \mathbf{w}_{k,i}\mathbf{xp}_{k,i} \tag{1.18h}$$

In this implementation, the initial distribution from which particles are sampled in equation 1.18a is from a normal curve with a mean of the initial orientation and a variance of the state uncertainty. The distribution the particle weights $\mathbf{w}_{k,i}$ are sampled from in Eq 1.18b is simply a normal curve with a mean as the difference between the observed measurement $\mathbf{y}_k$ and the particle-predicted measurement $\mathbf{z}_{k,i}$. The variance of this probability distribution is R_k, the sensor noise matrix .

The Particle Filter in Practice

The Bayes' Rule components discussed previously in 1.17 are modeled as normal curves, as previous literature has done [20]. Greater tests and practice can be used to tune the statistical distributions from which the particles are sampled. The likelihood model $P(\theta|Y)$ is simply a normal curve whose mean represents the difference between predicted and observed measurements, and whose variance is represented by the entire uncertainly associated with a measurement. It may be due to measured sensor variance, partial observations, environmental factors, sampling constraints, etc. It has been found in practice that the sensor noises of the accelerometer and magnetometer are sensitive parameters of the performance of the particle filter.

These probability distributions are not represented by analytical functions but as "particles," which are randomly distributed about the initial posterior probability curve. As the recursion is carried out, these particles evolve to represent the updated state posterior probability distribution. The state estimate $\mathbf{q}$ is computed as the weighted sum of these discrete probability representations. This transformation could be qualitatively described by stating,

$$P(\theta|Y)_{t=k} \rightarrow P(\theta)_{t=k+1} \rightarrow P(\theta|Y)_{t=k+1} \rightarrow P(\theta)_{t=k+2} \rightarrow \ldots \tag{1.19}$$

where the symbol → could be thought of as to mean "informs."

While a formal introduction to Markvoian processes and Monte Carlo simulation are beyond the scope of this thesis (see Pardoux [21]), one nuance of this implementation should be noted. Sampling the state estimate from any distribution requires that the individual state components are *independent*, i.e., sampling the state components from a distribution would represent a valid state. However, the quaternion violates this assumption, as the 4-component quaternion component must have a unit 2-norm. If 4 individual components are sampled from a distribution, they are not guaranteed (in fact, they would likely almost never) uphold this assumption. For that reason, a modified formulation of the particle filter must be used that estimates a properly independent state, unlike the quaternion.

Cheng and Crassidis [22] presents an interesting formulation of the particle filter to avoid this problem. An attitude error representation is formulated, where the global attitude representation is defined as a quaternion, but a local error representation is used with modified Rodriguez parameters as the state which is estimated via the particle filter. Rodriquez parameters are another method of representing orientation. Unlike the quaternion (and like Euler angles), the components of this representation are independent.

Since the estimated state is independent, the assumption of Monte Carlo processes has now been met. This formulation of the particle filter will be used for the remainder of this thesis.

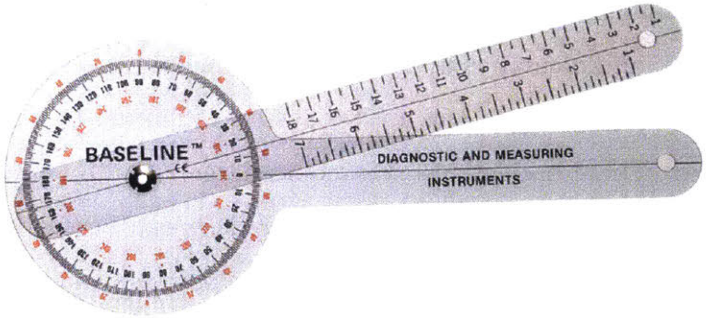


Figure 1.3 A goniometer, image from: `https://www.fab-ent.com/media/02_Evaluation/12-1000.jpg`

1.3 Human Joint Angle Measurement

The measurement of joint angles is a non-trivial task. The simplest measurement tool is the goniometer (Figure 1.3). While a manual tool like the goniometer is cheap and widely available, the tool can be misused by user through improper joint axis alignment. A difference in alignment of the instrument to an anatomical landmark will create a significant difference in measurement. De Kraker et al. [23] found that while the precision of a goniometer is +/- 2 degrees, in operational usage, the same measurement by multiple operators varied by +/- 8 degrees. Furthermore, the goniometer only yields a static measurement. This could be useful for understanding range of motion, but is not useful in dynamic analyses. Time-series data provides a more useful data for understanding how the human joint moves over time. In a case of occupational therapy, while the initial and final joint orientations are important, the trajectory to achieve those orientations is also relevant. Further, it is important to understand the relative motions between joints over time, which is not possible with static measures that can obtain one joint at a time.

1.3.1 Optical Methods

Optical tracking methods are the most robust and commonly used methods for tracking human joint angles. Optical methods rely on a camera to capture certain objects or features in the environment, and a computer to process the data into an interpretable form. While most methods of optical measurements are accurate in tracking a marker in space, these data require a model of human kinematics in order to map the data to human joint orientations. The measurement of human motion via optical system is limited by the underlying model of human kinematics. These optical systems require large laboratory capture volumes and possibly-impractical subject attire. Furthermore, markers can be occluded by the subject or other objects in the capture volume, and incomplete datasets can make analysis difficult and time-consuming post-hoc.

In any optical method, underlying models of the human subject are normally assumed. Optical methods are limited by the fidelity of the model used to estimate human kinematics. These models can possibly be incomplete or ill-posed for patients with certain physical impairments, with whom clinicians are often tasked. Software such as OpenSim (open source), AnyBody (AnyBody Technology Inc., Salem, MA, USA), and MSC ADAMS (MSC Software, Inc., Newport Beach, CA, USA) use motion capture and embedded models of kinematics to estimate joint angles. In general, these models require precise placement of markers on certain anatomical landmarks on the human body.

1.3.2 Inertial Sensors

Inertial sensors can also be used to measure human motion. The advantages of inertial sensors have been discussed previously. These sensors are low-cost, require no capture volume, and can work *in situ*. This enables motion capture for applications that do not allow for the infrastructure required for using optical motion capture technologies. Filtering for IMU orientation produces more robust results than direct interpretation of sensor information alone [24]. Sabitini [25] presented a novel EKF which fused magnetic measurements to provide a better estimate of heading, rather than relying on the drift-prone gyroscope measurements. Madgwick et al. [26] presented a novel orientation filter based around a gradient descent method to minimize error. This method was used to track position and orientation of a foot over a period of 60 seconds. Madgwick found RMS errors of less than 1.5 degrees about each IMU axis. The particle filter has been applied to body tracking in optical motion capture systems [27], but not to motion tracking via inertial sensors.

The use of integrated inertial sensors for measurement of human motion spans roughly two decades. In 1999, John Elwell posed [28] a method for inertial navigation for soldiers in the battlefield. Elwell was concerned with general body motion, but found that IMU drift became a challenge, even with state-of-the-art gyroscope and accelerometer packages and using a zero-velocity update. Prior to the development of filters with more robust complementary measurement updates, it was common to align an axis of an IMU with a human limb segment, so that the integration of the gyroscope could be interpreted as rotation of the limb directly [29]. Giansanti et al. [30] combined gyroscopes and accelerometers to track orientation in human motion, but restricted motion to only a few seconds to avoid IMU drift. Zhou et al. [31] measured motion over 12 seconds, while also applying a weighted least squares frequency filtering technique to damp erratic measurements from the gyroscope. Methods which rely on sensor alignment may include robust nonlinear filters [31–34], but alignment with an anatomical axis is imprècise. Seel et al. [2] illustrates the cumbersome process of measuring sensor misalignment and correction. Figure 1.4 shows this problem: multiple IMUs are placed on the limb, each of which has its own coordinate frame. These coordinate frames must be reconciled with the joint axis, and so that relative orientation must be computed. This calibration technique is only as accurate as the ability of the researcher to measure the relative orientation between the IMU and the joint axis of interest.

Another technique to align the IMU with a body frame for human joint angle computation involves static calibration poses. Cutti et al. [35] show a calibration procedure where IMUs are placed on the subject, and then the subject moves to a predefined pose. In this predefined pose, the true joint angles of the subject are assumed and the IMU 3D orientation in space can be measured. From these pieces of information, the static calibration quaternions can be discerned. Luinge et al. [29, 36] developed a method which used knowledge about elbow constraints and a static pose to compute orientation of the lower arm relative to the upper arm. Other methods [34, 37–39] similarly seek an IMU sensor-to-subject calibration quaternion from a static pose method.

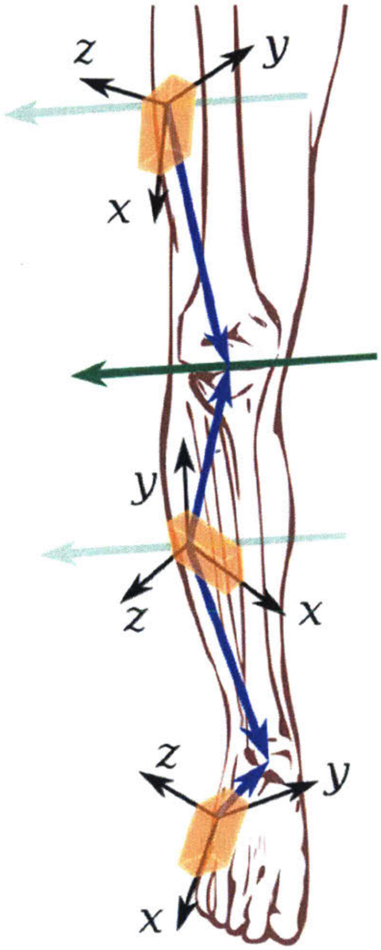


Figure 1.4 IMU alignment on a body limb [2]

Some calibration techniques use dynamic calibration motions. Mayagoitia et al. [40] and Lotters et al. [41] use individual accelerometers and gyroscopes, aligned to the lower extremity. Gyroscopes were calibrated through 90 degree and 180 degree motion sweeps in a set time, while accelerometers were calibrated through simple alignment with and normal to gravity. Favre et al. [39] presents a hybrid static-dynamic method to estimate the static calibration quaternions through both knee motion and static poses. Some works [37,42] require the subject to perform motions while holding a pose, e.g., flex the knee while maintaining a static ankle pose. In general, calibration can be a time-consuming process and requires known motions and/or postures for each joint of interest.

Müller et al. [3] recently presented a method to overcome the problems of calibration. This method, which will be explained more thoroughly in Section 1.4, estimates the joint axes of interest by decomposition of the angular velocity of IMUs. This method has potential to introduce IMUs to applications which otherwise would have been intractable because of cumbersome calibration and calibration quaternion misalignments.

1.3.3 Drift

As previously mentioned, a prevailing problem with inertial sensors is drift. In modern sensors, drift is primarily due to the integration of noisy gyroscope data. One method to solve this problem is through better sensor fusion techniques, as presented previously. In general, these techniques have been developed to provide a measurement update via accelerometers and/or magnetometers [43], or provide better models of motion. Other techniques include heuristics on gyroscope data. A commonly-used heuristic is the zero-velocity update [44]. This heuristic works by sensing times when the magnitude of gyroscope data is small, and then setting these data to zero angular velocity. The intuition for this heuristic comes from the gyroscope being an analog-to-digital sensor. This sensor cannot produce zero angular velocity due to noise. So if the rotation rate is near zero, we can argue that it is below a signal-to-noise threshold, and set these data to zero. This would reduce drift by not integrating gyroscope noise during periods of low rotation. However, this heuristic cannot apply in situations where drift is occurring in the gyroscope over a real angular velocity signal.

Drift confounds human motion measurement when measurements are desired to be taken over a long period of time. For example, in studies where long-term performance evaluation is required to assess fatigue, intractable IMU drift would prevent the researcher from being able to compare data over a long period of time. Methods are desired to solve this drift problem. In Section 1.4, this thesis will present an algorithm developed by Müller et al. [3] that has potential to aid in the correction of drift.

1.4 Müller Method

The human elbow joint can be modeled approximately as a two DOF joint. Anatomically, we call these rotations flexion/extension and pronation/supination of the elbow. Note that while pronation/supination is a more complex interaction of the radius and ulna bones of the forearm, this is commonly modeled as a single degree of freedom at the elbow.

An IMU is placed both distal and proximal to the elbow joint, such that all relative motions between the IMUs occur at the elbow joint. Figure 1.5 illustrates this. No assumption of alignment of the IMUs are made; the IMUs are free to be placed in any orientation on the arm, provided that one is distal and one is proximal to the joint.

The coordinate frame of the IMU on the upper arm will be referred to as frame **A**, and likewise, the frame of the IMU on the lower

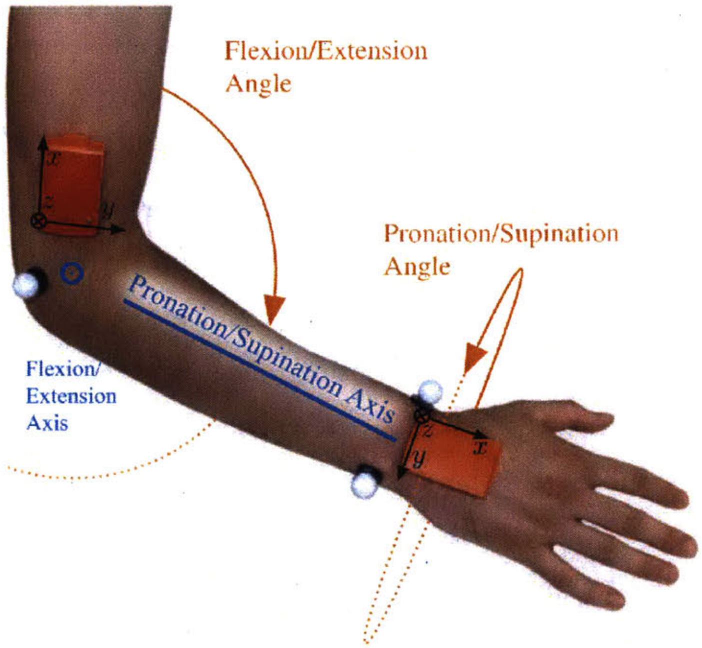


Figure 1.5 Human arm with IMUs and two DOF of the elbow illustrated [3] .

arm will be referred to as frame **B**.

The flexion/extension axis of the

elbow will be referred to as axis $\vec{\mathbf{a}}$, and the pronation/supination axis of the elbow as axis $\vec{\mathbf{b}}$. As such, the anatomical axis $\vec{\mathbf{a}}$ expressed in frame **A** will be noted as $\vec{\mathbf{a}}^{\mathbf{A}}$. The pronation/supination axis of the elbow expressed in terms of the coordinate frame of the lower arm would likewise be noted as $\vec{\mathbf{b}}^{\mathbf{B}}$.

The relative angular velocity between the two IMUs can be computed. The angular velocities of the respective IMUs would be noted in their local frames as $\vec{\omega}^{\mathbf{A}}_{arm}$ for the upper arm IMU and $\vec{\omega}^{\mathbf{B}}_{wrist}$ for the lower arm IMU. The relative angular velocity between the two IMUs can be expressed as:

$$\vec{\omega}^{\mathbf{A}}_{rel} = -\vec{\omega}^{\mathbf{A}}_{arm} + \vec{\omega}^{\mathbf{A}}_{wrist} = -\vec{\omega}^{\mathbf{A}}_{arm} + \mathbf{D}^{\mathbf{B}}_{\mathbf{A}}\vec{\omega}^{\mathbf{B}}_{wrist} \tag{1.20}$$

where $\mathbf{D}^{\mathbf{B}}_{\mathbf{A}}$ is the rotation matrix from frame **A** to frame **B**. This operation is simply the rotation of the lower arm angular velocity into the frame of the upper arm, and then a difference operation between $\vec{\omega}^{\mathbf{A}}_{arm}$ and $\vec{\omega}^{\mathbf{A}}_{wrist}$.

An interesting property holds about the two anatomical axes of the elbow relative to their IMUs: the pronation/supination axis $\vec{\mathbf{a}}$ is fixed in frame **A**, while the pronation/supination axis of the elbow $\vec{\mathbf{b}}$ is fixed in frame **B**. Taking advantage of this assumption about $\vec{\mathbf{a}}^{\mathbf{A}}$ and $\vec{\mathbf{b}}^{\mathbf{B}}$, we can express the relative angular velocity as a decomposition about these two axes.

$$\vec{\omega}^{\mathbf{A}}_{rel} = \alpha\vec{\mathbf{a}}^{\mathbf{A}} + \beta\mathbf{D}^{\mathbf{B}}_{\mathbf{A}}\vec{\mathbf{b}}^{\mathbf{B}} \tag{1.21}$$

where scalars α and β are simply angular rates about the two axes $\vec{\mathbf{a}}^{\mathbf{A}}$ and $\vec{\mathbf{b}}^{\mathbf{B}}$, respectively.

We know that due to factors like sensor noise and imperfect orientation estimation, these relative angular velocities cannot be estimated perfectly. So we introduce an error term $\vec{\mathbf{e}}$ and summarize our two expressions for relative angular velocity between the two IMUs.

$$\vec{\omega}_{rel}^{\mathbf{A}} = -\vec{\omega}_{arm}^{\mathbf{A}} + \mathbf{D}_{\mathbf{A}}^{\mathbf{B}}\vec{\omega}_{wrist}^{\mathbf{B}} + \vec{\mathbf{e}} \tag{1.22a}$$

$$\vec{\omega}_{rel}^{\mathbf{A}} = \alpha\vec{\mathbf{a}}^{\mathbf{A}} + \beta\mathbf{D}_{\mathbf{A}}^{\mathbf{B}}\vec{\mathbf{b}}^{\mathbf{B}} + \vec{\mathbf{e}} \tag{1.22b}$$

In a perfect world, this error $\vec{\mathbf{e}}$ would be zero. While in practice it cannot be zero, an optimization method is used to minimize the error to solve for the anatomical axes $\vec{\mathbf{a}}^{\mathbf{A}}$ and $\vec{\mathbf{b}}^{\mathbf{B}}$.

Rearranging equation 1.22b to solve for $\vec{\mathbf{e}}$:

$$\vec{\mathbf{e}} = (\alpha\vec{\mathbf{a}}^{\mathbf{A}} + \beta\mathbf{D}_{\mathbf{A}}^{\mathbf{B}}\vec{\mathbf{b}}^{\mathbf{B}} - \vec{\omega}_{rel}^{\mathbf{A}}) / \left\|\vec{\omega}_{rel}^{\mathbf{A}}\right\| \tag{1.23}$$

Given that we already know $\vec{\omega}_{arm}^{\mathbf{A}}$ from equation 1.20, we can employ an optimization method to compute the anatomical axes by minimizing the following cost function:

$$\mathbf{J}(\vec{\mathbf{a}}^{\mathbf{A}}, \vec{\mathbf{b}}^{\mathbf{B}}) = \frac{1}{N}\sum_{k=1}^{N} \vec{\mathbf{e}}_k^T \vec{\mathbf{e}}_k \tag{1.24}$$

This cost function is formulated as a common least-squares nonlinear objective function. The minimization of the cost function in equation 1.24 yields the optimal solution for the anatomical axes $\vec{\mathbf{a}}^{\mathbf{A}}$ and $\vec{\mathbf{b}}^{\mathbf{B}}$.

Müller et al. used an online gradient-descent method. In this thesis, the Levenberg-Marquardt algorithm was the used optimization method. The Levenberg-Marquardt algorithm interpolates between a Gauss-Newton and gradient-descent method. This method is robust and popular in practice for solving nonlinear, nonconvex least squares problem. More information on the algorithm can be found in Nocedal et al. [45].

Once the anatomical axes $\vec{\mathbf{a}}^{\mathbf{A}}$ and $\vec{\mathbf{b}}^{\mathbf{B}}$ are computed, the joint angles about these axes can be computed. Davenport [46] first considered the problem, and gave conditions for the existence of rotation angles. Davenport proved that rotational angles are only guaranteed to exist in the case of 3 orthogonal axes. Furthermore, the closer these axes are to parallel, the more likely it is that rotation angles do not exist about them. While the explanation of this computation is beyond the scope of this work, Piovan et al. [47] presents a compact algorithm to decompose the relative orientation $\mathbf{D}_{\mathbf{B}}^{\mathbf{A}}$ as two rotations about possibly-nonorthogonal axes $\vec{\mathbf{a}}^{\mathbf{A}}$ and $\vec{\mathbf{b}}^{\mathbf{B}}$. While these two rotation axes are slightly non-orthogonal, this effect is not strong enough to threaten the existence of the joint angles.

The nonorthogonality of axes $\vec{\mathbf{a}}^{\mathbf{A}}$ and $\vec{\mathbf{b}}^{\mathbf{B}}$ could be due to error, but can also be due to human anatomy. The carry angle of the human elbow is generally about 11 degrees for men and 15 degrees for women from orthogonal [48]. This angle allows the forearm to clear the hips in normal walking motion.

The Müller method was demonstrated in a proof-of-concept study [3], but these results are not generalizable to all human motion. The method is intriguing, but it is not clear what motions will yield accurate estimates of the joint axes, or whether natural motions are usable for on-line axis estimation.

1.5 Aims

There are methods for estimating human motion using inertial sensors. There has not been a comparative study of the EKF, UKF, and PF to human motion estimation. Furthermore, the particle filter has not been used for human motion estimation in inertial sensors. State estimation-based approaches still usually require precise calibrations, either statically or dynamically. The Müller method presents a possible method to calibrate human-inertial systems online, but the motions required to achieve accurate joint axis estimation are unknown. Hence, this works aims to:

1. Compare the orientation error as estimated by the EKF, UKF, and PF for a sample of upper extremity human motions relative to motion capture
2. Determine the sensitivity of the particle filter estimation error to its input parameters
3. Characterize the performance of the Müller algorithm when used for a set of human and parametric motions

In Chapter 2, we study the application of the aforementioned filters to human motion estimation by comparing the developed filters to an optical motion capture truth in the context of three motions performed by a single subject: a circle and square drawn with the hand in the horizontal plane, and a natural grasping motion. We continue Chapter 2 by then performing a sensitivity analysis of the particle filter to its parametric inputs to draw operational conclusions about recommended values for these parameters. Chapter 2 represents the first application of the particle filter to the estimation of human motion via inertial sensors. In Chapter 3, we evaluate the Müller method in the context of human-generated motion to evaluate the capability of motions common to occupational rehabilitation to provide robust axis estimation. We further evaluate the Müller method against sinusoidal motion, parameterized by frequency and amplitude, to determine the characteristics of motion which lead to robust axis estimation. Finally, in Chapter 4, we provide a summary of the results, limitations of the current work, and suggested future work.

Chapter 2

Experimental Comparison of Filters and Parameter Sensitivity Analysis

Chapter 1 outlined the development of three common filtering methodologies for this work–the Extended Kalman Filter (EKF), the Unscented Kalman Filter (UKF), and a Particle Filter (PF). Validation of these filters has been experimentally shown, but not in the context of inertially measuring human motions. This chapter will evaluate filter performance in this context.

Furthermore, for any given filter, the state is not estimated *perfectly*, so errors in estimation can be exacerbated by using inappropriate filter parameters. For example, the gyroscope, as discussed previously, is used as a predictor to transition the state estimate forward in time. This gyroscope has some measurable, physical noise. A parameter $\mathbf{Q}$ was introduced previously as the state covariance. The noise from the gyroscope is one, but not the only, factor that contributes to total state uncertainty. Some other notable sources of noise: analog-to-digital sampling, clock drift, and imperfect dynamics. The latter is important–remember that the EKF, as presented, uses only a first-order Taylor series approximation. To fully capture the dynamics, an infinite Taylor series would be required. This imperfect approximation contributes to state uncertainty. For this reason, the filter parameters should be tuned to acceptable values. In general, filter parameters are normally selected by experiment-specific tuning. This chapter will also evaluate the sensitivity of tuning parameters in the context of the particle filter.

2.1 Aims and Hypotheses

IMUs as a commercial product usually come with a proprietary, built-in black box algorithm for computing orientation. These black-box algorithms may be robust and commercially validated, but give the researcher little control over details of the algorithm which may be relevant to his or her situation. An open algorithm is more desirable because it allows the research to recalibrate more easily for different magnetic environments, to tune filter inputs for a given situation, and to implement heuristic methods which may be appropriate for a specific motion type.

To this end, it is desired to evaluate the performance of different filter types against a truth and also against a proprietary algorithm. In this case, the proprietary algorithm is an unscented Kalman filter with unknown heuristics as implemented by ADPM in the Opal v1 IMUs. The goal is to create an open algorithm that provides researcher flexibility that performs similar to or better than the proprietary algorithm.

This chapter addresses the following aims:

1. To establish levels of baseline accuracy for each filter type at estimation of human motion and compare performance
2. To understand the sensitivity of the particle filter to its inputs; specifically, the number of particles, accelerometer variance, and magnetometer variance

This first aim will be evaluated through three human-generated motions (evaluated at two different speeds each). The three motions selected were drawing a circle in the horizontal plane with the hand, drawing a square in the horizontal plane with the hand, and grasping a cup. The filter was compared via the root-mean-square error against a motion capture truth datum. This will be elaborated on in Section 2.2.1.

The second aim will be evaluated through a numerical experiment with statistical analysis. The following hypothesis will be evaluated:

1. H_1: Different levels of number of particles, accelerometer variance, and magnetometer variance will each significantly affect the accuracy of the particle filter

The accelerometer and magnetometer noise parameters are the main inputs to the filter. They describe the "trust" of the measurement model in the filter update step. The Kalman filter acts as the decision maker for which to trust–the state prediction model or the measurement update model. In anecdotal experience, we find that these measurement variances greatly impact the filter output. The particle filter additionally requires a scalar to represent the number of particles in the simulation. The accuracy of the particle filter is a function of how well the discrete particle set can represent the state posterior distribution. We can intuitively reason that more particles allows better the particle set to better capture this posterior distribution, and thus yield a more accurate filtered state.

2.2 Filter Comparison via Motion Capture

Once the filters were implemented into an algorithmic form, they were approximately tuned by visual observation of the estimated state. The observationally tuned algorithm was then compared to the proprietary algorithm for the selected motions. The following describes the experimental process and results.

2.2.1 Experimental Method

The comparison of human kinematic motions was performed using an Opal v1 IMU (APDM Inc., Seattle, Washington) and motion capture (Vicon Motion Systems, Inc., Oxfordshire, UK).

APDM Opal IMU

The APDM Opal IMU system is an off-the-shelf system for human tracking. The IMU is packaged as a 3-axis accelerometer, 3-axis gyroscope, and 3-axis magnetometer. The IMU system also features desirable noise characteristics, an output sample rate of up to 128 Hz, and sensor ranges well covering the human capability. The full specifications of the IMU are found in Appendix G.

This sensor outputs both raw and calibrated sensor data, as well as a filtered orientation estimate using an on-board UKF. The calibrated sensor data is of particular use, because APDM uses a fairly robust model to remove bias. It allows direct orientation filter comparison using the same set of bias-removed sensor data. The Opal IMU is shown in Figure 1.1. For the motion described below, one long sensor dataset was captured. The differing filter methodologies were applied to the same dataset for comparison.

The manufacturer IMU uses a black-box algorithm to compute orientation that cannot be tuned or recalibrated for different usages. The manufacturer's algorithm uses a set of unknown heuristics to increase performance. Two heuristics were implemented in these open algorithms.

The first is a heuristic to set the accelerometer variance high if the magnitude of acceleration deviates by 50% of Earth gravity (or approximately 4.9g). The observation model presented in Chapter 1 for the filters is a model for *static* orientation accelerometer measurement. In cases of dynamic acceleration, acceleration measurements would deviate from the observation model's expectation. To counteract this, the variance of the measurement is set high during these periods, meaning we "trust" the accelerometers less in these periods.

The second heuristic drives the variance of the manometer measurement when the magnitude of the magnetic field deviates by 50% from expected, or the magnetic dip angle deviates by ten degrees. This dip angle is the angle the magnetic field lines make with the local horizontal. In the same logic as the accelerometer heuristic, in these periods, we do not trust the magnetometer's reading, so the effective noise is set to a high value.

The open algorithms were performed with an accelerometer noise of 10 m^4/s^2 and a magnetometer noise of 100 μT^2. The particle filter was run with 20 particles in the simulation.

Vicon Motion Capture System

For laboratory estimation of human motion, the Vicon motion capture system is the gold standard. The system projects infrared light from a series of cameras. Infrared bounces from reflective spherical markers back to the cameras, which use a triangulation algorithm to estimate the position of the marker in space.

These small markers can be attached to many surfaces. For this experiment, reflective markers were attached to a 3D-printed IMU holder, which fits around the hand. The Vicon system returns the spatial coordinates of the individual markers in the Vicon coordinate frame. The apparatus shown in Figure 2.1 has a cavity on the underside which allows it to sit over the APDM Opal IMU. This smooth overlay allows for direct comparison of orientation via both the IMU and the Vicon system. The three reflective markers on top of the apparatus allow for cross-products to compute a three-dimensional local IMU coordinate frame. This coordinate frame's relative orientation to the Vicon frame can then be computed. The other two reflective markers on the sides are used for simple filling of data gaps in the Vicon Nexus software. Thus, the orientation estimate of the filter IMU readings can be directly compared to the computed orientation of the Vicon mount in space.

Human Motion Measured

For this study, only one subject was required to gather a pilot dataset. This subject consented to the study and the study protocol was approved by COUHES. The Opal IMU was attached to the subject's right hand, with the Vicon apparatus attached on top. The subject then completed 3 motions at 2 different speeds (slow and fast) each. The first motion was the drawing of a circle in the horizontal plane. The second motion was the drawing of a square in the horizontal plane. These first two motions seek to probe filter performance–can we artificially induce filter error by some nonlinear motion or motion with discrete changes

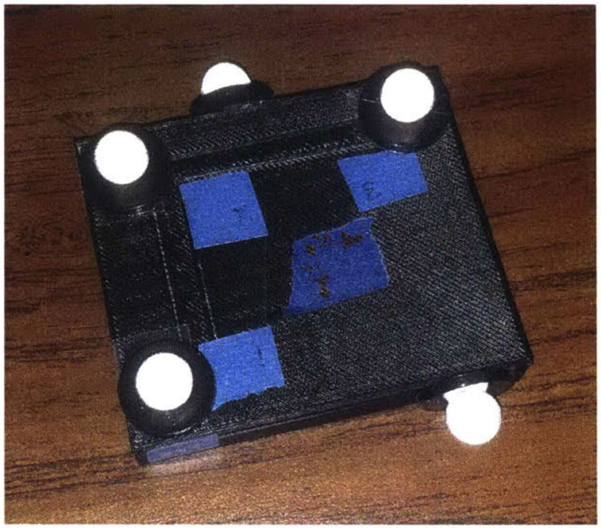


Figure 2.1 Vicon markers on a 3D-printed IMU mount

in direction? The motion discontinuities introduce a nonlinearity that may cause difficulty in the filters. The third motion is more kinematically natural: a grasping motion. The six discrete motions, in order, were:

1. Circle, slow
2. Circle, fast
3. Square, slow
4. Square, fast
5. Grasp A
6. Grasp B

The circles were approximately 30 centimeters in radius, and the squares were approximately 30 centimeters in half-width. The grasping motion consisted of the subject's right hand near the chest, and then extending at the shoulder, elbow, and wrist to simulate grasping a bottle of water or similar object. This was done twice, as shown above. At the point of grasp, the shoulder and elbow of the subject was fully extended. There was no physical object–this was a simulated motion.

Procedure

Before the IMU was outfitted onto the subject, the IMU was first aligned on a table with the Vicon coordinate system and measurements were taken. This measurement serves to document the magnetic heading of the Vicon coordinate frame relative to the North-West-Up (NWU) magnetic world frame, which is the common world frame of choice for inertial sensors.

The six motions studied were completed in one consecutive run. Between the six motions, an approximate still period of ten seconds was held. In these still period, the subject held their hand still in space. This experiment yielded a single Vicon dataset and a single IMU sensor dataset. These data must be parsed to pull out the individual motions. Also, the Vicon dataset and the IMU dataset were aligned in time.

Filter Error Metric

The response variable here must represent filter error from truth in some way. Recall that the filter is estimating orientation in space as a quaternion. However, it is typically more intuitive to interpret Euler angles, ϕ, θ, and ψ. For historical purposes [10], the 2-norm of the individual root mean square error component will be used. More explicitly,

$$FE = \left|\left|RMSE(\hat{\phi} - \phi_{true}), RMSE(\hat{\theta} - \theta_{true}), RMSE(\hat{\psi} - \psi_{true})\right|\right|_2 \tag{2.1}$$

2.2.2 Analysis

The implemented filters were computed relative to the magnetic frame which was aligned with the Vicon system. The orientation was therefore directly computed in the Vicon frame. The manufacturer's filter, however, needed to be rotated to align with the Vicon frame. The NWU frame and Vicon frame only differ in heading (rotation about the vertical axis, z). The manufacturer's orientation estimate was corrected by computing this heading offset and correcting by rotating the manufacturer's offset around the vertical axis by the offset amount.

The Vicon data needed to be synchronized with the IMU data. First, the Vicon data was transformed (via the cross-product calculation described previously) into the true orientation. This true orientation was synchronized with the IMU estimated orientation by visually inspection to match peaks of the dataset.

The long IMU dataset was parsed into the six individual motions. This was done by visual inspection using still periods between motions. The parsed motion start and end times were applied identically to each evaluated filter.

The computed error of the filter is shown in Equation 2.1. This is simply the norm of the vector which represents the x, y, and z root-mean-square errors of the computed Euler angles.

2.2.3 Results and Discussion

The magnetic offset between the Vicon frame and the NWU frame was found to be approximately 31 degrees. This heading correction was applied to the manufacturer's filter for proper comparison.

The estimated orientations for the entire experiment can be seen in Figure 2.2. The errors of these estimated orientations vs. the Vicon truth datum are shown in Figure 2.3. The RMSE errors parsed for each trial motion are shown in Table 2.1. The data of these individual motion trials are shown in Appendix A. By looking at the rotation about the z-axis, the six individual motions and the still periods between can be seen. The horizontal circle and square motions were intended to be completed in the horizontal plane, this was not strictly enforced, and no tactile or visual feedback was given to the subject. The data show deviations from horizontal which gave about a 20 degree range of motion about the y-axis and 15 degrees about the x-axis. The two grasp motions at the end show significant movement about the x and y axes, as expected.

All filtered results have RMSE less than 10.5 deg compared to the Vicon data. This provides a benchmark for filter performance. Ideally, our developed filters would perform on the order of the manufacturer's filter. As can be seen, while the manufacturer's filter

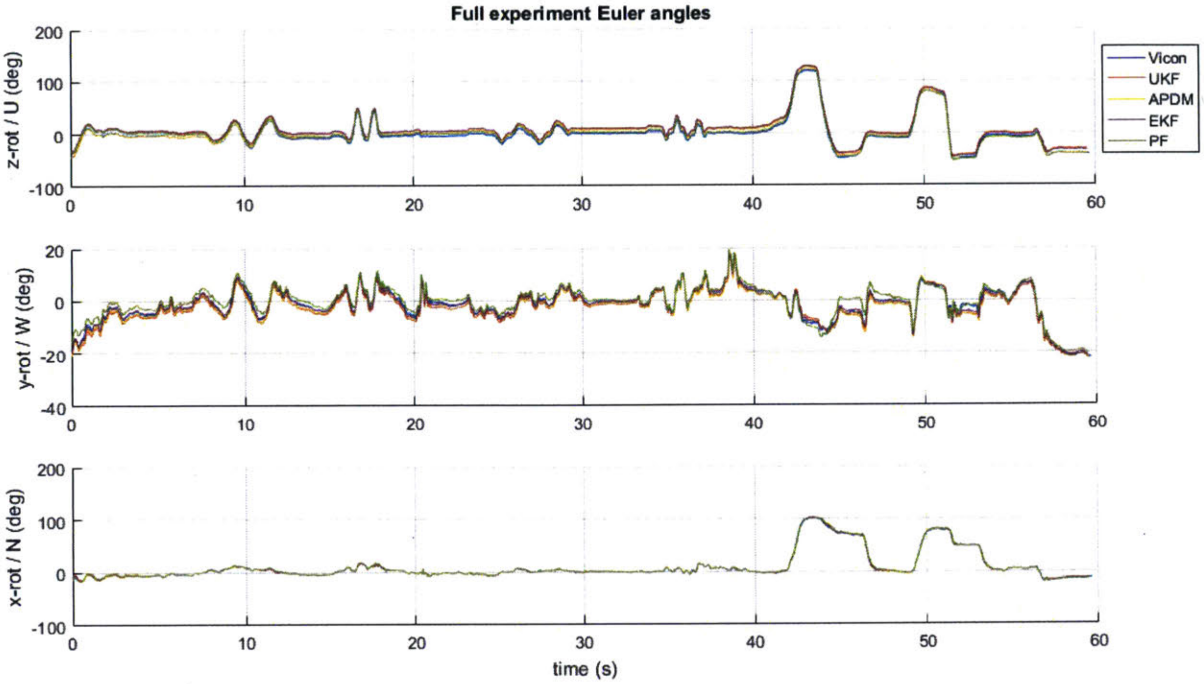


Figure 2.2 Truth (Vicon) vs. estimated orientations for entire experiment

performs slightly better than our EKF and UKF, when considering the total RMSE for the experiment, the difference is small (3 deg). The particle filter is seen to have a slightly smaller total error than the manufacturer's filter, but this difference may not be operationally relevant (0.41 deg).

A caveat to consider for operational usage is the computational complexity of the particle filter. Despite the higher accuracy, it takes far longer to compute. For this dataset, while the EKF and UKF were computed on the order of 3-5 seconds for the entire 60 second dataset, the PF took on the order of 80 seconds with 25 particles. In this context, the particle filter is an accurate filter which is inappropriate for online filtering in its current implementation, but would be appropriate for offline analysis.

A final desire of this experiment was to understand the motion domain for which each filter is most appropriate. Specifically, it was desired to understand how nonlinearities in motion affect filter performance. As discussed previously, the circle and square motion

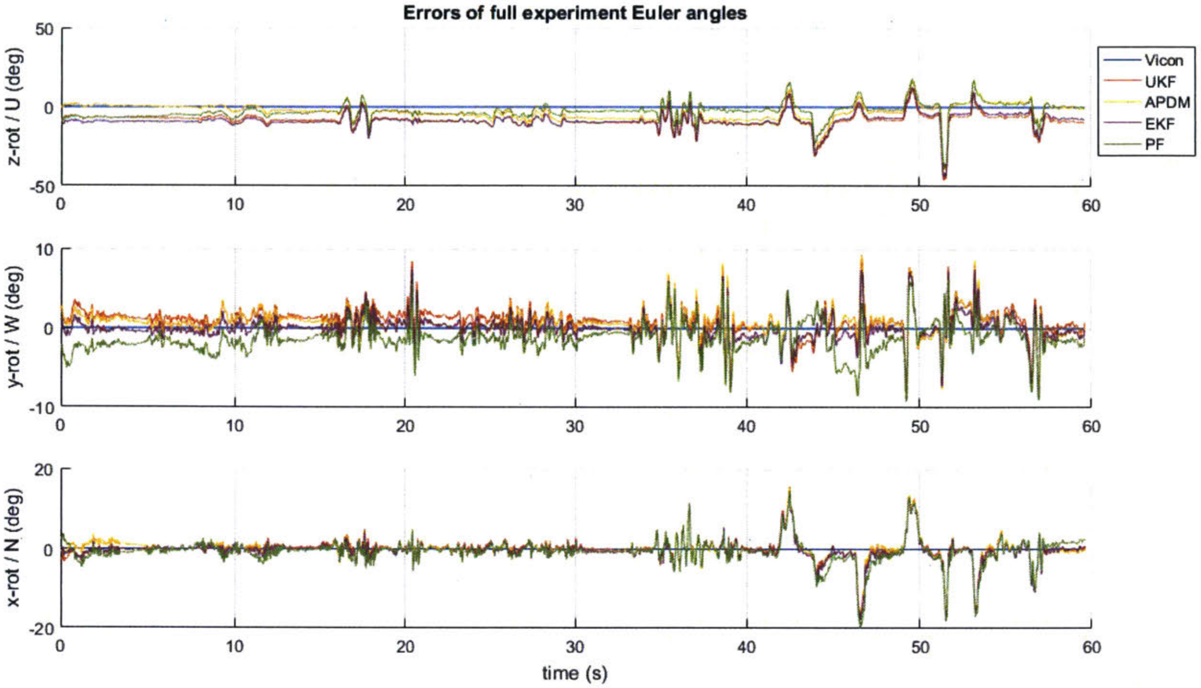


Figure 2.3 Error of estimated orientations for entire experiment

represent a simple nonlinear motion and a discrete, discontinuous motion.

As mentioned in Chapter 1, the EKF approximates nonlinear dynamics as a linearizion via the Taylor Series expansion. In the common case that a first-order expansion is used, one can intuit that this approximation is best suited for dynamics that are least nonlinear, i.e., most approximately linear. The omission of the higher-order Taylor Series expansion terms would lead to error in highly nonlinear dynamics, which cannot be appropriately captured by a linear approximation. The UKF, on the other hand, propagates a nonlinear map via a statistical transformation which preserves the lower-order moments of the distribution. Perhaps, it could be argued, the UKF would be inherently better suited for more nonlinear dynamics.

In the context of our experiment, both the circle and square motions were all in the horizontal plane, with limited out-of-plane rotations. It could thus be argued that the motions were not highly complex or nonlinear. The UKF seems to have performed slightly

Motion	EKF	UKF	PF	Manufacturer
Circle, slow	9.20	7.61	4.99	1.92
Circle, fast	8.27	7.63	3.26	2.81
Square, slow	9.16	9.23	3.80	6.11
Square, fast	9.32	9.84	2.59	7.87
Grasp A	8.45	9.33	4.73	5.70
Grasp B	4.65	6.23	3.08	3.21
Total experiment	10.25	10.22	6.93	7.34

Table 2.1 Filter RMSE (deg) for each motion tested

better in the circle motion, but not much different from the EKF in the square motion.

In the grasp motions, the second grasp was more robustly estimated by all filters. The second grasping motion was nearly identical to the first, but was completed about 10% faster. This could be due to the second grasp actually having more Euler angle constancy through the motion profile. The x-rotation in Figure 2.3 shows that the first grasp (about 42 seconds to 46 seconds) is smooth, whereas in the second grasp (49 seconds to 55 seconds), the Euler angle profile has some constant periods. This does not suggest the overall motion was less smooth, but rather that individual rotations about an axis during the motion were more discrete. The particle filter is shown to be more accurate than the EKF and UKF. As the particle filter is more computationally complex, this result makes sense. The particle filter performs better than the manufacturer for the square, but not for the circle motions. This could be due to the particle filter's resampling strategy. Around tight corners, particles which keep "going forward" might be quickly degenerated (i.e., the weight of these particles quickly becomes small).

In conclusion, all of the implemented filters performed within an error bound of a little over ten degrees. There is no operational difference between the performance of the EKF and UKF for human motion estimation. The particle filter is seen to perform better than the

EKF and UKF, and on the order of the manufacturer's proprietary filter. The particle filter has achieved our goal of performing on the order of the manufacturer's filter and enabling a open-algorithm opportunity for refining the model and understanding the sensitivity to parameters.

For future estimation work with the IMUs, the manufacturer's filter will be used when special calibrations or heuristics are not needed. In cases of tunings or re-calibrations being necessary, the offline particle filter will be used. It has been shown to perform as well as the manufacturer's filter, while also allowing freedom of tuning that the manufacturer's filter does not offer.

2.3 Evaluation of Particle Filter Performance

The particle filter, as a stochastic method, has a variety of tunable parameters which can affect accuracy and computational time. An experiment was devised to assess sensitivity to a subset of the parameters. This computational experiment uses the same dataset used in Section 2.2. The particle filter was run multiple times, while varying these tunable parameters.

2.3.1 Experimental Design

Factors

The performance of the particle filter is defined as the error between the estimate particle filter data and the true state. Low filter error is a high performance metric. It has been observed in anecdotal experience that the performance of the filter is sensitive to the input parameters. The three parameters (**factors**) which seem to have the highest impact are:

1. Accelerometer noise (variance)
2. Magnetometer noise (variance)
3. Number of particles in the simulation

The only major parameter not evaluated here is the initial state covariance $\mathbf{P}_0$. This is because the filter estimates this parameter iteratively. Choosing an inappropriate initial estimate of $\mathbf{P}$ can cause the filter to take more time to settle, but the initial state covariance does not, in general, affect filter orientation after a significant period of time.

The chosen levels of each factor were:

- Accelerometer Noise $= \begin{bmatrix} 1, 5, 10, 15 \end{bmatrix} m^2/s^4$
- Magnetometer Noise $= \begin{bmatrix} 10, 50, 100, 150 \end{bmatrix} \mu T^2$
- Number of Particles $= \begin{bmatrix} 5, 15, 25, 50 \end{bmatrix}$

Furthermore, $n = 5$ replicates were used at each treatment combination. These levels were chosen to investigate parameter levels around the visually-tuned parameters from Section 2.2. These original parameters were an accelerometer noise of 10 m^2/s^4, a magnetometer noise of 100 μT^2, and 20 particles.

Hypotheses

We assessed the following hypotheses:

1. H_1: Varying the accelerometer noise parameter will yield a significant effect on filter estimate
2. H_2: Varying the magnetometer noise parameter will yield a significant effect on filter estimate
3. H_3: Increasing the number of particles will yield a better filter estimate

Recall that the terms noise and variance may be used interchangeably–while it is generally called sensor "noise," it reflects the total uncertainty associated with that measurement, not just the physical sensor error.

Dependent Variable

The particle filter used in this study estimates orientation in space from noisy IMU accelerometer, gyroscope, and magnetometer data. Given that the IMU is used to track some kinematic motion we compare the estimate to the truth data just as in Section 2.2.

The response variable is shown in Equation 2.1. This is a single scalar response variable. Thus, this response variable does not capture the individual components of error about each axis; it provides a total error metric.

Procedure

The particle filter was computed over the total experiment dataset as described in Section 2.2.1 and for each simulation, the filter error was recorded. Order of the treatments are not important because this is a numerical simulation.

This experiment uses a full factorial design. In general, reduction techniques are desirable. However, this is a computational experiment. As such, we were only limited by computational resources and time. This makes a full factorial design much more feasible. This 4^3 factorial design with 5 replicates yields 320 combinations.

It took a little over 14 hours to compute all of these combinations and metrics. The major disadvantage of the particle filter when compared to deterministic methods like the Kalman filters is computational time. Further, computational time is increased with increasing numbers of particles.

2.3.2 Analysis

The data were found to not be normal (Appendix B), thus a Kruskal-Wallis test was used to assess the difference in treatments, and then pairwise comparisons were performed with a Scheffe correction to see differences between pairwise treatments. Interaction effects were found to be small, and are discussed in Appendix C.

2.3.3 Results and Discussion

Kruskal-Wallis Test for Main Effects

Effect of Accelerometer Noise A box plot filter error vs. accelerometer noise level is shown in Figure 2.4. The data support an effect of accelerometer noise (H=31.14, $p < 0.001$). Scheffe corrected confidence intervals (Table 2.2) suggest that levels 1 and 5 m^2/s^4 are not different ($p = 0.99$), but all other levels are significantly different.

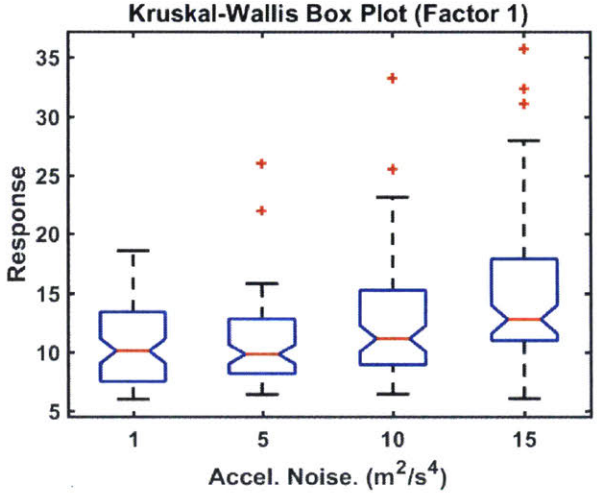


Figure 2.4 KW Box Plot for Factor 1

Accel. N.	Lower B.	Upper B.
1	10.03	11.42
5	9.97	11.37
10	12.09	13.49
15	14.11	15.51

Table 2.2 95% Response confidence interval for accelerometer noise

These results show an increase in error as the accelerometer noise parameter increases from 5 to 10 to 15 m^2/s^4. Operationally, this suggests that a value of 5 would yield better

Lvl 1 -	Lvl 2	LB	Mean	UB	p
1	5	-1.33	0.053	1.44	0.99
1	10	-3.45	-2.06	-0.67	$< .001$
1	15	-5.47	-4.09	-2.69	$< .001$
5	10	-3.5	-2.12	-0.73	$< .001$
5	15	-5.53	-4.14	-2.75	$< .001$
10	15	-3.41	-2.02	-0.63	0.001

Table 2.3 Pairwise comparison of treatments for factor 1

filter results than a higher parameter. A value of 5 is not significantly different than 1, however. Any value in this range should be suitable for good filter performance.

Theoretically, we know that choosing a value that is *too* small would increase filter error. This accelerometer noise parameter is supposed to capture the uncertainty in the measurement of the acceleration, which is larger than just the physical acceleration measurement variance. However, if this parameter were chosen to be too small such that it *underestimated* the true uncertainty, filter error would increase as a result of overtrusting the accelerometer measurement. This experiment, however, did not use small enough factor levels of the accelerometer variance to observe this phenomenon.

Effect of Magnetometer Noise A box plot of filter error vs. magnetometer factor level is shown in Figure 2.5. Unlike factors 1 and 3, magnetometer noise did not show a significant effect across treatments (Kruskal-Wallis H=4.303, $p = 0.231$). We conclude that magnetometer noise parameter, at least in the levels chosen, does not significantly impact the response of the particle filter.

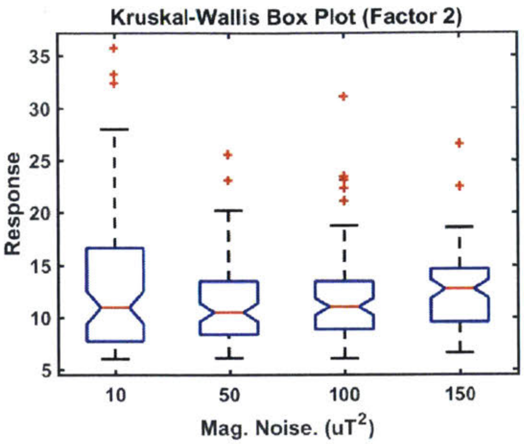


Figure 2.5 Treatment effect box plot of magnetometer noise

Operationally, a preferred level of magnetometer noise may not exist, as long as it remains in the reasonable range. The reasonable range is defined by these factor levels, keeping with the discussion earlier that a parameter that is *too* small can overtrust a sensor. Likewise, a factor level that is too high undertrusts a sensor, and leaves the filter prone to the drift of the IMU due to gyroscope integration with little sensor update.

Effect of Number of Particles A box plot showing filter error vs. number of particles level is shown in Figure 2.6. The number of particles showed a significant difference in response across factor levels (Kruskal-Wallis H=86.87, $p < 0.001$).

# Particles	Lower B.	Upper B.
5	16.07	17.48
15	11.10	12.50
25	9.96	11.36
50	9.05	10.45

Table 2.4 Pairwise comparison of treatments for factor 3

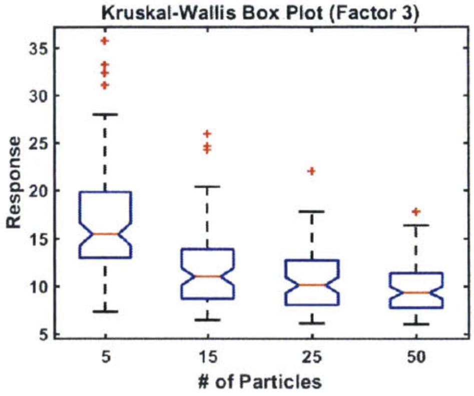


Figure 2.6 Number of particles treatment effect box plot

Lvl 1-	Lvl 2	LB	Mean	UB	p
5	15	3.59	4.97	6.36	$< .001$
5	25	4.73	6.12	7.51	$< .001$
5	50	5.64	7.03	8.41	$< .001$
15	25	-0.24	1.13	2.52	0.14
15	50	0.66	2.04	3.44	$< .001$
25	50	-0.48	0.91	2.29	0.33

Table 2.5 Pairwise comparison of treatments for factor 3

As expected, as the number of particles increases, the error of the filter decreases. Theoretically, we know that choosing a value for the number of particles that is too small would increase filter error significantly. This is because the particle filtering algorithm, as presented in Chapter 1, relies on the ability of multiple particles to approximately represent the posterior state distribution. One particle, for example, cannot represent the entire distribution. Expanding this logic, a small number of particles would do a very poor job of representing the distribution, and would thus increase filter error.

Theoretically, there is no such thing as too large of a value for the number of particles. An infinite number of particles could perfectly represent the posterior state distribution.

However, operationally, there is a practical limit. More particles requires larger computing resources. This computational limit motivates the scientist to seek a number of particles that is *good enough* to represent the posterior state distribution at the accuracy required. Here, the box plot in Figure 2.6 suggests a diminishing returns phenomenon. One can increase the number of particles in the simulation from 25 to 50, but receive no statistically significant benefit in accuracy. We operationally conclude that 25 particles is appropriate for this particle filter with the IMU sensor fusion problem.

2.3.4 Conclusion

In conclusion, it was found that both factors 1 and 3 had significant effects. While factor 2 did not yield a significant main effect by the Kruskal-Wallis test, a discussion counterpoint was offered to explain the lack of significance for the factor levels selected. The hypotheses may now be revisited with new insights:

1. H_1: Varying the accelerometer noise parameter will yield a significant effect on filter estimate

   The data support an effect of accelerometer noise on filter performance. Increasing the accelerometer noise was found to increase the estimation error. Operationally, we conclude that using a range of 1-5 m^2/s^4 would be appropriate.

2. H_2: Varying the magnetometer noise parameter will yield a significant effect on filter estimate

   The data do not support that varying the magnetometer noise between 10 and 150 μT^2 affected the estimation error. Operationally we conclude any value in this range is appropriate.

3. H_3: Increasing the number of particles will yield a better filter estimate

   The data support an effect of number of particles on filter performance. Increasing the number of particles decreased filter error. Operationally, we conclude that 25 particles is appropriate.

We discussed why there are theoretical values which could be too high and too low for each parameter. However, these theoretical limits are not operationally relevant. This study provides operationally-relevant evaluation of particle filter input parameters.

The study did have some limitations. Namely, these results are limited in scope to the motions selected. Further examination would be required for other types of dynamic motions and appropriate parameter selection for those motion domains. Here, most motion was in the horizontal plane with two grasping tasks at the end. The study also showed no significant difference between magnetometer noise factor levels, which could indicate an inappropriate choice in levels.

Furthermore, if the data was taken in a magnetically-foreign environment, we would tend to trust the magnetometer less and as a result increase associated sensor variance in the filter. These types of environments arise when there are changing magnetic fields or a different field for which the sensor has not been properly calibrated.

Chapter 3

Effect of Motion Profile on Joint Axis Error

As motivated in Chapter 1, we are evaluating a system which uses inertial sensors to measure human joint angles. Unfortunately, as discussed, inertial sensors suffer from needing precise positioning and/or calibration motions. A technique to overcome these issues would drastically improve the viability of inertial sensors for human joint angle measurement in real-world environments.

An elegant method was recently proposed by Müller et. al. [3] and presented in Section 1.4. This method calculates the joint axis from the individual IMU data. However, it is unclear how the input data to Müller's method affects the quality of joint axis estimation. In this chapter, Müller's algorithm will be evaluated against different motion profiles to better understand what types of motions lead to better or worse joint axis/angle estimation.

3.1 Introduction and Motivation

Müller's method, as introduced in Chapter 1, provides a framework to estimate robust anatomical axes (and thus joint angles) in a gamut of domains. One application of particular interest is that of occupational therapy. Currently, progress in rehabilitation is evaluated by experts visually observing patient motion [49]. Electronic systems which provide data feedback to the clinician may be beneficial, as long as said systems do not reduce situation awareness of the clinician. Joint angles of the subject can currently be found using motion capture systems in some research labs, but this technology is typically not used by clinicians. Furthermore, marker-based motion capture systems have no potential as an at-home system for the patient. Small IMU-based systems have potential as a robust human joint motion measurement system. Müller's method may provide a solution to the problem of calibration.

Occupational therapy, as a field, seeks to rehabilitate patients activities of daily living (ADLs) which include brushing ones teeth, or bathing oneself. These ADLs will be evaluated as an input to the Müller algorithm. If ADLs cannot provide robust joint axis estimators in the Müller framework, the Müller method may not be an appropriate tool to enable on-line calibration in clinical settings. Although, it could still be used with a set of pre-defined calibration motions that are repeated at set intervals. In this manner you could obtain an initial estimate of the axes and correct for drift over time.

In the experiment described in the following sections, a robotic arm was used to evaluate both ADL motions and control motions as inputs to the Müller method. We hypothesized that motions which have higher frequency and amplitude characteristics would produce lower-error axis estimations when using the Müller method. These hypotheses can be found in Section 3.3.1. The dependent measures (Section 3.3.2), the two joint axis estimations of the elbow, will be evaluated in a formal statistical analysis.

3.2 Experimental Method

To evaluate these questions, a robot was developed to emulate the human arm. Using this robotic test bed, specific motion profiles may be commanded. The anatomical axes of the robot are built-in, and may easily be compared to the estimated anatomical axes of the aforementioned algorithm.

3.2.1 Robot

The robot is meant to emulate the human upper arm, elbow joint, and forearm. The two degrees of freedom of the elbow, flexion/extension and pronation/supination, are achieved through commercial servos. In particular, the flexion/extension servo is a Dynamixel RX-24F (Robotis Ltd. Lake Forest, CA). The pronation/supination motion servo is the Dynamixel AX-12A. These servos are very fundamentally similar. The RX-24F has a bit higher torque ceiling, which is necessary for the flexion/extension servo. The rotational inertia of the entire lower arm about its base must be actuated by this servo, which is more inertia than just the rotational inertia of the lower arm about its longitudinal axis. The flexion/extensions servo (RX-24F) has a range of motion of 300 degrees, although it is constrained to 170 degrees for the purpose of this experiment. Likewise, the pronation/supination servo (AX-12A) has a range of motion of 360 degrees (not rotationally continuous), but is constrained to 180 degrees for the purpose of this experiment. The constraints are simply to mimic the human range of motion. Specifications of the Dynamixel servos can be found in Appendix F.

Figure 3.1 shows the robotic test bed developed for this test. The robot was fixed to the table by clamping the upper arm to the table surface. In this configuration, the upper arm is fixed in space (and hence the IMU on the upper arm is also still), while the lower arm is allowed to move freely. The two servos are powered by a 12V power supply and commanded from MATLAB using the Dynamixel SDK for MATLAB. The robot is turned on its side so that the flexion/extension plane is actually horizontal rather than vertical (as it normally is for a human). This is simply to remove the effect of gravity acting against the flexion/extension servo. In this horizontal configuration, the flexion/extension motions are smoother. Since Müller's algorithm is only a function of the *relative* orientation between the two IMUs, the human arm (or a robotic simulator) can be oriented in any plane.

A major advantage of the robot are that its anatomical axes are built-in. As the IMUs are placed on the robot, the true anatomical axes are already known.

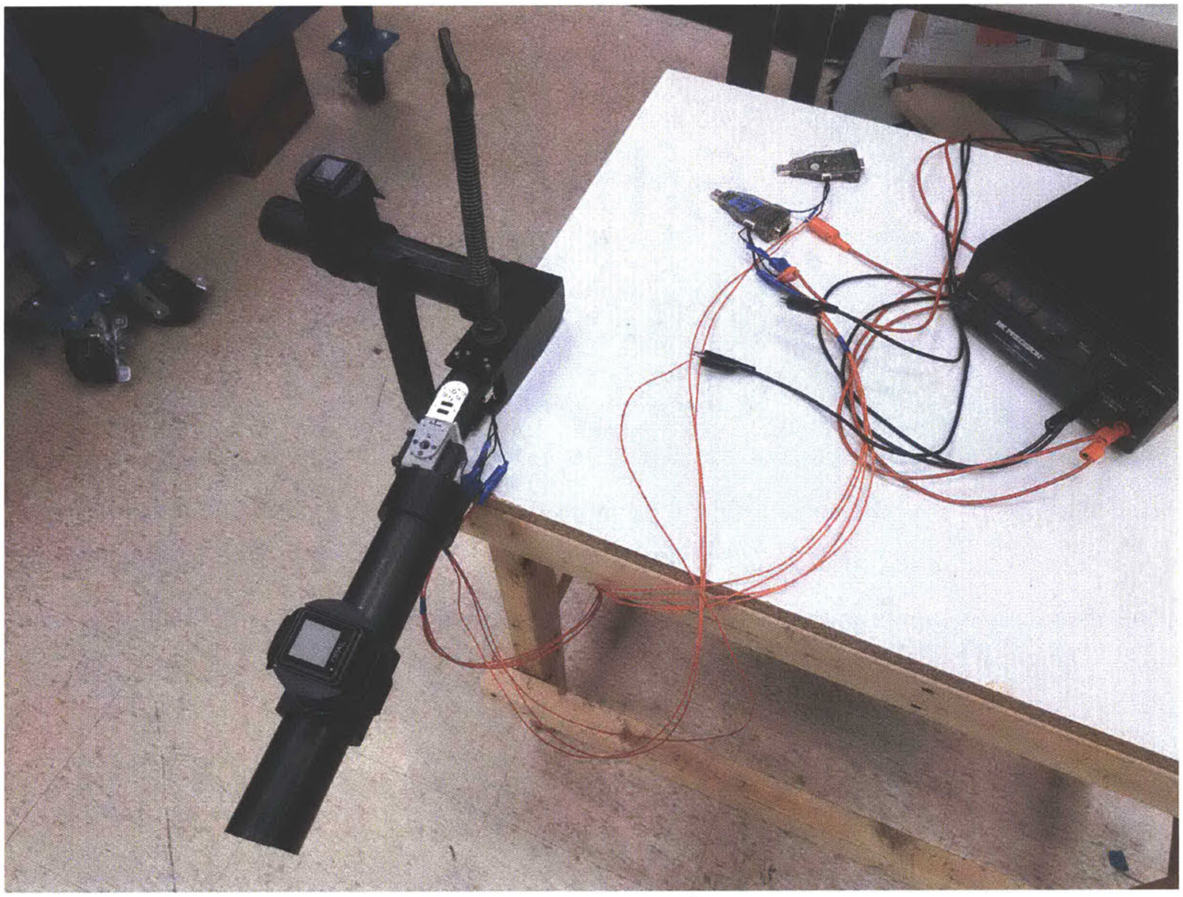


Figure 3.1 Robotic test bed, meant to simulate the human elbow

3.2.2 Inertial Measurement Units

The same APDM Opal IMUs used previously are used in this experiment. They can be seen in Figure 1.1. The IMUs are turned on prior to the long experiment, mounted, and record all motions in one dataset. The IMUs are calibrated for the magnetic environment that they are used in beforehand, to get the best filtered estimate possible. The calibration procedure involves spinning the IMU for multiple seconds to cover a full sphere of magnetometer measurements. This cloud of data should be spherical and centered at the origin. In practice, this measurement cloud needs to be circularized via a scale factor matrix. The offset from the origin is the static bias of the magnetometer, and can be subtracted away from measurements. The manufacturer's Unscented Kalman Filter was used to compute orientation in space.

3.2.3 Motions

Both ADL-type motions and control motions were completed using the robotic test bed.

ADL Motions

Motion data of the ADLs were taken from the Carnegie-Melon motion capture database (`mocap.cs.cmu.edu`), which computed joint angles from motion capture data. An example of one of these ADLs, brushing ones teeth, is shown in Figure 3.2. A catalog of all ADL motions can be found in Appendix D.

The ADLs were chosen by searching the CMU database for all available motions which: (a) provided multiple seconds of clean, usable data of the right arm, (b) would be considered common to a normal task of recreation, physical work, or self-care.

The ADLs chosen were: drinking soda, buttoning a front-button shirt, swimming, swinging a golf club, hammering a nail, dribbling a basketball, pitching/throwing a baseball, brushing ones teeth, coiling a rope, washing ones self, running, sweeping the floor with a broom, and walking.

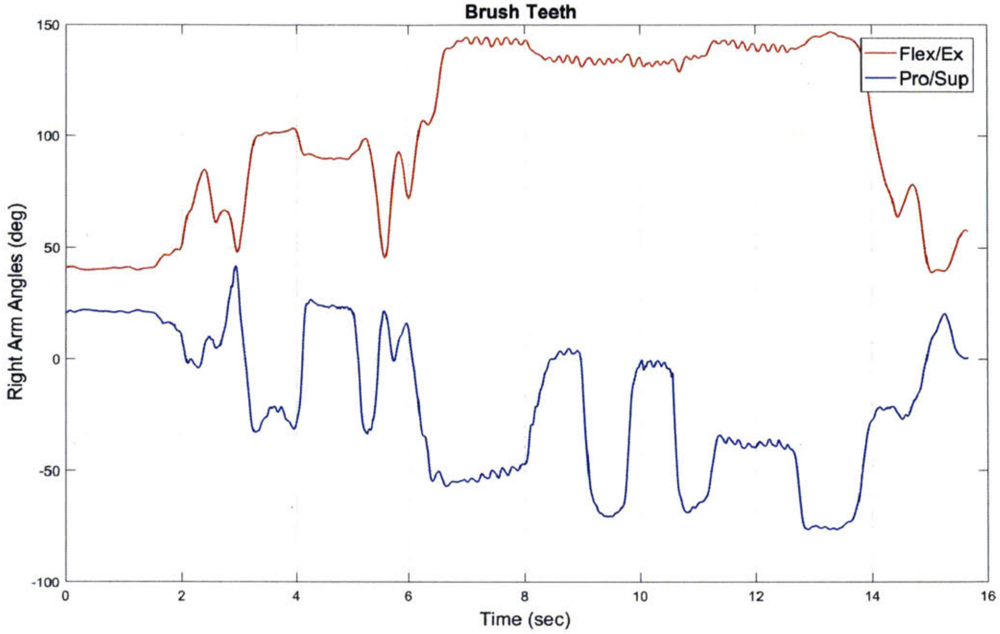


Figure 3.2 Elbow joint angles for brushing teeth in a right-handed subject

Control Motions

As a control, sinusoidal motions in flexion/extension and pronation/supination were performed. These sine waves were parameterized by amplitude and frequency. Each motion was precisely 6 seconds long (which corresponds to one period for the low frequency level and two periods for the high frequency factor level). Phase difference between the flexion/extension sine wave and the pronation/supination sine wave was not considered.

The chosen levels for amplitude were 10 and 65 degrees. The chosen levels for frequency were 60 and 120 degrees per second. As there are two motions to be conducted at a time (flexion/extension and pronation/supination), this yields $2^4 = 16$ treatment combinations. The levels chosen for amplitude and frequency of the sine waves were motivated by range of motion for amplitude factor levels and average velocity for frequency levels. The chosen levels of 10 and 65 degrees for amplitude yields a total range of motion of 20 and 130 degrees, respectively. These two values are shown in red dash lines in Figure 3.3. As can be seen,

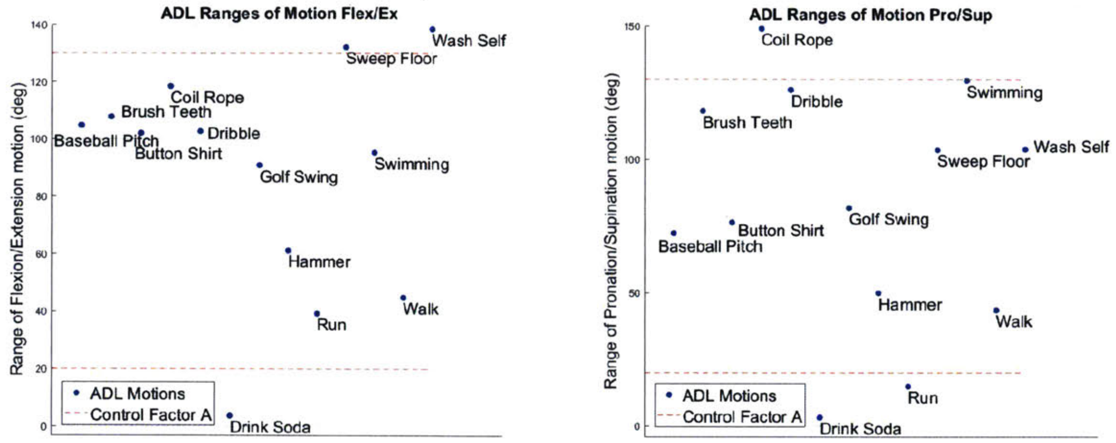


Figure 3.3 ADL range of motion (blue dots) and two control amplitude factor levels (red dotted line) for flexion/extension and pronation/supination (left and right, respectively)

these values do a fair job of describing the maximum and minimum functional ranges of motion of the ADLs. It was not desired to capture the extremely small range of motion of drinking soda, as this range was not representative of the typical ranges of the dataset.

The frequency levels of 60 and 120 degrees per second were chosen to produce sine waves which have average velocities similar to the ADLs. Figure 3.4 shows the average absolute velocities of the ADLs in blue dots and the control motions in red dots. Just as in the case of range of motion, the control motions nearly cover the range of average velocities as well.

Figure 3.5 shows an example motion plot for one of the 16 treatments. The sine wave for flexion/extension motion is centered about 90 degrees, which is the center of the range of motion for flexion/extension. The pronation/supination sine wave is centered about 0 degrees, which is the center of the range of motion in pronation/supination.

Between each six second motion, a five-second pause period is held. This is done for post-processing purposes. The pattern of a six-second motion followed by a five-second pause is repeated for all motions conducted.

All 16 control motions conducted by the robotic test bed are shown in Appendix E.

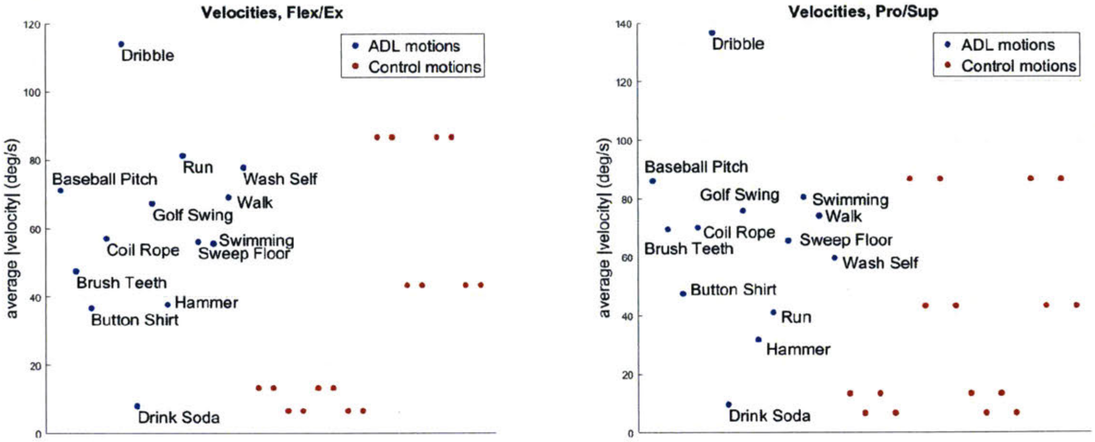


Figure 3.4 Mean velocity for ADL motions (blue dots) and control motions (red dots) for flexion/extension and pronation/supination (left and right, respectively)

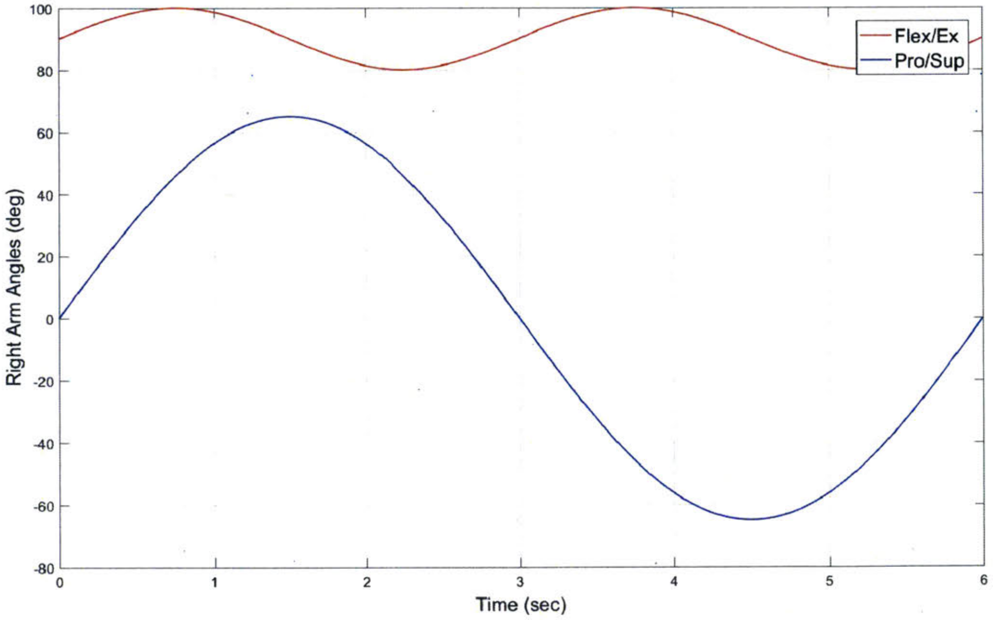


Figure 3.5 Example motion with $A_1 = 10deg$,$A_2 = 65deg$,$F_1 = 120deg/s$,$F_2 = 60deg/s$

3.3 Experimental Design and Analysis

The factors, measures, and hypotheses evaluated are presented below. Whenever motion is described hereafter as *high* motion, this will be interpreted to mean large amplitude and high frequency motion. Likewise, low motion would mean low amplitude and low frequency motion.

3.3.1 Hypotheses

While Müller's method is useful for online calibration, there is a gap in understanding which IMU-measured motions beget good anatomical axes measurements.

1. H_1: Higher amplitude (both DOFs) elbow motion will yield a lower-error estimate of both anatomical axes
2. H_2: Higher frequency (both DOFs) elbow motion will yield a lower-error estimate of both anatomical axes
3. H_3: Higher amplitude (both DOFs) elbow motion will yield a lower-variance estimate of both anatomical axes
4. H_4: Higher frequency (both DOFs) elbow motion will yield a lower-variance estimate of both anatomical axes

These hypotheses are meant to mean that *both* axes can be estimated more robustly by making motions about *both* axes higher in frequency and amplitude. This is motivated by the sensors–we know that there is some lower threshold of motion amplitude and frequency below which the IMU cannot distinguish movement. At this low level, effective motion would be indistinguishable from noise in the gyroscopes. It is also presumed that feeding noisy data into Müller's algorithm would not produce good estimation. So, we take the logical inverse to hypothesize that more motion would lead to better estimation results.

Continuing this signal-to-noise ratio argument, we hypothesize that not only is accuracy improved by increasing the amplitude and frequency of motion, but precision as well.

Formally, we state that the variance of the joint axis estimates will decrease as a result of increased amplitude and frequency of motion.

3.3.2 Dependent Variables

The dependent variables measured were the *error* of the estimated axis via Müller's algorithm when compared to the true axis. This error is computed as:

$$error = atan2(|axis_{true} \times axis_{est}|, axis_{true} \cdot axis_{est}) \tag{3.1}$$

Equation 3.1 will find the rotation angle between two axes (in this case, the true and estimated axes), as the smallest angle between the two vectors. More precisely, the plane this angle lies in is the plane orthogonal to the two axes. i.e., the plane defined by the cross product between the two axes.

There are *two* dependent measures here.

1. error of $\vec{\mathbf{a}}$: the error of the estimated flexion/extension axis
2. error of $\vec{\mathbf{b}}$: the error of the estimated pronation/supination axis

These two dependent variables will have a minimum of zero and a maximum of 90 degrees. Zero degrees would imply that the estimate is perfectly aligned with the true anatomical axis. 90 degrees would be the maximum error–implying that the estimated axis is perpendicular to the true axis. The directionality of the estimated axis is irrelevant–an axis and its negative are parallel and thus equivalent axes as a representation for the anatomical joint axis.

For each trial conducted, the following steps are used to calculate the two dependent measures:

1. Calculate relative orientation of the two IMUs: $\mathbf{q}_B^A = (\mathbf{q}_{NWU}^B)^{-1} \cdot \mathbf{q}_{NWU}^A$
2. Calculate relative angular velocity of the two IMUs, $\vec{\omega}_{rel}^{\mathbf{A}}$

3. Use modified Müller method (section 1.4) to compute anatomical axes $\vec{a}$ and $\vec{b}$. Repeat 5 times to avoid local minima as discussed in Section 3.3.3.

4. Compute error between estimated $\vec{a}$ and $\vec{b}$ from step 3 and true values $\vec{a} = [0, 0, 1]$ and $\vec{b} = [1, 0, 0]$ using equation 3.1 for each estimate from step 3. Select lowest error, as discussed in Section 3.3.3.

The true values $\vec{a} = [0, 0, 1]$ and $\vec{b} = [1, 0, 0]$ come from the built-in robot axes relative to the IMU.

3.3.3 Optimization Method

As stated previously, for each individual motion k, the two anatomical axes $\vec{a}^k$ and $\vec{b}^k$ can be computed and compared to the true values $\vec{a} = [0, 0, 1]$ and $\vec{b} = [1, 0, 0]$. A Levenberg-Marquardt optimization algorithm was implemented on the nonlinear objective function in Equation 1.24.

By nature, nonlinear objective functions can include multiple local minima. Overcoming the pitfalls of being "caught" in a local minima is an area of active research and a defining motivator of many optimization techniques. Due to the existence of local minima, the solution of an optimization method can be sensitive to the initial conditions which define where the solution search begins.

This problem was observed in solving for the anatomical axes in this experiment. In fact, a local minima existed at the 90 degree error solution, so many outliers at 90 degrees error existed. This is merely due to the surface of the object function, Equation 1.24. To overcome this, for each motion, the Levenberg-Marquardt solver was run 5 times, each time with random initial conditions. The best solution from the five runs was taken for each motion analyzed.

This method is reasonable because the local minima, in general, produced solution which were very high error and easily identifiable as outliers, rather than physical solutions. This is because the local minima "traps" in the solution surface were generally far away from the true physical solution at the global minima. In practice, it would be easy to throw away these non-physical solutions.

3.3.4 Test Matrix/Test Order

Table 3.1 lists all 16 treatments (numbered zero to fifteen) and the corresponding levels of each of the four factors that are associated with each treatment. Here, L is meant to mean low (10° amplitude or 60° /s frequency), and H is meant to mean high (65° amplitude or 120° /s frequency). Remember that each of the four factors only have two levels. For experimental design purposes, these levels are considered to be categorical and binary, hence their representation in the table.

Each treatment was repeated 100 times. The order was a repeated mirror design. Specifically, this was treatment 0 to treatment 15, then mirror to do treatment 15 to treatment 0. Then this block of 32 trials was repeated 50 times. Or,

$$\texttt{treatment } 0, 1, ...14, 15, 15, 14...1, 0 \texttt{ (repeat 50 times)} \tag{3.2}$$

The mirrored 50 block segments was used to balance out any ordering effects that could arise from simply repeating treatment 0 to 15 in the same order 100 times. Note that while this removes an ordering effect, IMU drift over time is a different effect which will be commented on later.

3.3.5 Statistical Approach

The factors were evaluated for main effects on mean via two-sample Wilcoxin Rank Sum tests on each (binary) factor level. Main effects of each factor on variance were evaluated bootstrapping the interquartile range (IQR) of a sample 1000 times with replacement. The new bootstrapped IQR distributions were compared using a Wilcoxin Rank Sum test.

The responses of the 16 individual treatments were found to be non-normal. These treatment responses were compared for differences in mean via a Kruskal-Wallis omnibus

Table 3.1 Treatment combinations with corresponding factor levels; A_1 is amplitude of flexion/extension, A_2 is amplitude of pronation/supination, F_1 is frequency of flexion/extension, F_2 is frequency of pronation/supination

Treatment	A_1	A_2	F_1	F_2
0	L	L	L	L
1	H	L	L	L
2	L	H	L	L
3	H	H	L	L
4	L	L	H	L
5	H	L	H	L
6	L	H	H	L
7	H	H	H	L
8	L	L	L	H
9	H	L	L	H
10	L	H	L	H
11	H	H	L	H
12	L	L	H	H
13	H	L	H	H
14	L	H	H	H
15	H	H	H	H

test with multiple comparisons corrected via the Dwass-Steel-Critchlow-Flinger method.

Interactions were tested by constructing groups that pool two factors (as opposed to three factors as main effect tests do). A Kruskal-Wallis omnibus test was performed on each pooled group to show significantly different subgroups. The data were found to be non-normally distributed. Pairwise medians were compared via a Wilcoxin Rank Sum test. Pairwise IQR of subgroups was estimated by bootstrapping. The boostrapped distributions of IQR were compared via a Wilcox Rank Sum test, similar to the evaluation of main effects on IQR. When interpreting effect sizes, we use Cohen's convention of effect sizes of 0.3, 0.5, and 0.8 to represent small, medium, and large effect sizes, respectively [50].

3.4 Results and Discussion

All statistical tests were run in SYSTAT (SYSTAT Software Inc.) and MATLAB (The Mathworks, Inc., Natick, Massachusetts). The results of the ADL motions will be presented first, followed by the results of the control motions.

3.4.1 ADL Motion Results

Figure 3.6 shows the axis estimation error of each anatomical axis. Intuition suggests that the lower-error motions are motions which are more varied in their profiles. Drinking soda does not require much elbow motion (range of motion: 4 degrees F/E, 3 degrees P/S), the shoulder generates most of movement. Likewise, with walking (range of motion: 10 degrees F/E, 8 degrees P/S) the elbow is typically more static and motion of the arm is due to flexion/extension of the shoulder. In contrast, some motions are very dynamic at the elbow. Washing ones self shows a range of motion of 130 degrees F/E and 70 degrees in P/S.

The control motion results are presented to better understand the effects of amplitude and frequency of motion as an effect on axis estimation error.

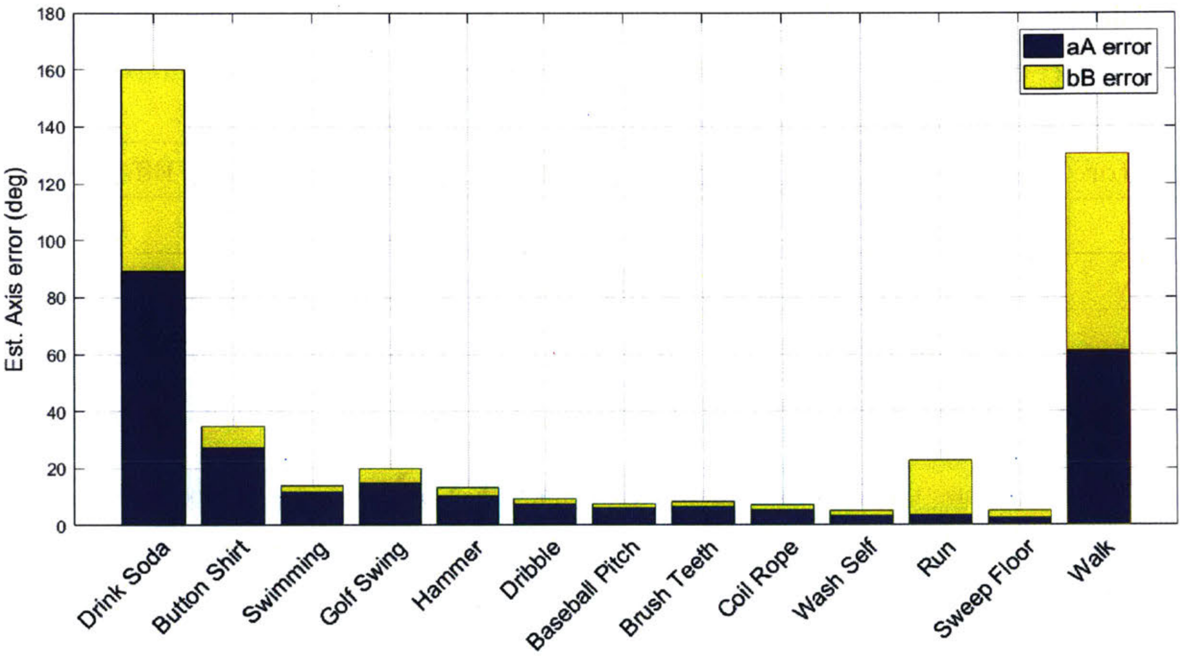


Figure 3.6 Results of axis error (in degrees) for each of the ADLs evaluated. Yellow represents pronation/supination axis (axis bB) error, and blue represents flexion/extension axis (axis aA) error

3.4.2 Control Motion Results

Section 3.2 designed an experiment to test four factors on the two axis estimations. First, the main effects of the four factors will be presented, followed by the treatment effects.

Main effects: error of flexion/extension axis estimation

The hypotheses in 3.3.1 are evaluated as main effects in the Wilcoxin Rank Sum test, because the data were found to be non-normal and possibly multi-modal for a subset of pooled groups.

Table 3.2 Medians (in degrees) of the four main factors on flexion/extension axis error

Factor	**Low Level Median**	**High Level Median**	**p-value**	**z-stat**	**Effect size**
A_1	4.62	3.11	$< .001$	6.40	0.23
A_2	2.73	5.62	$< .001$	20.37	0.72
F_1	2.87	3.47	$< .001$	5.13	0.18
F_2	3.20	3.50	0.041	2.04	0.07

Table 3.2 shows the four main factors and their effect on flexion/extension axis (axis 1) estimation. These results and the presence of a multi-modal distribution suggest interaction effects are present. A modified hypothesis was developed post-hoc. High amplitude and high frequency motion yields lower error estimation *on the axis of high amplitude and frequency.* High amplitude and high frequency motion on the off-axis *increases* error on the axis of interest.

Main effects: variance of flexion/extension axis estimation

Not only did we hypothesize about the *accuracy* of flexion/extension axis estimation (hypotheses 1 and 2), but about its *precision* also (hypotheses 3 and 4).

Table 3.3 shows the interquartile range (IQR) of the main four effects on flexion/extension axis estimation. Similar to our previous conclusion, we find that main effects support the new

Table 3.3 IQRs (in degrees) of the four main factors on flexion/extension axis error

Factor	Low Level IQR	High Level IQR	p-value	z-stat	Effect size
A_1	15.57	1.21	$< .001$	38.72	1.22
A_2	1.74	13.27	$< .001$	38.72	1.22
F_1	4.54	3.62	$< .001$	37.45	1.18
F_2	2.04	6.65	$< .001$	38.72	1.22

post-hoc hypothesis. This needs to be explored further with an interaction effects analysis.

Main effects: error of pronation/supination axis estimation

Similar to the flexion/extension main effect analysis, the medians of the factor levels were evaluated using a Wilcoxin Rank Sum test.

Table 3.4 Medians (in degrees) of the four main factors on pronation/supination axis error

Factor	Low Level Median	High Level Median	p-value	z-stat	Effect size
A_1	2.35	3.90	$< .001$	13.14	0.46
A_2	7.64	2.31	$< .001$	23.43	0.83
F_1	2.34	4.12	$< .001$	10.83	0.38
F_2	3.20	2.39	$< .001$	5.44	0.19

Table 3.4 shows the four main factors and their effect on pronation/supination axis (axis 2) estimation. These results echo the main effects on accuracy of flexion/extension with slightly stronger effects. The pronation/supination axis estimation is most sensitive to the amplitude of on-axis motion, with an effect size of 0.83 between the low and high factor medians of the on-axis amplitude (A_2). Similar to flexion/extension, the error is least sensitive to factor F_2. This is interesting because F_2 is an on-axis factor in pronation/supination, but an off-axis factor in flexion/extension.

Our understanding, as informed by main effects, is that whether the factor increases or decreases accuracy is related to if the factor is on the axis of measurement or not on the axis of measurement. However, the effect sizes between factor level medians are not related to an on-axis vs. off-axis notion. For both flexion/extension and pronation/supination accuracy, A_2 has the largest effect size (0.72 and 0.83, respectively). Likewise, F_2 has the smallest effect size (0.07 and 0.19, respectively).

Main effects: variance of pronation/supination axis estimation

Table 3.5 IQRs (in degrees) of the four main factors on pronation/supination axis error

Factor	Low Level IQR	High Level IQR	p-value	z-stat	Effect size
A_1	4.18	8.34	$< .001$	38.70	1.22
A_2	14.02	0.93	$< .001$	38.72	1.22
F_1	2.72	8.15	$< .001$	38.72	1.22
F_2	8.25	3.40	$< .001$	38.72	1.22

The results of comparing bootstrapped distributions of IQR are shown in Table 3.5. The effects of the factors on the variance (i.e, precision) of estimation of the pronation/supination axis follows identically to the conclusion from the flexion/extension axis results. Having high on-axis frequency and amplitude decreases variance of the axis estimation, leading to a more precise estimate. Conversely, having high off-axis frequency and amplitude yields a less precise estimation.

3.4.3 Flexion/Extension Axis Treatment Effects

While the main effects are important and shed light onto our original hypotheses, the distributions of the pooled main effects suggest that interaction effects may be present. Treatments were found to be statistically different (H=964.29, $p < 0.001$). Pairwise comparisons found almost all groups were significantly different. For ease, Appendix H shows the pairwise treatment comparisons along with corresponding statistic and p-value.

Figure 3.7 shows the effect of individual treatments on the flexion/extension axis estimation error. Some notable points arise with this plot. For example, when $A_1 = 1$, we see a much smaller variance estimate as compared to $A_1 = 0$. According the post-hoc hypothesis, high on-axis amplitude should decrease variance.

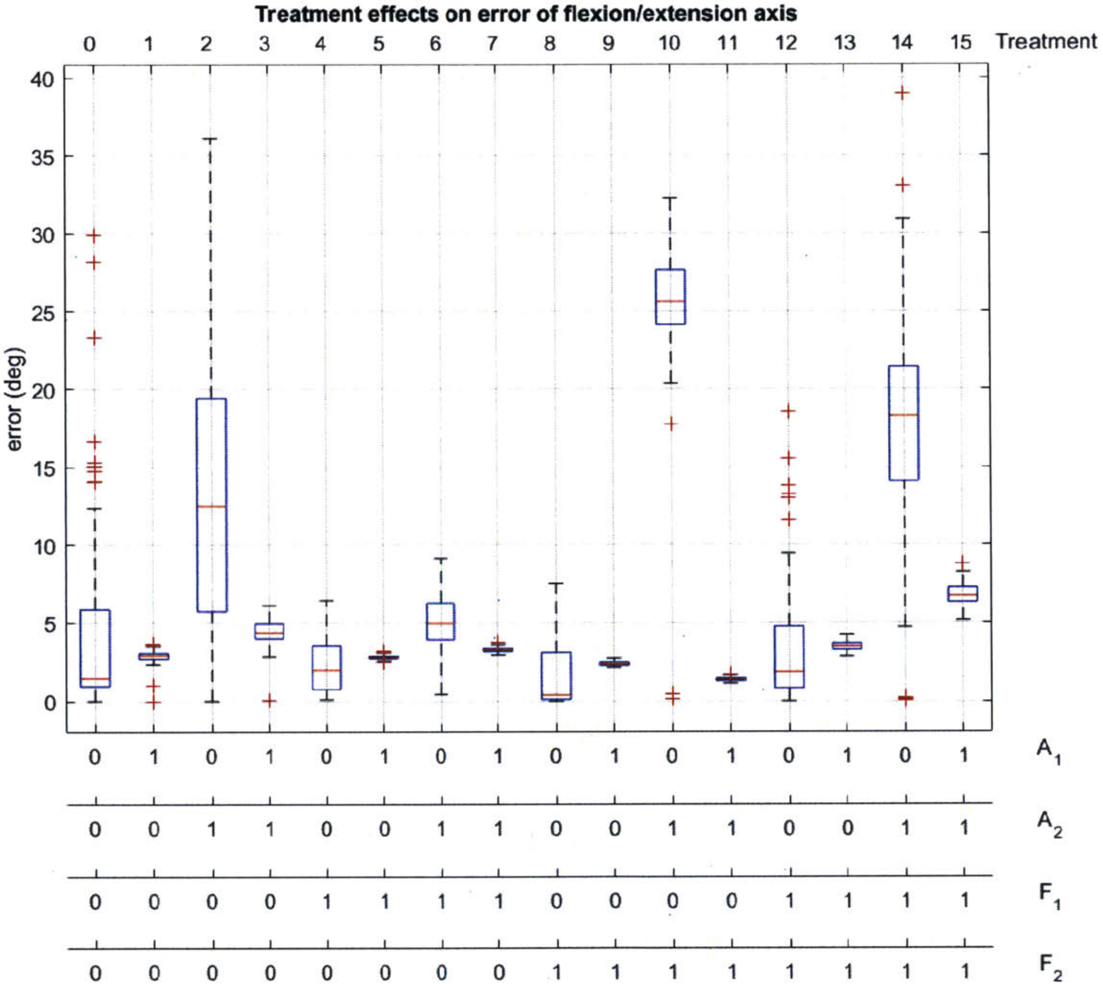


Figure 3.7 Flexion/extension axis error vs. individual treatment 0 to 15

This treatment plot, however, shows an interesting result with the both-low amplitude combination ($A_1 = 0$ and $A_2 = 0$ across all four combinations of frequency). The group of 4 treatments when both amplitudes are low appears to show higher variance than the general $A_1 = 1$ condition, but it seems to have a lower median in general. Individually, these treatments are treatments 0, 4, 8, and 12. This groups naturally as Appendix H shows that treatment 0 is not significantly different from 4 and 12, and treatment 4 is not significantly different from 12. This suggests that while the specific low amplitudes combination may have a greater variance, it in general gives a small median error (less than 5 degrees).

Conversely, treatments 2, 6, 10, and 14 represent the amplitude combination we hypothesized should give us the highest-median estimates: the combination where the on-axis amplitude is low but the off-axis amplitude is high ($A_1 = 0$ and $A_2 = 1$). Unlike the previous group, there is a visually observable effect across frequencies. In order to these results and possible interactions, we further analyze the data.

The main effects reported in section 3.4.2 inform our original hypotheses, but can hide interaction effects due to the pooling of many factors. When a main effect was previously evaluated via a two-sample t-test, the other 3 factors were pooled. For example, when the two levels of A_1 were compared via the test, the effects of factors A_2, F_1, and F_2 were all pooled. This analysis investigates two-way interactions by considering a *combination* of two factors. For four factors, there are six combinations of possible treatments to hold constant (i.e., to pool, as opposed to pooling three factors like previously). Kruskal-Wallis test shows that all 6 subgroups have significantly different medians and IQRs (all $p < 0.001$).

Flexion/extension axis estimation accuracy

In any of the six two-factor pooling groups, the four individual corresponding subsets were evaluated for support of the post-hoc hypotheses. Appendix J shows the six possible two-factor *pool groups*. Each pool group has four subsets (the four possible combinations of the remaining two unpooled, binary factors). The table, for each two-factor subset, shows the corresponding median and interquartile range (IQR) of the sample.

Our revised hypotheses suggest that on-axis high frequency and amplitude decreases error and error variance, while off-axis high frequency and amplitude increase error and error variance. Remember that error and error variance are analogous to accuracy and

Table 3.6 Comparison of highest and lowest median subgroups for each pool group in flexion/extension; associated subgroup is shown in parentheses

Pool Group	Highest Median	Lowest Median	p-value	z-stat	Effect
1	16.54 (1.3)	1.60 (1.1)	< 0.001	18.64	0.59
2	5.88 (2.1)	2.56 (2.2)	< 0.001	6.07	0.19
3	6.09 (3.4)	2.41 (3.1)	< 0.001	18.65	0.59
4	7.78 (4.3)	2.80 (4.4)	< 0.001	6.73	0.21
5	7.61 (5.4)	2.49 (5.3)	< 0.001	13.82	0.44
6	5.87 (6.4)	2.30 (6.3)	< 0.001	7.77	0.25

precision.

Between each subset of a pool group, a pairwise comparison is performed to test for difference in medians. The resulting six pairwise comparisons per group are completed using a Wilcoxin Rank Sum test. The results of the pairwise comparisons are shown in Appendix L, along with the effect size. This normalized effect size is calculated as $r = z/\sqrt{N}$, where N is the number of samples. Table 3.6 shows a condensed version of the appendix which shows the lowest and highest median subgroups within each pooling group.

For each pooling group, our hypotheses suggest that the lowest-median error would occur when any on-axis amplitude and frequency factors are high, and any off-axis amplitude and frequency factors are low. Likewise, the highest-median error should occur when any off-axis amplitude and frequency factors are high, and any on-axis amplitude and frequency factors are low.

Pool groups 1 through 5 offer mixed support for the post-hoc hypotheses. The groups do not support the hypotheses at expected low-median conditions, although the effect sizes between the expected lowest median and observed lowest median (0.23, 0.42, 0.21, 0.11, and 0.03 for groups 1-5 respectively) are very small. Group 5, in fact, is not statistically significant ($p = 0.361$). Group 1 through 5 do, however, support the high-median condition hypothesis.

Table 3.7 Comparison of highest and lowest IQR subgroups for each pool group in flexion/extension; associated subgroup is shown in parentheses

Pool Group	Highest IQR	Lowest IQR	p-value	z-stat	Effect
1	18.20 (1.3)	0.64 (1.2)	< 0.001	38.72	1.22
2	21.32 (2.1)	1.62 (2.4)	< 0.001	38.72	1.22
3	20.63 (3.2)	1.64 (3.3)	< 0.001	38.72	1.22
4	21.70 (4.3)	0.74 (4.2)	< 0.001	38.72	1.22
5	21.20 (5.4)	1.39 (5.1)	< 0.001	38.72	1.22
6	5.17 (6.4)	1.00 (6.2)	< 0.001	38.72	1.22

Group 6, however, does not support the post-hoc hypotheses at either the low or high median condition. Expected subgroup 6.2 was not the lowest ($p < 0.001$, effect size 0.19) and expected subgroup 6.3 was not the highest ($p < 0.001$, effect size 0.25). This pool group represents pooling both frequency factors, F_1 and F_2. While significant differences were observed between the pool group 6 contrasts, the effect sizes were small. The magnitude of the range between the high and low group medians was 3.57 deg (effect size 0.25), which may not be operationally relevant.

Flexion/extension axis estimation precision

The post-hoc hypotheses offered a similar explanation on both accuracy and precision: that accuracy and precision exhibit a similar response to on-axis and off-axis amplitude and frequency effects. Precision is analogous to the inter-quartile range of the data set, due to the underlying data being not normally distributed.

The major results of the pairwise comparisons are presented in table 3.7, while the full results are shown in Appendix M.

Groups 1 through 5 all support our post-hoc hypotheses, in both low and high IQR conditions. Group 6 supports the post-hoc hypothesis in the low-IQR condition, but not in the high-IQR case. However, the observed highest IQR case is actually not significantly different from another member of pool group 6 (p = 0.78). The post-hoc hypothesis on precision of the flexion/extension axis is strongly supported.

For operational use, it is useful to have both high precision and high accuracy. However, if high precision is maintained, the error in accuracy could be used as a measure of static bias. In other words, the estimation error under some given conditions may be high, but if it is consistent, it may be able to be corrected. Conversely, having a low-error estimation condition may not be useful if it cannot be consistently repeated.

3.4.4 Pronation/Supination Axis Treatment Effects

This same approach was taken to the pronation/supination axis results. Treatments were found to be statistically different (H=1177.36, $p < 0.001$). Appendix I shows all of the pairwise comparisons between treatments.

Figure 3.8 shows a first similarity to the flexion/extension results: as expected, we generally see a lower-error and lower-variance estimate for the case of $A_2 = 1$ when compared to $A_2 = 0$.

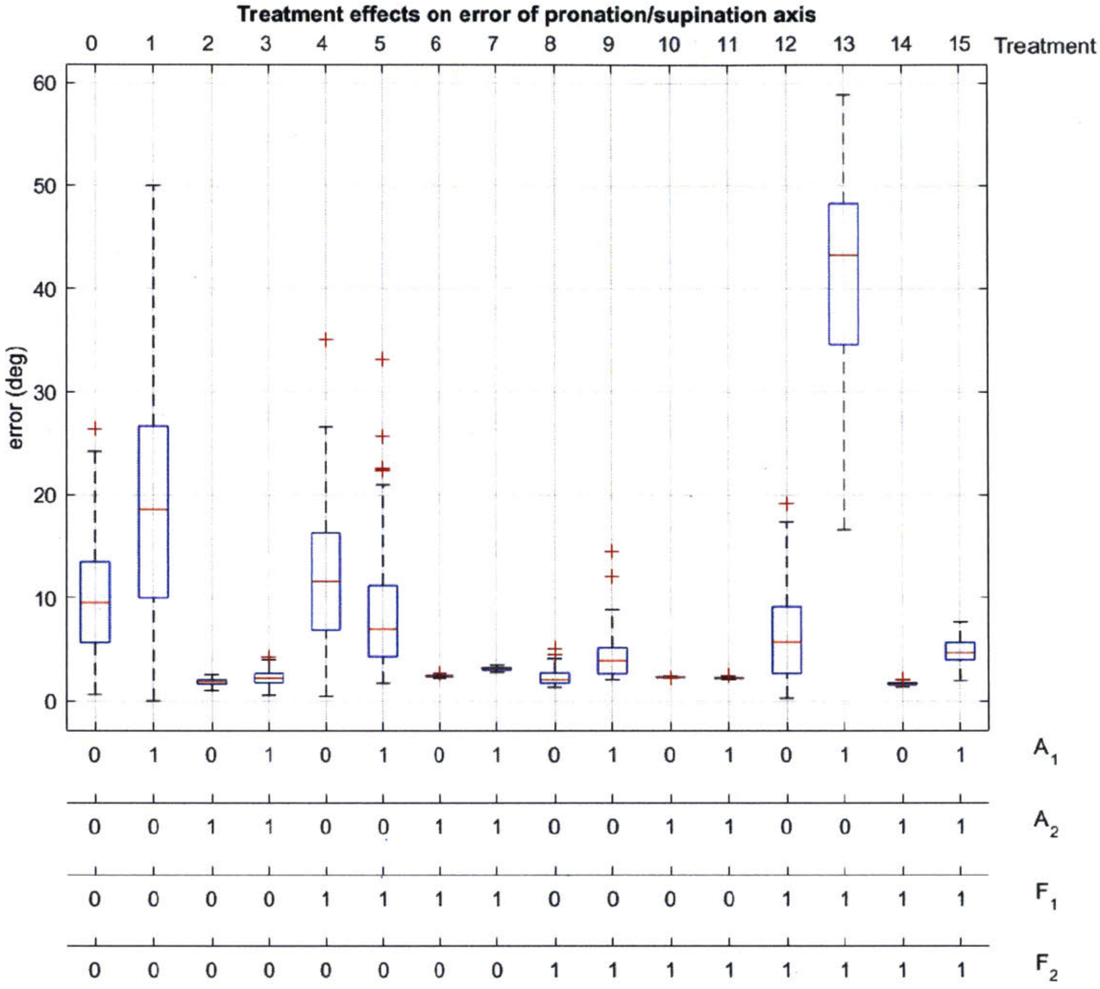


Figure 3.8 Flexion/extension axis error vs. individual treatment 0 to 15

We continue our analysis method as before with the two-factor pool groups. Appendix K displays the medians and interquartile ranges of the two-factor pool groups and their treatment subsets. Pairwise comparisons were conducted after Kruskal-Wallis supports that there are significantly different groups in both median and IQR (all $p < 0.001$).

Table 3.8 Comparison of highest and lowest median subgroups for each pool group in pronation/supination; associated subgroup is shown in parentheses

Pool Group	Highest Median	Lowest Median	p-value	z-stat	Effect
1	11.01 (1.2)	2.26 (1.3)	< 0.001	23.26	0.74
2	5.04 (2.4)	2.30 (2.1)	< 0.001	16.9	0.53
3	10.61 (3.3)	2.25 (3.2)	< 0.001	21.04	0.67
4	4.40 (4.4)	2.27 (4.3)	< 0.001	13.43	0.42
5	10.58 (5.1)	2.28 (5.4)	< 0.001	20.36	0.64
6	4.87 (6.4)	2.31 (6.3)	< 0.001	7.77	0.25

Pronation/supination axis accuracy

Just as in the case of flexion/extension, pairwise comparisons on medians were completed using a Wilcoxin Rank Sum test. The results of the pairwise comparisons are shown in Appendix N but results are highlighted in Table 3.8.

Pool groups 1, 2, 3, and 5 show low and high median subgroups that align with the post-hoc hypotheses. Group 4 supports the hypotheses in the low median condition, and the expected high median condition (subgroup 4.2, median 3.31) is not significantly different from the observed high median (subgroup 4.4, median 4.40, $p = 0.064$). We conclude that group 1 through 5 support the post-hoc hypotheses.

Group 6 supports the post-hoc hypothesis in the low median condition, but not the high median condition. However, this effect is not very large (0.38 effect size between expected high median subgroup 6.2 and observed high median subgroup 6.4). Just as in the case of flexion/extension, group 6 shows the smallest effect sizes on pairwise comparisons. The magnitude of difference in medians between the lowest and highest median subgroup is only 2.56 deg (effect size 0.25), which may not be operationally relevant.

The data support the post-hoc hypotheses that high on-axis and low-off axis amplitude and frequency increase accuracy (i.e., decrease median error). Pool group 6 shows mixed support, although has a smaller effect size and highlights that median error is more sensitive

Table 3.9 Comparison of highest and lowest IQR subgroups for each pool group in pronation/supination; associated subgroup is shown in parentheses

Pool Group	Highest IQR	Lowest IQR	p-value	z-stat	Effect
1	27.47 (1.2)	0.67 (1.3)	< 0.001	38.72	1.22
2	17.95 (2.4)	1.08 (2.1)	< 0.001	38.72	1.22
3	18.80 (3.3)	0.40 (3.2)	< 0.001	38.72	1.22
4	13.28 (4.4)	1.04 (4.3)	< 0.001	38.72	1.22
5	15.02 (5.3)	0.36 (5.4)	< 0.001	38.72	1.22
6	15.60 (6.4)	0.47 (6.3)	< 0.001	38.72	1.22

to changes in amplitude.

Pronation/supination axis precision

All pairwise comparisons of subgroup IQR were statistically significant. The complete set of pairwise comparisons is shown in Appendix O, and the highest and lowest subgroups are shown in Table 3.9.

Group 1 through 3 support the post-hoc hypotheses in both the low and high IQR conditions. Group 4 through 6 support the post-hoc hypotheses in the low IQR case, but not the high IQR case.

As the lowest IQRs supported the hypothesis for all pool groups, these data do provide insight on the recommended system usage. When on-axis amplitude and frequency are high and off-axis amplitude and frequency are low, we observe high precision (i.e., low IQR). As discussed previously, a high precision solution may be considered a static error in accuracy, and lend itself to a correction method.

The lack of support for the highest IQR condition in the latter 3 pool groups may point to a 3-way interaction between factors. Furthermore, IQR is only a single measure of the spread of a non-normal distribution. IQR explicitly is the 25th and 75th percentile data points. It is possible to have a very long tail in a distribution which falls outside of this

IQR, i.e., the range of errors in the top quartile may be wider than the range described by the two middle quartiles. In such a case, the IQR does not adequately capture the spread of the distribution. That is likely the case here, where IQR is consistently underpredicting the spread of pool group 6, which pools the two frequency factors. The additional spread in the upper quartile was driven by pooling treatments that had high and low individual IQR. This reinforces that the axis error precision is more sensitive to on and off-axis amplitudes than frequencies.

3.4.5 Conclusions

In conclusion, we evaluated the Müller method against human-generated motions and against sinusoidal motion. We proposed post-hoc hypotheses that on-axis high amplitude and frequency and low off-axis amplitude and frequency would decrease error and error variance, and that off-axis high amplitude and frequency and off-axis low amplitude and frequency would increase error and error variance. It was found that high on-axis motion and low off-axis motion did indeed decrease error variance (i.e., made axis estimation more precise). This hypothesis was also supported in establishing low error (i.e., high accuracy) in pronation/supination axis estimation. While it was not supported in establishing high accuracy on the flexion/extension, it was consistent with the condition of *low* accuracy. In practice, high accuracy would be desired on both axes of estimation. However, with knowledge of high precision conditions, a precise-but-inaccurate axis estimation may lead to a static bias correction method.

These results are consistent with a physical interpretation of the system. For an IMU mounted on the human wrist, flexion/extension motion has a much higher radius of rotation at the IMU than pronation/supination motion. If the angular velocities of motion are the same on both degrees of freedom (in this study frequency levels were the same for both degrees of freedom), then the IMU undergoes higher linear velocity in flexion/extension. Higher linear velocities would yield higher linear accelerations. As discussed in Section 1.2.2, the measurement model of the filter models static accelerations due to Earth gravity only. Dynamic body accelerations, therefore, introduce error in the measurement model. In cases of high flexion/extension motion, we conclude there should be more error in the filtered IMU orientation, which in turn would yield erroneous data for the Müller method to use. This is an explanation for why high pronation/supination motion yielded accurate axis estimations, but high flexion/extension motion did not. Future work can address an optimal range of amplitude and frequency of flexion/extension motion for use in online motion.

Chapter 4

Conclusions, Recommendations, and Future Work

4.1 Research Summary

We compared the open filters in Chapter 1 to a truth datum and a black-box filter from a manufacturer for estimation of human performance. We found that the EKF and UKF, with included heuristics, yielded around ten degrees RMSE over the entire experiment. Operationally, there was no difference in total performance between the two filters. The particle filter was found to perform better, on the order of the manufacturer's black-box filter. Notably, the manufacturer's filter seemed to perform better with the smooth circle than the particle filter, but the particle filter outperformed the manufacturer's filter around the discrete square. This could possibly be due to the manufacturer's filter including heuristics which prefer smooth dynamics. On the other hand, the particle filter might handle sharp discontinuities better because of its ability to lessen the weight of particles which significantly deviate from the measured behavior. We conclude that the particle filter is an appropriate open filter for use, but is computationally more expensive.

The particle filter was then the subject of a subsequent study to analyze its sensitivity to its input parameters to inform parameter selection, as currently parameter selection is performed via visual inspection. The chosen parameters were accelerometer uncertainty, magnetometer uncertainty, and the number of particles in the simulation. We found that the accelerometer uncertainty and number of particles significantly affected the accuracy of the filter for the levels chosen. Operationally, we recommend values of 1 to 5 m^2/s^4 for accelerometer uncertainty and 25 particles for usage within the implemented particle filter (Cheng and Crassidis [22]). The magnetometer uncertainty did not significantly affect filter performance at the levels chosen. We motivate that while no significant difference was found between levels, the theory tells us that there is still an operational range of appropriate levels for this parameter. Extreme values outside of this range underpredict or overpredict magnetometer uncertainty, leading to overtrust or undertrust of the sensor. Both conditions are inappropriate. An operational limit of 50-100 μT^2 is appropriate. A further explanation could be that for the motions selected, the magnetometer measurement update was small, and so the level chosen for magnetometer uncertainty would not impact the state much.

Finally, we evaluated the Müller auto-calibrating elbow system method in the context of both human motion and sinusoidal motion through the use of a robotic test bed. The human motions selected were activities of daily living (ADLs), which are commonly used within occupational therapy to rehabilitate patients into normal living. The method was also evaluated against sinusoidal motion which was parameterized by amplitude and frequency. We formulated post-hoc hypotheses: high on-axis amplitude and frequency decreases axis estimation error, while high off-axis amplitude and frequency increases axis estimation error. We found that in flexion/extension, this post-hoc hypothesis was supported in high-error (low-accuracy) conditions, and also supported in all precision conditions. In pronation/supination, our post-hoc hypotheses were supported in the high and low accuracy conditions, and were supported in the low IQR (high precision) condition.

Operationally, we conclude that high on-axis motion and low off-axis motion yields high precision axis estimations for both degrees of freedom. High accuracy is only established by this motion in pronation/supination. In general, high accuracy would ideally be established by this motion on both degrees of freedom. However, if an axis estimation is found to be precise, but not accurate, it still may be able to be corrected by a static error correction method. Furthermore, we understand the condition which produces low accuracy estimation in flexion/extension, and these conditions can be avoided when using a static error correction method.

4.2 Contributions

This thesis made the following contributions.

1. We compared the performance of the EKF, UKF, and PF in the context of a simple upper extremity human motion against optical motion capture. It was found that PF had lowest errors while EKF and UKF performed similarly. This work is the first to implement a particle filter for inertial sensing of human motion.
2. We performed a sensitivity analysis of the particle filter to its parameter inputs for use in human motion. Recommended operational ranges for accelerometer uncertainty, magnetometer uncertainty, and number of particles were found.
3. We investigated the potential of the Müller method for online axis updating by assessing performance under human occupational motion and parameterized sinusoidal input. The data supported that high on-axis motion and low off-axis motion yields high precision axis estimation, and also accurate estimation of the pronation/supination axis. Data also supported that high off-axis motion and low on-axis motion yields inaccurate axis estimations.

4.3 Limitations and Future Work

As with many studies in human motion, the experimental results of Chapter 2 may be limited to the motions performed. Since two of the three experimental motions are performed in the horizontal plane, 3D motion is not fully assessed in these motions. The grasping task does move in three dimensions, but is only one instance of a 3D motion. The robotic experiment performed in Chapter 3 had binary factor levels. In reality, the intricacies of treatment interaction on Müller axis estimation may not be able to be captured by pairs of high/low levels. However, this approach was a good starting point to provide information about the qualities of successful online calibration motion.

Future work will seek to take advantage of the continuous-estimation capability of the Müller method to correct drift of an IMU over time. While an inertial sensor may drift over long periods of time, the human joint axis does not. Knowledge of this joint axis, first established by undrifted sensors and the Müller method, may be recalled to continually update the possibly-drifting inertial frame over time. The conditions under which these updates are best performed are informed by the experimental results of Chapter 3, for example, high motion in pronation/supination but low motion in flexion/extension yields accurate axis estimations of the pronation/supination elbow axis. Functionally, this type of method would utilize knowledge of human kinematic constraints to provide a novel solution to inertial sensor drift. As motivated by the results of Chapter 3, future work can also seek to determine an optimal range of flexion/extension amplitude and frequency; data suggested that too little motion provides for poor estimation while too much motion introduces error into the filtered IMU orientation. Finally, further work can use more factor levels to build a continuous regression model of expected axis error, rather than the high/low interpretation of this thesis.

Appendix A

Vicon experiment motions

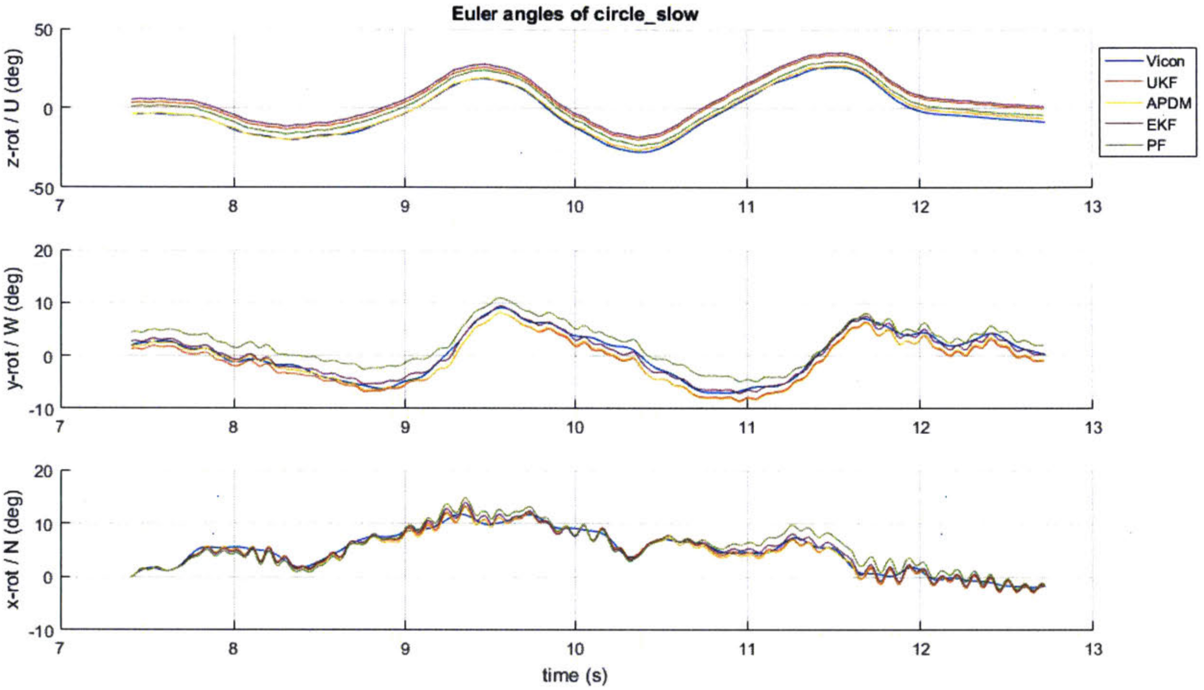


Figure A.1 Truth (Vicon) vs. estimated orientations for slow circle motion

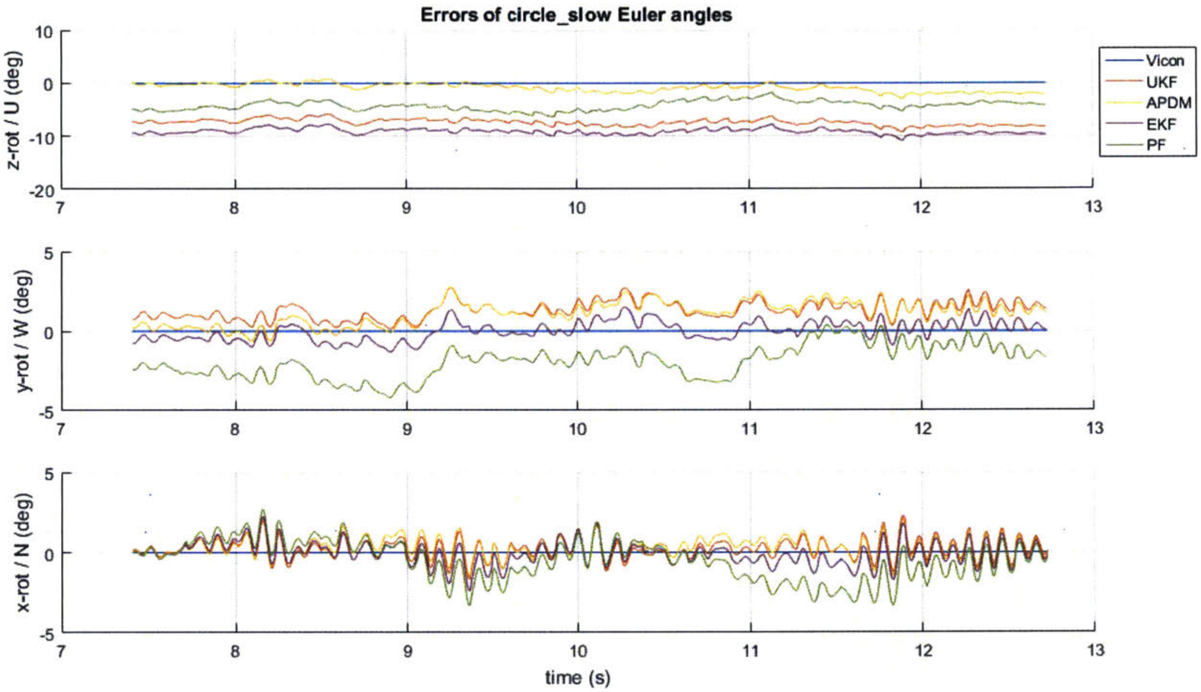


Figure A.2 Error of estimated orientations for slow circle motion

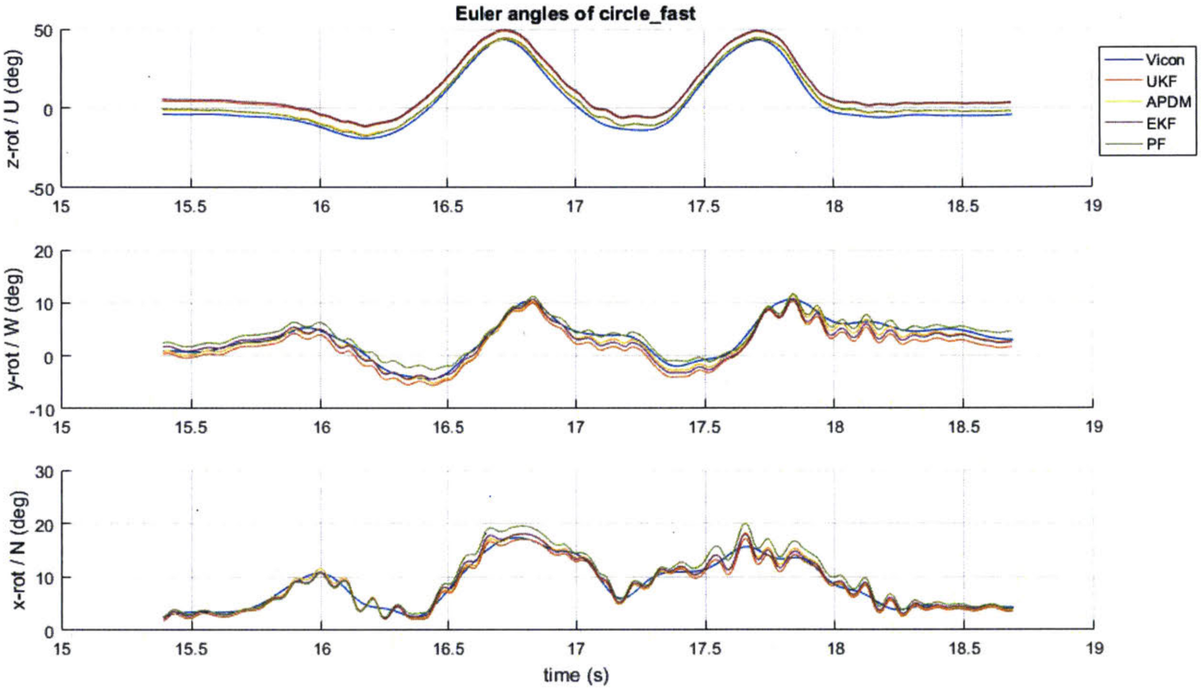


Figure A.3 Truth (Vicon) vs. estimated orientations for fast circle motion

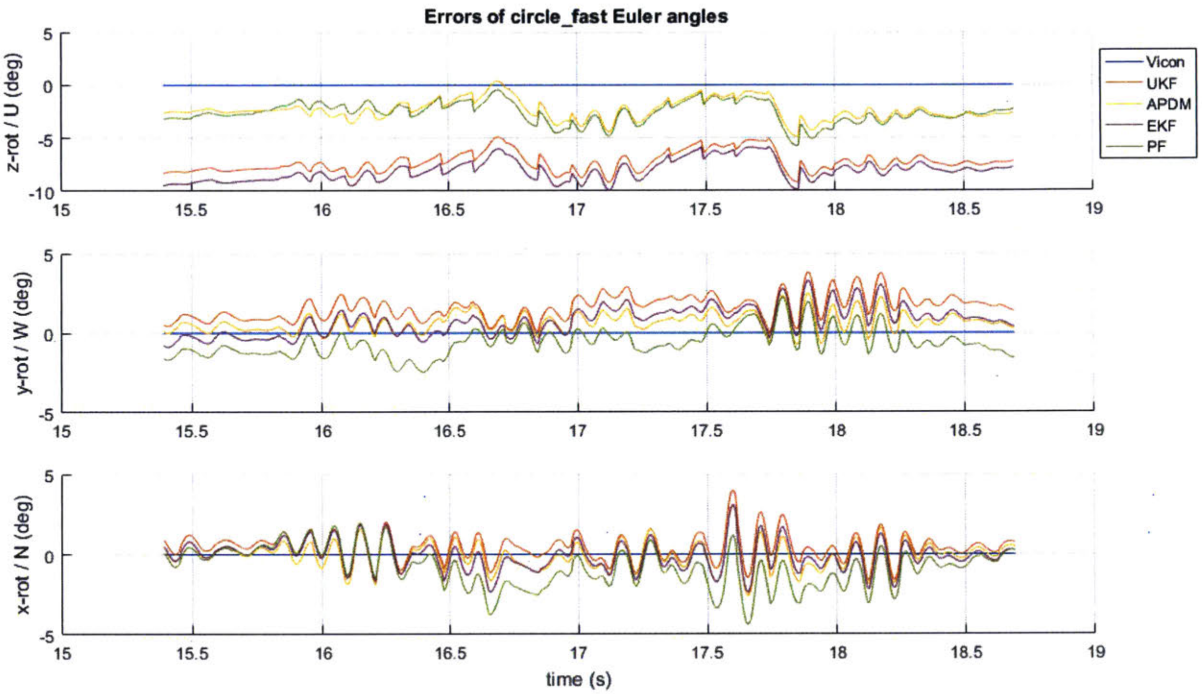


Figure A.4 Error of estimated orientations for fast circle motion

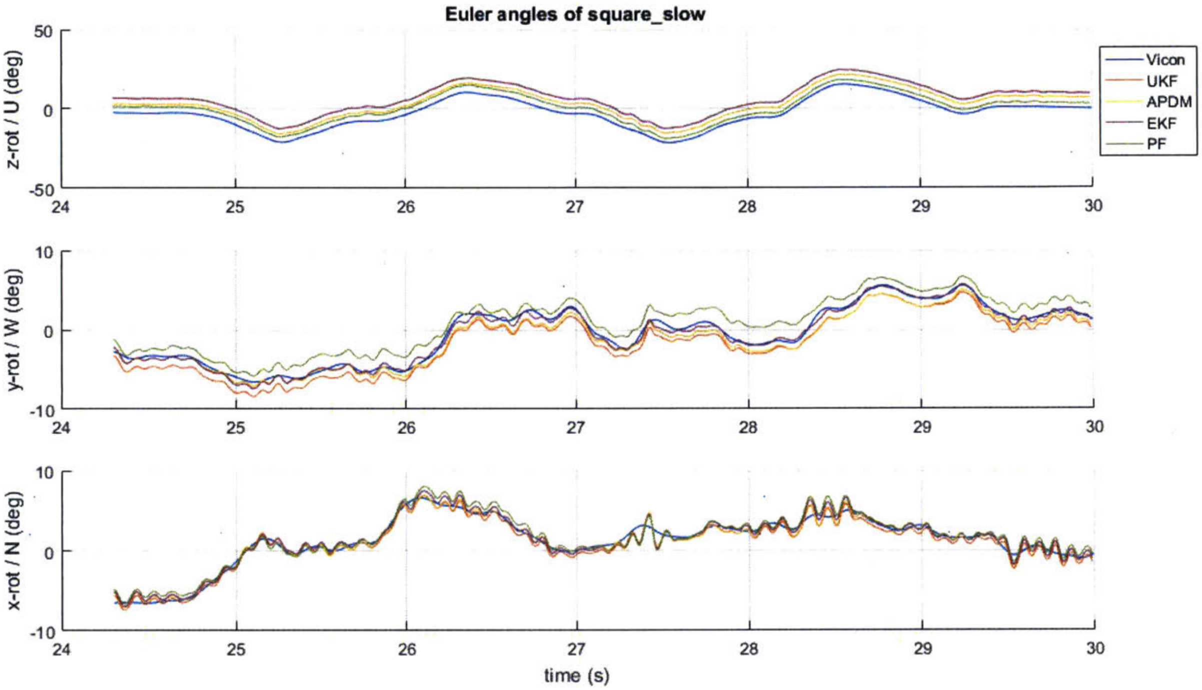


Figure A.5 Truth (Vicon) vs. estimated orientations for slow square motion

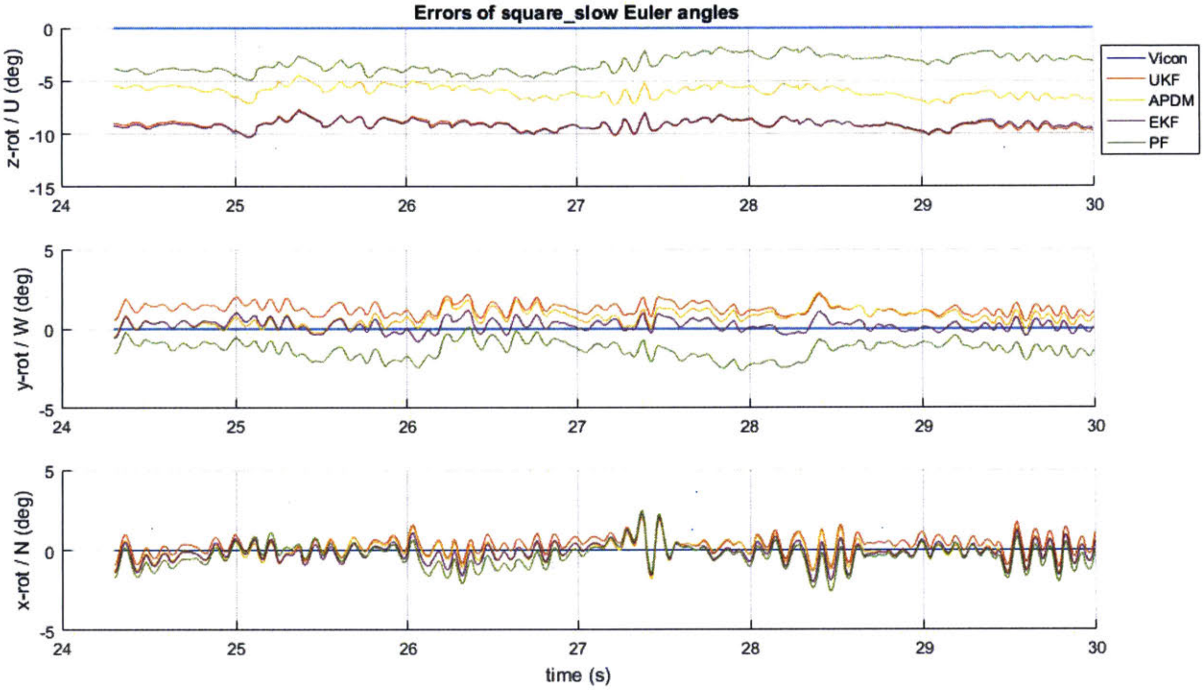


Figure A.6 Error of estimated orientations for slow square motion

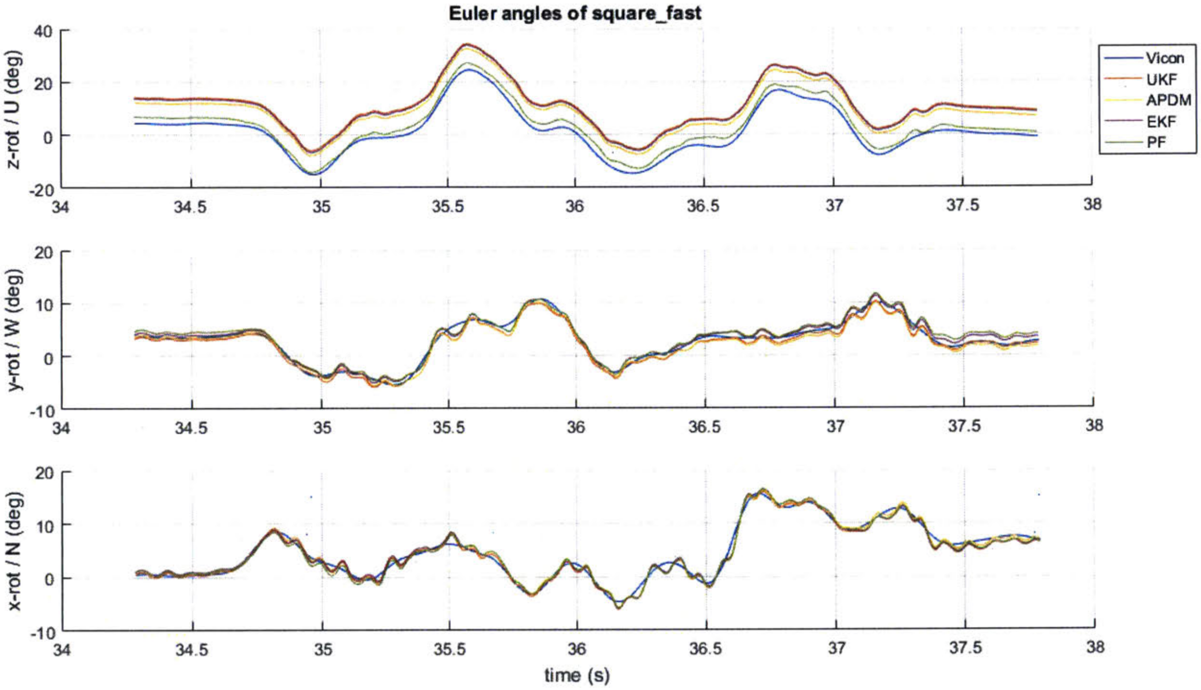


Figure A.7 Truth (Vicon) vs. estimated orientations for fast square motion

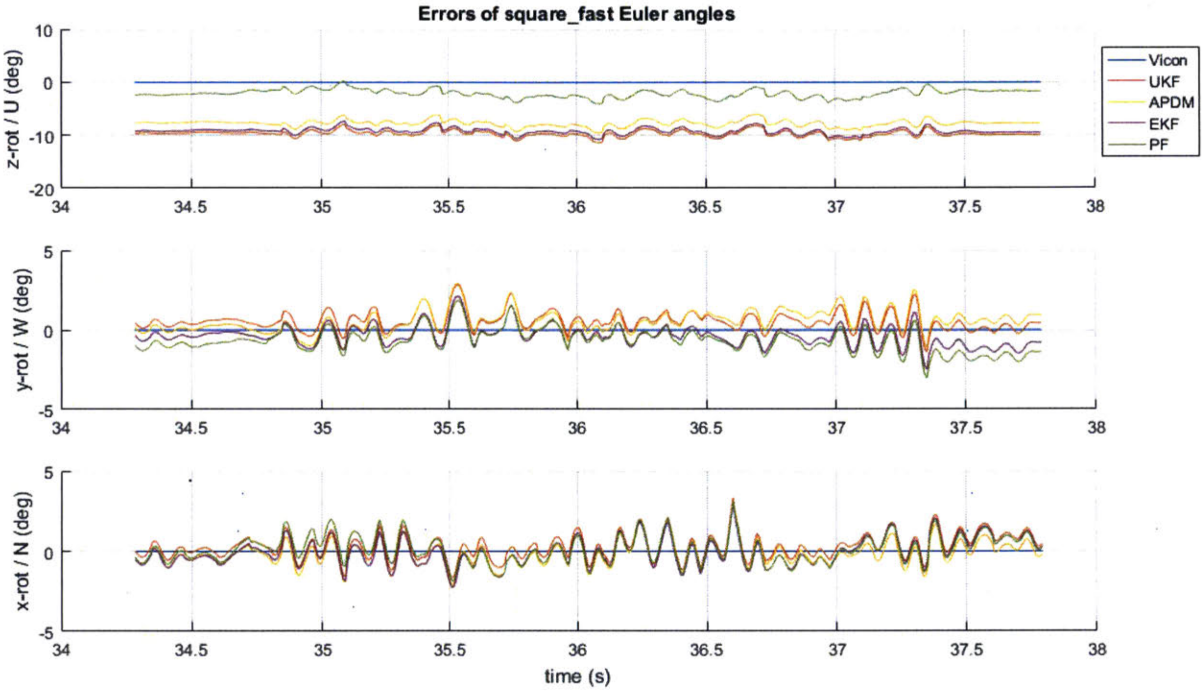


Figure A.8 Error of estimated orientations for fast square motion

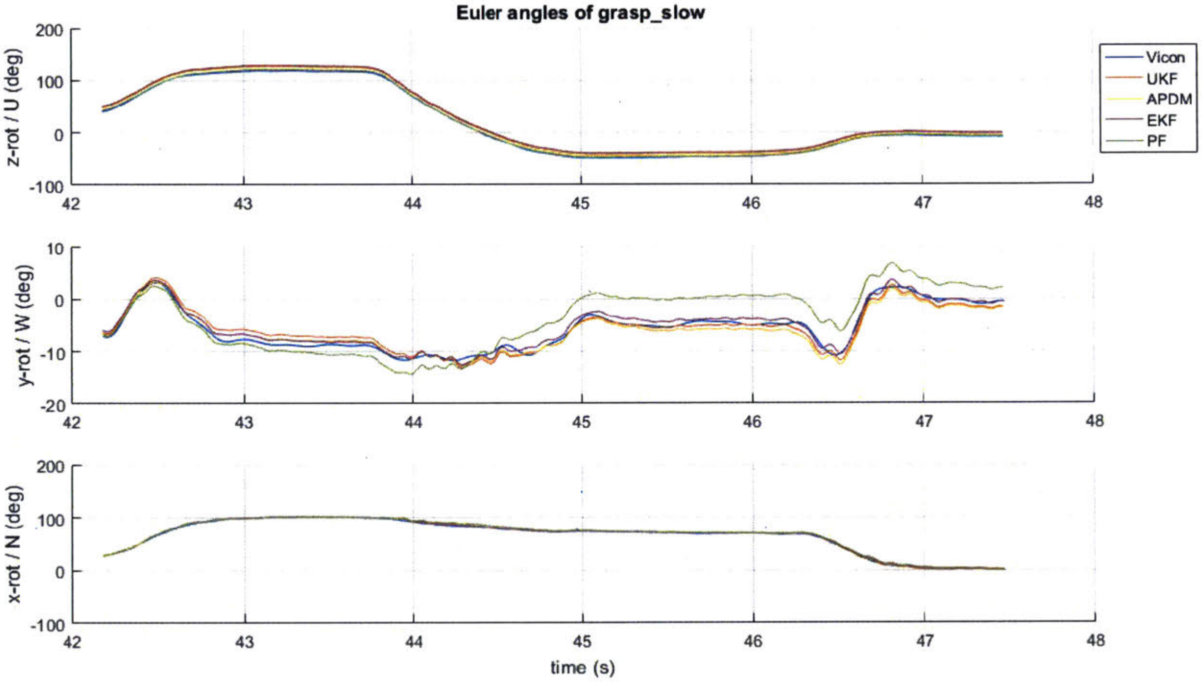


Figure A.9 Truth (Vicon) vs. estimated orientations for slow grasp motion

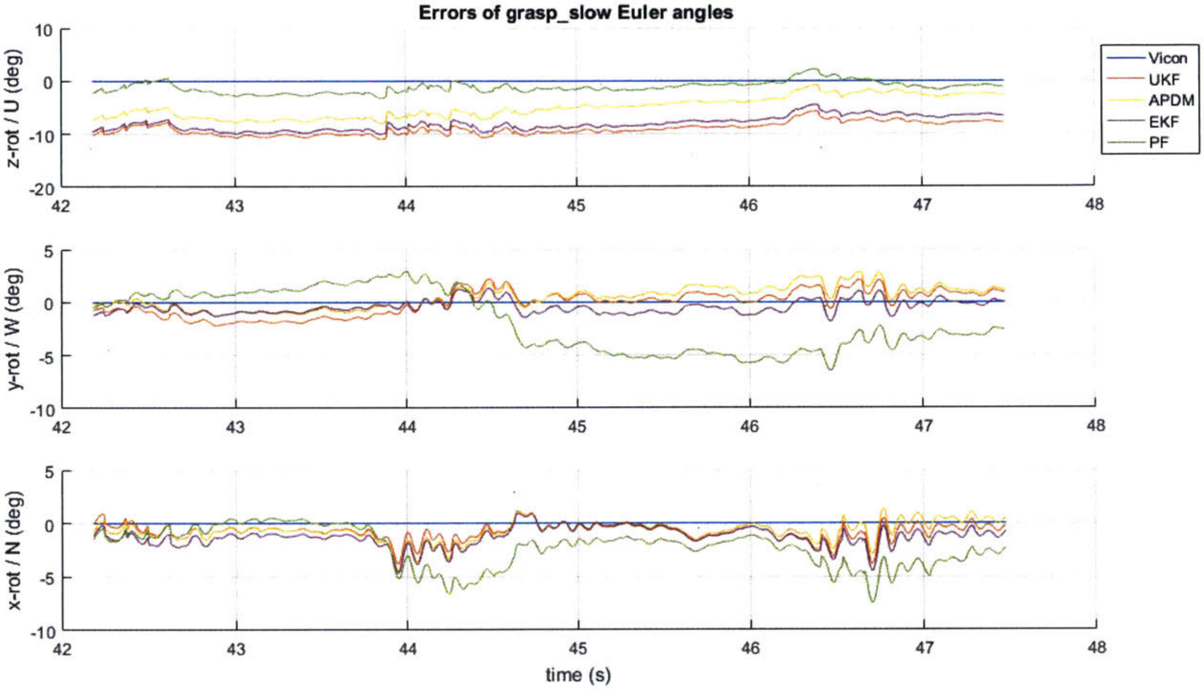


Figure A.10 Error of estimated orientations for slow grasp motion

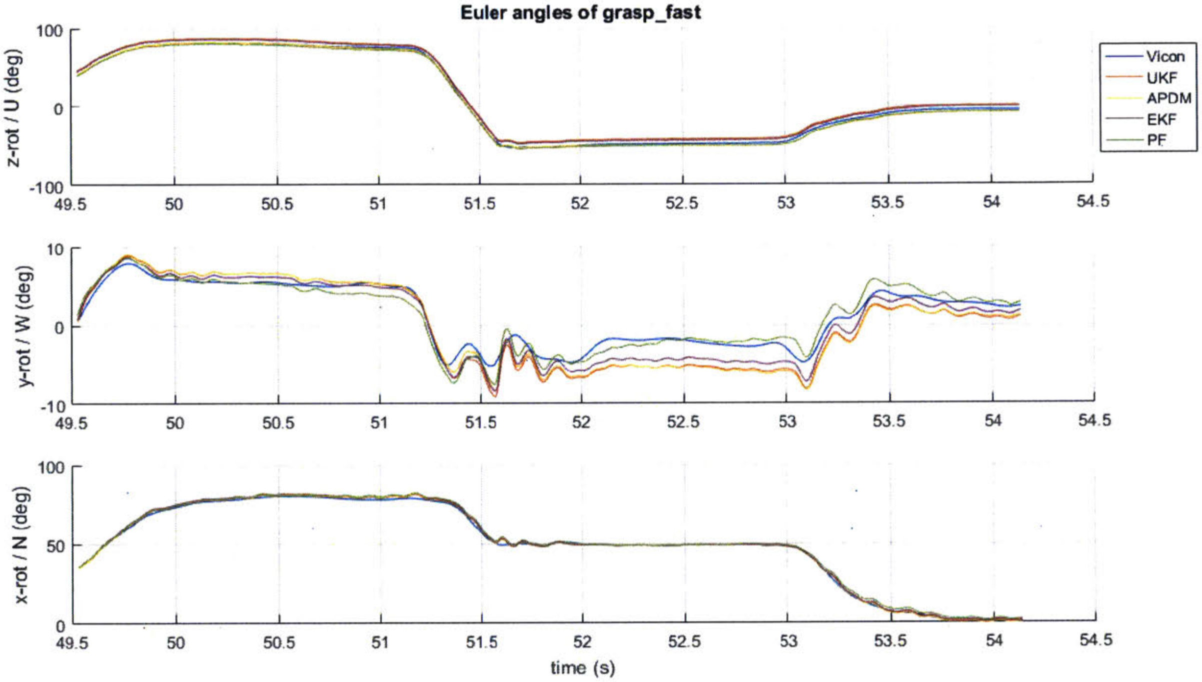


Figure A.11 Truth (Vicon) vs. estimated orientations for fast grasp motion

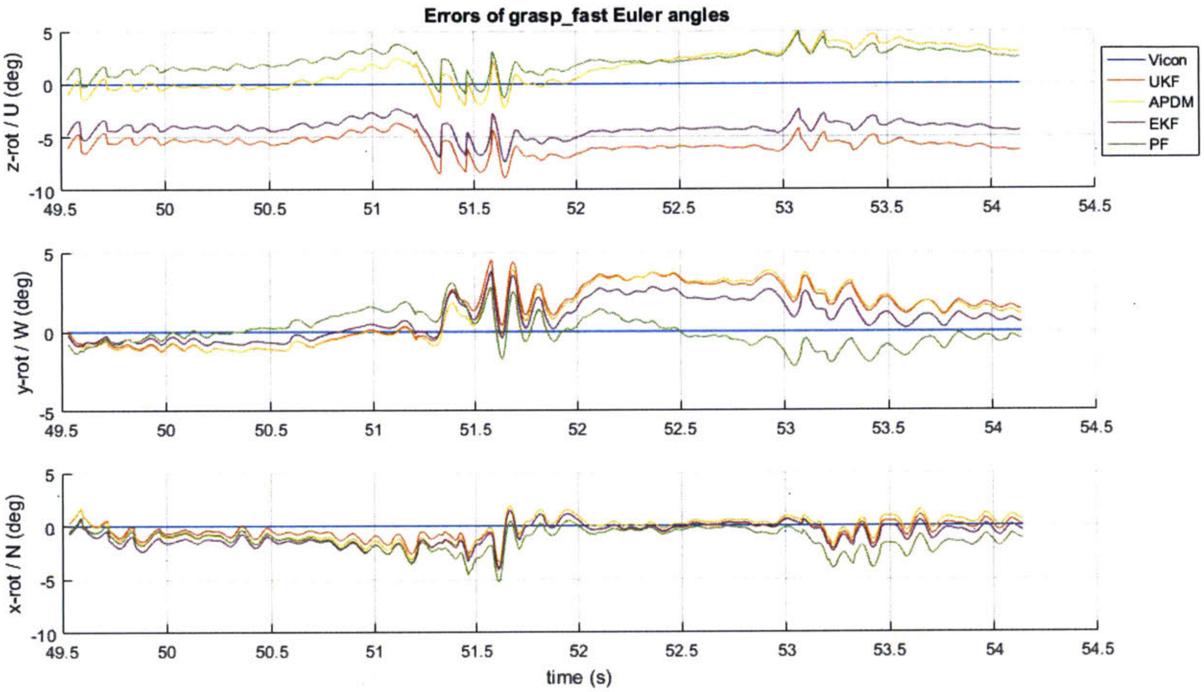


Figure A.12 Error of estimated orientations for fast grasp motion

Appendix B

Normality tests for particle filter parameter sensitivity analysis

This appendix establishes the lack of normality in response data to the particle filter sensitivity analysis experiment presented in Section 2.3.

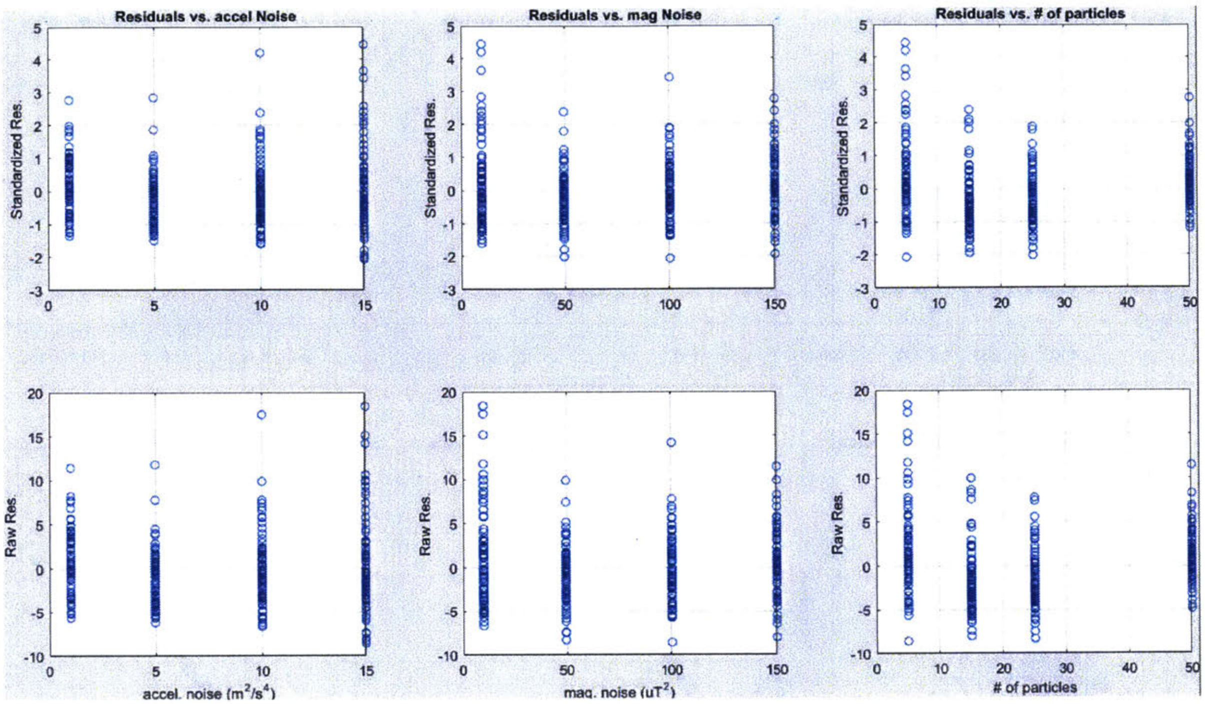


Figure B.1 Raw and Standardized residuals plotted against response

Figure B.1 does not seem to indicate any problems with the residuals at first glance. More interesting are the residuals as presented in Figure B.2. Plotting the residuals vs. normal probability and other plots can show more information.

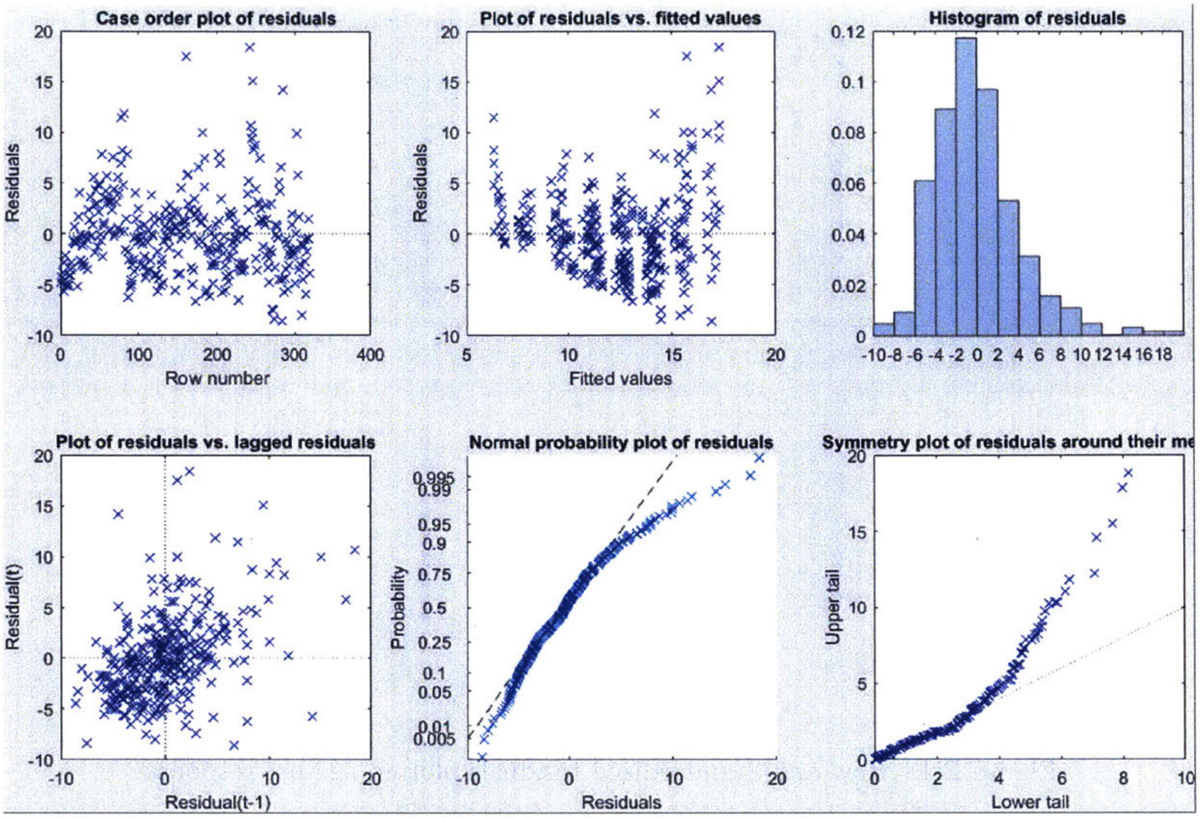


Figure B.2 Full residual plots

The first thing one might notice are the seemingly large deviations from normality that these residuals take. Of note is first the normal probability plot. The tails seem to make very large deviations from normality. This is not a full indictment in itself. Then one should note the symmetry of the residuals. Which is to say that they are not symmetric. Finally, the histogram gives a final dagger as it clearly shows non-normal residuals. The histogram is heteroscedastic. It skews towards a right tail. It almost looks like an F-distribution. Nonetheless, these residuals do not seem to meet the assumptions of ANOVA.

Kolmogorov-Smirnov Test Just to reinforce the intuition of non-normality of the residuals, a Kolmogorov-Smirnov test can be performed to test for normality. In this one-sample test, the null hypothesis is that the data comes from a normal distribution. Hence, a low p-value indicates non-normality of the data. The test on the standardized residuals reveals a $\mathbf{p = 0.0187}$. This confirms our previous suspicion: the residuals are not normal. Transforming the data is an option, but a non-parametric test will be employed here.

Appendix C

Normality tests for particle filter parameter sensitivity analysis

This appendix illustrates the lack of interaction among two specific factors of the experiment presented in Section 2.3.

Figures C.1 and C.2 show box plots of error vs. the number of particles with accelerometer and magnetometer noises as a grouped variable. Inspection of these plots reveals that the effect of the accelerometer and magnetometer noises on error is the same over the number of particles. Thus, it is concluded that interaction effects of the accelerometer and magnetometer noises are small.

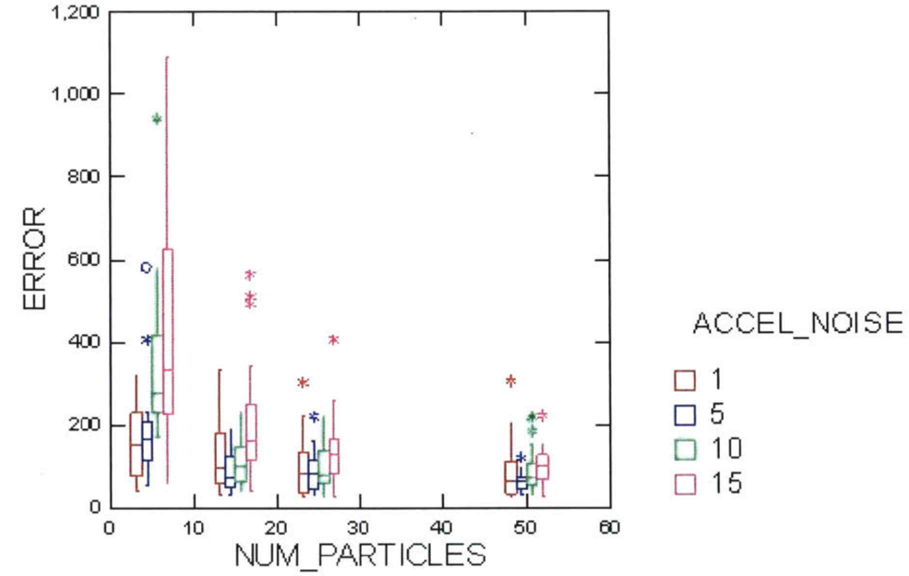


Figure C.1 Accelerometer noise interaction box plot

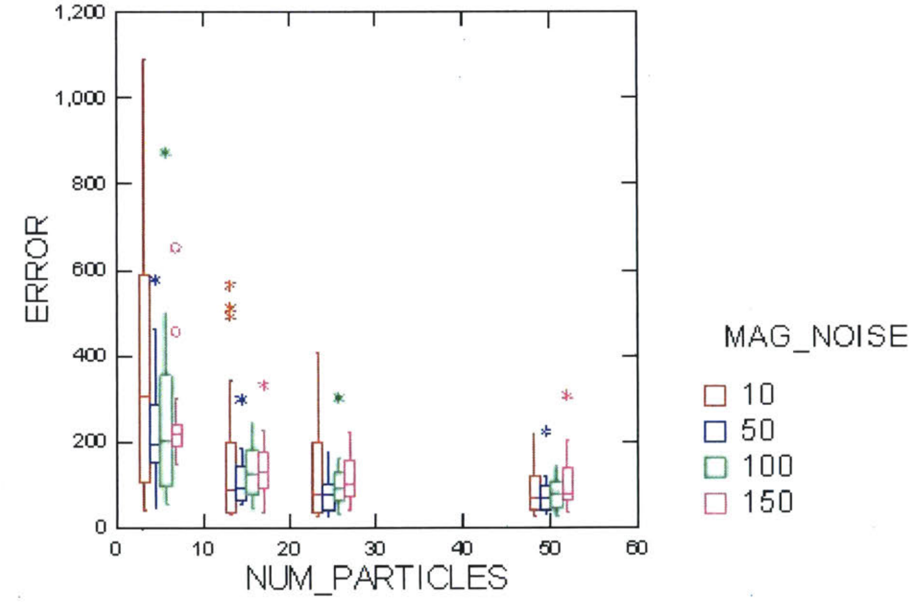
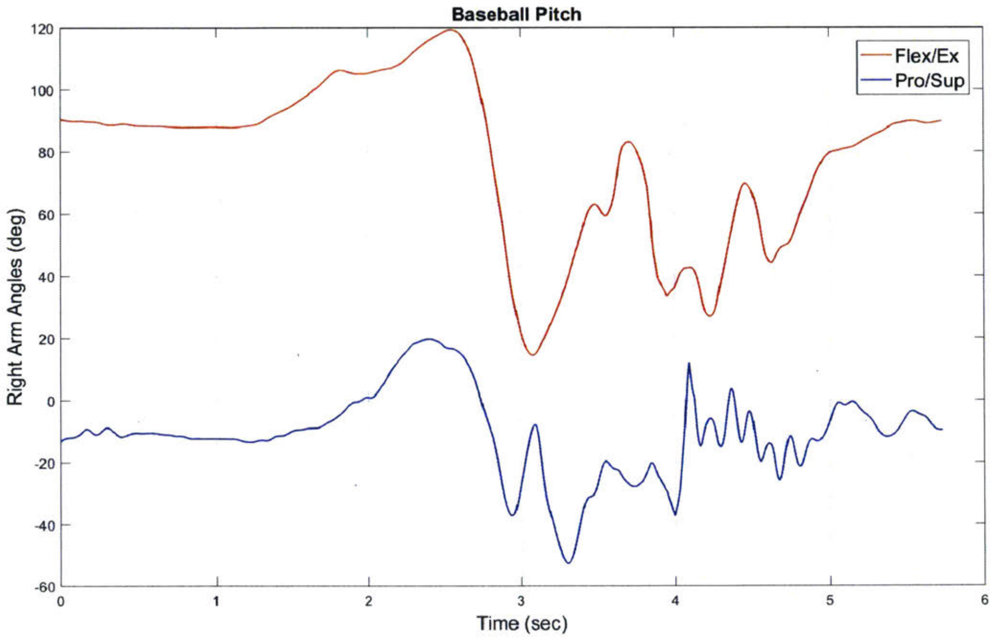
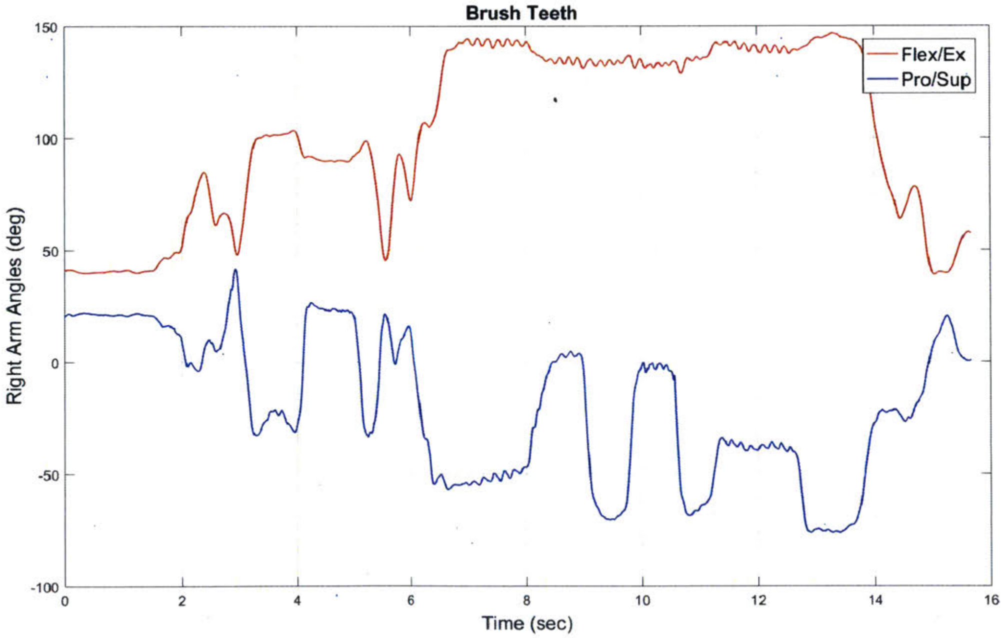


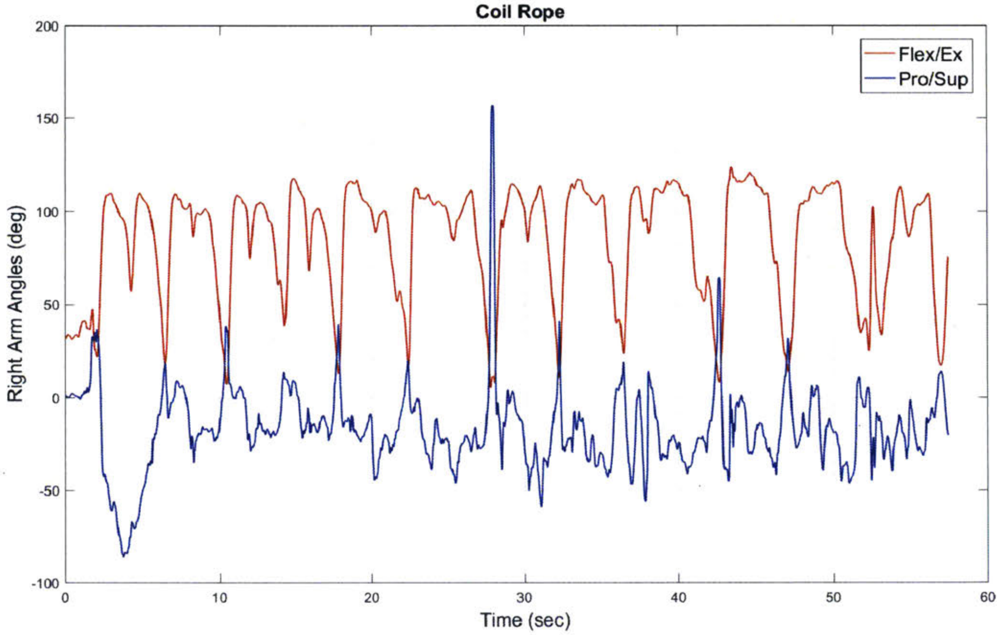
Figure C.2 Magnetometer noise interaction box plot

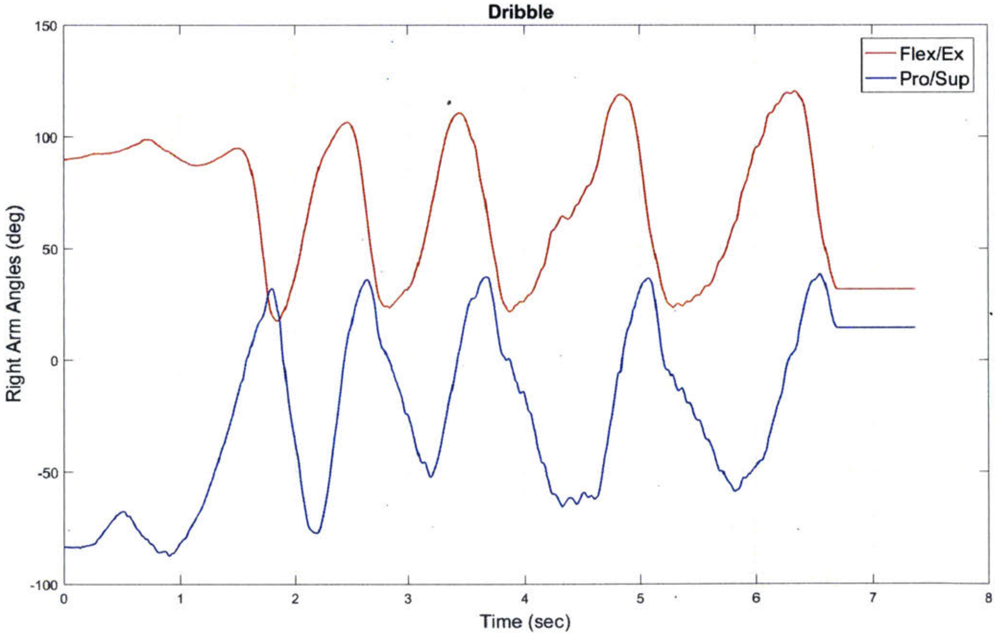
Appendix D

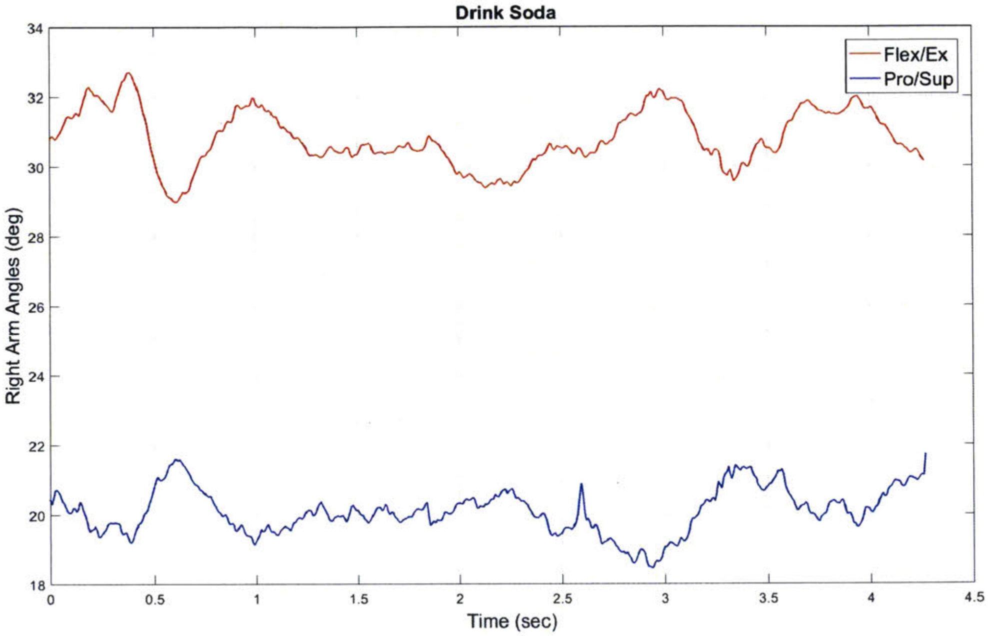
ADL Motion Profiles

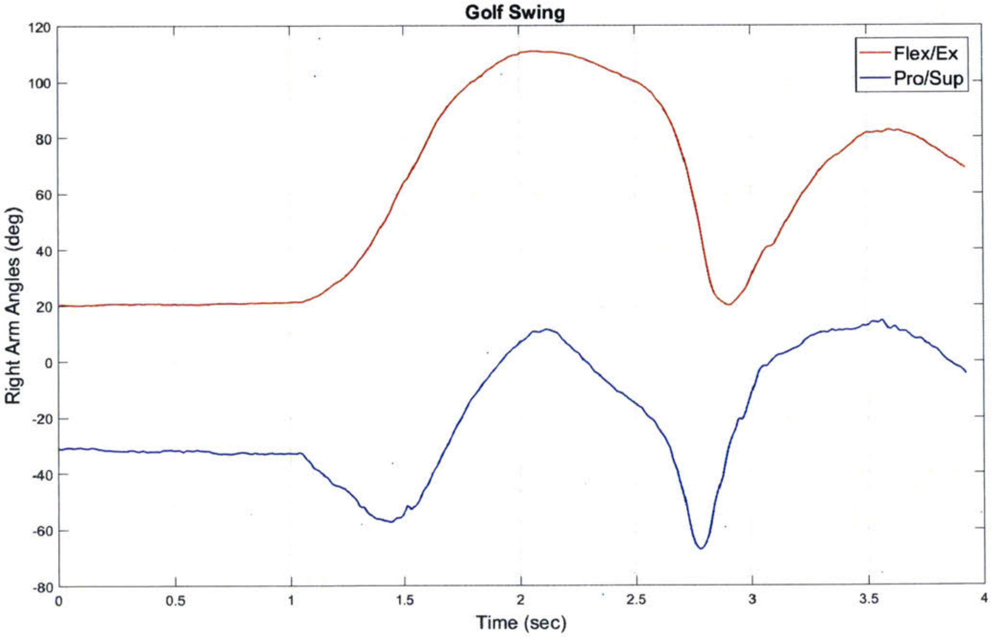

Baseball Pitch
Right Arm Angles (deg)
120
100
80
60
40
20
0
-20
-40
-60
0
1
2
3
4
5
6
Time (sec)
Flex/Ex
Pro/Sup

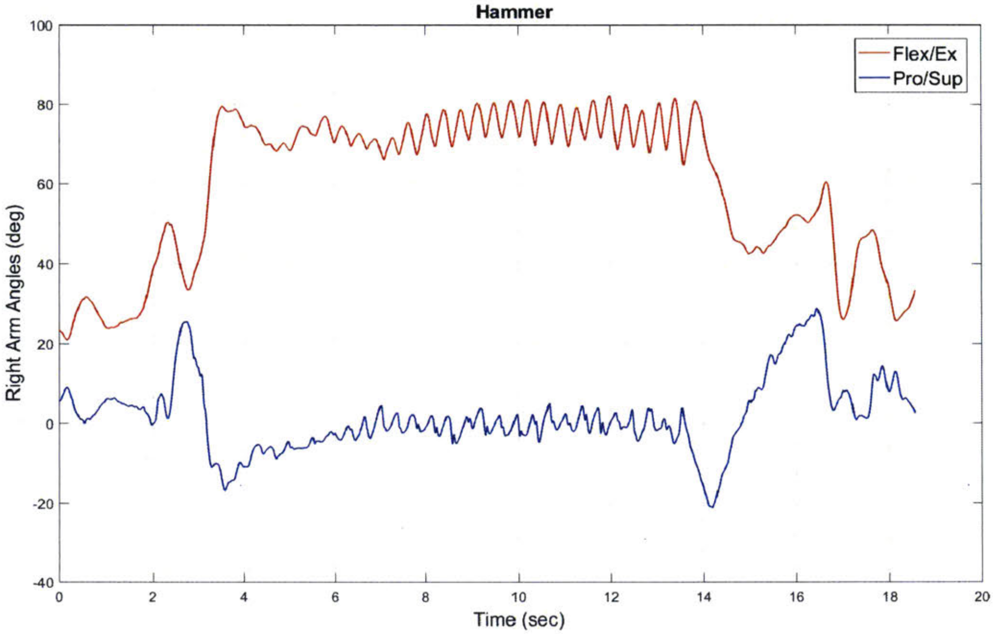

Brush Teeth
Right Arm Angles (deg)
150
100
50
0
-50
-100
0
2
4
6
8
10
12
14
16
Time (sec)
Flex/Ex
Pro/Sup

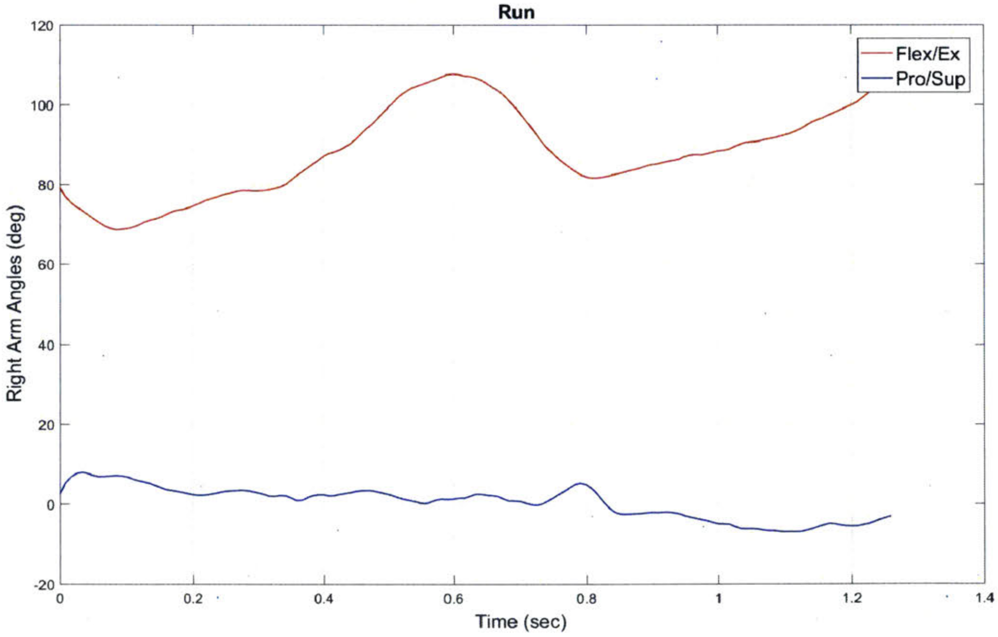

Coil Rope
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
200
150
100
50
0
-50
-100
0
10
20
30
40
50
60
Time (sec)

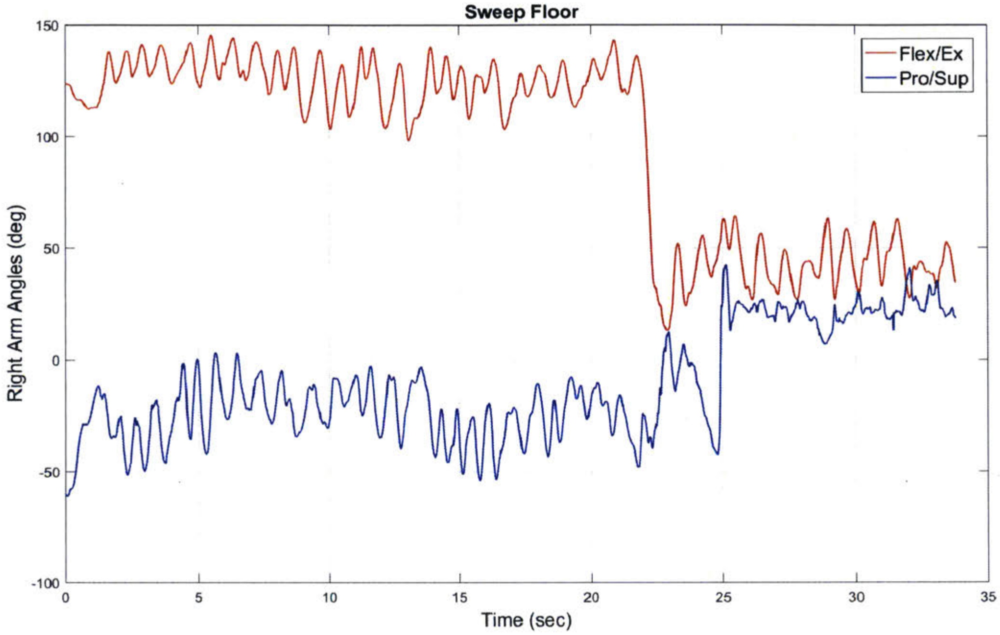

Dribble
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
150
100
50
0
-50
-100
0
1
2
3
4
5
6
7
8
Time (sec)

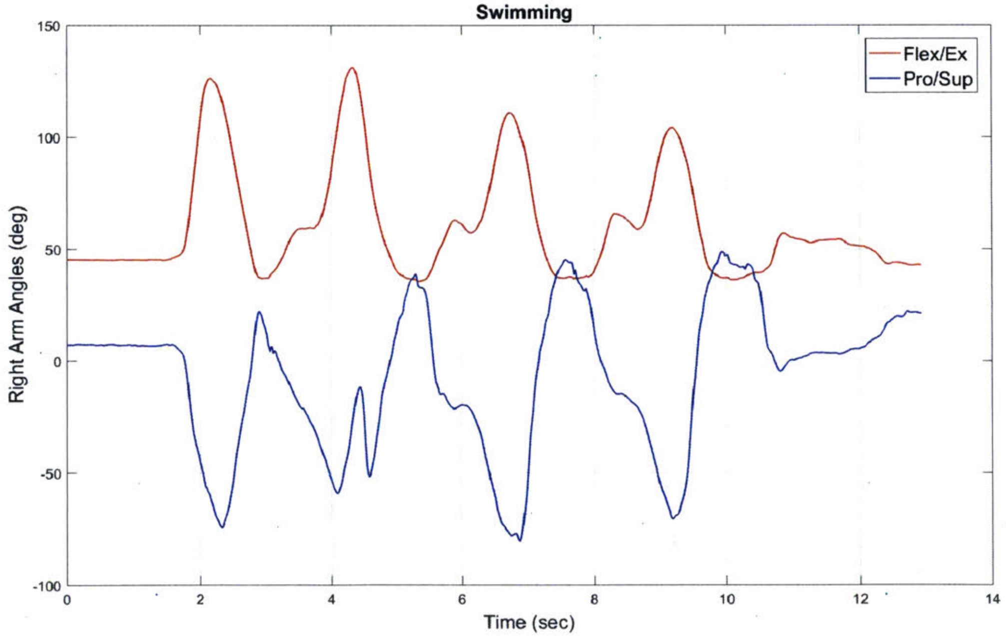

Drink Soda
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
34
32
30
28
26
24
22
20
18
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
Time (sec)

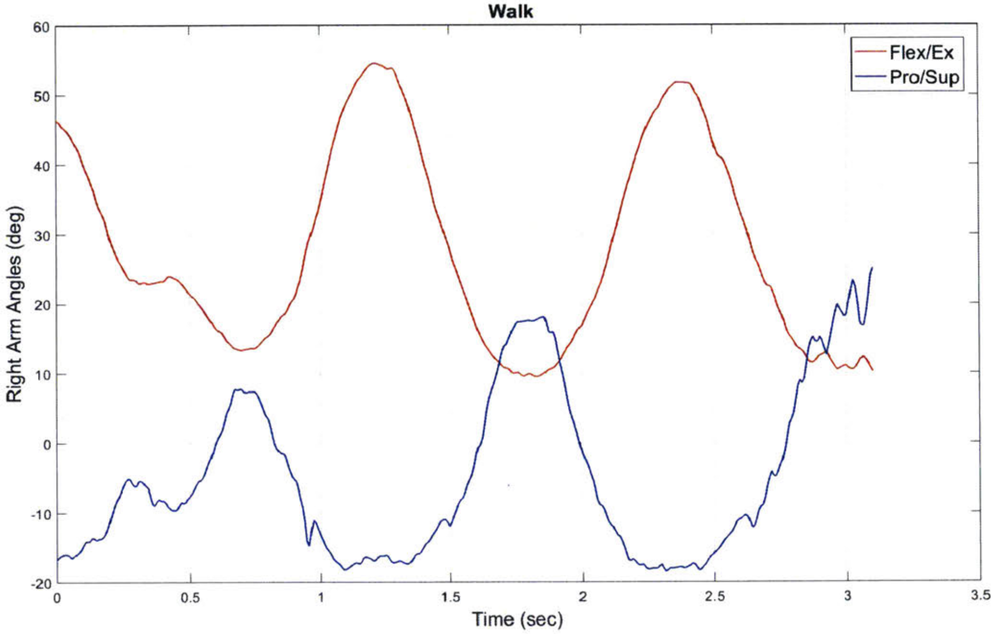

Golf Swing
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
120
100
80
60
40
20
0
-20
-40
-60
-80
0
0.5
1
1.5
2
2.5
3
3.5
4
Time (sec)

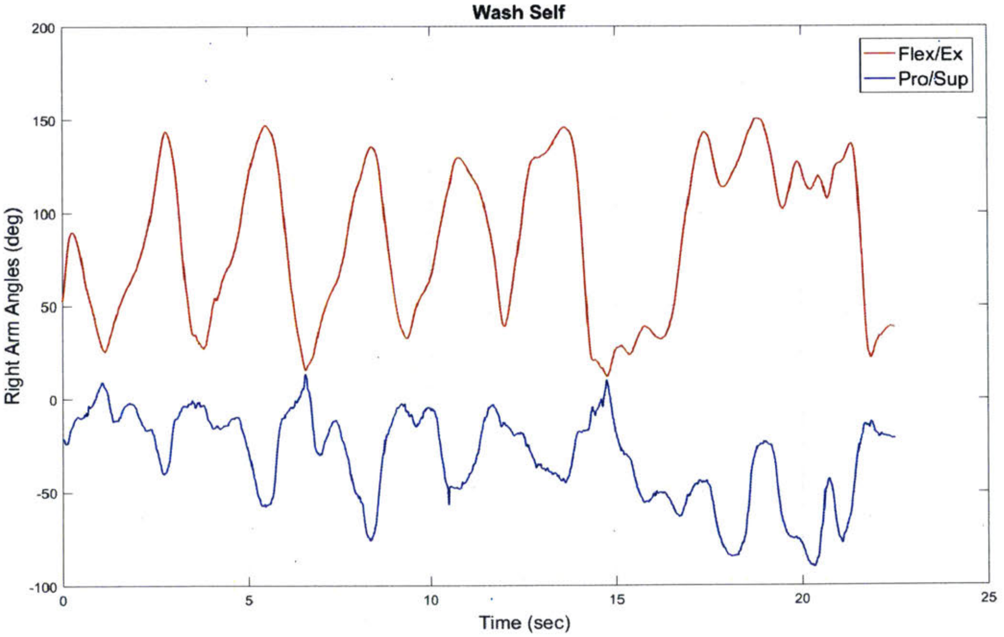

Hammer
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
100
80
60
40
20
0
-20
-40
0
2
4
6
8
10
12
14
16
18
20
Time (sec)


Run
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
120
100
80
60
40
20
0
-20
0
0.2
0.4
0.6
0.8
1
1.2
1.4
Time (sec)


Sweep Floor
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
150
100
50
0
-50
-100
0
5
10
15
20
25
30
35
Time (sec)


Swimming
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
150
100
50
0
-50
-100
0
2
4
6
8
10
12
14
Time (sec)

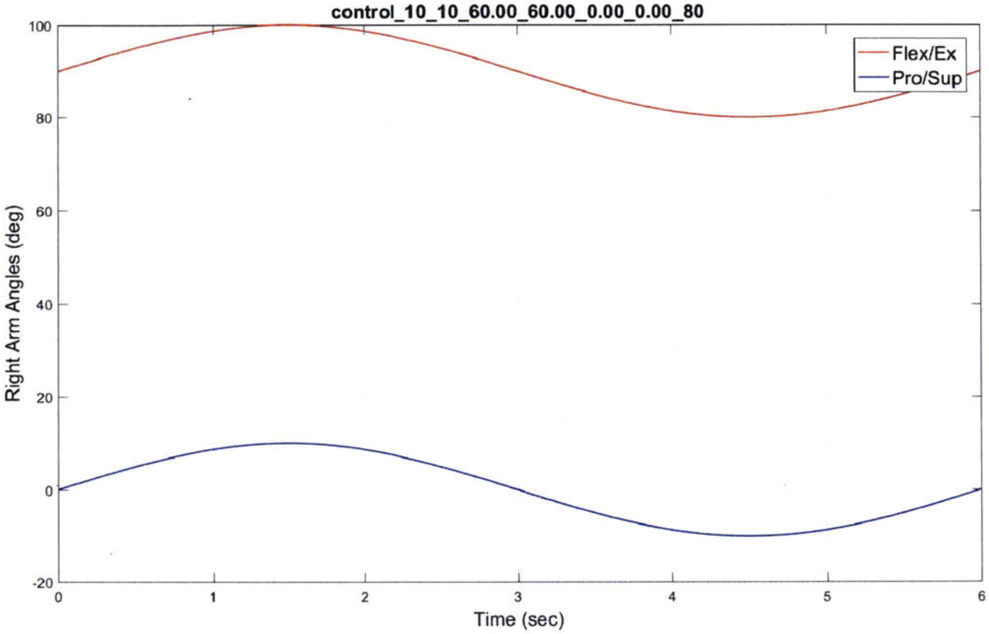

Walk
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
60
50
40
30
20
10
0
-10
-20
0
0.5
1
1.5
2
2.5
3
3.5
Time (sec)

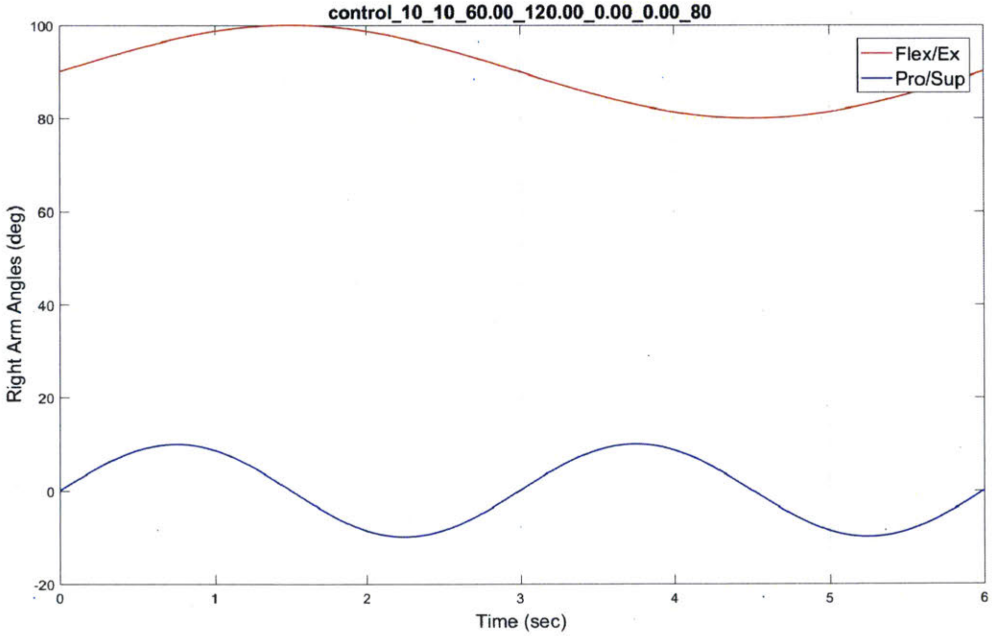

Wash Self
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
200
150
100
50
0
-50
-100
0
5
10
15
20
25
Time (sec)

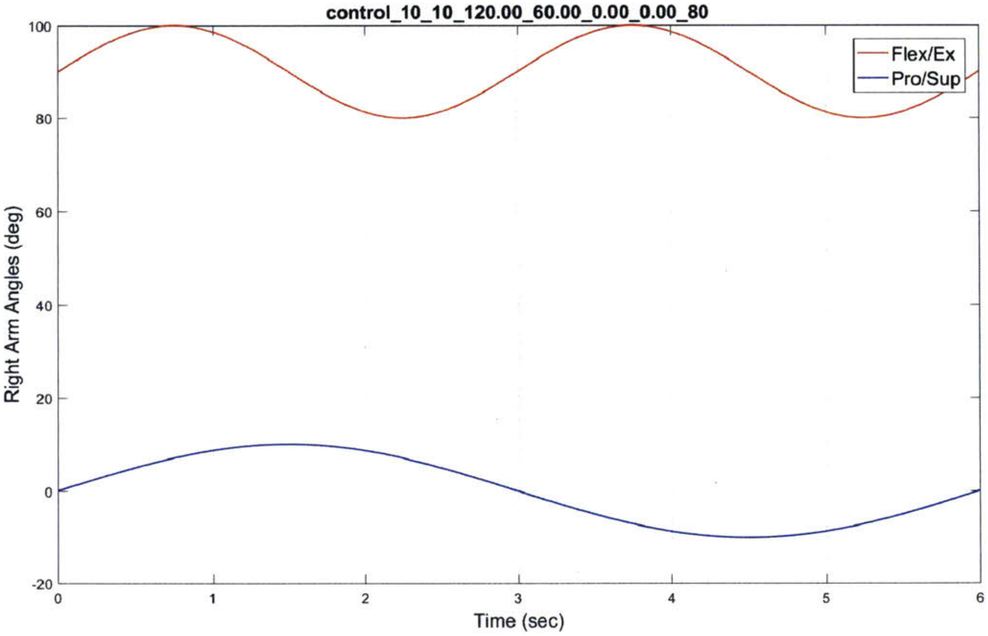
Appendix E

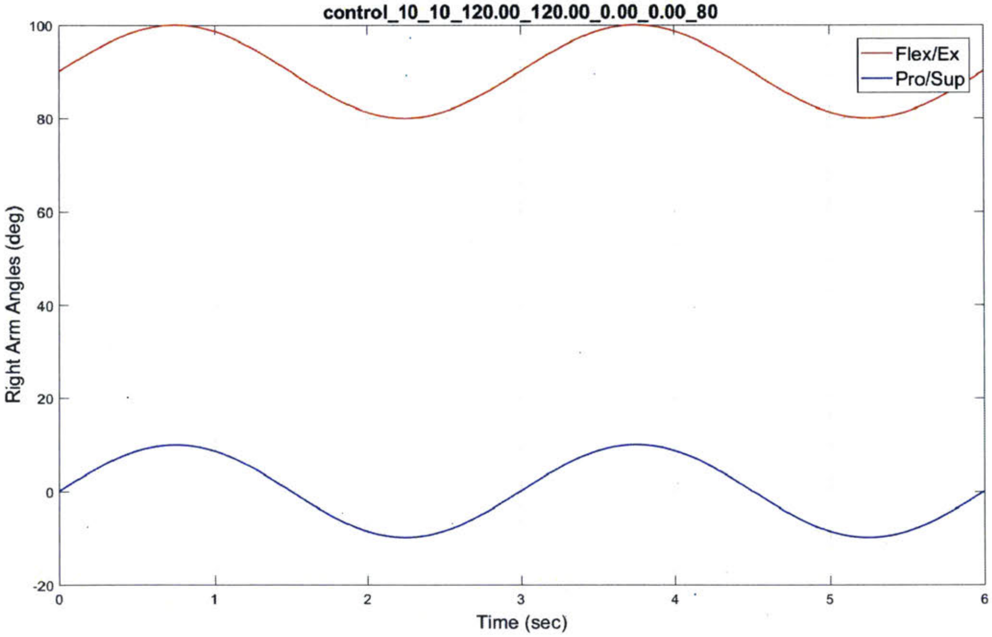
Control Motion Profiles

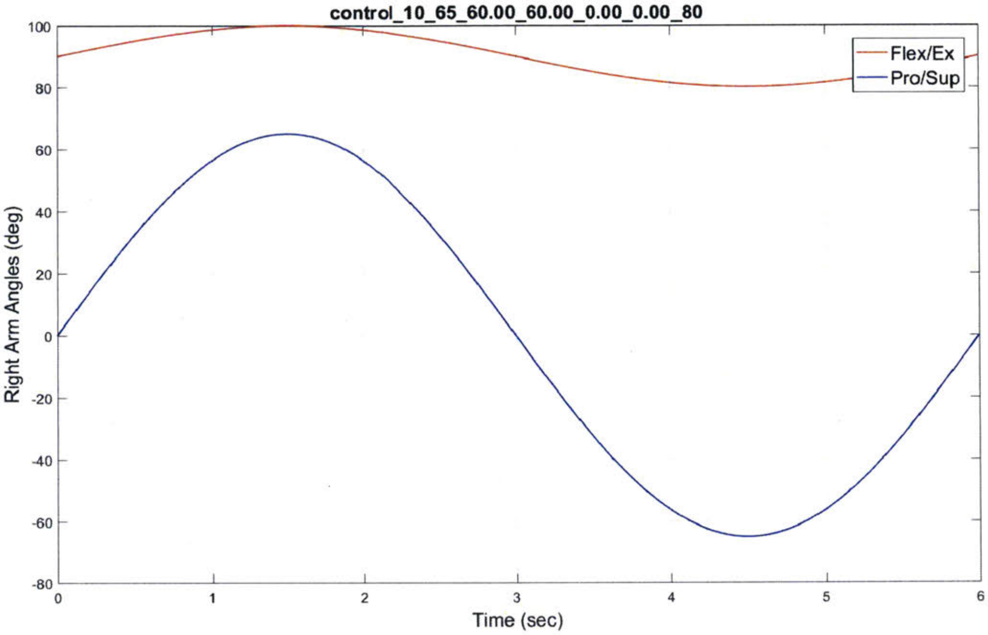
Titles of plots represent the parameters of the sine waves used to generate the control motion. These parameters are separated by underscores. They are read, in order, as: amplitude of sine wave 1 (deg), amplitude of sine wave 2 (deg), frequency of sine wave 1 (deg/s), frequency of sine wave 2 (deg/s), phase of sine wave 1 (deg), phase of sine wave 2 (deg), and sampling rate of motion (Hz). For the purposes of this study, all phase parameters are set to 0 deg, and motion were generated at a sampling rate of 80 Hz.


control_10_10_60.00_60.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
Time (sec)
100
80
60
40
20
0
-20
0
1
2
3
4
5
6

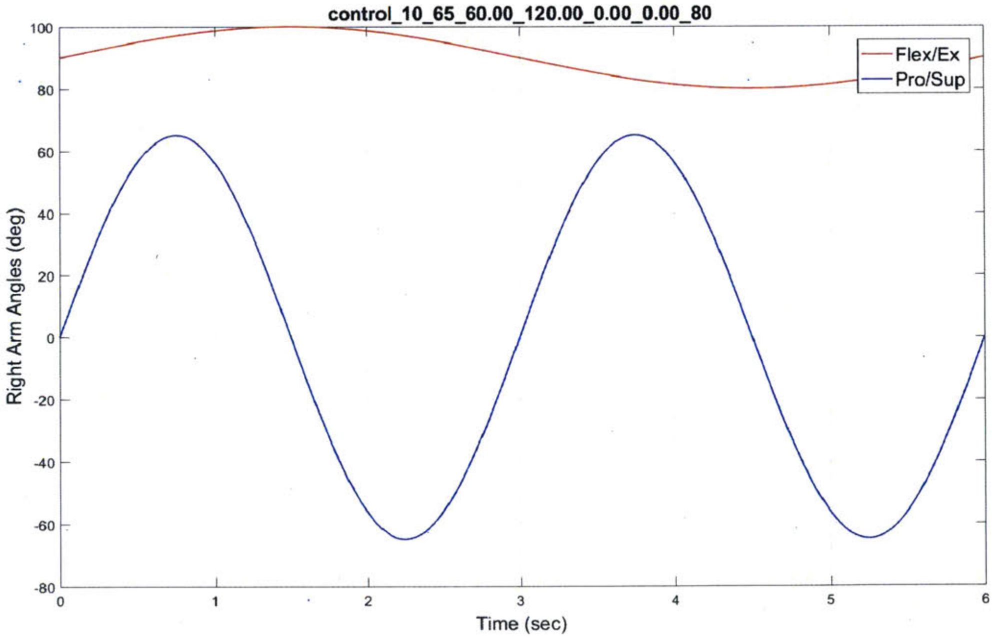

control_10_10_60.00_120.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
Time (sec)
100
80
60
40
20
0
-20
0
1
2
3
4
5
6


control_10_10_120.00_60.00_0.00_0.00_80
Right Arm Angles (deg)
Time (sec)
Flex/Ex
Pro/Sup
100
80
60
40
20
0
-20
0
1
2
3
4
5
6

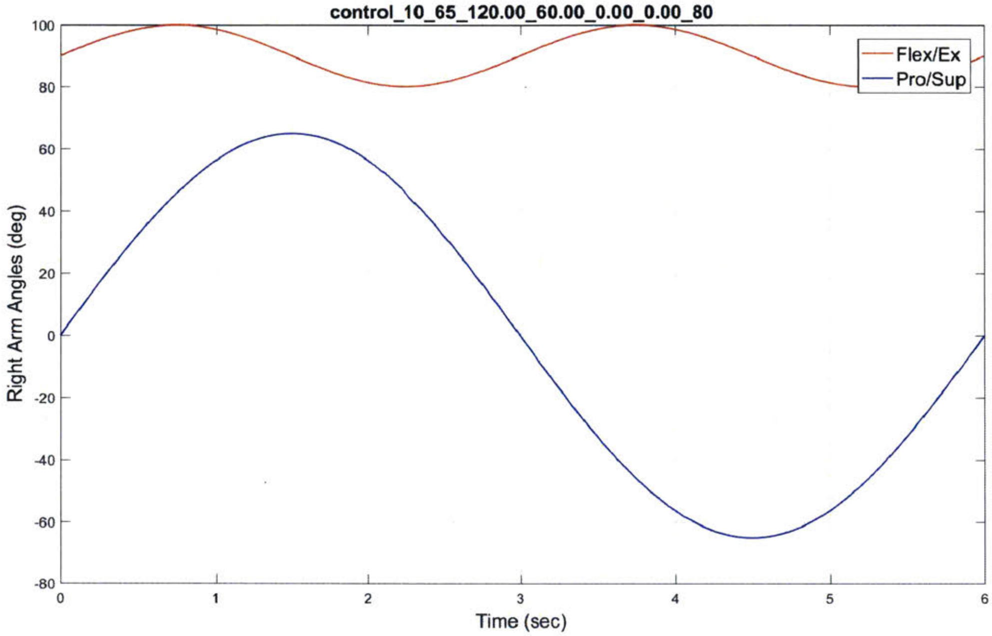

control_10_10_120.00_120.00_0.00_0.00_80
Right Arm Angles (deg)
Time (sec)
Flex/Ex
Pro/Sup
100
80
60
40
20
0
-20
0
1
2
3
4
5
6

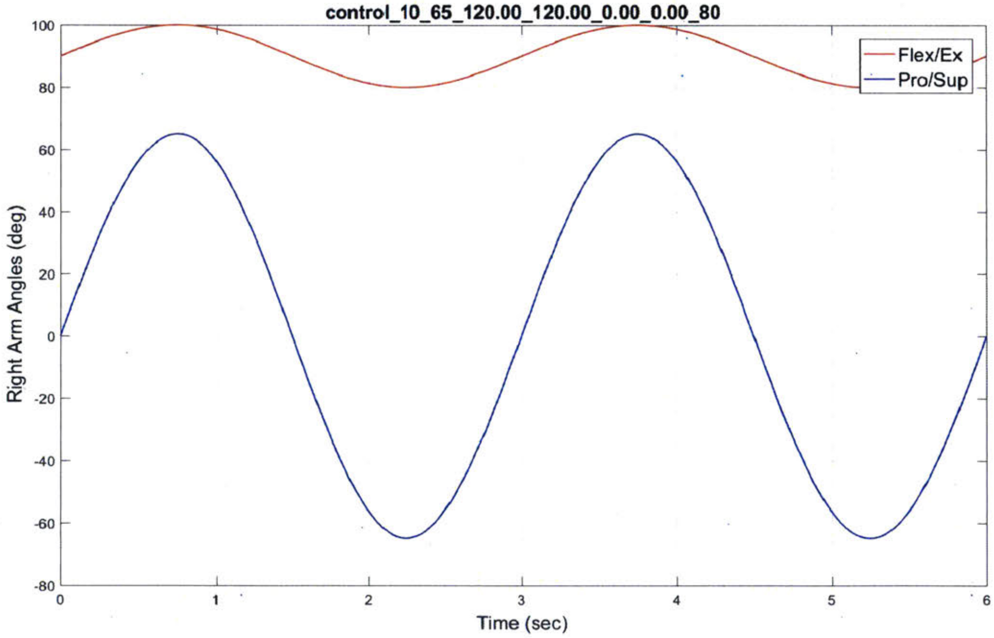


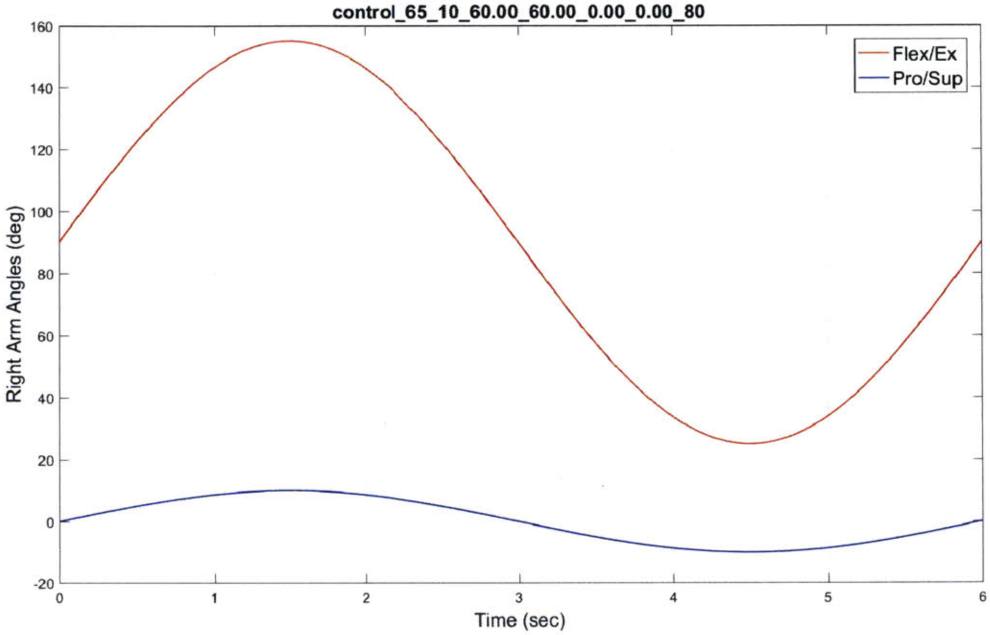
control_10_65_60.00_60.00_0.00_0.00_80

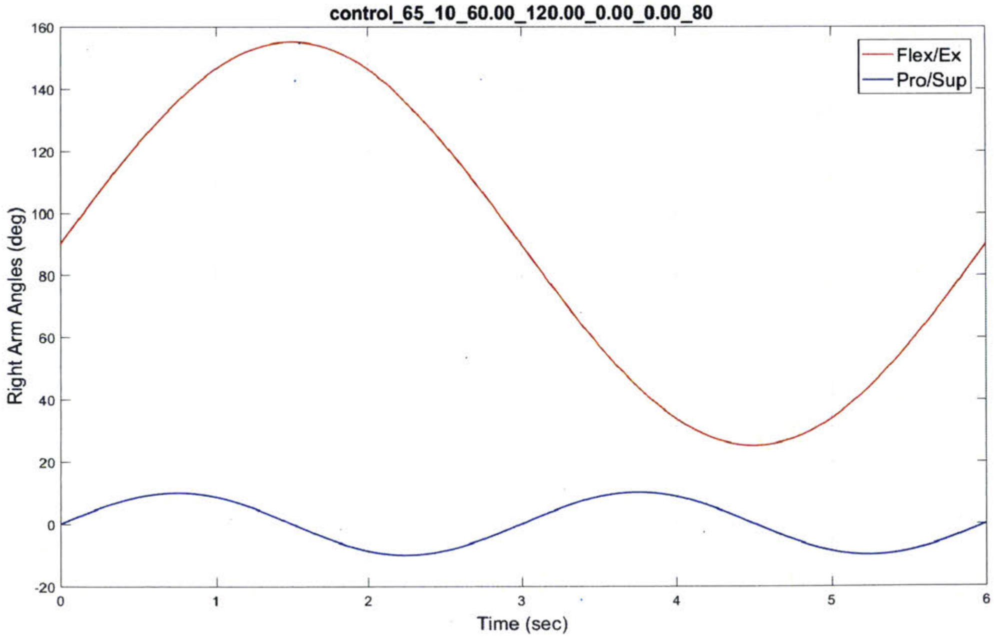


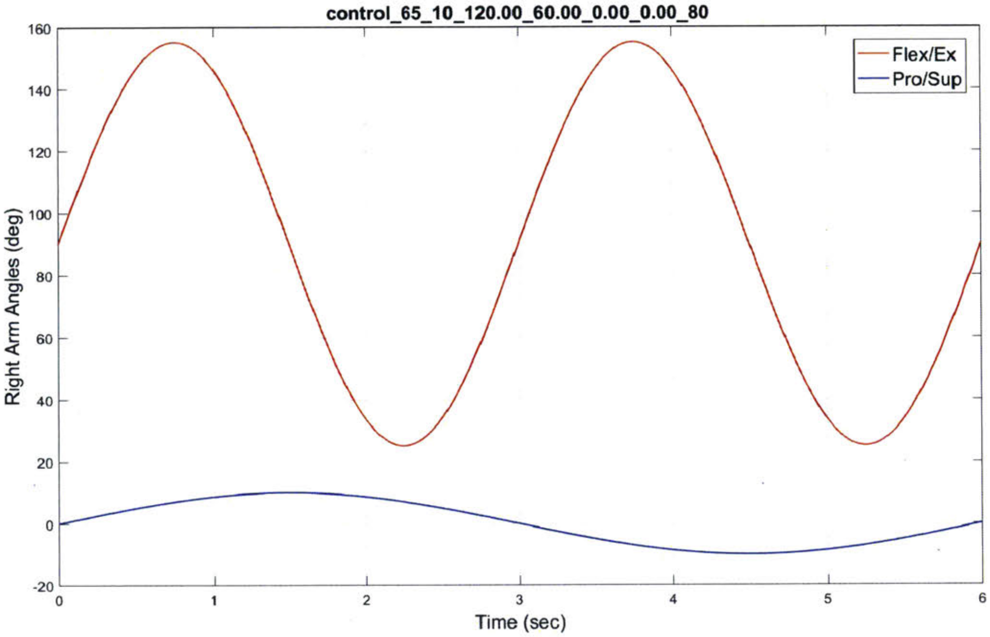
control_10_65_60.00_120.00_0.00_0.00_80

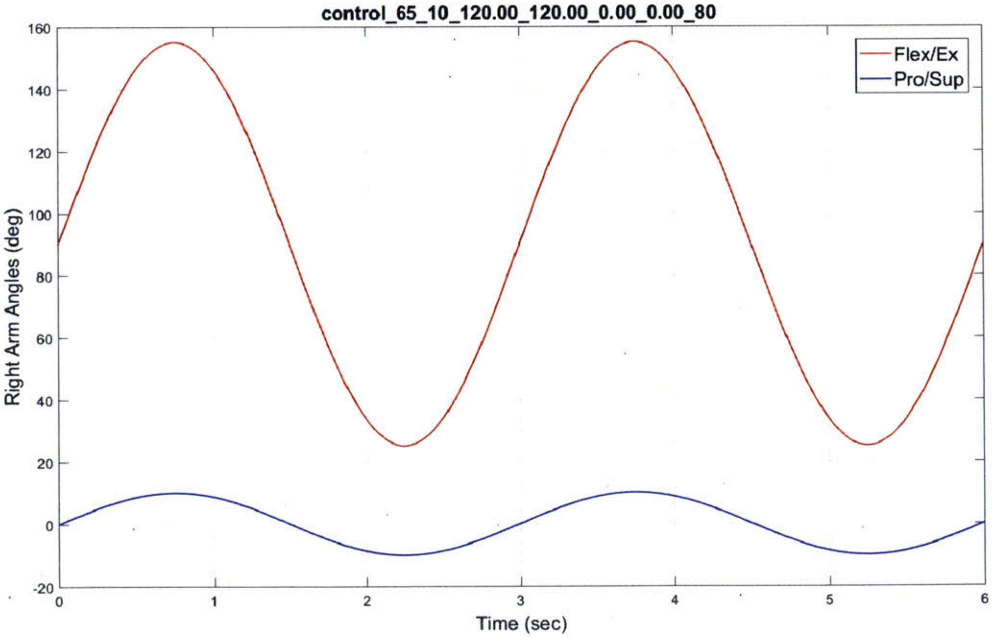

control_10_65_120.00_60.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
100
80
60
40
20
0
-20
-40
-60
-80
0
1
2
3
4
5
6
Time (sec)

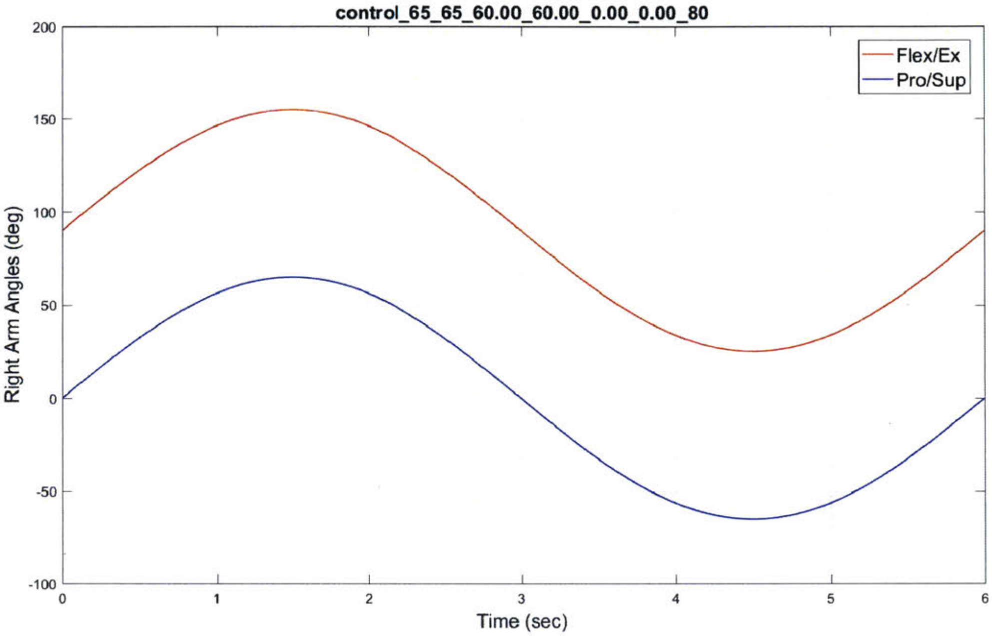

control_10_65_120.00_120.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
100
80
60
40
20
0
-20
-40
-60
-80
0
1
2
3
4
5
6
Time (sec)

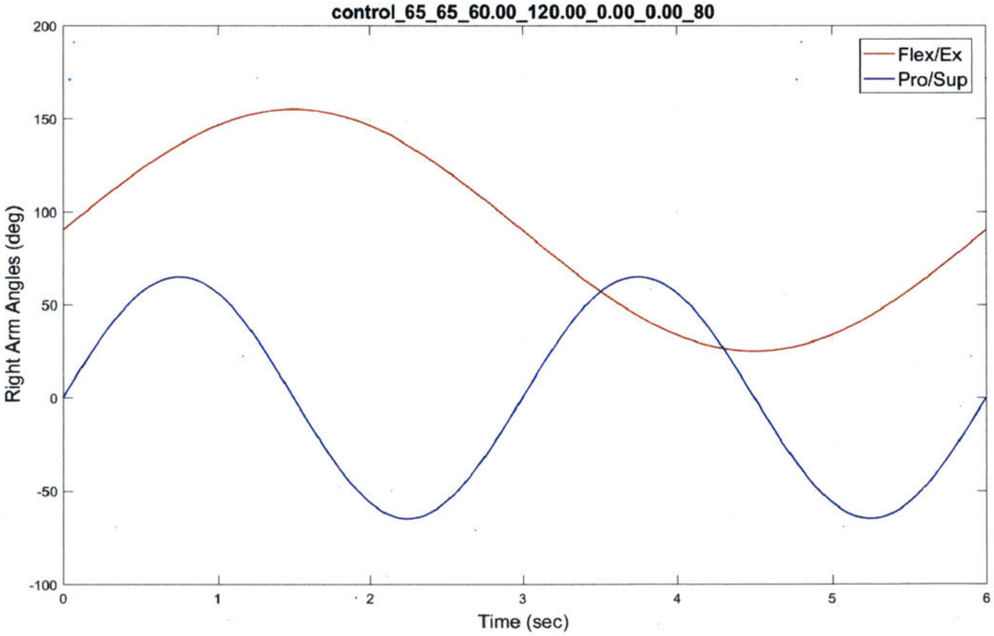

control_65_10_60.00_60.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
Time (sec)

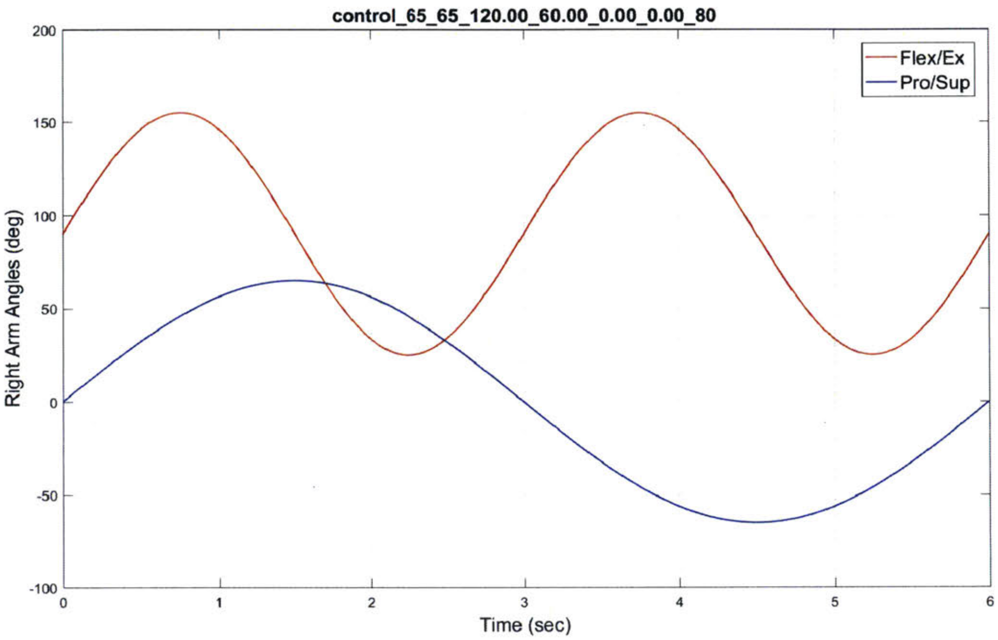

control_65_10_60.00_120.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
Time (sec)

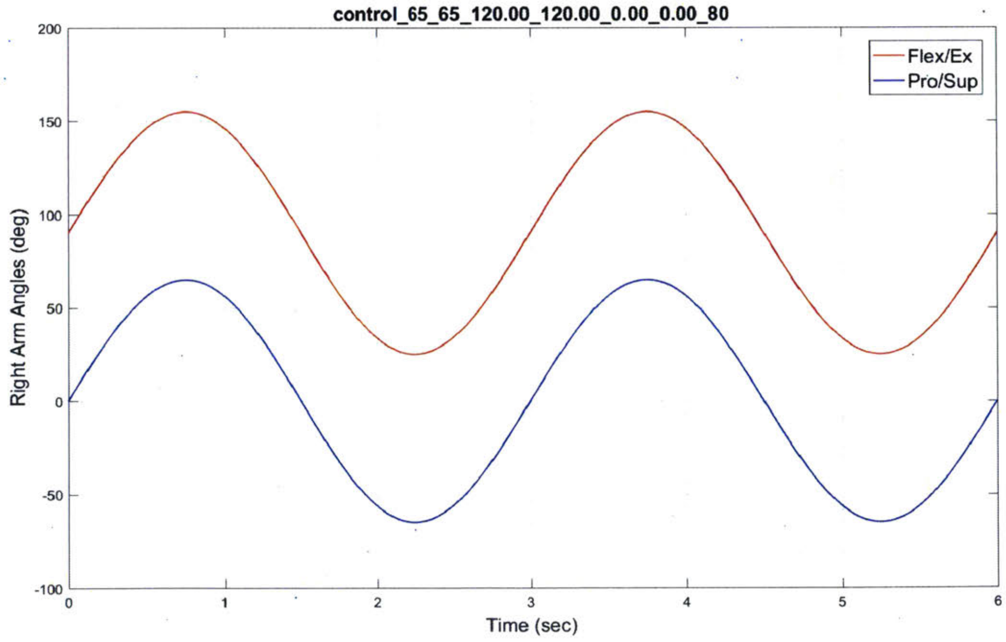

control_65_10_120.00_60.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
Time (sec)


control_65_10_120.00_120.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
Time (sec)


control_65_65_60.00_60.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
200
150
100
50
0
-50
-100
0
1
2
3
4
5
6
Time (sec)


control_65_65_60.00_120.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
200
150
100
50
0
-50
-100
0
1
2
3
4
5
6
Time (sec)


control_65_65_120.00_60.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
200
150
100
50
0
-50
-100
0
1
2
3
4
5
6
Time (sec)


control_65_65_120.00_120.00_0.00_0.00_80
Flex/Ex
Pro/Sup
Right Arm Angles (deg)
200
150
100
50
0
-50
-100
0
1
2
3
4
5
6
Time (sec)

Appendix F

Robot Servo Specifications

Table F.1 Specifications for the Dynamixel AX-12A servo from Robotis

Operating Voltage	12V
Stall Torque*	15.3 kg cm
	212 oz in
No-load Speed	59 RPM
	0.169sec/60 deg
Weight	55g
Size	32 x 50 x 40 mm
Resolution	0.29 deg
Reduction Ratio	1/254
Operating Angle	300 or Continuous Turn
Operating Voltage	9-12V (Recommended Voltage 11.1V)
Max Current	900 mA
Standby Current	50 mA
Internal Operating Temp	-5 degC $\sim$ 70 degC
Protocol	TTL Half Duplex Async Serial
Module Limit	254 valid addresses
Com Speed	7343bps $\sim$ 1Mbps
Position Feedback	Yes
Temp Feedback	Yes
Load Voltage Feedback	Yes
Input Voltage Feedback	Yes
Compliance/PID	Yes
Material	Plastic Gears and Body
Motor	Cored Motor

Table F.2 Specifications for the Dynamixel RX-24F servo from Robotis

Operating Voltage	12V
Stall Torque*	26 kg cm
	360 oz in
No-load Speed	0.079 sec/60 deg
Weight	67g
Size	50.6 x 35.6 x 35.5 mm
Resolution	0.29 deg
Reduction Ratio	1/193
Operating Angle	300 deg or Continuous Turn
Max Current	2400mA
Standby Current	50 mA
Operating Temp	-5 degC ∼ 85 degC
Protocol	RS485 Asynchronous Serial
Module Limit	254 valid addresses
Com Speed	7343bps ∼ 1Mbps
Position Feedback	Yes
Temp Feedback	Yes
Load Voltage Feedback	Yes
Input Voltage Feedback	Yes
Compliance/PID	Yes
Material	Metal Gears &
	Engineering Plastic Body
Motor	Maxon RE-MAX

Appendix G

APDM Opal IMU Specifications

Table G.1 APDM Opal IMU Specifications

	Accelerometer	**Gyroscope**	**Magnetometer**
Axes	3 axes	3 axes	3 axes
Range	16g	2000 deg/s	8 Gauss
Noise	5 mg/Hz	0.025 deg/s/Hz	2 mGauss/Hz
Sample Rate	20 to 128 Hz	20 to 200 Hz	20 to 200 Hz
Bandwidth	50 Hz	50 Hz	32.5 Hz
Resolution	14 bits	16 bits	12 bits

Appendix H

Kruskal-Wallis pairwise comparisons of flexion/extension treatments

Table H.1 Pairwise comparisons of all treatments in flexion/extension

Group(i)	Group(j)	Statistic	p-Value
0	1	4.366	0.135
0	2	10.004	0
0	3	7.457	0
0	4	-0.67	1
0	5	4.63	0.08
0	6	6.275	0.001
0	7	6.095	0.002
0	8	-6.545	0

0	9	3.563	0.457
0	10	15.861	0
0	11	-0.553	1
0	12	-0.097	1
0	13	6.607	0
0	14	13.812	0
0	15	9.067	0
1	2	11.336	0
1	3	16.833	0
1	4	-4.383	0.13
1	5	-2.998	0.753
1	6	12.436	0
1	7	13.293	0
1	8	-6.723	0
1	9	-15.026	0
1	10	16.562	0
1	11	-15.852	0
1	12	-2.51	0.925

1	13	15.2	0
1	14	15.876	0
1	15	17.241	0
2	3	-9.641	0
2	4	-12.215	0
2	5	-11.051	0
2	6	-9.368	0
2	7	-10.75	0
2	8	-13.87	0
2	9	-11.469	0
2	10	14.254	0
2	11	-12.571	0
2	12	-11.026	0
2	13	-10.608	0
2	14	5.867	0.003
2	15	-7.17	0
3	4	-12.557	0
3	5	-16.804	0

3	6	3.998	0.253
3	7	-15.747	0
3	8	-12.568	0
3	9	-16.932	0
3	10	15.985	0
3	11	-16.593	0
3	12	-7.295	0
3	13	-14.019	0
3	14	14.845	0
3	15	16.313	0
4	5	4.33	0.144
4	6	11.676	0
4	7	7.063	0
4	8	-5.798	0.004
4	9	1.707	0.998
4	10	16.441	0
4	11	-4.288	0.155
4	12	1.355	1

4	13	8.221	0
4	14	15.456	0
4	15	16.738	0
5	6	12.602	0
5	7	16.579	0
5	8	-6.973	0
5	9	-16.59	0
5	10	16.244	0
5	11	-16.59	0
5	12	-2.761	0.851
5	13	16.897	0
5	14	15.55	0
5	15	16.932	0
6	7	-10.612	0
6	8	-12.771	0
6	9	-13.577	0
6	10	16.279	0
6	11	-13.853	0

6	12	-7.212	0
6	13	-9.938	0
6	14	14.914	0
6	15	10.719	0
7	8	-9.427	0
7	9	-16.935	0
7	10	16.244	0
7	11	-16.59	0
7	12	-4.478	0.109
7	13	8.998	0
7	14	15.55	0
7	15	16.932	0
8	9	5.024	0.033
8	10	16.749	0
8	11	2.612	0.899
8	12	6.455	0.001
8	13	10.221	0
8	14	15.888	0

8	15	16.514	0
9	10	16.293	0
9	11	-16.59	0
9	12	-1.203	1
9	13	17.277	0
9	14	15.563	0
9	15	16.935	0
10	11	-16.414	0
10	12	-16.469	0
10	13	-16.241	0
10	14	-13.131	0
10	15	-16.251	0
11	12	2.619	0.897
11	13	16.932	0
11	14	15.242	0
11	15	16.59	0
12	13	5.194	0.022
12	14	14.897	0

12	15	11.762	0
13	14	15.515	0
13	15	16.932	0
14	15	-14.326	0

Appendix I

Kruskal-Wallis pairwise comparisons of pronation/supination treatments

Table I.1 Pairwise comparisons of all treatments in pronation/supination

Group(i)	Group(j)	Statistic	p-Value
0	1	7.861	0
0	2	-13.932	0
0	3	-13.639	0
0	4	2.408	0.947
0	5	-2.336	0.959
0	6	-13.601	0
0	7	-12.419	0
0	8	-13.497	0

0	9	-10.235	0
0	10	-13.697	0
0	11	-13.822	0
0	12	-6.178	0.001
0	13	16.735	0
0	14	-14.06	0
0	15	-9.938	0
1	2	-16.033	0
1	3	-15.907	0
1	4	-6.267	0.001
1	5	-8.808	0
1	6	-16.116	0
1	7	-15.113	0
1	8	-15.852	0
1	9	-13.432	0
1	10	-16.12	0
1	11	-16.189	0
1	12	-11.207	0

1	13	14.288	0
1	14	-15.852	0
1	15	-12.554	0
2	3	6.234	0.001
2	4	15.125	0
2	5	16.766	0
2	6	15.881	0
2	7	16.918	0
2	8	6.057	0.002
2	9	16.375	0
2	10	15.474	0
2	11	14.226	0
2	12	11.445	0
2	13	16.928	0
2	14	-5.667	0.006
2	15	16.458	0
3	4	14.637	0
3	5	16.324	0

3	6	3.967	0.266
3	7	13.179	0
3	8	-0.111	1
3	9	12.716	0
3	10	2.156	0.98
3	11	0.805	1
3	12	10.518	0
3	13	16.953	0
3	14	-9.271	0
3	15	15.75	0
4	5	-4.371	0.133
4	6	-14.451	0
4	7	-12.889	0
4	8	-14.579	0
4	9	-11.617	0
4	10	-14.544	0
4	11	-14.824	0
4	12	-8.048	0

4	13	16.452	0
4	14	-15.373	0
4	15	-11.403	0
5	6	-16.749	0
5	7	-14.789	0
5	8	-16.141	0
5	9	-9.596	0
5	10	-16.738	0
5	11	-16.932	0
5	12	-4.15	0.198
5	13	16.652	0
5	14	-16.814	0
5	15	-7.028	0
6	7	16.794	0
6	8	-4.879	0.046
6	9	11.7	0
6	10	-12.042	0
6	11	-16.109	0

6	12	9.347	0
6	13	16.932	0
6	14	-16.932	0
6	15	15.601	0
7	8	-13.114	0
7	9	5.449	0.011
7	10	-16.932	0
7	11	-17.277	0
7	12	8.259	0
7	13	16.932	0
7	14	-16.932	0
7	15	13.877	0
8	9	12.813	0
8	10	3.773	0.351
8	11	2.92	0.788
8	12	10.259	0
8	13	16.932	0
8	14	-10.066	0

8	15	15.405	0
9	10	-12.999	0
9	11	-14.568	0
9	12	4.35	0.139
9	13	16.932	0
9	14	-16.928	0
9	15	4.361	0.136
10	11	-12.847	0
10	12	9.834	0
10	13	16.932	0
10	14	-16.932	0
10	15	15.629	0
11	12	10.346	0
11	13	16.932	0
11	14	-16.932	0
11	15	16.158	0
12	13	16.946	0
12	14	-11.914	0

12	15	-2.972	0.765
13	14	-16.635	0
13	15	-16.932	0
14	15	16.69	0

Appendix J

Flexion/Extension two-factor pool groups

Table J.1 Two-way factor groups with median and interquartile range for flexion/extension axis response

Pool Group	**Subgroup**	**Pooling**	**Subset**	**Median**	**IQR**
1	1	$F_1; F_2$	$A_1 = 0; A_2 = 0$	1.60	3.24
1	2	$F_1; F_2$	$A_1 = 1; A_2 = 0$	2.83	0.64
1	3	$F_1; F_2$	$A_1 = 0; A_2 = 1$	16.54	18.20
1	4	$F_1; F_2$	$A_1 = 1; A_2 = 1$	3.52	3.82
2	1	$A_2; F_2$	$A_1 = 0; F_1 = 0$	5.88	21.32
2	2	$A_2; F_2$	$A_1 = 1; F_1 = 0$	2.56	1.83
2	3	$A_2; F_2$	$A_1 = 0; F_1 = 1$	4.39	6.97
2	4	$A_2; F_2$	$A_1 = 1; F_1 = 1$	3.35	1.62
3	1	$A_1; F_2$	$A_2 = 0; F_1 = 0$	2.41	1.80
3	2	$A_1; F_2$	$A_2 = 1; F_1 = 0$	4.99	20.63
3	3	$A_1; F_2$	$A_2 = 0; F_1 = 1$	2.88	1.64
3	4	$A_1; F_2$	$A_2 = 1; F_1 = 1$	6.09	4.40
4	1	$A_2; F_1$	$A_1 = 0; F_2 = 0$	3.95	6.17
4	2	$A_2; F_1$	$A_1 = 1; F_2 = 0$	3.14	0.74
4	3	$A_2; F_1$	$A_1 = 0; F_2 = 1$	7.78	21.70
4	4	$A_2; F_1$	$A_1 = 1; F_2 = 1$	2.80	2.75
5	1	$A_1; F_1$	$A_2 = 0; F_2 = 0$	2.78	1.39
5	2	$A_1; F_1$	$A_2 = 1; F_2 = 0$	4.33	2.81
5	3	$A_1; F_1$	$A_2 = 0; F_2 = 1$	2.49	2.20
5	4	$A_1; F_1$	$A_2 = 1; F_2 = 1$	7.61	21.20
6	1	$A_1; A_2$	$F_1 = 0; F_2 = 0$	3.67	3.30
6	2	$A_1; A_2$	$F_1 = 1; F_2 = 0$	3.12	1.00
6	3	$A_1; A_2$	$F_1 = 0; F_2 = 1$	2.30	4.89
6	4	$A_1; A_2$	$F_1 = 1; F_2 = 1$	5.87	5.17

Appendix K

Pronation/supination two-factor pool groups

Table K.1 Two-way factor groups with medians and interquartile range for pronation/supination axis response

Pool Group	**Subgroup**	**Pooling**	**Subset**	**Median**	**IQR**
1	1	$F_1; F_2$	$A_1 = 0; A_2 = 0$	5.98	9.40
1	2	$F_1; F_2$	$A_1 = 1; A_2 = 0$	11.01	27.47
1	3	$F_1; F_2$	$A_1 = 0; A_2 = 1$	2.26	0.67
1	4	$F_1; F_2$	$A_1 = 1; A_2 = 1$	2.90	1.09
2	1	$A_2; F_2$	$A_1 = 0; F_1 = 0$	2.30	1.08
2	2	$A_2; F_2$	$A_1 = 1; F_1 = 0$	2.65	3.96
2	3	$A_2; F_2$	$A_1 = 0; F_1 = 1$	2.46	6.21
2	4	$A_2; F_2$	$A_1 = 1; F_1 = 1$	5.04	17.95
3	1	$A_1; F_2$	$A_2 = 0; F_1 = 0$	4.82	10.20
3	2	$A_1; F_2$	$A_2 = 1; F_1 = 0$	2.25	0.40
3	3	$A_1; F_2$	$A_2 = 0; F_1 = 1$	10.61	18.80
3	4	$A_1; F_2$	$A_2 = 1; F_1 = 1$	2.63	1.28
4	1	$A_2; F_1$	$A_1 = 0; F_2 = 0$	2.50	8.13
4	2	$A_2; F_1$	$A_1 = 1; F_2 = 0$	3.31	8.05
4	3	$A_2; F_1$	$A_1 = 0; F_2 = 1$	2.27	1.04
4	4	$A_2; F_1$	$A_1 = 1; F_2 = 1$	4.40	13.28
5	1	$A_1; F_1$	$A_2 = 0; F_2 = 0$	10.58	11.74
5	2	$A_1; F_1$	$A_2 = 1; F_2 = 0$	2.40	1.02
5	3	$A_1; F_1$	$A_2 = 0; F_2 = 1$	4.52	15.02
5	4	$A_1; F_1$	$A_2 = 1; F_2 = 1$	2.28	0.36
6	1	$A_1; A_2$	$F_1 = 0; F_2 = 0$	2.94	10.77
6	2	$A_1; A_2$	$F_1 = 1; F_2 = 0$	3.26	6.30
6	3	$A_1; A_2$	$F_1 = 0; F_2 = 1$	2.31	0.47
6	4	$A_1; A_2$	$F_1 = 1; F_2 = 1$	4.87	15.60

Appendix L

Flexion/Extension pairwise comparisons of subgroups medians

Table L.1 Pairwise comparisons of subgroup medians for flexion/extension

Group 1	Group 2	p-value	z-stat	Effect size
1.1	1.2	< 0.001	7.27	0.23
1.1	1.3	< 0.001	18.64	0.59
1.2	1.3	< 0.001	19.39	0.61
1.1	1.4	< 0.001	10.4	0.33
1.2	1.4	< 0.001	9.17	0.29
1.3	1.4	< 0.001	16.4	0.52
2.1	2.2	< 0.001	6.07	0.19
2.1	2.3	0.006	2.74	0.09

2.2	2.3	< 0.001	7.1	0.22
2.1	2.4	0.002	3.17	0.1
2.2	2.4	< 0.001	13.32	0.42
2.3	2.4	0.079	1.75	0.06
3.1	3.2	< 0.001	11.89	0.38
3.1	3.3	< 0.001	6.72	0.21
3.2	3.3	< 0.001	10.71	0.34
3.1	3.4	< 0.001	18.65	0.59
3.2	3.4	0.874	0.15	0
3.3	3.4	< 0.001	16.33	0.52
4.1	4.2	0.011	2.53	0.08
4.1	4.3	< 0.001	5.02	0.16
4.2	4.3	< 0.001	6.53	0.21
4.1	4.4	0.021	2.3	0.07
4.2	4.4	< 0.001	3.47	0.11
4.3	4.4	< 0.001	6.73	0.21
5.1	5.2	< 0.001	16.29	0.52
5.1	5.3	0.361	0.91	0.03

5.2	5.3	< 0.001	14.46	0.46
5.1	5.4	< 0.001	13.03	0.41
5.2	5.4	< 0.001	8.11	0.26
5.3	5.4	< 0.001	13.82	0.44
6.1	6.2	< 0.001	4.61	0.15
6.1	6.3	< 0.001	5.8	0.18
6.2	6.3	< 0.001	5.93	0.19
6.1	6.4	< 0.001	5.42	0.17
6.2	6.4	< 0.001	11.98	0.38
6.3	6.4	< 0.001	7.77	0.25

Appendix M

Flexion/Extension pairwise comparisons of subgroup IQRs

Table M.1 Pairwise comparisons of subgroup IQR for flexion/extension

Group 1	Group 2	p-value	z-stat	Effect size
1.1	1.2	< 0.001	38.72	1.22
1.1	1.3	< 0.001	38.72	1.22
1.2	1.3	< 0.001	38.72	1.22
1.1	1.4	< 0.001	4.35	0.14
1.2	1.4	< 0.001	38.72	1.22
1.3	1.4	< 0.001	38.72	1.22
2.1	2.2	< 0.001	38.72	1.22
2.1	2.3	< 0.001	38.72	1.22

2.2	2.3	< 0.001	38.72	1.22
2.1	2.4	< 0.001	38.72	1.22
2.2	2.4	0.107	1.61	0.05
2.3	2.4	< 0.001	38.72	1.22
3.1	3.2	< 0.001	38.72	1.22
3.1	3.3	< 0.001	18.35	0.58
3.2	3.3	< 0.001	38.72	1.22
3.1	3.4	< 0.001	38.72	1.22
3.2	3.4	< 0.001	38.72	1.22
3.3	3.4	< 0.001	38.72	1.22
4.1	4.2	< 0.001	38.72	1.22
4.1	4.3	< 0.001	38.72	1.22
4.2	4.3	< 0.001	38.72	1.22
4.1	4.4	< 0.001	38.72	1.22
4.2	4.4	< 0.001	38.72	1.22
4.3	4.4	< 0.001	38.72	1.22
5.1	5.2	< 0.001	38.72	1.22
5.1	5.3	< 0.001	36.18	1.14

5.2	5.3	< 0.001	34.6	1.09
5.1	5.4	< 0.001	38.72	1.22
5.2	5.4	< 0.001	38.72	1.22
5.3	5.4	< 0.001	38.72	1.22
6.1	6.2	< 0.001	38.72	1.22
6.1	6.3	< 0.001	20.58	0.65
6.2	6.3	< 0.001	38.72	1.22
6.1	6.4	< 0.001	30.91	0.98
6.2	6.4	< 0.001	38.72	1.22
6.3	6.4	0.434	0.78	0.02

Appendix N

Pronation/supination pairwise comparisons of subgroup medians

Table N.1 Pairwise comparisons of subgroup medians for pronation/supination response

Group 1	**Group 2**	**p-value**	**z-stat**	**Effect size**
1.1	1.2	< 0.001	38.72	1.22
1.1	1.3	< 0.001	38.72	1.22
1.2	1.3	< 0.001	38.72	1.22
1.1	1.4	< 0.001	4.35	0.14
1.2	1.4	< 0.001	38.72	1.22
1.3	1.4	< 0.001	38.72	1.22
2.1	2.2	< 0.001	38.72	1.22
2.1	2.3	< 0.001	38.72	1.22

2.2	2.3	< 0.001	38.72	1.22
2.1	2.4	< 0.001	38.72	1.22
2.2	2.4	0.107	1.61	0.05
2.3	2.4	< 0.001	38.72	1.22
3.1	3.2	< 0.001	38.72	1.22
3.1	3.3	< 0.001	18.35	0.58
3.2	3.3	< 0.001	38.72	1.22
3.1	3.4	< 0.001	38.72	1.22
3.2	3.4	< 0.001	38.72	1.22
3.3	3.4	< 0.001	38.72	1.22
4.1	4.2	< 0.001	38.72	1.22
4.1	4.3	< 0.001	38.72	1.22
4.2	4.3	< 0.001	38.72	1.22
4.1	4.4	< 0.001	38.72	1.22
4.2	4.4	< 0.001	38.72	1.22
4.3	4.4	< 0.001	38.72	1.22
5.1	5.2	< 0.001	38.72	1.22
5.1	5.3	< 0.001	36.18	1.14

5.2	5.3	< 0.001	34.6	1.09
5.1	5.4	< 0.001	38.72	1.22
5.2	5.4	< 0.001	38.72	1.22
5.3	5.4	< 0.001	38.72	1.22
6.1	6.2	< 0.001	38.72	1.22
6.1	6.3	< 0.001	20.58	0.65
6.2	6.3	< 0.001	38.72	1.22
6.1	6.4	< 0.001	30.91	0.98
6.2	6.4	< 0.001	38.72	1.22
6.3	6.4	0.434	0.78	0.02

Appendix O

Pronation/supination pairwise comparisons of subgroup IQRs

Table O.1 Pairwise comparisons of subgroup IQRs for pronation/supination response

Group 1	**Group 2**	**p-value**	**z-stat**	**Effect size**
1.1	1.2	< 0.001	38.72	1.22
1.1	1.3	< 0.001	38.72	1.22
1.2	1.3	< 0.001	38.72	1.22
1.1	1.4	< 0.001	38.72	1.22
1.2	1.4	< 0.001	38.72	1.22
1.3	1.4	< 0.001	38.72	1.22
2.1	2.2	< 0.001	38.42	1.21
2.1	2.3	< 0.001	38.72	1.22

2.2	2.3	< 0.001	36.41	1.15
2.1	2.4	< 0.001	38.72	1.22
2.2	2.4	< 0.001	38.72	1.22
2.3	2.4	< 0.001	38.72	1.22
3.1	3.2	< 0.001	38.72	1.22
3.1	3.3	< 0.001	38.72	1.22
3.2	3.3	< 0.001	38.72	1.22
3.1	3.4	< 0.001	38.72	1.22
3.2	3.4	< 0.001	38.72	1.22
3.3	3.4	< 0.001	38.72	1.22
4.1	4.2	< 0.001	6.40	0.20
4.1	4.3	< 0.001	38.72	1.22
4.2	4.3	< 0.001	38.72	1.22
4.1	4.4	< 0.001	7.41	0.23
4.2	4.4	< 0.001	7.88	0.25
4.3	4.4	< 0.001	38.72	1.22
5.1	5.2	< 0.001	38.72	1.22
5.1	5.3	< 0.001	24.05	0.76

5.2	5.3	< 0.001	38.72	1.22
5.1	5.4	< 0.001	38.72	1.22
5.2	5.4	< 0.001	24.42	0.77
5.3	5.4	< 0.001	38.72	1.22
6.1	6.2	< 0.001	38.70	1.22
6.1	6.3	< 0.001	38.72	1.22
6.2	6.3	< 0.001	38.72	1.22
6.1	6.4	< 0.001	27.09	0.86
6.2	6.4	< 0.001	38.55	1.22
6.3	6.4	< 0.001	38.72	1.22

References

[1] E. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pp. 153–158, 2002.

[2] T. Seel, J. Raisch, and T. Schauer, "IMU-based joint angle measurement for gait analysis.," *Sensors (Basel, Switzerland)*, vol. 14, no. 4, pp. 6891–909, 2014.

[3] P. Muller, M.-A. Begin, T. Schauer, and T. Seel, "Alignment-Free, Self-Calibrating Elbow Angles Measurement using Inertial Sensors," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, no. 2, pp. 1–1, 2016.

[4] Y. Yang, H. Leung, L. Yue, and L. Deng, "Evaluating human motion complexity based on un-correlation and non-smoothness," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6298 LNCS, no. PART 2, pp. 538–548, 2010.

[5] T. B. Moeslund and E. Granum, "A Survey of Computer Vision-Based Human Motion Capture," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 231–268, 2001.

[6] E. Ceseracciu, Z. Sawacha, and C. Cobelli, "Comparison of markerless and marker-based motion capture technologies through simultaneous data collection during gait: Proof of concept," *PLoS ONE*, vol. 9, no. 3, 2014.

[7] 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 to create and analyze dynamic simulations of movement," *IEEE transactions on bio-medical engineering*, vol. 54, no. 11, pp. 1940–1950, 2007.

[8] N. Wiener, *Extrapolation, interpolation, and smoothing of stationary time series: with engineering applications*, vol. 47. 1949.

[9] H. W. Bode and C. E. Shannon, "A Simplified Derivation of Linear Least Square Smoothing and Prediction Theory," *Proceedings of the IRE*, vol. 38, no. 4, pp. 417–425, 1950.

[10] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. 1, p. 35, 1960.

[11] M. S. Grewal and A. P. Andrews, "Applications of Kalman filtering in aerospace 1960 to the present," *IEEE Control Systems Magazine*, vol. 30, no. 3, pp. 69–78, 2010.

[12] H. H. Kushner, "Approximations to optimal nonlinear filters," *Automatic Control, IEEE Transactions on*, vol. 12, no. October, pp. 546–556, 1967.

[13] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," in *Proceedings of the IEEE*, vol. 92, pp. 401–422, 2004.

[14] P. Del Moral, "Nonlinear filtering: Interacting particle resolution," *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, vol. 325, no. 6, pp. 653–658, 1997.

[15] J. B. Kuipers, "Quaternions and Rotation Sequences," 2000.

[16] S. J. Julier and J. K. Uhlmann, "A New Extension of the Kalman Filter to Nonlinear Systems," in *AeroSense '97*, vol. 3, pp. 182–193, 1997.

[17] M. Roth and F. Gustafsson, "An efficient implementation of the second order extended Kalman filter," in *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, pp. 1–6, 2011.

[18] S. J. Julier, "The scaled unscented transformation," in *Proceedings of the 2002 American Control Conference (IEEE Cat. No.CH37301)*, no. 2, pp. 4555–4559 vol.6, 2002.

[19] R. Van Der Merwe, A. Doucet, N. De Freitas, and E. Wan, "The Unscented Particle Filter," *Advances in Neural Information Processing Systems*, vol. 96, no. 6080, pp. 584–590, 2001.

[20] Y. Ren and X. Ke, "Particle Filter Data Fusion Enhancements for MEMS-IMU/GPS," *Intelligent Information Management*, vol. 02, no. 07, pp. 417–421, 2010.

[21] E. Pardoux, *Markov Processes and Applications*. 2005.

[22] Y. Cheng and J. L. Crassidis, "Particle filtering for attitude estimation using a minimal local-error representation," *AIAA Guidance, Navigation and Control Conference*, no. August, pp. 1–10, 2009.

[23] M. de Kraker, R. Selles, T. Schreuders, H. Stam, and S. Hovius, "Palmar Abduction: Reliability of 6 Measurement Methods in Healthy Adults," *The Journal of Hand Surgery*, vol. 34, no. 3, pp. 523–530, 2009.

[24] D. Roetenberg, H. J. Luinge, C. T. M. Baten, and P. H. Veltink, "Compensation of magnetic disturbances improves inertial and magnetic sensing of human body segment orientation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 13, no. 3, pp. 395–405, 2005.

[25] A. M. Sabatini, "Quaternion-based extended Kalman filter for determining orientation by inertial and magnetic sensing," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 7, pp. 1346–1356, 2006.

[26] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *IEEE International Conference on Rehabilitation Robotics*, 2011.

[27] J. Deutscher and I. Reid, "Articulated body motion capture by stochastic search," *International Journal of Computer Vision*, vol. 61, no. 2, pp. 185–205, 2005.

[28] J. Elwell, "Inertial navigation for the urban warrior," in *Proc. SPIE 3709, Digitization of the Battlespace IV*, p. 196, 1999.

[29] H. J. Luinge, P. H. Veltink, and C. T. M. Baten, "Estimating orientation with gyroscopes and accelerometers," *Technology and health care*, vol. 7, no. 6, p. 455, 1999.

[30] D. Giansanti, G. Maccioni, and V. Macellari, "The development and test of a device for the reconstruction of 3-D position and orientation by means of a kinematic sensor assembly with rate gyroscopes and accelerometers," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 7, pp. 1271–1277, 2005.

[31] H. Zhou and H. Hu, "Inertial motion tracking of human arm movements in stroke rehabilitation," *2005 IEEE International Conference on Mechatronics and Automation*, no. July, pp. 1306–1311, 2005.

[32] D. Young, S. D'Orey, R. Opperman, C. Hainley, and D. J. Newman, "Estimation of lower limb joint angles during walking using extended kalman filtering," *IFMBE Proceedings*, vol. 31 IFMBE, pp. 1319–1322, 2010.

[33] X. Yun and E. Bachmann, "Design, Implementation, and Experimental Results of a Quaternion-Based Kalman Filter for Human Body Motion Tracking," *Robotics, IEEE Transactions on*, vol. 22, no. 6, pp. 1216–1227, 2006.

[34] T. Seel and T. Schauer, "Joint Axis and Position Estimation from Inertial Measurement Data by Exploiting Kinematic Constraints," pp. 0–4, 2012.

[35] A. G. Cutti, A. Ferrari, P. Garofalo, M. Raggi, A. Cappello, and A. Ferrari, "'Outwalk': A protocol for clinical gait analysis based on inertial and magnetic sensors," *Medical and Biological Engineering and Computing*, vol. 48, no. 1, pp. 17–25, 2010.

[36] H. J. Luinge, P. H. Veltink, and C. T. M. Baten, "Ambulatory measurement of arm orientation," *Journal of Biomechanics*, vol. 40, no. 1, pp. 78–85, 2007.

[37] K. J. O'Donovan, R. Kamnik, D. T. O'Keeffe, and G. M. Lyons, "An inertial and magnetic sensor based technique for joint angle measurement," *J Biomech*, vol. 40, no. 12, pp. 2604–2611, 2007.

[38] E. Palermo, S. Rossi, F. Marini, F. Patan??, and P. Cappa, "Experimental evaluation of accuracy and repeatability of a novel body-to-sensor calibration procedure for inertial sensor-based gait analysis," *Measurement: Journal of the International Measurement Confederation*, vol. 52, no. 1, pp. 145–155, 2014.

[39] J. Favre, R. Aissaoui, B. M. Jolles, J. A. de Guise, and K. Aminian, "Functional calibration procedure for 3D knee joint angle description using inertial sensors," *Journal of Biomechanics*, vol. 42, no. 14, pp. 2330–2335, 2009.

[40] R. E. Mayagoitia, A. V. Nene, and P. H. Veltink, "Accelerometer and rate gyroscope measurement of kinematics: An inexpensive alternative to optical motion analysis systems," *Journal of Biomechanics*, vol. 35, no. 4, pp. 537–542, 2002.

[41] J. C. Lotters, J. Schipper, P. H. Veltink, W. Olthuis, and P. Bergveld, "Procedure for in-use calibration of triaxial accelerometers in medical applications," *Sensors and Actuators, A: Physical*, vol. 68, no. 1 -3 pt 2, pp. 221–228, 1998.

[42] S. Tadano, R. Takeda, and H. Miyagawa, "Three dimensional gait analysis using wearable acceleration and gyro sensors based on quaternion calculations," *Sensors (Basel, Switzerland)*, vol. 13, no. 7, pp. 9321–9343, 2013.

[43] E. Foxlin, "Inertial head-tracker sensor fusion by a complementary Separate-bias Kalman filter," *Proceedings of the IEEE 1996 Virtual Reality Annual International Symposium*, pp. 185–195, 1996.

[44] I. Skog, P. Handel, J. O. Nilsson, and J. Rantakokko, "Zero-Velocity Detection; An Algorithm Evaluation," *IEEE Transactions on Biomedical Engineering*, vol. 57, pp. 2657–2666, nov 2010.

[45] J. Nocedal, S. J. Wright, and SpringerLink (Online service)., *Numerical Optimization.* 2006.

[46] P. Davenport, "Rotations about Nonorthogonal Axes," *AIAA Journal*, vol. 11, no. 6, pp. 853–857, 1973.

[47] G. Piovan and F. Bullo, "On coordinate-free rotation decomposition: Euler angles about arbitrary axes," *IEEE Transactions on Robotics*, vol. 28, no. 3, pp. 728–733, 2012.

[48] G. Paraskevas, A. Papadopoulos, B. Papaziogas, S. Spanidou, H. Argiriadou, and J. Gigis, “Study of the carrying angle of the human elbow joint in full extension: a morphometric analysis,” *Surgical and Radiologic Anatomy*, vol. 26, no. 1, pp. 19–23, 2004.

[49] L. Stirling and J. MacLean, “Roadmap for the Development of at-Home Telemonitoring Systems to Augment Occupational Therapy,” *IEEE Transactions on Human Machine Systems*, vol. 46, no. 4.

[50] J. Cohen, “Statistical power analysis for the behavioral sciences,” 1988.