Sensitivity of Tremor Propagation to Model Parameters

Charles Paul Curtis Jr.
Brigham Young University

Follow this and additional works at: https://scholarsarchive.byu.edu/etd

Part of the Engineering Commons

BYU ScholarsArchive Citation
https://scholarsarchive.byu.edu/etd/9334

This Thesis is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of BYU ScholarsArchive. For more information, please contact ellen_amatangelo@byu.edu.
Sensitivity of Tremor Propagation to Model Parameters

Charles Paul Curtis Jr.

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

Steven Charles, Chair
Andrew Ning
John D. Hedengren

Department of Mechanical Engineering
Brigham Young University

Copyright © 2021 Charles Paul Curtis Jr.
All Rights Reserved
ABSTRACT

Sensitivity of Tremor Propagation to Model Parameters

Charles Paul Curtis Jr.
Department of Mechanical Engineering, BYU
Master of Science

Although Essential Tremor (ET) is the most common type of tremor, many patients are left without satisfactory treatment options. One potential alternative treatment to medication or surgery is a wearable tremor-suppressing device. However, optimizing the effectiveness of such a device requires knowledge of which muscles/joints are most responsible for tremor. To answer this question, current efforts simulate tremor propagation using a model of the neuromusculoskeletal dynamics of the upper limb. To guide efforts to identify realistic model parameters and use the model to determine the mechanical origin of tremor, we performed preliminary parameter estimation work and a thorough sensitivity analysis of this tremor propagation model.

The tremor propagation model included muscle activation inputs to the 15 major superficial muscles and joint displacement outputs in the 7 main degrees of freedom (DOF) from the shoulder to the wrist, resulting in 105 input-output relationships. We calculated the mean normalized sensitivities of all outputs to all 107 model parameters over the tremor band (4 to 8 Hz), resulting in approximately 12,000 sensitivities.

We found that sensitivities were relatively constant in the tremor band, except for shoulder adduction-abduction, which exhibited a large peak in sensitivity between 4 and 5 Hz. Averaged across the tremor band, the system was most sensitive to select elements of inertia, muscle force, muscle moment arm, damping, muscle time constants, and stiffness (in that order). The 19 highest all-input-excitation sensitivities were between 1.2 and 4.57, meaning a 100% change in parameter value produces 120-457% change in tremor. Conversely, the model includes many parameters to which the outputs are relatively insensitive. For example, the sensitivities to almost one third of the 107 parameters are below 0.1, meaning a 100% change in parameter value produces only a 10% change in tremor.

To gain additional insight, we compared the sensitivities of the full model to those of a simpler model including only two inputs and two outputs. Analyzing the two-input two-output model revealed patterns in sensitivity which persist in the full model. The sensitivities of the full model were further compared to past studies that performed rudimentary sensitivity analyses and were found to match while adding significantly more parameter-specific sensitivity information. Future work will extend this sensitivity analysis to tremor at the hand, where it matters most.

Keywords: sensitivity, tremor, propagation, model, parameter, estimation
ACKNOWLEDGEMENTS

I would like to thank Dr. Steven Charles for his ability to teach principles, then allow the freedom to explore and learn. Thanks to Dr. Andrew Ning and Dr. John Hedengren for their exceptional support and sharing their expertise freely. Thank you to Sadie Culter, Jason Huang, and Sydney Ward for their efforts that helped move the research forward. Finally, thanks to my wife Cindy and four children Abigail, Eli, Paige and Tanner for their unfailing support as we worked through this journey together. This work was supported and funded by an NIH NINDS R15 grant (NS087447).
TABLE OF CONTENTS

ABSTRACT ........................................................................................................................................... ii

LIST OF TABLES ................................................................................................................................. vi

LIST OF FIGURES ............................................................................................................................... vii

1 Introduction ........................................................................................................................................ 1

2 Postural Tremor Model ................................................................................................................... 3
  2.1 Model Structure .......................................................................................................................... 3
  2.2 Model Parameters and Their Default Values ............................................................................ 5

3 Preliminary Parameter estimation Efforts ....................................................................................... 10
  3.1 Problem Formulation ................................................................................................................ 10
  3.2 Data Pre-processing .................................................................................................................. 11
  3.3 Inverse Solution Stability ....................................................................................................... 12
  3.4 Inverse Solution Methods & Results ....................................................................................... 12
  3.5 Inverse Solution Limitations ................................................................................................... 13

4 Sensitivity Methods .......................................................................................................................... 15
  4.1.1 Single-Input-Excitation Sensitivity .................................................................................... 15
  4.1.2 All-Input-Excitation Sensitivity ......................................................................................... 16
  4.2 Simulation Protocol ................................................................................................................... 17
  4.3 Simulation Implementation ....................................................................................................... 18

5 Sensitivity Results ............................................................................................................................. 19
  5.1 Two-Input Two-Output Model .................................................................................................. 19
    5.1.1 SIE Sensitivities .............................................................................................................. 19
    5.1.2 AIE Sensitivities ............................................................................................................. 22
  5.2 Full Model .................................................................................................................................. 24
    5.2.1 SIE Sensitivities .............................................................................................................. 24
    5.2.2 AIE Sensitivities ............................................................................................................. 25

6 Discussion .......................................................................................................................................... 30
  6.1 Two-input Two-output Model .................................................................................................... 30
    6.1.1 SIE Sensitivities .............................................................................................................. 31
    6.1.2 AIE Sensitivities ............................................................................................................. 32
  6.2 Full Model .................................................................................................................................. 34
    6.2.1 SIE Sensitivities .............................................................................................................. 34
<table>
<thead>
<tr>
<th>Table 2-1:</th>
<th>Muscles included in the full model, their abbreviations, and default maximum-force values. The default maximum-force values are in the elements of the diagonal matrix $C$.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 2-2:</td>
<td>Default values of muscle moment arms in matrix $M$ for the limb configuration shown in Figure 2-1. Muscle moment arms are given in mm.</td>
</tr>
<tr>
<td>Table 2-3:</td>
<td>Default values of the inertia matrix, $I$ (kg-m$^2$).</td>
</tr>
<tr>
<td>Table 2-4:</td>
<td>Default values of the stiffness matrix, $K$ (Nm/rad).</td>
</tr>
<tr>
<td>Table 2-5:</td>
<td>Default values of the damping matrix, $D$ (Nm-s/rad).</td>
</tr>
</tbody>
</table>
Figure 2-1: Tremor propagation model. Muscle activity input in multiple muscles is transformed into muscle force in those same muscles, joint torques and eventually angular displacement in multiple degrees of freedom. Taken from [21].

Figure 2-2: Posture assumed in sensitivity analysis. The 7 main degrees of freedom from the shoulder to the wrist are indicated by their axes of rotation. Taken from [22].

Figure 5-1: SIE sensitivities of the two-input two-output model without impedance coupling (a) and with impedance coupling (b), shown for the input-output case from ECR to WFE. The tremor band (4-8Hz) is shaded purple.

Figure 5-2: Mean SIE and AIE sensitivities (left and right columns, respectively), averaged across the tremor band, of the two-input two-output model without and with impedance coupling (top and bottom rows, respectively). SIE sensitivities are shown for the input-output case from input 1 to output WFE or to output WRUD. White cells are a sensitivity exactly equal to zero.

Figure 5-3: AIE sensitivities of the two-input two-output model without impedance coupling (a) and with impedance coupling (b), shown for all inputs to output WFE. The tremor band (4-8Hz) is shaded purple.

Figure 5-4: SIE sensitivities of the full 15-input 7-output model, shown for the representative input-output case from DELT1 to SAA. The tremor band (4-8Hz) is shaded purple.

Figure 5-5: All mean SIE sensitivities (averaged across the tremor band) of the full 15-input 7-output system, labeled by parameter. Each parameter’s box is a 7-by-15 matrix containing the sensitivities of all input-output pairs (e.g., the sensitivity of output 3 to parameter $I_{2,2}$ (assuming only input 5 is active) is shown in the 3rd row and 5th column of the box labeled $I_{2,2}$).

Figure 5-6: AIE sensitivities of the full 15-input 7-output model, shown for the case from all inputs to output SFE. The tremor band (4-8Hz) is shaded purple.

Figure 5-7: AIE sensitivities from all inputs to DOF 2 showed large peaks within the tremor band.

Figure 5-8: Mean AIE sensitivities organized by parameter matrix. Each cell in a matrix has 7 sensitivities (1 for each DOF). The color bars have different scales.

Figure 5-9: Mean and standard deviation of AIE sensitivities over the tremor band. Each plot has 7 rows (1 for each DOF) and 107 columns (1 for each model parameter) so that each row has all 107 sensitivities associated with a given DOF.

Figure 5-10: The top subplot shows the maximum mean sensitivity for a given parameter which are ordered with largest sensitivity on the right. Sensitivities below 0.1 are considered negligible. The bottom subplot shows the max standard deviations of a given parameter and are ordered with greatest standard deviation on the right.

Figure 6-1: Input 1 to Output 1 without impedance coupling. The orange path shows the transfer function for SIE sensitivity for the two-input two-output system and is equivalent to a single-input single-output system.
Figure 6-2: Input 1 to output 1 with impedance coupling. The orange path demonstrates the transfer function for SIE sensitivities. Notice that input 1 has no effect on output 1. 

Figure 6-3: All inputs to output 1 without impedance coupling. The orange path shows that all inputs are active when calculating AIE sensitivities. Input 2 now effects output 1. 

Figure 6-4: All inputs to output 1 with impedance coupling. The orange path shows how AIE sensitivities are calculated. Impedance coupling greatly increase complexity of input/output relationships.
1 INTRODUCTION

Although essential tremor (ET) is the most common type of tremor, with an estimated 7 million people affected in the United States [1, 2], many patients are left without satisfactory treatment options. ET affects primarily the upper limbs and can be both postural and kinetic [3, 4]. This upper-limb tremor can make activities of daily living such as clothing, eating, and writing difficult or impossible. Currently the two main treatment options used to suppress tremor are medication and deep brain stimulation (DBS). Medications only reduce tremor by approximately 50%, and do so for only about half of the patient population [3, 5]. DBS reduces tremor by 55%-90% for approximately 70%-80% of ET patients [5]; however, due in part to the highly invasive nature of DBS, only about 1 in 30 ET patients opt for it [6]. When asked about the issues not being addressed by their current care, one of the top items reported by ET patients was “a treatment approach other than just medications and surgery” [7].

One potential alternative to medication and surgery is a wearable tremor-suppressing device, but to optimize such a device the mechanical origin and propagation of tremor needs to be known. For example, one might envision a wearable device such as a brace [8] or sleeve designed to reduce tremor through low-pass filtering [8-15] or electrical stimulation [16-19]. However, optimizing the effectiveness of such devices requires an understanding of the mechanical origin and propagation of tremor, which are currently unknown. To clarify, tremor at the hand is the result of tremorogenic muscle activity in one or multiple muscles; this muscle
activity creates tremorogenic joint torque in the degrees of freedom (DOF) spanned by those muscles; the joint torque in turn produces tremor directly in these DOF and, secondarily, in all DOF that are coupled through mechanical impedance (stiffness, damping, inertia, etc.) and afferent feedback. In this manner, the tremor propagates throughout the upper limb, resulting in tremor at the hand. Therefore, effectively suppressing tremor at the hand requires an understanding of where to intervene (which muscles or DOF), which depends on where the tremor originates mechanically and how it propagates, neither of which are known.

Models of tremor propagation (Chapter 2) can assist in determining the mechanical origin of tremor, but obtaining realistic model results depends on accurate model parameters. Therefore, we performed preliminary parameter estimation work (Chapter 3) to estimate subject-specific model parameters from model input and output data previously collected from tremor patients [22]. This preliminary work led to the realization that the model was over-parameterized, so we turned our attention to performing a thorough sensitivity analysis to determine the most important model parameters (Chapters 4-6). In addition to guiding parameter estimation efforts by focusing on the parameters of greatest importance, identifying the parameters to which the model is most sensitive will also aid in a) simulations of tremor propagation by guiding default parameter estimation efforts and in b) the design of effective treatment interventions by identifying key parameters to influence (e.g. where to add mass to optimally suppress tremor).
2 POSTURAL TREMOR MODEL

2.1 Model Structure

The neuromusculoskeletal dynamics of the upper limb were modeled using a linear time-invariant (LTI) model (Figure 2-1) which has been used in past studies of tremor propagation and transforms muscle activity into joint displacement. An LTI model is appropriate to study postural tremor because motion caused by tremor is small displacements about an equilibrium position, so neuromusculoskeletal properties may be considered constant during tremor. A more detailed description of the model is given in [20, 21] so only a summary of the model is given here. The model consists of three sub-models.

Figure 2-1: Tremor propagation model. Muscle activity input in multiple muscles is transformed into muscle force in those same muscles, joint torques and eventually angular displacement in multiple degrees of freedom. Taken from [21].

The first sub-model [23] represents the dynamics of excitation-contraction coupling and transforms muscle excitation to muscle force:

\[ T_1 T \ddot{f} + (T_1 + T_2) \dot{f} + f = Cu \]  
(2-1)
where \( \mathbf{u} \) is a time-varying vector of muscle excitation that varies between no activity (0) and maximal activity (1), \( C \) is a diagonal matrix of maximum force values that transforms muscle excitation to muscle force, \( \mathbf{f} \) is a time-varying vector of muscle forces, and \( T_1 \) and \( T_2 \) are diagonal matrices of muscle time constants. Since the matrices are all diagonal, this model is uncoupled: it transforms muscle excitation in a muscle to muscle force in that same muscle. The muscles included in the model are the 15 major superficial muscles of the upper limb (listed proximal to distal): anterior deltoid (DELT1), lateral deltoid (DELT2), posterior deltoid (DELT3), pectoralis major (PECM2), long head biceps brachii (BIClong), short head biceps brachii (BICshort), long head of triceps brachii (TRIlong), lateral head of triceps brachii (TRIlat), brachialis (BRA), brachioradialis (BRD), pronator teres (PT), flexor carpi radialis (FCR), flexor carpi ulnaris (FCU), extensor carpi radialis (brevis and longus together)(ECR), and the extensor carpi ulnaris (ECU).

The middle sub-model transforms muscle force into joint torque:

\[
\mathbf{M} \mathbf{f} = \mathbf{\tau} \tag{2-2}
\]

where \( \mathbf{M} \) is a matrix of muscle moment arms that maps muscle forces to joint torques (\( \mathbf{M} \) is equal to the transpose of the Jacobian from muscle space to joint space [23]), and \( \mathbf{\tau} \) is a time-varying vector of joint torques in the 7 major DOF from the shoulder to the wrist (listed from proximal to distal; for each DOF, the positive direction is listed first): shoulder flexion-extension (SFE), shoulder adduction-abduction (SAA), shoulder internal-external rotation (SIER), elbow flexion-extension (EFE), forearm pronation-supination (FPS), wrist flexion-extension (WFE), and wrist ulnar-radial deviation (WRUD).

The final sub-model transforms joint torque into joint displacement:
\( l\ddot{q} + D\dot{q} + Kq = \tau \) \hspace{1cm} (2-3)

where \( q \) is a time-varying vector of joint displacement in the same DOF as \( \tau \), and \( I, D, \) and \( K \) are matrices of joint inertia, damping, and stiffness respectively.

### 2.2 Model Parameters and Their Default Values

Matrices \( T_1 \) and \( T_2 \) are 15-by-15 diagonal matrices, each with 15 independent parameters. These parameters are muscle time constants representing excitation-contraction dynamics, known to be on the order of 30 ms and 40 ms, respectively [24]; we used these default values for all muscles. Matrix \( C \) is a 15-by-15 diagonal matrix, also with 15 independent parameters representing the maximum force in each muscle. Default values for \( C \) (Table 2-1) were taken from the literature [21].

**Table 2-1: Muscles included in the full model, their abbreviations, and default maximum-force values.** The default maximum-force values are in the elements of the diagonal matrix \( C \).

<table>
<thead>
<tr>
<th>Muscle</th>
<th>Abbreviation</th>
<th>Default Peak Force (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>anterior deltoid</td>
<td>DELT1</td>
<td>1218.9</td>
</tr>
<tr>
<td>lateral deltoid</td>
<td>DELT2</td>
<td>1103.5</td>
</tr>
<tr>
<td>posterior deltoid</td>
<td>DELT3</td>
<td>201.6</td>
</tr>
<tr>
<td>pectoralis major</td>
<td>PECM2</td>
<td>658.3</td>
</tr>
<tr>
<td>long head biceps brachii</td>
<td>BIClong</td>
<td>525.1</td>
</tr>
<tr>
<td>short head biceps brachii</td>
<td>BICshort</td>
<td>316.8</td>
</tr>
<tr>
<td>long head of triceps brachii</td>
<td>TRIlong</td>
<td>771.8</td>
</tr>
<tr>
<td>lateral head of triceps brachii</td>
<td>TRIlat</td>
<td>717.5</td>
</tr>
<tr>
<td>brachialis</td>
<td>BRA</td>
<td>1177.4</td>
</tr>
<tr>
<td>brachioradialis</td>
<td>BRD</td>
<td>276</td>
</tr>
<tr>
<td>pronator teres</td>
<td>PT</td>
<td>557.2</td>
</tr>
<tr>
<td>flexor carpi radialis</td>
<td>FCR</td>
<td>407.9</td>
</tr>
</tbody>
</table>
The moment-arm matrix $M$ is a 7-by-15 matrix; of its 105 elements, 57 are set to zero because the corresponding muscles do not act in the relevant DOF (e.g. extensor carpi ulnaris does not act in shoulder flexion-extension). Therefore, $M$ only has 48 independent parameters. Default values for a 50th percentile male (Table 2-2) were taken from OpenSim [25] and converted into the coordinate axes used in our model.

### Table 2-2: Default values of muscle moment arms in matrix M for the limb configuration shown in Figure 2-1. Muscle moment arms are given in mm.

<table>
<thead>
<tr>
<th></th>
<th>DELT1</th>
<th>DELT2</th>
<th>DELT3</th>
<th>PECM2</th>
<th>BIClong</th>
<th>BICshort</th>
<th>TRIlong</th>
<th>TRIlat</th>
<th>BRA</th>
<th>BRD</th>
<th>PT</th>
<th>FCR</th>
<th>FCU</th>
<th>ECR</th>
<th>ECU</th>
</tr>
</thead>
<tbody>
<tr>
<td>SFE</td>
<td>0.04</td>
<td>0.03</td>
<td>-0.03</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SAA</td>
<td>15.5</td>
<td>-34.1</td>
<td>-17.9</td>
<td>56.5</td>
<td>-5.33</td>
<td>30.7</td>
<td>6.93</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SIER</td>
<td>5.08</td>
<td>1.88</td>
<td>-8.58</td>
<td>9.61</td>
<td>5.84</td>
<td>4.19</td>
<td>-4.88</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>EFE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>45.8</td>
<td>45.8</td>
<td>-19.6</td>
<td>-19.6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FPS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-12.8</td>
<td>-12.8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5.09</td>
<td>9.95</td>
<td>1.97</td>
<td>-0.22</td>
</tr>
<tr>
<td>WFE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>14.9</td>
<td>14.9</td>
<td>-11.5</td>
<td>-6.30</td>
</tr>
<tr>
<td>WRUD</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-7.47</td>
<td>21.8</td>
<td>-17.3</td>
<td>25.1</td>
</tr>
</tbody>
</table>
Together, matrices $I$, $D$, and $K$ represent the joint impedance of the upper limb. Default values for these 7-by-7 matrices were described in detail in [20], so we provide only a summary here. Since the inertia matrix $I$ is symmetric by definition [26], it only has 28 independent parameters. These parameters were calculated from masses and lengths of body segments for a 50th percentile male [27]. In particular, default values (Table 2-3) were calculated for a single posture (Figure 2-2) using the RVC toolbox [28]. The stiffness matrix $K$ is known to be roughly symmetric with positive diagonal values [29-31]. Its off-diagonal elements represent stiffness coupling between DOF caused by multi-articular muscles. Of the 28 independent parameters of a symmetric 7-by-7 matrix, 14 are either known to be zero (because no muscles cross the corresponding DOF, e.g. SFE and WRUD) or else may be non-zero but are assumed to be negligible and/or unknown. Default values (Table 2-4) for the remaining 14 independent parameters were estimated from the literature. Past measurements of joint stiffness and damping have shown that the stiffness and damping ellipses have similar shape and orientation [32-34], indicating that the damping matrix $D$ is roughly similar to (i.e. a scaled version of) the stiffness
matrix $K$. Therefore, $D$ also has 14 independent parameters, whose default values were also estimated from the literature (Table 2-5). For more details, see [20].

Table 2-3: Default values of the inertia matrix, $I$ (kg-m2).

<table>
<thead>
<tr>
<th></th>
<th>SFE</th>
<th>SAA</th>
<th>SIER</th>
<th>EFE</th>
<th>FPS</th>
<th>WFE</th>
<th>WRUD</th>
</tr>
</thead>
<tbody>
<tr>
<td>SFE</td>
<td>0.269</td>
<td>0</td>
<td>0</td>
<td>0.076</td>
<td>0</td>
<td>0</td>
<td>-0.014</td>
</tr>
<tr>
<td>SAA</td>
<td>0</td>
<td>0.196</td>
<td>0.083</td>
<td>0</td>
<td>-0.002</td>
<td>0.009</td>
<td>0</td>
</tr>
<tr>
<td>SIER</td>
<td>0</td>
<td>0.083</td>
<td>0.079</td>
<td>0</td>
<td>0</td>
<td>0.011</td>
<td>0</td>
</tr>
<tr>
<td>EFE</td>
<td>0.076</td>
<td>0</td>
<td>0</td>
<td>0.076</td>
<td>0</td>
<td>0</td>
<td>-0.012</td>
</tr>
<tr>
<td>FPS</td>
<td>0</td>
<td>-0.002</td>
<td>0</td>
<td>0</td>
<td>0.002</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>WFE</td>
<td>0</td>
<td>0.009</td>
<td>0.011</td>
<td>0</td>
<td>0</td>
<td>0.003</td>
<td>0</td>
</tr>
<tr>
<td>WRUD</td>
<td>-0.014</td>
<td>0</td>
<td>0</td>
<td>-0.012</td>
<td>0</td>
<td>0</td>
<td>0.003</td>
</tr>
</tbody>
</table>

Table 2-4: Default values of the stiffness matrix, $K$ (Nm/rad).

<table>
<thead>
<tr>
<th></th>
<th>SFE</th>
<th>SAA</th>
<th>SIER</th>
<th>EFE</th>
<th>FPS</th>
<th>WFE</th>
<th>WRUD</th>
</tr>
</thead>
<tbody>
<tr>
<td>SFE</td>
<td>10.8</td>
<td>2.626</td>
<td>0.279</td>
<td>2.67</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SAA</td>
<td>2.626</td>
<td>5.468</td>
<td>3.821</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SIER</td>
<td>0.279</td>
<td>3.821</td>
<td>7.486</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>EFE</td>
<td>2.67</td>
<td>0</td>
<td>0</td>
<td>8.67</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FPS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.756</td>
<td>0.018</td>
<td>0.291</td>
</tr>
<tr>
<td>WFE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.018</td>
<td>0.992</td>
<td>-0.099</td>
</tr>
<tr>
<td>WRUD</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.291</td>
<td>-0.099</td>
<td>2.92</td>
</tr>
</tbody>
</table>
Table 2-5: Default values of the damping matrix, D (Nm-s/rad).

<table>
<thead>
<tr>
<th></th>
<th>SFE</th>
<th>SAA</th>
<th>SIER</th>
<th>EFE</th>
<th>FPS</th>
<th>WFE</th>
<th>WRUD</th>
</tr>
</thead>
<tbody>
<tr>
<td>SFE</td>
<td>0.756</td>
<td>0.184</td>
<td>0.02</td>
<td>0.187</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SAA</td>
<td>0.184</td>
<td>0.383</td>
<td>0.267</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SIER</td>
<td>0.02</td>
<td>0.267</td>
<td>0.524</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>EFE</td>
<td>0.187</td>
<td>0</td>
<td>0</td>
<td>0.607</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FPS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.021</td>
<td>0.001</td>
<td>0.008</td>
</tr>
<tr>
<td>WFE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.001</td>
<td>0.028</td>
<td>-0.003</td>
</tr>
<tr>
<td>WRUD</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.008</td>
<td>-0.003</td>
<td>0.082</td>
</tr>
</tbody>
</table>
3 PRELIMINARY PARAMETER ESTIMATION EFFORTS

3.1 Problem Formulation

Since parameter estimation of tremor propagation models had not been attempted previously, we began with the simplest methods. Specifically, we used linear regression of the generalized inverse problem as described in [35]. Previous researchers had collected muscle activity and joint displacement data from tremor patients [22], so we reformulated the model as

\[ G_i m_i = d_i \]  

where \( G \) is a matrix of joint displacement data, \( m \) is a vector of model parameters, \( d \) is a vector of muscle excitation data, and \( i \) represents “initial” (meaning the initial solution attempt). For details on how we formed our model into this equation, including sizes of matrices and vectors, see Appendix A.

Even though the model presented in Chapter 2 is linear in its variables, this form of the model is non-linear in its parameters. Therefore, we linearized the form around the default value of each parameter. If \( v \) represents all unknown elements of \( T_2, T_2, C, M, I, D, \) and \( K \), then \( m_i = f(v) \) can be linearized as \( m_i \approx f(v_o) + Fdv \), where \( dv = v - v_o \) and \( v_o \) is the vector of default parameter values. Therefore, the model form can be re-written as \((G_iF)(v - v_o) = d_i - G_i f(v_o)\).
The generalized inverse solution finds the value of $m$ that minimizes $\|G - d\|^2$ and $\|m\|^2$. To avoid favoring larger elements, we scaled all elements of $m$ to be of the same order of magnitude by normalizing by their default values ($v_0$):

$$(G_i F) \ast \text{repmat}(v_0, N, 1)[(v - v_0)./v_0] = d_i - G_if(v_0)$$  \hspace{1cm} (3-2)

where a period preceding an operator denotes element-wise operation, “repmat” is a MATLAB function that creates an appropriately sized matrix by repeating the values in $v_0$, and $N$ is the number of rows in $G_i F$. This allowed us to re-write the least-squares problem as

$$Gm = d$$  \hspace{1cm} (3-3)

where $G = (G_i F) \ast \text{repmat}(v_0, N, 1)$, $m = (v - v_0)./v_0$, and $d = d_i - G_if(v_0)$. Thus, each element of $100 \cdot m$ represents the percentage by which each element of $v$ varies from $v_0$.

### 3.2 Data Pre-processing

To test the feasibility of this approach, we used EMG data from the major 15 superficial muscles and joint displacement data from the 7 main DOF from the shoulder to the wrist collected from a subject with Essential Tremor. This subject’s tremor was chosen because it was large and stable. The muscle activity data was detrended (high-pass filter with cutoff frequency of 20 Hz), rectified (absolute value), low-pass filtered (20 Hz cutoff frequency), and normalized by maximum voluntary contraction (MVC). Joint displacement data was repeatedly low-pass filtered (15 Hz cut-off frequency) and differentiated to obtain the second, third, and fourth derivatives needed to construct the $G$ matrix (see Appendix A for details).
3.3 Inverse Solution Stability

When solving inverse problems, solution stability is often a concern [35]. We checked for system stability in two ways. First, we used MATLAB’s “cond” function to determine the condition number of \( G \) (the ratio of the largest singular value to the smallest singular value of \( G \)) and found that it was very large (1.1e+7), indicating that \( G \) is an ill-conditioned matrix. Second, we checked the discrete Picard condition as described in Chapter 3 of [35] and found that the rate of singular value spectrum decay qualified our problem as “severely ill-posed” (for details, see page 74 of [35]).

3.4 Inverse Solution Methods & Results

To solve the inverse problem despite its severely ill-posed nature, we performed truncated singular-value decomposition to remove small singular values and reduce the condition number of \( G \), but at a loss of resolution and introduction of bias. We also included zero-th order Tikhonov regularization to favor solutions with parameter estimates close to the default parameters as described in Chapter 4 of [35]. The regularization weight was chosen by L-curve criterion, and the effect of regularization was determined using the resolution matrix. After obtaining parameter estimates using Tikhonov regularization, we used the estimates to predict postural tremor by inputting measured muscle activity data into the model with identified parameters and comparing the predicted and measured joint displacements. Unfortunately, the match between predicted and measured data was poor. We performed Tikhonov regularization on two different time segments of the tremor data (10-20 sec and 20-30 sec) and found good correlation between the estimated parameters from each time segment, but joint displacement predictions with these estimates showed high coherence to measured joint displacements at frequencies well above the patient’s tremor frequency (5 Hz) suggesting poor parameter
estimates. Also, the model with identified parameters predicted tremors that were far larger than the measured tremors.

3.5 Inverse Solution Limitations

In hindsight, we identified several significant weaknesses with our approach of estimating parameters by solving the inverse problem. First, the inverse problem is severely ill-posed. Second, formulating the model as an inverse problem required inverting the non-square moment-arm matrix using its pseudo-inverse. This process assumes that the length of the solution is minimized, which is equivalent to minimizing muscle activity. Although this assumption is approximately true for voluntary movements in unimpaired subjects and is often used to predict muscle activity from kinematics, this assumption has not been validated for tremulous movements in patients with tremor and is unlikely to be true. Third, solving the inverse problem minimized the difference between measured and predicted muscle activity instead of measured and predicted joint tremor. Muscle activity is noisy and includes significant power at frequencies other than the tremor frequency, whereas joint tremor has been significantly low-pass filtered by the dynamics of the system. Therefore, minimizing the difference between measured and predicted joint tremor is far more preferred than minimizing the difference between measured and predicted muscle activity.

For these reasons, it would be much better to estimate parameters as a forward problem, and to validate this forward approach on a smaller, simpler model before moving on to the full model. In addition, it became clear that a thorough sensitivity analysis was needed to determine the parameters to which the system is most sensitive. Therefore, I turned my attention to a
sensitivity analysis of the full model while other lab members pursued parameter estimation of a smaller model using the forward approach.
Since the model is linear time-invariant, the relationship between each model input and each model output can be expressed as a transfer function. Together, the relationships between all model inputs and all outputs are given by a matrix of transfer functions (see Appendix for derivation):

\[
\begin{bmatrix}
q_1 \\
q_2 \\
\vdots \\
q_m \\
\end{bmatrix} =
\begin{bmatrix}
G_{1,1} & G_{1,2} & \ldots & G_{1,n} \\
G_{2,1} & G_{2,2} & \ldots & \vdots \\
\vdots & \vdots & \ddots & \vdots \\
G_{m,1} & \vdots & & G_{m,n} \\
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
\vdots \\
u_n \\
\end{bmatrix}
\]

(4-1)

where \( n \) is the number of model inputs, \( m \) is the number of outputs, and each transfer function is a function of a number of parameters (in our full model, \( n = 15 \) and \( m = 7 \)). The transfer function from input \( u_j \) \((j = 1, 2, \ldots, n)\) to output \( q_i \) \((i = 1, 2, \ldots, m)\) is \( G_{i,j} \).

### 4.1.1 Single-Input-Excitation Sensitivity

To aid in understanding the mechanical propagation of tremor and how to modify parameters to reduce tremor, we want to know how each parameter affects the prediction of tremor in each DOF caused by a single muscle. Therefore, we calculated the sensitivity of the output in each DOF to each parameter, assuming only one input is active at a time, and we called this sensitivity the “single-input-excitation (SIE) sensitivity”. If only input \( u_j \) is active, the normalized sensitivity of output \( q_i \) to parameter \( p_k \) is
\[
\frac{d(q_{i,j})}{dp_k/p_k} = \frac{d(G_{i,j}u_j)}{dp_k/p_k} = \frac{d(G_{i,j}u_j)}{dp_k} \frac{p_k}{G_{i,j}u_j}
\]

where \(q_{i,j}\) is the part of output \(q_i\) due to input \(u_j\).

Since the input is assumed to be independent of system parameters, the sensitivity becomes independent of input:

\[
\frac{d(G_{i,j})u_j}{dp_k} \frac{p_k}{G_{i,j}u_j} = \frac{d(G_{i,j})}{dp_k} \frac{p_k}{G_{i,j}}
\]

We were most interested in the sensitivity of the magnitude of the outputs (as opposed to the phase of the outputs), so we focused on the sensitivity of \(|q_{i,j}|\):

\[
\frac{d(|q_{i,j}|)}{dp_k/p_k} = \frac{d(|G_{i,j}u_j|)}{dp_k} \frac{p_k}{|G_{i,j}u_j|} = \frac{d(|G_{i,j}u_j|)}{dp_k} \frac{p_k}{|G_{i,j}|} \frac{p_k}{|u_j|} = \frac{d(|G_{i,j}|)}{dp_k} \frac{p_k}{|G_{i,j}|} \frac{p_k}{|u_j|}
\]

4.1.2 All-Input-Excitation Sensitivity

In addition, we were interested in understanding the sensitivity of each output when all muscles are active. Therefore, we also calculated the sensitivity of each output to each parameter assuming all inputs were active, and we called this the “all-input-excitation (AIE) sensitivity”. If all inputs are active, the normalized sensitivity of the magnitude of output \(q_i\) to parameter \(p_k\) is

\[
\frac{d(|q_i|)}{dp_k/p_k} = \frac{d(|G_{i,1}u_1 + G_{i,2}u_2 + \cdots + G_{i,n}u_n|)}{dp_k} \frac{p_k}{|G_{i,1}u_1 + G_{i,2}u_2 + \cdots + G_{i,n}u_n|} = \frac{d(|G_{i,1}u_1 + G_{i,2}u_2 + \cdots + G_{i,n}u_n|)}{dp_k} \frac{p_k}{|G_{i,1}u_1 + G_{i,2}u_2 + \cdots + G_{i,n}u_n|}
\]

This sensitivity depends on the inputs, which vary with time and between subjects (and unknown unless measured for a specific subject). To evaluate this sensitivity generally (for a
generic subject) and without knowledge of inputs, we assumed that all inputs were the same 
\( u_1 = u_2 = \cdots = u_n = u \); this allowed the sensitivity to be independent of inputs:

\[
\frac{d}{dp_k} \left| \left( G_{i,1} + G_{i,2} + \cdots + G_{i,n} \right) u \right| \frac{p_k}{\left| \left( G_{i,1} + G_{i,2} + \cdots + G_{i,n} \right) u \right|} \\
= \frac{d}{dp_k} \left| \left( G_{i,1} + G_{i,2} + \cdots + G_{i,n} \right) |u| \right| \frac{p_k}{\left| G_{i,1} + G_{i,2} + \cdots + G_{i,n} \right| |u|} \\
= \frac{d}{dp_k} \left| \left| G_{i,1} + G_{i,2} + \cdots + G_{i,n} \right| \right| \frac{p_k}{\left| G_{i,1} + G_{i,2} + \cdots + G_{i,n} \right|} = (4-6)
\]

4.2 Simulation Protocol

The full model has 15 inputs and 7 outputs for a total of 105 input-output relationships 
with 107 non-zero, independent parameters. Thus, the full system has 11,235 SIE sensitivities 
and 749 AIE sensitivities.

To gain conceptual and intuitive insight, we first evaluated the sensitivity of a two-input 
two-output system without impedance coupling (i.e., the diagonal terms in the impedance 
matrices \( I, D, \) and \( K \) were set equal to zero) and then with impedance coupling. Running the 
sensitivity analysis without impedance coupling decouples the system and helped ground the 
analysis in easily understood physical principles without more complex impedance interactions. 
This simplified system has four input-output relationships and a total of 17 non-zero, 
independent parameters, so there are 68 SIE sensitivities and 34 AIE sensitivities (14, 56, and 28 
without impedance coupling). The default values for these parameters were the values associated 
with the two distal-most DOF (WFE and WRUD) and the two distal-most muscles (ECR and 
ECU).
For both the two-input two-output system and the full system, we calculated all SIE and AIE sensitivities at frequencies at tremor frequencies inside the tremor band (4-8 Hz), as well as in neighboring bands (0-4 Hz and 8-15 Hz) to better observe trends.

4.3 Simulation Implementation

The sensitivity calculations were performed in the same manner for the two-input two-output system and for the full system, as follows. Combining all three sub-models, the matrix of transfer functions, \( G \), evaluated at the tremor frequency, \( \omega \), is:

\[
G = (-I\omega^2 + DJ\omega + K)^{-1} M[-T_1T_2\omega^2 + (T_1 + T_2)j\omega + 1]^{-1}C,
\]

where \( I \) is the identity matrix and \( j \) is the imaginary unit (see Appendix). This matrix was evaluated symbolically in MATLAB, yielding symbolic expressions for the 105 transfer functions. To calculate the partial sensitivity of transfer function \( G_{i,j} \) to parameter \( p_k \), default values for all other parameters were substituted into \( G_{i,j} \), after which the magnitude of \( G_{i,j} \) was determined and differentiated symbolically with respect to \( p_k \). This sensitivity was then evaluated at the default value of \( p_k \) and frequencies from 0 to 15 Hz (for details, see lines 240-256 in the main sensitivity script in Appendix C – MATLAB Code). To summarize the sensitivities relevant to tremor, we calculated the absolute value of the mean of each sensitivity across the tremor band. Some sensitivities varied significantly with frequency in the tremor band, so we also characterized the amount of this variation as the standard deviation of each sensitivity. The means and standard deviations of AIE sensitivities were determined similarly.
5 SENSITIVITY RESULTS

5.1 Two-Input Two-Output Model

To gain understanding of sensitivities in a simpler model before analyzing the full system, we computed sensitivities of a two-input two-output system.

5.1.1 SIE Sensitivities

*Without impedance coupling*

In the two-input two-output system without impedance coupling, the sensitivities of output 1 assuming only input 1 is active are relatively simple (Figure 5-1A): the sensitivities to system gains (the elements of the $C$ and $M$ matrices linking input 1 to output 1) are equal to 1 independent of frequency; the sensitivities to muscle time constants start at zero at 0 Hz and gradually change to -1 at higher frequencies; the system is sensitive to stiffness at low frequencies but not at high frequencies, whereas the system is sensitive to inertia at high frequencies but not at low frequencies, and the system is sensitive to damping at intermediate frequencies (peaking at the natural frequency of the impedance sub-model, where the sensitivities to stiffness and inertia are both zero), but not at low or high frequencies; finally, the sensitivities to other parameters are zero. The reason for zero sensitivities and other patterns mentioned here are explained in more detail in the Discussion section of this paper.
Two-Input Two-Output Partial Sensitivities Over Frequency (ECRB/ECRL to WFE)

(a)WITHOUT impedance coupling

(b) WITH impedance coupling

Figure 5-1: SIE sensitivities of the two-input two-output model without impedance coupling (a) and with impedance coupling (b), shown for the input-output case from ECR to WFE. The tremor band (4-8Hz) is shaded purple.

Averaging the magnitude of the sensitivities across the tremor band results in a single sensitivity value per parameter (Figure 5-2A). The sensitivities are relatively constant in the tremor band (especially compared to frequencies below the tremor band), so the mean sensitivities of output 1 assuming only input 1 is active (as shown in Figure 5-1A) are recognizable in the top row of Figure 5-2A. The mean sensitivities of output 2 assuming only
input 1 is active (bottom row of Figure 5-2A) are similar, but for the parameters linking input 1 to output 2. The sensitivities of output 2 assuming only input 2 is active, and the sensitivities of input 1 assuming only input 2 is active, are analogous (not shown).

Figure 5-2: Mean SIE and AIE sensitivities (left and right columns, respectively), averaged across the tremor band, of the two-input two-output model without and with impedance coupling (top and bottom rows, respectively). SIE sensitivities are shown for the input-output case from input 1 to output WFE or to output WRUD. White cells are a sensitivity exactly equal to zero.

*With impedance coupling*

The impedance matrices of the two-input two-output system are only weakly coupled (off-diagonal elements are small compared to diagonal elements), so adding impedance coupling changes the sensitivities only slightly (Figure 5-1B): some of the sensitivities that were equal to 1 independent of frequency before (without impedance coupling) are now slightly different from 1 and depend on frequency. Similarly, some of the sensitivities that were equal to zero independent of frequency before are now slightly different from zero and depend on frequency.
Otherwise, the sensitivities are similar to those of the system without impedance coupling (Figure 5-2C).

5.1.2 AIE Sensitivities

**Without impedance coupling**

In the two-input two-output system without impedance coupling, the AIE sensitivities of output 1 (i.e. assuming that both inputs are active) are similar to the SIE sensitivities of output 1 assuming that only input 1 is active (compare Figure 5-3A to Figure 5-1A). The main difference is that sensitivities to parameters linking input 2 to output 1 are now slightly non-zero (instead of zero), and sensitivities to parameters linking input 1 to output 2 are now slightly less than 1 (instead of 1). Otherwise, the AIE sensitivities are similar to SIE sensitivities (compare Figure 5-2B to Figure 5-2A).
Figure 5-3: AIE sensitivities of the two-input two-output model without impedance coupling (a) and with impedance coupling (b), shown for all inputs to output WFE. The tremor band (4-8Hz) is shaded purple.

*With impedance coupling*

As above, adding the weak impedance coupling has only a slight effect on the AIE sensitivities (compare Figure 5-3B to Figure 5-3A). Sensitivities that were independent of frequency without impedance coupling (either zero, slightly above zero, or slightly below 1) now depend on frequency. Thus, some of the sensitivities that were zero across all frequencies now average slightly above zero across the tremor band (Figure 5-2D). The normalized sensitivity to parameter $I_{1,2}$ remains zero because its default value is zero.
5.2 Full Model

Having developed understanding from the two-input two-output model, we computed sensitivities for the full 15-input 7-output model.

5.2.1 SIE Sensitivities

In the full system, SIE sensitivities preserved the patterns observed in the two-input two-output model (Figure 5-4): sensitivities to relevant system gains (elements of $M$ and $C$) are 1; the sensitivities to $T_1$ and $T_2$ are zero at 0 Hz and trend towards -1 as frequency increases; the system is sensitive to relevant stiffness elements at low frequencies, to relevant inertia elements at high frequencies, and to relevant damping elements in between; and a majority of sensitivities are zero because those parameters are not relevant to a particular input-output pair.

![Figure 5-4: SIE sensitivities of the full 15-input 7-output model, shown for the representative input-output case from DELT1 to SAA. The tremor band (4-8Hz) is shaded purple.](image)
The sensitivities are relatively constant in the tremor band, especially compared to frequencies below the tremor band (Figure 5-4), so we averaged across the tremor band and took the absolute value. The vast majority of these mean sensitivities are either zero or close to zero (Figure 5-5). In particular, the only non-zero elements in $M$ and $C$ are in columns associated with the input muscle (e.g. sensitivities to $M_{1,2}$ and $C_{2,2}$ are only non-zero in the second columns of $M$ and $C$ because no other SIE sensitivities depend on muscle 2). All of the highest SIE sensitivities are to select elements of inertia.

**Figure 5-5:** All mean SIE sensitivities (averaged across the tremor band) of the full 15-input 7-output system, labeled by parameter. Each parameter’s box is a 7-by-15 matrix containing the sensitivities of all input-output pairs (e.g., the sensitivity of output 3 to parameter $I_{2,2}$ (assuming only input 5 is active) is shown in the 3rd row and 5th column of the box labeled $I_{2,2}$).

### 5.2.2 AIE Sensitivities

The AIE sensitivities are similar to the SIE sensitivities (compare Figure 5-6 and Figure 5-4) except there are many more non-zero (though still small) sensitivities because there are many more parameters that link a given output to all inputs. For most outputs, the sensitivities...
are relatively constant over the tremor band; the exception is SAA, which shows large peaks of sensitivity in the 4-5 Hz range (Figure 5-7).

Figure 5-6: AIE sensitivities of the full 15-input 7-output model, shown for the case from all inputs to output SFE. The tremor band (4-8Hz) is shaded purple.

Figure 5-7: AIE sensitivities from all inputs to DOF 2 showed large peaks within the tremor band.
Averaging sensitivities across the tremor band revealed patterns in parameter matrices (Figure 5-8). All non-zero sensitivities associated with $T_1$ are the same, and all non-zero sensitivities associated with $T_2$ are the same (although the sensitivities to $T_1$ and $T_2$ are not identical because their default values are not identical). Generally, the highest sensitivities to elements of $C$, $M$, $D$, and $K$ are associated with distal muscles and DOF. Unlike the simpler model, the full model has strong inertial coupling, resulting in high sensitivities. In addition, because inertia can act over large distances, these high sensitivities to inertia apply to some DOF that are spatially distant. For example, the green-yellow colors in the second and third rows of the (6,3) element of the inertia matrix in Figure 5-8 indicate that tremor in SAA and SIER are very sensitive to the product of inertia that couples SIER to WFE. In contrast, joint damping and stiffness represent the effect of muscle damping and stiffness on joint DOF, so sensitivities in the $D$ and $K$ matrices are limited to DOF that share muscles and are therefore clustered close the matrix diagonal.

Figure 5-8: Mean AIE sensitivities organized by parameter matrix. Each cell in a matrix has 7 sensitivities (1 for each DOF). The color bars have different scales.
Comparing mean AIE sensitivities across all parameters (Figure 5-9A), it is clear that the DOFs with high sensitivity to a large number of parameters are SAA and SIER; for these DOF, the largest sensitivities are generally to select elements of $I$, $C$, and $M$. Because of the peaks in sensitivity of SAA between 4 and 5 Hz (Figure 5-7), sensitivities associated with SAA vary the most over the tremor band (Figure 5-9B).

**Figure 5-9**: Mean and standard deviation of AIE sensitivities over the tremor band. Each plot has 7 rows (1 for each DOF) and 107 columns (1 for each model parameter) so that each row has all 107 sensitivities associated with a given DOF.

The five largest sensitivities are to $I_{3,2}$, $I_{3,3}$, $I_{6,3}$, $C_{13,13}$, and $I_{6,2}$ with sensitivities of 4.6, 2.9, 2.7, 2.4, and 2.3, respectively (Figure 5-10). In contrast, there are many parameters with low sensitivity; for example, 33 of the 107 parameters have sensitivity below 0.1 and may potentially be ignored.
Figure 5-10: The top subplot shows the maximum mean sensitivity for a given parameter which are ordered with largest sensitivity on the right. Sensitivities below 0.1 are considered negligible. The bottom subplot shows the max standard deviations of a given parameter and are ordered with greatest standard deviation on the right.
6 DISCUSSION

With a large number of people affected by tremor and limited non-invasive treatment options it is important to understand the mechanical origin and propagation of tremor to provide more focused treatment. This sensitivity analysis is a key step on the path to understanding the mechanical origin and propagation of tremor. Specifically, understanding which parameters are most important is needed for: simulations of tremor propagation to 1) guide researchers’ efforts to identify default parameter values (i.e. to focus on accurately identifying the most important parameters) and to estimate uncertainty in simulation results; designing effective interventions (e.g. where to place mass to reduce tremor or how to design orthosis to most effectively suppress tremor); guiding and interpreting parameter estimation efforts to determine subject-specific models of tremor propagation.

6.1 Two-input Two-output Model

To gain further insight into sensitivities in general, we diagrammed the two-input two-output system and analyzed SIE and AIE sensitivities both with and without impedance coupling (Figure 6-1 to Figure 6-4)
6.1.1 SIE Sensitivities

*Without Impedance Coupling*

Figure 6-1 shows a two-input two-output model with no impedance coupling. Looking from input 1 to output 1 (the orange path) on the two-input two-output model is identical to a single-input single-output (SISO) model. The is also the path for SIE sensitivities which assumes only one active input, so the sensitivity for a SISO model is identical to the SIE sensitivity of a multiple-input multiple-output (MIMO) model. To be clear, the results in Figure 5-1a which are for the two-input two-output model with no impedance coupling are identical to the results of a SISO model which uses the same default values. By looking at Figure 6-1 it is obvious some sensitivities would be zero (e.g. the sensitivity for $M_{1,2}$, $M_{2,1}$, and $M_{2,2}$ are zero because input one is not active, so these parameters are not in the transfer function from input 1 to output 1).

![Figure 6-1: Input 1 to Output 1 without impedance coupling. The orange path shows the transfer function for SIE sensitivity for the two-input two-output system and is equivalent to a single-input single-output system.](image-url)
Adding impedance coupling causes cross-coupling between previously uncoupled muscles and DOF as shown in Figure 6-2. The orange path in this diagram is for SIE sensitivity which again assumes only one active input. It is evident from this diagram that adding impedance coupling without increasing the size of the system greatly increases the complexity of the transfer functions from input to output. With no impedance coupling the transfer function was a simple expression of seven independent parameters, but with impedance coupling it is a more complex expression of 14 independent parameters.

**Figure 6-2: Input 1 to output 1 with impedance coupling. The orange path demonstrates the transfer function for SIE sensitivities. Notice that input 1 has no effect on output 1.**

### 6.1.2 AIE Sensitivities

**Without Impedance Coupling**

The highlighted path in Figure 6-3 shows the AIE sensitivity path without impedance coupling where all inputs are assumed active. As shown, the parameters in $C$ and $M$ associated with all inputs are in the transfer function to each output so they will have non-zero sensitivities.
For example, $C_{2,2}$ and $M_{1,2}$ are in the transfer function for sensitivity to output 1 and Figure 5-2b shows non-zero values for these parameters as compared to Figure 5-2a. The other patterns found in the SIE sensitivity case are preserved, including the changes when moving from no impedance coupling to impedance coupling.

![Diagram](image)

**Figure 6-3**: All inputs to output 1 without impedance coupling. The orange path shows that all inputs are active when calculating AIE sensitivities. Input 2 now effects output 1.

With impedance coupling

For both SIE and AIE sensitivities the addition of impedance coupling causes a parameter associated directly to one output to influence the other output (Figure 6-4). For example, $M_{2,1}$, $M_{2,2}$, $C_{2,2}$, $I_{2,2}$, $D_{2,2}$, and $K_{2,2}$ are all parameters associated directly to WRUD and the sensitivity of WFE to each of these parameters without impedance coupling is zero (Figure 5-2b), but with impedance coupling the sensitivity of WFE to these same parameters is slightly non-zero (Figure 5-2d). These parameter sensitivities being only slightly non-zero means there is weak impedance
coupling between WRUD and WFE because strong impedance coupling would cause larger sensitivity values between DOF.

Figure 6-4: All inputs to output 1 with impedance coupling. The orange path shows how AIE sensitivities are calculated. Impedance coupling greatly increase complexity of input/output relationships.

6.2 Full Model

6.2.1 SIE Sensitivities

From SIE sensitivity calculations we learn that impedance coupling is an important factor in determining sensitivities of the outputs to model parameters and the results we see in Figure 5-5 suggest that when one input is active outputs are most sensitive to select elements of the inertia matrix, as all highest sensitivities are to parameters of I.
6.2.2 AIE Sensitivities

When analyzing AIE sensitivity for the full model we note that patterns found in the two-input two-output sensitivity analysis hold true, including the sum of parameter sensitivities to $M$ for a specific output being equal to the sensitivity to the element of $C$ associated with that output and the sum of all the sensitivities to parameters in $C$ for a specific output are equal to one. The top 21 parameter sensitivities are all elements of $I$, $C$, or $M$ with the max DOF sensitivity for these parameters being at or above 0.95 (Figure 5-10). All five of the top parameter sensitivities for $M$ are for elements in the three most distal DOF with four of the top five sensitivities to $C$ being for muscles associated with the four most distal DOF. Out of 14 parameters in $I$, 13 of them are in the top 35 sensitivities and 4 of the top 5 sensitivities are to parameters in $I$. These facts about sensitivity to $I$, $C$, and $M$ suggest that the outputs are most sensitive to the more distal DOF and to inertia in the system.

Looking at Figure 5-8 we see that the top sensitivities to $I$ are for diagonal and off-diagonal elements whereas the top five sensitivities for $D$ are for diagonal elements and four of the top five sensitivities to $K$ are for diagonal elements. Joint damping and stiffness stem from muscles where DOF coupling is caused by multi-articular muscles which implies that damping and stiffness sensitivities would be organized spatially where the greatest sensitivity would be to DOF that are along the diagonal or spatially close to each other. It has been found previously that ignoring the off-diagonal elements of damping and stiffness has little effect on the output of tremor ([20]) which would support spatially organized damping and stiffness sensitivities where the largest sensitivities would be to elements along the diagonals of $D$ and $K$. In contrast, inertia couples DOF that are far from each other (e.g. SIER and WFE especially because they act about a parallel axis), so sensitivities are expected to be more evenly distributed as shown in Figure
5-8. For diagonal elements of damping and stiffness the most important parameters are in WFE, WRUD, SIER, FPS and WFE, FPS, WRUD, SIER respectively which are the same DOF with most tremor as found in [22].

It is clear from the standard deviations shown in Figure 5-9 that the parameter sensitivities for SAA vary the most over the tremor band. After running the sensitivity analysis it was found that SAA had large sensitivity peaks in the tremor band (Figure 5-7) whereas other DOF have peaks outside the tremor band that are much smaller in magnitude (Figure 5-6). The highest sensitivity was to $I_{3,2}$ which means that the sensitivity of tremor in SAA (from inputs by all muscles) to inertial coupling between SAA and SIER is very high. It is important to note that the peaks are due to normalization because of dips in transfer function magnitude and not due to increases in the un-normalized sensitivity, but without normalization we could not directly compare the sensitivity of the outputs to all parameters so these peaks are acceptable in our analysis. Although this is important when modeling tremor propagation from muscles to individual joints, it may have little practical application to tremor at the hand because SAA has the lowest tremor as found by [22].

6.2.3 Past Studies

There are two past studies that performed rudimentary sensitivity analyses on parts of this model to look at how changes in model parameters affected tremor propagation patterns. First, Davidson and Charles [20] focused on the third sub-model varying values in $I$, $D$, and $K$. They were focused on tremor propagation patterns and found that the patterns were largely unchanged due to variations in $I$, $D$, and $K$ but that the magnitude of tremor was most sensitive to inertial coupling and less sensitive to damping and stiffness which is consistent with the findings of this
study. Second, Corie and Charles [21] focused on the first two sub-models varying the values of $T_1$, $T_2$, $C$, and $M$. They found that doubling and halving the values in $T_1$ and $T_2$ decreased and increased the output respectively, varying the values in $C$ simply scaled all the outputs associated with a given muscle, and that increasing and decreasing all values in $M$ together had a small effect on tremor output. Their findings for $T_1$, $T_2$, and $C$ are consistent with the findings of this study. Since they varied $M$ in its entirety they could not see that the outputs are more sensitive to certain elements of $M$ as found in this study. Through their analysis they concluded that the outputs are most sensitive to $I$ and secondly to $M$ which is consistent with the results of this study.

6.2.4 Limitations & Future Work

As mentioned in the Model Structure section of this paper the model used is an LTI model which means the model does not include non-linear dynamics or time-varying muscle properties (such as variations in damping and stiffness during muscle control). This study was focused on postural tremor so this limitation is acceptable as many of the time-varying parameters are expected to change very little while holding a posture, but we cannot expand these principles to kinetic tremor (tremor during voluntary motions). The model also does not include reflex feedback which could have an effect on tremor propagation, but is not included in this study.

In this study only a single posture was used which limits the results to only that posture. The difference in postures for this LTI model would be the default values used in the sensitivity calculations. In a future study this could be addressed by first estimating posture specific parameters for postures of interest, second determining reasonable distributions for each
parameter, and then calculating sensitivities with these parameters and associated distributions. Some default parameters have not been measured or have been assumed to be negligible and thus were not included in the sensitivity calculations for this study. Looking at the moment arm matrix \( M \), the only DOF that do not share muscles are the shoulder and wrist. Therefore, the elements of \( K \) and \( D \) that are known to be zero are six elements in the bottom left corner and, because of symmetry, six elements in the top right corner. The other elements that are zero in \( K \) and \( D \) have not been measured or were assumed to be negligible. Therefore, the default values for these specific elements of \( K \) and \( D \) were assumed to be zero because the sensitivity results would not be meaningful.

The method of normalization we used has a few limitations. As defined, a sensitivity of 1 means that a 100% change in parameter value would produce a 100% change in the output, but for some parameters a 100% change might not be realistic. Future researchers may choose to use a different method of normalization, such as using physiological parameter ranges or standard deviation rather than the default parameter value for normalization. Another limitation is that a parameter whose default value is set to zero automatically has a sensitivity of zero, which may mask important parameter sensitivities. Often it may be useful to look at absolute sensitivity, but this only allows for comparison of parameters with the same units (e.g. we can compare the sensitivity of inertial parameters to other inertial parameters, but not to stiffness or damping parameters), so we chose normalized sensitivity in this study.

While calculating the sensitivities for each joint angle in our model is an important step on the path to understanding the mechanical origin and propagation of postural tremor, it would be ideal to calculate the sensitivity of tremor magnitude at the hand. The reason this would be most useful is that tremor at the hand is what affects daily activities of tremor patients the most.
so it is most important to understand parameters that can be manipulated (by things such as wearable orthoses) to reduce total tremor output at the hand.

Future work could include 1) adding a fourth sub-model that transforms joint displacements to hand position, 2) using numeric differentiation methods (such as automatic differentiation) to improve calculation time, 3) applying a diversity of normalization methods, and 4) using parameter reduction methods (e.g. singular value decomposition) to eliminate co-linear parameters, thus reducing the parameter space to facilitate future model fitting and parameter estimation efforts.

6.2.5 Conclusion

As stated earlier in this study, this sensitivity analysis is a key to understanding the mechanical origin and propagation of postural tremor. We found agreement with past studies yet with a significant addition to parameter specific information rather than general trends. We have found that the outputs of this model are most sensitive to inertia parameters followed by certain muscle force and moment arm parameters, then by select damping, time constant, and stiffness parameters and that generally the distal DOF have the greatest sensitivity to parameters. This analysis can be used by future researchers to help guide efforts to identify key parameters for model simulation and to assist in focusing parameter estimation efforts for subject specific models.
REFERENCES


The following derivation shows how we formed our model into a least squares problem. Forming the problem this way was the base for all our preliminary parameter estimation efforts.

Caps are matrices:

\[
af + bf + f = Cu
\]  (7-1)

Where:

\[
a = t_1 t_2
\]  (7-1a)

\[
b = t_1 + t_2
\]  (7-1b)

\[t_1 \text{ and } t_2 \text{ are 15 by 15 diagonal matrices}\n\]

\[
\ddot{f} = \begin{bmatrix}
\ddot{f}_1 \\
\ddot{f}_2 \\
\vdots \\
\ddot{f}_{15}
\end{bmatrix}
\]  (7-1c)

\[
\dot{f} = \begin{bmatrix}
\dot{f}_1 \\
\dot{f}_2 \\
\vdots \\
\dot{f}_{15}
\end{bmatrix}
\]  (7-1d)

\[
f = \begin{bmatrix}
f_1 \\
f_2 \\
\vdots \\
f_{15}
\end{bmatrix}
\]  (7-1e)

\[
C = \begin{bmatrix}
C_1 & 0 & \cdots & 0 \\
0 & C_2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & C_{15}
\end{bmatrix}
\]  (7-1f)

\[
u = \begin{bmatrix}
u_1 \\
u_2 \\
\vdots \\
u_{15}
\end{bmatrix}
\]  (7-1g)
\( \ddot{\tau} = M \ddot{f} \)  \hspace{1cm} (7-2)

Where:
\[
\ddot{\tau} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_7 \end{bmatrix}
\]

\[
M = \begin{bmatrix}
M_{1,1} & M_{1,2} & \cdots & M_{1,15} \\
M_{2,1} & M_{2,2} & \cdots & M_{2,15} \\
\vdots & \vdots & \ddots & \vdots \\
M_{7,1} & M_{7,2} & \cdots & M_{7,15}
\end{bmatrix}
\]

\[
\ddot{f} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{15} \end{bmatrix}
\]

Differentiate (2) twice:

\( \dot{\ddot{\tau}} = M \dddot{f} \)  \hspace{1cm} (7-3)

\( \dddot{\tau} = M \dddot{f} \)  \hspace{1cm} (7-4)

\( \ddot{\tau} = I \ddot{q} + D \dot{q} + Kq \)  \hspace{1cm} (7-5)

Where:
\[
\dddot{\tau} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_7 \end{bmatrix}
\]

\[
I = \begin{bmatrix}
I_{1,1} & I_{1,2} & \cdots & I_{1,7} \\
I_{2,1} & I_{2,2} & \cdots & I_{2,7} \\
\vdots & \vdots & \ddots & \vdots \\
I_{7,1} & I_{7,2} & \cdots & I_{7,7}
\end{bmatrix}
\]

\[
\dddot{q} = \begin{bmatrix} \dddot{q}_1 \\ \dddot{q}_2 \\ \vdots \\ \dddot{q}_7 \end{bmatrix}
\]

\[
D = \begin{bmatrix}
D_{1,1} & D_{1,2} & \cdots & D_{1,7} \\
D_{2,1} & D_{2,2} & \cdots & D_{2,7} \\
\vdots & \vdots & \ddots & \vdots \\
D_{7,1} & D_{7,2} & \cdots & D_{7,7}
\end{bmatrix}
\]
\[ \dot{\mathbf{q}} = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_7 \end{bmatrix} \]
\[ \mathbf{K} = \begin{bmatrix} K_{1,1} & K_{1,2} & \cdots & K_{1,7} \\ K_{2,1} & K_{2,2} & \cdots & K_{2,7} \\ \vdots & \vdots & \ddots & \vdots \\ K_{7,1} & K_{7,2} & \cdots & K_{7,7} \end{bmatrix} \]
\[ \ddot{\mathbf{q}} = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_7 \end{bmatrix} \]

Differentiate (5) twice:
\[ \ddot{\mathbf{\tau}} = I \dddot{\mathbf{\tau}} + D \dddot{\mathbf{q}} + K \dddot{\mathbf{\tau}} \]
\[ \dddot{\mathbf{\tau}} = I \dddot{\mathbf{\tau}}^{(4)} + D \dddot{\mathbf{q}} + K \dddot{\mathbf{\tau}} \]

Set (2), (3), and (4) equal to (5), (6), and (7), then sub into (1)
\[ aM^+(Iq^{(4)} + D\dddot{\mathbf{q}} + K\dddot{\mathbf{\tau}}) + bM^+(I\dddot{\mathbf{q}} + D\dddot{\mathbf{q}} + K\dddot{\mathbf{\tau}}) + M^+(I\dddot{\mathbf{q}} + D\dddot{\mathbf{q}} + K\dddot{\mathbf{\tau}}) = Cu \]

Left multiply both sides of (8) by \( C^{-1} \), \( M^+ \) is the pseudoinverse of \( M \)
\[ C^{-1}(aM^+Iq^{(4)} + aM^+\dddot{\mathbf{q}} + aM^+K\dddot{\mathbf{\tau}} + bM^+I\dddot{\mathbf{q}} + bM^+\dddot{\mathbf{q}} + bM^+K\dddot{\mathbf{\tau}} + bM^+I\dddot{\mathbf{q}} + bM^+\dddot{\mathbf{q}} + bM^+K\dddot{\mathbf{\tau}} + M^+\dddot{\mathbf{q}} + M^+K\dddot{\mathbf{\tau}}) = u \]

Expand (9) and combine like terms:
\[ C^{-1}aM^+Iq^{(4)} + C^{-1}(aM^+D + bM^+I)\dddot{\mathbf{q}} + C^{-1}(aM^+K + bM^+D + M^+I)\dddot{\mathbf{\tau}} + C^{-1}(bM^+K + M^+D){\dddot{\mathbf{q}} + C^{-1}M^+K\dddot{\mathbf{\tau}}} = u \]

Define simpler terms:
\[ C_1 = C^{-1}aM^+I \]
\[ C_2 = C^{-1}(aM^+D + bM^+I) \]
\[ C_3 = C^{-1}(aM^+K + bM^+D + M^+I) \]
\[ C_4 = C^{-1}(bM^+K + M^+D) \]
\[ C_5 = C^{-1}M^+K \]

Where \( C_1, C_2, ... C_5 \) are each a 15 by 7 matrix

Substitute (10)a – (10)e into (10):
\[ C_1q^{(4)} + C_2\dddot{\mathbf{q}} + C_3\dddot{\mathbf{\tau}} + C_4\dddot{\mathbf{q}} + C_5\dddot{\mathbf{\tau}} = u \]
Form into Least squares problem:

To arrange into a least squares problem the elements of \( C_1, C_2, \ldots, C_5 \) must be placed into a single vector and the elements of \( \ddot{q}, \dot{q}, \dddot{q}, \dddot{q}, \dddot{q}^{(4)} \) must be placed into a single matrix. A simplified example is given below to develop the notation needed.

Assuming \( C_1, C_2, \ldots, C_5 \) are 3 by 2 matrices and \( \dot{q}, \ddot{q}, \dddot{q}, \dddot{q}, \ddot{q}^{(4)} \) are 2 by 1 vectors:

\[
C_i = \begin{bmatrix} C_{1,1} & C_{1,2} & C_{1,3} \\ C_{2,1} & C_{2,2} & C_{2,3} \\ C_{3,1} & C_{3,2} & C_{3,3} \end{bmatrix}_i
\]

\[
\ddot{C}_i = \begin{bmatrix} C_{1,1} \\ C_{1,2} \\ C_{1,3} \\ C_{2,1} \\ C_{2,2} \\ C_{2,3} \\ C_{3,1} \\ C_{3,2} \end{bmatrix}_i
\]

Where \( i = 1 \) to 5.

\[
Q^{(j)} = \begin{bmatrix} q_1^{(j)} & q_2^{(j)} & 0 & 0 & 0 & 0 \\ 0 & 0 & q_1^{(j)} & q_2^{(j)} & 0 & 0 \\ 0 & 0 & 0 & 0 & q_1^{(j)} & q_2^{(j)} \end{bmatrix}
\]

Where \( f(0 \) to 5) represents the \( j^{th} \) time derivative, is represented in dot notation for the first three time derivatives \( (\dot{q}, \ddot{q}, \dddot{q}) \) and is \( Q \) for \( j = 0 \).

Given equations (13) and (14):

\[
\dot{\mathbf{m}} = \ddot{d}
\]

Where,

\[
\mathbf{G} = [Q^{(4)} \quad \dddot{Q} \quad \dddot{Q} \quad \ddot{Q} \quad Q]
\]

\[
\ddot{m} = \begin{bmatrix} \ddot{C}_1 \\ \ddot{C}_2 \\ \ddot{C}_3 \\ \ddot{C}_4 \\ \ddot{C}_5 \end{bmatrix}
\]

\[
\ddot{d} = \ddot{u}
\]
This can be expanded to the full problem:

\[
C_i = \begin{bmatrix}
C_{1,1} & C_{1,2} & \cdots & C_{1,7} \\
C_{2,1} & C_{2,2} & \cdots & C_{2,7} \\
\vdots & \vdots & \ddots & \vdots \\
C_{15,1} & C_{15,2} & \cdots & C_{15,7}
\end{bmatrix}
\] (7-16)

\[
\tilde{C}_i = \begin{bmatrix}
C_{1,1} \\
C_{1,2} \\
\vdots \\
C_{1,7} \\
C_{2,1} \\
C_{2,2} \\
\vdots \\
C_{15,1} \\
C_{15,2} \\
\vdots \\
C_{15,7}
\end{bmatrix}
\] (7-17)

Where \(i = 1\) to \(5\), and each \(\tilde{C}_i\) is a 105 by 1 vector.

\[
Q^{(j)} = \begin{bmatrix}
(q^{(j)})^T & \tilde{0} & \cdots & \tilde{0} \\
\tilde{0} & (q^{(j)})^T & \cdots & \vdots \\
\vdots & \vdots & \ddots & \tilde{0} \\
\tilde{0} & \cdots & \tilde{0} & (q^{(j)})^T
\end{bmatrix}
\] (7-18)

Where \((q^{(j)})^T\) is the transpose of the \(j^{th}\) time derivative of \(q\) and is a 1 by 7 vector. Each \(\tilde{0}\) is also a 1 by 7 vector. This makes each \(Q^{(j)}\) a 15 by 105 matrix.

**Full Least-Squares Problem Statement:**

\[G\tilde{m} = \tilde{d}\] (7-19)

Where \(G\), \(m\), and \(d\) are as defined in equations (15a), (15b), and (15c) respectively. The dimensions for each are as follows:

- \(G\) is a 15 by 525 matrix
- \(\tilde{m}\) is a 525 by 1 vector
- \(\tilde{d}\) is a 15 by 1 vector
8.1 Transfer Function Matrix Derivation

1. \( T_1T_2\ddot{f} + (T_1 + T_2)\dot{f} + f = Cu \)

2. \( M\dot{f} = \tau \)

3. \( I\dddot{q} + D\dot{q} + Kq = \tau \)

Take the Laplace transform of 1, 2, and 3

4. \( T_1T_2F_s^2 + (T_1 + T_2)Fs + F = CU \)

5. \( MF = T \)

6. \( IQs^2 + DQs + KQ = T \)

Where \( T \) is capital tau.

Solve 4 for \( F \)

7. \( F = [T_1T_2s^2 + (T_1 + T_2)s + 1]^{-1}CU \)

where 1 is unity for a single-input single-output system or an appropriately sized identity matrix for multi-input multi-output systems.

Sub 6 and 7 into 5

8. \( M[T_1T_2s^2 + (T_1 + T_2)s + 1]^{-1}CU = IQs^2 + DQs + KQ \)
Solve 8 for $\frac{Q}{u}$

$$M[T_1T_2s^2 + (T_1 + T_2)s + 1]^{-1}CU = (Is^2 +Ds + K)Q$$

$$Q = (Is^2 +Ds + K)^{-1} M[T_1T_2s^2 + (T_1 + T_2)s + 1]^{-1}CU$$

9. \( G = (Is^2 +Ds + K)^{-1} M[T_1T_2s^2 + (T_1 + T_2)s + 1]^{-1}C \)

Substitute \( j\omega \) for \( s \)

10. \( G = (I(j\omega)^2 + D j\omega + K)^{-1} M[T_1T_2(j\omega)^2 + (T_1 + T_2)j\omega + 1]^{-1}C \)

Simplify,

11. \( G = (-I\omega^2 + D j\omega + K)^{-1} M[-T_1T_2\omega^2 + (T_1 + T_2)j\omega + 1]^{-1}C \)

11 is the transfer function matrix. \( G \) will have as many rows as outputs in the system and as many columns as the inputs to the system.
The full scripts including plotting are available on github: https://github.com/cpc-curtis/Postural-Tremor-Sensitivity-Analysis-Tools

Main Sensitivity Calculation Script

```matlab
1     %% Readme, clearing workspace, closing figures
2     % This will calculate single-input-excitation (SIE) sensitivities for a
3     % 2-input 2-output system or a 15-input 7-output system. In the "System
4     % size selection" section choose the size system you want by
5     % uncommenting
6     % the appropriate line for "mysys" and comment out the other one. For the
7     % 2-input 2-output system you can choose to run with or without
8     % impedance coupling by toggling the "noImpedanceCoupling" variable between 1
9     % (WITH impedance coupling) and 0 (WITHOUT impedance coupling). Choose
10    % whether to
11    % calculate sensitivities or just plot by toggling "plotOnly" between 1
12    % (load previously calculated sensitivities and plot) and 0 (calculate
13    % new
14    % sensitivities and plot). Choose to calculate new transfer function
15    % matrix
16    % or load a previously calculated matrix by toggling "calcT2" (1 =
17    % calculate new transfer function matrix, 0 = load previously
18    % calculated
19    % transfer function. The Seval3_norm structure that is created is the
20    % main
21    % sensitivity result we care about. It is saved as a structure where
22    % the
23    % elements of the structure are the parameter names (e.g.
24    % Seval3_norm(1,1).Ill.data would give you a 150 element vector which
```
% the sensitivity of output 1 to I11 with only input 1 active for SIE
% sensitivities and would be the sensitivity of output 1 to I11 with all
% inputs active for AIE sensitivities.

clear; close all;

%% Get default parameters
default_params

%% System size selection
plotOnly = 0; % 1 to skip calculations and just plot
calcT2 = 0; % 1 would calculate new transfer function matrix rather than load previously calculated transfer functions
AIE_Sens = 1; % 1 calculates all-input-excitation sensitivities (sums rows of transfer function matrix)

% mysys = '2x2'; % 2-input 2-output model
noImpedanceCoupling = 0; % 1 to eliminate cross terms in I,D,K (only for 2-input 2-output system)

mysys = '15x7'; % 15-input 7-output model

%% Mimo 2x2
if strcmp(mysys,'2x2')
    % Grab muscle for wrist flexion
t1val = t1_full(14,14);
t2val = t2_full(14,14);
C2temp  = C_full(14:15,14:15);
C2val = [C2temp(1,1),C2temp(2,2)];

    % Grab moment arms for wrist flexion
M2temp = M_full(6:7,14:15);
M2val = [M2temp(1,1),M2temp(1,2),
        M2temp(2,1),M2temp(2,2)];

    % Grab I, D, K for wrist flexion - extension

I2temp = I_full(6:7,6:7);
I2val = [I2temp(1,1),I2temp(1,2),I2temp(2,2)];
D2temp = D_full(6:7,6:7);
D2val = [D2temp(1,1),D2temp(1,2),D2temp(2,2)];
K2temp = K_full(6:7,6:7);
K2val = [K2temp(1,1),K2temp(1,2),K2temp(2,2)];
if noImpedanceCoupling == 1
    I2val = [I2val(1),0,I2val(3)];
    D2val = [D2val(1),0,D2val(3)];
    K2val = [K2val(1),0,K2val(3)];
end

% Define symbolic variables
syms t1 t2 real;
t1 = sym('t1','real');
t2 = sym('t2','real');
syms t1 t2; I2 = [I11, I12;
    I12, I22]; I2var = [I11, I12, I22];
syms D11 D12 D22;
D2 = [D11, D12;
    D12, D22]; D2var = [D11,D12,D22];
syms K11 K12 K22;
K2 = [K11, K12;
    K12, K22]; K2var = [K11,K12,K22];
syms M11 M12 M21 M22;
M2 = [M11, M12;
    M21, M22]; M2var = [M11,M12,...
    M21,M22];
syms C11 C22;
C2 = [C11, 0;
C2var = [C11,C22];
w2 = sym('w2','real');

% Define values to replace symbolic variables
myvals = [t1val t2val M2val I2val D2val K2val w2];
myvars = [t1 t2 M2var I2var D2var K2var w2];
end

%% MIMO 15x7
if strcmp(mysys,'15x7')
    % Grab muscle for wrist flexion
    t1val = t1_full(14,14);
t2val = t2_full(14,14);
    C2val = [C_full(1,1), C_full(2,2), C_full(3,3), C_full(4,4),
             C_full(5,5), C_full(6,6), C_full(7,7), C_full(8,8), C_full(9,9),
             C_full(10,10), C_full(11,11), C_full(12,12), C_full(13,13), C_full(14,14),
             C_full(15,15)];

    % Grab moment arms for wrist flexion
    M2val = [M_full(1,1), M_full(1,2), M_full(1,3), M_full(1,4),
             M_full(1,5), M_full(1,6), M_full(1,7),...]
             M_full(2,1), M_full(2,2), M_full(2,3), M_full(2,4),
             M_full(2,5), M_full(2,6), M_full(2,7),...]
             M_full(3,1), M_full(3,2), M_full(3,3), M_full(3,4),
             M_full(3,5), M_full(3,6), M_full(3,7),...]
             M_full(4,5), M_full(4,6), M_full(4,7), M_full(4,8),
             M_full(4,9), M_full(4,10), M_full(4,11), M_full(4,12), M_full(4,13),
             M_full(4,14), M_full(4,15),...]
             M_full(5,5), M_full(5,6), M_full(5,10), M_full(5,11),
             M_full(5,12), M_full(5,13), M_full(5,14), M_full(5,15),...]
             M_full(6,12), M_full(6,13), M_full(6,14), M_full(6,15),...]
             M_full(7,12), M_full(7,13), M_full(7,14), M_full(7,15)];

    % Grab I, D, K for wrist flexion - extension
    I2val = [I_full(1,1), I_full(2,2), I_full(3,3),
             I_full(4,1), I_full(4,2), I_full(4,3), I_full(5,2), I_full(5,3), I_full(5,5), I_full(6,2), I_full(6,3),
             I_full(6,6), I_full(7,1), I_full(7,4), I_full(7,7)];
D2val = [D_full(1,1), D_full(2,1), D_full(2,2), D_full(3,1),
D_full(3,2), D_full(3,3), D_full(4,1), D_full(4,4), D_full(5,5), D_full(6,5),
D_full(6,6), D_full(7,5), D_full(7,6), D_full(7,7)];

K2val = [K_full(1,1), K_full(2,1), K_full(2,2), K_full(3,1),
K_full(3,2), K_full(3,3), K_full(4,1), K_full(4,4), K_full(5,5), K_full(6,5),
K_full(6,6), K_full(7,5), K_full(7,6), K_full(7,7)];

% Define frequency vector
w2val = (0.1:0.1:15)*2*pi;

% Define symbolic variables

I2 = [I11, 0, 0, I41, 0, 0, I71; 0, I22, I32, 0, I52, I62, 0; 0, I32, I33, 0, 0, I63, 0; I41, 0, 0, I44, 0, 0, I74; 0, I52, 0, 0, I55, 0, 0; 0, I62, I63, 0, 0, I66, 0; I71, 0, 0, I74, 0, 0, I77];


K2 = [K11, K21, K22, K31, K32, K33, K41, K44, K55, K65, K66, K75, K76, K77];

K2var = [K11, K21, K22, K31, K32, K33, K41, K44, K55, K65, K66, K75, K76, K77];
syms D11 D21 D22 D31 D32 D33 D41 D44 D55 D65 D66 D75 D76 D77 real

D2 = [D11, D21, D31, D41, 0, 0, 0; D21, D22, 0, 0, 0, 0, 0; D31, D32, D33, 0, 0, 0, 0; D41, 0, 0, D44, 0, 0, 0; 0, 0, 0, 0, D55, D65, D75; 0, 0, 0, 0, D65, D66, D76; 0, 0, 0, 0, 0, D75, D76, D77];

D2var = [D11, D21, D22, D31, D32, D33, D41, D44, D55, D65, D66, D75, D76, D77];


syms M45 M46 M47 M48 M49 M410 M411 M412 M413 M414 M55 M56 M510 M511 M512 M513 M514 M515 M612 M613 M614 M615 real

syms M712 M713 M714 M715 real

M2 = [ M11, M12, M13, M14, M15, M16, M17, 0, 0, 0, 0, 0, 0; M21, M22, M23, M24, M25, M26, M27, 0, 0, 0, 0, 0; M31, M32, M33, M34, M35, M36, M37, 0, 0, 0, 0, 0; 0, 0, 0, 0, M45, M46, M47, M48, M49, M410, M411, M412, M413, M414, M415; 0, 0, 0, 0, M55, M56, 0, 0, 0, 0, M510, M511, M512, M513, M514, M515; 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, M612, M613, M614, M615; 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, M712, M713, M714, M715];

M2var = [ M11, M12, M13, M14, M15, M16, M17,... M21, M22, M23, M24, M25, M26,... M31, M32, M33, M34, M35, M36,... M45, M46, M47, M48, M49, M410, M411, M412,... M55, M56, M510, M511, M512, M513, M514,... M612, M613, M614,... M712, M713, M714,...]
M712, M713, M714, M715];

syms C11 C22 C33 C44 C55 C66 C77 C88 C99 C1010 C1111 C1212 C1313 C1414 C1515

C2 = [C11, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0; 0, C22, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0; 0, 0, C33, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0; 0, 0, 0, C44, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0; 0, 0, 0, 0, C55, 0, 0, 0, 0, 0, 0, 0, 0, 0; 0, 0, 0, 0, 0, C66, 0, 0, 0, 0, 0, 0, 0, 0; 0, 0, 0, 0, 0, 0, C77, 0, 0, 0, 0, 0, 0, 0; 0, 0, 0, 0, 0, 0, 0, C88, 0, 0, 0, 0, 0, 0; 0, 0, 0, 0, 0, 0, 0, 0, C99, 0, 0, 0, 0, 0; 0, 0, 0, 0, 0, 0, 0, 0, 0, C1010, 0, 0, 0, 0; 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, C1111, 0, 0, 0; 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, C1212, 0, 0, 0; 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, C1313, 0, 0; 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, C1414; 0; 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, C1515];

C2var = [C11, C22, C33, C44, C55, C66, C77, C88, C99, C1010, C1111, C1212, C1313, C1414, C1515];

w2 = sym('w2','real');

% Define values to replace symbolic variables
myvals = [t1val t2val M2val I2val D2val K2val w2];
myvars = [t1 t2 M2var C2var I2var D2var K2var w2];
end

%% Define frequency vector and create transfer function matrix
w3 = (0.1:0.1:15.0)*2.0*pi;
if plotOnly
    % Do nothing, just skip to plotting
else
    tic
    if calcT2
        T2 = inv(-I2.*w2.^2 + D2*1i.*w2 + K2)*M2*inv(-(t1*t2).*w2.^2 + (t1+t2).*1i.*w2 + 1)*C2;
    else
        disp('Loading Transfer Function Matrix')
        load('T2.mat','T2')
    end
    toc
end

if AIE_Sens
    tempT2 = T2;
    clear T2
    for ii = 1:size(tempT2,1)
        T2(ii,1) = sum(tempT2(ii,:));
    end
end

%% Loop to run through transfer function matrix and calculate single-
input-excitation (SIE) sensitivity for each parameter
if plotOnly
    % Do nothing, just skip to plotting
else
    tic
    f1 = waitbar(0,'Please wait...','Name','Calculating Sensitivities...');
for idx = 1:length(myvars)-1 % Index for the number of parameters
    f2 = waitbar(0,'Please wait...','Name','Working on Rows...');
    f2.Position = [508.5000  443.2500  270.0000 56.2500];
    for jdx = 1:size(T2,1) % Index for rows in Transfer function matrix (# of outputs)
        f3 = waitbar(0,'Please wait...','Name','Working on Columns...');
        f3.Position = [780.0000 443.2500 270.0000 56.2500];
        for kdx = 1:size(T2,2) % Index for columns in Transfer function matrix (# of inputs)
            % Define temporary vectors of parameters and their values, then
            % sub the values into the transfer function for all parameters
            % except the parameter of interest.
            tempmyvars = [myvars(1:idx-1) myvars(idx+1:end)];
            tempmyvals = [myvals(1:idx-1) myvals(idx+1:end)];
            tempT2 = subs(T2(jdx,kdx),tempmyvars,tempmyvals);

            % Find the numerator and denominator of the transfer function,
            % then take the magnitude.
            [num,den] = numden(tempT2);
            absT2num = sqrt(real(num).^2 + imag(num).^2);
            absT2den = sqrt(real(den).^2 + imag(den).^2);
            absT2 = absT2num./absT2den;

            % Take the derivative with respect to the parameter of interest, calculate the normalized sensitivity, then sub in
            % value for parameter of interest and the frequency vector.
            S2_norm = diff(absT2,myvars(idx))*myvars(idx)/absT2;
            Seval2_norm = subs(S2_norm,myvars(idx),myvals(idx));
            Seval3_norm(jdx,kdx).(char(myvars(idx))).data = double(subs(Seval2_norm,w2,single(w3)));
waitbar(kdx/size(T2,2), f3, [num2str(kdx/size(T2,2)*100), '% Complete (' num2str(kdx), ' of ' num2str(size(T2,2)), ')]
end

close(f3)

waitbar(jdx/size(T2,1), f2, [num2str(jdx/size(T2,1)*100), '% Complete (' num2str(jdx), ' of ' num2str(size(T2,1)), ')]
end

close(f2)

waitbar(idx/length(myvars), f1, [num2str(idx/length(myvars)*100), '% Complete (' num2str(idx), ' of ' num2str(length(myvars)), ')]
end

close(f1)
toc
end

%% Load or Save sensitivity matrix
if plotOnly
  % Load sensitivity data
  if strcmp(mysys,'2x2')
    if noImpedanceCoupling == 1
      file2Load = 'TwobyTwo_SIE_Sensitivity_No_Impedance_Coupling.mat';
    else
      file2Load = 'TwobyTwo_SIE_Sensitivity.mat';
    end
  elseif strcmp(mysys,'15x7')
    file2Load = 'Full_SIE_Sensitivity_S_Only.mat';
  else
    disp('No matching system size, and no file loaded.')
  end
  else
    if strcmp(mysys,'2x2')
      if noImpedanceCoupling == 1
        save('TwobyTwo_SIE_Sensitivity_No_Impedance_Coupling.mat','Seval3_norm','T2')
      else
        save('TwobyTwo_SIE_Sensitivity_No_Impedance_Coupling.mat','Seval3_norm','T2')
      end
    end

60
save('TwobyTwo_SIE_Sensitivity.mat','Seval3_norm','T2')
elseif strcmp(mysys,'15x7')
    save('Full_SIE_Sensitivity.mat','Seval3_norm','T2')
    save('Full_SIE_Sensitivity_S_Only.mat','Seval3_norm')
else
    disp('No matching system size, and no file saved.')
end
end

%% Plotting - Not currently functioning in conjunction with this script, use Plot_Outputs_2by2_Main and Plot_Outputs_Main instead
if strcmp(mysys,'2x2')
    Plot_Outputs_2by2_Main
elseif strcmp(mysys,'15x7')
    Plot_Outputs_Main
else
    disp('No matching system size, and nothing plotted')
end

Default Parameters

% Define size of system
n = 15;
DOF = 7;

%% t1, t2, C, a, b
% t1 full = eye(15)*30e-3; %s
% t2 full = eye(15)*40e-3; %s
C_full = diag([1218.9, 1103.5, 201.6, 658.3, 525.1, 316.8, 771.8, 717.5,...
1177.4, 276, 557.2, 407.9, 479.8, 589.8, 192.9]); % Newtons
for idx = 1:n
    t1(idx,idx) = t1_full(idx,idx);
    t2(idx,idx) = t2_full(idx,idx);
    C(idx,idx) = C_full(idx,idx);
end

%% M
% M full = 1e-3*[0.0390675, 0.0310464, -0.0287745, 0.0097311, 0.0081247,
0.013329, -0.016065, 0, 0, 0, 0, 0, 0, 0, 0, 0,... % mm because of the 1e-3
15.5, -34.1, -17.9, 56.5, -5.33,
30.7, 6.93, 0, 0, 0, 0, 0, 0, 0, 0;...]
%% I, D, K
I_full = [ 0.269, 0, 0, 0.076, 0, 0, -0.014;...
kgm^2
0, 0.196, 0.083, 0, -0.002, 0.009, 0;...
0, 0.083, 0.079, 0, 0, 0.011, 0;...
0.076, 0, 0, 0.076, 0, 0, -0.012;...
0, -0.002, 0, 0, 0.002, 0, 0;...
0, 0.009, 0.011, 0, 0, 0.003, 0;...
-0.014, 0, 0, -0.012, 0, 0, 0.003];

D_full = [0.756, 0.184, 0.020, 0.187, 0, 0, 0;...
Nms/rad
0.184, 0.383, 0.267, 0, 0, 0, 0;...
0.020, 0.267, 0.524, 0, 0, 0, 0;...
0.187, 0, 0, 0.607, 0, 0, 0;...
0, 0, 0, 0, 0.021, 0.001, 0.008;...
0, 0, 0, 0, 0.001, 0.028, -0.003;...
0, 0, 0, 0, 0.008, -0.003, 0.082];

K_full = [10.80, 2.626, 0.279, 2.670, 0, 0, 0;...
Nm/rad
2.626, 5.468, 3.821, 0, 0, 0, 0;...
0.279, 3.821, 7.486, 0, 0, 0, 0;...
2.670, 0, 0, 8.670, 0, 0, 0;...
0, 0, 0, 0, 0.756, 0.018, 0.291;...
0, 0, 0, 0, 0.018, 0.992, -0.099;...
0, 0, 0, 0, 0.291, -0.099, 2.920];