Input Dependence in Bio-Constrained Neural Control with an Eye Toward Human-Like Adaptability

Tiffany Hamstreet

Master's Project, Mechanical Engineering

Portland State University

July 13, 2018

Abstract - The human balance control system adapts continuously to changes in system parameters, such as mass, height, and musculature due to growth, injury, or circumstance. An inverted pendulum model of human balance about the ankle joint has been used to fit classical control models to experimental human balance data. Fits to experimental data indicate increases in control gains in response to increases in stimulus amplitudes and system parameters mass and height, a phenomenon attributed to adaptive control gains in the human control system. A neural network controller has been developed to match the frequency response of a single human dataset using classical control theory.

With the future goal of developing an adaptive, biologically constrained neural network controller which matches the trends of human balance data under variable conditions, this work seeks to understand the basic functionality of the governing non-spiking neural population theory and its implications in neural control systems. A neural controller is developed step-by-step, sources of input dependence are revealed, and possible analogs to the input dependence seen in human data are discussed. Classical control theory is shown to be a useful guideline in developing neural controllers and comparing fits to those obtained by fitting human data but can lead to incorrect conclusions about the actual control gains provided by the neural network due to the influence of input dependence. The relationship between input amplitude and fitted gain values in the neural balance control system is shown to follow trends seen in fits to human balance data. Challenges and research opportunities for integrating adaptive circuitry to the neural controller are discussed.

Table of Contents

1. Introduction

The human neuromuscular balance control system adapts continuously to changes in system parameters, such as mass, height, and musculature due to growth, injury, or circumstance. Identification of the mechanisms present in the human nervous system to perform such adaptive control will improve the accuracy of models of physiological nervous systems as well as the biomimetic realism of electronic control systems used in medicine and rehabilitation.

The human nervous system is believed to maintain upright stability through a feedback system involving visual, vestibular and proprioceptive senses. A wealth of human balance control data has been collected under a variety of test conditions including joint constraints and sensory deprivation for which analysis is conducted using classical control theory [1]. Using an inverted pendulum model of human balance, these analyses have fitted a number of classical controllers to human data, the simplest of which is the proportional derivative (PD) controller. These analyses have detected input dependence over the frequency range over which humans can maintain upright stability, in which the overall system gain increases with decreasing stimulus amplitude. Furthermore, they have detected an increase in fitted control gains with increasing stimulus amplitude and a decrease with increasing mass and height. It is therefore believed that the human balance control system performs adaptive control by varying control gains in response to internal parameters or external circumstance.

With the future goal of developing an adaptive, biologically constrained neural network controller which performs human-like balance control, this work uses a neural network PD controller to explore the basic functionality of the governing mean neural population theory and its implications for adaptive control. This controller is comprised of neural subnetworks, whose biologically constrained parameters are tuned to approximate basic mathematical operations [2]. Due to the presence of non-linear input dependence in the fundamental theory, the neural controller is shown to exhibit input dependence similar to that seen in human data. The impact of this input dependence in the neural controller is that control gains vary continuously as a function of their biological parameters.

Since classical control theory depends on control gains being constant, its usefulness as a means of neural control system analysis is explored. Non-constant gains result in fitted control equations finding inaccurate gains, which do not necessarily coincide with mean system gains as verified through direct measurement. While fitted gains are inaccurate, they do provide a useful method of finding best fits to the data and detecting gains in the vicinity of measured gains. Given that so much analysis of human data has been conducted using classical control methods, it is critical to employ these methods to compare findings with the main body of research. In this way, trends in fitted neural data can be compared with trends in fitted human data. If similar sources of input dependence are present in the human nervous system, and the neural network is modeling the actual underlying mechanisms of human balance, fitted trends will align. In this case, direct measurement of synthetic neural behavior may uncover similar trends in biological nervous systems.

Using classical methods to compare with human data, our network clearly reflects the trend in which fitted gains increase with increasing stimulus amplitude. However, our network shows that fitted gains increase with decreasing inertia, which is opposite to the trend observed in human data. These comparisons suggest both that we are on the right track with our use of neural theory, and that our controller is missing an adaptive mechanism present in the human control system.

2. Methods

2.1 Neural Theory and Subnetworks

A leaky integrator mean population neuron model is used in this work [3] with the membrane potential of each neuron governed by the differential equation

$$C_m \frac{dV}{dt} = I_{leak} + I_{syn} + I_{app} \quad (1)$$

where C_m is membrane capacitance, analogous to a time constant, V is membrane potential, and I_x are current sources and sinks.

In this model, each neuron can transmit information in the form of current to a downstream neuron only if its membrane potential is within the operating range of the connecting synapse. In this work, the operating range of each synapse is $R = 20\ mV$, where the lower threshold coincides with the resting potential of each neuron E_r. A membrane potential of 1 mV above resting potential translates to a current of 1 nA transmitted to the downstream neuron. Signals are transmitted continuously within the operating range, and signal strength is measured directly as signal amplitude above resting potential.

In the absence of input signals, current leaks from each neuron proportional to its membrane conductance G_m, such that membrane potential is driven back to resting potential, given by

$$I_{leak} = G_m(E_r - V) \quad (2)$$

Currents are transmitted via synapses within the operating range proportional to synaptic conductance G_s given by

$$I_{syn} = \sum_{i=1}^{n} G_{s,i}\,(E_{s,i} - V) \quad (3)$$

where E_s is the synaptic equilibrium potential. Signal transmission within the operating range is governed by the piecewise function

$$G_{s,i} = \begin{cases} 0 \ \ if\ V_{pre} < E_{lo} \\ g_{s,i} \cdot \frac{V_{pre} - E_{lo}}{E_{hi} - E_{lo}} \ \ if\ E_{lo} < V_{pre} < E_{hi} \\ g_{s,i} \ \ if\ \ V_{pre} > E_{hi} \end{cases} \quad (4)$$

where E_{hi} and E_{lo} are the limits of the operating range.

I_{app} is a signal representing external information such as the input of a measured angle to the controller. Equations 1 through 4 are analyzed in steady state to form Equation 5 which describes a single synaptic transmission between two neurons

$$U_{post} = \frac{\sum_{i=1}^{n} \frac{g_{s,i}}{R} \cdot U_{pre,i} \cdot \Delta E_{s,i} + I_{app}}{1 + \sum_{i=1}^{n} \frac{g_{s,i}}{R} \cdot U_{pre,i}} \quad (5)$$

$$\Delta E_{s,i} = E_{eq,synapse} - E_{r,neuron} \quad (6)$$

where U_{pre} and U_{post} refer to the membrane potential of the pre and post-synaptic neurons. Assuming there is no applied current and noting that $\Delta E_{s,i}$ is constant for a given neuron-synapse pair, U_{post}

depends on the synaptic conductance and the input voltage U_{pre}. The gain, A, occurring between two neurons is given by

$$A = \frac{U_{post}}{U_{pre}} \tag{7}$$

Gains may be applied between two neurons by increasing the synaptic conductance between them, according to the theory shown in Figure 1, developed from Equations 5, 6 and 7.

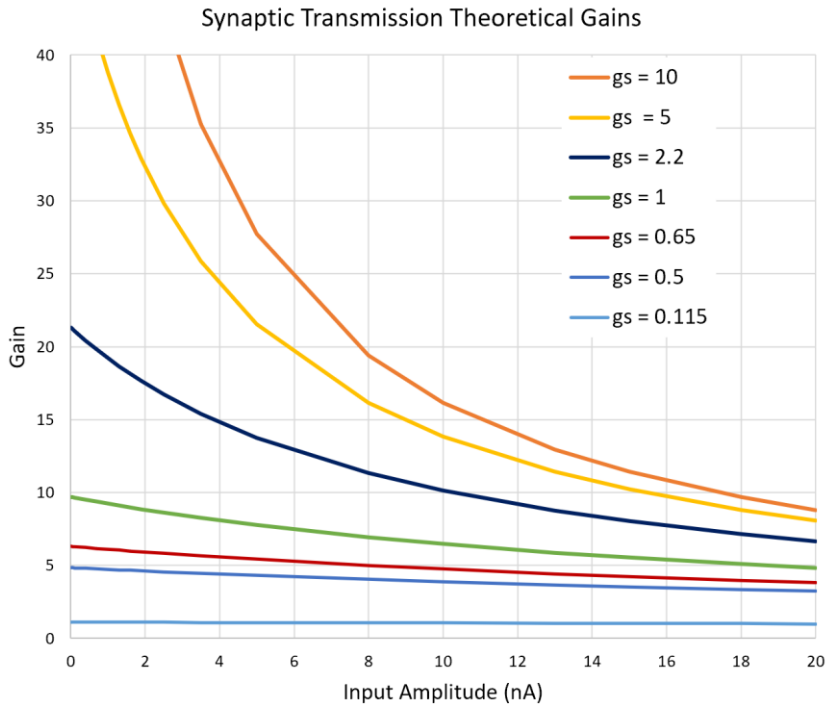

Figure 1: Gain is increased by increasing synaptic conductance which results in non-linear input dependence as input amplitudes approach zero.

It is clear from Equation 5 and Figure 1 that synaptic transmission gains result in non-linear input dependence as the amplitude of the input signal approaches zero. This is noteworthy because in feedback control systems, the error term is being driven toward zero where non-linear input dependence is most severe. The most neutral setting where gains are near 1:1 occurs at synaptic conductance 0.115 in Figure 1, although the true gain near zero input amplitude is 1.12.

Equation 5 is used in [2] as the basis of neural subnetworks whose biological properties are tuned to approximate basic mathematical operations such as addition, subtraction, multiplication and differentiation, shown in Figure 2.

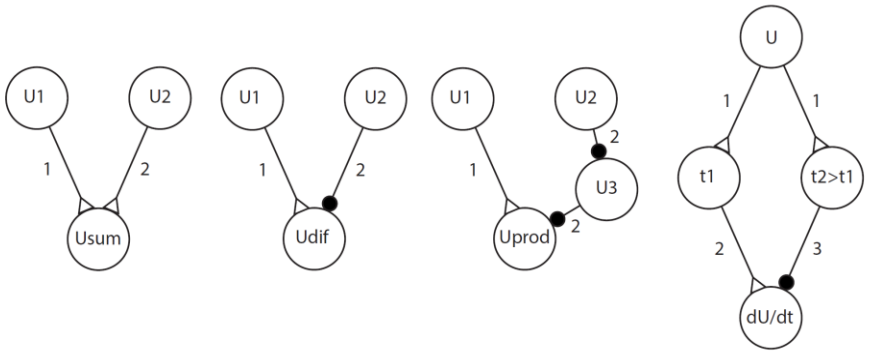

Figure 2: Subnetworks assembled using Equation 5 to approximate basic mathematical operations such as (left to right) addition, subtraction, multiplication and differentiation. Triangle connections indicate excitatory synapses and filled circles indicate inhibitory synapses.

Given that subnetworks are built from basic synaptic transmissions, each subnetwork introduces non-linear input dependence.

The biologically constrained neural network PD controller developed in [4] and shown in Figure 3 is assembled using these subnetworks and is used in this work as the basis for investigating input dependence and the future incorporation of adaptive networks.

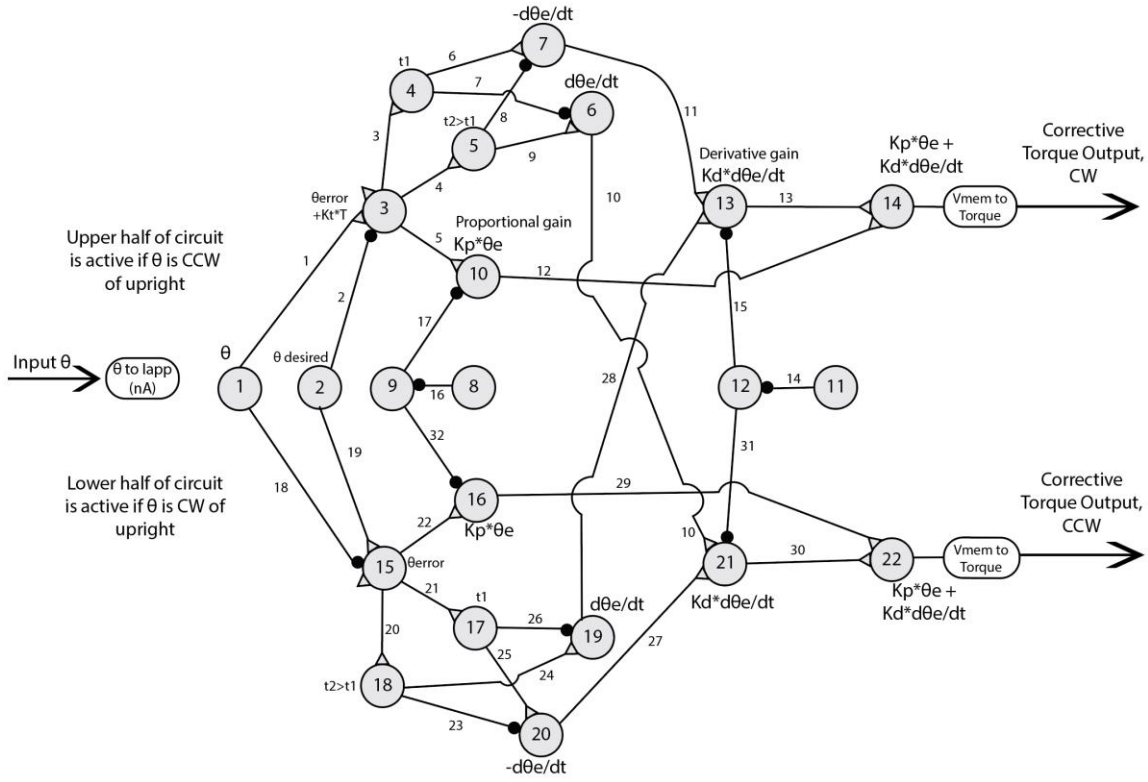

Figure 3: The neural network PD controller is developed from subnetworks tuned to perform mathematical operations using biological constraints [2, 4]. The controller is top-bottom symmetric to handle both positive and negative input angles, and the proportional gain K_p occurs at Neurons 10 and 16 and derivative gain K_d occurs at Neurons 13 and 21. Triangle connections indicate excitatory synapses and filled circles indicate inhibitory synapses.

2.2 Model and Setup

Our model for human balance is a single joint inverted pendulum which rotates about the ankle joint with corrective torque opposing the direction of displacement [4], [5]. This model is described by the differential equation

$$J\frac{d^2\theta}{dt^2} = mgh \cdot sin(\theta) + T_c \quad (8)$$

where *J* is the moment of inertia of the body, *m* is mass, *g* is acceleration due to gravity, *h* is the height of the center of mass, T_c is corrective torque and θ is the angle of the ankle joint.

A PD controller is the simplest controller fitted to human balance data in [1]. The block diagram for the closed loop feedback control system is shown in Figure 4 where the time delay represents the time between a nervous system control signal and muscle activation and the external gain is factored out of the PD control equation so that internal control gains are within the operating range of the neural controller. The input to the closed loop system is the desired angle, measured from zero in upright stance, and the output is the measured angle. This model assumes that the measured angle is provided by sensory systems.

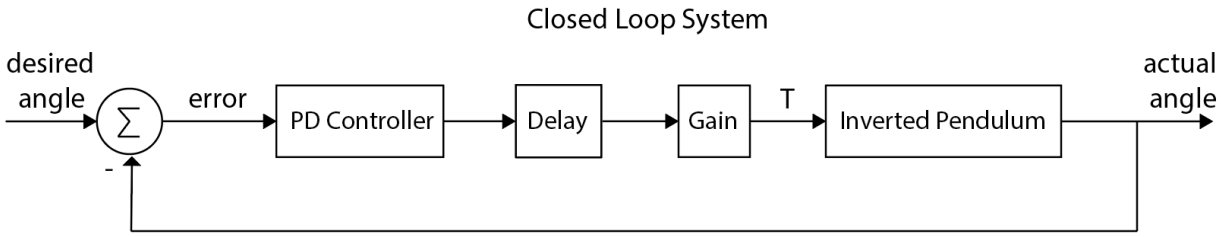

Figure 4: Closed loop feedback control system to emulate human balance is comprised of a neural network PD controller, time delay for muscle activation, external gain to convert control signal to torque, and inverted pendulum balance model.

In this work, a state space model of the inverted pendulum is simulated in Matlab and the neural PD controller is simulated in Animatlab with a virtual serial connection between them. Matlab calculates the position and velocity of the inverted pendulum and sends the desired and actual position to Animatlab. Animatlab sends the control signal to Matlab, where the external gain is applied, and the torque signal is inputted to the inverted pendulum simulation. Appendix 6.2 lists all codes and programs used for simulation which are provided in the supplementary materials.

2.3 Characterization of Open Loop Controller

The classical PD control equation is

$$\frac{PD_{out}}{PD_{in}} = K_p + K_d s \quad (9)$$

where K_p and K_d are the proportional and derivative gains respectively.

The neural PD controller has a very different frequency response from that of a classical PD controller. Each neuron acts as a low pass filter with a cutoff frequency f_c determined by its membrane capacitance, labeled “time constant” in Animatlab, according to

$$f_c = \frac{1}{(2\pi C_m)}\ Hz \tag{10}$$

The typical membrane capacitance was chosen to be 2 nF, analogous to a 2 ms time constant, so the typical cutoff frequency is 80 Hz. Since any path through the circuit passes through at least four typical neurons, the whole transfer function is multiplied by a fourth order 80 Hz filter.

In the derivative circuit, the error signal is passed to two different neurons, each with a different time constant, then the resulting signals are subtracted to approximate the derivative. The higher time constant is associated with the cutoff frequency of the circuit's low pass filter. Since the filter attenuates the signal, the cutoff frequency is inversely related to the circuit gain as shown in Figure 5, where the design cutoff frequency and attenuation is shown in red.

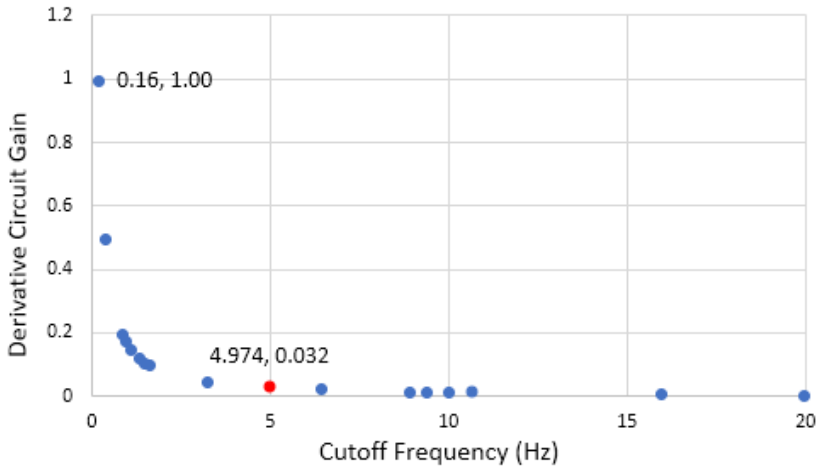

Figure 5: As the choice of membrane capacitance (time constant) increases, the cutoff frequency of a neuron's low pass filter decreases. To achieve a gain of 1:1, the derivative circuit's cutoff frequency would be 0.16 Hz, far too low for this system. A minimum cutoff frequency of 5 Hz was chosen for the derivative network, resulting in a theoretical base gain of 0.032 for the derivative network, shown in red.

To minimize attenuation, the smallest possible cutoff frequency was chosen to be 5 Hz for this system. As such, the derivative circuit alone is affected by a 5 Hz low pass filter, and the complete open loop PD transfer function is

$$\frac{PD_{out}}{PD_{in}} = \tau_d \left(\frac{1}{\frac{s}{2\pi f_{80}}+1}\right)^4 \left[K_p + K_d s \left(\frac{1}{\frac{s}{2\pi f_5}+1}\right)\right] \tag{11}$$

where f_{80} and f_5 are the 80 and 5 Hz low pass filters respectively. This transfer function was used to fit the frequency response for all open loop runs.

2.4 Controller Design and Implementation

Since the closed loop human balance system completes its entire frequency response by 2.5 Hz, as shown in Figure 6, neither the 5 nor 80 Hz filters significantly affect the transfer function. The closed loop transfer function is

$$\frac{\theta_{actual}}{\theta_{desired}} = \tau_d \cdot Gain \cdot \frac{K_p + K_d s}{Js^2 - mgh} \tag{12}$$

Using Equation 12, a PD controller was fitted to human balance data provided by Dr. Peterka for blindfolded subjects with all but their ankle joints constrained and profound vestibular loss due to illness or injury. The best fit parameters were $K_p = 1050$, $K_d = 250$ and $\tau_d = 0.15\ s$, shown in Figure 6.

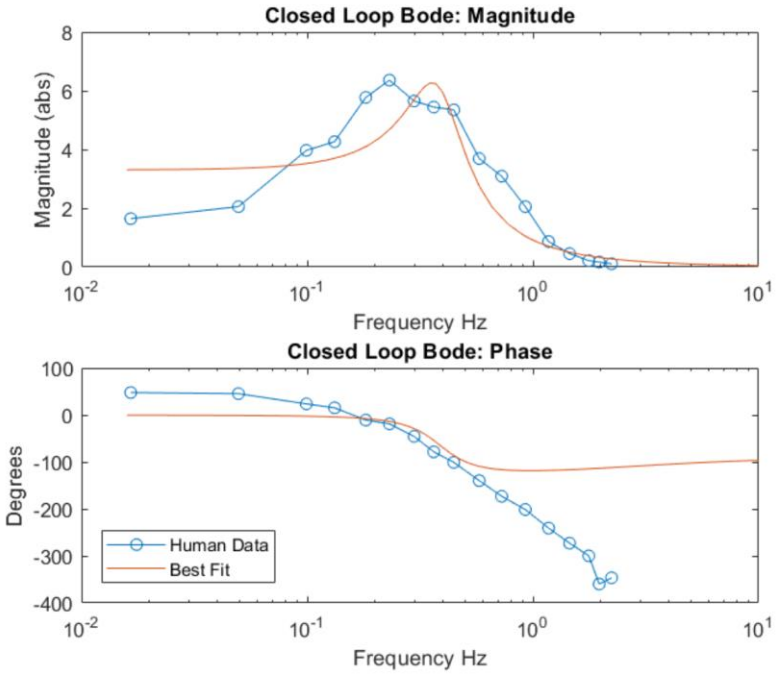

Figure 6: Closed loop PD controller and inverted pendulum system fitted to human balance data provided by Dr. Peterka. The best fit parameters are $K_p = 1050$, $K_d = 250$ and $\tau_d = 0.15\ s$.

In neutral settings, defined as the settings which cause nearly 1:1 synaptic gains with the least input dependence, the measured neural controller gains are $K_p = 1.26$ and $K_d = 0.046$. Due to the limited synaptic operating range, it is necessary to use an external gain to obtain the design gains shown in Figure 6. In order to maintain the K_p gain at neutral settings, the K_d term must be increased by a factor of 6 using a synaptic transmission gain, and an external gain must be applied to both terms. The resulting neural controller design gains are $K_p = 1.26$, $K_d = 0.28$ and $Gain = 833$.

In order to achieve the desired K_d gain, the synaptic conductance between Neurons 7 and 13 and 19 and 13 in Figure 3 and their symmetrical counterparts is increased to 0.65, the red curve shown in Figure 1. It is noted that for small input amplitudes between 0 and 5nA, the K_d gain is expected to vary between 5.5 and 6.3, while the K_p gain given by the light blue curve in Figure 1 should remain approximately constant.

2.5 Incorporation of Multiplication Circuit

The purpose of the multiplication circuit, the third subnetwork shown in Figure 2, is to enable a signal, applied at neurons 8 or 11 (U_2), to turn the gains up or down at neurons 10 and 13 (U_{prod}) and their symmetrical counterparts. In future adaptive applications, this is the circuit where weighting networks would tie in, to increase or decrease gains in order to find a stable equilibrium under changing conditions. At the maximum setting of $U_2 = 20\ mV$, U_3 is fully inhibited and the circuit has no impact

on downstream neurons, so U_2 must be set to a mid-range voltage so that it can be moved either up or down to have an amplifying or attenuating impact on K_p and K_d.

The multiplication circuit approximates multiplication of the form

$$\mathrm{U_{prod}} \approx \mathrm{U_1}\frac{\mathrm{U_2}}{\mathrm{R}} \tag{13}$$

where U_1 is the input signal to be modified, U_2 is the applied signal performing the modification, a value between 0 and 20 mV, and R is the typical operating range of 0 to 20 mV. Equation 13 shows that this circuit can only attenuate the incoming signal by applying a gain between 0 and 1 depending on the value of U_2. The actual behavior of the multiplication circuit is given by Equations 14 and 15 and plotted in Figure 7

$$\mathrm{U_3} = \frac{\mathrm{R}-\mathrm{U_2}}{1-\frac{\mathrm{U_2}}{\Delta \mathrm{E_{s2}}}} \tag{14}$$

$$\mathrm{U_{prod}} = \frac{\frac{\mathrm{g_{s3}}}{\mathrm{R}}\cdot \mathrm{U_{inter}}\cdot \Delta \mathrm{E_{s3}} + \frac{\mathrm{g_{s1}}}{\mathrm{R}}\cdot \mathrm{U_1}\cdot \Delta \mathrm{E_{s1}}}{1+\frac{\mathrm{g_{s3}}}{\mathrm{R}}\cdot \mathrm{U_3}+\frac{\mathrm{g_{s1}}}{\mathrm{R}}\cdot \mathrm{U_1}} \tag{15}$$

where $g_{s1} = 0.115\ \mu S$, $g_{s2} = g_{s3} = 20\ \mu S$, $\Delta E_{s1} = 194$, $\Delta E_{s2} = \Delta E_{s3} = -1\ mV$ and a constant current of 20 nA is applied at U_3, parameter settings outlined in [2].

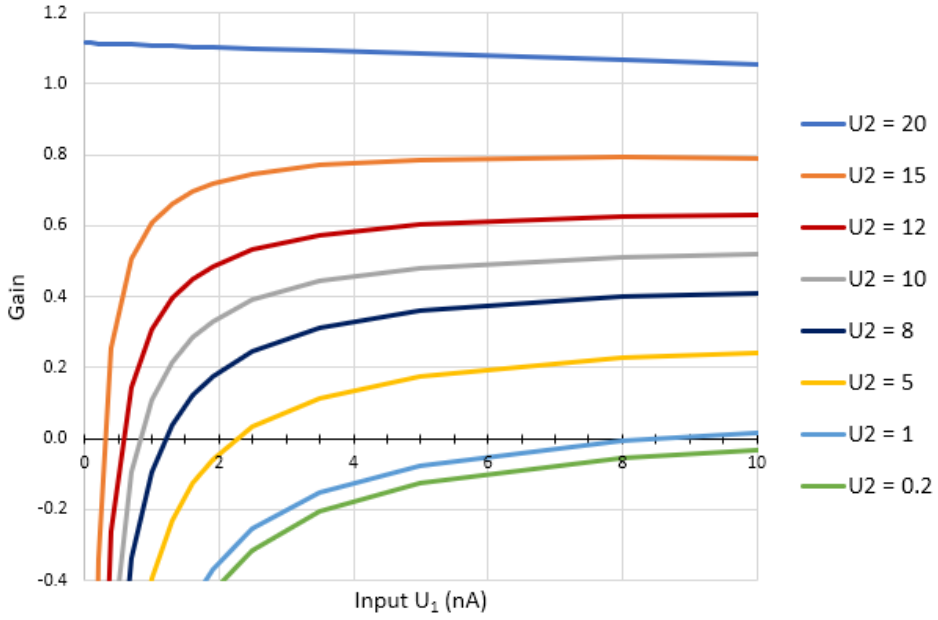

Figure 7: Actual gain of the multiplication circuit determined by selections of the U_2 input from Equations 14 and 15, meant to approximate multiplication of the form in Equation 13. It is clear that small input amplitude signals are severely, non-linearly attenuated and truncated at a gain of zero, below which the signal is outside the synaptic operating range and cannot be transmitted to a downstream neuron.

It is clear from Figure 7 that low input amplitudes at U_1 are severely attenuated and truncated at gains below zero, where the signal cannot be transmitted as it is outside the synaptic operating range. In this work, the mid-range voltage selected for U_2 is 12 nA, since distortion and data loss become more severe

as U_2 decreases. The effect of data loss at this setting is illustrated in Figure 8 where an input signal of 0.5 nA is inhibited to below the resting potential of the neuron at -60 mV, so the signal transmitted to a downstream neuron is zero.

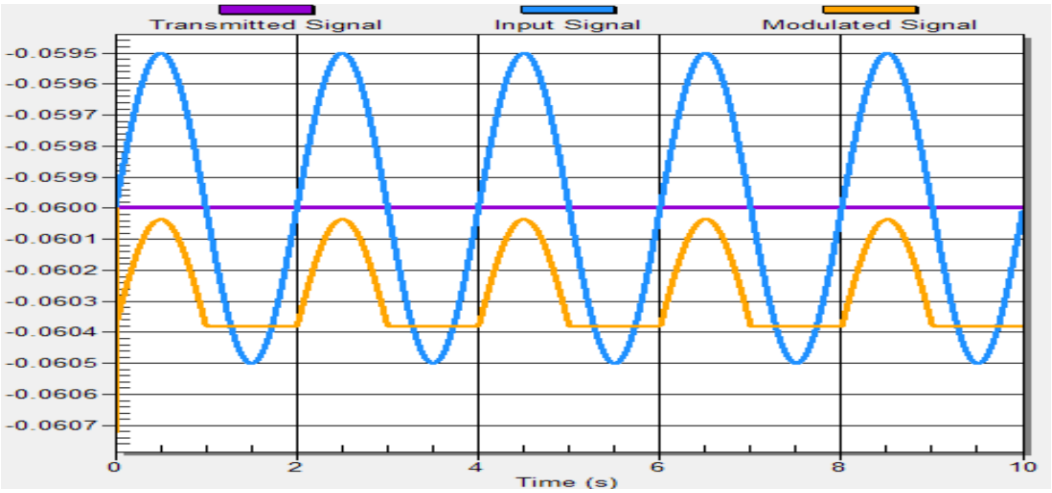

Figure 8: The multiplication approximates multiplication of the form $\mathrm{U_{prod}} \approx \mathrm{U_1}\frac{\mathrm{U_2}}{\mathrm{R}}$. *but inhibits low input signals to below the resting potential of the neuron at -60 mV. In this example, an input signal of 0.5 nA is completely lost when* $\mathrm{U_2} = 12\ nA$. *The Transmitted Signal shown in purple represents the signal seen by a downstream neuron, zero in this case.*

In this work, since U_2 is set to be a constant 12 nA, the impact of data loss is counteracted by a constant 0.6 nA offset applied at U_{prod}, derived from Figure 7. It is acknowledged, however, that a constant offset would not be appropriate for an adaptive circuit, since the variation of U_2 above or below 12 nA would still cause asymmetrical gain or data loss respectively.

2.6 Data Collection and Frequency Response Construction

Frequency responses are constructed for the open and closed loop systems using two methods: Auto Bode and White Noise/DFT. These two methods are also used to directly measure K_p and K_d gains by monitoring the neurons directly neighboring synapses where gains are applied.

2.6.1 Auto Bode

The Auto Bode method consists of applying a sine wave of a particular frequency and amplitude to the input of a system, measuring the amplitude and phase of the signal at the output, and calculating the magnitude ratio and phase between them. The frequency response is constructed one frequency at a time. The Auto Bode Matlab function is adapted from one developed by Dr. Turcic to include the calculation of standard deviation and output of mean input and output amplitudes. All Matlab scripts and data used in this work are listed in Appendix 6.1 and provided in the supplementary materials.

The Auto Bode time data is very easy to interpret, making it ideal for troubleshooting, and the low standard deviation makes it ideal for detecting differences between datasets. For the very low frequencies present in the human data, however, trial times are very long. Furthermore, serial communication failed for frequencies lower than 0.16 Hz. This method was used primarily to verify differences between datasets. Examples of the time data and frequency response obtained using the Auto Bode method are shown in Figure 9

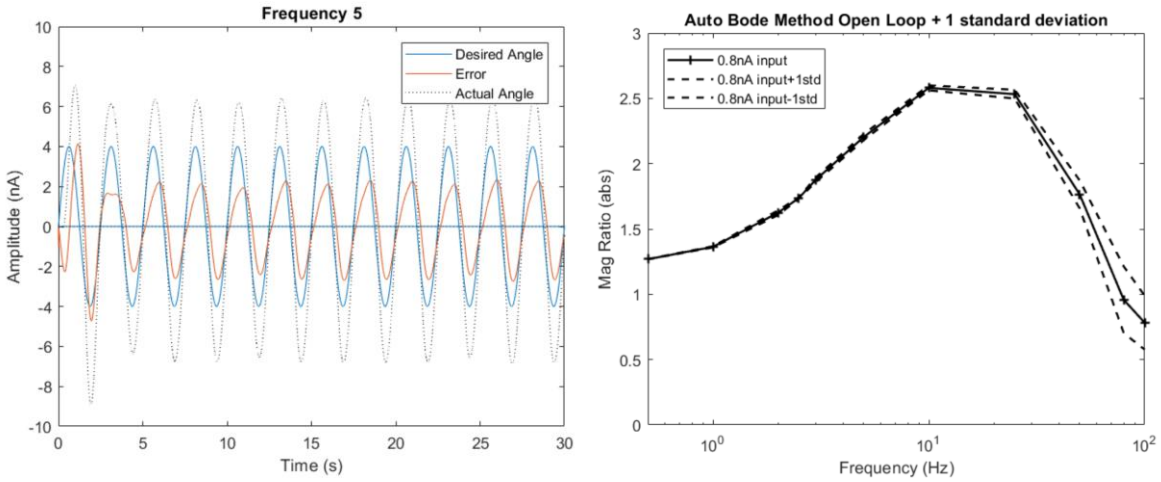

Figure 9: Time data (left) for one frequency step and a complete frequency response (right) constructed using the Auto Bode method. Time data is easy to interpret for troubleshooting data and the standard deviation is very low, making this method ideal for detecting differences between datasets.

2.6.2 White Noise and Discrete Fourier Transform

The White Noise/DFT method inputs a gaussian white noise signal to the system, comprised of all frequencies and various amplitudes and outputs another white noise signal. The discrete Fourier transform is applied to the input and output signals respectively, which supplies the amplitudes and phase corresponding to each frequency. The output amplitudes are divided by the input amplitudes to provide the magnitude ratio for the frequency response.

The White Noise/DFT method results are noisy, as shown in Figure 10, likely due to leakage and noise in serial communication timing.

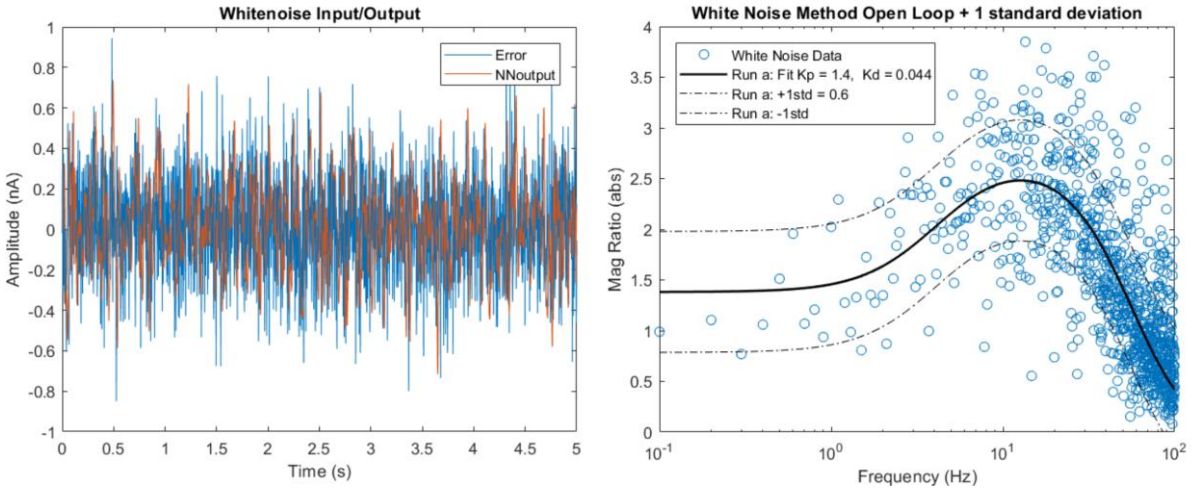

Figure 10: Example of system input and output (left) and frequency response (right) using the White Noise/DFT method.

In this work, since the amplitudes must be as accurate as possible, a rectangular window was used rather than another window type which would require a correction factor to calculate amplitude. To improve accuracy and maintain the required frequency resolution, a binning and averaging strategy was used with the characteristics shown in Table 1.

Table 1: Binning and averaging strategy characteristics for open and closed loop White Noise/DFT method to improve accuracy and maintain desired frequency resolution

	Frequencies of Interest (Hz)	Sample Frequency (Hz)	Signal Length (s)	Bin Length (s)	Frequency Resolution (Hz)
Open Loop	0.5 - 100	400	100	10	0.1
Closed Loop	0.01 - 5	200	400	100	0.01

The White Noise/DFT method had no experimental lower limit to frequency resolution, making it the preferred choice for closed loop analysis. Bin length must only be long enough to account for the lowest desired frequency, so trial lengths are reasonable. However, it has a high standard deviation, making it difficult to compare datasets. A typical strategy used in this work was to run White Noise/DFT tests, and if differences were detected between datasets, to follow up with the Auto Bode method to verify the difference.

3. Results

3.1 Establishing a Baseline for Comparison

The open loop PD controller with neutral settings shows no significant signs of input dependence. Figure 11 shows that gains fitted to white noise data align with measured baseline gains of $K_p = 1.26$ and $K_d = 0.046$. No significant difference is detected between the frequency responses of two extreme input amplitudes found using the Auto Bode method. A baseline standard deviation of the White Noise/DFT method in the absence of input dependence is established as 0.55.

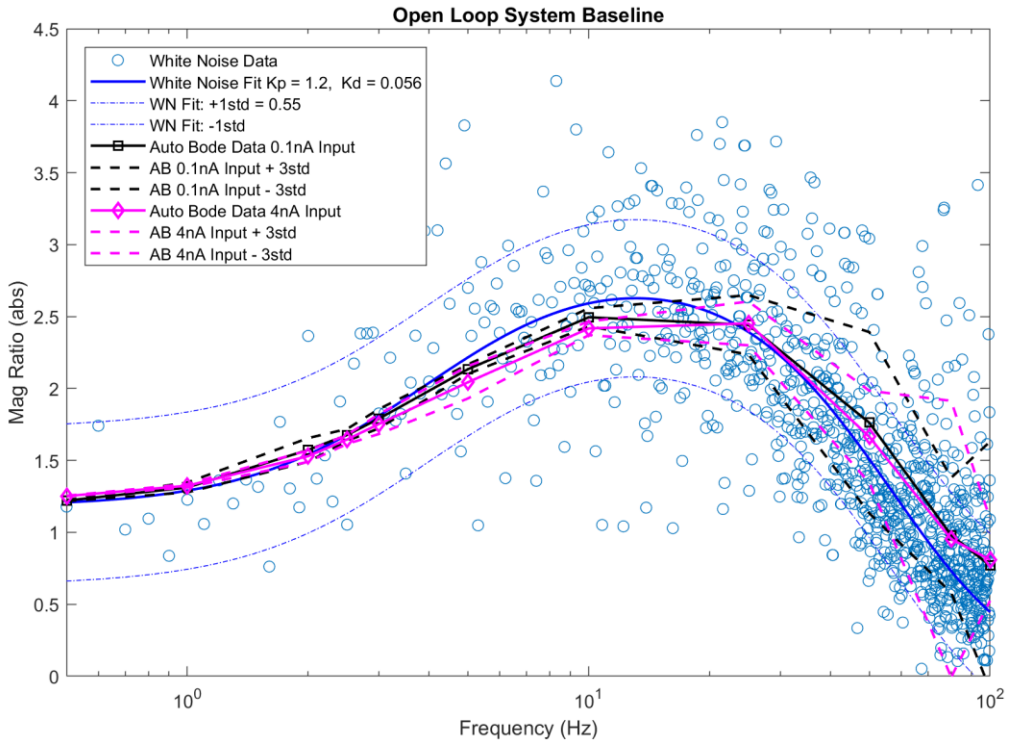

Figure 11: The open loop PD controller in neutral settings does not exhibit significant input dependence and provides a baseline standard deviation for the White Noise/DFT method of 0.55. No significant difference is detected between two Auto Bode runs at extreme input amplitudes. The fit to the White Noise/DFT data aligns with measured gains of $K_p = 1.26$ and $K_d = 0.046$.

3.2 The Source of Input Dependence in Open Loop Controller

When synaptic gain is added to achieve the design gain for K_d as discussed in Section 2.4, signs of input dependence are noted in the K_d circuit as well as the open loop system. Figure 12 shows a comparison of controller behavior with mean white noise signal input amplitudes 0.2 and 1.5 nA.

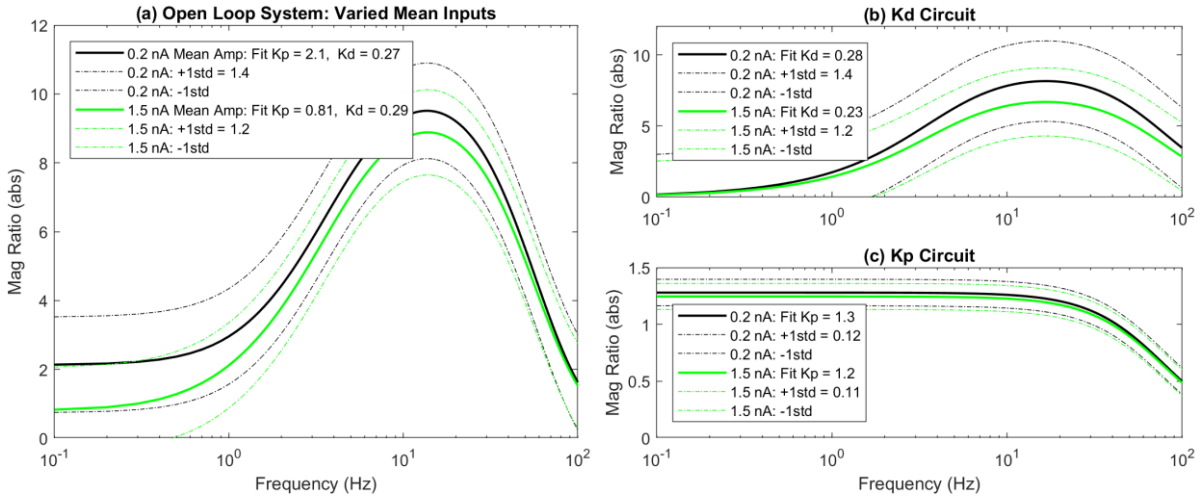

Figure 12: Input dependence is identified in the open loop PD controller K_p set to neutral and K_d amplified with a synaptic gain. The K_d circuit (b) and open loop system (a) fits are not the same shape and fitted gains vary significantly given two different mean input amplitudes, whereas K_p fitted gains (c) are consistent. Standard deviations in (a) and (b) are much higher than the baseline of 0.55.

Fits to the K_d circuit and open loop system show signs of input dependence, the symptoms of which are highly variable fitted gain values and the standard deviations well above the baseline White Noise/DFT standard deviation of 0.55. The Auto Bode method is used to verify this difference and measure gains directly as shown in Figure 13 given a small, mean and maximum amplitude from the white noise runs in Figure 12. The open loop system shows a significant different between the 4 nA input and two smaller inputs, and the K_d circuit shows a significant difference between all three input amplitudes.

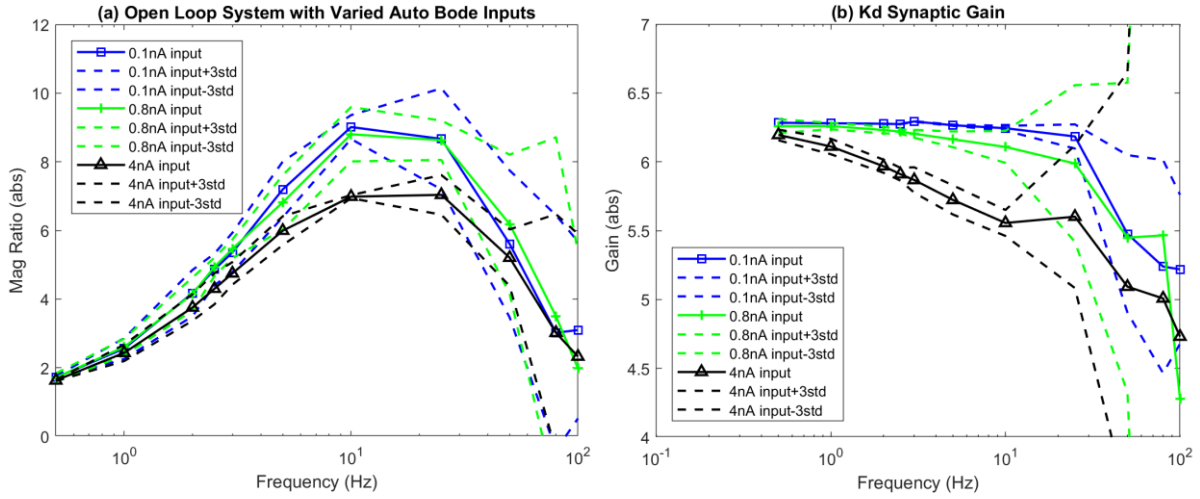

Figure 13: Auto Bode input amplitudes were selected from values present in the White Noise/DFT runs shown in Figure 12 to represent the minimum, mean, and maximum inputs. It is clear from the system frequency response (a) that the 4nA input is different from lower inputs. The directly measured K_d synaptic gain (b) verifies that K_d gains are significantly different for all

selected input amplitudes and variable over the frequency range. The 4 nA input shows K_d synaptic gain variation between 5.5 and 6.3, which covers the range of theoretical gains expected in this input range from the $g_s = 0.65$ curve in Figure 1.

The measured K_d synaptic gain varies depending on input amplitude within the 5.5 to 6.3 magnitude range predicted by theory for $g_s = 0.65$ shown in Figure 1. Since K_p shows no significant input dependence at neutral settings, this confirms that the ratio between control gains varies depending on input and within a single frequency response. Since classical control fits assume gains are constant, this explains the inaccurate gain fits and high standard deviations when input dependence is present.

Figure 14 gives a summary of open loop runs at neutral gains and with added K_d synaptic gain. It is observed that fitted gains are consistent and standard deviations very close to baseline for neutral runs, whereas fitted gains for non-neutral runs, particularly for K_p which we know to be constant, vary considerably with standard deviations well above baseline.

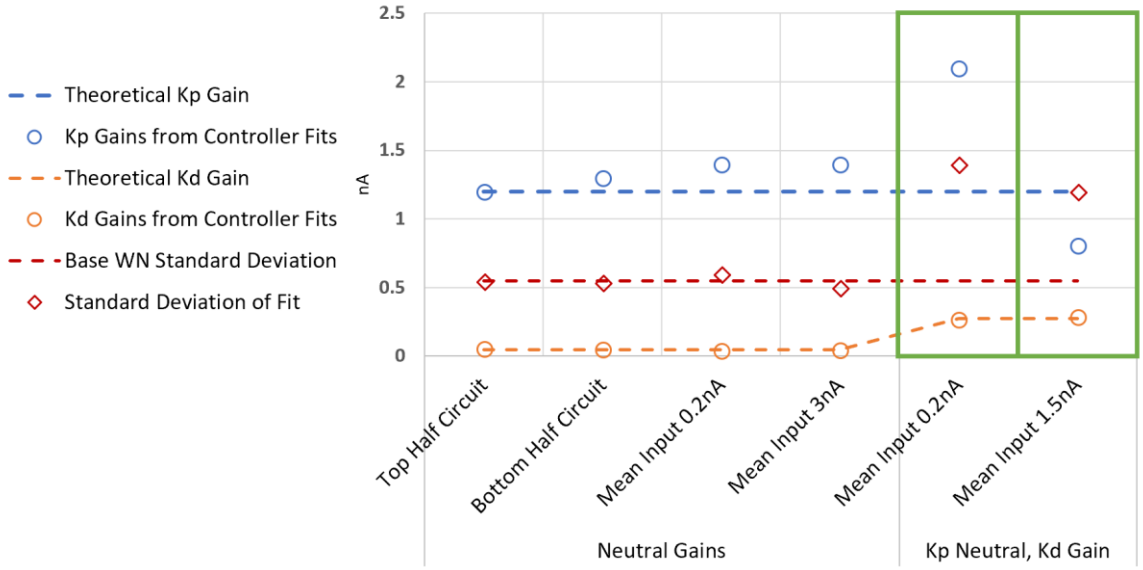

Figure 14: A summary of classically fitted gains to open loop runs using the White Noise/DFT method. At neutral settings, fitted gains are consistent and standard deviations near the baseline of 0.55. When K_d synaptic gain is added, fitted gains, particularly that of K_p which we know to be constant, are variable with standard deviations well above baseline. This is further evidence that unexpected gains and high standard deviations are indications of input dependence.

3.3 The Effects of Input Dependence in the Closed Loop System

The influence of input dependence in the closed loop system without the influence of the multiplication circuit is shown in Figure 15, where three different mean white noise input amplitudes are compared. The controller developed thus far is not stable with a 0.15 s time delay, so all further data is taken with a delay of 0.1 s. For this reason, both the original human fit with a 0.15 s time delay and an adapted fit, with the same parameters but a 0.1 s time delay are shown. Input amplitude is clearly a parameter of the system, moving the data around the human fit without altering the settings of the neural controller.

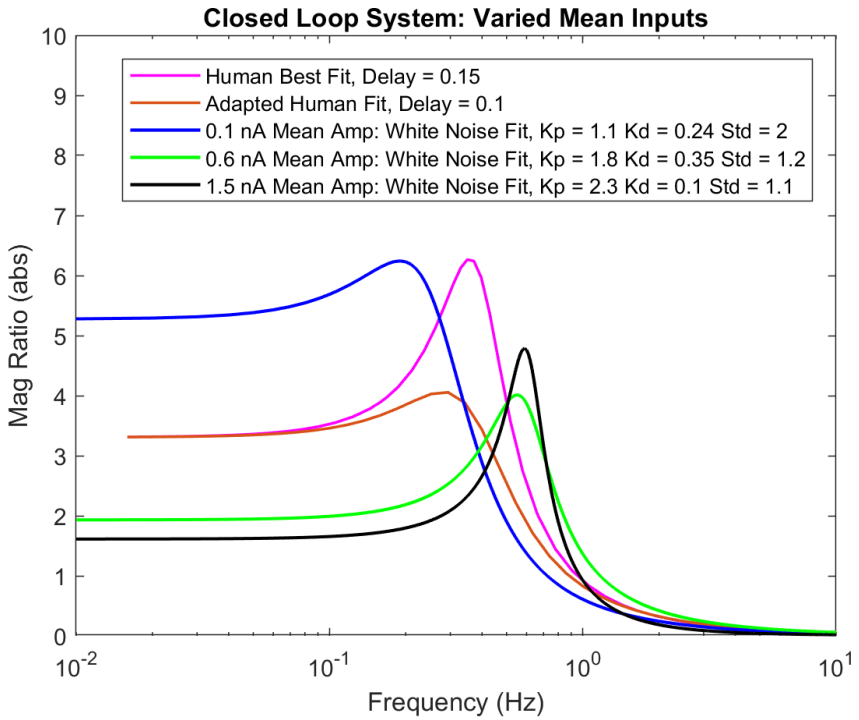

Figure 15: The effect of input dependence is investigated in the closed loop system with three different mean white noise input amplitudes. The smallest input amplitude results in the frequency response with the highest overall gain. The best fit to the human data from Figure 6 includes a 0.15 s time delay and a fit using the same gains and an alternative time delay of 0.1 s are shown. Changing the mean input alone moves the data around the best fit to the human data.

To further verify the impact of input dependence in the closed loop system, Auto Bode runs were conducted at a variety of input amplitudes as shown in Figure 16.

The error amplitude is the input to the PD controller, which is the source of input dependent controller behavior. Error amplitude increases as the input amplitude increases. A significant change in control behavior occurs where error amplitudes increase around 0.5 Hz, dividing the closed loop response into two regimes. For low error inputs, overall control gains are high, and for higher error inputs, a natural frequency peak forms around 0.65 Hz. The development of this peak points to a change in K_d gains. Direct measurement of K_d gains clearly illustrates that the synaptic transmission gain shown in Figure 1 causes gains to decrease with increasing input amplitudes.

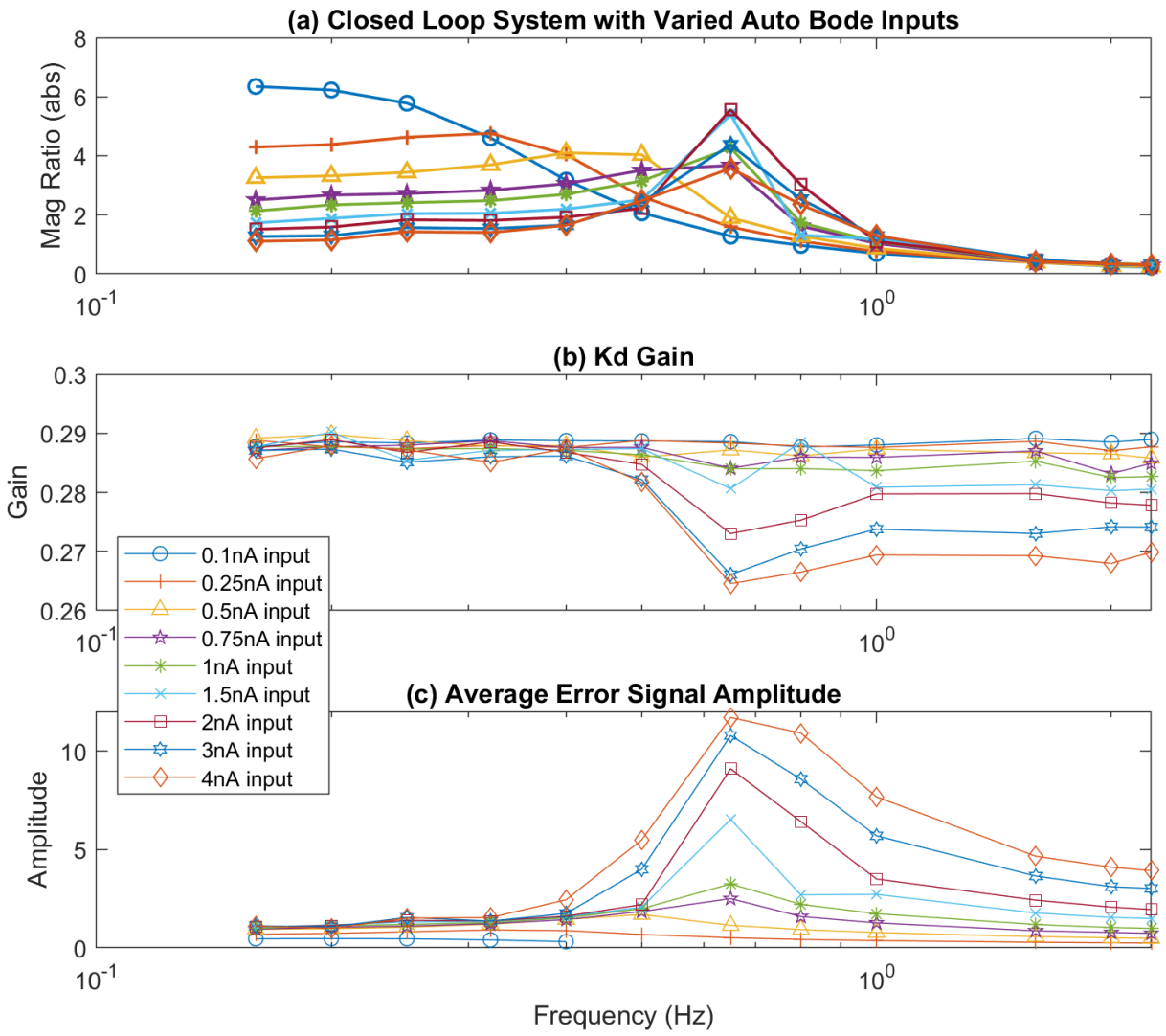

Figure 16: The input to the PD controller is the error term shown in (c). The amplitude of the error increases as the input to the closed loop system increases. The closed loop system (a) shows two different control regimes divided around 0.5 Hz. Before 0.5 Hz, overall system gain is high while error inputs are low, then where the error amplitude increases, a natural frequency peak occurs around 0.65 Hz. Directly measuring the K_d gain (b), we clearly see the trend that higher error amplitudes are associated with lower gains. This aligns with our expectation from theory shown in Figure 1.

3.4 Low Input Distortion from the Multiplication Circuit

To pave the way for the addition of adaptive circuitry as discussed in Section 2.5, the multiplication circuit is enabled with a constant offset to overcome low input data loss. New theoretical gains are $K_p = 0.76$ and $K_d = 0.17$ due multiplication of each gain by a theoretical 0.6 shown in Figure 7. To investigate the impact of the multiplication circuit, the same mean input amplitudes are applied to the "Post-mod" circuit as to the "Pre-mod" circuit shown in Figure 15 and the results are compared in Figure 17 and Figure 18. Measured mean gains for the Post-mod run at 0.6 nA mean input amplitude are shown in Figure 17 (b), where K_p matches theoretical values almost exactly and K_d is approximately 0.02 higher than theoretical. A higher mean K_d is expected for low input amplitudes.

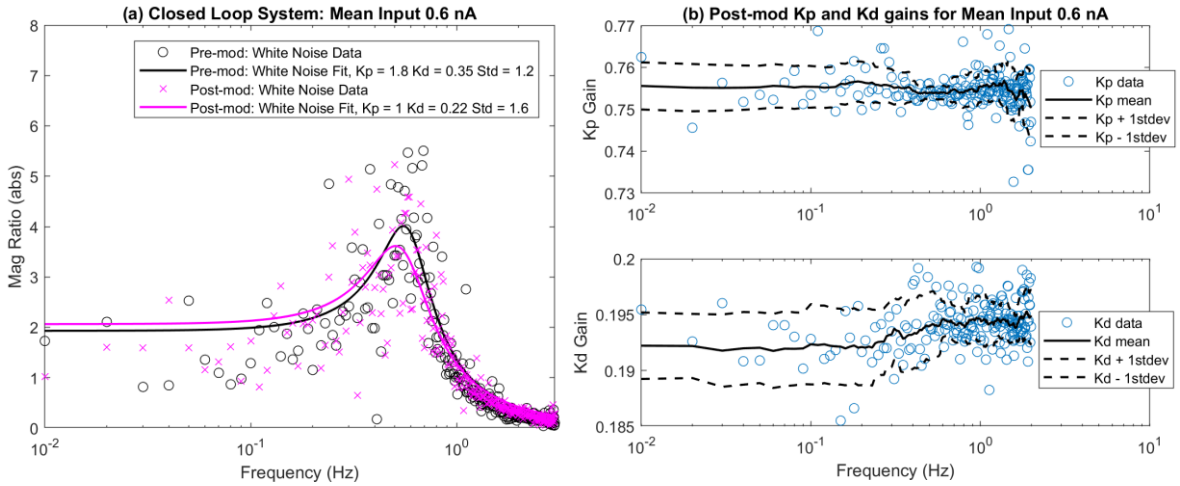

Figure 17: Pre-mod and Post-mod runs show similar fits for a mean white noise input amplitude of 0.6 nA (a). Measured mean K_p gain (b) for the Post-mod run matches theory almost exactly, whereas the mean K_d gain (c) is about 0.02 higher than theoretical, which is to be expected for a relatively low input amplitude.

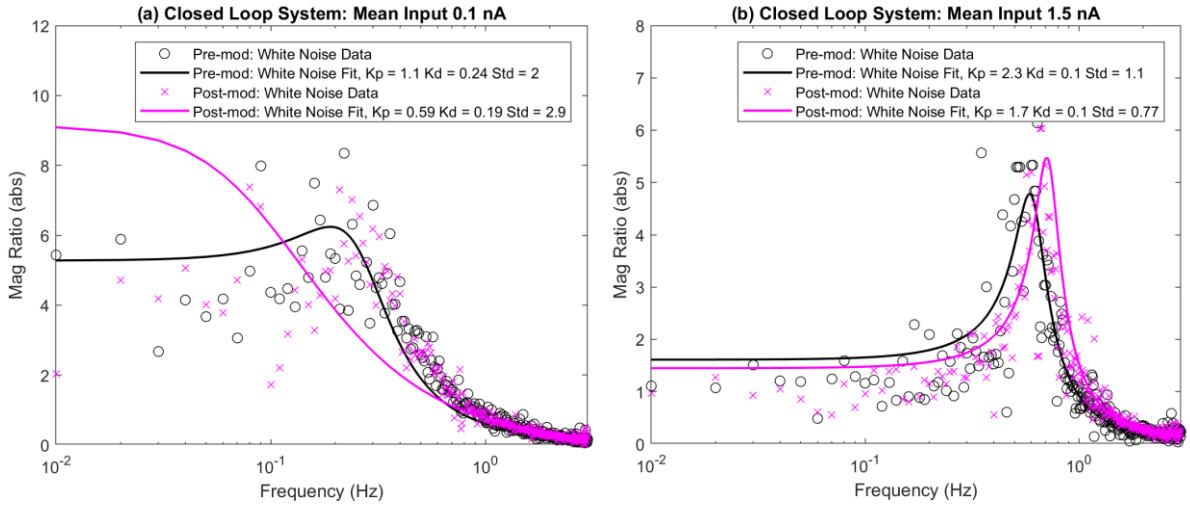

Figure 18: For very small mean input amplitudes (a), the Post-mod circuit cannot be accurately fitted with the classical control equation due to truncated and distorted data as shown in Figure 7. Pre and Post-mod behavior is similar for higher input amplitudes (b), while all fitted gains are inaccurate and standard deviations are high.

For the lowest mean input amplitude of 0.1 nA, the Post-mod fit no longer represents the data. This is to be expected due to severe distortion and truncation of data at very low input amplitudes as shown in Figure 7. This clearly shows that the constant offset applied at U_3 is not sufficient to fully reverse the impact of data loss and distortion. Higher mean input amplitudes show little difference between Pre and Post-mod runs.

Figure 19 gives a summary of *measured* gains with three standard deviations and Figure 20 gives a summary of *fitted* gains and standard deviation values for closed loop runs. Measured mean K_p gains are constant regardless of input, while mean K_d gains vary by input. Fitted gains do not match measured gains, and standard deviations vary wildly, indicating the influence of input dependence.

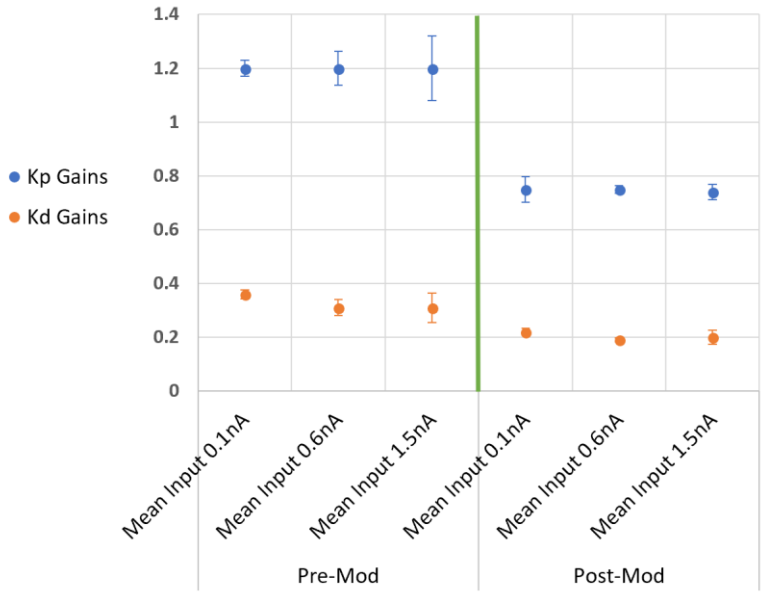

Figure 19: Mean measured gains are shown with three standard deviations. Mean K_p gains do not vary with input, while K_d gains do vary with input, most notably they are highest for the lowest input amplitude.

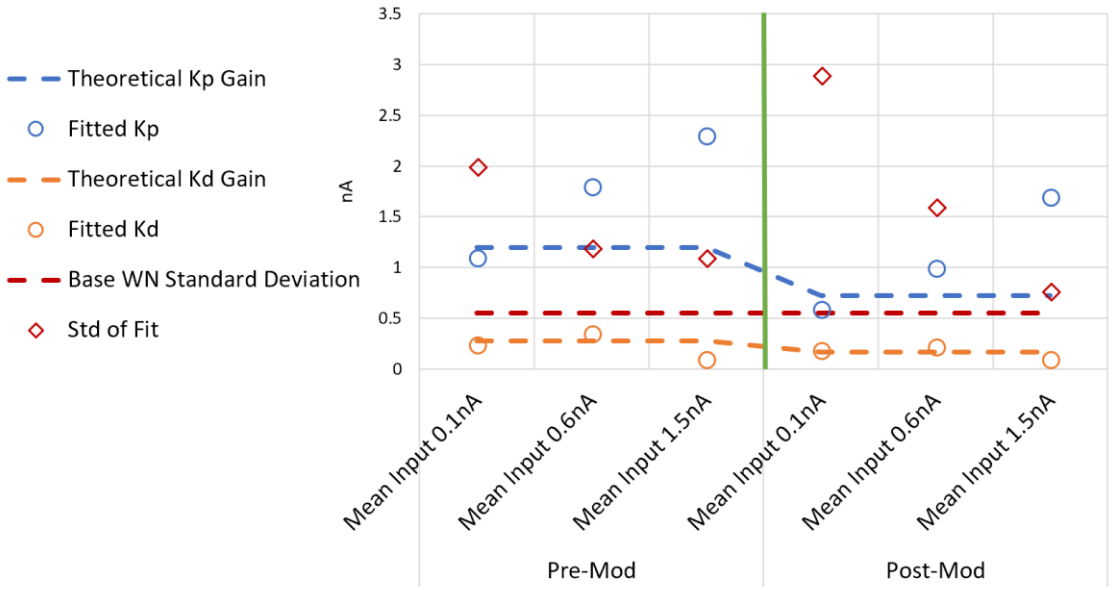

Figure 20: Classically fitted gains do not agree with measured gains shown in Figure 19, and standard deviations are well above baseline and vary wildly, indicating the influence of input dependence. Fitted K_p values show a clear trend of increasing with input amplitude, which matches a trend noted in [1] and discussed in Section 4.3.

3.5 Controller Response to Inertial Changes

Sensitivity tests are conducted to investigate the response of the neural controller when inertial properties are varied to establish a baseline for future work in adaptability. Figure 21 shows closed loop behavior for a significant decrease and increase to inertia by altering mass and height properties and compares it to the baseline response using the original inertia with which the controller was designed.

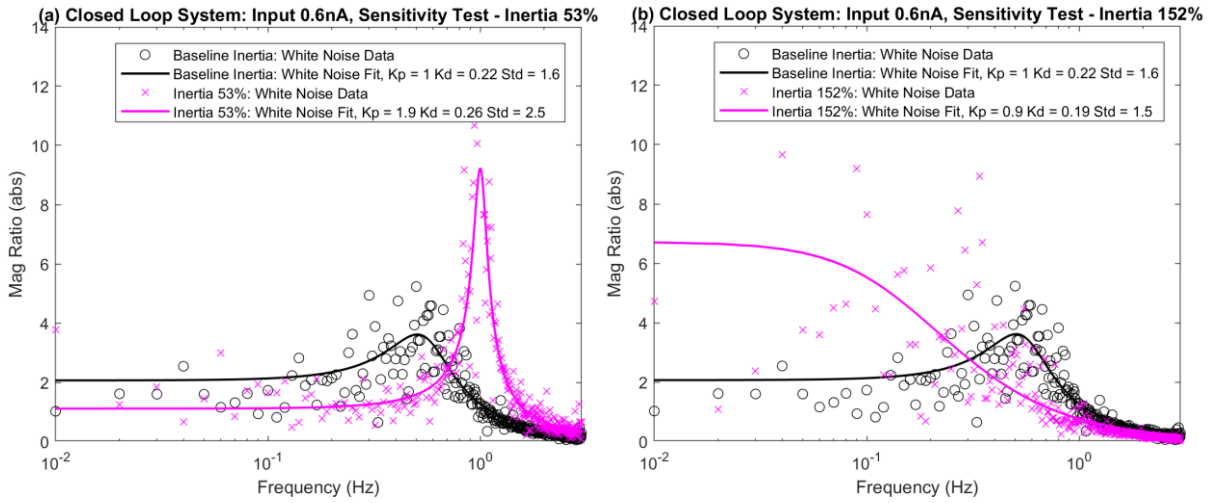

Figure 21: Given a 0.6 nA mean white noise input, the closed loop response given significant increases and decreases to inertia are compared to the baseline response with original inertia. While fitted gains and standard deviations still indicate the presence of input dependence, a trend is noted in which fitted gains increase with decreasing inertia.

Since the properties of the system being controlled have changed, the changes to the shape of the frequency response are to be expected. The fitted gains and standard deviations still show signs of input dependence. A trend is noted in which fitted gains increase for a decrease to inertia, and vice versa. This trend is opposite of the trend in fitted human balance data observed in [1] and discussed in Section 4.3.

4. Discussion

In the design of biologically constrained neural controllers, it is a useful practice to find a neutral baseline for comparison, in which the influence of input dependence is insignificant, complete and verify the design in open loop, then implement the controller in closed loop. In this way, one develops a means of quickly identifying the signs of input dependence or other sources of variation. This enables the use of well-known classical controls tools to aid in neural controller design and in comparison with other researchers' work, while remaining grounded in the realities of the neural controller, which may not agree with information gleaned from classical methods. For an engineer to successfully add adaptive circuitry to the neural controller, he or she must understand the underlying theory, its effect on the neural controller and its implications in the closed loop system.

4.1 Signs of Input Dependence in Neural Control

Using the baseline measured gains and measurement standard deviations in neutral settings, where input dependence is insignificant, input dependence is easily identified. In synthetic neural circuits, true gains are readily measured. Even without direct measurements, the hallmark signs of input dependence in the overall frequency response are significant variation of fitted gains which are known from baseline testing to be constant and significant increases in the standard deviation of the fit relative to the baseline. Maintaining at least one control gain in its neutral setting provides a very useful reference to quickly compare fitted gains as the controller is developed.

In the neural PD controller, input dependence is noted when synaptic transmission is used to apply control gains, consistent with subnetwork theory presented in Section 2.1. The multiplication circuit further distorts low input data even with a constant offset, as presented in Section 2.5.

4.2 Uses and Limitations of Classical Control Theory

If all control gains could be maintained in neutral settings, classical methods would reliably find accurate control gains. However, given that the ratio between control gains determines controller behavior, it is unlikely that all control gains can be maintained in their neutral setting to solve a given control problem. As soon as one gain is manipulated, input dependence is introduced.

Input dependence in the neural controller means that at least one of the control gains is continuously varying depending on input amplitude. Since classical methods are built around control gains being constant, they cannot possibly identify accurate control gains. In general, classical methods provide a useful means of finding a best fit line to the data but in doing so, identify inaccurate gains. In this work, K_p was held constant, but fitted values were rarely accurate due to the variation of K_d. In this work, while fitted gains were found to be inaccurate in the presence of input dependence, they remained in the vicinity of measured gains. If higher synaptic gains are applied, and/or they are applied to multiple gains in the same controller, this trend may no longer apply.

Since classical methods are a useful means of guiding controller design and finding a best fit line to the data with gains in the vicinity of measured gains, they are a useful means of comparing neural controllers to those developed by other researchers using the same methods of analysis.

4.3 Comparisons to Human Balance Data

In this work, classical fits to the closed loop system with a neural controller were compared with classical fits to human balance data provided by Dr. Peterka in [1]. Human balance data for blindfolded, normal subjects (with no vestibular loss) show a trend in which overall system gain decreases with increasing input amplitude. Figure 15 exhibits a similar trend with the neural balance controller. In Figure 16, this trend is seen for input amplitudes of 0.5 nA and less. For higher input amplitudes, this trend is present in frequencies below 0.5 Hz, but above this frequency, a natural frequency peak forms, which is not seen in human data. Since both the human balance control system and neural controller show input dependence, but the neural controller does so only over a certain range of inputs, it is reasonable to assume that the correct range of inputs must be found to align the neural and human responses.

Classical fits to the data collected from normal, blindfolded subjects also show a clear trend of increasing K_p gains with increasing input amplitude whereas K_d gains remain relatively constant. Figure 20 clearly shows this trend in classical fits to the neural control data. We know from our analysis of the neural controller that fitted gains do not reflect measured gains due to input dependence. Given that the human balance control system also exhibits input dependence, gains in the human system might be varying according to input amplitude in a similar manner. As such, fits to human data might be missing accurate nervous system gains in much the same way we have demonstrated with our neural controller.

Figure 21 shows a trend in the neural controller in which both fitted control gains decrease with increases to inertia. The trend in human data exhibits the opposite behavior, where increases to inertia result in increases to fitted control gains. This trend in human data is what we would expect from an adaptive circuit. In the inverted pendulum model, larger inertias require larger corrective torque to

return to a stable equilibrium, whereas smaller inertias require less. The fact that we do not see this trend in our neural controller indicates that we are missing the adaptive circuitry present in the human nervous system.

4.4 Looking Toward Incorporation of Adaptive Circuitry

Figure 16(b) shows that the default behavior of our input dependent neural controller is to provide the highest gains when the error is smallest and the smallest gains where error is largest. This is exactly opposite what we expect from an adaptive controller. In theory, if error is near zero, less gain is needed to avoid overshoot of upright equilibrium, but if error is large, larger gains are needed to return to stability. This suggests that the basic neural controller built from the subnetworks shown in Figure 2, although its gains are adaptive in the sense that they are input dependent, cannot natively provide adaptation in the direction we expect without further adaptive circuitry to overcome this default behavior.

In this work, the greatest barrier to the incorporation of adaptive circuitry is identified as the multiplication subnetwork, explored in Section 2.5. While we have overcome the most severe effects of low input data loss with a constant offset, this solution is not suitable to an adaptive network. If the value of the modulation input at U_2 is continuously varying between the curves shown in Figure 7, the offset required to overcome the data loss will also vary continuously. The variation of U_2 will have an asymmetrical impact on the resulting gain. For values of U_2 below the designed equilibrium, data loss will still occur because the constant offset will be insufficient to overcome it, but for values above equilibrium, the constant offset will cause higher gains than desired.

This multiplication network is critical to the adaptive neural control system in that it enables an external signal, or one generated from opposing weighted feedback signals, to influence control gains. In its current form, however, it seems to cause more problems than it solves. One possible solution is to return to the fundamental neural theory in Equation 5 and develop a new multiplication circuit which minimizes loss and distortion of low amplitude signals. Another is to develop an adaptive circuit whose sole purpose is to change the offset at U_{prod} in a manner complementary to the changing signal at U_2.

The use of mean neural population theory and mathematical subnetworks to construct biologically constrained neural controllers is a promising method of identifying how the human nervous system performs adaptive balance control. More work is needed to ensure that all trends in human data are matched by the neural controller while not losing track of its true measured behavior. While classical control fits are a useful means of comparing with the work of other researchers, the measured behavior of the neural control may provide further insights into the underlying behavior of the human nervous system.

5. Works Cited

[1] R. J. Peterka, "Sensorimotor Integration in Human Postural Control," *Journal of Neurophysiology,* pp. 1097-1118, 2002.

[2] N. S. Szczecinski, A. J. Hunt and R. D. Quinn, "A Functional Subnetwork Approach to Designing Synthetic Nervous Systems that Control Legged Robot Locomotion," 2017 Aug 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC552699/.

[3] T. P. Trappenberg, "Ch 3: Simplified Population and Neuron Models," in *Fundamentals of Computational Neuroscience*, New York, Oxford University Press, 2002, pp. 74-76.

[4] W. W. Hilts, N. S. Szczecinski, R. D. Quinn and A. J. Hunt, "Simulation of Human Balance Contnrol Using an Inverted Pendulum Model," in *Living Machines*, 2017.

[5] R. J. Peterka, "Simplifying the Complexities of Maintaining Balance," *IEEE Engineering in Medicine and Biology,* pp. 63-68, 2003.

6. Appendix

6.1 Datasets and Matlab Scripts used in Paper

All data and Matlab scripts are available in the Hunt Lab Shared Folder.

Figure	Dataset ID	External Gain	Data Description	Matlab Script
18(b)	WNCL_70(1.5)	833	Pre-mod, mean input 1.5 nA	Closed_Loop_FA_WN_Compare_TX.m
18(b)	WNCL_68(1.6)	1388	Post-mod, mean input 1.6 nA	Closed_Loop_FA_WN_Compare_TX.m
18(a)	WNCL_69(0.1)	833	Pre-mod, mean input 0.1 nA	Closed_Loop_FA_WN_Compare_TX.m
18(a)	WNCL_67(0.1)	1388	Post-mod, mean input 0.1 nA	Closed_Loop_FA_WN_Compare_TX.m
17(a & b)	WNCL_11a(0.6)	833	Pre-mod, mean input 0.6 nA	Closed_Loop_FA_WN_Compare_TX.m
17(a & b)	WNCL_32(0.6)	1388	Post-mod, mean input 0.6 nA	Closed_Loop_FA_WN_Compare_TX.m
21(a)	WNCL_32(0.6)	1388	Post-mod, mean input 0.6 nA	Closed_Loop_FA_WN_Compare_TX.m
21(a)	WNCL_36(0.6)	1388	Post-mod, mean input 0.6 nA, m_adjust = 0.4, h_adjust = 0.4	Closed_Loop_FA_WN_Compare_TX.m
21(b)	WNCL_32(0.6)	1388	Post-mod, mean input 0.6 nA, m_adjust = -0.4, h_adjust = -0.5	Closed_Loop_FA_WN_Compare_TX.m
21(b)	WNCL_35(0.6)	1388	Post-mod, mean input 0.6 nA	Closed_Loop_FA_WN_Compare_TX.m
15	WNCL_69(0.1)	833	Pre-mod, mean input 0.1 nA	Closed_Loop_FA_WN_Compare3_TX.m
15	WNCL_11a(0.6)	833	Pre-mod, mean input 0.6 nA	Closed_Loop_FA_WN_Compare3_TX.m
15	WNCL_70(1.5)	833	Pre-mod, mean input 1.5 nA	Closed_Loop_FA_WN_Compare3_TX.m
16	ABCL_9	833	Pre-mod, Auto Bode, 9 inputs	Closed_Loop_Batch_Bodes_TX.m
12(a & b)	WNOL_64(0.2)	n/a	Open Loop with Kd gain	Open_Loop_FA_WN_Compare_TX.m
12(a & b)	WNOL_66(1.5)	n/a	Open Loop with Kd gain	Open_Loop_FA_WN_Compare_TX.m
13(a & b)	ABOL_12	n/a	Open Loop with Kd gain, 3 inputs	Open_Loop_FA_WN_Compare_TX.m
11	ABOL_10	n/a	Open Loop neutral settings, 2 inputs	Open_Loop_FA_WN_AB_Compare_TX.m
6	BVL_SurfaceTilt_	n/a	Human data and parameters from Dr. Peterka	Fitting_closed_loop.m

6.2 Matlab Scripts and Animatlab Files for Simulation and Data Viewing

Matlab Script	Associated Program	Purpose
Closed_Loop_whiteNoise.m	PD Controller 200 Hz WN	to perform closed loop simulation via serial with Animatlab and save data files
Closed_Loop_Impulse.m	any 200 Hz Animatlab program	"
Closed_Loop_Auto_Bode.m	PD Controller 200 Hz Auto Bode	"
Open_Loop_whiteNoise.m	PD Controller 400 Hz White Noise TX	to perform open loop PD simulation via serial with Animatlab save data files
Open_Loop_Auto_Bode.m	PD Controller 400 Hz Auto Bode	"
Closed_Loop_NN_read_IOdata_files.m	n/a	to read time data from Auto Bode runs
Closed_Loop_NN_read_IOdata_files_WN.m	n/a	to read time data from White Noise runs
Open_Loop_NN_read_IOdata_files.m	n/a	to read time data from Auto Bode runs
Open_Loop_NN_read_IOdata_files_WN.m	n/a	to read time data from White Noise runs