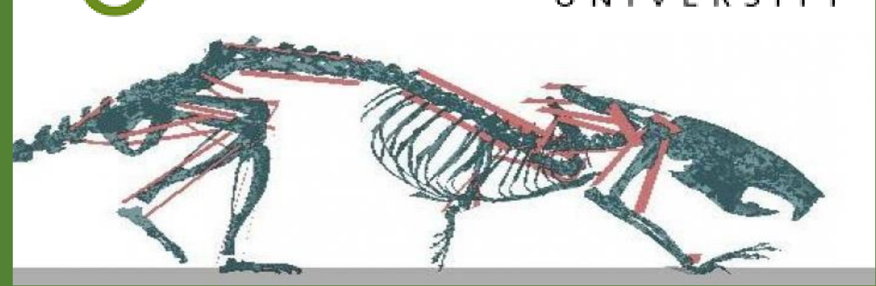
May 8th, 11:00 AM - 1:00 PM

Deriving Analytical Design Constraints For Absolute & Relative Encoding Schemes In Functional Subnetworks

Cody W. Scharzenberger-Braet
Portland State University

Alexander Hunt
Portland State University

Portland State UNIVERSITY



Agile & Adaptive Robotics Lab

A Comparison of Absolute & Relative Neural Encoding Schemes in Arithmetic Functional Subnetworks

Cody Scharzenberger[1], Alexander Hunt[1]

[1]Department of Mechanical & Materials Engineering, Portland State University, Portland, OR

National Science Foundation

C3NS NeuroNex Communication, Coordination, and Control in Neuromechanical Systems

Abstract

As neural networks have become increasingly prolific solutions to modern problems in science and engineering, there has been a congruent rise in the popularity of the numerical machine learning techniques used to design them. While numerical methods are highly generalizable, they also tend to produce unintuitive networks with inscrutable behavior. One solution to the problem of network interpretability is to use analytical design techniques, but these methods are relatively under-developed compared to their numerical alternatives. To facilitate the utilization of analytical techniques, this work extends previous efforts to quantify the impact that non-spiking neural encoding schemes have on the approximation quality of the arithmetic subnetworks of the functional subnetwork approach (FSA). In particular, novel design constraints are derived for inversion, division, and multiplication functional subnetworks using: (1) an "absolute" encoding scheme in which information is represented by the membrane voltages of the subnetwork's constituent neurons, and (2) a "relative" encoding scheme wherein information is represented by the percent activation of the subnetwork's constituent neurons. Numerical simulation results for each type of subnetwork indicate that there are both qualitative and quantitative advantages to selecting a relative encoding scheme over an absolute one, including an increased approximation accuracy of 3%-6% for normal operational ranges, greater numerical conditioning, and the freedom to choose more biologically realistic subnetwork parameters.

Background

Functional Subnetwork Approach (FSA): [1,2]
The FSA provides analytical rules for designing subnetworks that perform simple mathematical operations at steady state. It encodes information directly in the membrane voltages of the neurons.

1. Basic: Transmission, Modulation → 2. Arithmetic: Addition, Subtraction, Multiplication, Division → 3. Calculus: Derivation, Integration

The goal of this work is to compare a different "relative" encoding with the existing "absolute" encoding from prior work.

Neuron Model (Leaky Integrator): [1, 2, 3]
Consider a network of $n \in \mathbb{N}$ leaky integrator neurons. Then $\forall i \in \mathbb{N}_{\le n}$,

$$C_{m,i}\dot{U}_i = -G_{m,i}U_i + \sum_{j=1}^{n} g_{s,ij}\min\left(\max\left(\frac{U_j}{R_j},0\right),1\right)\left(\Delta E_{s,ij} - U_i\right) + I_{a,i} \quad (1)$$

where: U_i, $C_{m,i}$, $G_{m,i}$ are the membrane voltage, capacitance, and conductance of the ith neuron, respectively; R_j is the activation domain of the jth neuron; $g_{s,ij}$ and $\Delta E_{s,ij}$ are the conductance and reversal potential of the synapse from neuron j to i, respectively; and $I_{a,i}$ are the external currents applied to neuron i.

Leaky Integrator Steady State Behavior: [3]
Consider a system of $n \in \mathbb{N}$ neurons with each of the first $n-1$ neurons connected to the final nth neuron via some combination of excitatory and inhibitory synapses. The steady state membrane voltage of the output neuron is then

$$U_n^* = \frac{\sum_{i=1}^{n-1} g_{s,ni}\min\left(\max\left(\frac{U_i^*}{R_i},0\right),1\right)\Delta E_{s,ni} + I_{app,n}}{\sum_{i=1}^{n-1} g_{s,ni}\min\left(\max\left(\frac{U_i^*}{R_i},0\right),1\right) + G_{m,n}} \quad (2)$$

Eq. 2 describes the steady state behavior of the upcoming addition, subtraction, inversion, and division subnetwork architectures.

Addition/Subtraction Subnetwork

Consider the same network architecture whose steady state response is described by Eq. 2 (simplified architecture in Fig. 3a).

Absolute Subtraction: The membrane voltage of the output neuron is the sum of the membrane voltages of the excitatory input neurons less that of the inhibitory input neurons (Eq. 3a).

Relative Subtraction: The activation ratio of the output neuron is the average activation ratio of the excitatory input neurons less that of the inhibitory input neurons (Eq. 3b).

(3a) $$U_n^* = c\sum_{i=1}^{n-1} s_i U_i^*$$

(3b) $$U_n^* = cR_n\left(\frac{1}{n^+}\sum_{k=1}^{n^+}\frac{U_{i_k^+}^*}{R_{i_k^+}} - \frac{1}{n^-}\sum_{k=1}^{n^-}\frac{U_{i_k^-}^*}{R_{i_k^-}}\right)$$

To achieve agreement between the achieved (Eq. 2) and desired (Eq. 3a & 3b) steady state response, several design constraints must be satisfied, the most important of which are the synaptic conductances:

$$g_{s,ni} = \frac{I_{a,n} - cs_iG_{m,n}R_i}{cs_iR_i - \Delta E_{s,ni}} \qquad g_{s,ni} = \frac{n_i^\pm I_{a,n} - cs_iG_{m,n}R_n}{cs_iR_n - n_i^\pm \Delta E_{s,ni}}$$

where: s_i is the sign of the ith input neuron; $n_i^\pm$ is the number of excitatory inputs or the number of inhibitory inputs, depending on the sign of s_i. See supplementary materials for other design constraints.

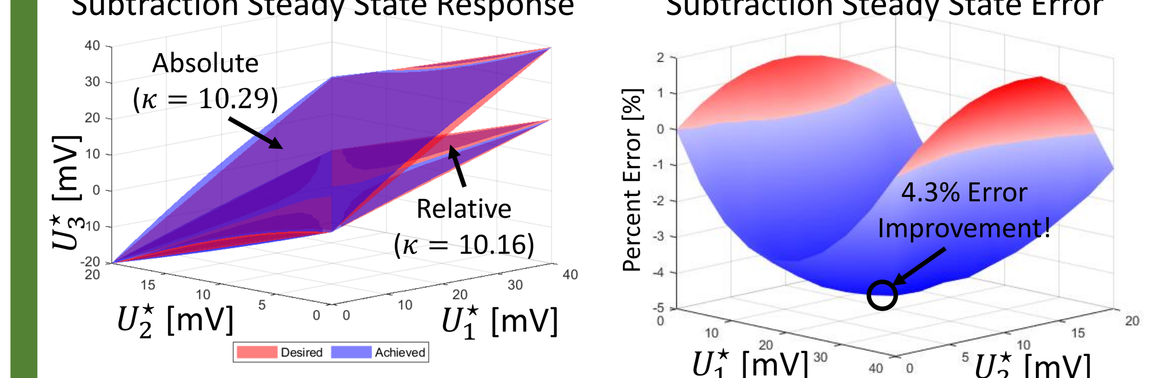


Figure 1. (a) A comparison of the steady state responses of an example absolute subtraction subnetwork ($R_1 = 40$[mV], $R_2 = 20$[mv], $R_3 = 40$[mV]) and an example relative subtraction subnetwork ($R_1 = 40$[mV], $R_2 = 20$[mv], $R_3 = 20$[mV]). (b) The difference in the percent error of each encoding scheme. Blue regions indicate that the "relative" scheme has less error, red regions indicate that the "absolute" scheme has less error.

Inversion Subnetwork

Consider a single input neuron connected to a single output neuron via an excitatory synapse as shown in Fig. 3b. This satisfies a simplified version of Eq. 2.

Absolute Inversion: The membrane voltage of the output neuron is a constant over the membrane voltage of the input neuron plus a small constant (Eq. 4a).

Relative Inversion: The activation ratio of the output neuron is a constant over the activation ratio of the input neuron plus a small constant (Eq. 4b).

(4a) $$U_3^* = \frac{c_1}{c_2U_2^* + c_3}$$

(4b) $$U_3^* = \frac{c_1R_2R_3}{c_2U_2^* + c_3R_2}$$

To achieve agreement between the achieved (Eq. 2) and desired (Eq. 4a & 4b) steady state membrane voltages, several design constraints must be satisfied, including the synaptic conductances:

$$g_{s,21} = c_2 = \frac{c_1 - \delta c_3}{\delta R_1} \qquad g_{s,21} = c_2 = \frac{(R_2 - \delta)c_3}{\delta}$$

Inversion Subnetwork

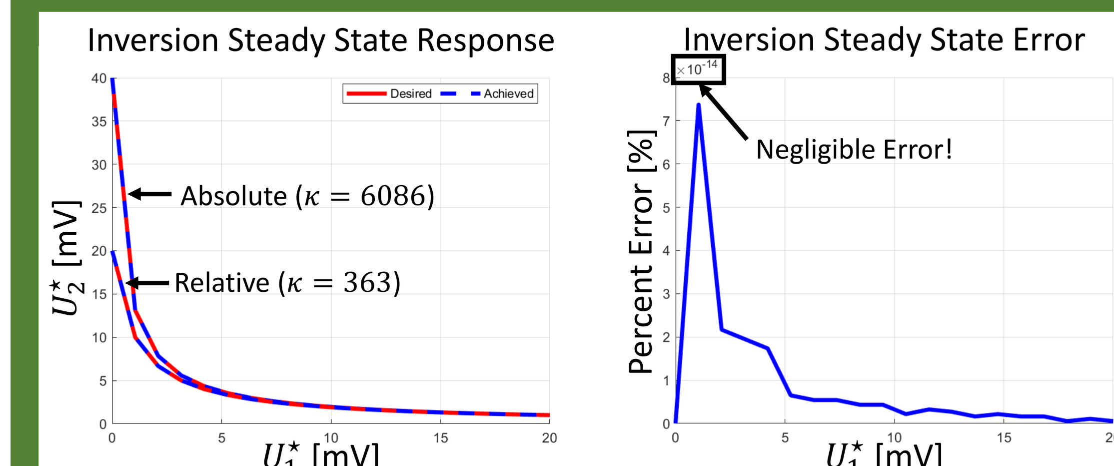


Figure 2. (a) A comparison of the steady state responses of an example absolute inversion subnetwork ($R_1 = 20$[mV], $R_2 = 40$[mv]) and an example relative inversion subnetwork ($R_1 = 20$[mV], $R_2 = 20$[mv]). (b) The difference in the percent error of each encoding scheme. Since the inversion operation can be approximated exactly (less numerical error), both schemes are nearly identical to machine precision.

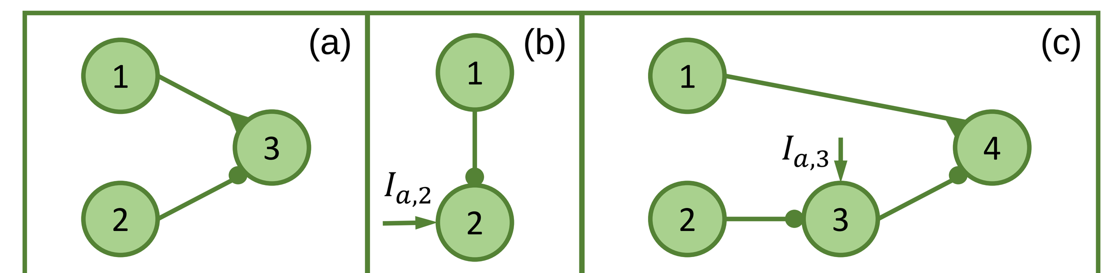


Figure 3. Functional subnetwork architectures: (a) Subtraction & Division (different parameters); (b) Inversion; and (c) Multiplication.

Division Subnetwork

Consider a network comprised of one excitatory input neuron and one inhibitory input neuron connected to a single output neuron as shown in Fig. 3a. Then this network satisfies a simplified version of Eq. 2.

Absolute Division: The membrane voltage of the output neuron is the quotient of the membrane voltages of the input neurons (Eq. 14a).

Relative Division: The activation ratio of the output neuron is the quotient of the activation ratios of the input neurons (Eq. 14b).

(5a) $$U_3^* = \frac{c_1U_1^*}{c_2U_2^* + c_3}$$

(5b) $$U_3^* = \frac{c_1R_2R_3U_1^*}{c_2R_1U_2^* + R_1R_2c_3}$$

To achieve agreement between the achieved (Eq. 2) and desired (Eq. 5) steady state membrane voltages, several design constraints must be satisfied, including the synaptic conductances:

$$g_{s,31} = \frac{c_1c_3}{R_2\left(c_3\Delta E_{s,31} - c_1R_1\right)} \qquad g_{s,31} = \frac{c_3R_3}{\Delta E_{s,31} - R_3}$$

$$g_{s,32} = \frac{c_3\Delta E_{s,31}\left(\Delta E_{s,31}c_3 - c_1R_1\right)}{R_1R_2\delta\left(c_1R_1 - \Delta E_{s,31}c_3\right)} \qquad g_{s,32} = \frac{c_3\Delta E_{s,31}(\delta - R_3)}{\delta\left(R_3 - \Delta E_{s,31}\right)}$$

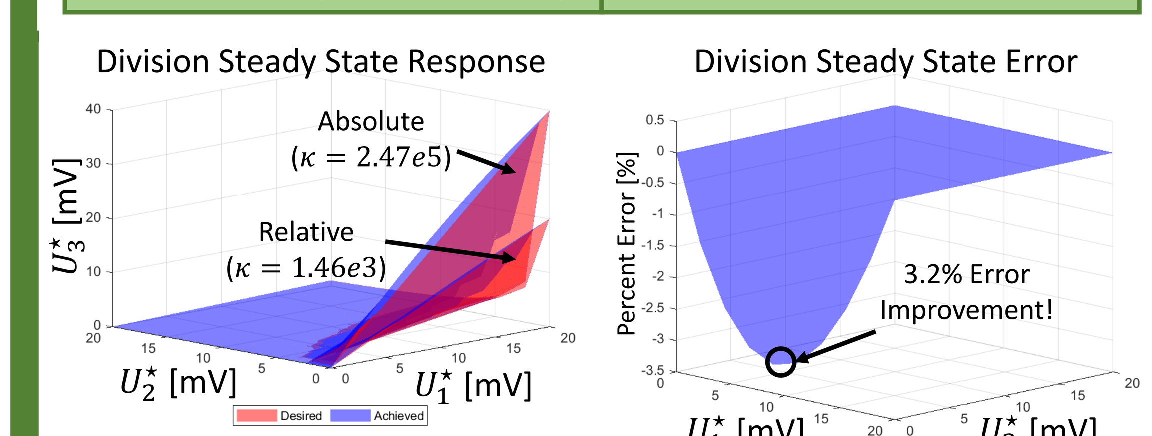


Figure 4. (a) A comparison of the steady state responses of an example absolute division subnetwork ($R_1 = 20$[mV], $R_2 = 20$[mv] , $R_3 = 40$[mv]) and an example relative division subnetwork ($R_1 = 20$[mV], $R_2 = 20$[mv] , $R_3 = 20$[mv]) (b) The difference in the percent error of each encoding scheme. In this example, the "relative" encoding scheme is always more accurate than the "absolute" encoding scheme.

Multiplication Subnetwork

Consider a network comprised of an inversion subnetwork in parallel with a division subnetwork as shown in Fig. 3c. The steady state membrane voltage of the output is

$$U_4^* = \frac{k_1U_1^*U_2^* + k_2U_1^* + k_3U_2^*}{k_4U_1^*U_2^* + k_5U_1^* + k_6U_2^*}$$

where the k_i are long expressions in the supplementary material.

Absolute Multiplication: The membrane voltage of the output neuron is the product of those of the input neurons (Eq. 6a).

Relative Multiplication: The activation ratio of the output neuron is the product of those of the input neurons (Eq. 6b).

(6a) $$U_4^* = \frac{c_2c_4U_1^*U_2^* + c_3c_4U_1^*}{c_2c_6U_2^* + c_1c_5 + c_3c_6}$$

(6b) $$U_4^* = \frac{c_2c_4R_4U_1^*U_2^* + c_3c_4R_2R_4U_1^*}{c_2c_6R_1U_2^* + (c_3c_6 + c_1c_5)R_1R_2}$$

The previously derived inversion and division subnetworks provide a convenient avenue for designing a multiplication subnetwork.

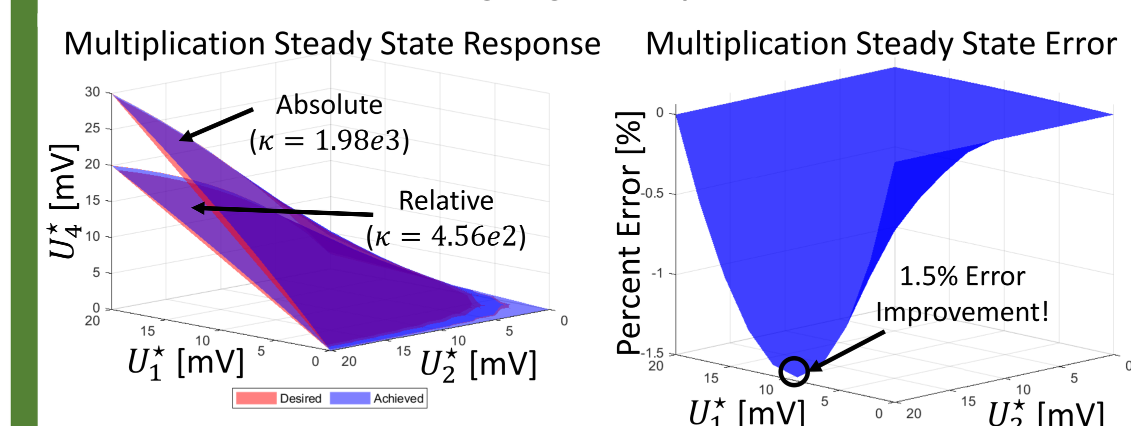


Figure 5. (a) A comparison of the steady state responses of an example absolute multiplication subnetwork ($R_1 = 20$[mV], $R_2 = 20$[mv] , $R_4 = 30$[mv]) and an example relative multiplication subnetwork ($R_1 = 20$[mV], $R_2 = 20$[mv] , $R_4 = 20$[mv]) (b) The difference in the percent error of each encoding scheme. In this example, the "relative" encoding scheme is always more accurate than the "absolute" encoding scheme.

Conclusions

1. For biological parameters, a relative encoding scheme produces a better numerically conditioned and more accurate approximation. especially as gains are increased.
2. Relative encoding schemes have practical design advantages, such as bounding the operational domain of the output.
3. Relative & absolute encoding schemes are isomorphic. This means that it is possible to choose contrived parameters that make their approximations theoretically the same, not accounting for numerical conditioning.

Acknowledgements

Supported by NSF DBI 2015317 as part of the NSF/CIHR//DFGFRQ/UKRI-MRCNext Generation Networks for Neuroscience Program.

References

[1] N. S. Szczecinski, A. J. Hunt, and R. D. Quinn, "A Functional Subnetwork Approach to Designing Synthetic Nervous Systems That Control Legged Robot Locomotion," *Front. Neurorobot.*, vol. 11, 2017, doi: 10.3389/fnbot.2017.00037.
[2] N. S. Szczecinski, R. D. Quinn, and A. J. Hunt, "Extending the Functional Subnetwork Approach to a Generalized Linear Integrate-and-Fire Neuron Model," *Front. Neurorobot.*, vol. 14, p. 577804, Nov. 2020, doi: 10.3389/fnbot.2020.577804.
[3] C. Scharzenberger and A. Hunt, "A Comparison of Absolute & Relative Neural Encoding Schemes in Addition & Subtraction Functional Subnetworks," *Biomimetic and Biohybrid Systems*, p. 12, Jul. 2023.