# PinT: Polynomial in Temperature Decode Weights in a Neuromorphic Architecture

Scott Reid, Antonio Montoya, and Kwabena Boahen *Fellow, IEEE*
Electrical Engineering, Stanford University

***Abstract*—We present Polynomial in Temperature (PinT) decode weights, a novel approach to approximating functions with an ensemble of silicon neurons that increases thermal robustness. In mixed-signal neuromorphics, computing accurately across a wide range of temperatures is challenging because of individual silicon neurons' thermal sensitivity. To compensate for the resulting changes in the neuron's tuning-curves in the PinT framework, weights change continuously as a polynomial function of temperature. We validate PinT across a 38°C range by applying it to tuning curves measured for ensembles of 64 to 1936 neurons on Braindrop, a mixed-signal neuromorphic chip fabricated in 28-nm FDSOI CMOS. LinT, the Linear in Temperature version of PinT, reduces error by a small margin on test data, relative to an ensemble with temperature-independent weights. LinT and higher-order models show much greater promise on training data, suggesting that performance can be further improved. When implemented on-chip, LinT's performance is very similar to the performance with temperature-independent decode weights. SpLinT and SpLSAT, the Sparse variants of LinT and LSAT, are promising avenues for efficiently reducing error. In the SpLSAT model, up to 90% of neurons on chip can be deactivated while maintaining the same function-approximation error.**

***Index Terms*—mixed-signal neuromorphics, thermal robustness**

## I. Thermally Robust Neuromorphic Computation

In the Neural Engineering Framework (NEF), optimized temperature-independent decode-weights are used to approximate functions [1]. Functions are approximated with weighted sums of spiking neuron responses (Fig. 1). The optimal decode-weights are found by minimizing the function-approximation error. When this optimization is done using tuning curves measured at a single training temperature, error grows quickly as the temperature deviates if the tuning curves are thermally sensitive. Such thermal sensitivity is found in biological systems as well as mixed-signal neuromorphics.

Least Squares Across Temperature (LSAT) extends decode-weight optimization to account for tuning-curves' thermal variation. It yields thermally robust behavior, but suffers much greater error at each temperature than the minimum possible error (i.e. if training was at just that temperature). Therefore, more neurons are needed to match the error achieved at a single temperature. On Neurogrid, it was found that 560 neurons are required to achieve the same precision across a 2°C range as 35 neurons achieved at a single temperature [2].

To improve the thermal robustness of function approximation, we introduce Polynomial in Temperature (PinT) decode weights to mitigate tuning-curves' thermal variation. In PinT, as the ambient temperature changes, new weights

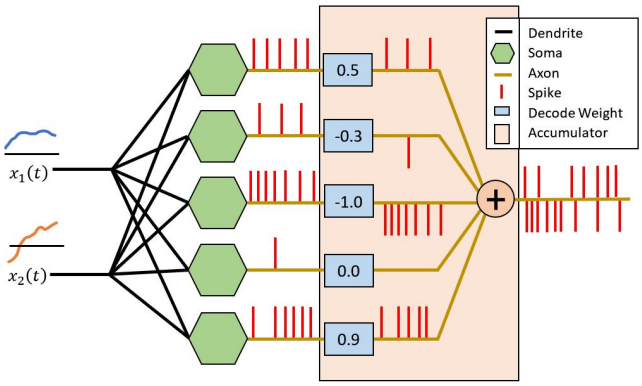


Fig. 1. **Braindrop Spiking Neural Network Architecture**
A multidimensional input signal $\mathbf{x}(t)$ drives an ensemble of neurons (*green*), thereby encoding the input signal into spike-trains (*red*). Decode weights (*blue*) are then applied to the spike-trains. Their magnitudes determine the probability that a spike will be output. Their sign determines if a spike is inverted. The decode weights are optimized so that the output spikes' instantaneous rate approximates a desired function $f(\mathbf{x}(t))$ [5].

are computed through a polynomial series expansion of the temperature, using coefficients stored in digital memory. In its Sparse Linear in Temperature (SpLinT) variant, negligible increases in weight memory and computation to update weights yield improvements to thermal robustness. In the Sparse LSAT (SpLSAT) model, thermal robustness and function-approximation accuracy are maintained after deactivating up to 90% of the neurons on the chip, yielding significant reductions to memory and energy costs.

Section II reviews the current approach to thermal robustness using temperature-independent decode weights. Section III introduces the PinT framework. Section IV introduces an error operator that acts on the function space to directly measure the function-approximation error across temperature. Section V compares the accuracy of different-sized ensembles of Braindrop silicon neurons with PinT and LSAT decode-weights, and validates the on-chip performance for LSAT and LinT across a 38°C range. Section VI introduces the Sparse Linear in Temperature (SpLinT) and Sparse Least Squares Across Temperature (SpLSAT) frameworks.

## II. Temperature-Independent Decode Weights

A $k$-dimensional input signal $\mathbf{x}(t) \in \mathbb{R}^D$ is encoded into the spike-rates of a population of $N$ neurons [4] by means of a nonlinear transformation. The $j^{\text{th}}$ neuron's spike rate as

978-3-9819263-2-3/DATE19/©2019 EDAA

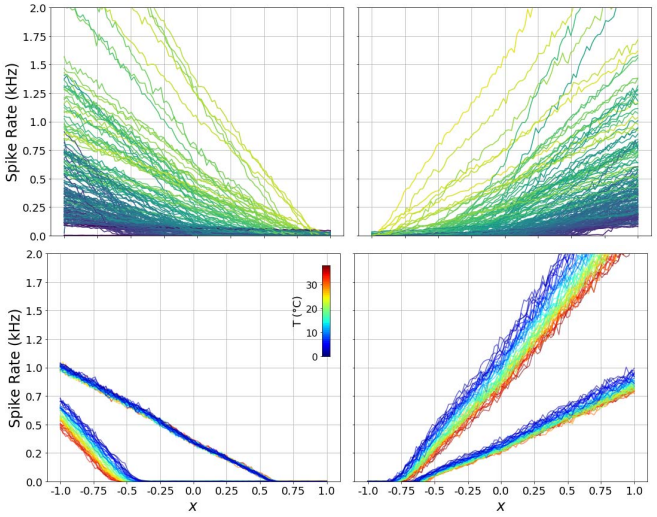


Fig. 2. **Tuning Curves Measured from 484 Braindrop Neurons**
**Top:** Spiking thresholds (*color coded*) and maximum spike-rates widely vary across the population at a single temperature (26°C). **Bottom:** As temperature changes, the thresholds and gains of individual neurons tend to change.

a function of the input $\mathbf{x}$, also known as its tuning curve, is given by,

$$a_j(\mathbf{x}) = G(\alpha_j(\mathbf{e}_j \cdot \mathbf{x}) + \beta_j) \tag{1}$$

where $G$ is the neuronal transfer-function, which maps input currents to steady-state spike-rates; $\alpha_j$ is a gain factor; $\mathbf{e}_j \in \mathbb{R}^D$ is a unit encoding vector that points in the direction of maximal activation; and $\beta_j$ is a bias. Across a population of neurons, diversity in encoding vectors, gains, and biases leads to varied responses, thereby encoding all of the information from the input signal into time-varying spike-rates across the population (Fig. 1). On Braindrop, neuronal tuning-curves resemble ReLU (Rectified Linear Unit) activations with temperature-dependent biases and gains (Fig. 2).

Tuning curves of an ensemble of neurons are measured across a range of temperatures to train optimal decode weights. We discretely sample the input signal $\mathbf{x} \in \mathbb{R}^D$ at $Q$ points $\mathbf{x}_1, \ldots, \mathbf{x}_Q$ rastered across the input space. The tuning curve $\mathbf{a}_j \in \mathbb{R}^Q$ of the $j^{\text{th}}$ neuron is measured at a particular temperature $T$ across the discretized inputs such that $(\mathbf{a}_j)_k = a_j(\mathbf{x}_k)$. We define a tuning-curve matrix $\mathbf{A}_T \in \mathbb{R}^{Q\times N}$ such that $\mathbf{A}_T$'s $j^{\text{th}}$ column is the $j^{\text{th}}$ neuron's tuning-curve at temperature $T$.

An approximation $\hat{f}(\mathbf{x})$ to a target function $f(\mathbf{x}) : \mathbb{R}^D \to \mathbb{R}$ is decoded from an ensemble through a weighted sum of its neuronal responses (see Fig. 1). The approximation decoded at temperature $T$, $\hat{\mathbf{f}} \in \mathbb{R}^Q$, defined on inputs $\mathbf{x}_1, \ldots, \mathbf{x}_Q$, is given by,

$$\hat{\mathbf{f}} = \mathbf{A}_T\mathbf{d} \tag{2}$$

The decode-weight vector $\mathbf{d} \in \mathbb{R}^N$ is defined such that $\mathbf{d}_j$ is the weight applied to the $j^{\text{th}}$ neuron's tuning curve.

Since tuning curves are measured in the presence of random noise, $\mathbf{d}$ is optimized to minimize the function-approximation error's expected value. Assuming that the noise is Gaussian

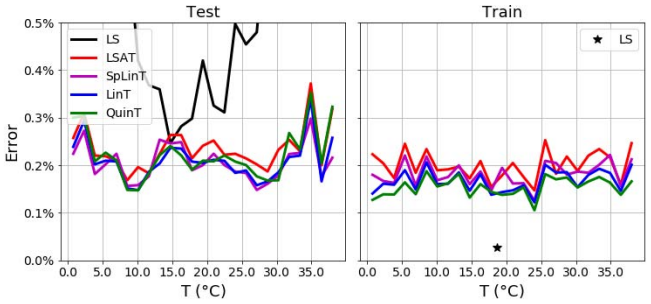


Fig. 3. **Decoding $f(x) = x^3$ Across Temperature**
For tuning curves measured from an ensemble of 100 neurons, LS achieves low error at the training temperature (*black star*), at the expense of large error elsewhere (*black curve*). LSAT, SpLinT (10% of neurons with LinT weights), LinT, and QuinT each achieve a uniform error across temperature. Noise regularization $\sigma = 0.05$ Hz was used for training.

with mean zero and standard deviation $\sigma$, we define a noise matrix $\mathbf{Z} \in \mathbb{R}^{Q\times N}$ with elements drawn from $\mathcal{N}(0, \sigma)$. Thus, the decoded function with noise is given by $\hat{\mathbf{f}}^\sigma = (\mathbf{A}_T + \mathbf{Z})\mathbf{d}$. The expected value of this approximation's squared-error gives us the Least Squares (LS) objective function.

$$J_{\text{LS}} = \text{E}(||(\mathbf{A}_T + \mathbf{Z})\mathbf{d} - \mathbf{f}||^2) \tag{3}$$

where $\mathbf{f} \in \mathbb{R}^Q$ is the discretized target function. We can rewrite this as,

$$J_{\text{LS}} = \mathbf{d}^\top(\mathbf{A}_T^\top\mathbf{A}_T + \sigma^2 QN\mathbb{I})\mathbf{d} - 2\mathbf{d}^\top\mathbf{A}_T^\top\mathbf{f} + \mathbf{f}^\top\mathbf{f} \tag{4}$$

making use of the expectations: $\text{E}(\mathbf{Z}^\top\mathbf{Z}) = \sigma^2 QN\mathbb{I}$, $\text{E}(\mathbf{Z}^\top\mathbf{A}_T) = 0$, and $\text{E}(\mathbf{Z}^\top\mathbf{f}) = 0$, where $\mathbb{I} \in \mathbb{R}^{N\times N}$ is the identity matrix. The L2 noise term $\sigma^2 QN\mathbf{d}^\top\mathbf{d}$ penalizes large decoding weights. We take the gradient of $J_{\text{LS}}$ with respect to $\mathbf{d}$ and set the resulting equation to zero, yielding:

$$\mathbf{d}_T^\star = (\mathbf{A}_T^\top\mathbf{A}_T + \sigma^2 QN\mathbb{I})^{-1}\mathbf{A}_T^\top\mathbf{f} \tag{5}$$

The correlation matrix $\mathbf{A}_T^\top\mathbf{A}_T$ represents the similarity between neurons' tuning curves: $(\mathbf{A}_T^\top\mathbf{A}_T)_{ij} = \mathbf{a}_i^\top\mathbf{a}_j$ is the dot product of the $i^{\text{th}}$ and $j^{\text{th}}$ neuron's tuning curves.

Since $\mathbf{d}_T^\star$ is trained only at temperature $T$, error increases as the temperature deviates from $T$ (Fig. 3). The function decoded at different temperatures is given by,

$$\hat{\mathbf{f}}(T_i) = \mathbf{A}_{T_i}\mathbf{d} \quad \text{and} \quad \begin{bmatrix}\hat{\mathbf{f}}(T_1)\\ \vdots \\ \hat{\mathbf{f}}(T_R)\end{bmatrix} = \begin{bmatrix}\mathbf{A}_{T_1}\\ \vdots \\ \mathbf{A}_{T_R}\end{bmatrix}\mathbf{d} \tag{6}$$

where $\{\mathbf{A}_{T_i}\}_{i=1}^R$ is the set of tuning curve matrices measured at $R$ different temperatures. This equation can be rewritten as $\hat{\mathbb{F}} = \mathbb{A}_0\mathbf{d}$, where $\mathbb{A}_0 \in \mathbb{R}^{RQ\times N}$ is a matrix of $R$ vertically-stacked tuning-curve matrices and $\hat{\mathbb{F}} \in \mathbb{R}^{RQ}$ is a vector of $R$ stacked decoded functions.[1] Since $\mathbf{A}_{T_i}$ varies with temperature (see Fig. 2), error increases as the temperature deviates from the training temperature.

[1] $\mathbb{A}_0$ has the subscript 0 because it is used in the case of $0^{\text{th}}$ order in temperature decode-weights. It is generalized to higher temperature order in the PinT framework.

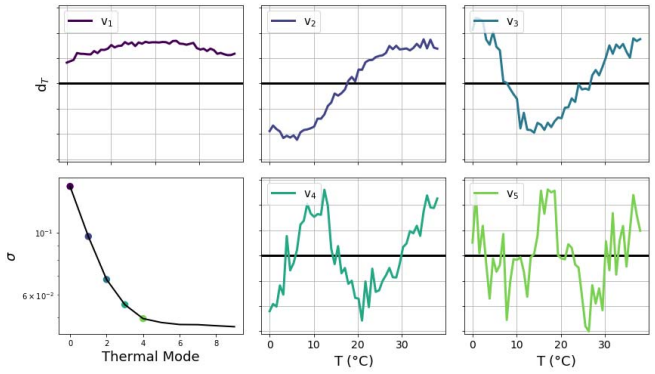


Fig. 4. **Thermal Decode Modes**
Successive modes resemble higher and higher order polynomials, and capture less and less (*bottom left*) of the thermal variation of the optimal decoding vector.

The Least Squares Across Temperature (LSAT) objective function includes $\mathbf{A}_{T_i}$'s variation across the temperature range,

$$J_{\text{LSAT}} = \tfrac{1}{R}||(\mathbb{A}_0 + \mathbb{Z})\mathbf{d} - \mathbb{G}\mathbf{f}||^2 \tag{7}$$

Here, $\mathbb{Z} \in \mathbb{R}^{RQ\times N}$ is a matrix of $R$ vertically stacked $\mathbb{R}^{Q\times N}$ noise matrices and $\mathbb{G} \in \mathbb{R}^{RQ}$ is a matrix of $R$ vertically stacked $Q$-dimensional identity matrices. Thus, $\mathbb{G}\mathbf{f} = \mathbb{F}$ is a stacked vector with $R$ copies of $\mathbf{f}$, reflecting our objective to decode the same function at each of the $R$ temperatures. This objective function can be rewritten,

$$J_{\text{LSAT}} = \tfrac{1}{R}\Big(\mathbf{d}^\top(\mathbb{A}_0^\top\mathbb{A}_0 + \sigma^2 QNR\mathbb{I})\mathbf{d} - 2\mathbf{d}^\top\mathbb{A}_0^\top\mathbf{f}\Big) + \mathbf{f}^\top\mathbf{f} \tag{8}$$

and is minimized by,

$$\mathbf{d}^\star_{\text{LSAT}} = (\mathbb{A}_0^\top\mathbb{A}_0 + \sigma^2 QNR\mathbb{I})^{-1}\mathbb{A}_0^\top\mathbb{G}\mathbf{f} \tag{9}$$

where $\mathbb{I} \in \mathbb{R}^{N\times N}$. Note that $\mathbb{A}_0^\top\mathbb{A}_0 = \sum_{i=1}^R \mathbf{A}_{T_i}^\top\mathbf{A}_{T_i}$ and $\mathbb{A}^\top\mathbb{G} = \sum_{i=1}^R \mathbf{A}_{T_i}^\top$ are the sums of correlations of tuning-curve matrices and tuning-curve matrices across temperature, respectively. Unlike LS, which achieves low error only at a single temperature and higher error elsewhere, LSAT achieves a uniform intermediate error (see Fig. 3).

## III. Polynomial in Temperature Decode Weights

The optimal LS decode vector $\mathbf{d}^\star_T$ varies continuously with training temperature $T$, and this variation can be decomposed into thermal decode-modes that closely resemble monomial functions. We take the Singular Value Decomposition (SVD) of a matrix whose $i^{\text{th}}$ column is a stacked vector of $\mathbf{d}^\star_{T_i}$ for a set of functions $\{\mathbf{f}_1, \ldots, \mathbf{f}_M\}$ (we chose the set of functions $\mathbf{f}_n = \mathbf{x}^n$, and $M = 10$), trained at temperature $T_i$. The right singular-vector $\mathbf{v}_k$ is the $k^{\text{th}}$ thermal-decode mode. The first three thermal decode-modes—which capture most of $\mathbf{d}^\star_T$'s thermal variation—resemble constant, linear, and quadratic functions of temperature (Fig. 4).

Inspired by the thermal decode-modes' monomial nature, the PinT framework strives to stably decode functions across temperature by expressing decode weights as polynomial functions of temperature,

$$\mathbf{d}(T) = \sum_{n=0}^{P} T^n\mathbf{d}_n \tag{10}$$

called $P^{\text{th}}$-order PinT weights. The coefficients $\mathbf{d}_0, \ldots, \mathbf{d}_P$ are vectors in $\mathbb{R}^N$, where $N$ is the number of neurons in the ensemble. The linear case is referred to as LinT, for Linear in Temperature ($P = 1$). Similarly, QuinT refers to Quadratic in Temperature weights ($P = 2$), and TrinT refers to Cubic in Temperature weights ($P = 3$). As we will see, LSAT weights are actually zeroth-order PinT weights.

In order to express PinT function-decoding as a linear transformation, we introduce $\mathbb{D}_P \in \mathbb{R}^{(P+1)N}$, the stacked vector of decode coefficients, and $\mathbb{A}_P \in \mathbb{R}^{RQ\times(P+1)N}$, an extension of $\mathbb{A}_0$, LSAT's vertically stacked tuning curve matrices. By definition, the function PinT decodes at a single temperature $T_i$ is given by,

$$\hat{\mathbf{f}}(T_i) = \sum_{n=0}^{P} T_i^n\mathbf{A}_{T_i}\mathbf{d}_n \tag{11}$$

The stacked vector of functions decoded at the $R$ temperatures can be written as the linear transformation,

$$\hat{\mathbb{F}} = \mathbb{A}_P\mathbb{D}_P, \text{ where } \mathbb{A}_P = \begin{bmatrix}\mathbb{T}^0\mathbb{A}_0 & \mathbb{T}^1\mathbb{A}_0 & \ldots & \mathbb{T}^P\mathbb{A}_0\end{bmatrix} \tag{12}$$

Here, $\mathbb{D}_P$ is the stacked vector of decode-coefficient vectors $\mathbf{d}_0, \ldots, \mathbf{d}_P$ and $\mathbb{T}^n \in \mathbb{R}^{RQ\times RQ}$ is a diagonal matrix with the first $Q$ diagonal entries equal to $T_1^n$, the second $Q$ diagonal entries equal to $T_2^n$, and so on through $T_R^n$. Note that $\mathbb{D}_0 = \mathbf{d}_0$, hence in the case where $P = 0$, we recover the LSAT expression $\hat{\mathbb{F}} = \mathbb{A}_0\mathbf{d}_0$.

PinT's objective-function extends LSAT's objective-function to include $\mathbf{A}_{T_i}$'s and now $\mathbf{d}(T_i)$'s thermal variation. We define the generalized gaussian noise matrix, where $\mathbb{Z}$ is defined as before.

$$\mathbb{N}_P = \begin{bmatrix}\mathbb{T}^0\mathbb{Z} & \mathbb{T}^1\mathbb{Z} & \ldots & \mathbb{T}^P\mathbb{Z}\end{bmatrix} \tag{13}$$

The PinT objective function is thus given by,

$$J_{\text{PinT}} = \tfrac{1}{R}||(\mathbb{A}_P + \mathbb{N}_P)\mathbb{D}_P - \mathbb{G}\mathbf{f}||^2 \tag{14}$$

Expanding this expression and taking expected values yields:

$$J_{\text{PinT}} = \tfrac{1}{R}\Big(\mathbb{D}_P^\top(\mathbb{C}_P + \sigma^2\mathbb{B}_P)\mathbb{D}_P - 2\mathbb{D}_P^\top\mathbb{W}_P\mathbf{f}\Big) + \mathbf{f}^\top\mathbf{f} \tag{15}$$

where $\mathbb{C}_P$, $\mathbb{B}_P$, and $\mathbb{W}_P$ are defined as follows.

The generalized correlation matrix $\mathbb{C}_P = \mathbb{A}_P^\top\mathbb{A}_P$ is composed of $n^{\text{th}}$-order correlation matrices $\mathbf{C}_n \in \mathbb{R}^{N\times N}$, arranged in a block-Hankel form,

$$\mathbb{C}_P = \begin{bmatrix}\mathbf{C}_0 & \mathbf{C}_1 & \ldots & \mathbf{C}_P \\ \mathbf{C}_1 & \mathbf{C}_2 & \ldots & \mathbf{C}_{P+1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_P & \mathbf{C}_{P+1} & \ldots & \mathbf{C}_{2P}\end{bmatrix} \text{ where } \mathbf{C}_n = \mathbb{A}_0^\top\mathbb{T}^n\mathbb{A}_0 \tag{16}$$

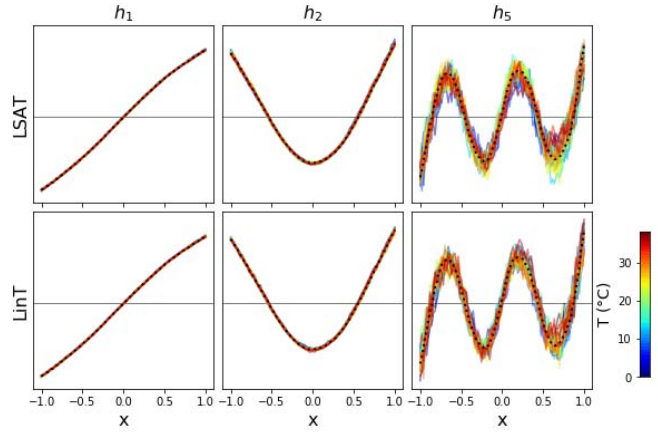


Fig. 5. **Decoding $\mathbb{H}_0$'s Eigenfunctions with LSAT and LinT Weights** With tuning-curve data from an ensemble of 256 Braindrop neurons, we decoded eigenfunctions $h_1$, $h_2$, and $h_5$ (*dotted lines*) of $\mathbb{H}_0$ (the LSAT error operator) across a 38°C range using LSAT decode-weights (*top*), and LinT decode-weights (*bottom*). As the eigenfunction's order increases, it is approximated at test temperatures (*colored lines*) less accurately (larger deviation from dotted line) and less robustly across temperature (larger spread across curves). LinT achieves slightly more accurate and robust approximations.

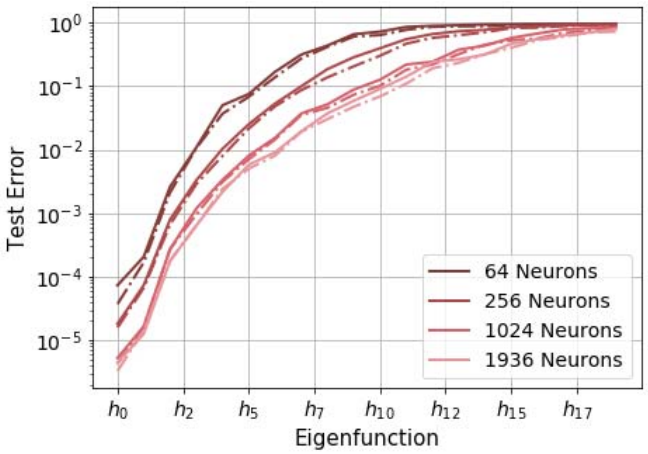


Fig. 6. **Test Error Across $\mathbb{H}_0$'s Eigenfunctions** Using Braindrop tuning curve measurements, we decoded $\mathbb{H}_0$'s eigenfunctions using LSAT (*solid lines*) and LinT (*dashed lines*) weights for different-sized ensembles. For each eigenfunction, we tuned the noise regularization parameter $\sigma$ to an optimal value that minimized test error. Optimal values of $\sigma$ ranged between 0.01 Hz and 0.5 Hz.

$\mathbf{C}_n$ has elements $(\mathbf{C}_n)_{kl} = \sum_{i=1}^{R} T_i^n \mathbf{a}_k^\top(T_i)\mathbf{a}_l(T_i)$ given by the sum over temperature of the dot-product of the tuning curves for neurons $k$ and $l$, scaled by the $n^{\text{th}}$ power of temperature.

The generalized noise-regularization matrix $\mathbb{B}_P$ is composed of $N \times N$ submatrices $\mathbf{B}_n$, arranged in a block-Hankel form similar to $\mathbb{C}_P$.

$$\mathbb{B}_P = \begin{bmatrix} \mathbf{B}_0 & \mathbf{B}_1 & \dots & \mathbf{B}_P \\ \mathbf{B}_1 & \mathbf{B}_2 & \dots & \mathbf{B}_{P+1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{B}_P & \mathbf{B}_{P+1} & \dots & \mathbf{B}_{2P} \end{bmatrix} \quad \text{where } \mathbf{B}_n = \frac{\mathbb{Z}^\top \mathbb{T}^n \mathbb{Z}}{\sigma^2} \tag{17}$$

Thus, $\mathbf{B}_n = QN\sum_{i=1}^{R} T_i^n \mathbb{I}$ where $\mathbb{I}$ is the $N$-dimensional identity matrix.

Finally, the generalized average tuning-curve matrix $\mathbb{W}_P = \mathbb{A}_P^\top \mathbb{G}$ is a vertically stacked matrix of matrices $\mathbf{W}_0, \dots, \mathbf{W}_P$, where $\mathbf{W}_n \in \mathbb{R}^{N \times Q}$ is given by $\mathbf{W}_n = \sum_{i=1}^{R} T_i^n \mathbf{A}_{T_i}^\top$.

The optimal decode-coefficient vector, $\mathbb{D}_P^\star$, which minimizes $J_{\text{PinT}}$, extends the LSAT solution $\mathbf{d}_{\text{LSAT}}^\star$ to polynomial-order decode-weights;

$$\mathbb{D}_P^\star = (\mathbb{C}_P + \sigma^2 \mathbb{B}_P)^{-1} \mathbb{W}_P \mathbf{f} \tag{18}$$

In the case where $P = 0$, we have that $\mathbb{C}_0 = \mathbb{A}_0^\top \mathbb{A}_0$, $\mathbb{B}_0 = QNR\mathbb{I}$, and $\mathbb{W}_0 = \mathbb{A}_0^\top \mathbb{G}$. Hence, $J_{\text{PinT}}$ reduces to $J_{\text{LSAT}}$ and $\mathbb{D}_P^\star$ reduces to $\mathbf{d}_{\text{LSAT}}^\star$ when $P = 0$ (see equations 8, 9, 14, and 18).

## IV. PinT Error Operator

We construct an operator $\mathbb{H}_P$ that directly measures the average squared function-approximation error for a target function $\mathbf{f}$, and can be defined separately for testing and training data. $\mathbb{H}_P \in \mathbb{R}^{Q \times Q}$ measures the squared error on a target function $\mathbf{f} \in \mathbb{R}^Q$ through a scaled inner-product,

$$\mathbf{f}^\top \mathbb{H}_P \mathbf{f} = \tfrac{1}{R} ||\hat{\mathbb{F}}_{\mathbf{f}}^\star - \mathbb{G}\mathbf{f}||^2 \tag{19}$$

where $\hat{\mathbb{F}}_{\mathbf{f}}^\star$ is the stacked vector of decoded functions at each temperature yielded by $\mathbb{D}_P^\star$, the optimized PinT decode-vector for $\mathbf{f}$. We reserve certain temperature measurements as a test set while the remaining measurements comprise the training set.

The test-error operator is derived by substituting $\mathbb{D}_P^{\text{tr}\star}$, defined on the training set with noise regularization $\sigma$, into $J_{\text{PinT}}$, defined on the test set without noise regularization. It is given by,

$$\mathbb{H}_P^{\text{te}} = \mathbb{I} + \tfrac{1}{R^{\text{te}}} \Big( \mathbb{W}_P^{\text{tr}\top} \mathbb{C}_{P,\sigma}^{\text{tr}-1} \mathbb{C}_{P,0}^{\text{te}} \mathbb{C}_{P,\sigma}^{\text{tr}-1} \mathbb{W}_P^{\text{tr}} - \mathbb{W}_P^{\text{tr}\top} \mathbb{C}_{P,\sigma}^{\text{tr}-1} \mathbb{W}_P^{\text{te}} - \mathbb{W}_P^{\text{te}\top} \mathbb{C}_{P,0}^{\text{tr}-1} \mathbb{W}_P^{\text{tr}} \Big) \tag{20}$$

Here, $\mathbb{I}$ is the $Q$-dimensional identity matrix and we use the compact notation $\mathbb{C}_{P,\sigma} = \mathbb{C}_P + \sigma^2 \mathbb{B}_P$. The training-error operator is obtained by setting $\mathbb{C}_{P,0}^{\text{te}} = \mathbb{C}_{P,0}^{\text{tr}}$, $\mathbb{W}_P^{\text{te}} = \mathbb{W}_P^{\text{tr}}$, and $R^{\text{te}} = R^{\text{tr}}$ in the expression above.

The error for a given function $\mathbf{f}$ can be expressed in terms of its projection onto $\mathbb{H}_P$'s eigenfunctions, weighted by their eigenerrors. $\mathbb{H}_P$'s eigendecomposition is given by $\mathbb{H}_P = \sum_{i=1}^{Q} \epsilon_i \mathbf{h}_i^\top \mathbf{h}_i$ where $\{\mathbf{h}_i\}_{i=1}^{Q}$ is a set of normalized eigenfunctions and $\{\epsilon_i\}_{i=1}^{Q}$ is a set of eigenerrors. The eigenerror $\epsilon_i$ gives the squared-error decoding $\mathbf{h}_i$,

$$\epsilon_i = \mathbf{h}_i^\top \mathbb{H} \mathbf{h}_i = \tfrac{1}{R} ||\hat{\mathbb{F}}_{\mathbf{h}_i}^\star - \mathbb{G}\mathbf{h}_i||^2 \tag{21}$$

Projecting a normalized function $\mathbf{f} \in \mathbb{R}^Q$ onto $\mathbb{H}_P$'s eigenfunction space as $\mathbf{f} = \sum_{i=1}^{Q} c_i \mathbf{h}_i$, where $c_i = \mathbf{f}^\top \mathbf{h}_i$, yields its function-approximation error,

$$\tfrac{1}{R} ||\hat{\mathbb{F}}_{\mathbf{f}}^\star - \mathbb{G}\mathbf{f}||^2 = \sum_{i=1}^{Q} c_i^2 \epsilon_i \tag{22}$$

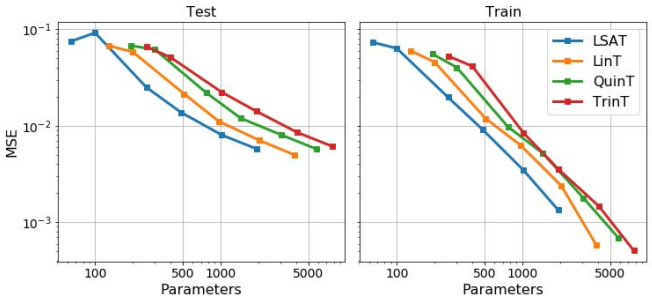


Fig. 7. **Error Scaling for $\mathbf{h}_5$ of the LSAT Error Operator $\mathbb{H}_0$**
As the number of parameters $N(P+1)$ increases, the test error decreases less rapidly than the training error. Thus, the gap between training and test error is larger for more neurons and higher order PinT weights. LSAT achieves the lowest error for a given number of parameters.

## V. PinT on Braindrop

We validated PinT's performance across a 38°C range first by evaluating function-approximation error on measured tuning-curve data and then by directly decoding functions on-chip. We collected tuning-curve data at 100 input values over 50 evenly spaced temperatures between 0°C and 38°C from five ensembles with 64, 100, 256, 484, 1024, and 1936 neurons (see Fig. 2). Throughout our analysis we reserved one quarter of the temperature measurements as a test set. The remaining temperature measurements comprised a training set.

Using the training temperature data, we trained the network to approximate the eigenfunctions of $\mathbb{H}_0$ (the LSAT error-operator) (Fig. 5) using LSAT, LinT, QuinT, and TrinT decode weights. For each eigenfunction, we tuned $\sigma$, the noise standard deviation, to minimize error at the test temperatures, treating it as a hyperparameter. The optimal value of $\sigma$ was typically in the range of $0.01$ Hz to $0.5$ Hz, and it increased gradually for higher-order eigenfunctions.

LinT achieved slightly lower test error compared to LSAT, however there was no benefit to using higher-order PinT weights. The error-reduction benefit of LinT weights compared to LSAT weights decreased as the size of the ensemble increased (Fig. 6). Higher-order PinT weights, not shown in Fig. 6, reached lower training errors than LinT but reached the same test errors as LinT, suggesting that these models with more parameters were overfitting to the training data. LSAT achieved the lowest error for a given number of parameters. LinT achieved slightly lower error than LSAT for the same number of neurons, but required twice the parameters (Fig. 7). Higher-order models did not reduce test error in proportion to their large number of parameters.

We also measured on-chip function approximation across a 38°C range for LSAT and LinT decode weights on the first 10 eigenfunctions of $\mathbb{H}_0$. The on-chip function-approximation error was consistently twice the error expected from measurements of tuning-curves. This discrepancy, although small, could be attributed to the numerical rounding of decode weights on Braindrop when they are stored with 8 bits (Fig. 8).

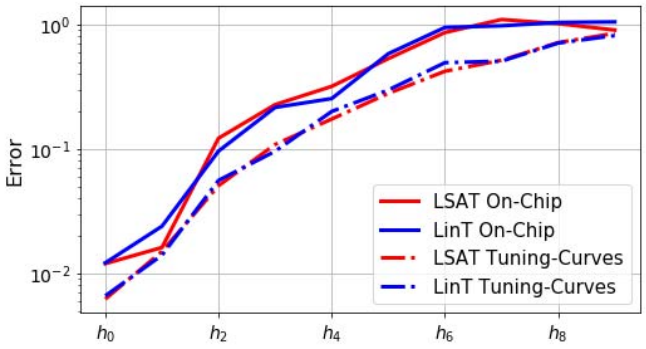


Fig. 8. **On-Chip Function Approximation Error**
The first 10 eigenfunctions of $\mathbb{H}_0$ are decoded on Braindrop for an ensemble of 100 neurons at 20 temperatures between 0°C and 38°C. LinT decode weights were implemented by manually changing decode weights at each new temperature. On-chip function approximation error is generally twice the error expected from the the LSAT and LinT training error-operator (*dashed lines*).

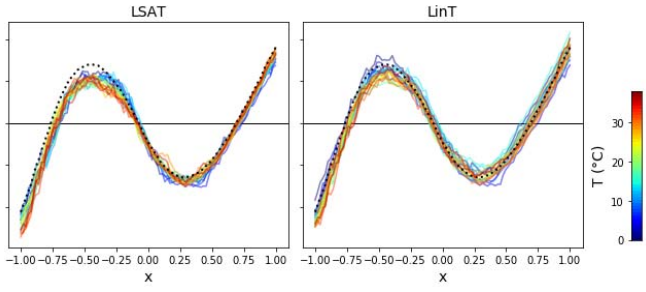


Fig. 9. **On-Chip Approximation of $\mathbf{h}_3$**
The third eigenfunction of $\mathbb{H}_0$ is approximated with an ensemble of 100 neurons on-chip across a 38°C range with LSAT and LinT decode-weights.

## VI. Sparse Decode Weight Models

In the Sparse Least Squares Across Temperature (SpLSAT) model, a sparse subset of all neurons equipped with nonzero decode weights and the rest of the neurons are disabled. In the Sparse Linear in Temperature (SpLinT) model, a sparse subset of all neurons are equipped LinT weights and the rest are given temperature-independent weights. In both models, fewer parameters are needed to achieve low function-approximation error. Additionally, in the case of SpLSAT, superfluous neurons are shut off, thus reducing the chip's power consumption.

We use an iterative Beam Search algorithm to solve for sparse decode weights. The $k^{\text{th}}$ iteration of Beam Search receives $B$ sets, each containing $k-1$ killed parameters, from the previous iteration of the algorithm, where $B$ is the *beam-width*. The $k^{\text{th}}$ iteration picks at most $B^2$ unique candidate sets of $k$ killed parameters using the heuristic that the nonzero parameters that are closest to zero are good candidates to be set to zero. This heuristic is evaluated on LSAT $\mathbf{d}_0$ weights in the case of SpLSAT and LinT $\mathbf{d}_1$ weights for SpLinT. Finally, these candidates are pruned down to the $B$ sets with the lowest training error, to be passed to the $k+1^{\text{th}}$ iteration of Beam Search.

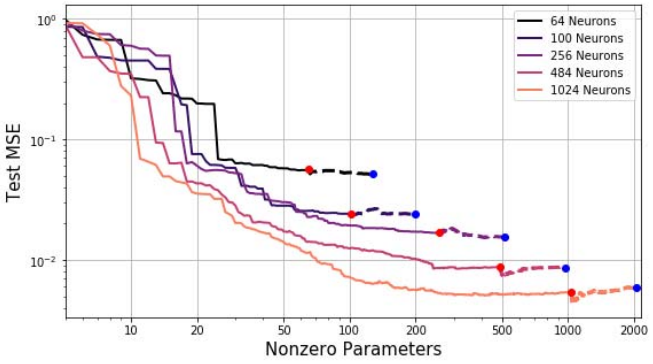


Fig. 10. **SpLSAT and SpLinT Error Scaling**
With tuning curves measured from ensembles of 64 to 1024 neurons, the eigenfunction $h_5$ of $\mathbb{H}_0$ is decoded with SpLSAT (*solid lines*) and SpLinT (*dotted lines*) decode weights. LSAT and LinT test errors are represented by red and blue circles respectively.

Given a subset $S$ of decode weight parameters to be set to zero, we obtain a SpLinT or SpLSAT solution as follows. We add to the original objective function an L2 penalty for the parameters in $S$ that will be set to zero.

$$J_{\text{Sp}} = J + \lambda \mathbb{D}^\top \mathbb{M}_S \mathbb{D} \tag{23}$$

Here, the diagonal matrix $\mathbb{M}_S$ is $\in \mathbb{R}^{N\times N}$ for SpLSAT and $\in \mathbb{R}^{2N\times 2N}$ for SpLinT. The diagonal entries of $\mathbb{M}_S$ are 1 for parameters in $S$ and 0 otherwise. This form ensures that $\mathbb{D}^\top \mathbb{M}_S \mathbb{D} = \sum_{j\in S} \mathbb{D}_j^2$, thereby penalizing non-zero values for the parameters in $S$. The parameter $\lambda$ sets the penalty's magnitude; it must be positive and as large as possible. The SpLSAT solution is expressed simply as $\mathbb{D}^\star_{Sp,0} = (\mathbb{C}_0 + \sigma^2 B_0 + \lambda \mathbb{M}_S)^{-1} \mathbb{W}_0 \mathbf{f}$. The SpLinT solution is $\mathbb{D}^\star_{Sp,1} = (\mathbb{C}_1 + \sigma^2 B_1 + \lambda \mathbb{M}_S)^{-1} \mathbb{W}_1 \mathbf{f}$.

With SpLinT decode weights and large pool sizes, a negligible increase in the number of parameters yields improved robustness and reduced approximation error than is achieved with LSAT weights. In addition, the sparse constraint has a regularizing effect which sometimes causes SpLinT test error to be lower than LinT test error (Fig. 10). Only a handful of LinT parameters are needed to reduce error. For an ensemble of 1024 neurons, only 23 LinT weights are needed to reduce test error by 18% compared to LSAT weights.

With SpLSAT decode weights, low error can be maintained while deactivating as many as 90% of the neurons on chip. For an ensemble of 1024 neurons, the LSAT test error can be matched even after deactivating nearly 900 neurons. By choosing an active set of 100 neurons from an ensemble of 1024 neurons, error can be reduced by a factor of 3.7 compared to a pool of 100 neurons with LSAT weights (see Fig. 10).

## VII. Conclusions

We have demonstrated for the first time that a mixed-signal neuromorphic chip can accurately and robustly decode functions across a wide temperature range (see Fig. 9). The PinT framework, which extends the Neural Engineering Framework to temperature-dependent decode weights, made this possible. We also developed the error operator, which informs us which functions an ensemble of neurons best approximates. Real on-chip performance matches its predictions (see Fig. 8).

In our analysis of PinT with Braindrop data, LinT achieved modest error reduction on test data compared to LSAT. On-chip, LinT did not achieve any error reduction compared to LSAT, possibly because LinT is more sensitive to numerical rounding when weights are stored with 8-bit precision. Higher-order polynomial weights did not yield any improvements beyond LinT's performance on test data. However, on training data, QuinT and TrinT each reduced error beyond what was achieved with LinT, suggesting that higher-order models were over-fitting to the training data. More data may help improve generalization. Finally, LSAT achieved lower error per number of parameters on test data compared to PinT.

Sparse models yield significant improvements to the energy-efficiency of thermally-robust function-approximation. In the SpLSAT model, up to 90% of neurons can be deactivated while maintaining the same performance. In the SpLinT model, a negligible increase in parameters yields significant error reduction compared to LSAT. This has significant ramifications for improving Braindrop's energy-efficiency. Further improvements may be possible with a hybrid of SpLSAT and SpLinT models, where SpLSAT first deactivates a subset of the neurons and then SpLinT equips a subset of the remaining active neurons with LinT weights.

We plan to extend our work to approximating temperature-dependent functions with PinT decode weights. Recurrent spiking neural networks can approximate arbitrary nonlinear dynamical systems by harnessing their low-pass synaptic filters' dynamics. Recent work has extended the underlying principle to synaptic filters that have dynamics beyond first-order and to mismatched time-constants [3]. However, no work to date has addressed the thermal variation of synaptic filters' time-constants in mixed-signal neuromorphics, which requires approximating temperature-dependent functions. PinT decode-weights may enhance approximation accuracy in this case. Hence, PinT decode weights could be useful in achieving thermally robust dynamics in mixed-signal neuromorphic systems.

## References

[1] C. Eliasmith and C. H. Anderson. *Neural engineering: Computation, representation, and dynamics in neurobiological systems.* MIT press, 2003.

[2] Kauderer-Abrams, E., Gilbert, A., Voelker, A., Benjamin, B., Stewart, T. C., and Boahen, K. "A population-level approach to temperature robustness in neuromorphic systems." *Circuits and Systems (ISCAS), 2017 IEEE International Symposium on.* IEEE, 2017.

[3] Voelker, A. R., Benjamin, B. V., Stewart, T. C., Boahen, K., and Eliasmith, C. "Extending the neural engineering framework for nonideal silicon synapses." *Circuits and Systems (ISCAS), 2017 IEEE International Symposium on.* IEEE, 2017.

[4] Neckar, Alexander, Stewart, Terrence C., Benjamin, Ben V., Boahen, Kwabena. "Optimizing an Analog Neuron Circuit Design for Nonlinear Function Approximation." *Circuits and Systems (ISCAS), 2018 IEEE International Symposium on.* IEEE, 2018

[5] Neckar, Alexander et. al. "Braindrop: A Mixed-Signal Neuromorphic Chip with a Dynamical Systems-Based Programming Model." *Proceedings of the IEEE* IEEE, January 2019.