Supplementary Information for:

pH-Controlled Enzymatic Computing for Digital Circuits and Neural Networks

Ahmed Agiza[a] Stephen Marriott[b] Jacob K. Rosenstein[c] Eunsuk Kim[b] Sherief Reda[c]

Preliminary: Gradient Descent Optimization

Gradient Descent has become a fundamental algorithm in the optimization of complex systems, notably within the context of machine learning. The algorithm operates by iteratively adjusting parameters in the direction of the steepest descent, or "gradient," of the function being optimized. Given its iterative nature, the algorithm has demonstrated utility in optimizing problems where the search space is high-dimensional and the objective function is non-convex[1,2]. In computational modeling, Gradient Descent has been applied to a range of applications, predominantly for neural network training[3]. However, gradient descent application is not confined to any single field, and it has made inroads into bioinformatics and systems biology for applications such as parameter estimation in complex biological models[4].

Enzyme Network for Computing Signal Concentrations

Computing the concentrations needed for the signals “0” or “1” is a main component of our system. We formulate an optimization problem modeled as a neural network, as depicted in Fig. 1. The architecture of this network comprises two weights: one for the concentration corresponding to a binary “0” and the other for the concentration corresponding to a binary ’1.’

The input to the neural network consists of the truth table of the logic function under design. During the forward pass, the network computes the resultant pH levels for each combination of input signals based on their corresponding concentrations, as modulated by the reactions in the buffers.

Since we want the network to learn the concentration that would cause the buffer to be in the pH that makes the enzymatic reaction move forward or backward, we define a ranged loss function, represented by Equation 1. y represents the network’s output, while a and b are the range in which we need the network’s output (i.e., the buffer’s pH) to lie, which is based on the pH range for the forward or reverse reaction to happen. This function is designed to incentivize the network to adjust its weights (signal concentration) such that the resultant pH falls within specified ranges according to the expected output (0 or 1). These ranges are dependent on the expected output from the logic function’s truth table. Specifically, the loss function pushes for a pH within the forward reaction range when the expected output is “1” and within the reverse reaction range when the expected output is ’0.’

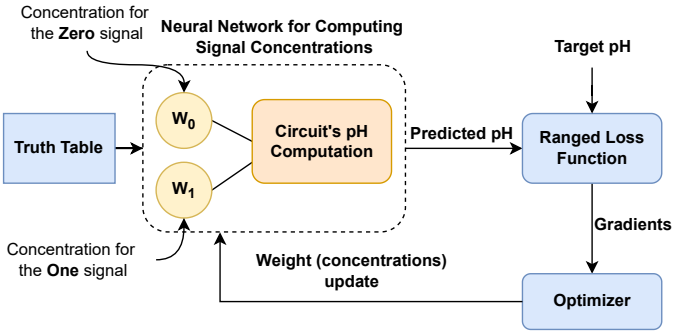


Fig. 1 Computing the concentrations for encoding the input signals using gradient descent. The concentrations are encoded as network weights. The pH computations capture the logic of the circuit’s design and evaluate the pH at the different buffers. The ranged loss function updates the concentrations untill it meets the target range.

$$\text{Range Loss}(y,(a,b)) = \begin{cases} 0 & \text{if } a \le y \le b, \\ |y-a| & \text{if } y < a, \\ |y-b| & \text{if } y > b. \end{cases} \quad (1)$$

The neural network’s training employs this specialized loss function, ensuring that the resultant pH for each logic condition aligns with the pre-specified ranges. Upon convergence, the network’s optimized weights yield the concentrations that should be employed for signals “0” and “1” in the enzymatic logic gates.

Signal Readout

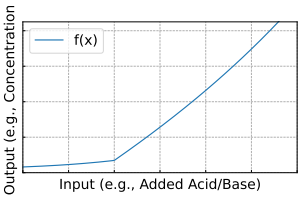
Detection of L-malic acid is usually done by applying a sequence of two catalyzed reactions to produce NADH.

$$\text{L}-\text{Malic acid} + \text{NAD}^+ \xrightleftharpoons{L-MDH} \text{oxaloacetate} + \text{NADH} + \text{H}^+ \quad (2)$$

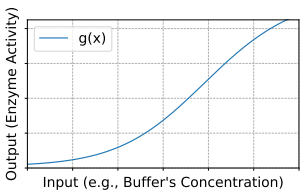
$$\text{Oxaloacetate} + \text{L}-\text{glutamate} \xrightarrow{\text{GOT}} \text{L}-\text{aspartate} + 2-\text{oxoglutarate} \quad (3)$$

The first reaction, outlined in Equation 2 is catalyzed by the enzyme L-malate dehydrogenase (L-MDH), whereby L-malic acid is oxidized to form oxaloacetate, with nicotinamide-adenine dinucleotide (NAD+) serving as the cofactor. The equilibrium of this reaction strongly favors the reactants L-malic acid and NAD+. To overcome this limitation and generate a detectable signal, a second reaction, outlined in Equation 3 is induced to trap

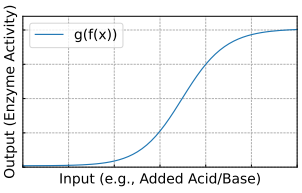
[a] Computer Science Department, Brown University, Providence, RI, USA. E-mail: ahmed_agiza@brown.edu
[b] Department of Chemistry, Brown University, Providence, RI, USA.
[c] School of Engineering, Brown University, Providence, RI, USA.



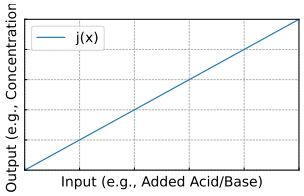
(a) Graph of the non-linear function $f(x)$, describing the behavior of chemical buffers in response to the addition of acid/base. The curve mimics a piecewise function corresponding to the behavior of the buffer within and outside the buffering range.



(b) Graph of the non-linear function $g(x)$, describing the enzyme activity at different concentrations, which increases as it approaches the optimal pH range.



(c) Graph of the non-linear composite function $(f \circ g)(x)$ describing the behavior of the enzyme while changing the buffer's concentration.



(d) Graph of the linear relation between input and output in linear computation systems, such as the change in the concentration of the system by varying the concentration of one of the substrates.

Fig. 2 Comparison between non-linear functions in our model (a) – (c), compared to the linear relation (d) induced by simpler computational models.

the resulting NADH product. This is accomplished by converting oxaloacetate to L-aspartate and 2-oxoglutarate, using an excess of L-glutamate as the substrate, in the presence of the enzyme glutamate-oxaloacetate transaminase (GOT). The amount of NADH generated corresponds to the original amount of L-malic acid present in the terminal buffer. The presence of NADH is subsequently quantified using its characteristic increase in absorbance at a wavelength of 340 nm. Measurement of this absorbance is executed with a UV spectrometer. An increase in absorbance at 340 nm indicates the presence of L-malic acid.

Visualizing Non-linear Components of the System

As we explained, our model introduces non-linearity through the chemical buffers that host the enzymatic reactions and the enzymatic reactions themselves.

If we model the buffer's change in pH in response to the concentration input of an acid or base through a non-linear function $f(x)$ and the enzyme's activity in response to the pH as a non-linear function $g(x)$. Thus, the system's overall response can be expressed as the composition of these two non-linear functions $(f \circ g)(x)$. This composition adds a layer of preferred non-linearity to our computational framework, allowing for further flexibility in the framework. For instance, Fig. 2a illustrates the non-linear characteristic of the function $f(x)$ where the curve captures the behavior of chemical buffers, represented by piecewise functions composed of linear segments to describe the buffer behavior inside and outside the buffer range. Fig. 2b depicts the non-linearity within $g(x)$, which captures enzyme activity at different pH levels, showing how enzyme activity increases and peaks towards the optimal pH. Meanwhile, Fig. 2c represents the composite non-linear function $(f \circ g)(x)$, which forms the foundational building block of our system, offering a more adaptable model in contrast to the conventional linearity, as indicated by Fig. 2d, found in some current unconventional computation methods[5], where the system's output is changed by linearly varying the concentration of one of the substrates. Mathematically, our framework can be seen as an approximation model for a given logic design's functionality, where the function $(f \circ g)(x)$ serves as the building block. The logic functionality's structure dictates the architecture and how these non-linear functions are composed together, which translates into the enzymatic reactions we carry. Meanwhile, the determination of the input signal concentrations is akin to parameter optimization in a model, aiming to refine the system's behavior to match the intended computational function closely.

Absorbance Curves for the Digital Gates and the Machine Learning Perceptron

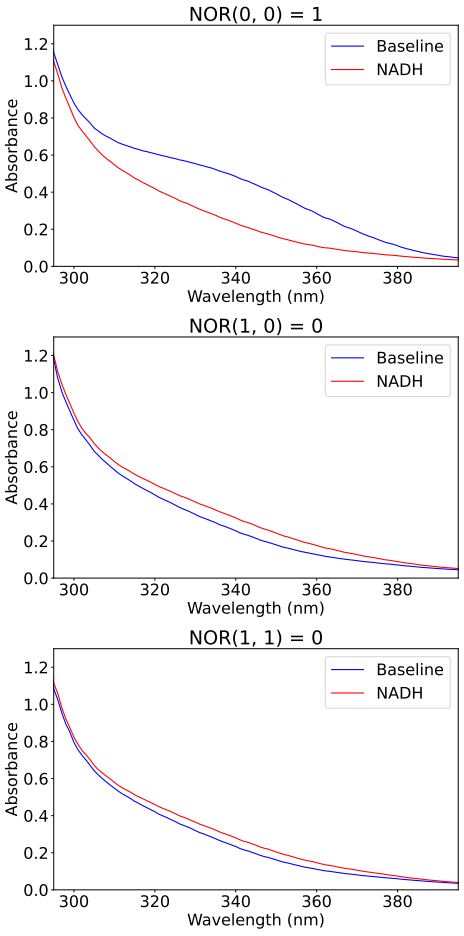


Fig. 3 UV-vis spectroscopy for the outputs of the NOR circuit modeled using enzymatic reactions.

We presented the quantitative output for our models of the digital gates and the machine learning perceptron. Fig. 3, 5, and 6 show the corresponding UV-vis absorbance curves.

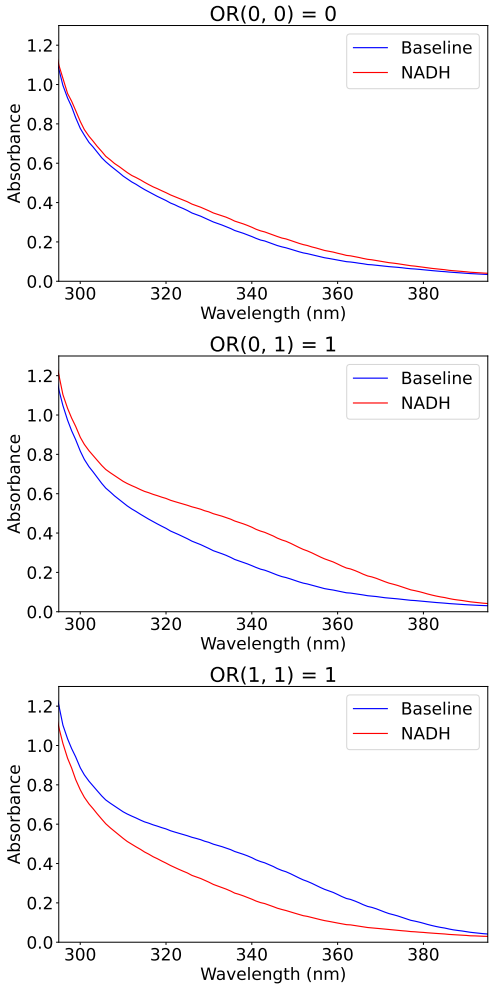


Fig. 4 UV-vis spectroscopy for the outputs of the OR circuit modeled using enzymatic reactions.

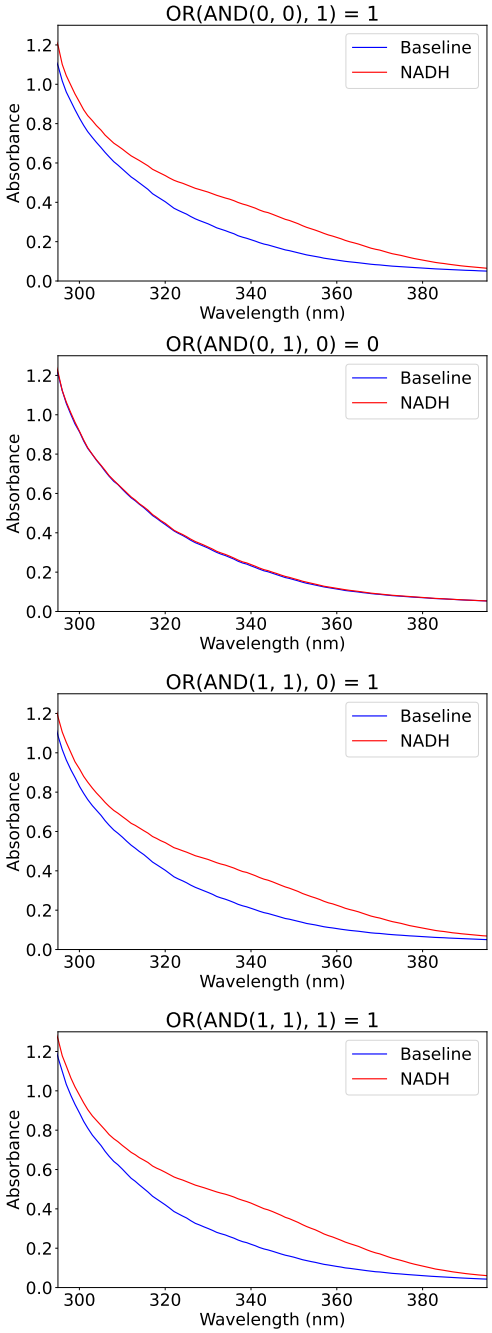


Fig. 5 UV-vis spectroscopy for the outputs of the AND-OR circuit modeled using enzymatic reactions.

Notes and references

1 S. Ruder, *arXiv preprint arXiv:1609.04747*, 2016.

2 D. P. Kingma and J. Ba, *arXiv preprint arXiv:1412.6980*, 2014.

3 Y. LeCun, Y. Bengio and G. Hinton, *nature*, 2015, **521**, 436–444.

4 C. G. Moles, P. Mendes and J. R. Banga, *Genome research*, 2003, **13**, 2467–2474.

5 A. A. Agiza, K. Oakley, J. K. Rosenstein, B. M. Rubenstein, E. Kim, M. Riedel and S. Reda, *Nature communications*, 2023, **14**, 496.

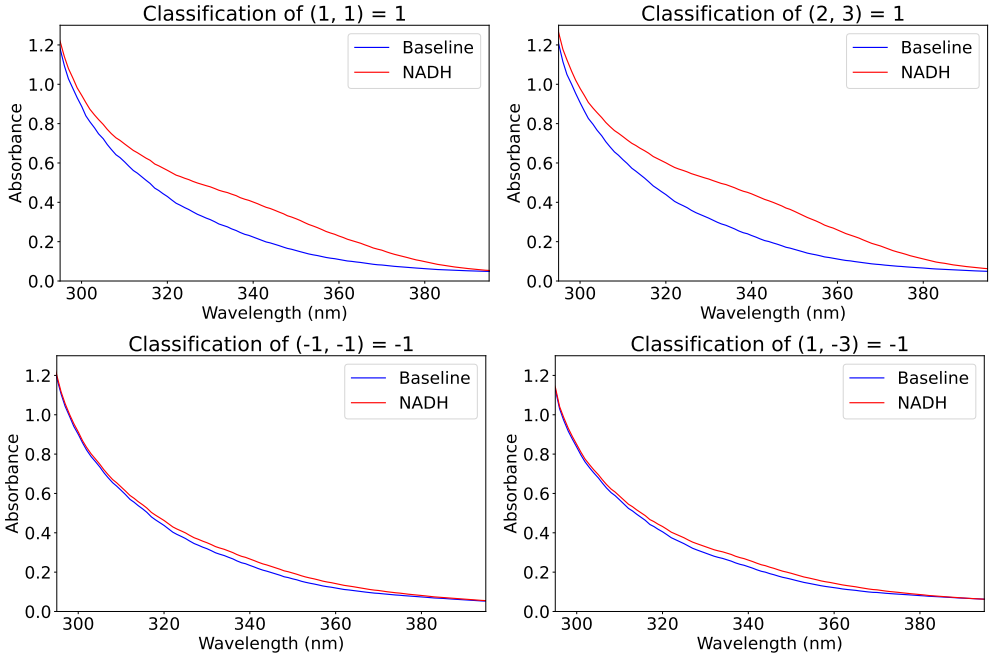


Fig. 6 Classification results for the machine learning perceptron modeled using enzymatic reactions.