Contrastive power-efficient physical learning in resistor networks

Menachem Stern[*]
Department of Physics and Astronomy
University of Pennsylvania

Sam Dillavou
Department of Physics and Astronomy
University of Pennsylvania

Dinesh Jayaraman
Department of Computer and Information Science
University of Pennsylvania

Douglas J. Durian
Department of Physics and Astronomy
University of Pennsylvania

Andrea J. Liu
Department of Physics and Astronomy
University of Pennsylvania

Abstract

The prospect of substantial reductions in the power consumption of AI is a major motivation for the development of neuromorphic hardware. Less attention has been given to the complementary research of power-efficient learning rules for such systems. Here we study self-learning physical systems trained by local learning rules based on contrastive learning. We show how the physical learning rule can be biased toward finding power-efficient solutions to learning problems, and demonstrate in simulations and laboratory experiments the emergence of a trade-off between power-efficiency and task performance.

1 Introduction

In recent years, the power consumption of training and inference using state-of-the-art ML models has risen exponentially, doubling every several months [1]. Such power requirements impede further ML development and pose significant sustainability problems [2, 3, 4]. This issue provides ample motivation for development of power-efficient systems capable of performing machine learning tasks. Neuromorphic hardware [5, 6, 7, 8] offers the prospect of lower power consumption compared to standard computers by $2-5$ orders of magnitude [9, 10, 11, 12]. In addition, power-efficient learning algorithms [13, 14, 15, 16] could further reduce power requirements in neuromorphic systems.

Recently, a new avenue was opened toward realizing power-efficient neuromorphic computing, *physical learning machines* or *self-learning physical networks* [17]. Rather than mimicking known learning algorithms such as backpropagation, such systems exploit their inherent physics in order to learn, using *local learning rules* that modify *learning degrees of freedom* based on locally available information, such that the system globally learns. In particular, *contrastive learning* [18, 19, 20, 21] rules enable supervised learning.

In order to realize any power gains, such learning rules must be implemented in hardware. *Coupled Learning*, a particular contrastive learning rule, has been realized successfully in laboratory hardware for electronic circuits of variable resistors [22, 23, 24, 25]. Such systems can be miniaturized and built to consume less power than conventional AI as they are analog rather than digital. Here we

[*]nachis@sas.upenn.edu

37th First Workshop on Machine Learning with New Compute Paradigms at NeurIPS 2023(MLNPCP 2023).

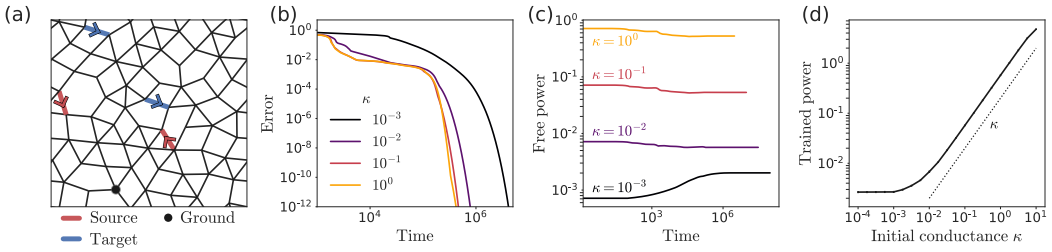

Figure 1: Simulations of physical circuits learning linear regression. (A) A network of size $N = 64$ nodes trained to infer linear relations between voltage drops on two input edges and two output edges. (B) Coupled learning succeeds in training this network regardless of conductance initialization, which is uniform. (C) However, the power dissipated by this network during training depends on the initialized conductance values. (D) Initializing conductances at low values produces more power-efficient learning solutions. The trained power is the free power of the system after training.

show that the propensity of self-learning electronic circuits to minimize power dissipation enables even greater reductions of power consumption via power-efficient learning rules.

In particular, we study how self-learning circuits can learn power-efficient solutions to computational tasks such as regression, by utilizing appropriate learning rules and initial conditions. We study these systems analytically, computationally and experimentally.

2 Power-efficient Coupled Learning

Consider an electronic circuit made of a network of edges consisting of adaptive resistors of conductance k_i, connected by nodes, indexed by a. The circuit physically equilibrates by minimizing the scalar *total power dissipation* $P = \sum_i k_i \Delta V_i^2$ subject to applied boundary conditions, where ΔV_i is the voltage drop across edge i. Thus, physics minimizes a "physical cost function" (P) by adjusting "physical degrees of freedom" (the node voltages V_a) to ensure that the current is balanced at every node (Kirchhoff's law). At the same time, a self-learning circuit adjusts its "learning degrees of freedom" (the edge conductances k_i) to minimize a "learning cost function," $\mathcal{C}$. In contrastive learning approaches such as Coupled Learning [20], the learning circuit locally compares two states that minimize the physical cost function, emerging in response to two sets of boundary conditions. First, voltage-drop inputs $\Delta V_{I,d}$ associated with training example d are applied as constraints to some input edges, and the network responds by finding a *free state* $\Delta V_{i,d}^F$ for all edges. Then, a second set of boundary conditions is applied, corresponding to a *clamped state*, chosen so as to encode an error signal.

For example, suppose we would like the circuit to respond by adopting certain output voltage drop values $\Delta \tilde{V}_{O,d}$ on output edges O, *i.e.* by minimizing a mean-squared-error cost function $\mathcal{C} \sim \sum_d [\Delta V_{O,d}^F - \Delta \tilde{V}_{O,d}]^2$. In Coupled Learning, a supervisor applies additional weak constraints to output nodes to nudge them closer to the desired voltage response $\Delta V_{O,d}^C = \Delta V_{O,d}^F + \eta[\Delta \tilde{V}_{O,d} - \Delta V_{O,d}^F]$, with $\eta \ll 1$, and the system finds a *clamped state* V_a^C. The contrast function $C_d \equiv \eta^{-1}\{P_d^C - P_d^F\}$ serves as a surrogate for the learning cost function $\mathcal{C}$, so that the partial derivative of C with respect to the conductances k_i leads to a local learning rule:

$$\dot{k}_{i,d} = \alpha\eta^{-1}\frac{\partial}{\partial k_i}\{P_d^F - P_d^C\} = \alpha\eta^{-1}\{(\Delta V_{i,d}^F)^2 - (\Delta V_{i,d}^C)^2\}. \tag{1}$$

This local rule is implemented physically in our laboratory self-learning circuits [25] to update the learning degrees of freedom, in lieu of a more conventional approach such as backpropagation. We emphasize that that no computer is involved with setting the values of the learning degrees of freedom, i.e. the conductance values.

We first apply this learning rule in computational simulations to train ($N = 64$ nodes, $N_e = 143$ edges, code available [26]) networks of variable linear resistors to perform linear regression, *i.e.*

to recover a linear relation between the voltage drops on two source edges and desired voltage drops at two target edges $\Delta V_O = \sum_i \tilde{A}_{OI} \Delta V_I$ (Fig. 1A), with $\tilde{A}_{OI}$ a 2×2 matrix sampled from $\tilde{A} \sim \begin{pmatrix} 0.2 & 0.3 \\ 0.1 & 0.5 \end{pmatrix} + 0.1\mathcal{N}(0,1)^{2\times 2}$. Networks are initialized with uniform conductance values at different scales $10^{-4} \leq \kappa \leq 10^{1}$, where the extreme values correspond to the conductance limits allowed in the simulation (conductances are clipped at these values, simulating the limited range of realistic adaptive resistors). Our simulated networks are successful in learning these tasks, reducing the error by orders of magnitude regardless of conductance initialization (Fig. 1B).

2.1 Initialization

The free state $V^F_{a,d}$ is associated with power dissipation P^F_d by Kirchhoff's laws:

$$P^F_d = \sum_i k_i (\Delta V^F_{i,d})^2. \tag{2}$$

This free state power is the power consumption of the circuit during inference, the process that typically requires the majority of energy in AI. We would like the learning machine to find solutions to the learning task that minimize the cost function $\mathcal{C}$ while keeping the free power dissipation P^F relatively low. Eq. 2 suggests that lower conductances k_i produce lower overall power dissipation. Indeed, we find that initializing the network with lower conductance values, the network learns solutions to the learning problem that dissipate less power (Fig. 1C), with no cost in error but some cost in training time. This trend of linear scaling of the free power with conductance initialization continues until the conductances are initialized close to their allowed minimal values; at that point the power saturates at a low value (Fig. 1D).

2.2 Modified learning rule

Another way to encourage the circuit to learn more power-efficient solutions is to utilize a local learning rule that minimizes both the error and free power at the same time. For example, we can introduce a new cost function $\mathcal{C}_\lambda = \mathcal{C} + \lambda P^F$, with $\lambda \geq 0$ the *power optimization amplitude*, a hyper-parameter that sets the relative importance of error and power minimization. Crucially, by substituting the surrogate contrastive function for $\mathcal{C}$, this form admits a local, physically-realizable learning rule similar to Eq. 1:

$$\dot{k}^{(\lambda)}_{i,d} = -\alpha \frac{\partial}{\partial k_i} C_\lambda = \alpha \eta^{-1} \{(1-\lambda)(\Delta V^F_{i,d})^2 - (\Delta V^C_{i,d})^2\}. \tag{3}$$

For over-parameterized networks, the network finds learning solutions with no error $\mathcal{C}_0(k_i) = 0$, and a quadratic error landscapes forms around these solutions. In such cases we can analyze this learning rule analytically to find how the error and free power scale as we introduce this power optimization λ. We find that the error scales quadratically, $\mathcal{C}_\lambda \sim \lambda^2$, and the power is reduced by a linear amount $P^F_0 - P^F_\lambda \sim \lambda$, as seen in simulations for linear regression in resistor networks (Fig. 2A).

As we increase the power optimization parameter λ beyond the infinitesimal regime, we see a continuation of these trends (Fig. 2B): the free power found for the learning solution is decreased considerably while the error keeps increasing. These trends underscore an emergent trade-off between performance and power-efficiency of learned solutions established by this kind of a physical learning rule. This is clearly seen in the Pareto front in Fig. 2C. Depending on the desired level of performance (error) the operator of the learning circuit can choose the power optimization level λ to increase the power efficiency associated with using this machine for inference. We can also measure the total energy used in learning by the circuit by integrating over the free state power during the training process, until the error reaches its minimal value. We find that increasing the power optimization parameter λ reduces this training energy (Fig. 2B), improving the power efficiency of the learning process as well as the inference.

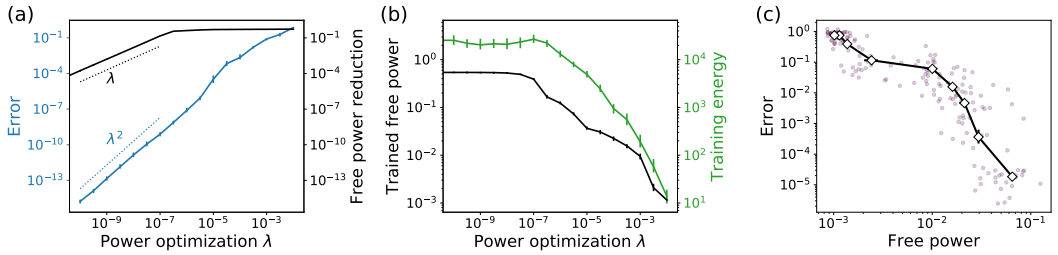

Figure 2: Emergent trade-off between error and power-efficiency in physical learning circuits. (A) Simulation results for self-learning circuits trained with Eq. 3 to perform linear regression with varying power optimization parameter λ. We observe that for low λ, linear improvement in power-efficiency for inference and a quadratically scaling error, consistent with analytical predictions. (B) For larger λ, the power dissipation of the trained network is reduced substantially, as well as the overall energy expended by the circuit to learn this solution (training energy). (C) These results reveal a trade-off between performance and power efficiency, evident in a Pareto front. A trainer can choose the hyper-parameter λ to adjust the power consumption and performance of the learned solutions.

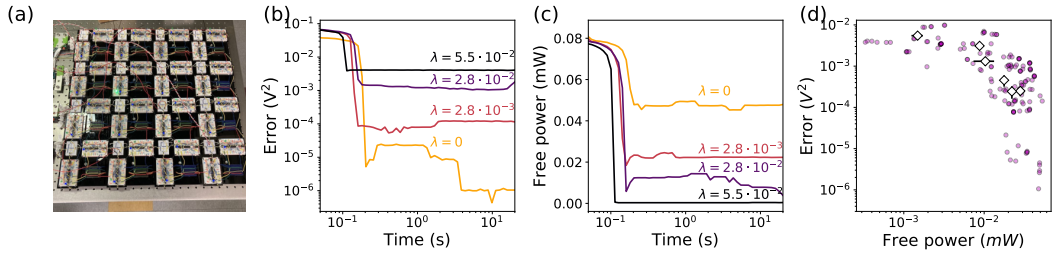

Figure 3: Power-efficient learning rules realized in a laboratory implementation of a physical learning circuit. (A) Photo of the experimental learning circuit, showing the network structure and self-learning elements [25]. (B) The circuit learns on its own to perform input-to-output mapping tasks, reducing the error to a floor set by experimental noise as well as the power optimization parameter λ. Increasing λ admits solutions with worse performance (higher error). (C) At the same time, increased power optimization helps the circuit learn more power-efficient solutions. (D) A trade-off emerges between the final trained error and power-efficiency, qualitatively similar to what is predicted analytically and computationally.

3 Physical experiments

We next test the modified learning rule in laboratory experiments. We use an experimental network of variable resistors implementing Coupled Learning, similar to realizations in previous works [22, 23, 24]. However, in this new implementation of the experiment, transistors replace the digital potentiometers in the role of variable resistors [25]. This circuit implements the continuous Coupled Learning rule (Eq. 1) for analog learning, as each resistance element is set by the charge of a capacitor on the gate of the transistor. Modifications to the learning rule of the form of Eq. 3 are achieved by varying the measurement amplification from the free and clamped networks. Unlike previous implementations, this new learning circuit operates continuously in time, with the clamped state value updated automatically via an electronic feedback loop. Training length is thus measured in real time rather than training steps. Because of unavoidable noise in the experiment, $\eta \ll 1$ is unobtainable; as the clamped state approaches the free state their difference becomes more and more difficult to measure. We therefore use a finite value $\eta = 0.22$ for these experiments, with an effective learning rate of $\alpha = \frac{1}{24ms}$. Experiments lasted 20 seconds each, and the conductance values completely settled at the end of each trial. The network architecture is a 4x4 square lattice of edges (Fig. 3A) with periodic boundary conditions, and edges are initialized with uniform conductance approximately in the middle of their range at the start of each experiment.

The network was trained for 150 two-source, two-target simple tasks, wherein the sources were held at the low and high end of the allowable range ($0V$ and approximately $0.45V$, respectively), with the

two desired target outputs at either 20 and 80% or at 10 and 90% of this range, respectively. Across these experiments, λ was varied with values ranging $0 - 0.055$. In all cases the network was able to lower the error, as shown for typical error vs training time curves in Fig. 3B. For these tasks, the network also consistently lowered the power of its free state, as shown for the complementary power curves over training time in Fig. 3C. Consistent with theoretical predictions, error and power increased and decreased (respectively) with increasing λ, with their trade-off shown in Fig. 3D. White diamonds correspond to the mean error and free power of all experiments performed with the same value of λ. These results agree qualitatively with trade-off of learning performance and power efficiency studied in simulations in the previous section.

4 Concluding remarks

Finding power-efficient ways to construct and train learning systems is an important challenge that must be overcome to maintain the rapid development in machine learning applications in a warming world. While most efforts in neuromorphic computing focus on power-efficient learning hardware, we emphasize that additional significant energy savings can be achieved via learning algorithms and initialization. Such learning may lower not only the energy necessary for inference, but the energy required to train the system for its task.

Our results show that physical learning machines, in particular learning adaptive circuits, can be encouraged to find power-efficient solutions to computational learning tasks like regression. One way to achieve this is by judicious initialization of the circuit conductances. Less obvious is the idea that modified learning rules, still permitting locality and physical realizability, can optimize error and power consumption simultaneously. Using such power optimization as a hyper-parameter enables the trainer to choose whether to emphasize lower power or lower error according to their needs. While we show power-efficient learning is practical and realizable for learning circuits and regression tasks, it is likely also viable for other implementations of learning machines and computational tasks (*e.g.* classification). The benefit of power-efficient learning in saving power for such cases is an important subject to future study.

Broader Impact

While computational machine learning is extremely useful and widely applicable, large scale deployment of artificial intelligence raises important sustainability issues. The development of power-efficient learning systems is a major motivation for neuromorphic hardware mimicking power-efficient media like our brain. In this work, we go beyond power-efficient learning hardware to study power-efficient learning algorithms that can be implemented in such hardware, allowing even further reduction in power consumption. We study such algorithms in the context of physical self-learning machines, a new type of neuromorphic hardware that has recently been realized in the lab. We show that power-efficient learning rules are indeed physically realizable and effective, and discuss the emergence of a trade-off between power efficiency and performance.

Acknowledgments and Disclosure of Funding

This research was supported by the U.S. Department of Energy, Office of Basic Energy Sciences, Division of Materials Sciences and Engineering award DE-SC0020963 (M.S.), the National Science Foundation via the UPenn MRSEC/DMR-1720530 and MRSEC/DMR-DMR-2309043 (S.D. and D.J.D.), and DMR-2005749 (A.J.L.), and the Simons Foundation (# 327939 to A.J.L.). S.D. acknowledges support from the University of Pennsylvania School of Arts and Sciences Data Driven Discovery Initiative. D.J.D. and A.J.L. thank CCB at the Flatiron Institute, as well as the Isaac Newton Institute for Mathematical Sciences under the program “New Statistical Physics in Living Matter" (EPSRC grant EP/R014601/1), for support and hospitality while a portion of this research was carried out.

References

[1] Dario Amodei, Danny Hernandez, Girish Sastry, Jack Clark, Greg Brockman, and Ilya Sutskever. Ai and compute. `https://openai.com/research/ai-and-compute`, 2018. Accessed:

2023-06-16.

[2] Aimee Van Wynsberghe. Sustainable ai: Ai for sustainability and the sustainability of ai. *AI and Ethics*, 1(3):213–218, 2021.

[3] Udit Gupta, Young Geun Kim, Sylvia Lee, Jordan Tse, Hsien-Hsin S Lee, Gu-Yeon Wei, David Brooks, and Carole-Jean Wu. Chasing carbon: The elusive environmental footprint of computing. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, pages 854–867. IEEE, 2021.

[4] Carole-Jean Wu, Ramya Raghavendra, Udit Gupta, Bilge Acun, Newsha Ardalani, Kiwan Maeng, Gloria Chang, Fiona Aga, Jinshi Huang, Charles Bai, et al. Sustainable ai: Environmental implications, challenges and opportunities. *Proceedings of Machine Learning and Systems*, 4:795–813, 2022.

[5] Carver Mead. Neuromorphic electronic systems. *Proceedings of the IEEE*, 78(10):1629–1636, 1990.

[6] Geoffrey W Burr, Robert M Shelby, Abu Sebastian, Sangbum Kim, Seyoung Kim, Severin Sidler, Kumar Virwani, Masatoshi Ishii, Pritish Narayanan, Alessandro Fumarola, et al. Neuromorphic computing using non-volatile memory. *Advances in Physics: X*, 2(1):89–124, 2017.

[7] Danijela Marković, Alice Mizrahi, Damien Querlioz, and Julie Grollier. Physics for neuromorphic computing. *Nature Reviews Physics*, 2(9):499–510, 2020.

[8] Catherine D Schuman, Shruti R Kulkarni, Maryam Parsa, J Parker Mitchell, Prasanna Date, and Bill Kay. Opportunities for neuromorphic computing algorithms and applications. *Nature Computational Science*, 2(1):10–19, 2022.

[9] Mrigank Sharad, Charles Augustine, Georgios Panagopoulos, and Kaushik Roy. Proposal for neuromorphic hardware using spin devices. *arXiv preprint arXiv:1206.3227*, 2012.

[10] Ivan K Schuller, Rick Stevens, Robinson Pino, and Michael Pechan. Neuromorphic computing–from materials research to systems architecture roundtable. Technical report, USDOE Office of Science (SC)(United States), 2015.

[11] Minseon Kang, Yongseok Lee, and Moonju Park. Energy efficiency of machine learning in embedded systems using neuromorphic hardware. *Electronics*, 9(7):1069, 2020.

[12] Mike Davies, Andreas Wild, Garrick Orchard, Yulia Sandamirskaya, Gabriel A Fonseca Guerra, Prasad Joshi, Philipp Plank, and Sumedh R Risbud. Advancing neuromorphic computing with loihi: A survey of results and outlook. *Proceedings of the IEEE*, 109(5):911–934, 2021.

[13] Emre O Neftci. Data and power efficient intelligence with neuromorphic learning machines. *Iscience*, 5:52–68, 2018.

[14] Martino Sorbaro, Qian Liu, Massimo Bortone, and Sadique Sheik. Optimizing the energy consumption of spiking neural networks for neuromorphic applications. *Frontiers in neuroscience*, 14:662, 2020.

[15] I Chakraborty, A Jaiswal, AK Saha, SK Gupta, and K Roy. Pathways to efficient neuromorphic computing with non-volatile memory technologies. *Applied Physics Reviews*, 7(2):021308, 2020.

[16] Simon Narduzzi, Siavash A Bigdeli, Shih-Chii Liu, and L Andrea Dunbar. Optimizing the consumption of spiking neural networks with activity regularization. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 61–65. IEEE, 2022.

[17] Menachem Stern and Arvind Murugan. Learning without neurons in physical systems. *Annual Review of Condensed Matter Physics*, 14(1):417–441, 2023.

[18] Javier R Movellan. Contrastive hebbian learning in the continuous hopfield model. In *Connectionist models*, pages 10–17. Elsevier, 1991.

[19] Benjamin Scellier and Yoshua Bengio. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in computational neuroscience*, 11:24, 2017.

[20] Menachem Stern, Daniel Hexner, Jason W Rocks, and Andrea J Liu. Supervised learning in physical networks: From machine learning to learning machines. *Physical Review X*, 11(2):021045, 2021.

[21] Vidyesh Rao Anisetti, Benjamin Scellier, and Jennifer M Schwarz. Learning by non-interfering feedback chemical signaling in physical networks. *Physical Review Research*, 5(2):023024, 2023.

[22] Sam Dillavou, Menachem Stern, Andrea J Liu, and Douglas J Durian. Demonstration of decentralized physics-driven learning. *Physical Review Applied*, 18(1):014040, 2022.

[23] Jacob F. Wycoff, Sam Dillavou, Menachem Stern, Andrea J. Liu, and Douglas J. Durian. Desynchronous learning in a physics-driven learning network. *The Journal of Chemical Physics*, 156(14):144903, 2022.

[24] Menachem Stern, Sam Dillavou, Marc Z Miskin, Douglas J Durian, and Andrea J Liu. Physical learning beyond the quasistatic limit. *Physical Review Research*, 4(2):L022037, 2022.

[25] Sam Dillavou, Benjamin Beyer, Menachem Stern, Marc Z Miskin, Andrea J Liu, and Douglas J Durian. Circuits that train themselves: decentralized, physics-driven learning. In *AI and Optical Data Sciences IV*, volume 12438, pages 115–117. SPIE, 2023.

[26] Menachem Stern. Contrastive power-efficient physical learning in resistor networks. `https://github.com/NachiStern/Power-efficient-learning`, 2023.