

A perspective on physical systems with a purpose: using tunable degrees of freedom to enable function

M. Lisa Manning¹

¹*Department of Physics and BioInspired Institute, Syracuse University, Syracuse NY 13244 USA*

(Dated: May 2026)

This perspective highlights an emerging framework for understanding evolved biological and designed physical systems as tunable materials. In analogy to supervised learning in artificial neural networks – where weights between nodes are adjusted to optimize a cost function during the training phase – recent work postulates that biological systems may be using similar tuning rules on their adjustable degrees of freedom to learn or perform tasks. Researchers are also using these algorithms to design synthetic physical systems that exhibit exotic mechanical responses or perform tasks like classification and regression. Here, we briefly review the algorithms that enable such behavior and highlight examples in both physical and biological systems, emphasizing major open questions in this nascent field. Our goal is to provide SPLASHY researchers with a jumping off point for considering whether their own systems of interest might be a platform for tunable matter, and spur future work to address open questions.

I. INTRODUCTION

Biological organisms have a purpose; they are optimized via evolution and interactions with their environment to enhance fitness. Often, this involves executing complex tasks. We would also like to be able to design bio-mimetic materials and physical structures that are optimized to achieve specified outcomes.

Historically, integrating the idea of ‘having a purpose’ has been antithetical to traditional condensed matter and physics approaches. On the other hand, researchers in the fields of control theory and machine learning in artificial neural networks explicitly identify desired outputs and develop prescriptions for design to optimize outputs. They also explicitly identify the degrees of freedom that can be tuned – e.g. inputs in control theory or weights in the network during training in supervised learning – which ultimately enable the system to execute complex tasks.

In the simple atomic or molecular systems typically considered by statistical and condensed matter physics, there are not obvious tunable parameters. E.g., in a bulk metallic glass or one-component polymer, molecules always interact with each other the same way. But in more complex, hierarchical systems – such as mixtures of biomolecules or polymers, biological tissues, or mechanical metamaterials – the coarse-grained “components” (edges in a 3D printed network or origami structure, cells in a tissue) are themselves mesoscopic objects with non-trivial geometries and compositions. Therefore, it is possible to tune individual properties of these components, much like the weights in an artificial neural network.

This opens up a new opportunity to imbue materials with function, or better understand how function emerges in biology. This is a key motivating idea for the emerging framework of ‘physical learning’ [1, 2]. One immediate difficulty in this emerging area is that ‘learning’ is a freighted word; different researchers use it to mean different things. Some researchers are interested in

developing systems that learn similar to the human brain and can perform tasks like generalization out of context. Other researchers want to design Turing machines that can generally compute, while still others are interested in less complex tasks, such as adaptation or even just on-the-fly optimization.

Excellent comprehensive reviews of physical learning and related algorithms already exist [1, 2]; this short perspective describes a few key recent developments that are relevant for SPLASHY physicists – algorithms for tuning, examples of tuning in designed physical and evolved biological systems – and highlights some important open questions and future directions.

II. BACKGROUND AND ALGORITHMS FOR TUNING IN PHYSICAL SYSTEMS

In supervised learning, training a layered neural network to perform a task consists of giving it a set of training data $\{x_i, y_i\}$ – inputs and outputs – and adjusting all the interactions between edges in the neural network (individually tunable weights) to minimize a cost or loss function,

$$\mathcal{C} = (\text{actual } y - \text{predicted } y \text{ from } x\text{'s})^2, \quad (1)$$

typically implemented using stochastic gradient descent.

For a mechanical or physical system, one could imagine implementing the same algorithm, replacing the weights in the artificial neural network with the tunable degrees of freedom. One can define a cost function

$$\mathcal{C} = (\text{desired state} - \text{predicted state})^2, \quad (2)$$

which penalizes both positive and negative deviations from the desired state. In some cases, researchers replace this cost function with simpler (e.g. linear) proxies that are more efficient to compute or easier to embed in physical systems.

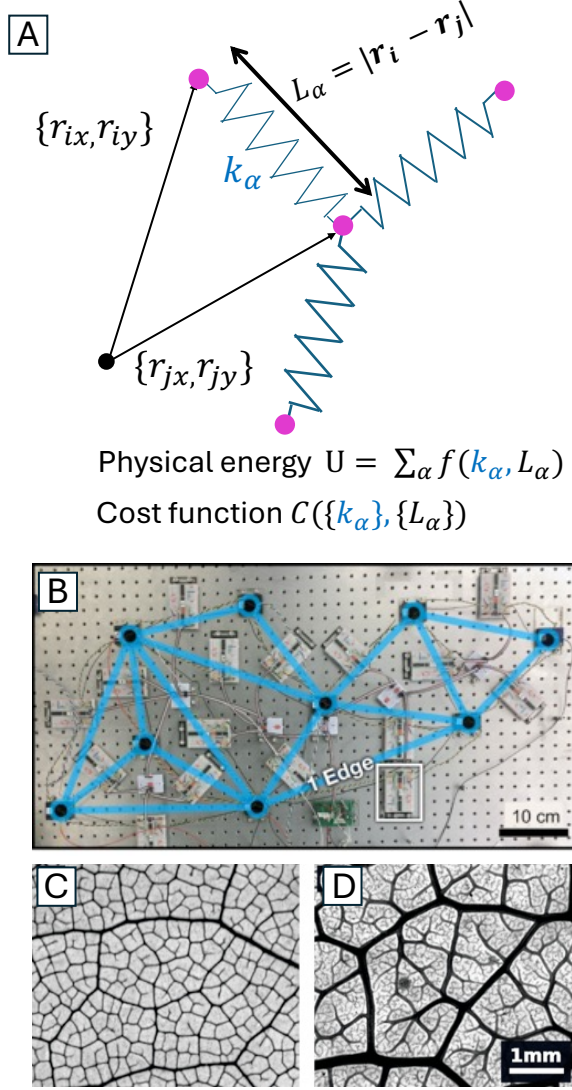


FIG. 1. A) Schematic of a set of nodes in a larger mechanical network. (B) Example of a physical resistor network that uses tunable resistors and a coupled learning algorithm to perform image recognition tasks. Used with permission from Ref. [3]. (C,D) Leaf veins of *Acer platanoides* (C) and *Protium dawsonei* (D) show vein patterns that are either reticulated or tree-like. Theoretical work suggests the degree of reticulation is controlled by a key feature of the tuning rules that adaptively adjust the network. Used with permission from Ref. [4]

The cost function $\mathcal{C}(\{k_\alpha\}, \{r_{i\mu}\})$ depends on tunable degrees of freedom (TDOFs) $\{k_\alpha\}$ which can be controlled/manipulated, e.g. spring stiffnesses or pipe conductances, and physical degrees of freedom (PDOFs) $\{r_{i\mu}\}$, which correspond to the physical configuration of the system, e.g. locations of vertices in a network or currents in a pipe (Fig 1A). Then an update rule could be implemented as gradient descent on the cost function in

the space of the tunable degrees of freedom:

$$k_\alpha^t = k_\alpha^{t-1} - \lambda \frac{\partial \mathcal{C}(\{k_\beta\}, \{r_{i\mu}\})}{\partial k_\alpha} \Big|_{t-1}, \quad (3)$$

where t counts the number of update iterations and λ is a learning rate.

Because the system is physical, it generally minimizes a (free) energy U that is also defined by the TDOFs and PDOFs (Fig 1A). With an external processor, access to all the degrees of freedom ($\{k_\alpha\}, \{r_{i\mu}\}$), and precise control on all the TDOFs $\{k_\alpha\}$, one could simply calculate this derivative and update the TDOFs, and the resulting system would optimize the cost function. Of course, in real physical or biological systems there may not be such an external processor and so it makes sense to consider whether it is possible to compute updates with only local information restricted to features of a node and its neighbors.

Neurons in the brain have inspired algorithms to imbue physical systems with the ability to execute tasks, since neurons are physical objects that collectively learn. Physicists are perhaps most familiar with the celebrated Hopfield model [5], which implements a version of Hebbian learning, e.g. ‘cells that fire together wire together’, where synaptic weights are calculated from the sum of the outer products of the training data. This allows patterns that are similar to be stored as stable minima of the energy functional, which facilitates recall from noisy input data.

The Hebbian rule is a simple example of a tuning rule that has been shown to facilitate learning certain types of tasks, but one would like to guarantee that an update rule “works”, e.g. that robustly allows the system to minimize a desired cost function.

An important algorithm for physical systems with tuning degrees of freedom is called contrastive learning [6–8]. Contrastive learning responds to differences between a free state, where the system is simply exposed to training inputs, and a “clamped state”, where the system is constrained to generate a response to each training input that is closer to the desired output state.

In a physical system, one can train by updating the TDOFs to make the free state more like the clamped state:

$$\Delta k_\alpha = \frac{\partial U^{\text{free}}}{\partial k_\alpha} - \frac{\partial U^{\text{clamped}}}{\partial k_\alpha}. \quad (4)$$

For a mechanical network, the partial derivatives are local because the physical energy at a site i depends only on the neighboring nodes connected by interactions α defined by the network connectivity graph g_{ij} . A challenge for this method is that sometimes the clamped state is far from the free state, so there is no guarantee that it is in the same minima of the landscape, and therefore the algorithm may not converge.

About a decade ago, Scellier and Bengio [9] developed an equilibrium propagation framework examining

differences between a free and an infinitesimally nudged “weakly clamped” state. The weakly clamped state is defined by a new energy functional of the form

$$\mathcal{F} = U - \beta\mathcal{C} \quad (5)$$

One can prove that this quite generally generates a local rule that encodes stochastic gradient descent of $\mathcal{C}$. This is what we were looking for – a precise formulation of a local rule to encode a learning cost function. Of course, to approximate this algorithm in real physical systems such as mechanical or flow networks, one must apply finite-sized nudges. This is the approach taken in coupled learning [10], where the nudges are applied to selected output nodes.

Another challenge to implementing contrastive learning algorithms in experimental networks is that there are different signs in front of the derivative for the free and (weakly) clamped states – one wants to nudge the TDOFS towards the clamped state (a Hebbian-like, reinforcement rule) and nudge them away from the free state (which is an anti-Hebbian, anti-reinforcement rule). Moreover, the system has to remember those two results and compare them. In general, it is difficult to both change the sign and remember the two results in a physical system without an external processor.

More recent algorithms, such as multi-mechanism learning [11], take advantage of different channels (such as AC/DC current) on the same physical network to enable learning, while others allow the system to execute dynamic/temporally varying tasks [12–14].

In addition, several authors have shown that systems can be trained to execute tasks even if the learning rule does not precisely follow the gradient of the cost function; approximate learning rules that only implement Hebbian-like reinforcement rule and ignore the anti-Hebbian component of contrastive learning are still effective [10].

III. EXAMPLES OF TUNING AND LEARNING IN SYNTHETIC PHYSICAL SYSTEMS

Explorations in developing experimental physical systems that learn have been rapidly gaining momentum over the last few years [15]. Here we highlight a handful of recent examples to help readers identify similarities with their own systems of interest.

Some physical systems have tuning rules that are designed via an external processor (human or computer). Wright and collaborators [16] used an external processor to model the physics of different systems – a vibrating plate, an electronic resistor network, and an optical network – in order to help train the physical systems to perform digit recognition. Evans and collaborators [17] developed sets of DNA origami tiles that were designed to nucleate into one of three mesoscopic shapes. By varying the input concentration of different tiles, the authors could control the output shape, and using an external

processor they were able to design a mapping from image pixels to tile concentrations that allowed the system to perform image or digit classification.

Other researchers have used similar ideas to design materials with specific mechanical properties. For example, Hagh and collaborators [18] used an external processor to alter TDOFS in a simulated disordered particle packing – namely the particle radii – in order to minimize a cost function that corresponds to the mechanical energy of the packing. This generates a highly dense packing with many more contacts per particle compared to standard packings, resulting in an ultra-stable material with exotic properties that are useful for understanding glass physics [19]. Altman et al. [20] studied a mechanical network with tunable rest lengths, implemented via spring-turnbuckle links. They were able to implement a coupled learning scheme that successfully accomplished motion-divider and allosteric tasks. Although the coupled learning scheme itself is local, implementing the rule required an external processor (a human intervener).

For scalability and flexibility, there is immense interest in developing physical systems that can learn or optimize themselves *without* an external processor. We expect that algorithms that operate without an external processor should be *local*, so that each tunable element only requires information about itself and perhaps nearby neighbors to compute and apply updates to its own state. This does not necessarily preclude global information inputs, such as biological morphogen gradients, that can be sensed locally by individual TDOFs.

Foundational work on this topic was developed by Pashine and collaborators [21], who identified an algorithm for directed aging in mechanical networks. In this work, researchers apply ‘clamped’ conditions that require an external applied force, and allow the material to age under this clamped condition. The key idea is that links in the mechanical network change their properties over time under stress – links that experience more stress in the clamped condition will extend their rest lengths during clamping, thus altering the mechanics of the network. By choosing the boundary conditions cleverly, the researchers were able to create mechanical networks that exhibit auxetic behavior – when compressed along one axis the material also compresses along the orthogonal axis.

In other words, the system executed a local rule that altered TDOFs in order to generate a material with a desired mechanical response, with no external processor required. Although the local rule is not analogous to contrastive learning (it only has the Hebbian, reinforcing component), and therefore does not directly follow gradient descent on a cost function corresponding to auxetic mechanical response, simulations and theory indicate that the rule still robustly attains the desired result [22].

To ensure convergence to the desired result, and perhaps allow more complex tasks, one would like to precisely implement contrastive learning in physical systems, despite the challenge of comparing Hebbian and anti-

Hebbian tuning. One implementation of coupled learning [3] cleverly solves this problem by physically attaching two copies of the same network together, one with a free boundary condition and one with a clamped boundary condition (Fig 1B).

Researchers are currently working to scale-up the design of such circuits [23], with the idea that these systems could help solve the “compute crisis” generated by increasing power requirements for standard AI computations [24]. More generally, an important and relatively mature application of tunable matter is in neuromorphic computing [2], which develops brain-inspired physical hardware and algorithms to enable more efficient artificial intelligence applications [25]. In addition to resistor networks, other promising physical substrates include photonic [26], quantum, and light-matter interaction systems [27]. Key open challenges in this field of “unconventional compute” include determining how to scale the systems to large sizes [25] and identifying applications, like in situ sensing, where neuromorphic computing has the potential to outperform traditional silicon chips [2].

IV. EXAMPLES OF TUNING AND LEARNING IN BIOLOGICAL SYSTEMS

A general tuning framework might also drive new hypotheses and bring new insight to biological systems. After all, at all scales biological organisms perform amazing tasks, entirely via physical processing. There is already an enormous amount of work focused on how neurons and brains compute using physical interactions [28, 29], so this perspective will focus instead on a few recent examples of learning that do not involve neurons.

Perhaps the next most obvious example of learning in a biological system is the adaptive immune system [30, 31], which exhibits many different types of tuning rules. For example, tuning rules that alter the selection probabilities of B cells based on past pathogen exposures can help the system generalize and mount a response to unseen pathogen variants [32]. Tuning rules that dynamically modulate the selection of T cells in the thymus and other tissues to help distinguish self from non-self help the system perform more accurate classification in a changing environment [31, 33, 34].

In another set of examples, molecular signaling networks inside cells can drive adaptation and learning. Beautiful recent experimental work on the single-cell organism *Stentor coeruleus* [35] identifies the molecular feedback loop that allows it to reduce its sensitivity to repeated mechanical tapping. In this case, the tuning rule involves inactivation of cell-surface receptors, accomplished via modulation of protein synthesis and ultimately modification of the receptors via phosphorylation and calcium signaling. In theoretical work, Trifonova et al. [36] demonstrate that molecular networks can be trained to perform tasks like Pavlovian response and even simple image classification, with a simple, bio-

logically plausible tuning rule, namely that each chemical species must autoregulate based on rapid changes in its own monomer levels.

Several other works explore biological analogues of the mechanical systems discussed in Section III. A recent paper by Banerjee et al [37] describes how mechanosensitive elements in cytoskeletal networks could allow the system to implement a contrastive learning scheme that would allow it to respond adaptively to environmental perturbations, performing allosteric, linear regression, and classification tasks. Similarly, work by Wang et al [38] on a model for dynamic actin network remodeling demonstrates that a set of density- and mechano-sensitive local tuning rules – consistent with experimental observations – can give rise to the emergent “rigidity homeostasis” behavior seen in the network as a whole. In rigidity homeostasis, the porous network manages to remodel extensively while remaining under tension. Independent work at a larger scale demonstrated that a similar set of rules is required to generate the “fluid-under-tension” behavior seen in the tissue that comprises the elongating body axis of chick embryos [39].

Other recent work has studied how tuning rules in tissue might generate desired patterns of stresses or strains, including convergent extension in fruit fly embryos [40] and prescribed cellular stress patterns in 3D epithelial tissues [41].

Another set of examples involves the development and homeostasis of vascular flow networks in mammals, plants and slime models, where adaptive rules – e.g. changes in the conductances (pipe width) are based on the existing currents through the pipe – lead to robust and functional distribution networks [4, 42, 43] (Fig 1C,D).

V. WHAT’S NEXT?

An overarching question is whether thinking of systems as physical learning analogues of artificial neural networks gives us additional insight that would not be obvious from other approaches. One paradigm shift that is already underway comes from results in artificial neural networks focused on overparameterization. While scientists used to think that one should generally use fewer parameters in a model than the number of desired features in the model output, AI developers have shown that artificial neural networks are effective and robust when there is an enormous number of tunable weights in the network, vastly larger than features they hope to classify with those networks. This is often described as a “double descent” in the cost function as a function of number of tunable weights [44, 45].

Double descent has recently been shown in a physical resistor network [46]; more broadly this implies that physical and biological systems, which have enormous reservoirs of tunable degrees of freedom, could easily use them to perform robust computation. An open ques-

tion is whether overparameterization helps to explain the observation that many non-exact tuning rules do surprisingly well at generating desired behavior. Does the fact that overparameterized systems have many nearly equally good solutions help explain why imperfect algorithms are still effective?

A related, nearly completely unaddressed question in the field is how the scale and architecture TDOF networks affect the tasks they can accomplish. For example, how does the complexity of the task that can be accomplished scale with the number of tunable degrees of freedom? Image classification could be performed using 9 nodes in a resistor network [3] (Fig 1B), so learning systems can be surprisingly small. How few tunable degrees of freedom can we get away with? A related question is how the nature of the tuning rule impacts performance. For example, it is clear that tuning rules that include nonlinear components substantially outperform linear tuning rules, and are required to learn nonlinear tasks. Is it possible to develop a “universal approximation theorem” for these physical systems that provides guidance about what types of tuning rules are optimal? How does the information processing and memory capacity of each tuning element impact learning algorithms and outcomes?

In addition, very little is known about how the network connectivity impacts the compute time or complexity of task – are certain topologies more efficient? Is there a matching between the architecture and the task? Are there tasks that could be accomplished with one connectivity but not another? A related set of theory questions involve the tasks themselves: How many tasks can be simultaneously tuned as a function of the number of nodes and the network connectivity? Are there some tasks that are mutually exclusive? Is there a way to predict *a priori* which tasks would be compatible or incompatible with one another?

Ultimately, one hopes that answering these questions might lead to a taxonomy or classification that explains how different network architectures, tuning rules, and tuning element capacities support different levels of task/function complexity.

From an applications standpoint, a pivotal choice involves deciding where to focus our design efforts. One potential set of applications centers on developing better computers: e.g. new types of specialized AI chips based on physical learning principles that work at vastly lower power than artificial neural networks that operate on bits, or decentralized computing structures that can survive severe environments or high damage situations [2, 23, 25].

Another avenue would be developing physical systems that do not learn or function as general computers, but instead accomplish less-complex tasks such as adapting and optimizing their behavior in response to their environment. Such systems would presumably have less stringent constraints on functionality of components, such as requiring fewer TDOFs or less complex tuning rules. For

example, one could imagine material for the wing of an airplane that alters its mechanical properties to allow smoother flight, or injecting particulate ‘interveners’ into a granular material to improve the flow properties for industrial applications like pharmaceutical processing.

A current major roadblock to realizing the promise of synthetic tunable physical systems is that there are not a lot of existing platforms where the material components are able to sense their local environment and update their properties. DNA tiles show promise [47], but may be expensive to scale up manufacturing for bulk materials. Another possibility might be microrobotic systems [48], perhaps embedded in a material backbone [49]. Other, currently unexplored platforms could be designed proteins or protein condensates, chemical reaction networks, hybrid engineered living systems [50], or dynamically reconfigurable polymer networks [51].

On the biology side there are an enormous number of open problems as well. Multiple theoretical/computational manuscripts predict specific tuning rules that could enable specific tasks in a wide range of biological systems at different scales. To carefully test/validate those predictions in experiments, researchers must identify observables that will distinguish between different tuning rules, and then experimentally perturb those tuning rules and compare to model predictions. Does this approach – identifying tasks and systematically identifying tuning rules that could enable them – simply confirm feedback loops that were already known, or can it generate new and surprising hypotheses about the underlying biological mechanisms?

Another avenue is exploring the utility of this approach in biological systems where it has not been applied so far – e.g. can it help in understanding organismal-scale phenomena like muscle memory and walking gait? Or maternal-fetal interactions in mammals? Would it be a useful way of organizing and searching for new feedback loops that lead to tropism in plants [52]? Could it help in analyzing community organization and emergent structures in social organisms like bees and ants [53–55]?

Obviously, these approaches will also intersect with (and hopefully complement) existing tools to understand complex biological organization. For example, the structure of PDOFs and TDOFs in tuning approaches is quite similar to dynamical systems approaches to understanding cell fate decision making [56] and population dynamics in ecology [57]. It also shares features with control theory, a common tool in systems biology [58]. As in control theory, it will be important to incorporate both the cost of control and the limits of controllability in future theories for tunable systems.

Finally, even from a canonical statistical physics/condensed matter perspective, materials that can tune themselves open up new challenges that should drive the field forward. If the small-scale interactions between units can tune themselves based on large-scale emergent behavior (e.g. rigidity or phase transitions), then our entire paradigm for coarse-graining breaks

down. For example, standard renormalization group approaches must be substantially modified to account for a direct large-to-small scale feedback. What are the observables, state parameters, and flows that would enable a robust theory of phase transitions and emergent phenomena in tunable systems with a purpose?

VI. ACKNOWLEDGMENTS

We acknowledge conversations with A.J. Liu, D. Durian, P. Mehta, J. Kondev, A. Murugan, I. Cohen,

R. Phillips, and the entire CZI Theory in Biology team that helped to shape this perspective. Support for an initial draft of this work was made possible in part by grant number 2023-329572 from the Chan Zuckerberg Initiative DAF, an advised fund of Silicon Valley Community Foundation. Additional support was provided by NSF DMR-CMMT-2532170.

-
- [1] Menachem Stern and Arvind Murugan. Learning without neurons in physical systems. *Annual Review of Condensed Matter Physics*, 14(1):417–441, 2023.
 - [2] Ali Momeni, Babak Rahmani, Benjamin Scellier, Logan G Wright, Peter L McMahon, Clara C Wanjura, Yuhang Li, Anas Skalli, Natalia G Berloff, Tatsuhiko Onodera, et al. Training of physical neural networks. *Nature*, 645(8079):53–61, 2025.
 - [3] Sam Dillavou, Menachem Stern, Andrea J. Liu, and Douglas J. Durian. Demonstration of Decentralized Physics-Driven Learning. *Physical Review Applied*, 18(1):014040, July 2022. Publisher: American Physical Society.
 - [4] Henrik Ronellenfitch and Eleni Katifori. Phenotypes of vascular flow networks. *Physical review letters*, 123(24):248101, 2019.
 - [5] Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.
 - [6] J Movellan. Contrastive hebbian learning in interactive networks. *Neural Computation*, 1990.
 - [7] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.
 - [8] Xiaohui Xie and H Sebastian Seung. Equivalence of backpropagation and contrastive hebbian learning in a layered network. *Neural computation*, 15(2):441–454, 2003.
 - [9] Benjamin Scellier and Yoshua Bengio. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in computational neuroscience*, 11:24, 2017.
 - [10] 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, May 2021. Publisher: American Physical Society.
 - [11] Vidyesh Rao Aniseti, Benjamin Scellier, and Jennifer M Schwarz. Learning by non-interfering feedback chemical signaling in physical networks. *Physical Review Research*, 5(2):023024, 2023.
 - [12] Martin Falk, Adam Strupp, Benjamin Scellier, and Arvind Murugan. Contrastive learning through non-equilibrium memory. *arXiv preprint arXiv:2312.17723*, 2023.
 - [13] Rituparno Mandal, Rosalind Huang, Michel Fruchart, Pepijn G Moerman, Suriyanarayanan Vaikuntanathan, Arvind Murugan, and Vincenzo Vitelli. Learning dynamical behaviors in physical systems. *arXiv preprint arXiv:2406.07856*, 2024.
 - [14] Menachem Stern, Adam G Frim, Raúl Candás, Andrea J Liu, and Vijay Balasubramanian. Contrastive learning in tunable dynamical systems. *arXiv preprint arXiv:2603.26969*, 2026.
 - [15] Engineering National Academies of Sciences and Medicine. *Frontiers of Materials That Learn: Proceedings of a Workshop*.
 - [16] Logan G Wright, Tatsuhiko Onodera, Martin M Stein, Tianyu Wang, Darren T Schachter, Zoey Hu, and Peter L McMahon. Deep physical neural networks trained with backpropagation. *Nature*, 601(7894):549–555, 2022.
 - [17] Arthur A. Evans, Jesse L. Silverberg, and Christian D. Santangelo. Lattice mechanics of origami tessellations. *Physical Review E*, 92(1):013205, July 2015. Publisher: American Physical Society.
 - [18] Varda F. Hagh, Sidney R. Nagel, Andrea J. Liu, M. Lisa Manning, and Eric I. Corwin. Transient learning degrees of freedom for introducing function in materials. *Proceedings of the National Academy of Sciences*, 119(19):e2117622119, May 2022. Publisher: Proceedings of the National Academy of Sciences.
 - [19] Ludovic Berthier and Mark D Ediger. Facets of glass physics. *Physics today*, 69(1):40–46, 2016.
 - [20] Lauren E Altman, Menachem Stern, Andrea J Liu, and Douglas J Durian. Experimental demonstration of coupled learning in elastic networks. *Physical Review Applied*, 22(2):024053, 2024.
 - [21] Nidhi Pashine, Daniel Hexner, Andrea J. Liu, and Sidney R. Nagel. Directed aging, memory, and nature’s greed. *Science Advances*, 5(12):eaax4215, December 2019. Publisher: American Association for the Advancement of Science.
 - [22] Menachem Stern, Chukwunonso Arinze, Leron Perez, Stephanie E Palmer, and Arvind Murugan. Supervised learning through physical changes in a mechanical system. *Proceedings of the National Academy of Sciences*, 117(26):14843–14850, 2020.
 - [23] Sam Dillavou, Benjamin D Beyer, Menachem Stern, Andrea J Liu, Marc Z Miskin, and Douglas J Durian. Machine learning without a processor: Emergent learning in a nonlinear analog network. *Proceedings of the National Academy of Sciences*, 121(28):e2319718121, 2024.
 - [24] Maxwell Aifer, Zach Belateche, Suraj Bramhavar, Kerem Y Camsari, Patrick J Coles, Gavin Crooks, Dou-

- glas J Durian, Andrea J Liu, Anastasia Marchenkova, Antonio J Martinez, et al. Solving the compute crisis with physics-based asics. *arXiv preprint arXiv:2507.10463*, 2025.
- [25] Dhireesha Kudithipudi, Catherine Schuman, Craig M Vineyard, Tej Pandit, Cory Merkel, Rajkumar Kubendran, James B Aimone, Garrick Orchard, Christian Mayr, Ryad Benosman, et al. Neuromorphic computing at scale. *Nature*, 637(8047):801–812, 2025.
- [26] Bhavin J Shastri, Alexander N Tait, Thomas Ferreira de Lima, Wolfram HP Pernice, Harish Bhaskaran, C David Wright, and Paul R Prucnal. Photonics for artificial intelligence and neuromorphic computing. *Nature Photonics*, 15(2):102–114, 2021.
- [27] Nikita Stroeve and Natalia G Berloff. Analog photonics computing for information processing, inference, and optimization. *Advanced Quantum Technologies*, 6(9):2300055, 2023.
- [28] John J Hopfield. Brain, neural networks, and computation. *Reviews of modern physics*, 71(2):S431, 1999.
- [29] Alison I Weber and Adrienne L Fairhall. The role of adaptation in neural coding. *Current opinion in neurobiology*, 58:135–140, 2019.
- [30] Arup K Chakraborty, Anupama Jayaraman, and Daniel P Newton. How the immune system learns. *PRX Life*, 4(2):021001, 2026.
- [31] Thierry Mora and Aleksandra M Walczak. Towards a quantitative theory of tolerance. *Trends in Immunology*, 44(7):512–518, 2023.
- [32] Jason G Cyster and Patrick C Wilson. Antibody modulation of b cell responses—incorporating positive and negative feedback. *Immunity*, 57(7):1466–1481, 2024.
- [33] Francesco Camaglia, Arie Ryvkin, Erez Greenstein, Shlomit Reich-Zeliger, Benny Chain, Thierry Mora, Aleksandra M Walczak, and Nir Friedman. Quantifying changes in the t cell receptor repertoire during thymic development. *Elife*, 12:e81622, 2023.
- [34] Harikesh S Wong, Kyemyung Park, Anita Gola, Antonio P Baptista, Christine H Miller, Deeksha Deep, Meng Lou, Lisa F Boyd, Alexander Y Rudensky, Peter A Savage, et al. A local regulatory t cell feedback circuit maintains immune homeostasis by pruning self-activated t cells. *Cell*, 184(15):3981–3997, 2021.
- [35] Deepa H Rajan, Ashley Albright, Hyeyoon Kim, Ulises Diaz, Yina Hudnall, Niklas Steube, Gautam Dey, Tao Liu, and Wallace F Marshall. Molecular pathways for learning in the single-cell stentor coeruleus. *Current Biology*, 2026.
- [36] Kristina Trifonova, Martin J Falk, Mason Rouches, Suriyanarayanan Vaikuntanathan, Michael Elowitz, and Arvind Murugan. Trainable computation in molecular networks. *BioRxiv*, 2025.
- [37] Deb S Banerjee, Martin J Falk, Margaret L Gardel, Aleksandra M Walczak, Thierry Mora, and Suriyanarayanan Vaikuntanathan. Learning via mechanosensitivity and activity in cytoskeletal networks. *PRX Life*, 4(1):013001, 2026.
- [38] Haina Wang, Marco A Galvani Cunha, John C Crocker, and Andrea J Liu. Mechanosensitive remodeling sustains rigidity homeostasis in actin cortex models. *bioRxiv*, pages 2025–10, 2025.
- [39] Alex T Grigas, Rajendra Singh Negi, Eirini Maniou, Gabriel L Galea, Arthur Michaut, Alessandro Mongera, and M Lisa Manning. Sparse mesenchymal cell networks as a fluid under tension. *bioRxiv*, pages 2025–12, 2025.
- [40] Sadjad Arzash, Andrea J Liu, and M Lisa Manning. Epithelial convergent extension as a tuning process. *bioRxiv*, pages 2025–11, 2025.
- [41] Shabeeb Ameen, Tao Zhang, and JM Schwarz. Training cell stress patterns in 3d cellular packings. *arXiv preprint arXiv:2604.25439*, 2026.
- [42] Lauren E Altman, Nadia Aguilar, Douglas J Durian, Miguel Ruiz-Garcia, and Eleni Katifori. Collective behavior and memory states in flow networks with tunable bistability. *Nature Communications*, 2026.
- [43] Karen Alim, Gabriel Amselem, François Peaudecerf, Michael P Brenner, and Anne Pringle. Random network peristalsis in physarum polycephalum organizes fluid flows across an individual. *Proceedings of the National Academy of Sciences*, 110(33):13306–13311, 2013.
- [44] Jason W Rocks and Pankaj Mehta. Bias-variance decomposition of overparameterized regression with random linear features. *Physical Review E*, 106(2):025304, 2022.
- [45] Rylan Schaeffer, Mikail Khona, Zachary Robertson, Akhilan Boopathy, Kateryna Pistunova, Jason W Rocks, Ila Rani Fiete, and Oluwasanmi Koyejo. Double descent demystified: Identifying, interpreting & ablating the sources of a deep learning puzzle. *arXiv preprint arXiv:2303.14151*, 2023.
- [46] Sam Dillavou, Jason W Rocks, Jacob F Wycoff, Andrea J Liu, and Douglas J Durian. Analog physical systems can exhibit double descent. *arXiv preprint arXiv:2511.17825*, 2025.
- [47] Constantine Glen Evans, Jackson O’Brien, Erik Winfree, and Arvind Murugan. Pattern recognition in the nucleation kinetics of non-equilibrium self-assembly. *Nature*, 625(7995):500–507, 2024.
- [48] Marc Z Miskin, Alejandro J Cortese, Kyle Dorsey, Edward P Esposito, Michael F Reynolds, Qingkun Liu, Michael Cao, David A Muller, Paul L McEuen, and Itai Cohen. Electronically integrated, mass-manufactured, microscopic robots. *Nature*, 584(7822):557–561, 2020.
- [49] Qingkun Liu, Wei Wang, Himani Sinhmar, Itay Griniasty, Jason Z Kim, Jacob T Pelster, Paragkumar Chaudhari, Michael F Reynolds, Michael C Cao, David A Muller, et al. Electronically configurable microscopic metasheet robots. *Nature Materials*, 24(1):109–115, 2025.
- [50] Chenghai Li, Nico Schramma, Zijun Wang, Nada F Qari, Maziyar Jalaal, Michael I Latz, and Shengqiang Cai. Ultrasensitive and robust mechanoluminescent living composites. *Science Advances*, 9(42):eadi8643, 2023.
- [51] Matthew J Webber and Mark W Tibbitt. Dynamic and reconfigurable materials from reversible network interactions. *Nature Reviews Materials*, 7(7):541–556, 2022.
- [52] Yasmine Meroz. Plant tropisms as a window on plant computational processes. *New Phytologist*, 229(4):1911–1916, 2021.
- [53] Michael S Engel and Tom Jackson. *Insect architecture: how insects build, engineer, and shape their world*. Princeton University Press, 2025.
- [54] Golnar Gharooni-Fard, Morgan Byers, Varad Deshmukh, Elizabeth Bradley, Carissa Mayo, Chad M Topaz, and Orit Peleg. A computational topology-based spatiotemporal analysis technique for honeybee aggregation. *npj Complexity*, 1(1):3, 2024.
- [55] Andrew Slezak and Varda F Hagh. Systematic design of local rules for directing emergent structure in bottom-up

- systems. *arXiv preprint arXiv:2604.12057*, 2026.
- [56] Meritxell Saez, James Briscoe, and David A Rand. Dynamical landscapes of cell fate decisions. *Interface focus*, 12(4), 2022.
- [57] Wenping Cui, Robert Marsland III, and Pankaj Mehta. Les houches lectures on community ecology: From niche theory to statistical mechanics, 2024.
- [58] Eduardo D Sontag. Some new directions in control theory inspired by systems biology. *IEE Proceedings-Systems Biology*, 1(1):9–18, 2004.