Physical Networks Become What They Learn

Menachem Stern[1,2] Marcelo Guzman[1] Felipe Martins[1] Andrea J. Liu[1,3] and Vijay Balasubramanian[1,3,4]

[1]*Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, Pennsylvania 19104, USA*
[2]*AMOLF, Science Park 104, 1098 XG Amsterdam, The Netherlands*
[3]*Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, New Mexico 87501, USA*
[4]*Rudolf Peierls Centre for Theoretical Physics, University of Oxford, Oxford OX1 3PU, United Kingdom*

Physical networks can develop tuned responses, or functions, by design, by evolution, or by learning via local rules. In all of these cases, tunable degrees of freedom characterizing internal interactions are modified to lower a *cost* penalizing deviations from desired outputs. An important class of such networks follows dynamics that minimize a global physical quantity, or Lyapunov function, with respect to physical degrees of freedom. In such networks, learning is a "double optimization" process in which two quantities, one defined by the task and the other prescribed by physics, are minimized with respect to different but coupled sets of variables. Here, we show how this learning process couples the high-dimensional "cost landscape" to the "physical landscape," linking the physical and cost Hessian matrices. Physical responses of trained networks to random perturbations thus reveal the functions to which they were tuned. Our results, illustrated using electrical networks with adaptable resistors, are generic to networks that perform tasks in the linear response regime.

Many physical networks have reciprocal edges, so that the interactions of nodes i with j and j with i are the same. Such networks can reach equilibrium or steady state by minimizing a global scalar function, or Lyapunov function, of the physical degrees of freedom. For example, mechanical networks minimize their (free) energy to achieve force balance on every node. Likewise, electrical resistor networks minimize dissipated power subject to current or voltage boundary conditions [1], or, equivalently, satisfy Kirchhoff's laws. This minimization produces specific voltages at network nodes, and can be understood as a form of physical computation [2–4]. Such physical networks can develop desired linear responses to perturbations by simultaneously minimizing a cost function specifying the desired response. For example, electrical networks that use local rules that approximate gradient descent to learn desired computations [5–12] can be made in the laboratory [13–17], and are arguably the simplest systems in which collective learning emerges from local dynamical processes.

Systems that optimize both a cost function and a physical Lyapunov function have physical imprints of the double optimization process in their linear response: soft modes aligned with learned tasks [18–20] appear in the dynamics. Here, we elucidate the precise connection between the physical linear response of the network and its cost function.

We show that in electrical networks that satisfy constraints on their responses to weak perturbations, the Hessian in the "physical landscape," characterizing curvatures around the minimized dissipated power in the space of node voltages, is directly related to the Hessian in the "cost landscape," characterizing curvatures around the minimized cost function in the space of adaptable degrees of freedom, or network parameters, namely the edge conductances. As in deep neural networks [21–24], the highest eigenmodes of the cost Hessian (directions of high curvature in the cost landscape) correspond to parameter changes that maximally impair performance of learned tasks. We derive an equation relating these stiff eigenmodes, after a transform determined by the network topology, to soft modes of the physical Hessian (directions of low curvature in the physical landscape).

Calculating the cost Hessian requires knowing the task. By contrast, the physical Hessian is a network property that can be measured *via* responses to random perturbations [25]. Our results imply that the physical Hessian gives insight into an adaptable network's function *without knowing the task*. This provides a new tool for understanding adaptable systems. While our exposition is focused on electrical resistor networks, our results are general to all physical networks that achieve desired linear responses via a double optimization process.

For networks that learn tasks via a contrastive learning procedure using local rules (such as electrical contrastive local learning networks that have been realized in the lab [13,16]), we derive a relation between the cost and physical Hessians during the learning process.

Physical and cost Hessians—Consider a linear electrical resistor network, with N nodes indexed by **a**, carrying

voltages $\mathbf{V_a}$ collected into a voltage vector $\mathbf{V}$. Nodes are connected by N_e edges indexed by i, of conductance κ_i. We use bold font for dynamical physical degrees of freedom (physical d.o.f.; node voltages), and Greek font for adaptive degrees of freedom (adaptive d.o.f.; edge conductances). Electrical resistor networks minimize total power dissipation. Because our networks are linear, the dissipated power is

$$\mathbf{P} = \frac{1}{2}\mathbf{V}^T\mathbf{HV}, \tag{1}$$

with a symmetric *physical* $N \times N$ Hessian matrix,

$$\mathbf{H_{ab}} = \frac{\partial^2 \mathbf{P}}{\partial \mathbf{V_a}\partial \mathbf{V_b}} \equiv 2[\mathbf{\Delta}^T\kappa\mathbf{\Delta}]_{\mathbf{ab}}, \tag{2}$$

where $\mathbf{a}$, $\mathbf{b}$ index N nodes; i, j index N_e edges connecting pairs of nodes; κ is a diagonal matrix of edge conductances with $\kappa_{ij} = \kappa_i\delta_{ij}$; and $[\cdots]_{\mathbf{ab}}$ indicates a component of the matrix in the parentheses. H is also the graph Laplacian, but we refer to it generically as the Hessian. $\mathbf{\Delta}$ is an "incidence matrix" specifying the network geometry, with $\mathbf{\Delta}_{j\mathbf{a}} = 1$, $0, -1$ depending on whether edge j is incoming, disconnected, or outgoing to node $\mathbf{a}$, with arbitrarily assigned edge directions. Thus, the physical Hessian $\mathbf{H}$ depends on the adaptive conductances κ_i.

When external currents $\mathbf{I_a}$, collected into a vector $\mathbf{I}$, are applied to network nodes, the network minimizes the "free state" power dissipation [1], i.e., total power subject to inputs $\mathbf{P}^F = \mathbf{V}^T(\frac{1}{2}\mathbf{HV} - \mathbf{I})$. The minimum of the free state power is achieved by the free state voltage values,

$$\mathbf{V}^F(\kappa; \mathbf{I}) = \mathbf{H}^{-1}(\kappa)\mathbf{I}. \tag{3}$$

For electrical networks, the Hessian in (2) is not strictly invertible because it has a zero mode—shifting all voltages by a constant leaves dissipated power unchanged. In practice we include an additional ground node with constrained voltage $V = 0$ [see Supplemental Material (SM) [26] Note 1]; this removes the zero mode and renders the Hessian invertible.

In this context, learning a target response amounts to adapting conductances to satisfy a constraint on the free state response $c(\mathbf{V}^F) = 0$ [28]. We quantify this objective by a scalar cost function C, the square of the constraint

$$C \equiv \frac{1}{2}c^2. \tag{4}$$

Multiple learning constraints (distinct input-output pairs in the machine learning literature) specified by c_i are codified by a joint cost $C = \frac{1}{2}\sum_i c_i^2$. For simplicity we consider tasks with a single constraint and show in SM Note 2 that our results on general relations between the physical and cost Hessian extend to multiple constraints, exemplified in

a linear regression task. The process of modifying the adaptable conductances to minimize the cost function C is called supervised learning when training examples with labels are provided. The cost measures how well the network reproduces labels, or, more generally, the desired response.

Learning terminates when the system reaches a minimum of the cost function, so that both the cost function and its gradient with respect to κ_i vanish. The cost landscape is then locally described to lowest nonvanishing order by the "cost Hessian," an $N_e \times N_e$ matrix,

$$\mathcal{H}_{ij} \equiv \frac{d^2C}{d\kappa_i d\kappa_j}, \tag{5}$$

where N_e counts adaptable parameters. The cost Hessian is studied in machine learning approaches [29]. For a network well-trained for n_T independent constraints, the cost Hessian has n_T finite eigenvalues [21–23].

Relation between physical and cost Hessians—The physical and cost Hessians have different dimensions and units, but we will see that they are nevertheless related. We can rewrite the cost Hessian as

$$\mathcal{H}_{ij} = \frac{dc}{d\kappa_i}\frac{dc}{d\kappa_j} + c\frac{d^2c}{d\kappa_i d\kappa_j} \equiv \mathcal{H}_{ij}^{\text{sat}} + \mathcal{H}_{ij}^{\text{dyn}}, \tag{6}$$

a sum of the "satisfied" term $\mathcal{H}_{ij}^{\text{sat}} \equiv (dc/d\kappa_i)(dc/d\kappa_j)$ and the "dynamic" term $\mathcal{H}_{ij}^{\text{dyn}} \equiv c(d^2c/d\kappa_i d\kappa_j)$. $\mathcal{H}_{ij}^{\text{dyn}}$ vanishes if the network achieves learning since $c = 0$.

From (6) we see that for one constraint, $\mathcal{H}_{ij}^{\text{sat}}$ is rank-1, as it is an outer product of a vector $g_i \equiv (dc/d\kappa_i)$ with itself. g_i is proportional to an eigenmode of $\mathcal{H}_{ij}^{\text{sat}}$.

To relate the physical and cost Hessians, recall that the constraint $c(\mathbf{V}^F)$ is an explicit function of the free state response, so that

$$g_i = \frac{\partial c}{\partial\mathbf{V}^F}\cdot\frac{d\mathbf{V}^F}{d\kappa_i} = -\frac{\partial c}{\partial\mathbf{V}^F}\cdot\mathbf{H}^{-1}\frac{d\mathbf{H}}{d\kappa_i}\mathbf{H}^{-1}\mathbf{I}, \tag{7}$$

where we used (3) along with the formula for the derivative of the inverse of a matrix. Finally, applying the κ_i derivative to the physical Hessian in Eq. (2) we arrive at

$$g_i = -2\left(\frac{\partial c}{\partial\mathbf{V}^F}\cdot\mathbf{H}^{-1}\mathbf{\Delta}_i^T\right)(\mathbf{\Delta}_i\mathbf{H}^{-1}\mathbf{I}), \tag{8}$$

where by $\mathbf{\Delta}_i$ we mean the ith row of $\mathbf{\Delta}$. So $\mathbf{\Delta}_i$ is a vector of length N whose entries are 1 for the node that edge i enters, -1 for the node it leaves, and 0 for the other nodes. We find that

$$\mathcal{H}_{ij}^{\text{sat}} = g_i g_j, \tag{9}$$

where $\mathcal{H}_{ij}^{\text{sat}}$ contains four inverse powers of $\mathbf{H}$. The additional factors of $\boldsymbol{\Delta}$ from (8) implement a transform determined by the network topology, transporting the inverse physical Hessian into network parameter space.

Equations (8) and (9) encode our main result, a remarkable connection between the physical and cost landscapes when the constraint is satisfied, thus minimizing cost: the curvature around the solution in the *cost* landscape is directly related to the curvature around the minimum of the *physical* landscape. This connection arises in electrical networks because power must be minimized to satisfy Kirchhoff's law at every node while cost is decreased; this couples the two landscapes in their respective spaces.

The satisfied cost Hessian in (9) is an outer product of g_i (the gradient constraint c with respect to κ_i) with itself. Thus, for one constraint, we can think of a *reduced* cost Hessian of rank-1, with one nonzero eigenvalue and corresponding eigenmode g_i associated with the learned task. Moreover, from the definition of the cost function (4),

$$\frac{dC}{d\kappa_i} = cg_i. \qquad (10)$$

As the cost vanishes, its gradient is proportional to the nontrivial cost Hessian eigenmode.

Our analysis generalizes to physical systems that minimize a scalar physical global quantity, including mechanical networks that minimize energy or free energy (SM Note 4), or not only linear but also nonlinear flow and electrical networks that minimize dissipated power or, in the nonlinear case, a "cocontent" [6,30]. The reason is that, for weak inputs, the dynamics are controlled by a quadratic energy functional with a fixed Hessian, and hence linear responses, leading to equations of the same form as those we have solved.

To relate the physical and cost Hessian eigenmodes, we rotate to natural coordinates of the physical Hessian,

$$H = \boldsymbol{v}^T \Lambda \boldsymbol{v}; \mathbf{i} \equiv \boldsymbol{v}\mathbf{I}; \mathbf{o} \equiv \boldsymbol{v}\frac{dc}{d\mathbf{V}^F}. \qquad (11)$$

Here, the rows of matrix $\boldsymbol{v}$ are eigenmodes of the physical Hessian, Λ is a diagonal matrix of associated eigenvalues $\lambda > 0$, and $\mathbf{i}$, $\mathbf{o}$ are the input current and constraint gradient rotated to the physical Hessian reference frame.

In this coordinate system,

$$g_i = -2\sum_{\mu\nu} \left(\frac{\mathbf{i}}{\lambda}\right)_{\boldsymbol{\mu}} \left(\frac{\mathbf{o}}{\lambda}\right)_{\boldsymbol{\nu}} (\boldsymbol{\Delta}_i \boldsymbol{v}_{\boldsymbol{\mu}}^T)(\boldsymbol{\Delta}_i \boldsymbol{v}_{\boldsymbol{\nu}}^T). \qquad (12)$$

The vector $(\mathbf{i}/\lambda)_{\boldsymbol{\mu}}$ is the input current projected onto the direction of eigenmode $\boldsymbol{v}_{\boldsymbol{\mu}}$, divided by the associated eigenvalue $\lambda_{\boldsymbol{\mu}}$; similarly for $(\mathbf{o}/\lambda)_{\boldsymbol{\mu}}$. $\boldsymbol{\Delta}_i \boldsymbol{v}_{\boldsymbol{\mu}}^T$ is the voltage difference across the ith edge for the $\boldsymbol{\mu}$th eigenvector.

Intuitively, a relation arises between the cost and physical Hessians' eigenmodes in trained systems because the highest eigenmodes of the cost Hessian *define* a desired *physical response*. This physical linear response $\mathbf{V}^F = \mathbf{H}^{-1}\mathbf{I} = \sum_{\mu}(\mathbf{i}/\lambda)_{\boldsymbol{\mu}}\boldsymbol{v}_{\mu}^T$ is controlled by the inverse physical Hessian and dominated by its lowest eigenmodes. Thus, the lowest physical modes that project onto the desired response, having large $(\mathbf{o}/\lambda)_{\boldsymbol{\mu}}$, are related to the highest eigenmodes of the cost Hessian.

We previously showed that in systems learning linear responses, the physical Hessian develops soft modes with a small eigenvalue and large projections onto the desired responses [20]. Note that soft modes that are not learned could exist if a network has certain symmetries or structural features. However, such unlearned modes are independent of the adaptive degrees of freedom, and generally do not project much onto the desired system response, and thus do not couple to g_i or the cost Hessian. In simple cases where only one output constraint is learned, one low eigenmode arises with a large projection on the constraint. If its eigenvalue is much smaller than the rest, we can approximate (12) by one term, as discussed above for the cost Hessian of networks trained to achieve one constraint,

$$g_i^{\text{Approx}} = -2\frac{\mathbf{i}_1 \mathbf{o}_1}{\lambda_1^2}(\boldsymbol{\Delta}_i \boldsymbol{v}_1^T)^2. \qquad (13)$$

In this case we find that the stiff mode of the cost Hessian $\mathcal{H}$ has components proportional to the squared differences of node voltages belonging to the soft mode of the physical Hessian $\mathbf{H}$. Note that $\mathbf{H}$ can be estimated from physical responses alone [25], with no knowledge about the constraints the system was required to satisfy. This implies that without knowing $\mathbf{i}$, $\mathbf{o}$ we can obtain a good approximation of the stiff mode of $\mathcal{H}^{\text{sat}}$. One does not need to specify the task—to know the constraint or cost function—in order to learn which edges most strongly affect the error in a trained network. It is more complicated to relate individual physical eigenmodes to constraints when multiple soft modes of $\mathbf{H}$ are relevant. We leave these cases to future work.

Figure 1 shows a network trained so that a voltage drop applied between the two white input nodes yields an equal voltage drop between the two black output nodes. Here, inputs are applied voltages rather than applied currents as discussed above. In our analysis, inputs could be set as currents or voltages (relative to ground) *supplied to nodes*, or as fixed voltages *across edges*. Also, we consider output constraints that do not explicitly depend on the adaptive conductances κ_i (SM Notes 1 and 5). If the output constraints depend explicitly on κ_i (e.g., for power constraints), or if they do not have a leading linear term in the response $\mathbf{V}^F$, our results are modified as discussed in SM Note 3. In Fig. 1, training partitions the network into high
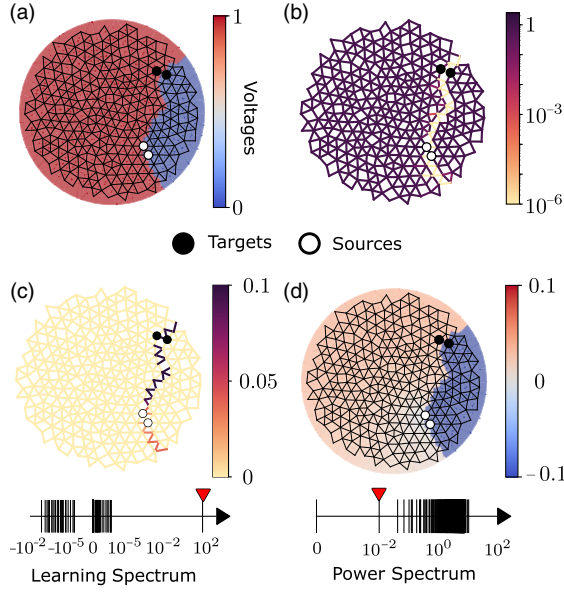
FIG. 1. Linear electric resistor networks have correlated modes in the physical and cost spaces. Here, the task requires the voltage drop between two output nodes (black) to equal the voltage drop between two input nodes (white). (a) The response of the network. Color interpolates between node voltages. The response is partitioned into two sectors, with high uniform voltage and low uniform voltage, respectively. (b) The edge conductances; the network develops a crack of low conductance. (c) The stiff mode of the cost Hessian picks out the edges in the crack. (d) The soft mode of the physical Hessian reflects the voltage response and complements the stiff mode of the learning cost Hessian.

voltage and low voltage sectors [Fig. 1(a)], separated by a crack of low conductance [Fig. 1(b)].

Figure 1(c) shows that the stiff mode of the cost Hessian is spatially localized to the low-conductance crack; if these conductances are varied, the cost increases strongly. Likewise Fig. 1(d) shows that the soft mode of the physical Hessian reflects the voltage response in Fig. 1(a) and complements the stiff mode of the cost Hessian. This is the soft response mode reached by learning. Note that many other solutions, each with its own physical Hessian, are also possible [31] but following [20,32,33], systems trained for such linear responses typically find solutions with stiff cost eigenmodes and soft physical eigenmodes. Significantly, the lowest nontrivial eigenmode of the physical Hessian [Fig. 1(d)], captures the network response to the input, without knowledge of the expected input or output. Applying the difference operator Δ_i to this soft mode, we find that $(\Delta_i v_1)^2$ is spatially localized to the crack; the soft mode reflects key edges associated with the stiff cost Hessian mode. The latter demarcates the boundary between sectors separated by the physical Hessian, which can also be revealed through persistent homology analysis of the trained network [33,34].

Once the constraint is satisfied, we can approximate the cost Hessian as (9)

$$\mathcal{H}_{ij}^{\text{Approx}} \approx 4 \frac{(\mathbf{i}_1\mathbf{o}_1)^2}{\lambda_1^4} (\Delta_i \boldsymbol{v}_1^T)^2 (\Delta_j \boldsymbol{v}_1^T)^2. \quad (14)$$

This has rank-1 as the outer product of a vector. The trace equals the nonzero eigenvalue: $\lambda_1^{\mathcal{L}} = \text{Tr}(\mathcal{H}^{\text{Approx}})$, relating the stiff cost to the soft physical eigenvalue,

$$\lambda_1^{\mathcal{L}} = 4\sum_i g_i^2 \approx \frac{(\mathbf{i}_1\mathbf{o}_1)^2}{\lambda_1^4} \sum_i (\Delta_i \boldsymbol{v}_1^T)^4 \propto \lambda_1^{-4}. \quad (15)$$

These results are illustrated in SM Note 6 and Fig. S3 with resistor networks trained on tasks similar to Fig. 1.

Dynamical Hessian relations—We now examine how the two Hessians are related during learning, not just when the constraints have been satisfied. From (6),

$$\mathcal{H}_{ij} = g_i g_j + 4cL_i M_{ij} R_j + 4cL_j M_{ji} R_i$$
$$+ 4cR_i R_j \sum_{\mathbf{ab}} (\Delta_i \mathbf{H}^{-1})_{\mathbf{a}} \frac{\partial^2 c}{\partial \mathbf{V}_{\mathbf{a}}^F \partial \mathbf{V}_{\mathbf{b}}^F} (\mathbf{H}^{-1}\Delta_j^T)_{\mathbf{b}}, \quad (16)$$

where we defined $L_i \equiv (\partial c/\partial \mathbf{v}^F) \cdot \mathbf{H}^{-1}\Delta_i^T$, $R_i \equiv \Delta_i \mathbf{H}^{-1}\mathbf{I}$, and $M_{ij} \equiv \Delta_i \mathbf{H}^{-1}\Delta_j^T$ (details in SM Note 7). The last three terms arise from $\mathcal{H}_{ij}^{\text{dyn}}$. Recall that $\mathcal{H}_{ij}^{\text{sat}} \sim \mathbf{H}^{-4}$. The first two terms in $\mathcal{H}_{ij}^{\text{Dyn}}$ scale as $c\mathbf{H}^{-3} \sim \sqrt{C}\mathbf{H}^{-3}$, while the last term scales as $c\mathbf{H}^{-4} \sim \sqrt{C}\mathbf{H}^{-4}$. For systems trained with linear constraints, the last term vanishes because $(\partial^2 c/\partial \mathbf{V}_a^F \partial \mathbf{V}_b^F) = 0$. Then $\mathcal{H}^{\text{sat}}$ dominates, $\text{Tr}[(\mathcal{H}^{\text{sat}})^2] \gg \text{Tr}[(\mathcal{H}^{\text{dyn}})^2]$ once the error is less than a scale set by the lowest physical Hessian eigenvalue, $C \ll \lambda_1^{-2}$. For nonlinear constraints, $\mathcal{H}^{\text{sat}}$ still dominates, at least for small enough error, $C \ll 1$. The change of the physical Hessian $\mathbf{H}$ during learning is proportional to the cost function gradient. Thus, the physical Hessian $\mathbf{H}$ becomes nearly constant when C is low enough.

Figure 2 shows results for 300 networks trained with a linear constraint to produce specified voltage drops to applied input currents, and calculated $\mathcal{H}^{\text{sat}}$ and $\mathcal{H}^{\text{dyn}}$. The cost Hessian $\mathcal{H}$ is well described by $\mathcal{H}^{\text{sat}}$ [Fig. 2(a)]—the normalized double dot product $\mathcal{H}^{\text{sat}} : \mathcal{H} \equiv \{\text{Tr}[\mathcal{H}^{\text{sat}}\mathcal{H}]/\sqrt{\text{Tr}[(\mathcal{H}^{\text{sat}})^2]}\sqrt{\text{Tr}[\mathcal{H}^2]}\}$ approaches 1 (black) when the cost is below a threshold λ_1^{-2} set by the lowest physical Hessian eigenvalue, λ_1. Similarly, the physical approximation of g_i (8) approaches the stiffest cost eigenvector $\{\mathcal{H}\}_1$, generating its rank-1 reduced form. Figure 2(a) also shows that the ratio of the dynamic and satisfied terms scales with $\sqrt{C}$ (blue), as predicted. Figure 2(b) shows the cost Hessian for a network before and after training. Initially $\mathcal{H}^{\text{sat}}$ approximates the cost Hessian poorly, but afterward captures it well, with the same sparse, low-rank features.
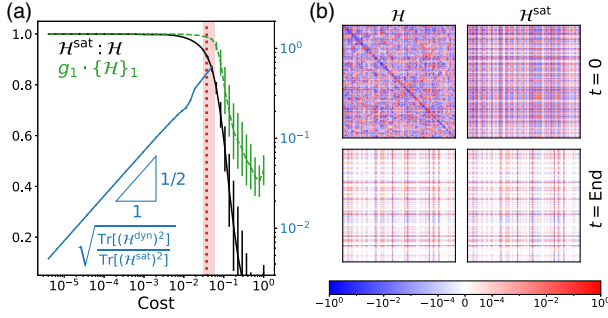
FIG. 2. Cost Hessian as cost C decreases during the learning for a task; constraining voltage drops between output nodes in response to currents applied at input nodes. (a) The cost Hessian is captured well by $\mathcal{H}^{\text{sat}}$ (black curve), when C is less than the scale set by the lowest physical eigenvalue $C \lesssim \lambda_1^{-2}$. Vertical red line = median of λ_1^{-2}. Middle 80% percentile of the range = red shaded region. Similarly, below that same scale, the g_i vector in Eq. (8) (green curve) captures the stiffest vector of the cost Hessian ($\{\mathcal{H}\}_1$). The ratio of dynamic to satisfied cost Hessians (blue curve) scales as $\sqrt{C}$, as predicted. (b) Comparison of the full and satisfied cost Hessians before and after the learning process.

Discussion—We derived relations between the physical and cost Hessians for networks responding to weak forces near equilibrium as they adapt to perform tasks. For simplicity, in the main text we focused on networks satisfying a single constraint, and thus have a rank-1 reduced cost Hessian. In SM Note 2 we show how networks that satisfy n_T constraints have reduced cost Hessians with rank up to n_T. We also showed how, after learning, the high eigenmodes of the cost Hessian are related to low eigenmodes of the physical Hessian, and used examples to illustrate how these high modes are supported primarily on network edges that adjust to satisfy the constraints. For a physical system that satisfies a single constraint, our results show that we should expect a single low physical eigenmode, separated by a gap from the higher modes, and directly related to a single high cost Hessian eigenmode. Then, the satisfied constraint can be discovered by examining the physical responses of the network around equilibrium without any prior knowledge of the task. Even when the low physical eigenmodes are not separated by a gap, the connection between the Hessians suggests strategies for learning about stiff cost Hessian eigenmodes from physical information only [35]. While we treated the case of linear electric resistor networks, our results hold generally for steady-state physical networks that minimize a scalar cost specifying a linear response (SM Note 4). In other words, we find that, in a sense, *physical networks become what they learn*.

Our approach applies to adaptable mechanical systems [36–40], including proteins, where correlations between physical structure and function [41–46] appear in regions highly conserved over evolutionary times, suggesting their functional importance [42,47]. Conserved regions in proteins were associated with slow collective modes—corresponding to low eigenmodes of the physical Hessian [48]. Because our approach links the physical and cost Hessians for each system, it can give insight into individual systems as well as for an ensemble of solutions, e.g., a protein family, shedding light on idiosyncratic features that play functional roles [49].

More broadly, circuits in the brain display structures conserved between individuals and species [50]. In many cases these conserved structures are evolved or learned adaptations to the environment or to tasks, for example, in circuits supporting vision [51–53], audition [54,55], olfaction [56,57], and spatial cognition [58]. However, individual differences are also functionally important, e.g., for encoding memories [59,60]. Our findings linking structure and function in individual networks may help to explain how conserved network adaptations can be accompanied by substantial individual variations [61,62]. Finally, the connections we have uncovered between the cost Hessian and physically measurable quantities suggest that we can apply the rich machine learning literature exploiting the cost Hessian to develop more efficient physical learning rules [29].

[1] Sri Krishna Vadlamani, Tianyao Patrick Xiao, and Eli Yablonovitch, Physics successfully implements Lagrange multiplier optimization, Proc. Natl. Acad. Sci. U.S.A. **117**, 26639 (2020).

[2] Rolf Landauer, Computation: A fundamental physical view, Phys. Scr. **35**, 88 (1987).

[3] Gualtiero Piccinini, *Physical Computation: A Mechanistic Account* (Oxford University Press, Oxford, 2015).

[4] Herbert Jaeger, Beatriz Noheda, and Wilfred G Van Der Wiel, Toward a formal theory for computing machines made out of whatever physics offers, Nat. Commun. **14**, 4911 (2023).

[5] Benjamin Scellier and Yoshua Bengio, Equilibrium propagation: Bridging the gap between energy-based models and backpropagation, Front. Comput. Neurosci. **11**, 24 (2017).

[6] Jack Kendall, Ross Pantone, Kalpana Manickavasagam, Yoshua Bengio, and Benjamin Scellier, Training end-to-end analog neural networks with equilibrium propagation, arXiv:2006.01981.

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

[8] Corinna Kaspar, Bart Jan Ravoo, Wilfred G. van der Wiel, Seraphine V. Wegner, and Wolfram H. P. Pernice, The rise of intelligent matter, Nature (London) **594**, 345 (2021).

[9] Menachem Stern and Arvind Murugan, Learning without neurons in physical systems, Annu. Rev. Condens. Matter Phys. **14**, 417 (2023).

[10] Victor Lopez-Pastor and Florian Marquardt, Self-learning machines based on Hamiltonian echo backpropagation, Phys. Rev. X **13**, 031020 (2023).

[11] Vidyesh Rao Anisetti, Benjamin Scellier, and Jennifer M. Schwarz, Learning by non-interfering feedback chemical signaling in physical networks, Phys. Rev. Res. **5**, 023024 (2023).

[12] Vidyesh Rao Anisetti, Ananth Kandala, Benjamin Scellier, and J. M. Schwarz, Frequency propagation: Multimechanism learning in nonlinear physical networks, Neural Comput. **36**, 596 (2024).

[13] Sam Dillavou, Menachem Stern, Andrea J. Liu, and Douglas J. Durian, Demonstration of decentralized physics-driven learning, Phys. Rev. Appl. **18**, 014040 (2022).

[14] Jacob F. Wycoff, Sam Dillavou, Menachem Stern, Andrea J. Liu, and Douglas J. Durian, Desynchronous learning in a physics-driven learning network, J. Chem. Phys. **156**, 144903 (2022).

[15] Menachem Stern, Sam Dillavou, Marc Z. Miskin, Douglas J. Durian, and Andrea J. Liu, Physical learning beyond the quasistatic limit, Phys. Rev. Res. **4**, L022037 (2022).

[16] 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, Proc. Natl. Acad. Sci. U.S.A. **121**, e2319718121 (2024).

[17] Menachem Stern, Sam Dillavou, Dinesh Jayaraman, Douglas J. Durian, and Andrea J. Liu, Training self-learning circuits for power-efficient solutions, APL Mach. Learn. **2**, 016114 (2024).

[18] Benjamin B. Machta, Ricky Chachra, Mark K. Transtrum, and James P. Sethna, Parameter space compression underlies emergent theories and predictive models, Science **342**, 604 (2013).

[19] Vidyesh Rao Anisetti, Ananth Kandala, and J. M. Schwarz, Emergent learning in physical systems as feedback-based aging in a glassy landscape, arXiv:2309.04382.

[20] Menachem Stern, Andrea J. Liu, and Vijay Balasubramanian, Physical effects of learning, Phys. Rev. E **109**, 024311 (2024).

[21] Levent Sagun, Leon Bottou, and Yann LeCun, Eigenvalues of the Hessian in deep learning: Singularity and beyond, arXiv:1611.07476.

[22] Levent Sagun, Utku Evci, V. Ugur Guney, Yann Dauphin, and Leon Bottou, Empirical analysis of the Hessian of over-parametrized neural networks, arXiv:1706.04454.

[23] Mingwei Wei and David J. Schwab, How noise affects the Hessian spectrum in overparameterized neural networks, arXiv:1910.00195.

[24] Mahalakshmi Sabanayagam, Freya Behrens, Urte Adomaityte, and Anna Dawid, Unveiling the Hessian's connection to the decision boundary, arXiv:2306.07104.

[25] Ke Chen, Wouter G. Ellenbroek, Zexin Zhang, Daniel T. N. Chen, Peter J. Yunker, Silke Henkes, Carolina Brito, Olivier Dauchot, Wim Van Saarloos, Andrea J. Liu *et al.*, Low-frequency vibrations of soft colloidal glasses, Phys. Rev. Lett. **105**, 025501 (2010).

[26] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.134.147402 for results in nonlinear systems, regression tasks, and more information about simulation protocols, which includes Ref. [27].

[27] Carl P. Goodrich, Andrea J. Liu, and Sidney R. Nagel, The principle of independent bond-level response: Tuning by pruning to exploit disorder for global behavior, Phys. Rev. Lett. **114**, 225501 (2015).

[28] Pankaj Mehta, Marin Bukov, Ching-Hao Wang, Alexandre G. R. Day, Clint Richardson, Charles K. Fisher, and David J. Schwab, A high-bias, low-variance introduction to machine learning for physicists, Phys. Rep. **810**, 1 (2019).

[29] Yann N. Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio, Identifying and attacking the saddle point problem in high-dimensional non-convex optimization, in *Advances in Neural Information Processing Systems, Montreal, Canada*, edited by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger (MIT Press, Cambridge, MA, 2014), Vol. 27.

[30] Mauro Parodi and Marco Storace, *Linear and Nonlinear Circuits: Basic & Advanced Concepts* (Springer, New York, 2018), Vol. 1.

[31] Jason W. Rocks, Henrik Ronellenfitsch, Andrea J. Liu, Sidney R. Nagel, and Eleni Katifori, Limits of multifunctionality in tunable networks, Proc. Natl. Acad. Sci. U.S.A. **116**, 2506 (2019).

[32] Jason W. Rocks, Nidhi Pashine, Irmgard Bischofberger, Carl P. Goodrich, Andrea J. Liu, and Sidney R. Nagel, Designing allostery-inspired response in mechanical networks, Proc. Natl. Acad. Sci. U.S.A. **114**, 2520 (2017).

[33] Jason W. Rocks, Andrea J. Liu, and Eleni Katifori, Hidden topological structure of flow network functionality, Phys. Rev. Lett. **126**, 028102 (2021).

[34] Jason W. Rocks, Andrea J. Liu, and Eleni Katifori, Revealing structure-function relationships in functional flow networks via persistent homology, Phys. Rev. Res. **2**, 033234 (2020).

[35] Marcelo Guzman, Felipe Martins, Menachem Stern, and Andrea J. Liu, Microscopic imprints of learned solutions in adaptive resistor networks, arXiv:2412.19356.

[36] Nidhi Pashine, Daniel Hexner, Andrea J. Liu, and Sidney R. Nagel, Directed aging, memory, and nature's greed, Sci. Adv. **5**, eaax4215 (2019).

[37] Menachem Stern, Chukwunonso Arinze, Leron Perez, Stephanie E. Palmer, and Arvind Murugan, Supervised

learning through physical changes in a mechanical system, Proc. Natl. Acad. Sci. U.S.A. **117,** 14843 (2020).

[38] Chukwunonso Arinze, Menachem Stern, Sidney R. Nagel, and Arvind Murugan, Learning to self-fold at a bifurcation, Phys. Rev. E **107,** 025001 (2023).

[39] Lauren E. Altman, Menachem Stern, Andrea J. Liu, and Douglas J. Durian, Experimental demonstration of coupled learning in elastic networks, Phys. Rev. Appl. **22,** 024053 (2024).

[40] Tie Mei and Chang Qing Chen, In-memory mechanical computing, Nat. Commun. **14,** 5204 (2023).

[41] Najeeb Halabi, Olivier Rivoire, Stanislas Leibler, and Rama Ranganathan, Protein sectors: Evolutionary units of three-dimensional structure, Cell **138,** 774 (2009).

[42] Tiberiu Teşileanu, Lucy J. Colwell, and Stanislas Leibler, Protein sectors: Statistical coupling analysis versus conservation, PLoS Comput. Biol. **11,** e1004091 (2015).

[43] Arjun S. Raman, K. Ian White, and Rama Ranganathan, Origins of allostery and evolvability in proteins: A case study, Cell **166,** 468 (2016).

[44] Michael R. Mitchell, Tsvi Tlusty, and Stanislas Leibler, Strain analysis of protein structures and low dimensionality of mechanical allosteric couplings, Proc. Natl. Acad. Sci. U.S.A. **113,** E5847 (2016), https://www.pnas.org/doi/full/10.1073/pnas.1609462113.

[45] Tsvi Tlusty, Albert Libchaber, and Jean-Pierre Eckmann, Physical model of the genotype-to-phenotype map of proteins, Phys. Rev. X **7,** 021037 (2017).

[46] Eric Rouviere, Rama Ranganathan, and Olivier Rivoire, Emergence of single-versus multi-state allostery, PRX Life **1,** 023004 (2023).

[47] Victor H. Salinas and Rama Ranganathan, Coevolution-based inference of amino acid interactions underlying protein function, eLife **7,** e34300 (2018).

[48] Kabir Husain and Arvind Murugan, Physical constraints on epistasis, Mol. Biol. Evol. **37,** 2865 (2020).

[49] Martin J. Falk, Jiayi Wu, Ayanna Matthews, Vedant Sachdeva, Nidhi Pashine, Margaret L. Gardel, Sidney R. Nagel, and Arvind Murugan, Learning to learn by using nonequilibrium training protocols for adaptable materials, Proc. Natl. Acad. Sci. U.S.A. **120,** e2219558120 (2023).

[50] Ricardo Olivares, Juan Montiel, and Francisco Aboitiz, Species differences and similarities in the fine structure of the mammalian corpus callosum, Brain Behav. Evol. **57,** 98 (2001).

[51] Charles P. Ratliff, Bart G. Borghuis, Yen-Hong Kao, Peter Sterling, and Vijay Balasubramanian, Retina is structured to process an excess of darkness in natural scenes, Proc. Natl. Acad. Sci. U.S.A. **107,** 17368 (2010).

[52] Patrick Garrigan, Charles P. Ratliff, Jennifer M. Klein, Peter Sterling, David H. Brainard, and Vijay Balasubramanian, Design of a trichromatic cone array, PLoS Comput. Biol. **6,** e1000677 (2010).

[53] Ann M. Hermundstad, John J. Briguglio, Mary M. Conte, Jonathan D. Victor, Vijay Balasubramanian, and Gašper Tkačik, Variance predicts salience in central sensory processing, eLife **3,** e03722 (2014).

[54] Michael S. Lewicki, Efficient coding of natural sounds, Nat. Neurosci. **5,** 356 (2002).

[55] Evan C. Smith and Michael S. Lewicki, Efficient auditory coding, Nature (London) **439,** 978 (2006).

[56] Tiberiu Teşileanu, Simona Cocco, Remi Monasson, and Vijay Balasubramanian, Adaptation of olfactory receptor abundances for efficient coding, eLife **8,** e39279 (2019).

[57] Kamesh Krishnamurthy, Ann M. Hermundstad, Thierry Mora, Aleksandra M. Walczak, and Vijay Balasubramanian, Disorder and the neural representation of complex odors, Front. Comput. Neurosci. **16,** 917786 (2022).

[58] Xue-Xin Wei, Jason Prentice, and Vijay Balasubramanian, A principle of economy predicts the functional architecture of grid cells, eLife **4,** e08362 (2015).

[59] Edward K. Vogel and Maro G. Machizawa, Neural activity predicts individual differences in visual working memory capacity, Nature (London) **428,** 748 (2004).

[60] Lisa Feldman Barrett, Michele M. Tugade, and Randall W. Engle, Individual differences in working memory capacity and dual-process theories of the mind, Psychol. Bull. **130,** 553 (2004).

[61] Romke Rouw and H. Steven Scholte, Neural basis of individual differences in synesthetic experiences, J. Neurosci. **30,** 6205 (2010).

[62] Svetlana V. Shinkareva, Vicente L. Malave, Marcel Adam Just, and Tom M. Mitchell, Exploring commonalities across participants in the neural representation of objects, Hum. Brain Mapp. **33,** 1375 (2012).