Microscopic Imprints of Learned Solutions in Tunable Networks

Marcelo Guzman,[1,*] Felipe Martins,[1,*] Menachem Stern,[2] and Andrea J. Liu[1]
[1]*Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, Pennsylvania 19104, USA*
[2]*AMOLF, Science Park 104, 1098 XG Amsterdam, The Netherlands*

(Received 29 January 2025; accepted 15 July 2025; published 27 August 2025)

In physical networks trained using supervised learning, physical parameters are adjusted to produce desired responses to inputs. An example is an electrical contrastive local learning network of nodes connected by edges that adjust their conductances during training. When an edge conductance changes, it upsets the current balance of every node. In response, physics adjusts the node voltages to minimize the dissipated power. Learning in these systems is therefore a coupled double-optimization process, in which the network descends both a cost landscape in the high-dimensional space of edge conductances and a physical landscape—the power dissipation—in the high-dimensional space of node voltages. Because of this coupling, the physical landscape of a trained network contains information about the learned task. Here, we derive a structure-function relation for trained tunable networks and demonstrate that all the physical information relevant to the trained input-output relation can be captured by a tuning susceptibility, an experimentally measurable quantity. We supplement our theoretical results with simulations to show that the tuning susceptibility is correlated with functional importance and that we can extract physical insight into how the system performs the task from the conductances of highly susceptible edges. Our analysis is general and can be applied directly to mechanical networks, such as networks trained for protein-inspired function such as allostery.

DOI: 10.1103/f2hb-c9s1

Subject Areas: Complex Systems, Metamaterials, Soft Matter

I. INTRODUCTION

In artificial neural networks, the cost landscape describes the value of the cost function C in terms of the parameters, or "tunable degrees of freedom," k_i [1]. Learned solutions correspond to minima of this landscape, which are locally described by the cost Hessian [2–4], the matrix of second derivatives of the cost with respect to the tunable degrees of freedom, $\mathcal{H}_{ij} = \partial_{k_i}\partial_{k_j}C$. In the directions of the highest eigenmodes of the cost Hessian, corresponding to the highest positive curvatures, small changes in the tunable degrees of freedom cause a substantial increase of the cost. These key tunable degrees of freedom reveal how the system achieves the task; for example, in classification, they correspond to the decision boundary [2,5].

In physical networks, physical parameters—such as the stiffness or presence of springs connecting nodes in mechanical networks [6–10], or conductances in resistor networks [11]—can be tuned to achieve a desired physical response. In particular, contrastive local learning networks [12,13] use a local rule [14] rather than gradient descent to adjust their conductances to reach minima in their cost landscapes, enabling them to perform supervised learning without using a processor. Like artificial neural networks, trained physical networks have directions of high curvature in the cost landscape [15] that identify the key tunable resistors, or key edges, responsible for performing the tasks.

Unlike artificial neural networks, these physical networks are also described by a physical landscape. For electrical contrastive local learning networks, this is the dissipated power $\mathcal{P}$ as a function of all the node voltages, or "physical degrees of freedom," V_a. The response to applied voltages or currents is defined by the minimum of $\mathcal{P}$, which translates to Kirchhoff's current law [16]. The physical landscape for such systems is parabolic and thus fully described by the physical Hessian, the second derivative of the power with respect to the node voltages, $\mathbf{H}_{ab} = \partial^2 P/\partial V_a \partial V_b$.

Learning in electrical resistor networks therefore involves two optimizations. On one hand, the system naturally responds to applied currents or voltages by settling into a voltage configuration that minimizes the power defined by the edge conductances. On the other hand, the system adjusts the edge conductances, modifying the power landscape and response, to minimize the cost

[*]These authors contributed equally to this work.

function. This double optimization leaves strong signatures in the physical landscape for generic Hopfield [17], elastic [17–21], flow, and resistor networks [17,22]. These signatures include low-dimensional physical responses and alignment of low-lying physical eigenmodes with the functional response. A connection between the two landscapes was recently studied in Ref. [15], showing that the stiff eigenmodes of the cost Hessian are related to the soft eigenmodes of the physical Hessian. In particular, for a simple task in which an applied voltage drop across one edge leads to an equal voltage drop across another edge, the lowest physical mode and the highest cost mode contain the same information [15]. In this case, the lowest eigenmode of the physical Hessian $\mathbf{H}$ gives us the key edges, namely, the edges that are most responsible for performing the task. Even though this relation quickly breaks down for more complex tasks, this simple example conveys a new possibility: Information about the learned solution can potentially be accessed by the physics of the system.

In this article, we introduce a new structure-function relation for any trained system that minimizes a physical Lyapunov function. We show how the function—the learned input-output relation encoded in the cost landscape and captured by the stiff modes of the cost Hessian—depends on the structure—the edge conductances, geometry, and connectivity of the network. We formally connect the cost Hessian and the stiff modes of fully trained systems to a new physical observable: the tuning susceptibility. Importantly, the tuning susceptibility can be directly measured without any knowledge of the learned tasks, elucidating precisely how much information about the cost Hessian can be extracted solely from the physical Hessian. Our results provide a general way of identifying the key edges from structural properties of the networks alone. We demonstrate how this identification can be used to gain physical insight into the inner workings of trained networks—something that is not possible in artificial neural networks.

Our theoretical results are general to any physical network that develops function from a double optimization process, including mechanical networks that lie at a minimum of the energy or free energy.

II. SYSTEMS STUDIED

We frame our analysis in the language of tunable resistor networks of N nodes and M edges, in which each edge corresponds to a linear resistor with adjustable conductance. Each node a in the network is characterized by a voltage V_a with respect to a ground. Each edge i connecting two nodes a and b carries a current proportional to the voltage difference, $I_i = k_i(V_a - V_b)$, with k_i being the conductance of the edge. Throughout the article, we reserve indices $\{a, b\}$ to indicate nodes and $\{i, j\}$ to indicate edges. We use vectors to describe voltage states $\mathbf{V} = (V_1, V_2, \ldots, V_N)$, conductance states $\boldsymbol{k} = (k_1, k_2, \ldots, k_M)$, and current states $\mathbf{I} = (I_1, I_2, \ldots, I_M)$.

For the numerical analyses, we use tunable resistor networks of size $N = 300$ nodes obtained from jammed packings, as in previous studies [11,23], although our results hold for any type of network architecture. Inspired by real laboratory implementations, we constrain edge conductances to lie within a finite but wide range of $[10^{-6}, 10^{6}]$. We train the networks using coupled learning [11] but emphasize that our theoretical results are independent of the training process. The main assumption is that the cost is near the global minimum value of zero, so the task has been learned. For training details, initial conditions, and hyperparameters of the different examples shown in the paper, see Appendix F.

III. RELATION BETWEEN THE COST AND PHYSICAL HESSIANS

The following derivation holds for any trained task. However, it is useful to have in mind an illustrative example. As such an example, consider a network trained for a linear regression task, Fig. 1. The network has two source nodes, S_1 and S_2, and two target nodes, T_1 and T_2, satisfying the following relation:

$$\mathbf{V}^{\text{des}} \equiv \begin{pmatrix} V_{T_1} \\ V_{T_2} \end{pmatrix} = \begin{pmatrix} 0.15 & 0.3 \\ 0.2 & 0.1 \end{pmatrix} \begin{pmatrix} V_{S_1} \\ V_{S_2} \end{pmatrix}. \tag{1}$$

In addition, we hold a node at ground voltage [Fig. 1(a)], as in experiments, which removes the trivial zero mode of the physical landscape corresponding to uniform shifts of

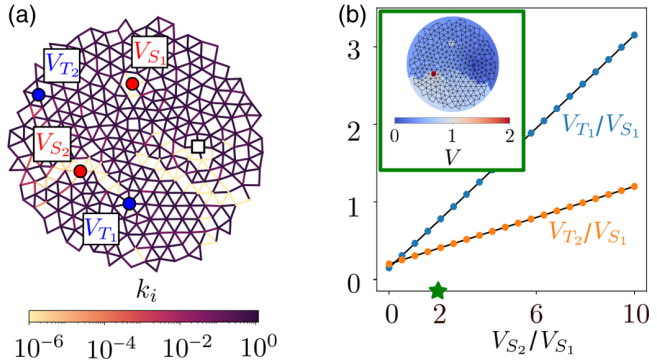

FIG. 1. Linear regression task in a resistor network. (a) Conductance configuration satisfying a linear regression task between its input (red) and output (blue) nodes. The network is grounded at the white node, fixing the zero voltage reference. (b) Target node voltages as a function of the source node voltages. To represent the relation in two dimensions, all variables are scaled by the first source voltage V_{S_1}. Circles correspond to the response of the trained circuit and solid lines to the desired relation. Inset: voltage response (color) when applying the inputs indicated by the green star: $V_{S_1} = 1$ and $V_{S_2} = 2$.

voltage on all the nodes. The physical Hessian for a grounded linear resistor network is given by

$$\mathbf{H} = \begin{pmatrix} 2\Delta K \Delta^T & \boldsymbol{q} \\ \boldsymbol{q}^T & 0 \end{pmatrix}, \tag{2}$$

where Δ is the incidence matrix of the arbitrarily oriented network, with components $\Delta_{a,j} = 1$ if edge j enters node a, -1 if it leaves, and 0 if not connected. Note that $K_{ij} = \delta_{ij} k_i$ is the diagonal matrix of conductances, and $q_a = \delta_{a,a'}$ is a canonical vector with entry 1 at the fixed ground node a'. This extended Hessian or bordered Laplacian [15,23] removes the trivial zero mode of the network, allowing the physical Hessian to be inverted, $\mathbf{H}^{-1}$. Under this formulation, the voltage configuration is extended with the current (Lagrange multiplier) λ needed to ground the system, $\mathbf{V} = (V_1, V_2, \ldots, V_N, \lambda)$. The extra zero at the bottom-right corner of $\mathbf{H}$ ensures that the constrained optimization is linear in λ.

Additional external input currents associated, e.g., with training samples, can, in general, be encoded into a current vector of the form $\mathbf{I} = (I_1, I_2, \ldots, I_N, 0)$, where the last entry corresponds to the value of the ground voltage. The system's voltage response $\mathbf{V}^F$ to all of the applied currents must minimize the dissipated power $\mathcal{P}$,

$$\mathcal{P} = \frac{1}{2}\mathbf{V}^T\mathbf{H}\mathbf{V} - \mathbf{I}^T\mathbf{V}, \tag{3}$$

leading to

$$\mathbf{V}^F = \mathbf{H}^{-1}\mathbf{I}. \tag{4}$$

Supervised learning corresponds to modifying the response $\mathbf{V}_r^F$ to a given set of inputs $\mathbf{I}_r$ to satisfy constraints $c_r(\mathbf{V}_r^F) = 0$, with r going from 0 to R, the total number of constraints. Usually, these constraints act on specific nodes designated as output voltage nodes. Denoting by $\mathbf{Q}_r$ the projector onto an output node—essentially a canonical vector with entry 1 at the position of the output node—each constraint can be written

$$c_r(\mathbf{V}_r^F) = \mathbf{Q}_r^T\mathbf{V}_r^F - V_r^{\mathrm{des}}, \tag{5}$$

where V_r^{des} is the desired output value for the inputs $\mathbf{I}_r$.

The cost C is then naturally defined as the sum of the squared constraints over the R constraints, or samples, for which the network is being trained, corresponding to the mean squared error (MSE):

$$C = \frac{1}{2}\sum_{r=1}^{R} c_r^2. \tag{6}$$

Under the assumption that all the constraints are satisfied at the end of training, $c_r = 0,\ \forall\, r$, the cost Hessian reads [15]

$$\mathcal{H}_{ij} = \frac{\partial^2 C}{\partial k_i \partial k_j} = \sum_{r=1}^{R} \frac{\partial c_r}{\partial k_i}\frac{\partial c_r}{\partial k_j}. \tag{7}$$

Using Eqs. (4) and (5), we carry out the derivatives of the constraints over the voltage response,

$$\frac{\partial c_r}{\partial k_i} = \mathbf{Q}_r^T \frac{\partial \mathbf{V}_r^F}{\partial k_i} = \mathbf{Q}_r^T \frac{\partial \mathbf{H}^{-1}}{\partial k_i}\mathbf{I}_r, \tag{8}$$

leading to

$$\mathcal{H}_{ij} = \sum_{r=1}^{R}\left(\mathbf{Q}_r^T \frac{\partial \mathbf{H}^{-1}}{\partial k_i}\mathbf{I}_r\right)\left(\mathbf{Q}_r^T \frac{\partial \mathbf{H}^{-1}}{\partial k_j}\mathbf{I}_r\right). \tag{9}$$

We can explicitly separate physical and training contributions of the cost Hessian by employing higher-order tensors. Denoting the outer product by $\otimes$, we define the fourth-rank training tensor $\mathbf{L}$ and the fourth-rank tuning susceptibility tensor $\mathbf{S}_{ij}$ as

$$\mathbf{L} = \sum_{r=1}^{R} \mathbf{Q}_r \otimes \mathbf{I}_r \otimes \mathbf{Q}_r \otimes \mathbf{I}_r, \tag{10}$$

$$\mathbf{S}_{ij} = \frac{\partial \mathbf{H}^{-1}}{\partial k_i} \otimes \frac{\partial \mathbf{H}^{-1}}{\partial k_j}, \tag{11}$$

from which we can express the cost Hessian as the full contraction of these last two tensors:

$$\mathcal{H}_{ij} = \mathbf{L} \vdots \mathbf{S}_{ij} = \sum_{a,b,c,d} \mathbf{L}_{abcd}\mathbf{S}_{ij,abcd}. \tag{12}$$

Equation (12) shows that the cost Hessian depends on two objects of very different nature. The training tensor contains only training data: the location of the outputs ($\mathbf{Q}_r$), and the location of the inputs together with their training values ($\mathbf{I}_r$) for each training sample r. Importantly, it does not depend on the physical properties of the network—$\mathbf{L}$ does not change with training. Conversely, the tuning susceptibility tensor contains only physical properties and is defined regardless of the training data. However, its entries are modified by training. Its components measure the sensitivity of the inverse of the physical Hessian (the response) to changes in conductances. We highlight that while the derivation of Eq. (12) assumed the MSE as the cost function, Eq. (6), this result is valid for any form of cost provided it is differentiable and has a minimum with $C = 0$ (see Appendix B).

The tuning susceptibility $\mathbf{S}$ and the cost Hessian $\mathcal{H}$ underpin a deeper structure-function relation. The desired input-output relation, the function, is characterized by the cost Hessian and its eigenvalue-eigenvector pairs. The structure—the learned values of conductances and the network topology—relates to the function through the

tuning susceptibility $\mathbf{S}$. In what follows, we provide a simpler analytical expression of the tuning susceptibility tensor and link it to the eigenvectors of the cost Hessian and the linear response of the system. Hereafter, we may refer to $\mathbf{S}$ simply as the susceptibility; however, it should not be confused with the similarly named linear response function, Green's function, or transfer function.

IV. TUNING SUSCEPTIBILITY TENSOR

Using Eq. (2) and the derivative of the matrix inverse, we explicitly compute the derivatives of the inverse physical Hessian appearing in Eq. (11):

$$\frac{\partial \mathbf{H}^{-1}}{\partial k_i} = -\mathbf{H}^{-1}\frac{\partial \mathbf{H}}{\partial k_i}\mathbf{H}^{-1} = -\mathbf{H}^{-1}\begin{pmatrix} 2\Delta_i\Delta_i^T & \mathbf{0} \\ \mathbf{0}^T & 0 \end{pmatrix}\mathbf{H}^{-1}, \tag{13}$$

where Δ_i is the vector formed by the ith row of the incidence matrix Δ. For clarity, we define the extended vector (bold font) $\mathbf{\Delta}_i = (\Delta_i, 0)$ by adding an extra zero, and we define the tuning susceptibility vector $\mathbf{s}_i$ as

$$\mathbf{s}_i \equiv \mathbf{H}^{-1}\mathbf{\Delta}_i, \tag{14}$$

corresponding to a vector of dimensions $N+1$ defined per edge i. Under this notation, the derivative of the inverse physical Hessian reads

$$\frac{\partial \mathbf{H}^{-1}}{\partial k_i} = -2\mathbf{s}_i \otimes \mathbf{s}_i, \tag{15}$$

yielding the final expression for the susceptibility tensor:

$$\mathbf{S}_{ij} = 4\mathbf{s}_i \otimes \mathbf{s}_i \otimes \mathbf{s}_j \otimes \mathbf{s}_j. \tag{16}$$

This last expression shows the relation between both Hessians, in agreement with the spectral analysis of trained networks in Ref. [15]. Equation (14) explicitly shows that the susceptibility vector scales as the inverse of the physical Hessian, $\mathbf{s}_i \sim \mathbf{H}^{-1}$. Thus, the susceptibility $\mathbf{S}_{ij}$, and therefore the cost Hessian $\mathcal{H}$, contains four inverse powers of the physical Hessian. Without resorting to the susceptibility, Ref. [15] showed that, in the case of single tasks ($R = 1$), the nonzero eigenvalue of the cost Hessian μ_C scales with the softest eigenvalue of the physical Hessian $\mu_{\mathcal{P}}$, as $\mu_C \sim \mu_{\mathcal{P}}^{-4}$, and the respective eigenvectors are approximately proportional, $\Psi_C \sim (\Delta\Psi_{\mathcal{P}})^2$. This spectral relation, however, does not hold, in general, for more complex tasks but corresponds to a specific case of the structure-function relation of Eq. (12).

While each entry $\mathbf{S}_{ij}$ is a fourth-rank tensor, rendering $\mathbf{S}$ a sixth-order tensor, its components are all determined by the susceptibility vectors $\mathbf{s}_i$ defined per edge. Moreover, the susceptibility vectors are dense as opposed to the sparse nature of the input currents $\mathbf{I}_r$ and the projectors $\mathbf{Q}_r$, Fig. 2.

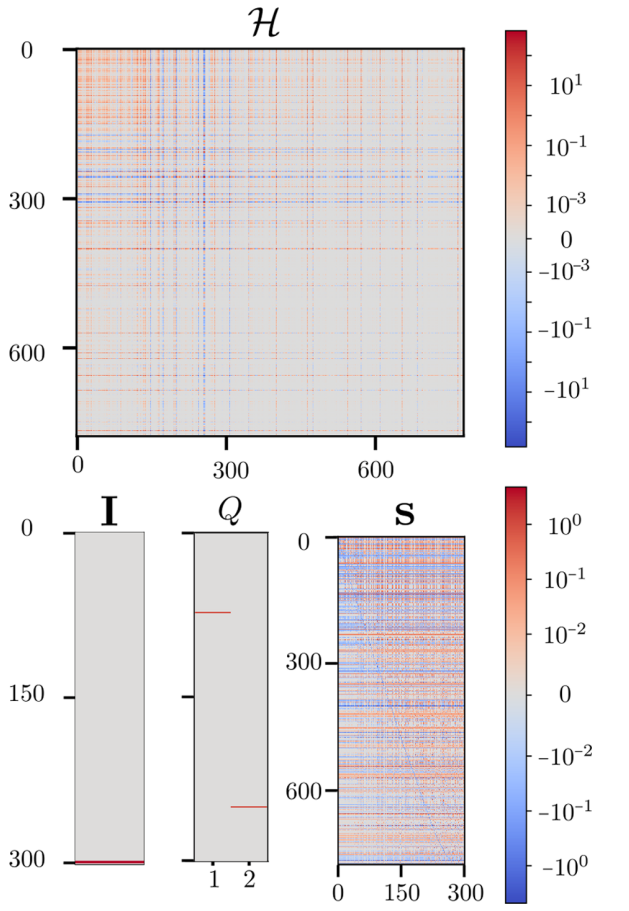

FIG. 2. Learning and physical contributions to the cost Hessian. We show entries of the cost Hessian $\mathcal{H}$, input current $\mathbf{I}$, output projector $\mathbf{Q}$, and tuning susceptibility matrix $\mathbf{s}$ with elements $s_{i,\alpha}$ for the trained network shown in Fig. 1.

In other words, the larger the susceptibility $\mathbf{s}_i$, the larger the entry $\mathbf{S}_{ij}$, which, on average and depending on the training tensor $\mathbf{L}$, tends to manifest as large entries of the cost Hessian $\mathcal{H}_{ij}$.

The definition of $\mathbf{s}_i$ is directly related to the response of the system. Contracting it with an input current $\mathbf{I}$ leads to a voltage drop at edge i of the physical response to the current $\mathbf{I}$,

$$\mathbf{s}_i^T\mathbf{I} = \mathbf{\Delta}_i^T(\mathbf{H}^{-1}\mathbf{I}). \tag{17}$$

Thus, each element $\mathbf{s}_{i,a}$ is the voltage drop at edge i of the response to a unit current at node a, Fig. 3. The magnitude of $\mathbf{s}_i$ quantifies the voltage drop linked to all possible unit currents in the network and assigns a scalar measure of susceptibility per edge:

$$\|\mathbf{s}_i\|^2 = \sum_a \|\mathbf{s}_{i,a}\|^2. \tag{18}$$

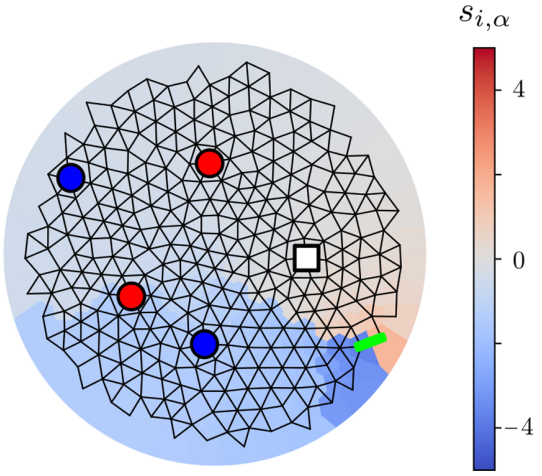

FIG. 3. Tuning susceptibility vector. Example of tuning susceptibility vector $\mathbf{s}_i$ for the highlighted edge of the trained network of Fig. 1. In general, the closer the node α is to the edge, the larger the entry of $\mathbf{s}_{i,\alpha}$.

Denoting by ν_α and $\mathbf{W}_\alpha$ the eigenvalues and normalized eigenvectors of the physical Hessian, $\mathbf{H}$, we can recast the susceptibility as

$$\mathbf{s}_i = \sum_\alpha \frac{1}{\nu_\alpha}(\mathbf{\Delta}_i^T \mathbf{W}_\alpha)\mathbf{W}_\alpha^T, \tag{19}$$

with norm

$$\|\mathbf{s}_i\|^2 = \sum_\alpha \frac{1}{\nu_\alpha^2}(\mathbf{\Delta}_i^T \mathbf{W}_\alpha)^2, \tag{20}$$

where we have used the orthonormality of the eigenvectors $\mathbf{W}_\alpha$. Equation (20) shows that the magnitude of the susceptibility $\mathbf{s}_i$ is dominated by soft modes (small ν_α) with large voltage drops across the edge i.

The tuning susceptibility tensor, codifying the structure of the system, is thus redundantly determined by the tuning susceptibility vectors $\mathbf{s}_i$, which, in turn, are measurable properties corresponding to the voltage drop response to unit currents across all possible nodes. In the next section, we deepen the structure-function relation of Eq. (12) and show the general relation of the cost eigenvectors with the tuning susceptibility.

V. RELATION BETWEEN THE STIFF COST EIGENVECTORS AND THE TUNING SUSCEPTIBILITY

We denote by μ_ρ and Ψ_ρ the nonzero eigenvalue-eigenvector pairs of $\mathcal{H}$. From Eq. (7), we first notice that any vector $\boldsymbol{v}$ perpendicular to the space spanned by the gradients of the constraints, i.e., $\sum_i \boldsymbol{v}_i \partial_{k_i} c_r = 0 \ \forall\, r$, is an eigenvector of zero eigenvalue (curvature). Therefore, Ψ_ρ is a linear combination of the $R' \leq R$ independent gradients of constraints:

$$\Psi_{\rho,i} = \sum_r^{R'} \alpha_r^\rho \frac{\partial c_r}{\partial k_i}. \tag{21}$$

Inserting this expression into $\mathcal{H}\Psi_\rho = \mu_\rho \Psi_\rho$ and exploiting the linear independence of the R' gradients, we obtain a reduced eigenvalue problem for the linear coefficients:

$$\sum_r \mathbf{G}_{q,r}\alpha_r^\rho = \mu_\rho \alpha_q^\rho, \tag{22}$$

where $\mathbf{G}_{q,r}$ is the Gram matrix of the independent gradients, defined as

$$\mathbf{G}_{q,r} = \sum_i \frac{\partial c_q}{\partial k_i}\frac{\partial c_r}{\partial k_i} = (\mathbf{Q}_q \otimes \mathbf{I}_q \otimes \mathbf{Q}_r \otimes \mathbf{I}_r) \vdots \sum_i \mathbf{S}_{ii}. \tag{23}$$

In the last equality, we used Eqs. (5), (15), and (16), showing that the Gram matrix and its eigenvectors α_r^ρ depend on all the susceptibilities (sum over all indices i).

Using Eqs. (5), (15), and (21), we rewrite the stiff eigenvectors in terms of the vector susceptibilities as

$$\Psi_{\rho,i} = \mathbf{s}_i^T A_\rho \mathbf{s}_i = \|\mathbf{s}_i\|^2 \left(\frac{\mathbf{s}_i^T}{\|\mathbf{s}_i\|^2} A_\rho \frac{\mathbf{s}_i}{\|\mathbf{s}_i\|^2} \right), \tag{24}$$

where A_ρ is a matrix defined as

$$A_\rho \equiv -2 \sum_r^{R'} \alpha_r^\rho \mathbf{Q}_r \otimes \mathbf{I}_r. \tag{25}$$

Equation (24) is another manifestation of the structure-function relationship shown in Eq. (12). We first notice that A_ρ is the same for all entries i, implying that the distinction between them is ultimately determined by the susceptibilities. It also shows that large entries of the stiff modes—the key edges—can arise by two nonexclusive mechanisms: first, because of the large magnitudes of the susceptibility $\|\mathbf{s}_i\|^2$, which is measurable and can be computed without any knowledge of the training data and input-output nodes; second, because of the contraction of the matrix A_ρ with the normalized susceptibility vectors $\mathbf{s}_i/\|\mathbf{s}_i\|^2$. This second contribution, however, explicitly depends on the training data through A_ρ. In the particular case of a single task, A_ρ is a scalar and Eq. (24) shows that the stiff mode is the norm squared of the susceptibility vector.

In summary, we have demonstrated that all the physical information of the cost Hessian boils down to the dense tuning susceptibility tensor $\mathbf{S}$ and, ultimately, to the tuning susceptibility vectors $\mathbf{s}_i$, encoding the structure of the system. The susceptibilities are dominated by soft modes of the physical Hessian with large voltage drops, and they correlate with the cost Hessian.

These results, although written in the language of resistor networks, are far more general. There are only three necessary conditions to derive Eqs. (12), (16), and (24):

(1) The tunable degrees of freedom satisfy all the constraints, and therefore $C = 0$ [Eq. (6)]. In other words, the system has been trained successfully.
(2) The physical response to externally applied inputs $\mathbf{I}$ minimizes a scalar function $\mathcal{P}$ with respect to the physical degrees of freedom, $\mathbf{V}^F = \min_{\mathbf{V}} \mathcal{P}(\mathbf{V}, \mathbf{I})$. In other words, the physical system optimizes a Lyapunov function.
(3) The response $\mathbf{V}^F$ to inputs $\mathbf{I}$ is approximated by the perturbations around a known local minimum $\mathbf{V}_{\min}$ of $\mathcal{P}$, $\mathbf{V}^F \approx \mathbf{V}_{\min} + \delta\mathbf{V}$.

In low-Reynolds-number flow networks, the tunable degrees of freedom may correspond to pipe diameters, the physical degrees of freedom, $\mathbf{V}$, to node pressures, the input data to flow current, and the scalar function $\mathcal{P}$ to the power dissipated. In elastic networks, the tunable degrees of freedom may correspond to the stiffness or rest lengths of the springs, the physical degrees of freedom to the node positions or displacements, the input data to applied forces on nodes, and the physical landscape $\mathcal{P}$ to the elastic or free energy [6,11,24,25]. For mechanical networks, however, special care must be taken due to the vectorial nature of the physical degrees of freedom, which leads to a slight modification of the tuning susceptibility in terms of compatibility and rigidity matrices (see Appendix C).

In practice, condition (1) can be relaxed to trained systems with small nonzero cost, as we will show numerically in the next sections. If the constraints are unsatisfiable, learning will reach a minimum with finite cost C due to some nonzero constraints c_r. In such cases, the cost Hessian has an extra term due to the Hessian of single constraints:

$$\mathcal{H}_{ij} = \sum_{r=1}^{R} \frac{\partial c_r}{\partial k_i}\frac{\partial c_r}{\partial k_j} + \sum_{r=1}^{R} c_r \frac{\partial^2 c_r}{\partial k_i \partial k_j}. \tag{26}$$

As shown in Appendix A, the Hessian of the constraints can also be split into training and physical variables, the latter being the product of susceptibilities,

$$\frac{\partial^2 c_r}{\partial k_i \partial k_j} = 4\mathbf{Q}_r \otimes \mathbf{I}_r : [(\mathbf{\Delta}_i^T \mathbf{s}_j)(\mathbf{s}_i \otimes \mathbf{s}_j + \mathbf{s}_j \otimes \mathbf{s}_i)]. \tag{27}$$

In the regime of small constraints $c_r \ll 1$, however, this extra term becomes a small perturbation, and the previous analysis holds to first order, in particular, Eq. (24).

Conditions (2) and (3) allow for a generic perturbative description of nonlinear physical landscapes in terms of the closest local minimum $\mathbf{V}_{\min}$ to the response $\mathbf{V}^F$. In such cases, Eq. (3) becomes $\mathcal{P} = P(\mathbf{V}) - \mathbf{I}^T\mathbf{V}$, with P a nonlinear function of the physical degrees of freedom, $\mathbf{V}$. To second order,

$$\mathcal{P}(\mathbf{V}_{\min} + \delta\mathbf{V}) \approx \mathcal{P}(\mathbf{V}_{\min}) + \frac{1}{2}\delta\mathbf{V}^T\mathbf{H}\delta\mathbf{V} - \mathbf{I}^T\delta\mathbf{V}, \tag{28}$$

where the physical Hessian is given by $\mathbf{H}_{ab} = \partial^2 P(\mathbf{V}_{\min})/\partial V_a \partial V_b$. Then, the response is given by $\mathbf{V}^F = \mathbf{V}_{\min} + \mathbf{H}^{-1}\mathbf{I}$, and the same derivations ensue from Eq. (8).

Sections III–V establish the connection between the tuning susceptibilities (structure) and the cost landscape around a learned minimum through its cost Hessian and eigenvectors (function). The tuning susceptibility, defined without any explicit reference to the training information, is nevertheless implicitly shaped by the learning process. In the next sections, we will explore this aspect by studying how much the stiff cost eigenvectors, and their large entries, are predicted by the structure alone, the tuning susceptibility. We ignore the A_ρ matrix from Eq. (24) and instead focus on the normalized norm of the susceptibilities $\hat{\mathbf{s}}_i = \|\mathbf{s}_i\|^2 / \sum_j \|\mathbf{s}_j\|^2$ since the overall scale is case dependent and does not affect the analysis. We study three different learning tasks, previously implemented in experimental networks [12,13], and show that the large entries of the stiff modes are highly correlated with the largest susceptibilities. These numerical results establish that, although the tuning susceptibility does not capture all of the information about the function, it captures a significant portion of it. Finally, we connect the tuning susceptibility to the low-dimensional response observed in trained networks [17] and the topological nature of the response in allosteric networks [26], and we study the physical behavior of highly susceptible edges.

VI. LARGE ENTRIES OF STIFF MODES CORRELATE WITH HIGH SUSCEPTIBILITY

Figure 4(a) shows the distribution of cost eigenvalues μ_ρ before and after training for the linear regression task of Fig. 1. Before training, the network has a wide distribution of cost eigenvalues and a nonzero cost gradient. After training, the spectrum is highly degenerate. Most of the eigenvalues are close to zero while four are very high. The same behavior has been observed in artificial neural networks [2,3]. The modes with high eigenvalues, or stiff modes, correspond to the number of coefficients learned by the linear regression task (details in Fig. 1). The entries of each of the stiff modes $\mathbf{\Psi}_\rho$ measure the importance of the given edge to the function [Fig. 4(b)]. Perturbing any of those edges amounts to moving in conductance space along, at least partially, directions of high curvature, substantially increasing the cost C. Remarkably, the scalar field of the norm of susceptibilities shown in Fig. 4(c) reveals the same patterns as we see in the stiff modes, Fig. 4(b). In Fig. 4(d), we quantify this observation with a correlation scatter plot over the edges of the network, showing that high tuning susceptibility strongly correlates with large entries of the stiff modes.

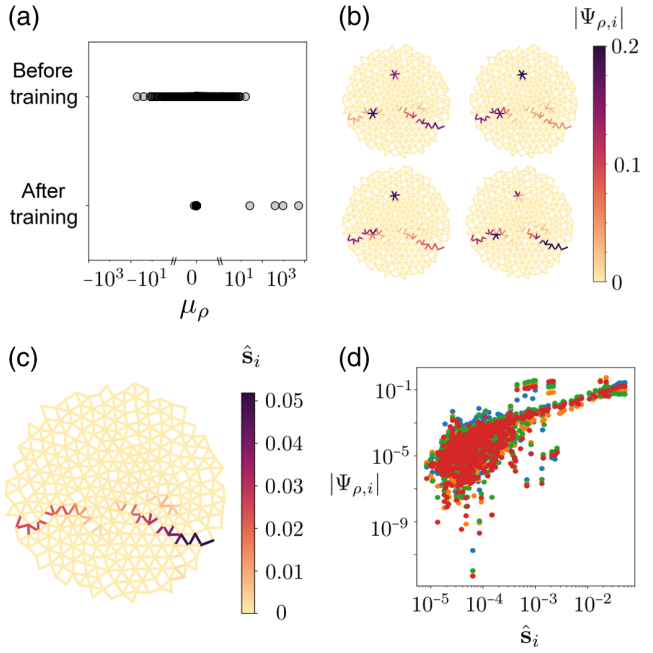

FIG. 4. Stiff modes and tuning susceptibility norm. (a) Eigenvalues of the cost Hessian, μ_ρ, before and after training. The spectrum has a spectral gap spanning more than 2 orders of magnitude, separating four stiff outliers from the remaining almost-zero modes. Parallel lines indicate the break from linear to logarithmic scale. (b) Four associated eigenvectors depicted as scalar fields over the networks. Dark colors highlight the conductances that are most important to the function. (c) Scalar field $\hat{\mathbf{s}}_i$ capturing the main structure revealed by stiff modes of the cost Hessian in panel (b). (d) Scatter plot of the absolute values of the entries of each of the stiff modes $|\Psi_{\rho,i}|$ and susceptibilities $\hat{\mathbf{s}}_i$ for each edge in the network. Different colors indicate different stiff modes.

We illustrate the correlation between edge susceptibility and stiff modes of the cost Hessian for three more tasks. First, we consider a more complicated linear regression task, consisting of two input and five output nodes. The vector of desired voltages as a function of input voltages is given by

$$\mathbf{V}^{\text{des}} = \begin{pmatrix} 0.15 & 0.3 \\ 0.2 & 0.1 \\ 0.05 & 0.2 \\ 0.1 & 0.05 \\ 0.3 & 0.15 \end{pmatrix} \begin{pmatrix} V_{S_1} \\ V_{S_2} \end{pmatrix}, \tag{29}$$

where, as before, a ground node is kept at zero voltage, making all the linear relations noncolinear.

The network is trained down to a cost of $C \approx 10^{-9}$, ending up with a set of conductances that satisfy the task, Figs. 5(a) and 5(b). As with the previous linear regression task, high tuning susceptibility edges positively correlate with those singled out from stiff modes, Figs. 5(c) and 5(d).

A more stringent test corresponds to linear classification. Here, we train the network to classify three different classes depending on two input voltages, Figs. 5(e) and 5(f). We encode the corresponding classes with three output nodes. To convert classification into a numerical task, we use "one-hot encoding" for each output node. For example, a point belonging to class 2 corresponds to $\mathbf{V}^{\text{des}} = (0, 1, 0)$. For this case, a more suitable cost function corresponds to the cosine similarity:

$$\mathcal{C} = \sum_r (1 - \cos(\mathbf{V}_r^F, \mathbf{V}^{\text{des}})), \tag{30}$$

where $\cos(\boldsymbol{V}_T, \boldsymbol{V}_T^{\text{des}})$ corresponds to the cosine of the angle between the two vectors.

This task is significantly harder for the network to achieve for three reasons. First, the physical restriction of having positive conductances only allows for decision boundaries with positive slope. To overcome this restriction, we represent each input value as two different and opposite voltage nodes, Fig. 5(e). We do the same for the ground nodes, placing one low (-1) and one high $(+1)$ ground. Second, the two-dimensional embedding of the network topologically constrains current flows, leading to output nodes being effectively isolated from input nodes. We solve this problem by doubling the information and copying each input node twice, as well as the low and high grounds. Third, achieving perfect one-hot encoding is impossible for a linear resistor network since it corresponds to a highly nonlinear output as a function of the input voltages. Nevertheless, we can still train and interpret the result using a winner-take-all strategy, in which the network classifies according to the maximum target value, $\max(V_{T_1}, V_{T_2}, V_{T_3})$. Figure 5(f) shows that the network trains successfully, finding a set of edge conductances that lead to a classification accuracy of 99%. Even though the training and task are more involved, large values of $\hat{\mathbf{s}}_i$ indicate large entries of the stiff modes, as shown in Figs. 5(g) and 5(h).

A. Connection to persistent homology

As a last family of examples, we analyze networks trained to provide specified voltage drops across output edges in response to a voltage drop across an input edge [23] [Fig. 6(a)]. Such long-range effects require precise values of the conductances in resistor or flow networks [23]. Compared to the previous regression and classification tasks, this task is significantly simpler due to the fixed input values it is trained on. One of the remarkable results for networks trained for such tasks is that the response is topological in structure [26–28]. Regardless of the values the network is trained on, the response of the system tends to partition into sectors of roughly homogeneous voltages, captured by a topological data analysis approach known as persistent homology, as shown in Fig. 6(a) [26,27]. It turns

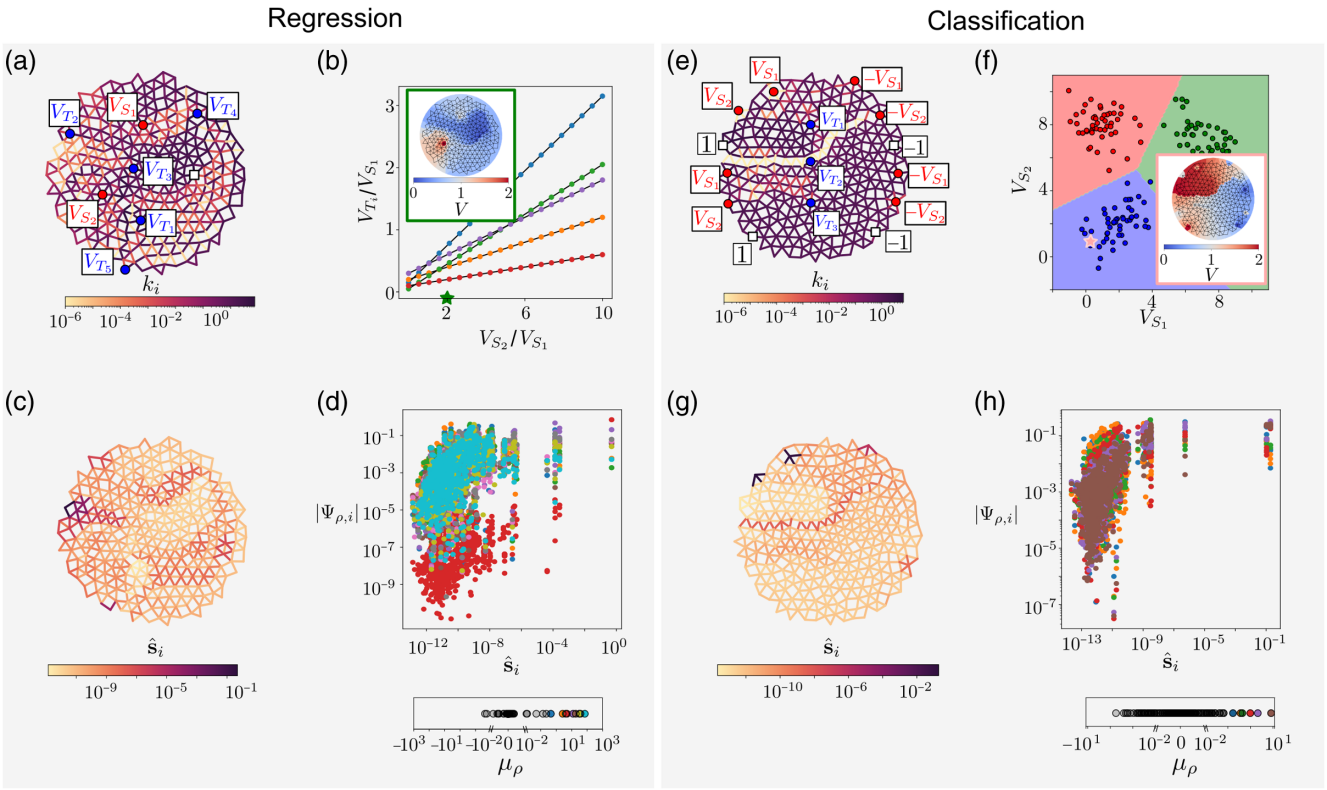

FIG. 5. Stiff modes and tuning susceptibility for linear regression and classification examples. Regression: (a) Conductance configuration of a network trained for the five target linear regression tasks specified by the coefficients in Eq. (29). Source nodes are denoted by red circles, target nodes by blue circles, and the grounded node by a white square. (b) Ratio of the voltage at each of the target nodes V_{T_i} with respect to the voltage source V_{S_1} (circles), which obeys the required linear dependence (solid lines). Inset: voltage response associated to the green star ($V_{S_2} = 2$ and $V_{S_1} = 1$). (c) Scalar field of $\hat{\mathbf{s}}_i$, with darker colors indicating larger susceptibilities. (d) Edges of higher tuning susceptibility identifying large entries of the ten stiffest modes of the network (bottom panel). Classification: (e), (g), and (h) with the same information as (a), (c), and (d), respectively, for a network trained to classify three classes of data as shown in panel (f), where the inset corresponds to the voltage response to the inputs indicated by the pink star.

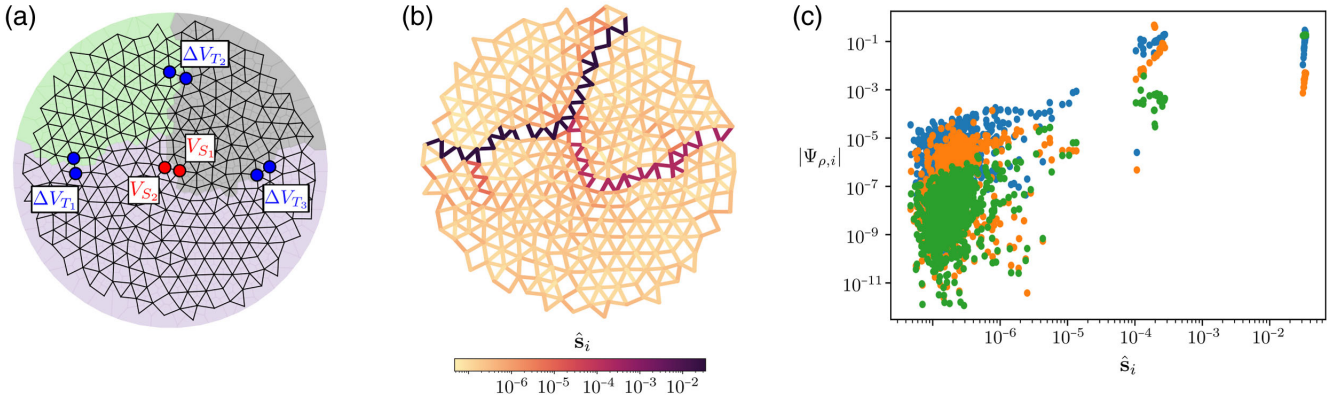

FIG. 6. Susceptible edges define topological sectors. (a) Network trained for an allosteric task in which pairs of target nodes (blue) must have a voltage drop of 0.5, when the voltages at the input nodes (red) are 0 and 1. The background color corresponds to the different topological sectors found by the persistent homology analysis of the voltage response (for details, see Ref. [27]). (b) Without the information of input and output nodes and values, the susceptibility field explicitly captures the boundaries between the persistent sectors. (c) For large values, the tuning susceptibility positively correlates with the entries of the three stiffest modes (different colors).

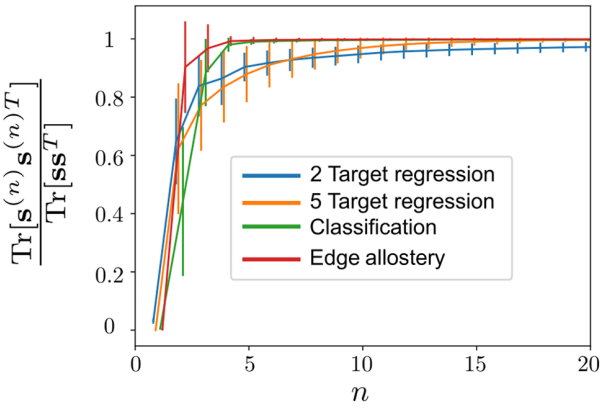

FIG. 7. Tuning susceptibility and dependence on number of modes. Ratio of partial to full tuning susceptibility as a function of number of modes for the four kinds of tasks treated in the main text. Error bars correspond to standard deviations over ensembles of 50 different initial conditions for each case. The curves are slightly shifted horizontally with respect to each other to distinguish the different error bars.

out that the boundaries of the topological sectors are precisely the edges with large tuning susceptibility [Fig. 6(b)]. They also correspond to edges singled out by stiff modes of the cost Hessian [Fig. 6(c)]. In Appendix D, we show that the tuning susceptibility is even better at capturing the important edges for weak training signals.

VII. LOW-DIMENSIONAL RESPONSE

All the trained networks shown present a sloppy cost eigenvalue spectra: a handful of stiff eigenvalues, roughly logarithmically spaced, followed by a gap separating them from a large number of far smaller eigenvalues [29,30]. Here, we show that, in addition to the low dimensionality of the stiff subspace, the susceptibility is also low dimensional in the space of physical modes.

It has been empirically shown that physical systems such as Hopfield, flow/electrical, and mechanical networks develop learned physical responses defined by a few soft eigenmodes of the physical Hessian [17]. As we show next, this low dimensionality is present in the tuning susceptibility vector $\mathbf{s}_i$. From Eq. (20), we define the partial susceptibility, up to mode n, as

$$\mathbf{s}_i^{(n)} = \sum_{\alpha=1}^{n} \frac{1}{\nu_\alpha} (\boldsymbol{\Delta}_i^T \mathbf{W}_\alpha) \mathbf{W}_\alpha^T. \tag{31}$$

Clearly, we recover the full susceptibility when $n = N + 1$, i.e., the total number of modes. Figure 7 shows the ratio between the sum of the partial and full susceptibility norms, $\sum_i \|\mathbf{s}_i^{(n)}\|^2 / \sum_i \|\mathbf{s}_i\|^2 = \mathrm{Tr}[\mathbf{s}^{(n)}\mathbf{s}^{(n)T}]/\mathrm{Tr}[\mathbf{s}\mathbf{s}^T]$, as a function of the number of modes for all four cases studied in the paper. In agreement with the dimensionality reduction of trained responses, for all cases, the norm saturates at values ranging from $n = 2$ to $n = 10$ out of a total of 301 modes, showing that most of the physical information is encoded in a few soft modes.

VIII. PHYSICAL INTERPRETATION OF ROLE OF KEY EDGES

We have established that the tuning susceptibility, a purely structural quantity independent of any explicit training information, correlates with the large entries of the stiff modes of the cost Hessian. Highly susceptible edges of trained networks are therefore important for the functionality of the network. But how do highly susceptible edges affect the physical response? We next study the conductance values of the trained networks and analyze them in relation to their susceptibility values.

We consider the linear regression task of Fig. 1 with four sets of random initial conditions: uniformly distributed between 10^{-2} and 1 and sharper distributions peaked at 10^{-6}, 10^{-3}, and 10^{-1}. The results are shown in Fig. 8: Different initial configurations lead to qualitatively different final states, each of them performing exactly the same task. The existence of multiple good solutions is a signature of the overparametrization of the networks (first and second columns). With the exception of the first example (first row), conductance is, in general, not correlated with the tuning susceptibility (third column). This observation also holds for all the examples presented in the main text (see Appendix E). The fourth column shows the relation between the relative conductance $k_i/\langle k_i \rangle$ of each edge with its susceptibility $\hat{\mathbf{s}}_i$, where $\langle k_i \rangle$ is the average conductance of the adjacent edges. While there is no clear correlation between both quantities, one striking empirically observed feature is common across all the examples: High-susceptibility edges have low relative conductances $k_i/\langle k_i \rangle < 1$ (see Appendix E for more examples). We emphasize that this does not need to be the case. One can imagine a functional network relying on connected segments of very high conductance (pipelines) to accomplish the same task. The training procedure we use (Coupled Learning) does not appear, in general, to yield such solutions; how different training protocols might affect the nature of trained solutions is an interesting subject for further study.

In Fig. 9, we illustrate how edges with high tuning susceptibility and low relative conductance effectively act as current blockers, or barriers, for generic input voltages. We analyze the four tasks of the main text: linear regression with two targets and with five targets, classification, and trained voltage drops at output edges (different columns). The first row illustrates the voltage response (background color) for given input voltages and highly susceptible edges (gray scale). The current field, shown in the second row, shows how current flows from inputs to outputs in the

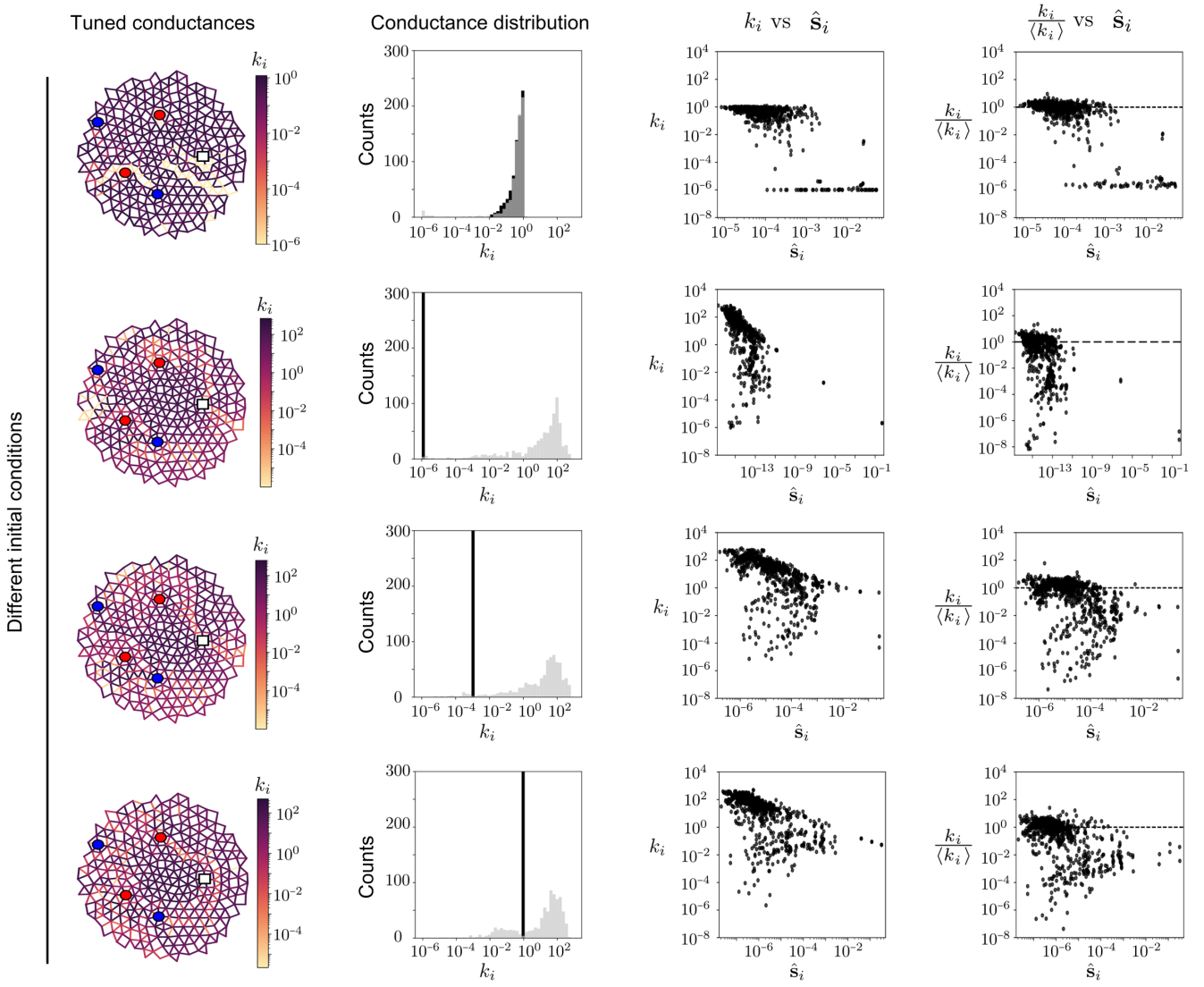

FIG. 8. Physical properties of four networks (rows) trained for the same task with different initial conditions. From left to right: conductance configuration of the trained networks; initial (light gray) and final (black) distribution of conductances; relation between final conductances and susceptibilities; relation between final relative conductances and susceptibilities (horizontal dashed line at 1).

network. Highly susceptible edges block the current along them, redirecting it by at least two identifiable mechanisms: acting as a wall to concentrate the voltage on either side (highlighted in green) or forming corridors through which the current is concentrated and routed to specific locations (highlighted in pink). Take, for example, the linear regression task for two targets (first row). A wall separates nodes T_2 from S_2 but not S_1. This strategy is sensible because T_2 is nearly equally distant from both S_1 and S_2 but was trained for a dependence on S_2 that is only half as strong as the dependence on S_1 [see Eq. (1)]. Similar conclusions can be drawn for the other tasks. These results show that the joint information of tuning susceptibility and conductance can provide physical insight into how the systems accomplish their tasks.

IX. DISCUSSION

We have shown several examples of resistor networks trained to different tasks. In each case, we have taken advantage of the property that trained resistor networks have their solutions imprinted into their physical landscapes [15,31] to describe the local geometry of the cost landscape. We have shown that the cost Hessian can be written as the tensor product of two quantities. The first is the training tensor $\mathbf{L}$, which depends solely on the task the system has been trained to perform: the location of input and output nodes, and the input currents corresponding to the training samples. The second quantity is the tuning susceptibility tensor $\mathbf{S}$, which depends solely on physical properties of the network, namely, its architecture and the edge conductances.

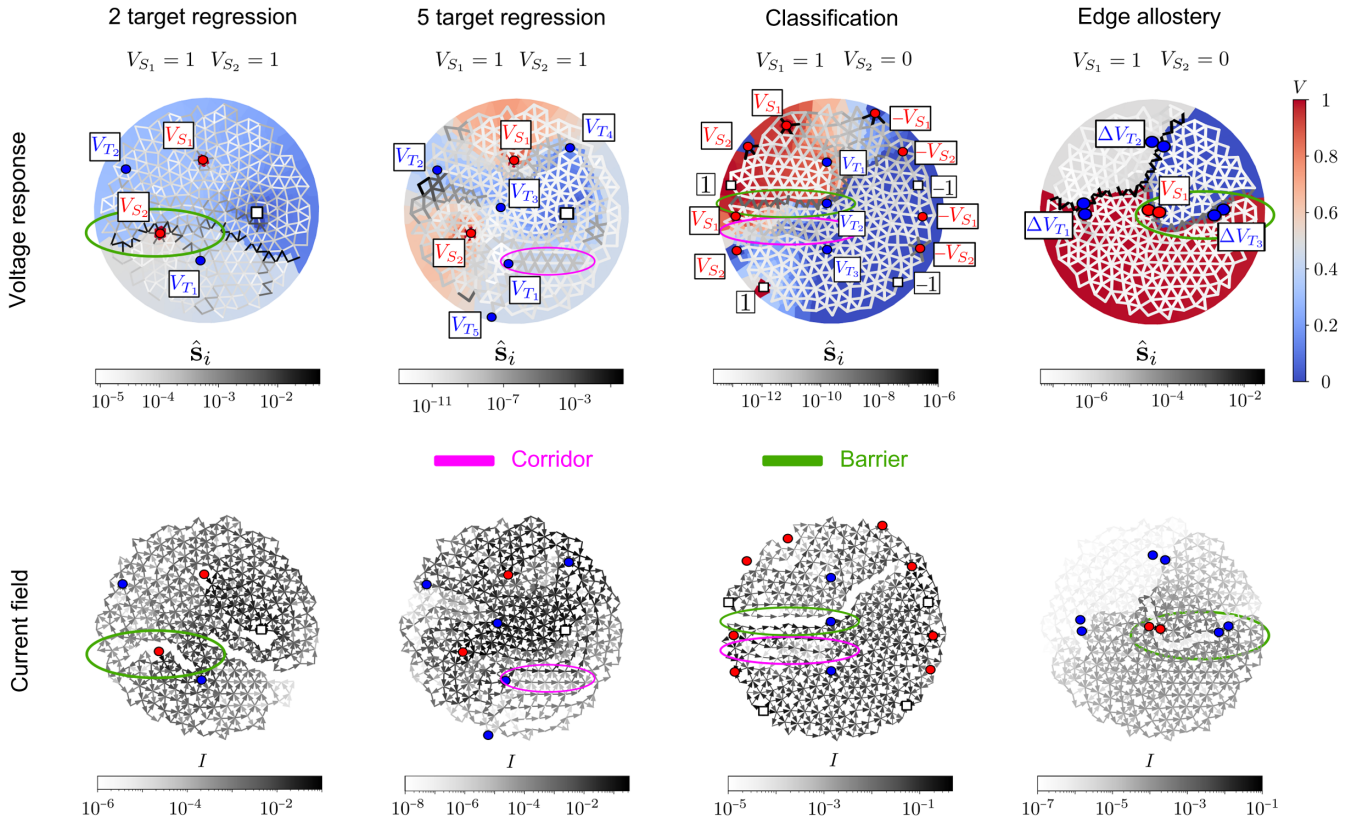

FIG. 9. Voltage and current responses of the four trained networks considered in the main text (columns). Top diagrams: voltage response (color) to the input values shown by the labels. The edges are colored according to their tuning susceptibility. Bottom diagrams: associated current response represented as a vector field with orientation indicated by the arrows and magnitude indicated by the gray scale. Across all examples, highly susceptible edges have low relative conductance and therefore block the current. We highlight two distinct functional mechanisms of the susceptible edges. We use green to highlight susceptible edges that block the current to concentrate the voltage and pink to highlight those that create a corridor to direct the current along it.

The explicit decoupling of training (task-dependent) and physical quantities has several consequences.

First, it reveals a fundamental structure-function relation in trained physical systems. We have formally related the tuning susceptibility, the structure, to the learned function. The function is characterized by the local geometry of the cost landscape around the learned minimum, in terms of the cost Hessian and its stiff modes. The key edges are those indicated by the large entries of the stiff modes of the cost Hessian and are the most important edges for performing the task. Here, we have demonstrated theoretically that they can also be identified from the tuning susceptibilities via the contraction of the susceptibilities with a matrix A_ρ depending on the training data. Finally, we have shown numerically that many of the key edges are characterized by large susceptibility norms, so they can be identified from physical properties of the network alone, without needing to access training information. The tuning susceptibilities are therefore microscopic physical imprints of the learned solutions.

Second, it establishes a fundamental bound on the physical information contained in the cost Hessian. The physical properties of linear resistor networks are encoded into the physical Hessian $\mathbf{H}$, which changes along the training process by softening, in general, its soft modes and aligning the physical response with the input forces [17]. Its relation to the cost Hessian has so far only been explored spectrally in simple tasks [15]. For a single task, it has been shown that the stiffest cost mode can be approximated by the softest physical mode, suggesting, for the first time, that learning information can be retrieved by the physics of the system. However, this spectral relation is only approximate. Here, we have shown that the tuning susceptibility is the physical-Hessian-based property that enters into the cost Hessian. For linear resistor networks, all of the structural information relevant for the trained tasks is encoded in the tuning susceptibilities. For nonlinear networks, as exemplified by elastic systems in Appendix C, the susceptibilities contain all the relevant information up to quadratic order in the physical landscape. The susceptibility tensor $\mathbf{S}$, which can be decomposed into susceptibility vectors $\mathbf{s}_i$, explains and unifies previous observations on physical learning such as the low-dimensional response [17], the

mode correspondence in simple tasks [15], and the topological structure of allosteric networks [26].

Third, it provides an experimentally measurable quantity independent of the training details. Measurements such as the response to localized input currents can be used to extract $\mathbf{s}_i$, revealing critical information about how the task is performed. While the mathematical formalism assumed perfectly trained systems (reaching zero cost), in practice, constraints may be unsatisfiable, and the cost Hessian has new contributions in terms of the susceptibilities (Appendix A). Nevertheless, as long as the cost is small, the new terms become perturbatively small, and the main results hold even for systems with nonzero cost. This case is shown in the numerical examples of Figs. 5(d) and 5(h), where the cost Hessian can have a wide distribution of eigenvalues, but large susceptibilities still reveal key edges. In addition, the definition of the susceptibilities are independent of the actual form of the cost function (Appendix B).

In short, the double optimization at the heart of physical learning in many systems makes it possible to gain physical insight into how trained tasks are accomplished. Such insight does not exist for artificial neural networks, which do not possess a physical landscape.

While all the derivations were explicitly carried out in the context of resistor networks, the same results naturally extend to elastic networks (Appendix C), where the response minimizes energy, and, in general, to any physical network that minimizes a scalar quantity (a Lyapunov function) and whose response can be approximated by a physical Hessian around a known minimum.

As an example, elastic networks have been trained to have responses similar to protein allostery, in which strain applied by binding one molecule affects the ability of a second type of molecule to bind to the protein elsewhere. Spring networks have been trained to develop allosteric responses using global gradient descent [6], Monte Carlo methods [19,32], or local learning rules such as directed aging [33] and coupled learning [11]. By contrast, real proteins have been trained to perform allostery via biological evolution. Because the protein remains in thermal equilibrium during evolution, it attains its function via a double-optimization process. The generality of our theoretical analysis implies that it may be applied independent of the process by which a system developed the ability to perform a task. This generality suggests that the connection [21] between slow physical modes of proteins and underlying nonlinearities that correlate nonpairwise additive effects of mutations of amino acids in the sequence (global epistasis) can be generalized to gain physical insight into global epistasis even in cases where allostery is not characterized by a single slow mode. Folded proteins have been modeled as mechanical spring networks [34]. Our theoretical analysis specifies how much of the function of an elastic network model of an allosteric protein can be captured by structure, as embodied in the susceptibility tensor.

Furthermore, our methods should yield potentially useful insight into how proteins perform allosteric tasks. Using persistent homology, a topological data analysis tool, it has been shown that the functional response of trained allosteric networks can often be described by robust macroscopic sectors highlighting large-scale structures in the network [28] akin to functional regions observed in allosteric proteins [35]. The persistent homology analysis depends on both training (input nodes and values) and physical (response) information. Here, we have shown that similar topological structures can be obtained with physical information alone, through the tuning susceptibility tensor, in a closely related system. An important direction for future work is to apply our analysis to mechanical networks with allostery to extract key edges and compare to persistent homology results. Our numerical results suggest that the most highly susceptible edges will be important for function. When applied to elastic network models of proteins, our analysis might allow identification of the key amino acids responsible for allosteric function.

More generally, our results pave the way towards understanding and interpreting how collective trained behavior emerges. Here, we have studied linear networks, for which the physical landscape is completely convex and possesses only one minimum. Biological, physical, and artificial systems, however, heavily rely on nonlinearities to achieve many complex tasks [8,32,36–38], which is the case, for example, of elastic networks with multistable allostery [32] and also resistor networks that learn nonlinear regression and classification [13,39]. For such systems, the physical landscape is decorated with several minima, with each minimum characterized by different tuning susceptibilities. Understanding how distributions and correlations among edge tuning susceptibilities might vary among minima is an important area of future work, akin to the study of statistical comparisons of the properties of different jammed minima in sphere packings [40,41].

ACKNOWLEDGMENTS

This work was supported by the U.S. Department of Energy Basic Energy Sciences through Grant No. DE-SC0020963 (M. G., F. M., and M. S.), the University of Pennsylvania National Science Foundation (NSF) NRT Grant No. DGE-2152205 (F. M.), and the Simons Foundation through Investigator Grant No. 327939 to A. J. L. M. S. was also supported by National Institutes of Health Collaborative Research in Computational Neuroscience Grant No. 1R01MH125544-01 and NSF Grant No. CISE 2212519.

DATA AVAILABILITY

The data that support the findings of this article are openly available [42]; embargo periods may apply.

APPENDIX A: COST HESSIAN AND TUNING SUSCEPTIBILITY FOR UNSATISFIED CONSTRAINTS

The mathematical derivations of the main text assumed that the physical network was trained to reach zero cost, i.e., that all the constraints were satisfied, $c_r = 0\ \forall\ r$. Here, we generalize our derivations to the case in which constraints cannot be satisfied and the minimum cost is nonzero. A direct computation of the cost Hessian leads to

$$\mathcal{H}_{ij} = \sum_{r=1}^{R} \frac{\partial c_r}{\partial k_i}\frac{\partial c_r}{\partial k_j} + \sum_{r=1}^{R} c_r \frac{\partial^2 c_r}{\partial k_j \partial k_i}, \tag{A1}$$

where the constraint can be directly expressed in terms of the physical Hessian,

$$c_r = \mathbf{Q}_r^T \mathbf{H}^{-1} \mathbf{I}_r - V_r^{\mathrm{des}}, \tag{A2}$$

and the gradient of a constraint can be expressed in terms of the susceptibility vector $\mathbf{s}_i = \mathbf{H}^{-1}\boldsymbol{\Delta}_i$:

$$\frac{\partial c_r}{\partial k_i} = \mathbf{Q}_r^T \frac{\partial \mathbf{H}^{-1}}{\partial k_i}\mathbf{I}_r = -2(\mathbf{Q}_r \otimes \mathbf{I}_r) \vdots (\mathbf{s}_i \otimes \mathbf{s}_i). \tag{A3}$$

To compute the Hessian of the constraints, we thus need the gradient of the susceptibility

$$\frac{\partial \mathbf{s}_i}{\partial k_j} = \frac{\partial \mathbf{H}^{-1}}{\partial k_j}\boldsymbol{\Delta}_i = -2(\mathbf{s}_j \otimes \mathbf{s}_j)\boldsymbol{\Delta}_i = -2(\boldsymbol{\Delta}_i^T \mathbf{s}_j)\mathbf{s}_j, \tag{A4}$$

leading to

$$\begin{aligned}\frac{\partial^2 c_r}{\partial k_j \partial k_i} &= -2(\mathbf{Q}_r \otimes \mathbf{I}_r) \vdots \left(\frac{\partial \mathbf{s}_i}{\partial k_j} \otimes \mathbf{s}_i + \mathbf{s}_i \otimes \frac{\partial \mathbf{s}_i}{\partial k_j}\right) \\ &= 4(\mathbf{Q}_r \otimes \mathbf{I}_r) \vdots [(\boldsymbol{\Delta}_i^T \mathbf{s}_j)(\mathbf{s}_i \otimes \mathbf{s}_j + \mathbf{s}_j \otimes \mathbf{s}_i)].\end{aligned} \tag{A5}$$

Putting all these results together, we have that the cost Hessian can be split into training and physical information as

$$\begin{aligned}\mathcal{H}_{ij} = \mathbf{L} \vdots \mathbf{S}_{ij} &+ 4\mathbf{L} \vdots (\mathbf{H}^{-1} \otimes (\boldsymbol{\Delta}_i^T \mathbf{s}_j)(\mathbf{s}_i \otimes \mathbf{s}_j + \mathbf{s}_j \otimes \mathbf{s}_i)) \\ &- 4\left(\sum_r V_r^{\mathrm{des}} \mathbf{Q}_r \otimes \mathbf{I}_r\right) \vdots [(\boldsymbol{\Delta}_i^T \mathbf{s}_j)(\mathbf{s}_i \otimes \mathbf{s}_j + \mathbf{s}_j \otimes \mathbf{s}_i)],\end{aligned} \tag{A6}$$

where $\mathbf{L}$ is the training tensor defined in Eq. (10). The first term corresponds to the cost Hessian under fully satisfied constraints. The two new ones arise when the system is at a cost minimum but constraints are not fully satisfied. As with the main result, these last two terms can be written as the contraction of a tensor depending only on training information (left) with a tensor depending only on physical properties (right).

In practice, most of the trained systems belong to this more general case, in which the error is never lowered to zero and, even more, is never exactly at a minimum (see the negative eigenvalues of trained systems in Fig. 5). However, when training is successful, the cost is small, which translates to small constraint values c_r. In such a scenario, the second and third terms of Eq. (A1) can be seen as a perturbation to the first, and our results hold to first order, as demonstrated by the numerical examples.

APPENDIX B: L-S DECOMPOSITION FOR GENERAL COST FUNCTIONS

For constraints that are different from the MSE used in the main text, the cost function C can be written as

$$C = \frac{1}{2}\sum_{r=1}^{R} c_r^2, \tag{B1}$$

where $c_r = c_r(\mathbf{Q}_r^T \mathbf{V}_r^F)$ are generic (nonlinear) differentiable constraints depending on the voltages at the output nodes, $\mathbf{Q}_r^T \mathbf{V}_r^F$. For fully trained networks, $C = 0$, and the cost Hessian is given by

$$\mathcal{H}_{ij} = \frac{\partial^2 C}{\partial k_i \partial k_j} = \sum_{r=1}^{R} \frac{\partial c_r}{\partial k_i}\frac{\partial c_r}{\partial k_j}, \tag{B2}$$

where

$$\frac{\partial c_r}{\partial k_i} = c_r' \mathbf{Q}_r^T \frac{\partial \mathbf{V}^F}{\partial k_i} = c_r' \mathbf{Q}_r^T \frac{\partial \mathbf{H}^{-1}}{\partial k_i}\mathbf{I}_r, \tag{B3}$$

and $c_r' = c_r'(V_r^{\mathrm{des}})$ is the derivative of the constraint with respect to its arguments evaluated at the desired output values. In the case of the MSE cost used in the main text, this last term is equal to 1.

Using the last two equations, we can write the cost Hessian as

$$\mathcal{H}_{ij} = \sum_{r=1}^{R} c_r'^2 \left(\mathbf{Q}_r^T \frac{\partial \mathbf{H}^{-1}}{\partial k_i}\mathbf{I}_r\right)\left(\mathbf{Q}_r^T \frac{\partial \mathbf{H}^{-1}}{\partial k_j}\mathbf{I}_r\right), \tag{B4}$$

which can again be split into a learning $\mathbf{L}$ and a susceptibility $\mathbf{S}$ tensor:

$$\mathcal{H}_{ij} = \mathbf{L} \vdots \mathbf{S}_{ij} = \sum_{a,b,c,d} \mathbf{L}_{abcd}\mathbf{S}_{ij,abcd}, \tag{B5}$$

with

$$\mathbf{L} = \sum_{r=1}^{R} c_r'^2 \mathbf{Q}_r \otimes \mathbf{I}_r \otimes \mathbf{Q}_r \otimes \mathbf{I}_r, \tag{B6}$$

$$\mathbf{S}_{ij} = \frac{\partial \mathbf{H}^{-1}}{\partial k_i} \otimes \frac{\partial \mathbf{H}^{-1}}{\partial k_j}. \tag{B7}$$

As expected, the susceptibility tensor is not modified by the specifics of the cost function.

APPENDIX C: L-S DECOMPOSITION FOR ELASTIC SYSTEMS

Here, we treat the general case in which the physical degrees of freedom are not scalar values, such as voltages, but vectors, like displacements in elastic networks. Moreover, this appendix provides a concrete example on how to treat a nonlinear system according to conditions (1)–(3) of Sec. V. Let us denote by d_i and l_i the elongation and rest length of edge i, and by $\boldsymbol{r}_a$ the position of node a, with as many entries as dimensions in the system. The elongation of edge i connecting nodes a and b is given by

$$d_i = \|\boldsymbol{r}_a - \boldsymbol{r}_b\| - l_i. \tag{C1}$$

We next consider small perturbations $\boldsymbol{u}_a$ to the equilibrium positions $\bar{\boldsymbol{r}}_a$, $\boldsymbol{r}_a = \bar{\boldsymbol{r}}_a + \boldsymbol{u}_a$, as well as small perturbations e_i to the equilibrium elongations $\bar{d}_i$, $d_i = \bar{d}_i + e_i$. To linear order in the perturbations, we have

$$\bar{d}_i = (\|\bar{\boldsymbol{r}}_a - \bar{\boldsymbol{r}}_b\| - l_i), \tag{C2}$$

$$e_i = \hat{n}_i \cdot (\boldsymbol{u}_a - \boldsymbol{u}_b), \tag{C3}$$

with $\hat{n}_i \equiv (\bar{\boldsymbol{r}}_a - \bar{\boldsymbol{r}}_b)/\|\bar{\boldsymbol{r}}_a - \bar{\boldsymbol{r}}_b\|$ the unit vector connecting the two equilibrium positions. Collecting all the perturbed elongations into a single vector $\mathbf{e} = (e_1, e_2, \ldots)$, and all the perturbed positions into a vector $\mathbf{u} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots)$, the previous relation reads

$$\mathbf{e} = R^T \mathbf{u}, \tag{C4}$$

where R^T is the so-called compatibility matrix and R the equilibrium matrix [43]. Its components are defined in terms of the equilibrium configuration as

$$R_{a,i} = \begin{cases} \hat{n}_i & \text{if edge } i \text{ enters node } a \\ -\hat{n}_i & \text{if edge } i \text{ exits node } a \\ 0 & \text{if edge } i \text{ is not connected to node } a. \end{cases} \tag{C5}$$

The compatibility matrix generalizes the incidence matrix of resistor networks. Indeed, considering an elastic system with fixed nodes (the analog of ground nodes in the resistor network) to remove the trivial translational and rotational modes, the extended physical Hessian of the elastic energy reads

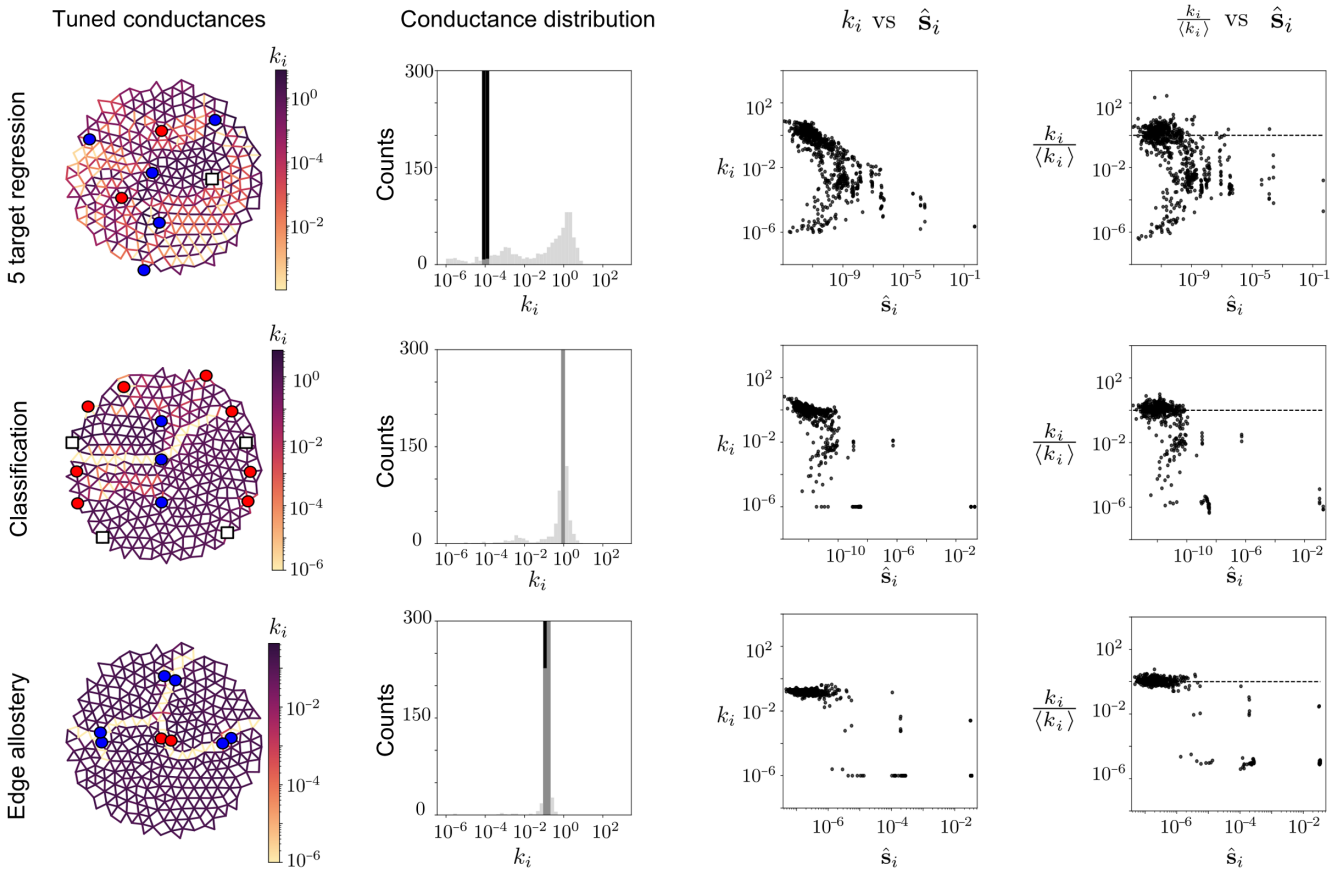

FIG. 10. Physical properties of three networks (rows) trained for the different tasks. From top to bottom, we have circuits trained for linear regression response at five targets, classification of three clusters (both shown in Fig. 5), and edge allostery, shown in Fig. 6. From left to right: conductance configuration of the trained networks; initial (light gray) and final (black) distribution of conductances; relation between final conductances and susceptibilities; relation between final relative conductances and susceptibilities (horizontal dashed line at 1).

$$\mathbf{H} = \begin{pmatrix} 2RKR^T & \mathbf{q} \\ \mathbf{q} & 0 \end{pmatrix}, \tag{C6}$$

where $K_{ij} = \delta_{ij} k_i$ is the diagonal matrix of stiffness and $\mathbf{q}$ is the projector onto the fixed nodes, which now becomes a matrix instead of a vector since more than 1 physical degree of freedom is needed to remove all the zero modes. Under this formulation, the displacement configuration is extended with Lagrange multipliers to ground the system, $\mathbf{u} = (\boldsymbol{u}_1, \ldots \boldsymbol{u}_N, \lambda_1, \lambda_2, \ldots)$. As before, the bottom-right corner is a zero matrix ensuring that the constrained optimization is linear in all the Lagrange multipliers.

As with the power in resistor networks, the linear response to external node forces $\mathbf{F} = (\boldsymbol{F}_1, \boldsymbol{F}_2, \ldots, \boldsymbol{F}_N, 0, 0, \ldots)$, where the last entries correspond to the zero displacements of the ground nodes, minimizes the elastic energy,

$$\mathcal{E} = \frac{1}{2}\mathbf{u}^T \mathbf{H} \mathbf{u} - \mathbf{F}^T \mathbf{u}, \tag{C7}$$

leading to

$$\mathbf{u}^F = \mathbf{H}^{-1} \mathbf{F}. \tag{C8}$$

Formally, the difference with the physical Hessian of resistor networks is the exchange of the incidence matrix by the compatibility matrix, which changes the gradient of $\mathbf{H}^{-1}$ [Eq. (13) in the main text] as follows:

$$\frac{\partial \mathbf{H}^{-1}}{\partial k_i} = -\mathbf{H}^{-1} \frac{\partial \mathbf{H}}{\partial k_i} \mathbf{H}^{-1} = -\mathbf{H}^{-1} \begin{pmatrix} 2R_i R_i^T & \mathbf{0} \\ \mathbf{0}^T & 0 \end{pmatrix} \mathbf{H}^{-1}, \tag{C9}$$

where R_i is the vector formed by the ith row of the equilibrium matrix R. As in the main text, we define the extended vector (bold font) $\mathbf{R}_i = (R_i, \mathbf{0})$ by padding as many zeros as Lagrange multipliers. We can then generalize the susceptibility vector by simply promoting the incidence matrix to the equilibrium matrix,

$$\mathbf{s}_i \equiv \mathbf{H}^{-1} \mathbf{R}_i, \tag{C10}$$

from which we obtain

$$\frac{\partial \mathbf{H}^{-1}}{\partial k_i} = -2\mathbf{s}_i \otimes \mathbf{s}_i, \tag{C11}$$

which is identical to Eq. (15). From here, all the results derived in the main text ensue. As numerical examples, we compare the susceptibilities and the stiff cost modes of two mechanical networks trained for an allosteric and a

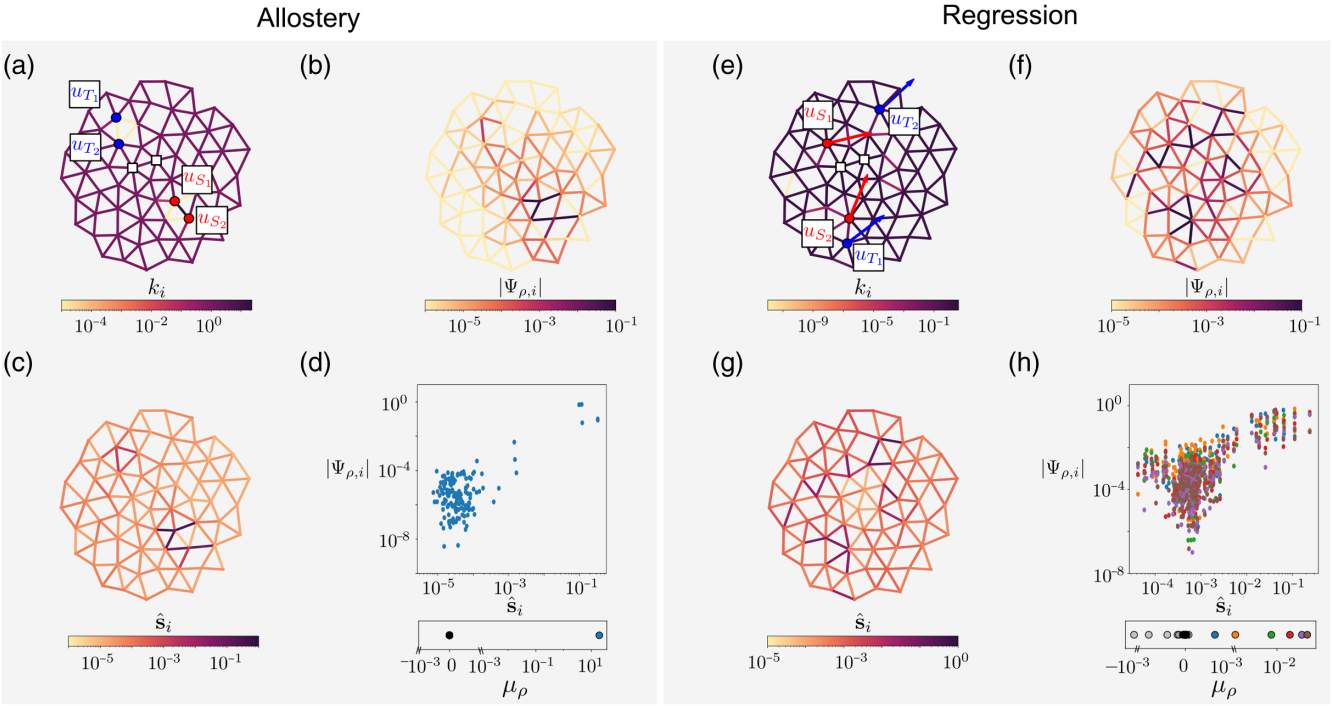

FIG. 11. Stiff modes and tuning susceptibility for allostery and regression tasks in mechanical networks. Allostery: (a) Stiffness configuration of an elastic network trained to produce a 8% strain at the output edge in response to a 10% strain at the input edge. Nodes of the input (output) edge are denoted by red (blue) circles. (b) Magnitudes of the entries of the stiff cost mode, $|\Psi_{\rho,i}|$. (c) Normalized tuning susceptibility norm $\hat{\mathbf{s}}_i$, with darker colors indicating larger susceptibilities. (d) Edges of higher susceptibility indicating the largest entries of the stiff mode. Bottom: eigenvalue spectrum of the cost Hessian. Regression: (e), (f), (g), and (h) with the same information as (a), (b), (c), and (d), respectively, for a network trained to produce displacements at output nodes as the following combination of input forces: $\begin{pmatrix} \boldsymbol{u}_{T_1} \\ \boldsymbol{u}_{T_2} \end{pmatrix} = \begin{pmatrix} 0.7 & 0.4 \\ 0.2 & 0.1 \end{pmatrix} \begin{pmatrix} \boldsymbol{F}_{S_1} \\ \boldsymbol{F}_{S_2} \end{pmatrix}$. In panel (e), red arrows indicate a pair of input forces, while blue arrows indicate the desired displacement response.

regression task, respectively (see Fig. 11). Aligned with the results of the main text, high susceptibilities indicate the large entries of the stiff modes.

APPENDIX D: COMPARISON BETWEEN PERSISTENT SECTORS AND TUNING SUSCEPTIBILITY IN ALLOSTERIC NETWORKS

Here, we provide further evidence that the sectors obtained by persistence homology in allosteric networks are readily captured by the pattern of susceptibilities and that they correspond to the stiff modes of the cost Hessian. We consider the same task as in the main text: two input nodes with voltages 0 and 1, and three target edges trained for the same voltage drop value ΔV^{des}. In addition to the original task shown in Fig. 6 ($\Delta V^{\mathrm{des}} = 0.5$), we consider two more subtle cases: $\Delta V^{\mathrm{des}} = 0.3$ and $\Delta V^{\mathrm{des}} = 0.1$, which require much less training but leave weaker imprints on the cost Hessian. The voltage response field shows sharper variations as ΔV^{des} increases, Fig. 12(a), leading to clear partitions, or sectors, captured by persistent homology [27], Fig. 12(b). As ΔV^{des} decreases, the susceptible edges have a smaller overlap with the boundaries of the persistent sectors, yet they accurately capture the stiff modes of the cost Hessian, Fig. 12(c). For details of the persistent homology algorithm, see Ref. [27].

APPENDIX E: TUNING SUSCEPTIBILITIES AND CONDUCTANCES OF REMAINING EXAMPLES

In Fig. 10, we show the distribution of conductances and edge susceptibilities for the circuits in Figs. 5(a), 5(e), and 6. For all these cases, the same general features mentioned in the main text hold: High-susceptible edges have low relative conductances, in general, while the conductance itself is not indicative of the susceptibility values.

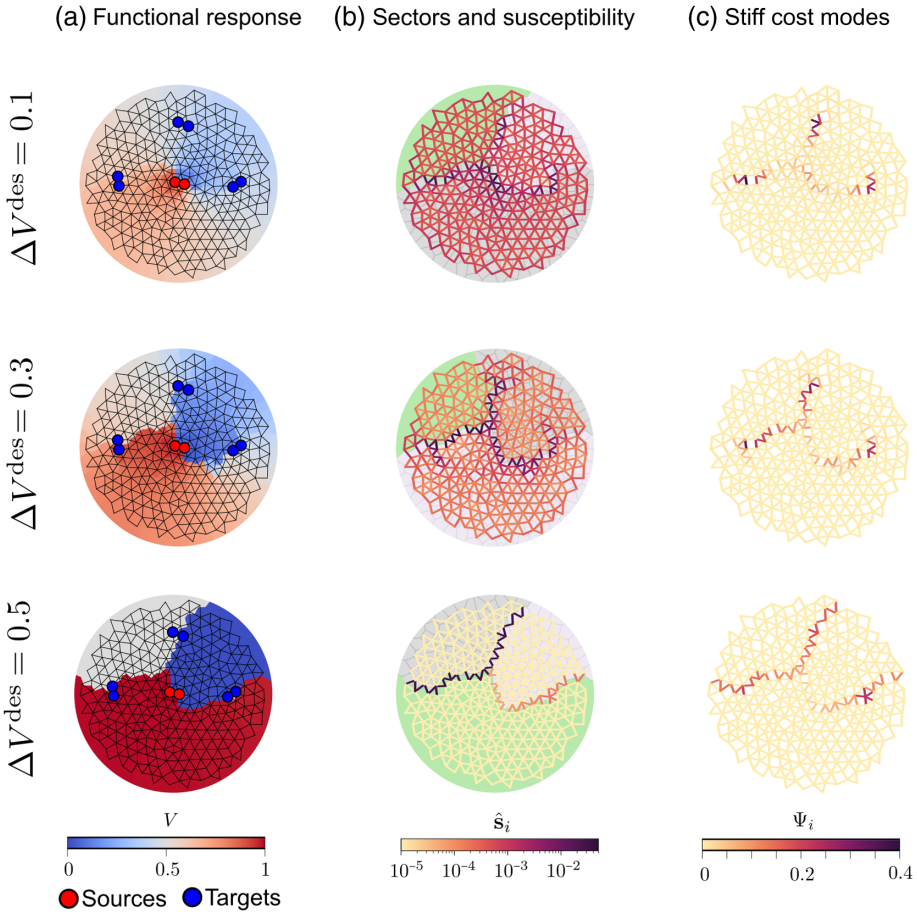

FIG. 12. From top to bottom, each row corresponds to the same circuit trained for an allosteric response of increasing strength ΔV^{des} at three target sites (blue) in response to source voltages 0 and 1 (red). (a) Voltage response to inputs 0 and 1 (red). Stronger responses showcase a clearer partitioning of the voltage field. (b) Persistent homology provides a topological partition of the network into distinct sectors (background color), based on the voltage response. Edges are colored according to their tuning susceptibility, delimiting the topological sectors as ΔV^{des} increases. (c) For all cases, the three stiffest modes of the cost Hessian closely match the highly susceptible edges, despite the mismatch with the topological sectors. Edges are colored by the sum of stiff modes ($\sum_\rho |\Psi_{\rho,i}|$) after normalization.

APPENDIX F: TRAINING PROTOCOL

All the networks were trained using coupled learning [31], a contrastive local learning rule with two hyperparameters: the learning rate α and the nudge parameter η (for details of the training scheme, see Ref. [31]).

The first linear regression circuit (Fig. 1) was trained for approximately 10^6 iterations, each iteration batching 30 training samples, using $\alpha = 10^{-4}$, $\eta = 10^{-2}$, and initial conductances drawn from a uniform distribution in [0.01, 1], reaching a final MSE cost of $C \approx 10^{-5}$. The input data consisted of 112 pairs of voltages (V_{S_1}, V_{S_2}) with corresponding outputs given by the linear relation of Eq. (1).

The second linear regression circuit [Fig. 5(a)] was trained for approximately 3×10^5 iterations, each iteration batching 30 training samples, using $\alpha = 10^{-2}$, $\eta = 10^{-3}$, and initial conductances drawn from a uniform distribution in [0.0001, 0.00011], reaching a final MSE cost of $C \approx 10^{-8}$. The input data consisted of 112 pairs of voltages (V_{S_1}, V_{S_2}) with corresponding outputs given by the linear relation of Eq. (29).

The classification circuit [Fig. 5(e)] was trained for approximately 2×10^4 iterations, each iteration batching 30 training samples, using $\alpha = 10^{-1}$, $\eta = 10^{-2}$, and initial conductances all equal to 1, reaching a final cosine similarity cost of $C \approx 0.05$, training accuracy of 99% (120 points), and test accuracy of 100% (30 points). The input data consisted of three clusters of points, one per class, generated from the following normal distributions:

$$\mathcal{N}\left([2,2], \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}\right), \tag{F1}$$

$$\mathcal{N}\left([7,7], \begin{pmatrix} 1 & -0.5 \\ -0.5 & 1 \end{pmatrix}\right), \tag{F2}$$

$$\mathcal{N}\left([1,8], \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\right). \tag{F3}$$

The allosteric circuit (Fig. 6) was trained for 5×10^4 iterations, no batching, using $\alpha = 10^{-2}$ and $\eta = 10^{-3}$, and initial conductances were drawn from a uniform distribution in [0.1, 0.2], reaching a final MSE cost of $C \approx 1 \times 10^{-5}$. The input data were $V_{S_1} = 1$, $V_{S_2} = 0$, with corresponding output $\Delta V^{\mathrm{des}} = 0.5$. The two additional allosteric circuits in Fig. 12 were trained for 5×10^4 iterations, no batching, using $\alpha = 10^{-2}$ and $\eta = 10^{-3}$, and initial conductances were drawn from a uniform distribution in [0.1, 0.2], reaching final MSE costs of $C \approx 2 \times 10^{-8}$ ($\Delta V^{\mathrm{des}} = 0.1$) and $C \approx 7 \times 10^{-10}$ ($\Delta V^{\mathrm{des}} = 0.3$).

The elastic network of the allosteric strain-strain relation was trained with $\alpha = 10^3$, $\eta = 0.1$, and initial stiffness values equal to 1. The final cost was $C \approx 10^{-14}$. The dataset consisted of a fixed input-output relation with 10% input strain and a desired 8% output strain. The elastic network of the regression example was trained with $\alpha = 10^{-2}$, $\eta = 0.2$, and initial stiffness values equal to 1. The final cost obtained was $C \approx 10^{-7}$. The dataset consisted of 50 different pairs of input forces and desired displacements, following

$$\begin{pmatrix} \boldsymbol{u}_{T_1} \\ \boldsymbol{u}_{T_2} \end{pmatrix} = \begin{pmatrix} 0.7 & 0.4 \\ 0.2 & 0.1 \end{pmatrix} \begin{pmatrix} \boldsymbol{F}_{S_1} \\ \boldsymbol{F}_{S_2} \end{pmatrix}. \tag{F4}$$

APPENDIX G: TUNING SUSCEPTIBILITY FOR WEAKLY TRAINED NETWORKS

Figure 13 shows how the tuning susceptibility depends on the task strength. We consider an allosteric task with two

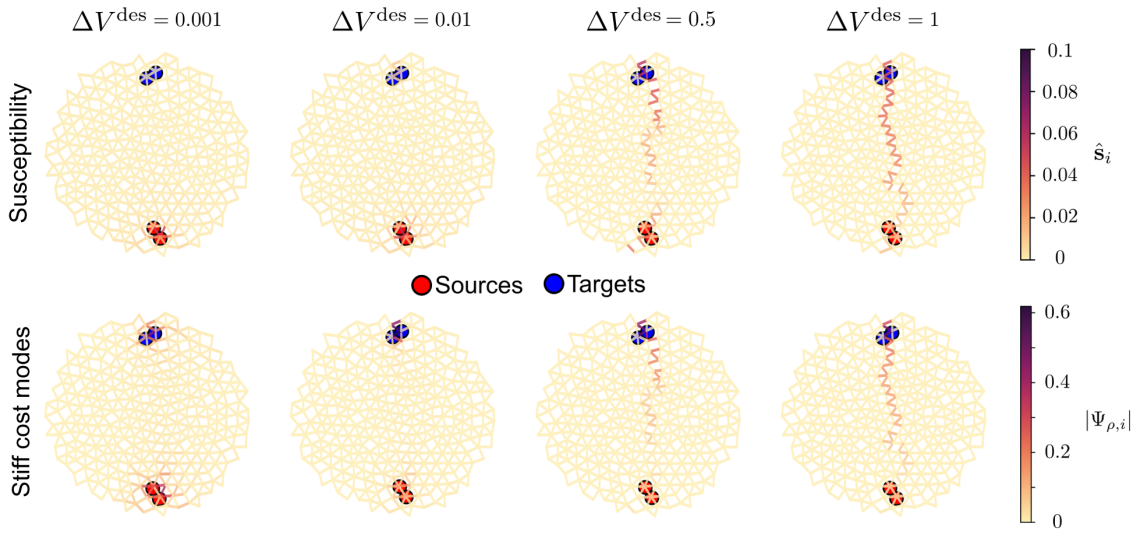

FIG. 13. From left to right, the normalized tuning susceptibility (top) for increasing allosteric coupling ΔV^{des}. Most of the highly susceptible edges correspond to the ones captured by the single stiff mode of the cost Hessian (bottom).

input voltages (0 and 1) and one edge target with an increasing desired voltage drop $\Delta V^{\text{des}} = 0.001, 0.01, 0.5, 1$. The lower the desired voltage drop, the easier the task, and less training steps are required. In Fig. 13, we show that, despite the weak signals, the susceptibilities still capture the relevant edges highlighted by the stiff mode of the cost Hessian.

[1] P. Mehta, M. Bukov, C.-H. Wang, A. G. Day, C. Richardson, C. K. Fisher, and D. J. Schwab, *A high-bias, low-variance introduction to machine learning for physicists*, Phys. Rep. **810**, 1 (2019).

[2] L. Sagun, L. Bottou, and Y. LeCun, *Eigenvalues of the Hessian in deep learning: Singularity and beyond*, arXiv:1611.07476.

[3] L. Sagun, U. Evci, V. U. Guney, Y. Dauphin, and L. Bottou, *Empirical analysis of the Hessian of over-parametrized neural networks*, arXiv:1706.04454.

[4] Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, *Identifying and attacking the saddle point problem in high-dimensional non-convex optimization*, arXiv:1406.2572.

[5] M. Sabanayagam, F. Behrens, U. Adomaityte, and A. Dawid, *Unveiling the Hessian's connection to the decision boundary*, arXiv:2306.07104.

[6] J. W. Rocks, N. Pashine, I. Bischofberger, C. P. Goodrich, A. J. Liu, and S. R. Nagel, *Designing allostery-inspired response in mechanical networks*, Proc. Natl. Acad. Sci. U.S.A. **114**, 2520 (2017).

[7] M. B. Pinson, M. Stern, A. Carruthers Ferrero, T. A. Witten, E. Chen, and A. Murugan, *Self-folding origami at any energy scale*, Nat. Commun. **8**, 15477 (2017).

[8] D. Hexner, N. Pashine, A. J. Liu, and S. R. Nagel, *Effect of directed aging on nonlinear elasticity and memory formation in a material*, Phys. Rev. Res. **2**, 043231 (2020).

[9] N. Pashine, *Local rules for fabricating allosteric networks*, Phys. Rev. Mater. **5**, 065607 (2021).

[10] C. Arinze, M. Stern, S. R. Nagel, and A. Murugan, *Learning to self-fold at a bifurcation*, Phys. Rev. E **107**, 025001 (2023).

[11] M. Stern, D. Hexner, J. W. Rocks, and A. J. Liu, *Supervised learning in physical networks: From machine learning to learning machines*, Phys. Rev. X **11**, 021045 (2021).

[12] S. Dillavou, M. Stern, A. J. Liu, and D. J. Durian, *Demonstration of decentralized physics-driven learning*, Phys. Rev. Appl. **18**, 014040 (2022).

[13] S. Dillavou, B. D. Beyer, M. Stern, A. J. Liu, M. Z. Miskin, and D. J. Durian, *Machine learning without a processor: Emergent learning in a nonlinear analog network*, Proc. Natl. Acad. Sci. U.S.A. **121**, e2319718121 (2024).

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

[15] M. Stern, M. Guzman, F. Martins, A. J. Liu, and V. Balasubramanian, *Physical networks become what they learn*, Phys. Rev. Lett. **134**, 147402 (2025).

[16] S. K. Vadlamani, T. P. Xiao, and E. Yablonovitch, *Physics successfully implements Lagrange multiplier optimization*, Proc. Natl. Acad. Sci. U.S.A. **117**, 26639 (2020).

[17] M. Stern, A. J. Liu, and V. Balasubramanian, *Physical effects of learning*, Phys. Rev. E **109**, 024311 (2024).

[18] T. Tlusty, A. Libchaber, and J.-P. Eckmann, *Physical model of the genotype-to-phenotype map of proteins*, Phys. Rev. X **7**, 021037 (2017).

[19] L. Yan, R. Ravasio, C. Brito, and M. Wyart, *Architecture and coevolution of allosteric materials*, Proc. Natl. Acad. Sci. U.S.A. **114**, 2526 (2017).

[20] L. Yan, R. Ravasio, C. Brito, and M. Wyart, *Principles for optimal cooperativity in allosteric materials*, Biophys. J. **114**, 2787 (2018).

[21] K. Husain and A. Murugan, *Physical constraints on epistasis*, Mol. Biol. Evol. **37**, 2865 (2020).

[22] V. R. Anisetti, A. Kandala, and J. Schwarz, *Emergent learning in physical systems as feedback-based aging in a glassy landscape*, arXiv:2309.04382.

[23] J. W. Rocks, H. Ronellenfitsch, A. J. Liu, S. R. Nagel, and E. Katifori, *Limits of multifunctionality in tunable networks*, Proc. Natl. Acad. Sci. U.S.A. **116**, 2506 (2019).

[24] A. Zaccone, *Theory of Disordered Solids* (Springer, Cham, 2023).

[25] L. E. Altman, M. Stern, A. J. Liu, and D. J. Durian, *Experimental demonstration of coupled learning in elastic networks*, Phys. Rev. Appl. **22**, 024053 (2024).

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

[27] J. W. Rocks, A. J. Liu, and E. Katifori, *Hidden topological structure of flow network functionality*, Phys. Rev. Lett. **126**, 028102 (2021).

[28] J. W. Rocks, E. Katifori, and A. J. Liu, *Topological characterization of the continuum of allosteric response*, arXiv:2401.13861.

[29] R. N. Gutenkunst, J. J. Waterfall, F. P. Casey, K. S. Brown, C. R. Myers, and J. P. Sethna, *Universally sloppy parameter sensitivities in systems biology models*, PLoS Comput. Biol. **3**, e189 (2007).

[30] M. K. Transtrum, B. B. Machta, K. S. Brown, B. C. Daniels, C. R. Myers, and J. P. Sethna, *Perspective: Sloppiness and emergent theories in physics, biology, and beyond*, J. Chem. Phys. **143**, 010901 (2015).

[31] M. Stern, S. Dillavou, M. Z. Miskin, D. J. Durian, and A. J. Liu, *Physical learning beyond the quasistatic limit*, Phys. Rev. Res. **4**, L022037 (2022).

[32] E. Rouviere, R. Ranganathan, and O. Rivoire, *Emergence of single-versus multi-state allostery*, PRX Life **1**, 023004 (2023).

[33] D. Hexner, A. J. Liu, and S. R. Nagel, *Periodic training of creeping solids*, Proc. Natl. Acad. Sci. U.S.A. **117**, 31690 (2020).

[34] Y. Togashi and H. Flechsig, *Coarse-grained protein dynamics studies using elastic network models*, Int. J. Mol. Sci. **19**, 3899 (2018).

[35] N. Halabi, O. Rivoire, S. Leibler, and R. Ranganathan, *Protein sectors: Evolutionary units of three-dimensional structure*, Cell **138**, 774 (2009).

[36] K. Alim, N. Andrew, A. Pringle, and M. P. Brenner, *Mechanism of signal propagation in physarum polycephalum*, Proc. Natl. Acad. Sci. U.S.A. **114**, 5136 (2017).

[37] V. R. Anisetti, A. Kandala, B. Scellier, and J. Schwarz, *Frequency propagation: Multi-mechanism learning in nonlinear physical networks*, arXiv:2208.08862.

[38] K. Hornik, M. Stinchcombe, and H. White, *Multilayer feedforward networks are universal approximators*, Neural Netw. **2**, 359 (1989).

[39] M. Stern, S. Dillavou, D. Jayaraman, D. J. Durian, and A. J. Liu, *Training self-learning circuits for power-efficient solutions*, APL Mach. Learn. **2**, 016114 (2024).

[40] A. J. Liu and S. R. Nagel, *The jamming transition and the marginally jammed solid*, Annu. Rev. Condens. Matter Phys. **1**, 347 (2010).

[41] A. Zaccone and E. Scossa-Romano, *Approximate analytical description of the nonaffine response of amorphous solids*, Phys. Rev. B-**83**, 184205 (2011).

[42] M. Guzman, F. Martins, M. Stern, and A. J. Liu, Zenodo V1 (2025), 10.5281/zenodo.14680896.

[43] C. R. Calladine, *Buckminster Fuller's, "tensegrity" structures and Clerk Maxwell's rules for the construction of stiff frames*, Int. J. Solids Struct. **14**, 161 (1978).