Flow networks as learning machines [1]

Menachem Stern, Daniel Hexner, Jason W. Rocks, Andrea J. Liu

Supervised learning (classification)

Distinguish between labeled classes in data.

Learn by example – we do not specify what dog/cat are.

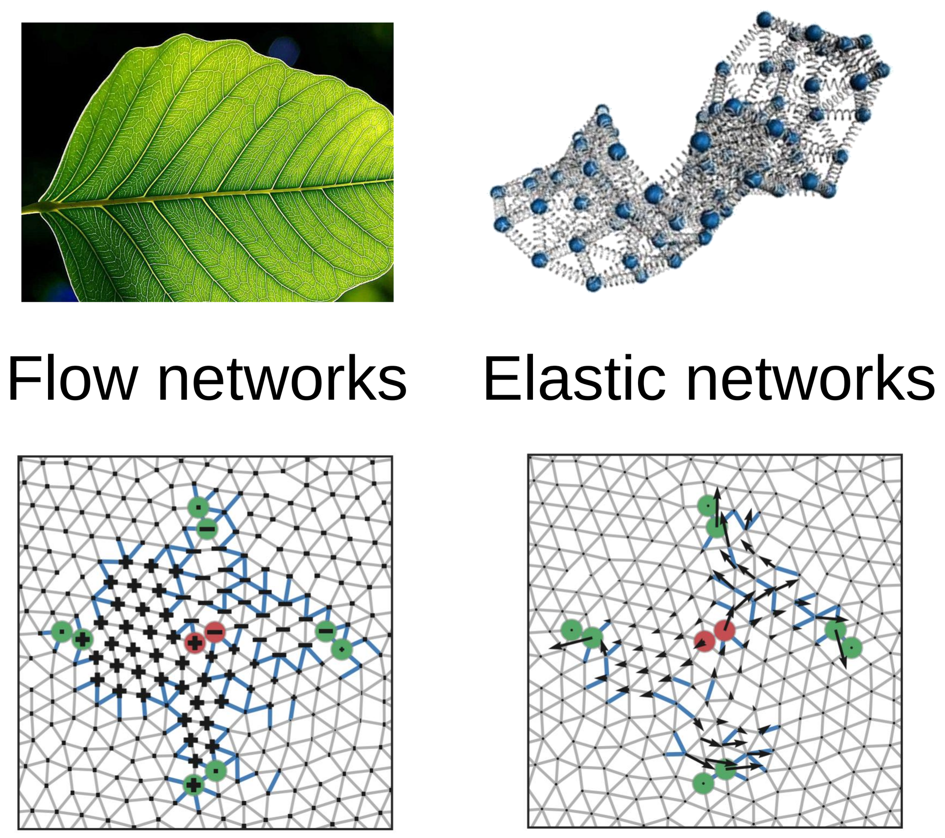
Generalize – predict the correct class for unseen examples.

Goal: Realize adaptive (learning) materials/machines

- Learn tasks in-situ **by examples** of use
- **Generalize** to diverse operational settings

Functions (e.g. allostery) were previously programmed onto machines by tuning of a cost function [2], e.g. modify system component w in proportion to the gradient of a cost function.

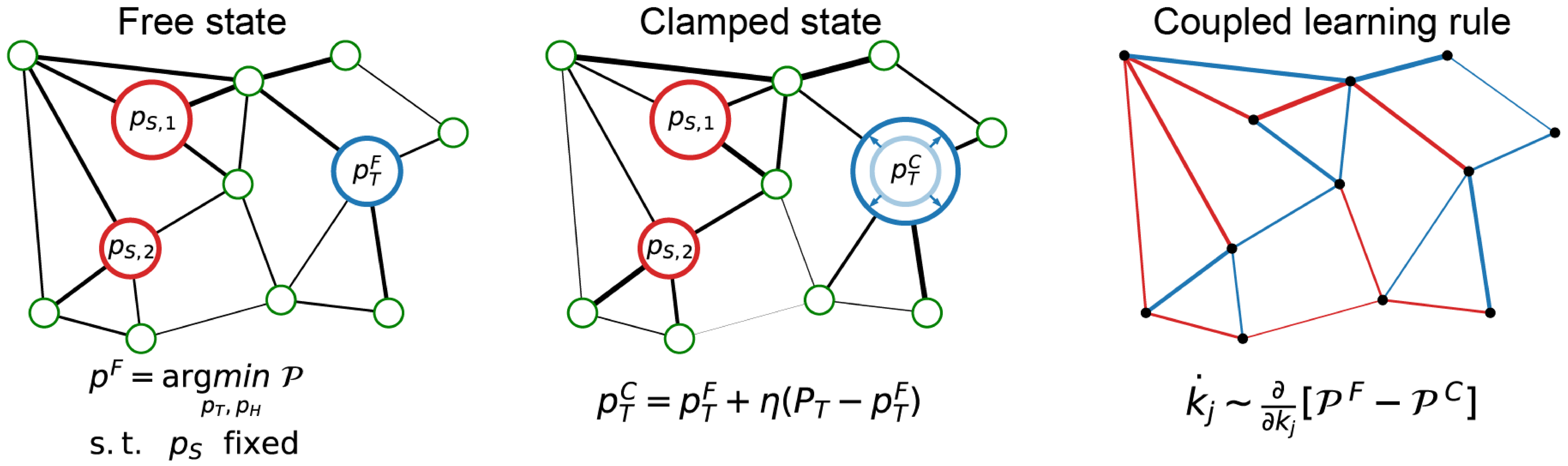
$$Cost \sim (\text{Desired response} - \text{Actual response})^2$$
$$\Delta w \sim -\partial_w Cost$$



However, this approach requires explicit knowledge of the desired response and accounting for non-local effects – it requires an *external designer.*

Coupled local supervised learning

Based on advances in computer science and neuroscience [3, 4, 5], we propose a class of local, physically plausible learning rules, coupled learning.



In coupled learning, every network element contrasts its own response in two states, the *free state*, where only source constraints are applied, and *clamped state*, where target nodes are slightly nudged towards their desired value.

Coupled learning in flow networks

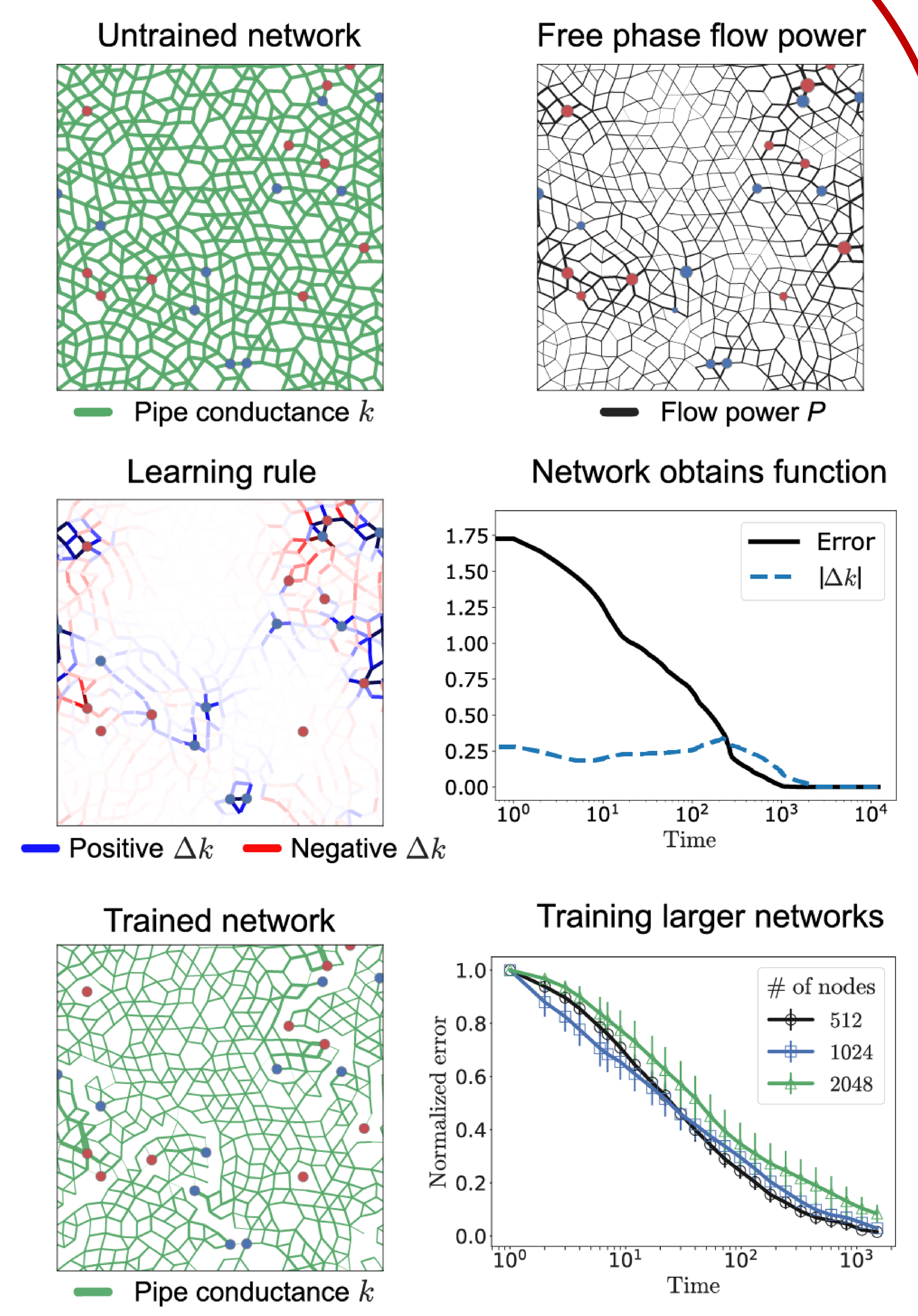
Applying the coupled learning approach, we find physically plausible learning rules for flow networks.

Dissipated power at the free and clamped states

$$\mathcal{P}^F = \frac{1}{2}\sum_{j \in \text{edges}} k_j \, \Delta p_j^2 (\text{applied inputs only})$$
$$\mathcal{P}^C = \frac{1}{2}\sum_{j \in \text{edges}} k_j \, \Delta p_j^2 (\text{inputs, outputs } at\ p_o^C)$$
$$p_o^C = (1-\eta)(\text{obtained } p_o) + \eta(\text{desired } P_o)$$

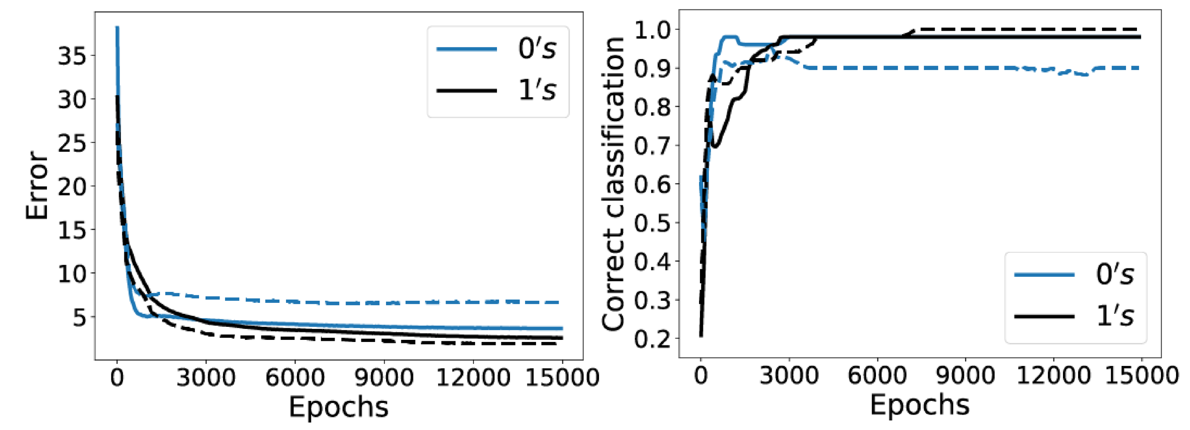
In the clamped state, output nodes are slightly nudged towards the desired state P_o, with $\eta \ll 1$. Conductance values k are updated by

$$\Delta k_j \sim \eta^{-1} \nabla_k [\mathcal{P}^F - \mathcal{P}^C] \sim \eta^{-1}\left[\left(\Delta p_j^F\right)^2 - \left(\Delta p_j^C\right)^2\right]$$

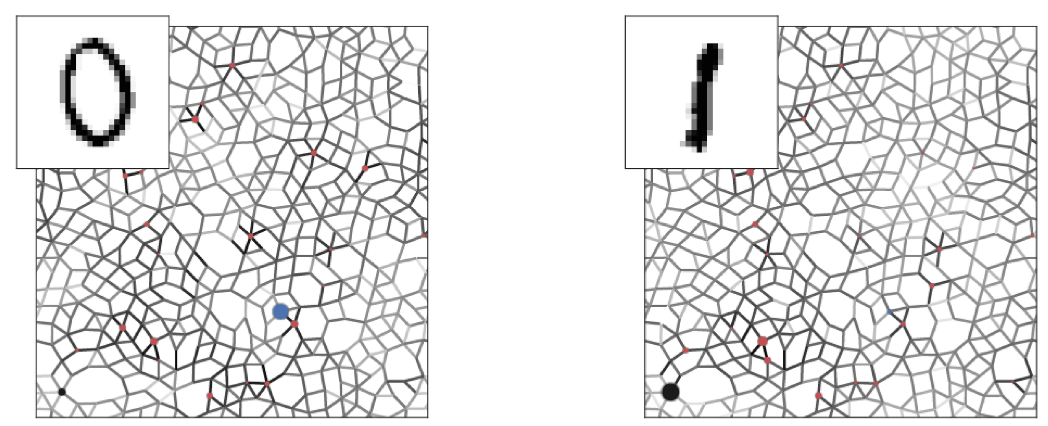


Coupled learning allows the training of flow networks for complex tasks, reducing the error in the obtained functionality by orders of magnitude.

Applying coupled learning to handwritten digit recognition (MNIST), we train a flow network to distinguish images of 0's and 1's with similar performance to simple machine learning algorithms.



The coupled learning framework can be applied to arbitrary physical networks, yielding local, physically plausible learning rules. In addition to flow networks, we were able to train elastic spring networks for complex functions.



Recent published work has studied the application of equilibrium propagation to flow/resistor networks [6].

Penn
Physics & Astronomy

1. **Supervised learning in physical networks**. M. Stern, D. Hexner, J. W. Rocks, A. J. Liu, *arxiv:2011.03861 (2020).*
2. **Limits of multifunctionality in tunable networks**. J. W. Rocks et. al. *PNAS* **116**, 7 (2019).
3. **Contrastive Hebbian learning in the continuous Hopfield model**. J R. Movellan, *Connectionist Models 10-17* (1991).
4. **Equilibrium propagation: Bridging the gap between energy based models and backpropagation**. B. Scellier, Y. Bengio, *Front. Comp. Neuro., 11:24* (2017).
5. **Directed Aging, memory and nature's greed**. N. Pashine, D. Hexner, A. J. Liu, S. R. Nagel, *Science Advances* **5**, 12 (2019).
6. **Training end-to-end analog neural networks with equilibrium propagation**. J. Kendall et. al. *arxiv:2006.01981 (2020).*