2024 Boulder School on Self-Organizing Matter
Problems on "Learning Metamaterials"

Andrea Liu <ajliu@upenn.edu>

1. Come up with an example of a biological function and see if you can apply the adaptive matter framework to it.

   (a) What is the function? Can you express it in terms of constraints?

   (b) What are the adjustable parameters?

   (c) Are the physical degrees of freedom controlled by a Lyapunov function? (It's fine for the answer to be "No."

   (d) What is the process by which the parameters are (or were) adjusted? If it is by local rules, specify them if you know what they might be or what some of they might be; otherwise just say "Unknown local rules."

2. (Open ended; this is an active area of research for us and if you'd like to put in additional effort to carry the results to publishable stage, please let me know!) Classification of downsampled MNIST on networks of different architecture. Dr. Menachem (Nachi) Stern, who developed the coupled learning framework, has prepared a data set of 10×10-pixel images of the numerals 0, 1, 2. The goal is to classify a training set of these images correctly using a linear Constrastive Local Learning Network (CLLN). Note that current has to flow from the highest voltage to the lowest so it is be useful to set a highest voltage at one node and ground at another node as two extra input nodes. Also, in order to have negative as well as positive weights, the inputs can be entered as voltage differences between two nodes, and outputs can likewise be interpreted in terms of the voltage drops between two nodes. In other words, the goal is to have the digit 0 have the largest (signed) voltage difference between the two output nodes corresponding to that digit label, etc..

   The code for the network, dataset, etc. are at this link: https://github.com/NachiStern/MNIST-Contrasive-Local-Learning-Networks

   For questions about the code, dataset, etc. please ask Nachi Stern (nachis@sas.upenn.edu).

   It would be great if each team could choose two different network architectures for comparison. Among the possible architectures are (1) square lattice, (2) 3 layered structure with 1 input layer, one hidden layer, one output layer, (3) fully connected network with every node connected to every other node, (4) square lattice with higher connectivity (e.g. next-nearest neighbors, next-next-nearest neighbors), (5) random connectivity but with a fixed coordination, (6) random connectivity but with variance in coordination, ...

   Please fix the total number of edges when making comparison, since this fixes the total number of parameters. For this problem, you might want to start with 1000 edges and then vary the number of edges to see how that affects classification accuracy.