Ising Machines: Theory and Practice

Naomi Sagan
Jaijeet Roychowdhury

Electrical Engineering and Computer Sciences
University of California, Berkeley

May 12, 2023

Ising Machines: Theory and Practice

by

Naomi Sagan

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, in partial satisfaction of the requirements for the degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

Committee:

Professor Jaijeet Roychowdhury
Research Advisor

May 12, 2023

(Date)

* * * * * * *

Professor Rajit Manohar
Second Reader

May 12, 2023

(Date)

Abstract

Ising machines have generated much excitement in recent years due to their promise for solving the Ising problem, a graph-based hard combinatorial optimization problem. In particular, Oscillator-based Ising Machines (OIMs), which consist of resistively-coupled nonlinear oscillators, are a promising on-chip iteration of Ising Machines. The system dynamics, represented as differential equations in oscillator phase, admit a Lyapunov function, *i.e.*, a function with a non-positive time derivative. This Lyapunov function, at stable equilibria, matches the Ising Hamiltonian, the discrete cost function of the Ising problem. However, there are both practical barriers to large-scale on-chip implementation and many unanswered fundamental theoretical questions.

First, achieving physical all-to-all connectivity in integrated circuit (IC) implementations of large, densely-connected Ising machines remains a key challenge. We present a novel approach, DaS, that uses low-rank decomposition to achieve effectively-dense Ising connectivity using only sparsely interconnected hardware. The innovation consists of two components. First, we use the SVD to find a low-rank approximation of the Ising coupling matrix while maintaining very high accuracy. This decomposition requires substantially fewer nonzeros to represent the dense Ising coupling matrix. Second, we develop a method to translate the low-rank decomposition to a hardware implementation that uses only sparse resistive interconnections. We validate DaS on the MU-MIMO detection problem, important in modern telecommunications. Our results indicate that as problem sizes scale, DaS can achieve dense Ising coupling using only 5%-20% of the resistors needed for brute-force dense connections (which would be physically infeasible in ICs). we also show the impact of this sparsification on achieved minimization using simulated annealing, a widely-used Ising solver. We achieve nearly 90% sparsity without degrading solution quality *at all*, even with loose thresholds on the approximation error of the sparsification process.

Second, it is not well-understood why OIMs perform well on certain problems and don't get stuck in local minima. In particular, parameter cycling, *i.e.*, gradually increasing the value of a particular OIM parameter over during the solution process, can help the system reach global minima. Past work on Ising machines indicates that traversal of bifurcations, or points where the number of solutions to a system changes, is essential to their Hamiltonian minimization properties. We explore bifurcations for OIMs with two or three oscillators, both analytically and using numerical methods. In addition, we draw connections between the landscape of the Lyapunov function and the potential of OIMs to find global Ising Hamiltonian minima, using a combination of theoretical analysis, numerical simulation, and Monte Carlo methods.

Contents

1 Introduction and Background

Over the last decade or so, a new and exciting technology called Ising machines has arisen for solving hard combinatorial optimization (CO) problems. CO problems [1] are important in a wide variety of practical applications, including protein folding, optimal logistics for healthcare/military operations/transportation, chip routing, cyber-security and cryptography, secure grids and communication networks, autonomous vehicles and robotics, *etc.*. These problems are generally very difficult to solve, *e.g.*, they are typically NP-hard/complete [2]. It has been shown that these problems can be mapped onto an equivalent NP-hard CO problem known as the Ising problem [3]. Ising machines, which use analog hardware to solve the Ising problem, show great promise for outperforming digital and software techniques for solving CO problems [4–13]. The Ising problem is described using a weighted, undirected graph where the nodes are called spins. As explained in Section 1.1, solving this problem corresponds to finding spin values that minimize a quantity known as the Ising Hamiltonian.

1.1 The Ising Problem

The Ising problem involves a weighted, undirected graph of n spin nodes, where there are no self-loops. The value of spin i is denoted s_i, and the weight between spins i and j is J_{ij}. $\vec{s}$ is the vector of spin values and J is the symmetric weighted adjacency matrix of the Ising graph, known as the Ising coupling matrix.

The objective of the Ising problem is to minimize the Ising Hamiltonian,

$$H(\vec{s}) = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j\neq i}J_{ij}s_i s_j, \tag{1}$$

where the spins are restricted to be either $+1$ or -1 (or, in shorthand, $\in \{\pm 1\}$).

1.2 Oscillator-based Ising Machines

The focus of this report is Oscillator-based Ising Machines (OIMs) [9–11], which use networks of resistively-coupled nonlinear electronic oscillators to minimize the Ising Hamiltonian. This section summarizes the relevant aspects of [9] on the dynamics of OIMs.
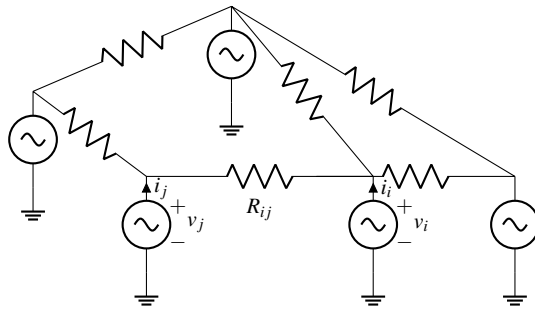
Fig. 1: Example of an Ising Machine connectivity mesh.

1.2.1 High-level Circuit Description

On a network level, OIMs are resistively-connected nonlinear oscillator, as pictured in Figure 1. Each voltage source represents a spin node, or oscillator; the Ising weights are encoded in the conductances $J_{ij} \propto \frac{1}{R_{ij}}$. Concretely, $K_c J_{ij} = \frac{1}{R_{ij}}$, where $K_c > 0$ is a scalar constant representing the coupling strength of the

network. Applying Kirchhoff's Current Law (KCL) at the output of every spin gives us

$$\left(\begin{bmatrix} \sum_j J_{1j} & \cdots & 0 \\ & \ddots & \\ 0 & \cdots & \sum_j J_{nj} \end{bmatrix} - \begin{bmatrix} J_{11} & \cdots & J_{1n} \\ \vdots & \ddots & \vdots \\ J_{n1} & \cdots & J_{nn} \end{bmatrix} \right) \begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix} = \begin{bmatrix} i_1 \\ \vdots \\ i_n \end{bmatrix} ; \text{ or}$$

$$K_c(D - J)\vec{v} = \vec{i},$$

(2)

where $D - J$ is the Laplacian matrix of the network.

The current injection at spin j due to spin i (the transconductance from spin i to spin j) determines the Ising coupling via the oscillator's phase response [11].

As D is diagonal, the $D\vec{v}$ term does not represent any interactions between spins. Instead, it constitutes resistive loading on each spin node, which we incorporate into the node's internal resistance. The $-J\vec{v}$ term encodes the weights of the Ising problem that is being solved; the transconductance matrix J is, in fact, the Ising coupling matrix.

1.2.2 Basic System Dynamics

The oscillators in Figure 1 can be each represented by the differential-algebraic equation:

$$\frac{d}{dt}\vec{q}(\vec{x}) + \vec{f}(\vec{x}) + \vec{b}(t) = \vec{0},$$

(3)

where $\vec{x}$ is the system state and $\vec{b}$ is a small time-varying current injection. They are self-sustaining: given a nonzero initial condition and no input, the state will settle to a periodic steady-state waveform, which we will denote $\vec{x}_s(t)$. A consequence of this feature is that such oscillators will always be unstable.[1]

However, they are amplitude-stable: if an spin node's state is perturbed slightly from its steady state waveform, it will asymptotically return to the same waveform. The perturbation of the state by the input, $\vec{b}(t)$, however, causes a phase shift, $\alpha(t)$, from the nominal $\vec{x}_s(t)$, *i.e.*,

$$\vec{x}(t) = \vec{x}_s(t + \alpha(t)).$$

(4)

As such, we can model the oscillator's response to $\vec{b}(t)$ by the time-varying differential equation

$$\frac{d}{dt}\alpha(t) = \vec{v}^\top(t + \alpha(t))\vec{b}(t).$$

(5)

$\vec{v}$, the Perturbation Projection Vector (PPV), is a function of the differential-algebraic equation that defines the system.

1.2.3 Injection Locking and Oscillator Phase

From here on, let us examine oscillator i in the network from Figure 1. The differential equation of the time-varying oscillator phase shift is now

$$\frac{d}{dt}\alpha_i(t) = \vec{v}_i^\top(t + \alpha_i(t))\vec{b}_i(t).$$

(6)

Let us consider the current injection from one oscillator, *i.e.*, $\vec{b}_i(t)$ is proportional to the output voltage of oscillator j, which can be assumed to be a periodic waveform. For the analysis in this report, we will assume that the state of each oscillator is the scalar sinusoid

$$x_{s,i}(t) = \sin(\omega_i t + \Delta\phi_i(t)),$$

(7)

---

[1]The definition of internal asymptotic stability states than any non-zero initial condition will eventually decay to $\vec{0}$.

where A_i is the amplitude of the waveform, ω_i is the oscillator's natural frequency in radians/second, and $\Delta\phi_i(t)$ is the oscillator phase with respect to some reference waveform at frequency ω_i.

Under this model, the input waveform can be represented by

$$b_i(t) = K_c J_{ij} A_j \sin(\omega_j t + \Delta\phi_j(t)), \tag{8}$$

where $\Delta\phi_j(t)$ is the phase of the input with respect to some reference waveform at frequency ω_j.

In this case, we can apply a form of averaging called Adlerization to produce a differential equation for $\Delta\phi_{i,j}(t)$, the phase difference between oscillators i and j:

$$\frac{d}{dt}\Delta\phi_{i,j}(t) = \omega_i - \omega_j - \omega_i K_c J_{ij} \frac{A_j A_i}{2} \sin(\Delta\phi_{i,j}(t)). \tag{9}$$

The above equation, known as Adler's equation, can be used to explain the the phenomenon of injection locking: when two or more oscillators with natural frequencies that are "close enough" are coupled, then they will "lock" to the same frequency. Mathematically, this means that the oscillators have a constant phase difference, *i.e.*, $\lim_{t\to\infty}\frac{d}{dt}\Delta\phi_{i,j}(t) = 0$. Thus, injection locking can occur when (9) has a stable operating, or equilibrium, point: $\exists\Delta\phi_{i,j}^*$ such that

$$\frac{\omega_j - \omega_i}{\omega_i} = -K_c J_{ij} \frac{A_j A_i}{2} \sin(\Delta\phi_{i,j}^*), \tag{10}$$

$$\left(\frac{d}{d\Delta\phi_{i,j}} \frac{d\Delta\phi_{i,j}(t)}{dt}\right)\Bigg|_{\Delta\phi_{i,j}^*} = -\omega_i K_c J_{ij} \frac{A_j A_i}{2} \cos(\Delta\phi_{i,j}^*) < 0. \tag{11}$$

From these conditions, we determine that injection locking occurs, only if

$$\left|\frac{\omega_j - \omega_i}{\omega_i}\right| < K_c J_{ij} A_j, \tag{12}$$

and there is a single stable operating point (as well as one unstable operating point).

Now, let us consider the full oscillator network. We can generate, via superposition, a set of differential equations for the phase of every oscillator in the network,

$$\frac{d}{dt}\phi_i(t) = \omega_i - \omega_0 - \frac{K_c A_i}{2}\omega_0\sum_j A_j \sin(\phi_i(t) - \phi_j(t)), \forall i, \tag{13}$$

where ω_0 is the natural frequency of the entire network (assuming that injection locking occurs, the network locks to frequency ω_0). This set of differential equations is known as the Kuramoto equations. Note that, in the Kuramoto equations, we drop the Δ in front of each phase, and $\phi_i(t)$ is understood to be the phase of oscillator i with respect to a reference waveform at frequency ω_0.

For ease of analysis, we will make a few simplifying assumptions. First, we will assume that the variation in natural frequency between the oscillators is negligible such that we can make the approximation $\omega_i \approx \omega_0$, $\forall i$. We will also assume that $\omega_0 \frac{A_i A_j}{2} = 1$, for every pair of oscillators.[2]

With these assumptions, the Kuramoto equations become

$$\frac{d}{dt}\phi_i(t) = -K_c\sum_j \sin(\phi_i(t) - \phi_j(t)), \forall i. \tag{14}$$

---

[2] In other words, differences in oscillator amplitude are negligible and any constant scaling factor on the summation in (13) is incorporated into K_c.

1.2.4 The Lyapunov Function and Sub-harmonic Injection Locking

The Kuramoto equations have a Lyapunov function, *i.e.*, a function with a nonpositive time derivative:

$$E(\vec{\phi}(t)) = -\frac{1}{2}\sum_i \sum_{j\neq i} J_{ij}\cos(\phi_i(t) - \phi_j(t)), \tag{15}$$

where $\vec{\phi} \triangleq \begin{bmatrix} \phi_1 & \cdots & \phi_n \end{bmatrix}^\top$. Computing the gradient,

$$\frac{d}{d\vec{\phi}}E(\vec{\phi}(t)) = -\frac{1}{K_c}\frac{d}{dt}\vec{\phi}(t). \tag{16}$$

So, the time derivative of the Lyapunov function is

$$\frac{d}{dt}E(\vec{\phi}(t)) = \left(\frac{d}{d\vec{\phi}}E(\vec{\phi}(t))\right)^\top \left(\frac{d\vec{\phi}}{dt}\right) = -\frac{1}{K_c}\left\|\frac{d\vec{\phi}}{dt}\right\|_2 \leq 0. \tag{17}$$

As the time derivative of $E(\vec{\phi}(t))$ is ≤ 0 and $E(\vec{\phi}(t))$ is bounded from below,[3] the system will asymptotically settle to local minima of $E(\vec{\phi}(t))$.

If all phases are restricted to be either 0 or π, then

$$\cos(\phi_i(t) - \phi_j(t)) = \begin{cases} 1, & \phi_i(t) = \phi_j(t) \\ -1, & \phi_i(t) \neq \phi_j(t) \end{cases}.$$

If we say that a spin has value $s_i = 1$ when $\phi_i(t) = 0$ and $s_i = -1$ when $\phi_i(t) = \pi$, then

$$\cos(\phi_i(t) - \phi_j(t)) = s_i s_j,$$

and (15) exactly matches the Ising Hamiltonian, (1).

This phase bistability is achieved through a phenomenon called sub-harmonic injection locking (SHIL). In brief, when an input to an oscillator is at approximately twice its natural frequency, its frequency will settle to half the input frequency.

Let us again consider Adler's equation, this time with input $b_i(t) = K_s\sin(2\omega_s t)$, where $\omega_s \approx \omega_0$. This $b_i(t)$ is known as the SYNC signal.

$$\frac{d}{dt}\Delta\phi_i(t) = \omega_0 - \omega_s - \omega_0 K_s \sin(2\Delta\phi_i(t)), \tag{18}$$

where $\Delta\phi_i(t)$ is the phase difference between oscillator i and a reference waveform at frequency ω_s.

Following the same analysis as in (10) and (11), we see that there are two stable operating points at $\Delta\phi_i^*$ and $\Delta\phi_i^* + \pi$.[4] Defining the reference signal such that $\Delta\phi_i^* = 0$, $\Delta\phi_i(t)$ will lock to either 0 or π.

Returning to analysis of the oscillator network, we consider the Kuramoto equations under both fundamental harmonic injection locking (from adjacent oscillators) and SHIL (from the SYNC signal):

$$\frac{d}{dt}\phi_i(t) = -K_c\sum_j \sin(\phi_i(t) - \phi_j(t)) - K_s\sin(2\phi_i(t)), \forall i, \tag{19}$$

where we assume $\omega_s = \omega_0$ and $\phi_i(t)$ is understood to be represent $\Delta\phi_i(t)$. Since the input to each oscillator involves terms at ω_0 and $2\omega_0$, the analysis of injection locking is more complicated than what was presented

---

[3] By the triangle inequality, $E(\vec{\phi}(t)) \geq -K_c\sum_i\sum_{j\neq i}|J_{ij}|$.

[4] As long as $\left|\frac{\omega_0 - \omega_s}{\omega_0}\right| < K_s$.

in this section. Further analysis of the Kuramoto system with SHIL is presented in Section 3. For the purpose of this section, it suffices to know that, if K_s is high enough with respect to K_c, that SHIL occurs and each phase is close to either 0 or π.

The Lyapunov function corresponding to (19) is

$$E(\vec{\phi}(t)) = -\frac{1}{2}\sum_i\sum_{j\neq i}J_{ij}\cos(\phi_i(t) - \phi_j(t)) - \frac{K_s}{2K_c}\cos(2\phi_i(t)). \tag{20}$$

Under SHIL, this reduces to

$$E(\vec{\phi}(t)) = -\frac{1}{2}\sum_i\sum_{j\neq i}J_{ij}s_is_j - \frac{K_s}{2K_c}, \tag{21}$$

which is equal to the Ising Hamiltonian, with a constant offset. As such, global minimization of the Lyapunov function is equivalent to global minimization of the Ising Ha miltonian.

1.2.5 Local Minimization of the Ising Hamiltonian

The existence of a Lyapunov function for the Kuramoto equations means that the system will settle to local minima of (20). However, as the Ising Hamiltonian defines a discrete minimization problem, it may not have the same local minima as the (continuous) Lyapunov function. In fact, for K_s such that SHIL occurs, every "grid point," *i.e.*, $\vec{\phi} \in \{0, \pi\}^n$ (where n is the number of oscillators) is a stable operating point of the Kuramoto equations and thus a local minimum of the Lyapunov function. Only some of these points, however, will be discrete local minima of the Ising Hamiltonian. One key direction of future exploration is in the relationship between local minimization of the Lyapunov function and the Ising Hamiltonian, as the strength of the SYNC signal is varied.

2 Dense as Sparse: Sparsifying Dense On-chip Ising Machines

2.1 Introduction

The structure of the graph defining an Ising problem varies significantly from problem to problem. Notably, depending on the application, the Ising graph may be densely or sparsely connected. In other words, either there is all-to-all (or nearly all-to-all) connectivity between the spins, or each spin is connected to only a few other spins. Dense connectivity is generally difficult to achieve at scale in hardware; this is especially true for IC implementations of Ising machines [11, 13] as the number of spins increases, due to the difficulty of routing $O(n^2)$ connections between n spins. This constitutes a serious barrier to the development of scalable on-chip Ising machines for problems that require dense connections. This problem does not appear to have been considered, far less addressed, in prior work on the subject, despite how important it is for practical realization of Ising machines.

One such dense problem, important in telecommunications, is the MU-MIMO (Multi-User Multiple-Input-Multiple-Output) detection problem [14, 15]. The efficacy of Ising machines for MU-MIMO has been demonstrated in simulation [16], but building hardware to realize this promise, especially for larger problems, runs into the issue of implementing dense connectivity. It is essential to achieve dense Ising connectivity at scale if practical Ising machine ICs that solve, *e.g.*, the MU-MIMO problem, are to become a reality.

In this project, we devise a highly accurate method to represent a useful class of dense Ising problems using only sparse on-chip resistive networks. Such sparse networks are practical to implement at scale on ICs. Our method exploits low-rank structure in the matrix of Ising couplings (*i.e.*, weights of the Ising graph). This low-rank structure results in a singular value decomposition (SVD) [17] where most singular values are negligible. As a result, the matrix can be represented, with very little error, using a truncated SVD that requires far fewer nonzero values than the original dense matrix needs.

The next task is to map this SVD approximation onto a sparse on-chip resistive network. We first devise a technique for doing so when all required resistors are positive-valued. We use a small number of Kirchhoff's Current Law (KCL)-enforcing auxiliary nodes that form bipartite connections with the spins. We then show the current-voltage relationship between the spins is equivalent to that from the truncated SVD. There are as many auxiliary nodes as non-negligible singular values, so this constitutes a sparse connectivity fabric when the Ising coupling matrix has low-rank structure. We devise a mathematical procedure for generating this sparse bipartite resistive network from a truncated SVD of the Ising coupling matrix.

Typical Ising problems require a mix of positive and negative resistors,[5] the latter being complicated to implement physically. We show that even negative values in the truncated SVD mapping can be elegantly implemented using only positive resistors. If each spin has a differential output, any connectivity pattern can be realized using only positive resistors between the auxiliary nodes and either the positive or negative spins. Our sparse synthesis algorithm is thus able to treat any Ising coupling matrix using only positive physical resistors.

We validate DaS on a range of MU-MIMO detection problems, produced using MATLAB's Phased Array System Toolbox [19, 20]. For large problem sizes, the MU-MIMO Ising coupling matrix typically has low effective rank, *i.e.*, many negligible singular values. We achieve density reductions of ~ 80 to 95%, depending on the number of scatterers that interfere with the communication link. The SVD approximation has very little impact on the coupling matrix; the absolute sum of the element-wise approximation error is less than 10^{-8} times the absolute sum of the Ising weights.

We also illustrate a simple layout technique, based on a crossbar-style physical architecture, that is suitable for implementing the sparse, bipartite resistive networks generated by DaS. The technique allows programmable physical interconnectivity, *i.e.*, any arbitrary Ising problem with a low-rank structure can be programmed onto the chip.

---

[5]Negative resistors [18] have a negative R in the Ohm's Law relation $V = IR$.

2.2 Background and Overview

2.2.1 The MU-MIMO Detection Problem

The MU-MIMO (Multi-User Multiple-Input-Multiple-Output) detection problem is an important problem in telecommunications that can be mapped to the Ising problem. In modern wireless communication, multiple users, each with one or more transmit antennas, simultaneously transmit to multiple receive antennas. The received signals are, therefore, a noisy combination of each user's transmitted symbols. Because the transmitted symbols are discrete, recovering them from the received signals turns out to be a hard combinatorial optimization problem, *i.e.*, the discrete maximum-likelihood estimation (MLE) problem. The subsequent paragraphs summarize how to map the MU-MIMO problem to the Ising problem, allowing it to be solved by an Ising machine [15, 16].

Consider a MU-MIMO problem with N_t transmitters and N_r receivers. Let the transmitted signal be denoted by a vector $\vec{x}$ of length N_t, such that $x_i \in \{\pm 1\}$. The resulting vector of received signals is

$$\vec{y} = H_C \vec{x} + \vec{n}, \tag{22}$$

where $H_C \in \mathbb{C}^{N_r \times N_t}$ is the channel transmission matrix and $\vec{n} \in \mathbb{C}^{N_r}$ represents additive white Gaussian noise (AWGN).

$H_C, \vec{y}$, and $\vec{n}$ are complex, so we represent them using real numbers by vertically stacking their real and imaginary components:

$$H_{C,\mathfrak{Re}} = \begin{bmatrix} \mathfrak{Re}\{H_C\} \\ \mathfrak{Im}\{H_C\} \end{bmatrix} \in \mathbb{R}^{2N_r \times N_t}, \ \vec{y}_{\mathfrak{Re}} = \begin{bmatrix} \mathfrak{Re}\{\vec{y}\} \\ \mathfrak{Im}\{\vec{y}\} \end{bmatrix} \in \mathbb{R}^{2N_r},$$

$$\vec{n}_{\mathfrak{Re}} = \begin{bmatrix} \mathfrak{Re}\{\vec{n}\} \\ \mathfrak{Im}\{\vec{n}\} \end{bmatrix} \in \mathbb{R}^{2N_r}. \tag{23}$$

In the detection problem, we would like to determine the symbols $\vec{x}$ that minimize the mean-squared error, given by

$$||H_{C,\mathfrak{Re}}\vec{x} - \vec{y}_{\mathfrak{Re}}||^2 = (H_{C,\mathfrak{Re}}\vec{x} - \vec{y}_{\mathfrak{Re}})^T (H_{C,\mathfrak{Re}}\vec{x} - \vec{y}_{\mathfrak{Re}}). \tag{24}$$

To reach an Ising formulation for this minimization problem, we want to write $H_{C,\mathfrak{Re}}\vec{x} - \vec{y}_{\mathfrak{Re}}$ as a single matrix-vector product. To do so, we define

$$\hat{H}_{C,\mathfrak{Re}} = \begin{bmatrix} H_{C,\mathfrak{Re}} & -\vec{y}_{\mathfrak{Re}} \end{bmatrix}, \text{ and } \hat{x} = \begin{bmatrix} \vec{x} \\ 1 \end{bmatrix}. \tag{25}$$

With these definitions, (24) becomes

$$\hat{x}^T \hat{H}_{C,\mathfrak{Re}}^T \hat{H}_{C,\mathfrak{Re}} \hat{x}. \tag{26}$$

As the objective of the Ising problem is to minimize

$$-\frac{1}{2}\sum_{i,j} J_{ij} s_i s_j = -\frac{1}{2}\vec{s}^T J \vec{s}, \ s_i \in \{\pm 1\}, \tag{27}$$

the MU-MIMO problem matches the Ising problem if

$$J = -2\hat{H}_{C,\mathfrak{Re}}^T \hat{H}_{C,\mathfrak{Re}}, \text{ and } \vec{s} = \hat{x}. \tag{28}$$

Ising machines, therefore, can be used to solve the MU-MIMO detection problem.

The channel transmission matrix, and therefore the Ising coupling matrix, is invariably dense. Although on-chip implementations of dense Ising solvers are viable for smaller problems, routing becomes infeasible as the problems scale. Therefore, we wish to sparsify the MU-MIMO problem to make it more amenable to large on-chip Ising solver implementations.

9

2.2.2 Densely-Connected On-chip Ising Machines

Analog Ising machine connectivity is implemented as resistive meshes, as described in Section 1.2.1 [11–13]. If there is all-to-all connectivity, this mesh has $\frac{n(n-1)}{2}$ resistors.

Our goal is to utilize low-rank structures in the J matrix to represent the current-voltage relationship in (2) sparsely, and to translate the sparse mathematical representation to a physically sparse network of resistive connections. For the purposes of this section, we will ignore the constant scaling factor K_c and instead match

$$(D-J)\vec{v} = \vec{i}. \tag{29}$$

2.2.3 Overview of Our Method

We show, in Section 2.3, that a singular value decomposition (SVD) of a low-rank dense coupling matrix can yield a sparse representation. To reach this sparse representation, we compute the truncated SVD of coupling matrix $J \in \mathbb{R}^{n \times n}$, $J = U_m \Sigma_m V_m^T$. $U_m \in \mathbb{R}^{n \times m}$, $\Sigma_m \in \mathbb{R}^{m \times m}$, and $V_m \in \mathbb{R}^{m \times n}$, where m is the number of singular values above a certain threshold. This threshold can be chosen such that the Ising coupling matrix from the truncated SVD is almost identical to the Hamiltonian of the original coupling matrix. If $m \ll n$, U_m, Σ_m, and V_m have far fewer values than those needed to represent a dense coupling matrix; this is the provenance of the desired sparsification.

To use this sparse representation for on-chip Ising solvers, it is crucial to map it onto a sparse resistive network. This process, however, is non-trivial, especially if there are negative entries in U_m and V_m and if $U_m \neq V_m$ (*i.e.*, J is not positive semi-definite), both of which are invariably the case in problems of interest. We devised a general procedure to translate the SVD representation to a sparse resistive network via a set of m auxiliary nodes, as described in Section 2.4. We also show, in Section 2.4.5, the basic idea behind how our scheme can be laid out using a programmable crossbar-style architecture.

In Section 2.5, we show that Ising coupling matrices for many MU-MIMO problems have a low-rank structure, especially for large numbers of transmitters and receivers. As a result, they are amenable to sparsification using DaS. In general, the sparsity of a MU-MIMO problem increases as the size of the problem (number of spins) increases, as the number of scatterers obstructing the communication channel decreases, or as the spacing between individual transmitters or receivers decreases. That sparsity increases as problem sizes increase is, in particular, a desirable feature from the standpoint of scalable Ising machine implementation.

2.3 Low-Rank Decompositions of Resistive Connections

The Ising coupling matrix J may have low effective rank, *i.e.*, a large proportion of its singular values may be much smaller than its largest singular value. Assuming that J has m non-negligible singular values, its truncated SVD [21] is $J \approx U_m \Sigma_m V_m^T$, where $U_m, V_m \in \mathbb{R}^{n \times m}$ and $\Sigma_m \in \mathbb{R}^{m \times m}$.

As J is a dense, symmetric matrix with zeros on the diagonals, we require $\frac{n(n-1)}{2}$ values to fully characterize J. Using the truncated SVD, we can represent J using $m(2n+1)$ values (mn each for U_m and V_m and m for Σ_m). If $m \ll n$, then this is a sparse representation: we can represent J using much fewer values than $\frac{n(n-1)}{2}$, with high accuracy, as seen in the examples in Section 2.5. Here, we define sparsity as the number of values (later, number of resistors) needed to represent J accurately compared to the $\frac{n(n-1)}{2}$ resistors required in Section 2.2.2.

2.3.1 Improved Sparsity of the Truncated SVD for Positive and Negative Semi-Definite Matrices

We can decrease the number of values required to represent the truncated SVD of J by making J positive semi-definite (PSD) or negative semi-definite (NSD), *i.e.*, having only positive or only negative eigenvalues [22]. If J is PSD, then its truncated SVD is $J = U_m \Sigma_m U_m^T$. A symmetric NSD matrix can be represented as the negative of a symmetric PSD matrix, so its truncated SVD is $J = -U_m \Sigma_m U_m^T$.

As V_m is the same as or the negative of U_m, J can be represented using only $m(n+1)$ values.

While J is always symmetric in Ising problems, it is generally neither PSD nor NSD. We can, however, add a constant to the diagonals of J to get a PSD or NSD matrix. For instance,

$$J_{\text{PSD}} \triangleq J + \alpha I, \text{ or } J_{\text{NSD}} \triangleq J + \beta I \tag{30}$$

Additions to the diagonal of J do not affect the Ising problem, since the diagonal contributes nothing to the Ising Hamiltonian (1). So, we can replace J with J_{PSD} or J_{NSD} without changing the parameters of the Ising problem.

Consider the case where J is PSD. We wish to minimize the magnitude of α to increase the likelihood that J_{PSD} has low effective rank. As shown below, we achieve this by setting $\alpha = -\lambda_{\min}(J)$, where $\lambda_{\min}(J)$ is the smallest eigenvalue of J.

A matrix $J \in \mathbb{R}^{n \times n}$ is PSD if and only if $\vec{x}^T J \vec{x} \geq 0, \forall \vec{x} \in \mathbb{R}^n$. Without loss of generality, let us consider $\vec{x}$ such that $||\vec{x}||_2 = 1$. Then, it is guaranteed that $\vec{x}^T J \vec{x} \geq \lambda_{\min}(J)$, with equality if $\vec{x}$ is in the span of the eigenvector corresponding to $\lambda_{\min}(J)$.

Setting $\alpha = -\lambda_{\min}(J)$,

$$\vec{x}^T J_{\text{PSD}} \vec{x} = \vec{x}^T J \vec{x} + \alpha \vec{x}^T \vec{x} \geq \lambda_{\min}(J) + \alpha = 0. \tag{31}$$

Thus, $J + \alpha I$ is PSD when $\alpha = -\lambda_{\min}(J)$. This is the minimum value of α such that $J + \alpha I$ is PSD: if $\alpha < -\lambda_{\min}(J)$ and $\vec{x}$ is in the span of the eigenvector corresponding to $\lambda_{\min}(J)$,

$$\vec{x}^T J_{\text{PSD}} \vec{x} = \vec{x}^T J \vec{x} + \alpha \vec{x}^T \vec{x} = \lambda_{\min}(J) + \alpha < 0. \tag{32}$$

This scheme is generally effective if the largest negative eigenvalue of J is sufficiently smaller than its largest positive eigenvalue. Otherwise, adding αI to J will result in a substantial increase in m, making the truncated SVD no longer sparse.

Likewise, to make J NSD, we set $\beta = -\lambda_{\max}(J)$. In general, making J NSD improves the sparsity if the low-rank decomposition of the largest negative eigenvalue of J is much larger than the largest positive eigenvalue.

2.3.2 Scaling U and V for Increased Sparisty

As we will see in Section 2.4, our hardware implementation requires fewer resistors than the $m(2n+1)$ (for an arbitrary matrix) or $m(n+1)$ (for a PSD or NSD matrix) values in the truncated SVD.

Let us consider an arbitrary, *i.e.*, not PSD or NSD, Ising coupling matrix. Instead of decomposing the coupling matrix as $J \approx U_m \Sigma_m V_m^T$, we will have a relation of the form $J \approx G_{\text{in}} D_c G_{\text{out}}^T$, where D_c is the diagonal matrix of the column sums of G_{out} (see Section 2.4, specifically Section 2.4.2 for implementation details, as well as motivation behind the naming of these matrices). The columns of V_m and Σ_m are scaled to obtain G_{out} and D_c such that $U_m \Sigma V_m^T = G_{\text{in}} D_c G_{\text{out}}^T$ and D_c is the diagonal matrix of the column sums of G_{out}. So, we represent J using only $2mn$ values, as D_c is obtained from G_{out}.

For a PSD matrix, we represent J_{PSD} as GD_cG^T, where the columns of G are scaled such that $U_m \Sigma_m V_m^T = GD_cG^T$ and D_c consists of the column sums of G. In doing so, we represent J_{PSD} using mn values. Likewise, we can also represent an NSD matrix using mn values.

2.3.3 Impact of the Truncated SVD Approximation on the Ising Hamiltonian

In choosing m, the number of singular values to use, we must examine the impact of representing J with a truncated SVD on the Ising Hamiltonian. Let $\hat{J} = U_m \Sigma_m V_m^T$ be the rank-m approximation of the coupling matrix. Define the Ising Hamiltonians for the original and SVD-approximated coupling matrices as

$$H(\vec{s}) = -\frac{1}{2} \sum_{i \neq j} J_{ij} s_i s_j \text{ and } \hat{H}(\vec{s}) = -\frac{1}{2} \sum_{i \neq j} \hat{J}_{ij} s_i s_j, \tag{33}$$

where $\vec{s}, s_i \in \{-1, 1\}$ is a vector of spin values and J_{ij} is the element of J at row i, column j.

The error between $H(\vec{s})$ and $\hat{H}(\vec{s})$ can be bounded as follows:

$$|H(\vec{s}) - \hat{H}(\vec{s})| = \left| -\frac{1}{2}\sum_{i \neq j}(J_{ij} - \hat{J}_{ij})s_i s_j \right| \leq \frac{1}{2}\sum_{i \neq j}|(J_{ij} - \hat{J}_{ij})s_j s_j| = \frac{1}{2}\sum_{i \neq j}|J_{ij} - \hat{J}_{ij}|. \tag{34}$$

Thus, if we would like to choose m to enforce some bound on the approximation error of the Ising Hamiltonian, $|H(\vec{s}) - \hat{H}(\vec{s})| < \varepsilon$, we can choose the minimum m such that

$$\frac{1}{2}\sum_{i \neq j}|J_{ij} - \hat{J}_{ij}| = \frac{1}{2}\sum_{i \neq j}|J_{ij} - (U_m \Sigma_m V_m^T)_{ij}| < \varepsilon. \tag{35}$$

2.4 Low-Rank Decompositions as Sparse Resistive Networks

In this section, we devise a method to represent the truncated SVD of a matrix using a resistive network. This method involves two sets of nodes with bipartite connections: spins and auxiliary nodes, as shown in Figure 2. There are n spin nodes, where n is the size of the Ising graph, and m auxiliary nodes, where m is the effective rank of J, as defined in Section 2.3.

The resistor between spin j and auxiliary node k, R_{jk}, has conductance $G_{jk} = R_{jk}^{-1}$.

Define the transconductance matrix

$$G \triangleq \begin{bmatrix} G_{11} & \cdots & G_{1m} \\ \vdots & \ddots & \vdots \\ G_{n1} & \cdots & G_{nm} \end{bmatrix}, \tag{36}$$

Fig. 2: Bipartite network.

and define D_c and D_r as the diagonal matrices of the row and column sums of G, respectively. As in Section 2.2.2 and Figure 1, define i_j to be the current leaving the jth spin node.

To see how this formulation encodes the truncated SVD of a matrix, let us examine the equations produced by applying KCL at the the auxiliary and spin nodes. At the auxiliary nodes,
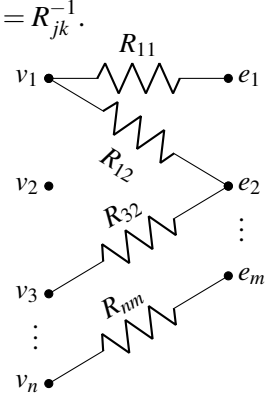
$$\begin{bmatrix} \sum_j G_{j1} \cdots & 0 \\ & \ddots & \\ 0 & \cdots \sum_j G_{jm} \end{bmatrix}\begin{bmatrix} e_1 \\ \vdots \\ e_m \end{bmatrix} - \begin{bmatrix} G_{11} & \cdots & G_{n1} \\ \vdots & \ddots & \vdots \\ G_{1m} & \cdots & G_{nm} \end{bmatrix}\begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix} = \vec{0}; \text{ or } \vec{e} = D_c^{-1}G^T\vec{v}. \tag{37}$$

At the spin nodes,

$$\begin{bmatrix} \sum_k G_{1k} \cdots & 0 \\ & \ddots & \\ 0 & \cdots \sum_k G_{nk} \end{bmatrix}\begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix} - \begin{bmatrix} G_{11} & \cdots & G_{1m} \\ \vdots & \ddots & \vdots \\ G_{n1} & \cdots & G_{nm} \end{bmatrix}\begin{bmatrix} e_1 \\ \vdots \\ e_m \end{bmatrix} = \begin{bmatrix} i_1 \\ \vdots \\ i_n \end{bmatrix}; \text{ or } D_r\vec{v} - G\vec{e} = \vec{i}. \tag{38}$$

Using (37) in (38), we obtain

$$\vec{i} = (D_r - GD_c^{-1}G^T)\vec{v}. \tag{39}$$

As in Section 2.2.2, we incorporate the loading terms, $D_r\vec{v}$, into each spin unit.[6] Comparing (39) with (29), we have $J = GD_c^{-1}G^T$. This matches the form of the SVD of a positive semi-definite matrix, $J_{\text{PSD}} = U_m\Sigma_m U_m^T$.

---

[6]These loading terms do not affect the Ising problem or its Hamiltonian, but circuits implementing Ising spins need to take the loading into account.

2.4.1 Implementation of Negative Entries in G

Although U_m of the SVD of J can contain positive and negative entries, G is comprised of conductances and is thus restricted to positive values.[7] In order to properly implement the SVD, we must revise our formulation.

To do so, we use differential spin nodes, *i.e.*, each spin has a positive and negative output. In the bipartite network (Figure 2), we replace each spin v_j with the spin pair v_j and $-v_j$. A connection between $\pm v_j$ and auxiliary node e_k is positive if there is a resistor $R_{jk+} = G_{jk+}^{-1}$ between $v_{j+} = v_j$ and e_k and negative if there is a resistor R_{jk-} between $v_{j-} = -v_j$ and e_k.

The transconductance matrix is now

$$
G \triangleq G_+ - G_- =
\begin{bmatrix}
G_{11+} & \cdots & G_{1m+} \\
\vdots & \ddots & \vdots \\
G_{n1+} & \cdots & G_{nm+}
\end{bmatrix}
-
\begin{bmatrix}
G_{11-} & \cdots & G_{1m-} \\
\vdots & \ddots & \vdots \\
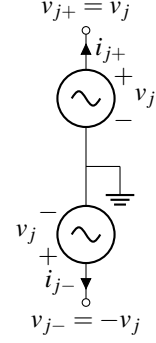G_{n1-} & \cdots & G_{nm-}
\end{bmatrix}.
$$

We define $D_c = D_{c+} + D_{c-}$, where D_{c+} and D_{c-} are the column sums of G_+ and G_-, respectively. Equivalently, $D_r = D_{r+} + D_{r-}$, where D_{r+} and D_{r-} are the row sums of the respective transconductance matrix.

Fig. 3: Differential spin node.

For this setup, we examine the differential current[8] $\vec{i}_d = \vec{i}_+ - \vec{i}_-$, where $\vec{i}_+$ is the vector of currents leaving the positive spin nodes and $\vec{i}_-$ is the vector of currents leaving the negative spin nodes. Performing KCL at the auxiliary nodes,

$$
D_c \vec{e} - (G_+^T \vec{v} - G_-^T \vec{v}) = D_c \vec{e} - G^T \vec{v} = \vec{0}; \text{ or } \vec{e} = D_c^{-1} G^T \vec{v}. \tag{40}
$$

At the positive spin outputs, KCL produces the equation

$$
\vec{i}_+ = D_{r+} \vec{v} - G_+ \vec{e}, \tag{41}
$$

and, at the negative outputs, we have

$$
\vec{i}_- = -D_{r-} \vec{v} - G_- \vec{e}. \tag{42}
$$

Taking the differential current,

$$
\vec{i}_d = (D_{r+} + D_{r-}) \vec{v} - (G_+ - G_-) \vec{e} = D_r \vec{v} - G \vec{e} = (D_r - G D_c^{-1} G^T) \vec{v}. \tag{43}
$$

This is equivalent to (39), except that G can now hold negative entries. As $G = G_+ - G_-$, positive entries in G correspond to positive connections and negative entries to negative connections.

Now, we must show that setting the differential current as in (43) actually relates to the weights of the Ising Hamiltonian. To do so, we linearize the positive and negative spin units around their steady state waveforms, with $\vec{i}_+$ and $\vec{i}_-$ as current inputs. We examine the pair of spin nodes v_{j+}, v_{j-}, looking at the current inputs i_{j+} and i_{j-} in superposition.

First, we will determine the effect of i_{j+} on $v_j(t)$, setting i_{j-} to 0. To demonstrate the core concept, we represent the positive spin by the differential equation

$$
\frac{d}{dt} v_j(t) = f(v_j(t)) + b i_{j+}(t), \tag{44}
$$

---

[7]While it is possible to approximate negative resistors using nonlinear electronic circuits [18], these greatly increase the complexity of hardware implementation and cannot be perfectly linear.

[8]It is the differential current that constitutes the coupling input into a spin from another, as we show later in this section.

for some nonlinear function $f(\cdot)$ and constant b.[9] Assume that the spin unit's differential equation has steady state $v_{js}(t)$. We then linearize (44) around this steady state to get

$$\frac{d}{dt}\Delta v_j(t) = \frac{d}{dt}(v_j(t) - v_{js}(t)) = \left.\frac{\partial f}{\partial v_j}\right|_{v_{js}(t)} \Delta v_j(t) + bi_{j+}(t), \tag{45}$$

where $\Delta v_j(t)$ represents the change in spin voltage from the steady state due to small perturbation $bi_{j+}(t)$.

Now, we examine the effect of i_{j-}, setting i_{j+} to 0. Assume the differential equation governing the negative spin is

$$\frac{d}{dt}(-v_j(t)) = f(-v_j(t)) + bi_{j-}(t); \text{ or } \frac{d}{dt}v_j(t) = -f(-v_j(t)) - bi_{j-}(t). \tag{46}$$

Linearizing around the steady state $-v_{js}(t)$,

$$\frac{d}{dt}\Delta v_j(t) = -\left(-\left.\frac{\partial f}{\partial v_j}\right|_{-v_{js}(t)}\right)\Delta v_j(t) - bi_{j-}(t) = \left.\frac{\partial f}{\partial v_j}\right|_{-v_{js}(t)}\Delta v_j(t) - bi_{j-}(t). \tag{47}$$

Applying superposition, we add (45) and (47) to get

$$\frac{d}{dt}\Delta v_j(t) = \frac{\left.\frac{\partial f}{\partial v_j}\right|_{v_{js}(t)} + \left.\frac{\partial f}{\partial v_j}\right|_{-v_{js}(t)}}{2}\Delta v_j(t) + b\frac{i_{j+} - i_{j-}}{2} = \frac{\left.\frac{\partial f}{\partial v_j}\right|_{v_{js}(t)} + \left.\frac{\partial f}{\partial v_j}\right|_{-v_{js}(t)}}{2}\Delta v_j(t) + b\frac{i_{jd}}{2}, \tag{48}$$

where i_{jd} is the differential current of spin pair v_j, $-v_j$.

Therefore, the effective current input that determines the dynamics of the voltage response $\Delta v_j(t)$ of each differential spin unit is proportional to the differential current i_{jd}.

2.4.2 Ensuring Column Sums of G Match Singular Values

To make the current-voltage relationship in (39) match that of dense connectivity mesh in (29), we must have $GD_c^{-1}G^T = U_m\Sigma_mU_m^T$, where $J = U_m\Sigma_mU_m^T$ for a PSD Ising coupling matrix. Now that we are able to set G to match U_m, we must ensure that D_c^{-1} matches Σ_m, the diagonal matrix of singular values.

The most straightforward solution is to add a resistor from each auxiliary node to ground. Denote the resistor from node e_k to ground by R_{e_k}, its conductance by G_{e_k}, and the diagonal matrix of these conductances by D_e. Now, the KCL equations at the auxiliary nodes are

$$D_c\vec{e} + D_e\vec{e} - G^T\vec{v} = (D_c + D_e)\vec{e} - G^T\vec{v} = \vec{0}; \text{ or } \vec{e} = (D_c + D_e)^{-1}G^T\vec{v}. \tag{49}$$

The equation for the differential current $\vec{i_d}$ is now

$$\vec{i_d} = (D_r - G(D_c + D_e)^{-1}G^T)\vec{v}. \tag{50}$$

Thus, we can set D_e such that $(D_c + D_e)^{-1} = \Sigma_m$, or

$$G_{e_k} + \sum_j G_{jk+} + \sum_j G_{jk-} = \sigma_k^{-1}, \tag{51}$$

where σ_k is the kth singular value of J.

If $\sigma_k^{-1} > \sum_j G_{jk+} + \sum_j G_{jk-}$, then this method allows (50) to match the truncated SVD of J without any added resistive loading on the spin nodes. However, if $\sigma_k^{-1} < \sum_j G_{jk+} + \sum_j G_{jk-}$, then G_{e_k} must be a negative resistor. This, while possible to implement, is design- and area-intensive.

---

[9]Any analog Ising machine spin can be represented by such a differential equation [23].

Alternatively, we can scale the elements of G such that $GD_c^{-1}G^T = U_m\Sigma_m U_m^T$, without any added resistors. Let us denote the column sums of U_m by $D_{c,U}$. We will set the transconductance matrix G to $U_m D_\beta$, where D_β is a diagonal matrix of scaling factors $\text{diag}(\beta_1, \ldots, \beta_m) = D_\beta$. In this configuration, β_k multiplies the kth column of U_m, so the column sums of G are $D_c = D_{c,U}D_\beta$.

To have $GD_c^{-1}G$ match $U_m\Sigma_m U_m^T$, we must have

$$U_m D_\beta D_\beta^{-1} D_{c,U}^{-1} D_\beta U_m^T = U_m D_{c,U}^{-1} D_\beta U_m^T = U_m \Sigma_m U_m^T, \tag{52}$$

so we must define D_β such that

$$D_{c,U}^{-1} D_\beta = \Sigma_m \implies D_\beta = D_{c,U}\Sigma_m. \tag{53}$$

One disadvantage of this method is that, if the elements of $D_{c,U}\Sigma_m$ are too large, it could lead to excessive loading on some spin units. This can be mitigated by scaling J by some constant, positive factor γ.[10] If $\Sigma_m' = \gamma\Sigma_m$ is the singular value matrix after scaling, we have $D_\beta = \gamma D_{c,U}\Sigma_m$. The resistive loading term on the spins is $D_r\vec{v}$, so we can choose γ to be small enough such that the row sums of $G = \gamma U_m D_{c,U}\Sigma_m$ do not constitute excessive loading.

2.4.3 Representing Negative Semi-Definite (NSD) Matrices

If the coupling matrix is negative semi-definite, we have $J_{\text{NSD}} = -U_m\Sigma_m U_m$. So, at the spin nodes, we must have the current-voltage relationship

$$\vec{i}_d = (D - J_{\text{NSD}})\vec{v} = (D + U_m\Sigma_m U_m^T)\vec{v}, \tag{54}$$

where D is a diagonal loading matrix.
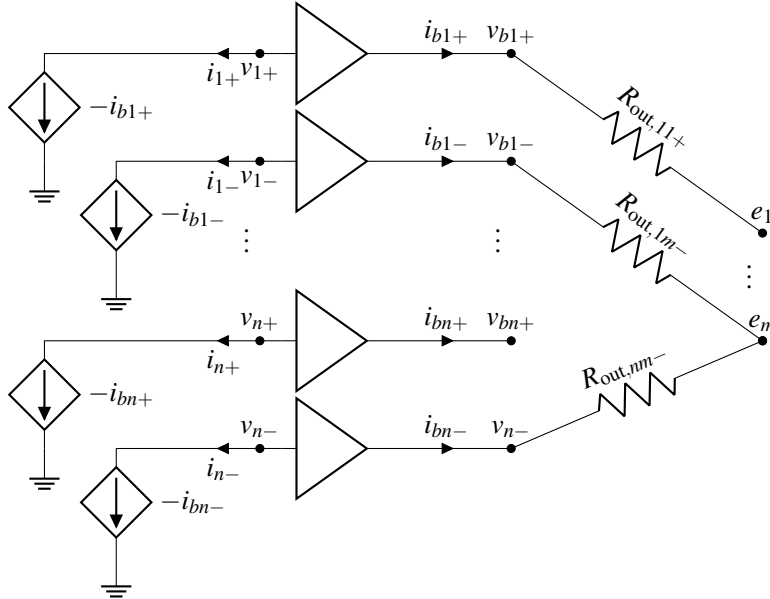
Fig. 4: Formulation for representing NSD matrices, with buffers and current-controlled current sources at each spin node.

To implement this formulation as a resistive network, we add unity gain buffers between the spin nodes and the resistors connecting them to the auxiliary nodes. We refer to the outputs of these buffers as "buffered nodes". The current output of these buffers can then be reflected to the spin nodes via current mirrors, such that the current leaving spin node v_{j+} is $-i_{bj+}$. For the purpose of this report, we represent this process using

---

[10]As this modification scales the Ising Hamiltonian by a constant, positive number, it does not affect Hamiltonian minimization.

current-controlled current sources rather than showing the detailed transistor-level implementation we have devised. As the number of buffers and current-controlled current sources scales linearly with the number of spins, they do not significantly impede on-chip layout.

Applying KCL to the buffered nodes in Figure 4, we get the same result as in (43):

$$\vec{i}_{b+} - \vec{i}_{b-} = (D_r - GD_c^{-1}G^T)\vec{v}_b, \tag{55}$$

where D_r and G are defined as in Section 2.4.1, $\vec{i}_{b+}$ is the vector of currents leaving the positive buffered spin nodes, and $\vec{i}_{b-}$ is the currents leaving the negative buffered spin nodes.

As a result, the differential current $\vec{i}_d$ at the spin nodes is

$$\vec{i}_d = -(D_r - GD_c^{-1}G^T)\vec{v}_b = (-D_r + GD_c^{-1}G^T)\vec{v}. \tag{56}$$

This matches the desired relation in (54).

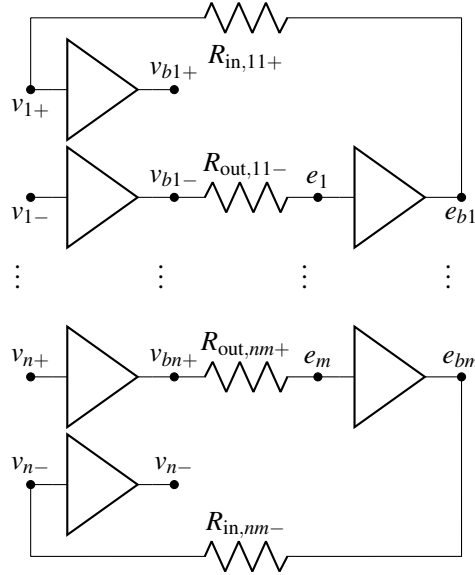### 2.4.4 Representing Arbitrary Matrices

Fig. 5: Bipartite network with buffers at every spin and auxiliary node, allowing for unidirectional connections.

For coupling matrices that are not PSD or NSD (and cannot be made PSD or NSD without sacrificing their low-rank structure), we must implement $J = U_m \Sigma_m V_m^T$, where $U_m \neq V_m$. To achieve this relationship, we add buffered nodes to each spin and auxiliary node, as shown in Figure 5. The number of buffers scales as $n+m$, so, as in Section 2.4.3, they do not form a barrier to on-chip implementation.

We have resistors $R_{\text{out},jk+} = G_{\text{out},jk+}^{-1}$ from buffered spin node v_{bj+} to auxiliary node e_k, and an analogous setup for a negative connection.[11] Likewise, we have resistors $R_{\text{in},jk+} = G_{\text{in},jk+}^{-1}$ from buffered auxiliary node e_{bk} to spin v_{j+}, and an analogous setup for a negative connection. Define

$$G_{\text{in}} \triangleq \begin{bmatrix} G_{\text{in},11+} & \cdots & G_{\text{in},1m+} \\ \vdots & \ddots & \vdots \\ G_{\text{in},n1+} & \cdots & G_{\text{in},nm+} \end{bmatrix} - \begin{bmatrix} G_{\text{in},11-} & \cdots & G_{\text{in},1m-} \\ \vdots & \ddots & \vdots \\ G_{\text{in},n1-} & \cdots & G_{\text{in},nm-} \end{bmatrix},$$

---

[11]The "in" subscript refers to resistors with current flowing into the spin nodes, and the "out" subscript refers to current flowing out of the (buffered) spin nodes into the auxiliary nodes.

$$G_{\text{out}} \triangleq \begin{bmatrix} G_{\text{out},11+} & \cdots & G_{\text{out},1m+} \\ \vdots & \cdots & \vdots \\ G_{\text{out},n1+} & \cdots & G_{\text{out},nm+} \end{bmatrix} - \begin{bmatrix} G_{\text{out},11-} & \cdots & G_{\text{out},1m-} \\ \vdots & \cdots & \vdots \\ G_{\text{out},n1-} & \cdots & G_{\text{out},nm-} \end{bmatrix},$$

the row sum matrices as $D_{r,\text{in}}$ and $D_{r,\text{out}}$, and the column sum matrices as $D_{c,\text{in}}$ and $D_{c,\text{out}}$.

Performing KCL at the (non-buffered) auxiliary nodes, we have

$$D_{c,\text{out}}\vec{e} = G_{\text{out}}^{T}\vec{v} \implies \vec{e} = D_{c,\text{out}}^{-1}G_{\text{out}}^{T}\vec{v}, \tag{57}$$

as $v_{bj} = v_j$. At the (non-buffered) spin nodes, we get

$$\vec{i}_d = (D_{r,\text{in}} - G_{\text{in}})\vec{e} = D_{r,\text{in}}\vec{v} - G_{\text{in}}D_{c,\text{out}}^{-1}G_{\text{out}}^{T}\vec{v}. \tag{58}$$

This matches the desired relationship $J = U_m\Sigma_m V_m^{T}$, if we set G_{in} to be proportional to U_m, $D_{c,\text{out}}$ to be proportional to Σ_m, and G_{out} to be proportional to V_m.

2.4.5 Analog DaS Crossbar Programmable Implementation

DaS can be implemented in a programmable fashion using variable resistors and a crossbar switch architecture. Programmability is essential in order to use the same chip to solve multiple problems that fit within a maximum number of spins and feature low-rank structure.

A simple physical layout scheme for programmable connectivity is depicted in Figure 6. The spins are laid out horizontally, and the horizontal wires represent the auxiliary nodes.[12] Each square is a switch potentially connecting a spin to a variable resistor leading to an auxiliary node. There are M layers of switches and M auxiliary nodes, where M is the maximum effective rank we choose to allow.

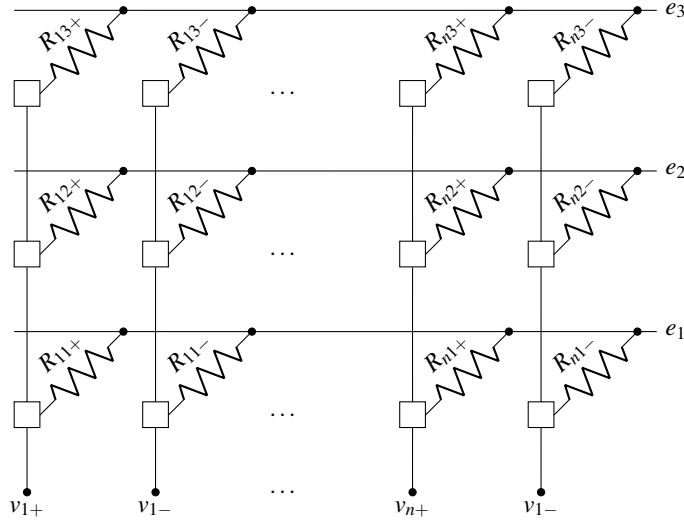

Fig. 6: DaS crossbar implementation for 3 possible auxiliary nodes.

---

[12]This simple layout is to illustrate the concept; in practice, the spin nodes are typically laid out in a rectangular grid, and a few layers of interconnect will be needed. Also, though Figure 6 shows the crossbar architecture concept for a PSD matrix, it is easily extended to support arbitrary coupling matrices.

2.5 Results

2.5.1 Sparsification of Coupling Matrices from MU-MIMO

We demonstrate results on a range of MU-MIMO detection problems, where the channel transmission matrix is produced via MATLAB's `scatteringchannelmtx` function in the Phased Array System Toolbox [19, 20] and the Ising coupling matrix is defined as in (28).

There are several parameters that define a MU-MIMO detection problem. The number of transmitters and receivers determine the shape of the channel transmission matrix and therefore the size (number of spins) of the Ising problem: the number of spins is equal to the number of transmitters, plus one. The behavior of the channel transmission matrix is primarily shaped by the spacing between transmitters and receivers (in number of wavelengths) and the number of obstructors, or scatterers, in the channel. In general, the channel transmission matrix is smoother for smaller spacing and for fewer obstructors.

In Figures 7a to 7c, we plot the Ising coupling matrices for 65-spin MU-MIMO problems (64 transmitters, 128 receivers) with different sets of parameters. For the sake of plotting, we omit the last row and column of J, as these tend to be several times larger in magnitude than the rest of the Ising weights. Figure 7a shows the coupling matrix produced in the case that the spacing is small and there are few scatterers (0.1-wavelength spacing and 5 scatterers). This coupling matrix is very smooth and follows an almost sinusoidal pattern. The coupling matrix in Figure 7b has the same spacing as Figure 7a, but with 15 scatterers instead of 5. The matrix, although still smooth, has more variation in the magnitude and location of its peaks and troughs. In Figure 7c, we show the opposite end of the spectrum: relatively spaced out transmitters or receivers, and many scatterers for the size of the problem (0.45-wavelength spacing and 50 scatterers). This matrix is not very smooth at all, due to the larger spacing and high number of scatterers.

The MU-MIMO problem (for certain choices of parameters) is well-suited for DaS for two reasons. First, for relatively smooth MU-MIMO Ising coupling matrices, there are many negligible singular values, *i.e.*, the matrix has low effective rank. Second, the coupling matrix is nearly negative semi-definite: its largest negative singular value is much greater in magnitude than its largest positive singular value. So, we use the method in Section 2.3.1 to make J NSD before performing the truncated SVD. This means that we can implement an Ising machine for MU-MIMO on-chip using mn resistors, where n is the number of spins (number of transmitters, plus 1) and m is the effective rank of J. We choose m such that the approximation error on the Ising Hamiltonian is less than 10^{-8} times the absolute sum of the Ising weights (see Section 2.3.3).

We performed DaS on MU-MIMO problems with different sets of parameters, choosing m as described above. We then calculated the relative density of each DaS representation: the number of resistors used in DaS divided by the number of resistors required for dense connectivity, *i.e.*, $\frac{nm}{n(n-1)/2} = \frac{2m}{n-1}$.

In Figure 8, we fixed the number of scatterers at 20 and calculated the density of DaS for spacings of 0.05, 0.25, and 0.45 wavelengths, varying the number of transmitters. Overall, the density decreases as the number of transmitters increases. For smaller numbers of transmitters, larger spacing results in higher density, but spacing makes little to no difference as the number of transmitters increases to several hundred. All three curves converge to $\sim 8\%$ density.

In Figure 9, we fixed the spacing to be 0.45 wavelengths and examined problems with 5, 25, and 45 scatterers. As in Figure 8, we varied the number of transmitters, and saw that the density decreases with more transmitters. The density increases with the number of scatterers. The curves for each number of scatterers get closer together as the number of transmitters increases, but they do not converge to the same value. The curves with 5, 25, and 45 scatterers converge to $\sim 2, 9, 18\%$ density, respectively.

2.5.2 Hierarchical Decomposition

Many matrices representing electrostatic and electromagnetic processes have Green's functions that are smooth in the far field. As a result, although they do not have low effective rank, they have a low-rank structure in off-diagonal blocks.
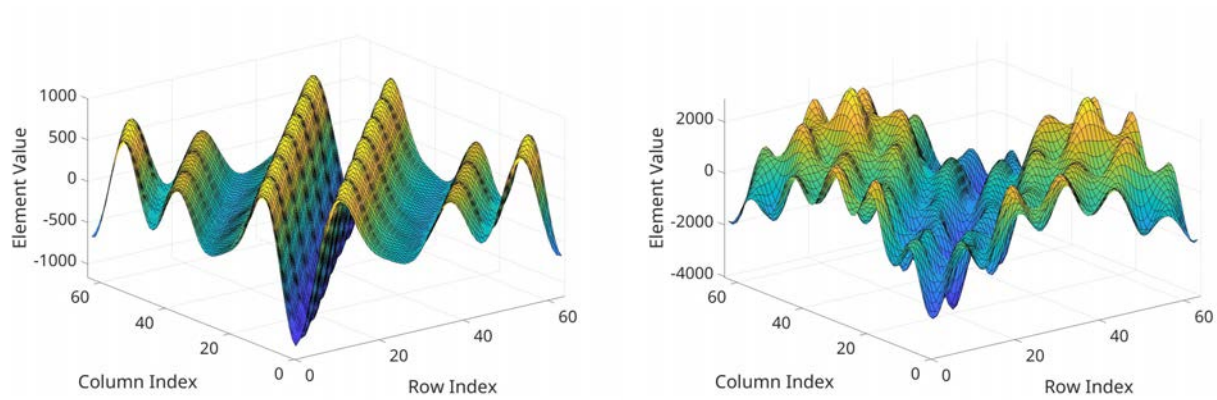
(a) 64 transmitters, 128 receivers, 5 scatterers, and 0.1 wavelengths between adjacent transmitters or receivers.

(b) 64 transmitters, 128 receivers, 15 scatterers, and 0.1 wavelengths between adjacent transmitters or receivers.
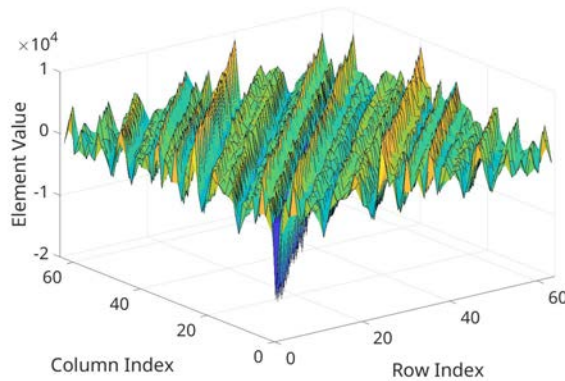
(c) 64 transmitters, 128 receivers, 50 scatterers, and 0.45 wavelengths between adjacent transmitters or receivers.

Fig. 7: Ising coupling matrix J for MU-MIMO problems with various combinations of parameters.
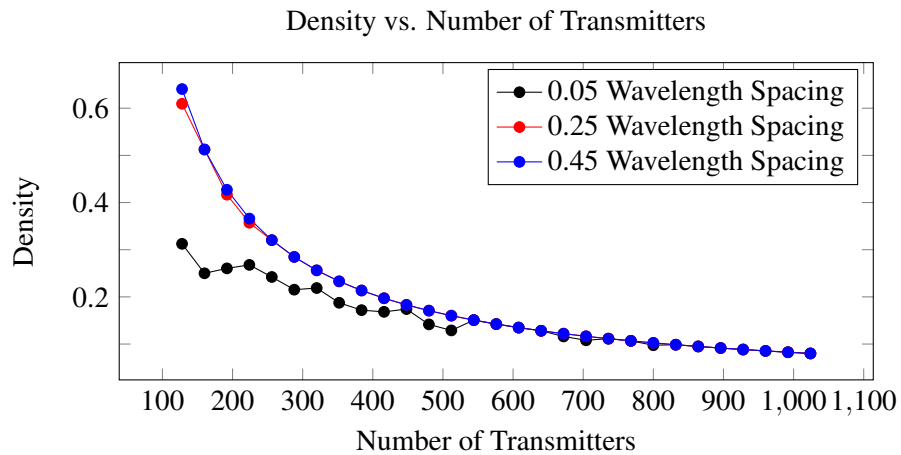

Fig. 8: Normalized number of resistors needed to represent J (1 is dense) vs. number of transmitters, for various spacing. The number of scatterers is fixed at 20.
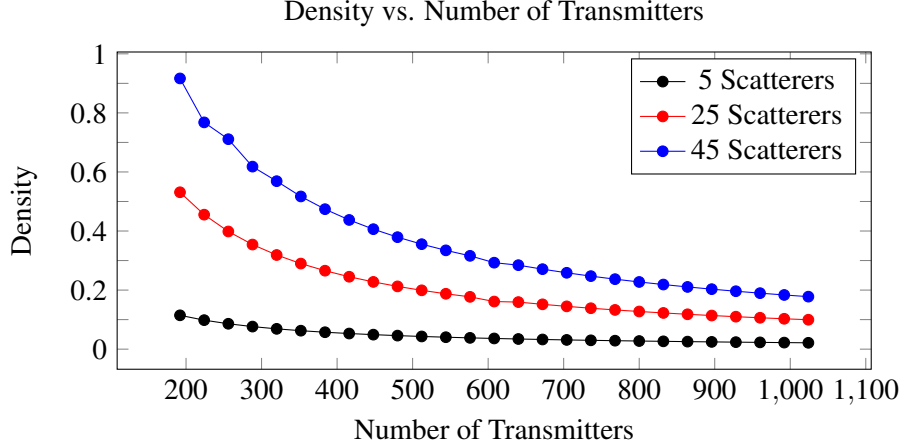
19

Fig. 9: Normalized number of resistors needed to represent J (1 is dense) vs. number of transmitters, for various numbers of scatterers. The spacing is fixed at 0.45 wavelengths.

For Ising coupling matrices derived for such processes, we compute the SVD of matrix blocks as follows:

1. We compute the SVD of J and take note of its effective rank m (the number of singular values above a certain threshold, determined based on the value of the largest singular value).

2. The number of values needed to represent the SVD of J is $N_v = mn$ if J is PSD or NSD, and $N_v = 2mn$ otherwise.

3. Divide J into four equally-sized blocks:

$$J\left(1:\frac{n}{2}, 1:\frac{n}{2}\right), \quad J\left(1:\frac{n}{2}, \frac{n}{2}+1:n\right), J\left(\frac{n}{2}+1:n, 1:\frac{n}{2}\right) \quad, \quad J\left(\frac{n}{2}+1:n, \frac{n}{2}+1:n\right). \quad (59)$$

4. For each block, recursively compute its rank map.

5. Denote the number of values required to represent each sub-block as N_{v1}, V_{v2}, N_{v3}, and N_{v4}. If $N_{v1}+N_{v2}+N_{v3}+N_{v4} < N_v$, *i.e.*, sub-dividing J decreased the density of the representation, set $N_v = N_{v1}+N_{v2}+N_{v3}+N_{v4}$ and return the recursively-determined rank map. Otherwise, return the SVD of the full J matrix.

The SVD of each matrix block can be implemented in hardware via the process in Section 2.4.4.

We examine $J \triangleq A^T A$, where A is the discretized Green's function of some asymptotically smooth electromagnetic process. This discretized Green's function plays the same role as the scattering channel matrix H in the MU-MIMO problem, so we take the product $A^T A$ to produce an Ising coupling matrix.

For the sake of demonstration, we will approximate the Green's function by the sum of matrices of the following form:

$$A_\ell(i,j) = \begin{cases} \frac{1}{|i-j-\ell|}, & i-j \neq \ell \\ 0, & i-j = \ell. \end{cases} \quad (60)$$

In Figures 10a and 10b, we show the results of recursively decomposing various size-512 matrices of the form $J = A^T A$. We measure the density of each block as the number of resistors required to implement it in sparse hardware divided by the number of resistors required for dense connectivity (essentially, $\frac{m}{n}$). Blocks of J are colored according to their density; cyan denotes sparser blocks and magenta denotes denser blocks.

Recursively subdividing these matrices, we achieved a density of 21.36% for $A = A_{25}$ and a density of 28.52% for $A = A_{400} + A_{100}$. Although we did not need this hierarchical low-rank decomposition method for

MU-MIMO problems, it can serve as a useful component of our DaS toolbox of sparsification methods for other classes of dense real-world Ising problems.
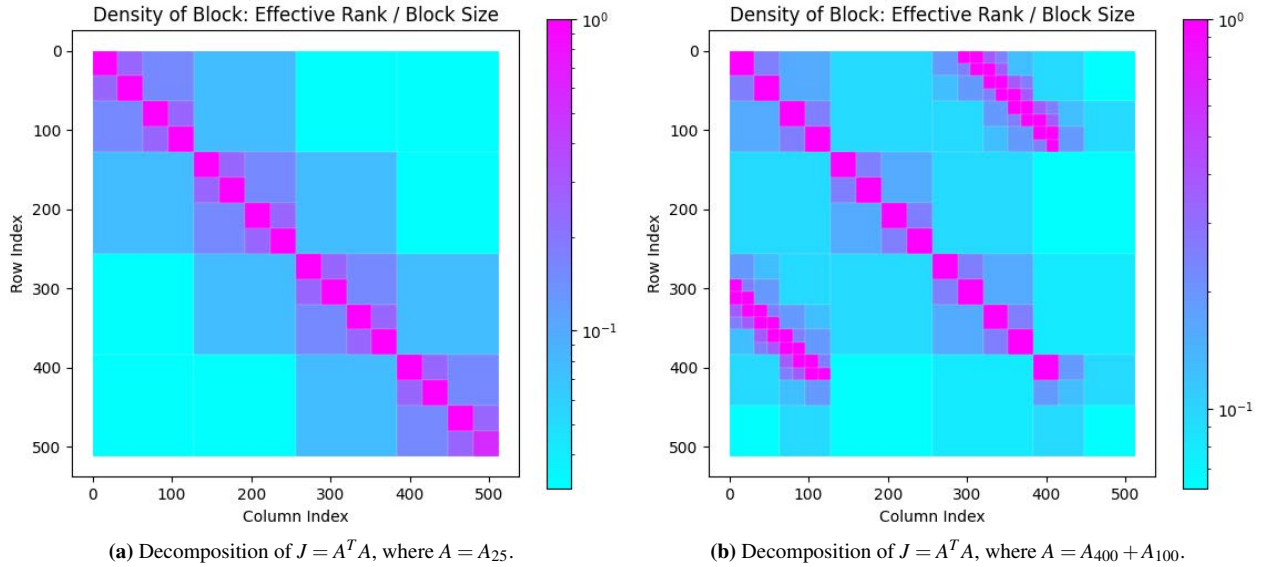

(a) Decomposition of $J = A^T A$, where $A = A_{25}$.

(b) Decomposition of $J = A^T A$, where $A = A_{400} + A_{100}$.

Fig. 10: Sparsity of rank map decompositions of different hypothetical coupling matrices.

2.6 Effect of Dense as Sparse on Ising Solver Results

Using DaS to sparsify dense Ising problems results in approximation error in the couplings between spins. *i.e.*, the coupling term in (29), $-J\vec{v}$, and the coupling term in (58), $-U_m \Sigma_m V_m^T \vec{v}$, are not precisely equal. The error between J, which we shall call the dense coupling matrix, and $U_m \Sigma_m V_m^T$, which we will call the sparsified form, decreases as m increases (or as the DaS representation becomes denser). In this section, we demonstrate how the truncated SVD approximation impacts the quality of solutions produced by simulated annealing. We also show how this impact changes as we increase or decrease m.

2.6.1 MU-MIMO Problem Generation

We generated 10 different MU-MIMO problems using the same process as in Section 2.5.1, each with 384 transmitters, 768 receivers, 10 scatterers, and a spacing between adjacent transmitters and receivers of 45 wavelengths. Each problem has 385 Ising spins (number of transmitters, plus one).

For each problem, we try a variety of thresholds to choose m, the effective rank of J. We define m as the minimum number of singular values such that the Frobenius norm of the approximation error, $J - U_m \Sigma_m V_m^T$, is less than the threshold times the Frobenius norm of J.[13] We use threshold values of $10^{-1}, 10^{-2}, \ldots, 10^{-6}$.

2.6.2 Comparison of Dense and Sparse Coupling Matrices

First, we determine the impact that the truncated SVD has on the coupling matrix itself. We consider two quantities: the density and the difference in Ising Hamiltonian values between the dense and sparse coupling matrices. The density, as defined in Section 2.5.1, determines how many resistors would be required to implement an on-chip Ising machine using DaS. The error in the Ising Hamiltonian is intrinsically tied to the solution produced by an Ising machine, as the Hamiltonian is the quantity the Ising machine minimizes. It is computationally infeasible to calculate the error in the Hamiltonian for all possible combinations of

---

[13]Though this differs from the method in Section 2.3.3, using the absolute sum of the errors, we have found empirically that this makes very little difference in the results.

spins,[14] so we examine the case where the spins are equal to the transmitted symbols.[15] In other words, we consider the Hamiltonian at our desired solution (as explained in Section 2.2.1, the goal of the MU-MIMO problem is to find the transmitted symbols).

In Figure 11, we present a table of these two quantities, for each MU-MIMO problem we generated (enumerated in the "Trials" column). We group results by the threshold we place on the Frobenius norm of the approximation error, $J - U_m \Sigma_m V_m^T$. The approximation error of the Ising Hamiltonian we present is the relative error, or `abs((H_dense-H_sparse)/H_dense)`.

Trial	Density	Relative Error in Hamiltonian	Density	Relative Error in Hamiltonian	Density	Relative Error in Hamiltonian
	Threshold: 1E-6		Threshold: 1E-5		Threshold: 1E-4	
1	10.9375%	4.17E-14%	10.9375%	4.17E-14%	10.9375%	4.17E-14%
2	10.9375%	0%	10.9375%	0%	10.9375%	0%
3	10.9375%	8.50E-14%	10.9375%	8.50E-14%	10.9375%	8.50E-14%
4	10.9375%	1.90E-14%	9.8958%	7.41E-05%	9.8958%	7.41E-05%
5	10.9375%	4.30E-14%	10.9375%	4.30E-14%	10.9375%	4.30E-14%
6	10.9375%	8.94E-14%	10.9375%	8.94E-14%	10.9375%	8.94E-14%
7	10.9375%	2.12E-14%	10.9375%	2.12E-14%	10.9375%	2.12E-14%
8	10.9375%	7.84E-14%	10.9375%	7.84E-14%	10.9375%	7.84E-14%
9	10.9375%	8.87E-14%	10.9375%	8.87E-14%	10.9375%	8.87E-14%
10	10.9375%	8.87E-14%	10.9375%	8.87E-14%	10.9375%	8.87E-14%
	Threshold: 1E-3		Threshold: 1E-2		Threshold: 1E-1	
1	10.9375%	4.17E-14%	9.895%	1.82E-01%	6.2%	2.02E+00%
2	10.9375%	0%	9.37%	1.93E+00%	6.2%	1.35E+00%
3	10.9375%	8.50E-14%	9.895%	1.20E+00%	6.2%	3.33E-01%
4	9.8958%	7.41E-05%	9.37%	5.46E+00%	5.729%	9.67E-01%
5	10.9375%	4.30E-14%	9.895%	2.21E-01%	6.2%	5.56E-01%
6	10.9375%	8.94E-14%	10.416%	3.65E+00%	7.812%	1.29E-01%
7	10.9375%	2.12E-14%	8.854%	1.38E+00%	4.166%	7.12E-01%
8	10.9375%	7.84E-14%	9.895%	8.05E-02%	6.770%	5.21E-01%
9	10.9375%	8.87E-14%	10.416%	5.04E+00%	5.208%	1.94E+00%
10	10.9375%	8.87E-14%	10.416%	2.64E+00%	5.208%	1.07E+00%

Fig. 11: Table of the density achieved by DaS and the relative error in the Ising Hamiltonian for the transmitted symbols between the dense and sparse coupling matrices. m was chosen using thresholds on the Frobenius norm of $J - U_m \Sigma_m V_m^T$ of $10^{-1}, 10^{-2}, \ldots, 10^{-6}$. Both the density and error in the Hamiltonian are in units of percent.

All trials except for trial 4 follow a similar pattern. Below a threshold of 10^{-2}, the Hamiltonian error is extremely small. For a threshold of 10^{-2}, the density decreases marginally, and the error in the Hamiltonian value increases significantly. For a threshold of 10^{-1}, the density is almost halved, while the Hamiltonian error remains approximately the same as for the previous threshold.

To an extent, the behavior of trial 4 differs from the rest: the Hamiltonian error is only extremely small below the threshold of 10^{-5}. Between 10^{-5} and 10^{-3} (inclusive), the approximation error on the Hamiltonian

---

[14]The number of possible combinations grows exponentially with the problem size; for $n = 385$, it is not computationally feasible to enumerate 2^n possibilities.

[15]More precisely, the first n spins are equal to the transmitted symbols and the $(n+1)^{\text{th}}$ spin is 1.

is larger—on the order of 10^{-4} percent rather than 10^{-14} percent. For a threshold of 10^{-2} and above, the behavior mirrors the other trials.

To explain the differences in trial 4, we examine the singular values of the Ising coupling matrices for each trial. For each trial except trial 4, there are $m = 21$ significant singular values, on the order of 10^{-1} to 10^{-4}. The rest of the singular values are extremely small, $\approx 10^{-17}$ and below. For trial 4, however, there are 19 singular values on the order of 10^{-1} to 10^{-4} and two that are approximately 10^{-8}. As with the other trials, the remainder of the singular values are negligible. For thresholds between 10^{-5} and 10^{-3}, the singular values near 10^{-8} are discarded. As these singular values are not negligible, their removal leads to a higher error on the Hamiltonian value.

2.6.3 Simulation Process

To quantify the effect of the SVD approximation of Ising solution quality, we run simulated annealing using both dense and sparsified forms, with parameters we have empirically found to be well-suited to the problem.

For each MU-MIMO problem and threshold, we ran simulated annealing 10 times. Each time, we ensured that simulated annealing was run with the same random seed for both the dense and sparse coupling matrices. For every run, we stored two quantities:

1. The best (*i.e.*, lowest) Hamiltonian value found. For comparison, we found the relative difference between the results from the dense and sparse coupling matrices. As previously, this is calculated as `abs((H_dense-H_sparse)/H_dense)`. For every MU-MIMO problem tested, we averaged these differences across all simulated annealing runs.

2. The final Ising spins found by simulated annealing. We calculated the difference between the dense and sparse results as the number of bit errors, *i.e.*, the number of spins that differ, between the two.

2.6.4 Comparison of Simulated Annealing Results

Trial	Average Relative Hamiltonian Difference					
	Thresh=1E-6	1E-5	1E-4	Thresh=1E-3	1E-2	1E-1
1	2.898E-13%	2.481E-13%	2.710E-13%	2.940E-13%	2.896E-01%	3.701E+00%
2	1.197E-13%	1.841E-13%	2.560E-13%	1.749E-13%	3.295E+00%	2.163E-01%
3	3.317E-13%	2.126E-13%	3.338E-13%	2.849E-13%	1.537E+00%	1.410E+00%
4	2.685E-13%	2.233E-01%	2.895E-01%	3.410E-01%	6.818E+00%	3.002E+00%
5	2.862E-13%	2.755E-13%	1.958E-13%	2.152E-13%	4.177E-01%	9.523E-01%
6	2.236E-13%	2.558E-13%	2.487E-13%	2.433E-13%	4.081E+00%	8.934E-01%
7	2.436E-13%	2.394E-13%	2.584E-13%	1.419E-13%	2.047E+00%	1.520E+00%
8	2.391E-13%	2.548E-13%	3.626E-13%	3.058E-13%	2.426E-01%	9.367E-01%
9	2.022E-13%	1.650E-13%	2.679E-13%	1.969E-13%	5.769E+00%	2.467E-01%
10	2.539E-13%	2.029E-13%	2.605E-13%	1.918E-13%	4.221E+00%	3.094E+00%

Fig. 12: Average difference in best Ising Hamiltonians found by running simulated annealing run on dense and sparse coupling matrices, over 10 runs per MU-MIMO problem. We use the same 10 MU-MIMO problems and the same thresholds on the Frobenius norm of the approximation error on the coupling matrix as in Figure 11.

In Figure 12, we show the average relative percent difference in best Ising Hamiltonians found by simulated annealing, as defined in Section 2.6.3. The results largely align with Section 2.6.2. For every trial except 4, the difference is very small (on the order of $10^{-13}\%$) until the threshold placed on the Frobenius norm of $J - U_m \Sigma_m V_m^T$ grows to 10^{-2} or higher, after which it varies from approximately 0.3 to 6%.

For trial 4, the difference in Hamiltonians found is only negligible for a threshold of 10^{-6}.

This indicates that the difference in Hamiltonian values when the spins are equal to the transmitted symbols (Figure 11) correlates to the difference in simulated annealing solutions. As the goal of simulated annealing is to minimize the Ising Hamiltonian, it makes sense that a higher difference in the Hamiltonian value for a certain set of spins would lead to a higher difference in simulated annealing results.

Trial	Average Bit Errors					
	Thresh=1E-6	1E-5	1E-4	1E-3	1E-2	1E-1
1	0	0	0	0	45.1	60.5
2	0	0	0	0	79.8	80.3
3	0	0	0	0	58.3	74.8
4	0	46.2	43.7	43.5	53.1	56.6
5	0	0	0	0	71.1	69.7
6	0	0	0	0	86.3	75.4
7	0	0	0	0	51.7	57
8	0	0	0	0	64.3	85.7
9	0	0	0	0	65.6	67
10	0	0	0	0	56.8	64.8

Fig. 13: Number of bit errors between the final spins produced by simulated annealing run on dense and sparse coupling matrices, averaged over 10 runs per problem.

Finally, we display the average number of bit errors in final spins produced via simulated annealing in Figure 13. For a threshold of 10^{-6}, running simulated annealing on the dense and sparse coupling matrices produces the exact same Ising spins. For all problems except trial 4, this is also the case for thresholds below 10^{-2}. At and above a threshold of 10^{-2}, there are 50–90 bit errors, *i.e.*, 13–23% of the spins differ between between results from the original and the sparsified coupling matrices. For trial 4, this behavior starts with a threshold of 10^{-5}. Overall, this is consistent with the results we obtained for the differences in Ising Hamiltonian values (Figures 11 and 12).

These results indicate that, for almost any threshold that is not very crude, DaS provides both high sparsity and extremely high accuracy of Ising solver results. Given 384-transmitter MU-MIMO problems with 10 scatterers, DaS with a threshold of 10^{-6} achieves a density of approximately 10%. In addition, running simulated annealing on the original and sparsified coupling matrices for this threshold produces the same Ising spins and negligible differences in Ising Hamiltonian values.

3 Exploration: Ising Hamiltonian Minimization Properties

3.1 Introduction

For OIM, the existence of a Lyapunov function that, under sub-harmonic injection locking, matches the Ising Hamiltonian, guarantees that the system will minimize the Ising Hamiltonian locally over time, *i.e.*, the system will settle at local minima of the Lyapunov function. For specific problems and conditions on system parameters, however, OIMs often empirically find global or near-global minima [11, 16]. Specifically, the ability of the system to reach near-global minima increases significantly under parameter cycling, *i.e.*, keeping either the strength of the SYNC signal, K_s constant and periodically ramping the strength of oscillator couplings, K_c, up and down, or vice versa. Without loss of generality, we will consider the case when K_c is held constant at some positive value and K_s is varied. Without loss of generality, we choose $K_c = 1/2$.[16] By inspection of (19), the ratio between K_c and K_s determines the location and stability of operating points.

3.1.1 Homotopy and Arclength Continuation

We aim to use the tools of homotopy and bifurcation theory to explain this phenomenon and to find conditions on the Ising coupling matrix, under which OIMs are likely to find near-global minima. Generally speaking, a homotopy is a function, $g(\vec{x}, \lambda)$, defined on $\vec{x} \in S$ (for some domain S) and $\lambda \in [A, B]$, that varies in a continuous and differentiable manner between $g(\vec{x}, A)$ and $g(\vec{x}, B)$.

We use a procedure called arclength continuation [24, 25] to numerically determine homotopies, given one or more approximate starting points. The general concept is is to step along the tangent line of the homotopy curve defined by $g(\vec{x}, \lambda) = 0$, given initial condition $(\vec{x}_0, \lambda_0)$. The direction of the tangent line is the limit of the sequence of secant line segments

$$\vec{y} = \vec{y}_0 + \Delta\vec{y}, \text{ s.t. } \frac{\vec{g}(\vec{y}_0 + \Delta\vec{y}) - \vec{g}(\vec{y}_0)}{\|\Delta\vec{y}\|_2} = 0, \tag{61}$$

as $\Delta\vec{y} \to \vec{0}$. where $\vec{y} \triangleq \begin{bmatrix} \vec{x} & \lambda \end{bmatrix}^\top$.

Recognizing that, in the limit $\Delta\vec{y} \to \vec{0}$, the right-hand side becomes $\frac{d\vec{g}}{d\vec{y}}\big|_{\vec{y}_0} \triangleq J_{\vec{y}_0}$, finding the direction of the tangent line reduces to finding the null space of the $J_{\vec{y}_0}$.[17]

In most cases, a point on this tangent line that is sufficiently close to $\vec{y}_0$ will be near a solution of $\vec{g}(\vec{y}) = \vec{0}$. So, we can use a point on the tangent line as the initial condition for a numerical solver of $\vec{g}(\vec{y}) = \vec{0}$. This idea, along with techniques to handle certain edge cases, comprises the arclength continuation algorithm.

3.1.2 Bifurcations

A bifurcation is a point on the homotopy $g(\vec{x}, \lambda) = 0$ where the number of solutions and stability of such solutions changes. For an example, see the "pitchforks" in Figure 14. [26] characterizes quantum Ising machines as finding the lowest-energy state (equivalently, the minimum Ising Hamiltonian value) via traversal of bifurcations (where the system is viewed as a homotopy, with λ being a system parameter that is varied). We hope to understand whether and how this concept applies to the dynamics of OIMs, setting $\lambda = K_s$.

3.2 Motivation: One-Dimensional System

We first considered the simplest possible OIM: a system with two spins, coupled with strength proportional to J_{12}. The Ising Hamiltonian evaluates to

$$H = -J_{12}s_1s_2 = \begin{cases} -J_{12}, & s_1 = s_2 \\ J_{12}, & s_1 \neq s_2 \end{cases}, \tag{62}$$

---

[16]In this case, the gradient of the Lyapunov function is exactly equal to $d\vec{\phi}/dt$, with a constant offset of $-K_s$.

[17]Assuming the dimension of the null space is 1, which is common but not guaranteed. Techniques exist to deal with a higher-dimensional null space.

as $s_1, s_2 \in \{\pm 1\}$.

There are four possible combinations of spins (two spins, each with two possible values), but only two values of the Ising Hamiltonian. Multiplying all spins by -1 will not change the value of the Hamiltonian. So, we can simplify the system by considering s_1 relative to s_2: we fix the value of s_2 at 1 to get

$$H = -s_1 J_{12}. \tag{63}$$

This concept generalizes to higher dimensions: we can always fix the value of the final spin at 1. For this two-spin system, the Hamiltonian is minimized when $s_1 = -\operatorname{sgn}(J_{12})$, and the minimum value is $-|J_{12}|$.

The Kuramoto equation and Lyapunov function for this system are as follows:

$$\frac{d}{dt}\phi_1(t) = -K_c J_{12} \sin(\phi_1) - K_s \sin(2\phi_1) \triangleq -f_1(\phi_1), \tag{64}$$

$$E(\phi_1) = -J_{12}\cos(\phi_1) - \frac{K_s}{2K_c}\cos(2\phi_1). \tag{65}$$

Thus, a solution to the Kuramoto equations must have

$$0 = \frac{d}{dt}\phi_1(t) = -K_c J_{12}\sin(\phi_1) - K_s \sin(2\phi_1). \tag{66}$$

In addition, as described in Section 1.2.3, an operating point is stable if and only if

$$\frac{d}{d\phi_1}\frac{d}{dt}\phi_1(t) < 0 \implies K_c J_{12}\cos(\phi_1) + 2K_s\cos(2\phi_1) > 0. \tag{67}$$

If the state is perturbed slightly in the neighborhood of a stable operating point, ϕ_1^*, the system dynamics will ensure that the state returns to ϕ_1^*.[18] As $\frac{d}{d\phi_1}E = -\frac{d\phi_1}{dt} = f(\phi_1)$, a stable operating point is one such that the derivative of the Lyapunov function is 0 and its second derivative is positive, *i.e.*, the set of stable operating points is equivalent to the set of local minima of the Lyapunov function.

For a one-dimensional system, the stable and unstable operating points can be analytically determined. Applying trigonometric identities, we get that

$$f(\phi_1) = K_c J_{12}\sin(\phi_1) + K_s \sin(\phi_1)\cos(\phi_1) = \sin(\phi_1)\left(K_c J_{12} + K_s\cos(\phi_1)\right), \tag{68}$$

so the solutions are $\sin(\phi_1) = 0$, *i.e.*, $\phi_1 \in \{0, \pi\}$, and $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$. If $\left|\frac{K_c J_{12}}{K_s}\right| \geq 1$, the only solutions are $\{0, \pi\}$;[19] otherwise, there will be four solutions.

Applying trigonometric identities to the stability condition, we have that an operating point is stable if and only if

$$K_c J_{12}\cos(\phi_1) + 4K_s\cos^2(\phi_1) - 2K_s > 0. \tag{69}$$

Applying this condition to each possible solution,

- The $\phi_1 = 0$ solution is stable iff

$$0 < K_c J_{12}\cos(0) + 4K_s\cos^2(0) - 2K_s = K_c J_{12} + 2Ks \implies K_s > -\frac{K_c J_{12}}{2}.$$

---

[18]Intuitively, if the state is $\phi_1^* + \varepsilon$, for sufficiently small $\varepsilon > 0$, then (by a first-order Taylor approximation), $d\phi_1/dt$ will be negative, and ϕ_1 will decrease until it returns to ϕ_1^*. Similar logic applies for $\phi_1 = \phi_1^* - \varepsilon$.

[19]If $\left|\frac{K_c J_{12}}{K_s}\right| = 1$, then $K_c J_{12} + K_s\cos(\phi_1) = 0$ when $\cos(\phi_1) = 1$ or $\cos(\phi_1) = -1$, which occur at $\phi_1 = 0$ and $\phi_1 = \pi$, respectively.

- The $\phi_1 = \pi$ solution is stable iff

$$0 < K_c J_{12} \cos(\pi) + 4K_s \cos^2(\pi) - 2K_s = -K_c J_{12} + 2Ks \implies K_s > \frac{K_c J_{12}}{2}.$$

- If $0 < \frac{K_c J_{12}}{K_s} < 1$, then the $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ solution is stable iff

$$0 < K_c J_{12} \cos(\pi) + 4K_s \cos^2(\pi) - 2K_s = \frac{(K_c J_{12})^2}{K_s} + 4K_s \frac{(K_c J_{12})^2}{K_s^2} + 2K_s,$$

  which is true iff $K_s < 0$. This is only possible if $J_{12} < 0$.

- Likewise, If $-1 < \frac{K_c J_{12}}{K_s} < 0$, then the $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ solution is stable iff $K_s > 0$ ($J_{12} > 0$).

3.2.1 Homotopy Experiment

We now examine the stable operating points under a simple form of parameter cycling: keeping K_c constant, increasing the value of K_s from some negative number, $-K$, to some positive number, K. for the purposes of this section, we choose $K = 2$.

To visualize the solutions that we analytically determined for the one-dimensional system, we performed arclength continuation to determine the homotopy $0 = f(\phi_1, K_s) \triangleq -\frac{d\phi_1}{dt}$, with respect to the parameter K_s. The following starting points were sufficient to determine all solutions:

1. $\phi_1 = 0$ and $K_s = K$, initially decreasing K_s: this will find the $\phi_1 = 0$ solution.

2. $\phi_1 = \pi$ and $K_s = K$, initially decreasing K_s: this will find the $\phi_1 = \pi$ solution.

3. $\phi_1 \approx \frac{\pi}{2}$ and $K_s = K$, initially decreasing K_s: If $J_{12} > 0$, this finds the section of the $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ solution where $0 < \frac{K_c J_{12}}{K_s} < 1$. If $J_{12} < 0$, this finds the section where $-1 < \frac{K_c J_{12}}{K_s} < 0$. Note that, as K_s increases, $\frac{K_c J_{12}}{K_s}$ approaches 0, meaning that this solution approaches $\phi_1 = \pm\frac{\pi}{2}$.

4. $\phi_1 \approx \frac{\pi}{2}$ and $K_s = -K$, initially increasing K_s: If $J_{12} > 0$, this finds the section of the $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ solution where $-1 < \frac{K_c J_{12}}{K_s} < 0$. If $J_{12} < 0$, this finds the section where $0 < \frac{K_c J_{12}}{K_s} < 1$.

As the stability of the solutions depends only on the sign of J_{12}, we performed this experiment with $J_{12} = 1$ and $J_{12} = -1$. Plots for both experiments are pictured in Figure 14.

For both cases ($J_{12} = \pm 1$), there are four stable solutions at $K_s = -K$. $\phi_1 = 0$ and π are unstable, and the two branches of $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ are stable. Both cases also have four stable solutions at $K_s = K$, where 0 and π are stable, and $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ is unstable. The region in between, however, differs between the two experiments.

For $J_{12} = 1$, $\frac{K_c J_{12}}{K_s}$ approaches 1 as $K_s \to -\frac{1}{2}$, so both branches of $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ approach $\phi_1 = 1$. When $\phi_1 = -\frac{1}{2}$, there is a bifurcation point: the $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ solution "disappears," and the $\phi_1 = 0$ solution becomes stable. For $-\frac{1}{2} < K_s < \frac{1}{2}$, the only solutions are $\phi_1 = 0$ and $\phi_1 = \pi$, and only the $\phi_1 = 0$ solution is stable. At $K_s = \frac{1}{2}$, there is another bifurcation point: the $\phi_1 = \pi$ solution becomes stable, and the $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ solution re-emerges, this time starting at $\phi_1 = \pi$ and approaching $\pm\frac{\pi}{2}$ as K_s increases further.

For $J_{12} = -1$, both branches of $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ approach $\phi_1 = \pi$ as $K_s \to -\frac{1}{2}$, and the $\phi_1 = \pi$ solution becomes stable at that bifurcation point. Then, at the $\phi_1 = \frac{1}{2}$ bifurcation, the $\phi_1 = 0$ solution becomes stable and the unstable $\cos(\phi_1) = \frac{K_c J_{12}}{K_s}$ solution branches out from $\phi_1 = 0$. In the region between the two bifurcation points, only the $\phi_1 = \pi$ solution is stable.
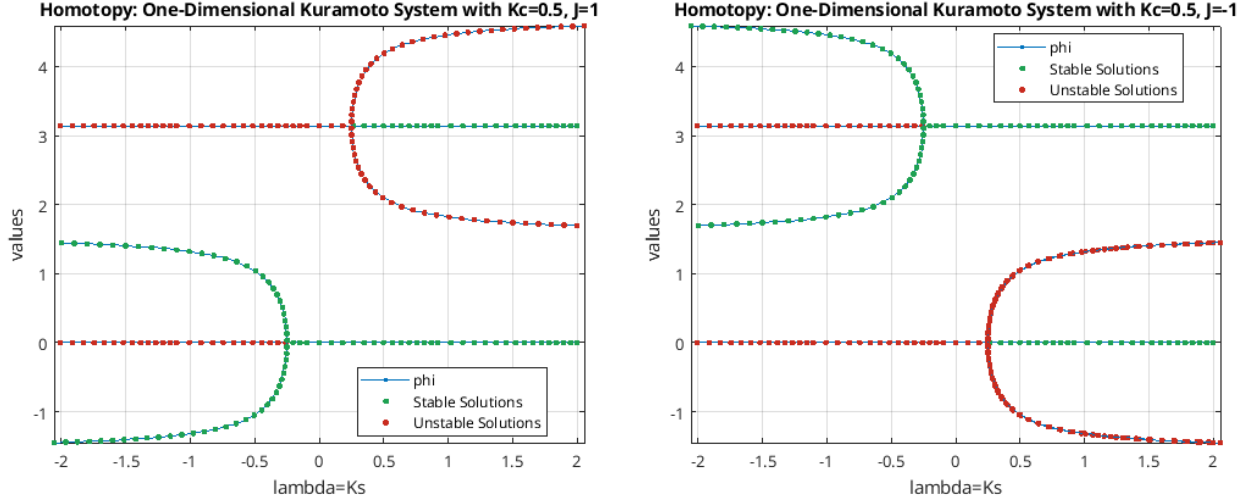
Fig. 14: Homotopy curves for the one-dimensional Kuramoto system, with $K_c = \frac{1}{2}$ and Ising coupling of 1 (left) and -1 (right).

3.2.2 Asymmetry and Homotopy Plots

In the previous analysis, we assumed that each oscillator has the exact same SYNC current injection. We also wish to examine how asymmetry between spins affects the solutions of the Kuramoto equations under parameter cycling. To do so, we introduce asymmetry in the form of a phase shift of the SYNC signal:

$$\frac{d}{dt}\phi_1(t) = -K_c J_{12} \sin(\phi_1) - K_s \sin(2\phi_1 - \alpha_1), \tag{70}$$

where we refer to α_1 as the asymmetry constant for oscillator 1 (with respect to the fixed oscillator 2).
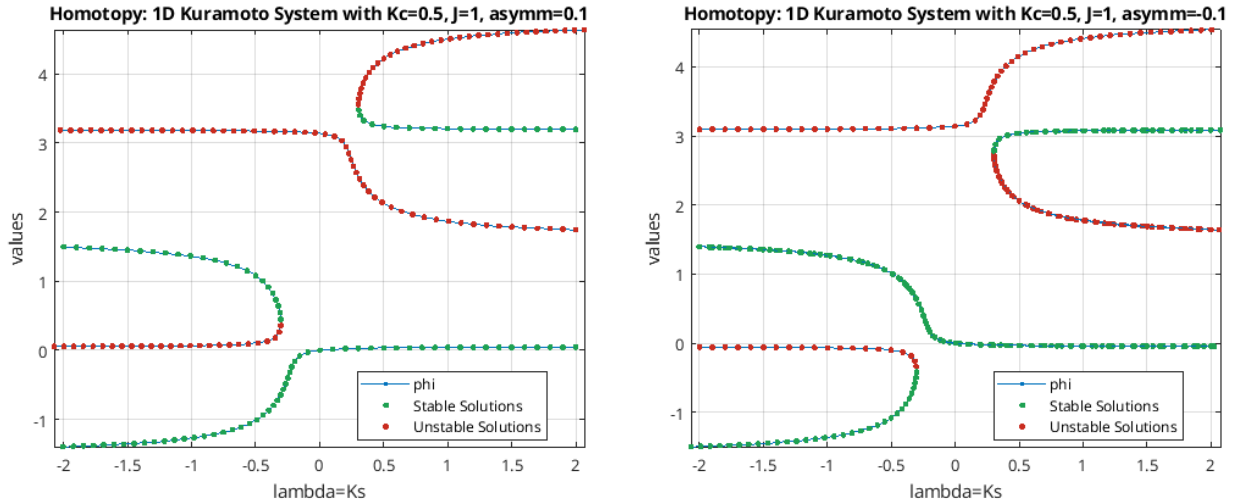

Fig. 15: Homotopy curves for the one-dimensional Kuramoto system, with $K_c = \frac{1}{2}$, positive coupling, and small positive (left) and negative (right) asymmetry constants.

Homotopy curves for systems with asymmetry values of ± 0.1 are shown in Figure 15 (positive coupling) and Figure 16 (negative coupling). In all cases, the asymmetry "breaks" the pitchfork bifurcations into two separate segments. The bifurcations are each split into two segments: one continuous segment that is either consistently stable or consistently unstable for all K_s, and one "sideways U"-shaped segment that has a stable branch and an unstable branch.
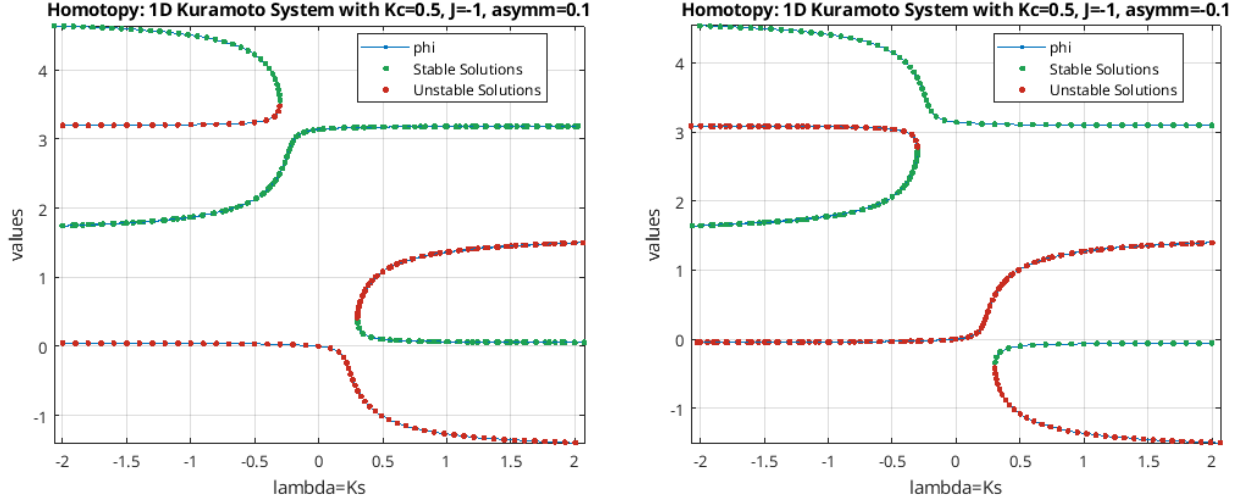
28

Fig. 16: Homotopy curves for the one-dimensional Kuramoto system, with $K_c = \frac{1}{2}$, negative coupling, and small positive (left) and negative (right) asymmetry constants.

For positive coupling, the consistently stable curve starts at $\phi_1 \approx -\frac{\pi}{2}$ at $K_s = -K$ and moves to $\phi_1 = 0$ near $K_s = -\frac{1}{2}$. The consistently unstable curve starts at $\phi_1 = \pi$ and moves to $\phi_2 \approx \frac{\pi}{2}$ near $K_s = \frac{1}{2}$. For negative coupling, the stable curve starts at $\phi_1 \approx \frac{3\pi}{2}$ and moves to $\phi_1 = \pi$ near $K_s = -\frac{1}{2}$. The unstable curve starts at $\phi_1 = 0$ and moves to $\phi_1 \approx \frac{\pi}{2}$ near $K_s = \frac{2}{2}$. It may be significant that, in both cases, the stable curve converges to the value of ϕ that minimizes the Ising Hamiltonian (if we set $s_1 = 1$ when $\phi = 0$ and $s_1 = -1$ when $\phi = -\pi$). More exploration is needed on the role of asymmetry in Ising Hamiltonian minimization, especially for higher-dimensional systems.

3.2.3 Connection Between Stability of $\phi_1 \in \{0, \pi\}$ and the Ising Hamiltonian

In a Kuramoto system with no asymmetry, the grid points ($\vec{\phi} \in \{0, \pi\}^n$) will always be equilibria of the Kuramoto equations, (19), as all of the sines will become zero. We would like to quantify when a grid point is a stable equilibrium and draw a connection to the stability region of a grid point and the Ising Hamiltonian, evaluated at that grid point.

For a one-dimensional system, the connection is clear: a grid point is stable, if and only if, $\frac{K_s}{K_c} > \cos(\phi_1)\frac{J_{12}}{2}$. Setting the Ising spin value as $s_1 \triangleq \cos(\phi_1)$, we get that the grid point is stable when

$$\frac{K_s}{K_c} > -\cos(\phi_1)\frac{J_{12}}{2} = -\frac{1}{2}J_{12}s_1 = \frac{1}{2}H(s_1), \tag{71}$$

where $H(s_1)$ is the Ising Hamiltonian, evaluated at $s_1 = \cos(\phi_1)$. So, the grid point corresponding to the lower Hamiltonian (*i.e.*, the solution to the Ising problem), will become stable first as we increase K_s from $K_s = -K < -\frac{K_c}{2}$.

Let us assume that, as we cycle K_s from $-K$ to K, that the state remains in the neighborhood of a stable equilibrium point at all times.[20] Then this, combined with the analysis of Figure 14, guarantees that the system will settle to the grid point that minimizes the Ising Hamiltonian as we increase K_s.

We hope to explore whether such a connection exists, theoretically and experimentally, for higher-dimensional systems. The current work in this direction is described in the rest of this report, and there is much future research to be done.

---

[20]Based on the Lyapunov minimization property, the state will converge to a stable equilibrium point given that K_s is held constant for long enough. So, this is a reasonable assumption to make if we cycle K_s slowly enough.

3.3 Visualization of Solutions to a Two-Dimensional System and Stability Regions

We will now consider a two-dimensional system: three coupled oscillators, where our frame of reference is the phase of the third oscillator (*i.e.*, we fix $s_3 = 1$, or $\phi_3 = 0$). The Kuramoto equations are

$$\frac{d}{dt}\vec{\phi} = \begin{bmatrix} \frac{d}{dt}\phi_1 \\ \frac{d}{dt}\phi_2 \end{bmatrix} = \begin{bmatrix} -K_c J_{12}\sin(\phi_1 - \phi_2) - K_c J_{13}\sin(\phi_1) - K_s\sin(2\phi_1) \\ -K_c J_{12}\sin(\phi_2 - \phi_1) - K_c J_{23}\sin(\phi_2) - K_s\sin(2\phi_1) \end{bmatrix} \triangleq -\vec{f}(\vec{\phi}_1) = \begin{bmatrix} -f_1(\vec{\phi}) \\ -f_2(\vec{\phi}) \end{bmatrix}, \quad (72)$$

the Lyapunov function is

$$E(\vec{\phi}) = -J_{12}\cos(\phi_1 - \phi_2) - J_{13}\cos(\phi_1) - J_{23}\cos(\phi_2) - \frac{K_s}{2K_c}\cos(2\phi_1) - \frac{K_s}{2K_c}\cos(2\phi_1), \quad (73)$$

and the Ising Hamiltonian is

$$H = -J12s_1s_2 - J_{13}s_1s_3 - J_{23}s_2s_3 = -J_{12}s_1s_2 - J_{13}s_1 - J_{23}s_2. \quad (74)$$

As it is much less tractable to analytically determine stable solutions than for the one-dimensional case, and analytical solutions (if possible to determine for a general three-node graph) are likely to be complicated and not yield immediate insight, we aim to visualize:

1. Setting K_c at $\frac{1}{2}$, how the stable and unstable solutions to the Kuramoto equations change as we vary K_s from $-K$ to K (for a sufficiently-large value of K).

2. For a fixed ratio $\frac{K_s}{K_c}$, the locations of the stable and unstable equilibria. If the system is given an initial condition $\vec{\phi}_0$, how can we determine the steady state, $\lim_{t\to\infty}\vec{\phi}(t) = \vec{\phi}^*$, where $\vec{\phi}^*$ is a stable solution?

3.3.1 Kuramoto Equilibria, Varying K_s: Homotopy Experiments

To determine the solutions of the two-dimensional Kuramoto system under K_s cycling, we perform homotopy experiments similar to Section 3.2.1. As before, we pay special attention to the system behavior as we pass across bifurcation thresholds.

The arclength continuation initial conditions were all permutations of $\phi_1, \phi_2 \in \{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$, at both $K_s = -3 \triangleq -K$ and $K_s = K$. This is because, as $K_s \to \pm\infty$ (and the SYNC term dominates), the solutions approach $\phi_1, \phi_2 \in \{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$. So, for a relatively high value of K_s, all of the solutions will be near these initial conditions. This procedure is not guaranteed to give us all solutions to the Kuramoto equations, $\forall K_s \in [-K, K]$, but it is sufficient to at least partially understand how the system evolves under parameter cycling.

Homotopy curves for three random coupling matrices are shown in Figures 17 to 19. The solutions are colored based on whether they are a minimum (green), maximum (red), or saddle point (orange) of the Lyapunov function. Due to the large number of solutions at any given value of K_s and overall complexity of the homotopy curves, it is difficult to draw definitive, overarching conclusions about the Kuramoto solutions. However, we have seen trends, specifically in bifurcations and stability of the grid points, that provide directions for future research.

Notably, we tend to see pitchfork bifurcations, as we did for the one-dimensional system, when grid points become stable. We see similar bifurcations when a grid point goes from being a local maximum of the Lyapunov function to a saddle point. If the bifurcations occur at smaller K_s, the pitchfork tends to point towards $-K$. In these cases, the stability of the outer solutions matches the stability of the grid point at $K_s = K_s^* + \varepsilon$, where K_s^* is the location of the bifurcation and $\varepsilon > 0$ (*e.g.*, they are stable if the grid point becomes stable at the bifurcation). For bifurcations at larger K_s, the pitchfork points towards K and the stability of the outer solutions matches the stability of the grid point at $K_s^* - \varepsilon$ (*e.g.*, they are Lyapunov saddle points if the grid point becomes stable).
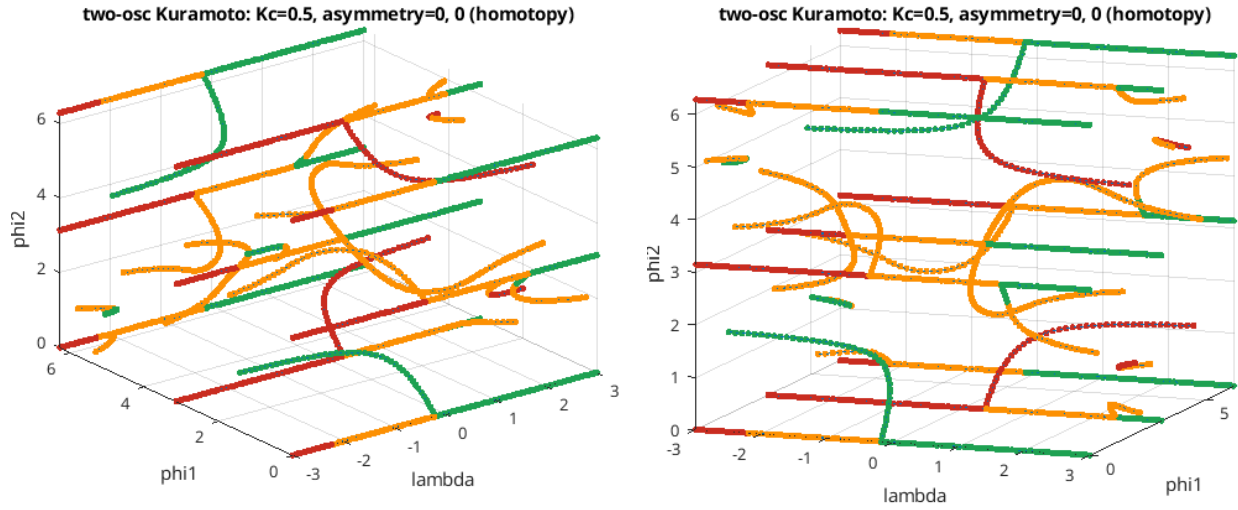
Fig. 17: Homotopy curves for $J = \begin{bmatrix} 0 & 3.90 & 2.27 \\ 3.90 & 0 & -0.39 \\ 2.27 & -0.39 & 0 \end{bmatrix}$ and $K_s = \frac{1}{2}$, where $\lambda = K_s$. As this is a 3D plot, we show two different angles. Stable solutions are plotted in green, local maxima of the Lyapunov function are plotted in red, and saddle points of the Lyapunov function are plotted in orange. Note that ϕ_1 and ϕ_2 are plotted in the range $[0, 2\pi]$, so some of the curves appear to be split in half due to phase wrapping.
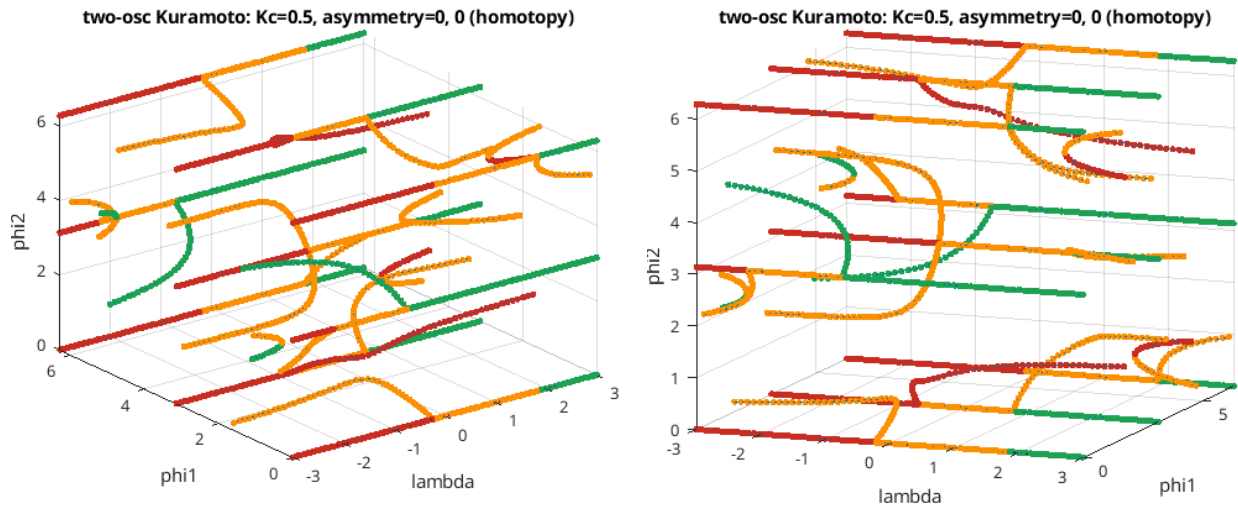

Fig. 18: Homotopy curves for $J = \begin{bmatrix} 0 & -2.90 & 2.61 \\ -2.90 & 0 & -3.33 \\ 2.61 & -3.33 & 0 \end{bmatrix}$ and $K_s = \frac{1}{2}$, from two angles.
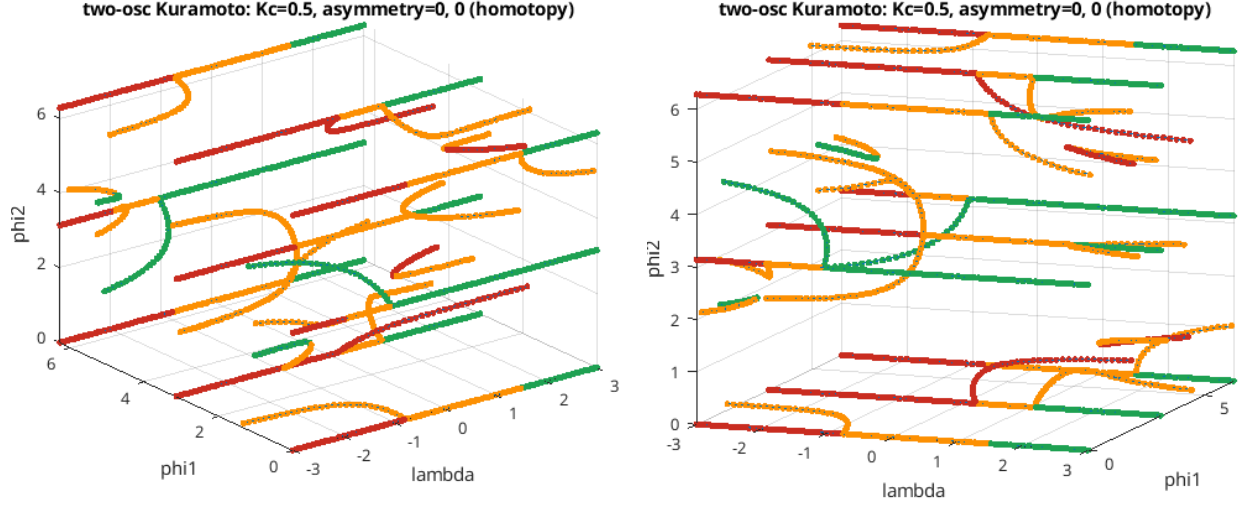
Fig. 19: Homotopy curves for $J = \begin{bmatrix} 0 & -1.75 & 4.47 \\ -1.75 & 0 & -3.96 \\ 4.47 & -3.96 & 0 \end{bmatrix}$ and $K_s = \frac{1}{2}$, from two angles.

It might be significant that grid points that become stable at lower K_s tend to have pitchfork bifurcations pointing towards $-K$. When we increase K_s, if the state follows the stable prongs of the bifurcation, it will end up at the grid point. As explored in Section 3.5, for many Ising problems, there is a correlation towards grid points being stable at lower K_s and global minimization of the Ising Hamiltonian. So, there may be a connection between traversing these bifurcations and the ability of OIMs to find global Hamiltonian minima. This intuition provides context for future analytical work in understanding bifurcations of the two-dimensional Kuramoto system.

We also considered the two-dimensional Kuramoto system with asymmetry, *i.e.*,

$$\frac{d}{dt}\vec{\phi} = \begin{bmatrix} \frac{d}{dt}\phi_1 \\ \frac{d}{dt}\phi_2 \end{bmatrix} = \begin{bmatrix} -K_c J_{12}\sin(\phi_1 - \phi_2) - K_c J_{13}\sin(\phi_1) - K_s \sin(2\phi_1 - \alpha_1) \\ -K_c J_{12}\sin(\phi_2 - \phi_1) - K_c J_{23}\sin(\phi_2) - K_s \sin(2\phi_1 - \alpha_2) \end{bmatrix}. \tag{75}$$

Some homotopy plots for systems with asymmetry are shown in Figures 20 and 21. As with the one-dimensional system, the asymmetry "breaks" the pitchfork bifurcations (even if only one oscillator has asymmetry). Beyond this fact, more exploration is required to understand the solutions of the two-dimensional asymmetric Kuramoto system.

3.3.2 Kuramoto Equilibria for Fixed K_s

To examine the trajectory of the Kuramoto system from an initial condition $\vec{\phi}_0$ with respect to minimization of the Lyapunov function, we plotted $\vec{\phi}$ superposed on the level sets of $E(\vec{\phi})$. The results for two different coupling matrices and initial conditions are shown in Figure 22.

The trajectory is orthogonal to the level sets of the Lyapunov function (*i.e.*, tangent to the gradient), as $\frac{d\vec{\phi}}{dt} \propto -\frac{d}{d\vec{\phi}}E(\vec{\phi})$.[21] As a result, we can use features of the Lyapunov function and the Lyapunov minimization property to demarcate regions of the $\vec{\phi}$ space such that converge to each stable operating point.

Empirically, we have seen that the the $\vec{\phi}$ space is almost always subdivided by a series of simple closed curves that we call "ridges," were the interior of each curve contains exactly one local minimum. We define a

---

[21]As a side note, we can think of the Kuramoto system as performing gradient descent on the Lyapunov function, $\vec{\phi}(t_{k+1}) = \vec{\phi}(t_k) - \eta \frac{d}{d\vec{\phi}}E(\vec{\phi})$, where $t_{k+1} - t_k \to 0$, $\eta \to 0$, and $\frac{\eta}{t_{k+1}-t_k} = \frac{1}{K_c}$.
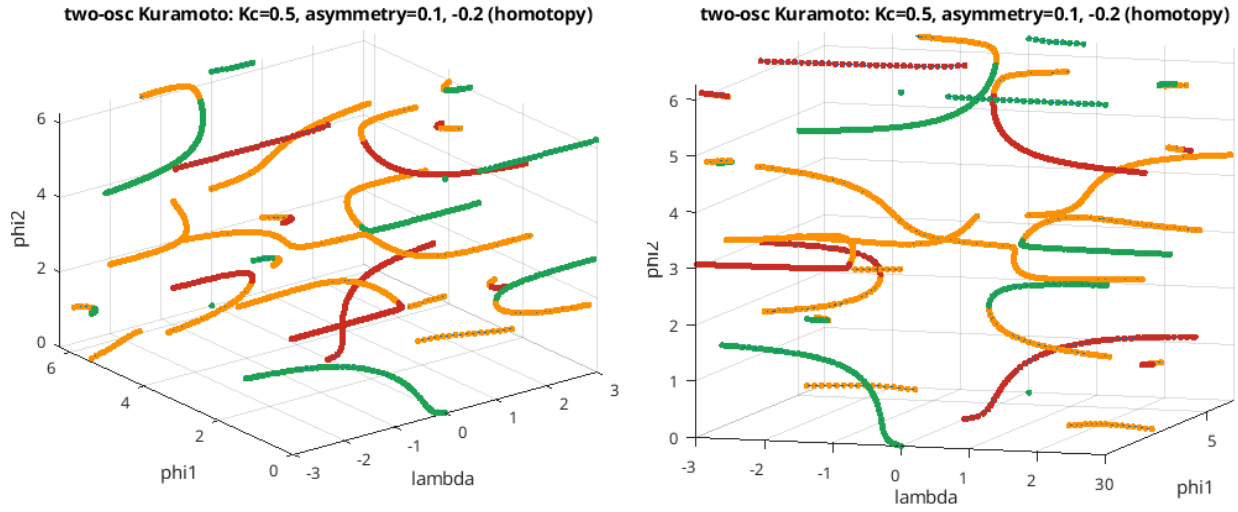
Fig. 20: Homotopy curves for $J = \begin{bmatrix} 0 & 2.84 & 4.60 \\ 2.84 & 0 & -0.17 \\ 4.60 & -0.17 & 0 \end{bmatrix}$, $K_s = \frac{1}{2}$, and $\begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \triangleq \vec{\alpha} = \begin{bmatrix} 0.1 \\ -0.2 \end{bmatrix}$, from two angles. Note that phase wrapping has caused some single-point curves to appear: those are an artifact of plotting, not of the system itself.
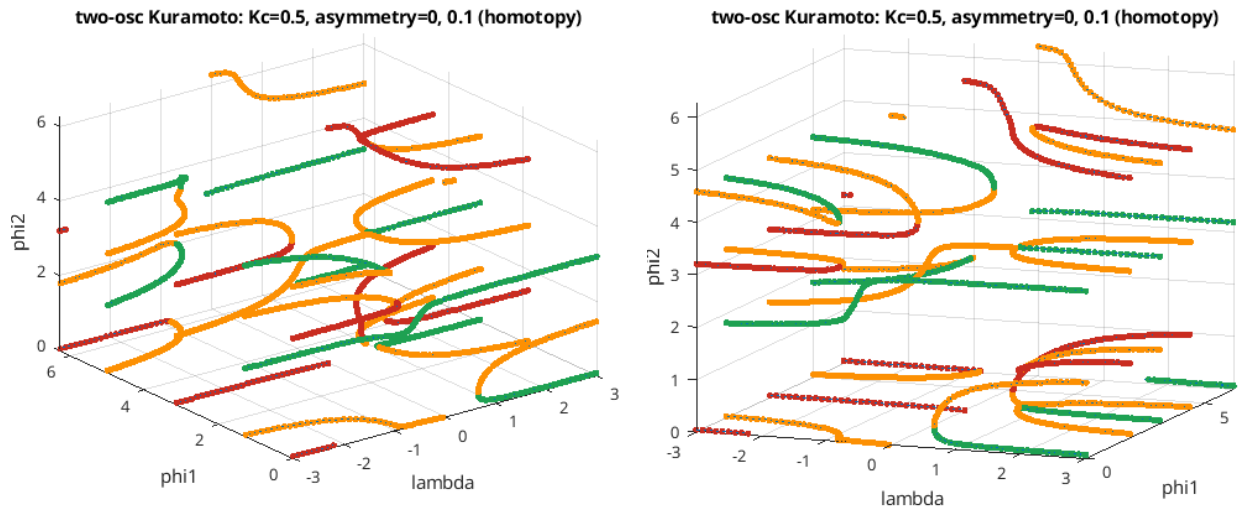

Fig. 21: Homotopy curves for $J = \begin{bmatrix} 0 & 0.10 & 3.34 \\ 0.10 & 0 & -2.28 \\ 3.34 & -2.28 & 0 \end{bmatrix}$, $K_s = \frac{1}{2}$, and $\vec{\alpha} = \begin{bmatrix} 0 \\ 0.1 \end{bmatrix}$, from two angles.
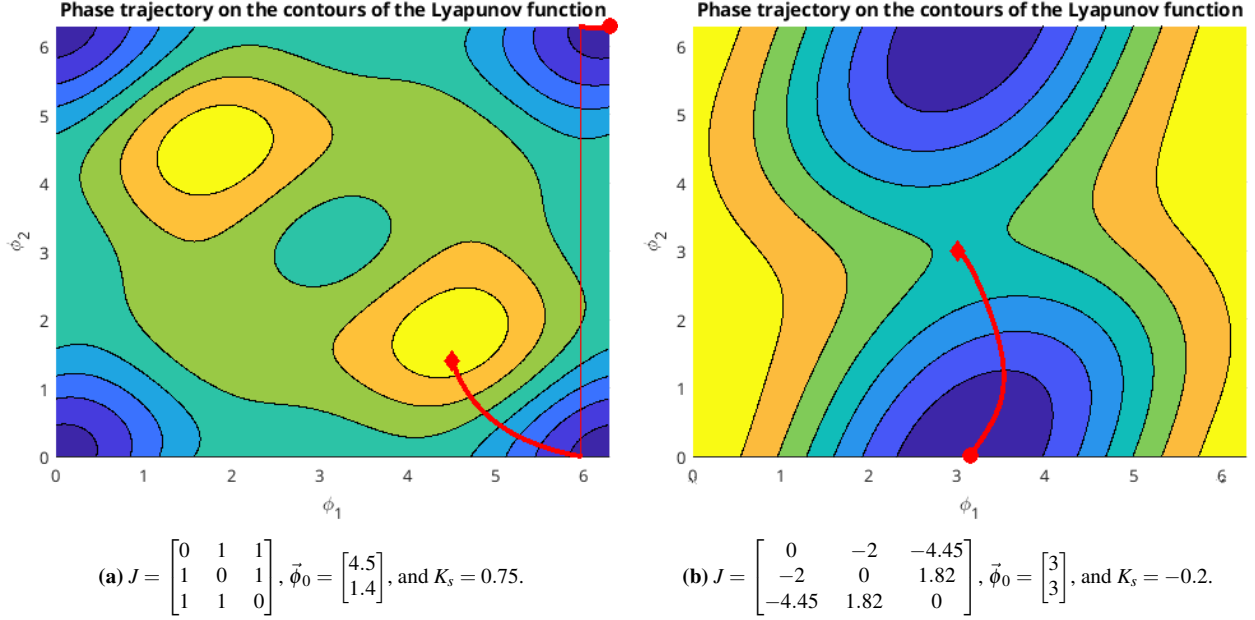
33

(a) $J = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$, $\vec{\phi}_0 = \begin{bmatrix} 4.5 \\ 1.4 \end{bmatrix}$, and $K_s = 0.75$.

(b) $J = \begin{bmatrix} 0 & -2 & -4.45 \\ -2 & 0 & 1.82 \\ -4.45 & 1.82 & 0 \end{bmatrix}$, $\vec{\phi}_0 = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$, and $K_s = -0.2$.

Fig. 22: State, $\vec{\phi} = \begin{bmatrix} \phi_1 & \phi_2 \end{bmatrix}^\top$, of the Kuramoto system, for two different coupling matrices and initial conditions, and values of K_s (with $K_c = \frac{1}{2}$). The initial condition is represented by ♦ and the steady state is represented by •. $\vec{\phi}$, being a phase, is plotted mod 2π, and there is phase wrapping in the left plot.

ridge as a curve such that every point is a local maximum of the Lyapunov function in the direction orthogonal to the curve. Intuitively, these look like ridges in geology: areas that are "raised" with respect to their surroundings. By this definition, $\nabla E(\vec{\phi})$ is tangent to the curve. Either $\nabla E(\vec{\phi}) = 0$ or the direction orthogonal to $\nabla E(\vec{\phi})$ must be aligned with an eigenvector of $\nabla^2 E(\vec{\phi})$ corresponding to a negative eigenvalue. Though we have yet to prove that such ridges are guaranteed to exist, we will consider the behavior of the system in the case that they do exist.

We would like to prove that the ridges determine the steady state corresponding to each initial condition. Namely, if the system's initial condition starts in the interior of a ridge, then we hypothesize that the state will always converge to the local minimum of the Lyapunov function contained within that ridge. We have made progress towards a proof of this, based on the Lyapunov minimization property of the Kuramoto system. In this proof, we have the additional constraint that, $\forall \lambda(\vec{\phi})$, where $\lambda(\vec{\phi})$ is the eigenvalue of $\nabla E(\vec{\phi})$ corresponding to the direction orthogonal to the ridge, $\lambda(\vec{\phi}) \leq -\lambda^* < 0$.

1. The state will never leave the interior of the ridge.

   We first show that, $\exists \varepsilon > 0$ such that, $\forall \vec{\phi}_0$ a distance of less than ε from the ridge, there is a component of $\frac{d\vec{\phi}}{dt}$ that points away from the ridge. Let $\vec{\phi}_r$ be the point on the ridge closest to $\vec{\phi}_0$. We know that $(\vec{\phi}_0 - \vec{\phi}_r)$ is orthogonal to the ridge, or $(\vec{\phi}_0 - \vec{\phi}_r) \perp \nabla E(\vec{\phi}_r)$.

   By the Taylor approximation of $\nabla E(\vec{\phi})$ around $\vec{\phi}_r$,

   $$-\frac{1}{K_c} \frac{d\vec{\phi}}{dt}\bigg|_{\vec{\phi}_0} = \nabla E(\vec{\phi}_0) = \nabla E(\vec{\phi}_r) + \nabla^2 E(\vec{\phi}_r)(\vec{\phi}_0 - \vec{\phi}_r) + O\left(\left\|\vec{\phi}_0 - \vec{\phi}_r\right\|_2^2\right)$$

   $$= \nabla E(\vec{\phi}_r) + \lambda(\vec{\phi}_r)(\vec{\phi}_0 - \vec{\phi}_r) + O\left(\left\|\vec{\phi}_0 - \vec{\phi}_r\right\|_2^2\right).$$

   The remainder terms, represented by $O\left(\left\|\vec{\phi}_0 - \vec{\phi}_r\right\|_2^2\right)$, are $\leq \alpha \left\|\vec{\phi}_0 - \vec{\phi}_r\right\|_2^2 \leq \alpha\varepsilon^2$ in magnitude,

where $\alpha > 0$ is a constant based on the higher-order terms of the Taylor series.[22] So, the component of $\frac{d\vec{\phi}}{dt}\big|_{\vec{\phi}_0}$ orthogonal to the ridge lies between $\frac{1}{K_c}(-\lambda(\vec{\phi}_r) - \alpha\varepsilon)\varepsilon\vec{u}$ and $\frac{1}{K_c}(-\lambda(\vec{\phi}_r) + -\alpha\varepsilon)\varepsilon\vec{u}$, where $\vec{u}$ is the unit vector pointing in the direction of $\vec{\phi}_0 - \vec{\phi}_r$ (away from the ridge. If $\varepsilon < \lambda^*/\alpha$, then, as $\lambda(\vec{\phi}_r) < -\lambda^* < 0$, this component is guaranteed to point away from the ridge.

Intuitively, since $\frac{d\vec{\phi}}{dt}$ has a component pointing away from the ridge, the system's trajectory will push the state farther from the ridge. In future research, we hope to prove this rigorously.

2. The system is guaranteed to settle at a local minimum of the Lyapunov function. As the state must remain inside the curve, it will settle to the unique local minimum in the ridge's interior.

The fact that $\nabla E(\vec{\phi}) = -\frac{1}{K_c}\frac{d\vec{\phi}}{dt}$ is tangent to the ridge allows us to find ridges via transient simulation: if we start transient simulation of the Kuramoto equations at a point precisely on a ridge, then the system will trace the shape of the ridge. If there is numerical error (due to finite-precision floats and finite simulation timestep), however, the state will not remain precisely on the ridge. In that case, as derived above, it will instead settle to a local minimum of the Lyapunov function somewhere in the interior of the ridge.

We can circumvent this by negating the Kuramoto equations (and, therefore, their Lyapunov function). Loosely speaking, a ridge is now a local minimum of $E(\vec{\phi})$ in its orthogonal direction, meaning that the system will return to the ridge after a small deviation from the curve. This intuition yields the following practical algorithm for finding ridges of the Lyapunov function:

1. *Find the local maxima and saddle points of the Lyapunov function.* Unless the ridge is constant-valued, it will contain at least one local maximum of the Lyapunov function and one saddle point. This is because the ridge is a closed set and the Lyapunov function is continuous, so $E : C \to \mathbb{R}$, where C denotes the ridge, attains a maximum and minimum on C. As the Lyapunov function is a local maximum in the direction orthogonal to the ridge, a local maximum of $E : C \to \mathbb{R}$ is a local maximum of $E : \mathbb{R}^2 \to \mathbb{R}$, and a local minimum of $E : C \to \mathbb{R}$ is a saddle point of $E : \mathbb{R}^2 \to \mathbb{R}$. By similar logic, between any two distinct, adjacent local maxima on the ridge, there will be one saddle point.

2. *For every saddle point, negate K_c and K_s, and therefore the Kuramoto equations and Lyapunov function and run two transient simulations:*

   (a) Set the initial conditions to be on either side of the saddle point. Specifically, the initial conditions are $\vec{\phi}_s + \varepsilon\vec{v}$ and $\vec{\phi}_s - \varepsilon\vec{v}$, where $\vec{\phi}_s$ is the saddle point, ε is a small constant, and $\vec{v}$ is the eigenvector of $\nabla^2 E(\vec{\phi}_s)$ (the original non-negated Lyapunov Hessian) with a positive eigenvalue ($\vec{v}$ must be tangent to the ridge, assuming the saddle point lies on a ridge: by the Spectral Theorem, it is orthogonal to the eigenvector with a negative eigenvalue).

   (b) Run transient simulation, starting at each initial condition. As $\frac{d\vec{\phi}}{dt} \propto \nabla E(\vec{\phi})$, this will trace the ridge, ending at a local minimum of the negated Lyapunov function.

Results of this procedure, applied to random coupling matrices and various values of K_s, are in Figure 23. In Figure 24, we plotted the results on top of a surface plot of the Lyapunov function to verify, visually, that the procedure indeed traced the ridges of the Lyapunov function.

---

[22] α can be assumed to be absolute with respect to $\vec{\phi}_r$. Since the Lyapunov function is a finite linear combination of sinusoids, there exists an absolute bound on each term of the third derivative of E with respect to $\vec{\phi}$. So, by the Taylor remainder theorem, an absolute bound can also be placed on the remainder term of the Taylor approximation.
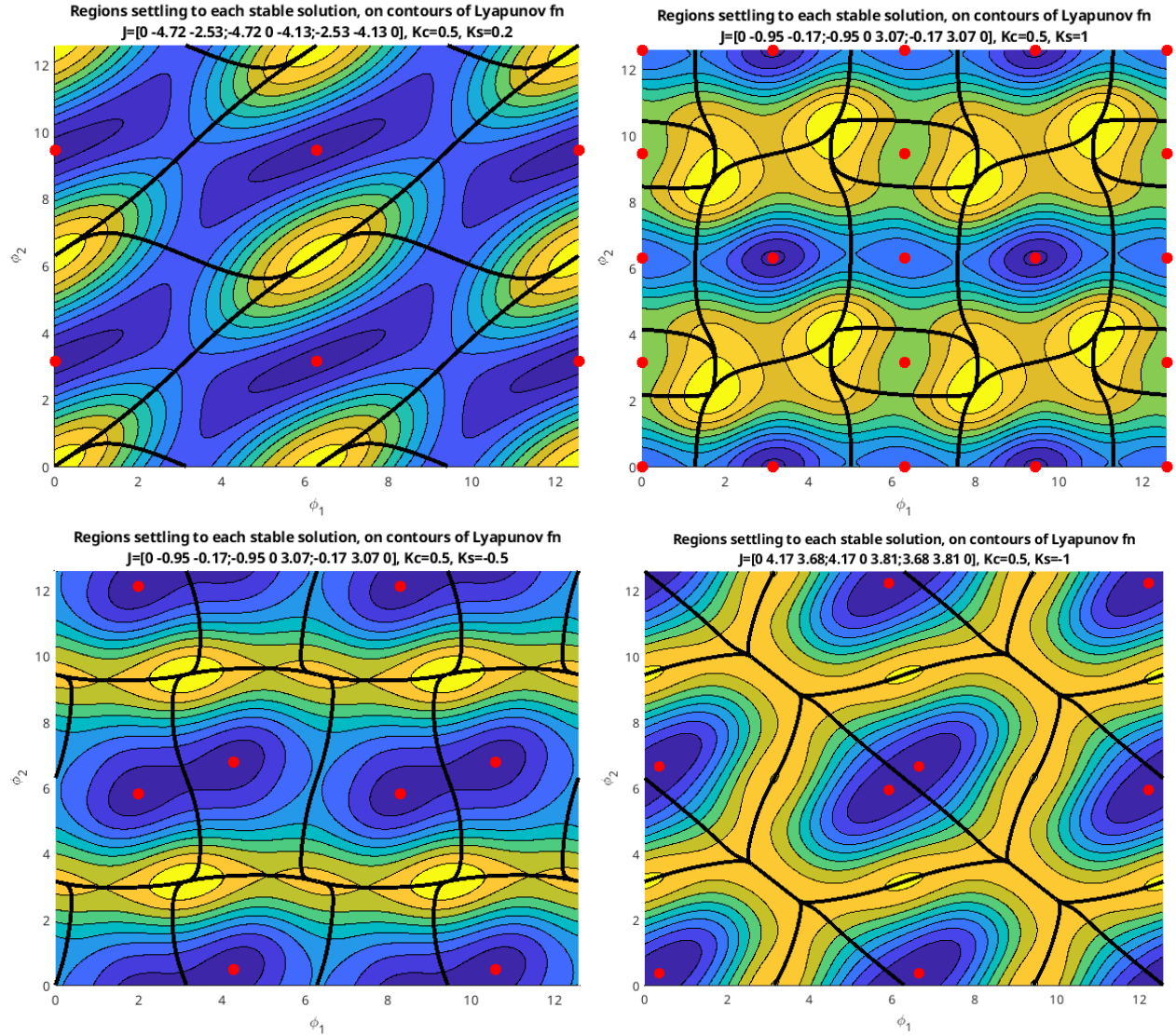
Fig. 23: Presumed ridges of the Lyapunov function, plotted on top of the level sets of the Lyapunov function, for various values of J and K_s (K_c is fixed at $\frac{1}{2}$). Stable solutions of the Kuramoto equations are shown as red dots. To showcase the continuous and closed nature of these curves, ϕ_1 and ϕ_2 are plotted between 0 and 4π instead of 0 and 2π. Deep blue represents the lowest values of the Lyapunov function, and yellow represents the highest values.
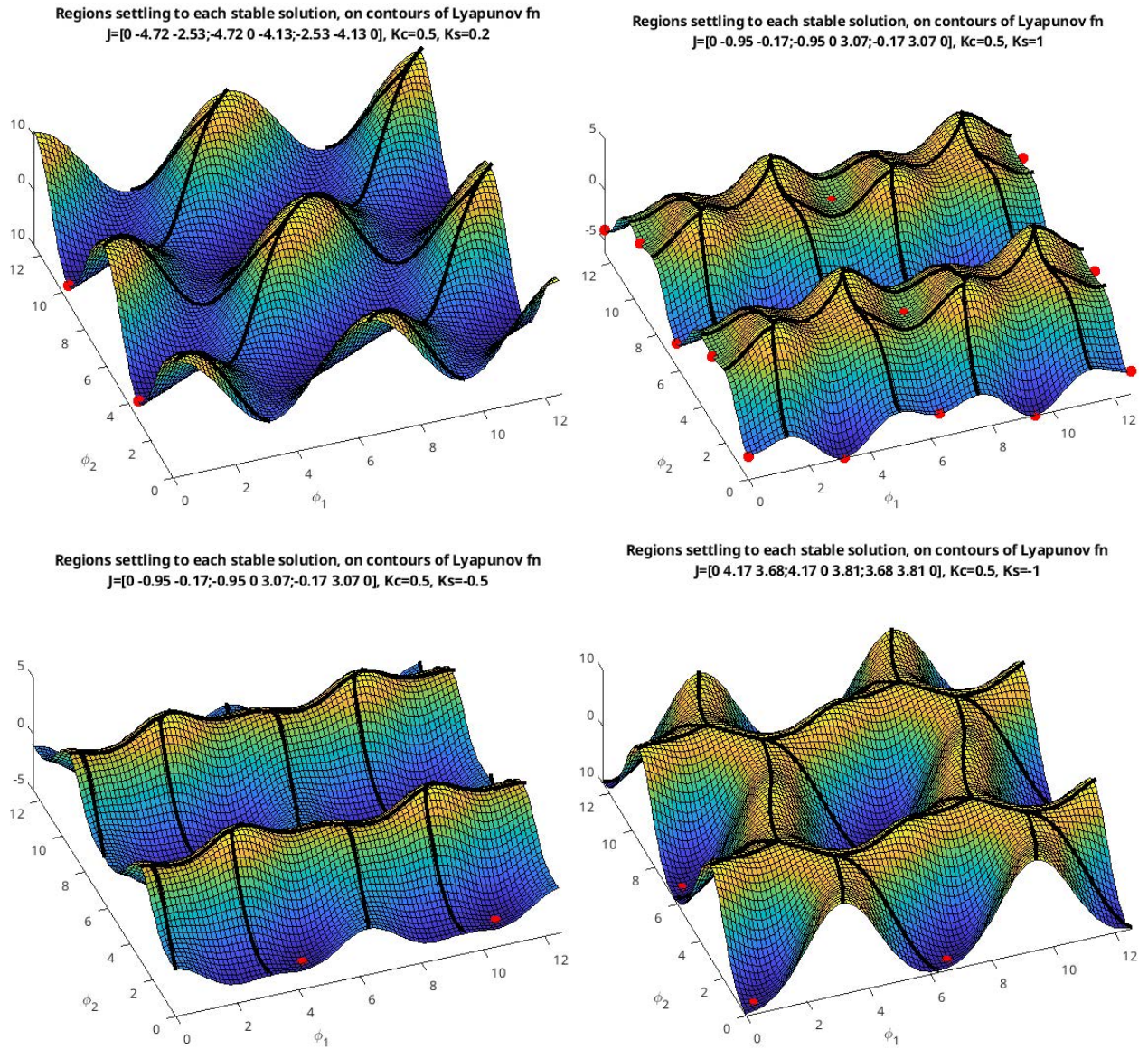
Fig. 24: Ridges from Figure 23, on a 3D plot of the Lyapunov function.

3.4 Eigenvalues of the Lyapunov Hessian for Grid Points

In general, a solution to the Kuramoto equations is stable if and only if the Jacobian of the Kuramoto equations, $\frac{d}{d\vec{\phi}}\frac{d\vec{\phi}}{dt}$, is negative definite. Equivalently, the Hessian of the Lyapunov function must be positive definite (PD), *i.e.*, all of its eigenvalues must be positive.

First, let us determine the relationship between the stability of $\vec{\phi} \in \{0, \pi\}^n$ (which, as mentioned in Section 3.2.3, will always be operating points) and the ratio $\frac{K_s}{K_c}$. To do so, we split the Lyapunov function onto two functions: one representing contributions of couplings between spins, and one representing the contribution of SYNC: $E(\vec{\phi}) = E_c(\vec{\phi}) + E_s(\vec{\phi})$, where

$$E_c(\vec{\phi}) = -\frac{1}{2}\sum_i\sum_j J_{ij}\cos(\phi_i - \phi_j), \tag{76}$$

$$E_s(\vec{\phi}) = -\frac{K_s}{2K_c}\sum_i\cos(2\phi_i). \tag{77}$$

So, by the linearity of differentiation,

$$\nabla^2 E(\vec{\phi}) = \nabla^2 E_c(\vec{\phi}) + \nabla^2 E_s(\vec{\phi}), \tag{78}$$

where $\nabla^2 E_c(\vec{\phi})$ does not depend on $\frac{K_s}{K_c}$ and

$$\nabla^2 E_s(\vec{\phi}) = \frac{2K_s}{K_c}\text{diag}\left(\cos(2\phi_1), \ldots, \cos(2\phi_n)\right). \tag{79}$$

At grid points, $\cos(2\phi_i) = 1$, so $\nabla^2 E_s(\vec{\phi})$ becomes $2\frac{K_s}{K_c}$ times the identity matrix. This matrix has the form $\nabla^2 E_c(\vec{\phi}) + 2\frac{K_s}{K_c}I$; as such, the eigenvalues of $\nabla^2 E(\vec{\phi})$ have a linear relationship with $\frac{K_s}{K_c}$:

$$\lambda_i\left\{\nabla^2 E(\vec{\phi})\right\} = \lambda_i\left\{\nabla^2 E_c(\vec{\phi})\right\} + 2\frac{K_s}{K_c}. \tag{80}$$

This has a few relevant consequences:

1. If a grid point is stable for $K_s = K_{s,0}$, then it will be stable $\forall K_s \geq K_{s,0}$.

2. Fix $\vec{\phi}_0 \in \{0, \pi\}^n$, and denote the minimum eigenvalue of the Hessian as $\lambda_{\min, \vec{\phi}_0}$. If $\lambda_{\min, \vec{\phi}_0}$ is positive, then the grid point $\vec{\phi}_0$ is stable, as all of the eigenvalues of the Lyapunov Hessian are positive. If we set K_s to some constant and order $\lambda_{\min, \vec{\phi}}$, for all grid points, that ordering will remain constant as we vary K_s. Thus, we only need to consider the eigenvalues of $\nabla^2 E_c(\vec{\phi})$ to determine the which grid point becomes stable first as we cycle K_s.

Analytically determining a relationship between the eigenvalues of the Lyapunov Hessians for the two-dimensional (three-spin) Kuramoto system and the corresponding Hamiltonian is a direction for future research. As of now, we have not analytically found a conclusive relationship. However, we have empirically seen a connection between the minimum Hamiltonian and the grid point with the largest $\lambda_{\min}$. Two examples of this phenomenon, as well as the linear relationship between $\lambda_{\min}$ and K_s, are shown in Figure 25. The smallest eigenvalue of $\nabla^2 E(\vec{\phi})$ at grid points is plotted with respect to K_s, for a randomly-generated two-dimensional (three-spin) and three-dimensional (four-spin) Ising problem.

In both cases, the grid point with the largest eigenvalue also corresponded to the lowest Hamiltonian values. This indicates that there might be some relationship, either analytical or probabilistic, between the grid point that minimizes the Ising Hamiltonian and the one with the largest $\lambda_{\min}$. As the value of $\lambda_{\min}$

determines at which K_s the grid point first becomes stable, this would mean that the grid point that becomes stable first under K_s cycling (as we increase K_s starting from some negative value) is likely to be the one that minimizes the Ising Hamiltonian.

In lieu of analytical results, we have turned to Monte Carlo-style experiments to quantify when and how often this phenomenon occurs.
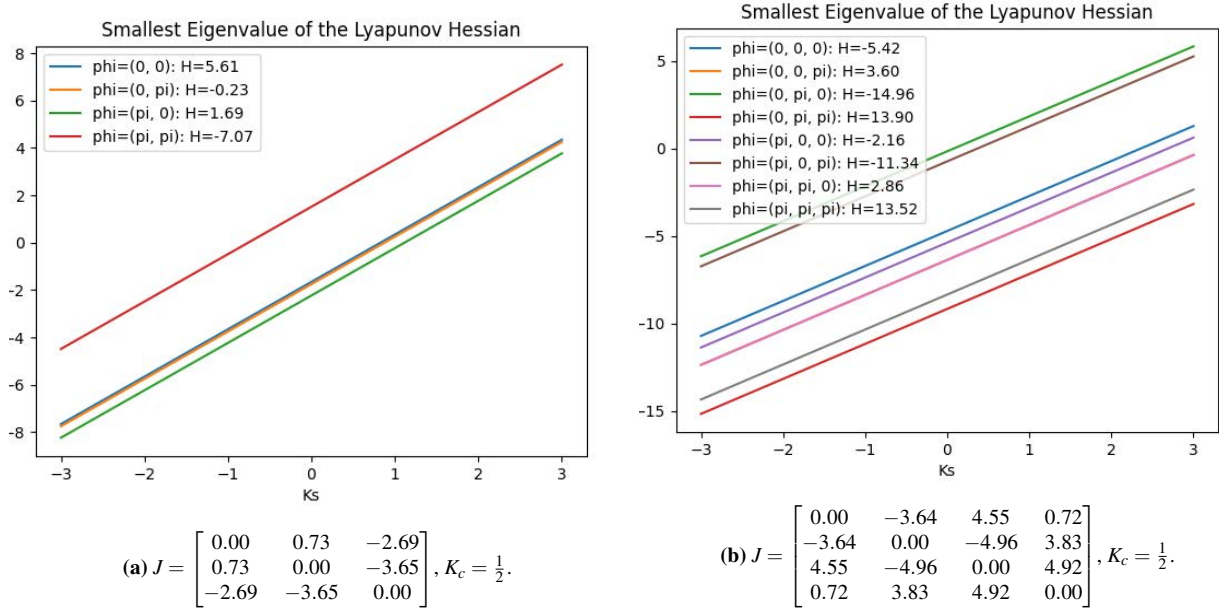

Fig. 25: Minimum eigenvalue of the Lyapunov Hessian, plotted with respect to K_s.

3.5 Results: Monte Carlo Simulations

Although there is not a direct relationship between the eigenvalues of $\nabla^2 E(\vec{\phi})$ and the value of the Ising Hamiltonian at grid points, we empirically saw a correlation between the configuration of spins producing the minimum Hamiltonian and the grid point that becomes stable with the lowest K_s.

To verify this, we ran Monte Carlo simulations to determine the probability that the grid point with the largest $\lambda_{\min}$ corresponds to the "ground state," or minimum Ising Hamiltonian. To do so, we first generate some large number of Ising coupling matrices. For every coupling matrix, we perform a brute force search over all configurations of spins ($\vec{s} \in \{\pm1\}^n$, where n is the number of non-fixed spins), storing the spin configurations that produce the minimum Hamiltonian and the maximum $\lambda_{\min}$. We then determine the Hamming distance between the two sets of spin values (*i.e.*, the number of spins that differ between the two). The proportion of Hamming distances that are zero estimates the desired probability.

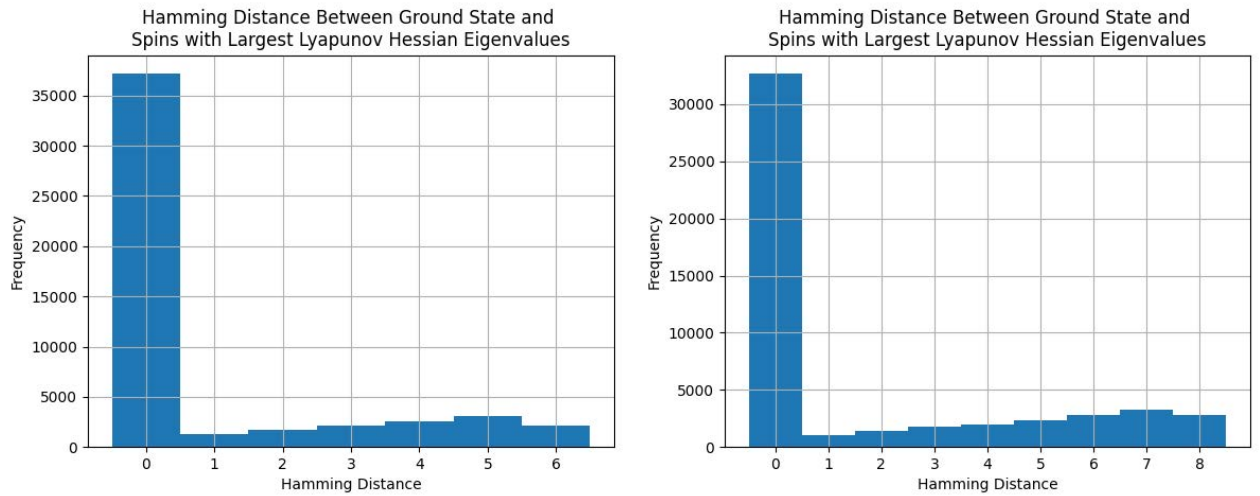#### 3.5.1 Uniform Random Coupling Matrices

First, we generated random coupling matrices, where each non-diagonal element of J was sampled from uniform distribution between -10 and 10. For problem sizes (number of spins, minus one) ranging from two to eight, we generated 50,000 coupling matrices of each size. Histograms of the Hamming distances produced by these experiments are in Figure 26. As the problem size increases, the height of the "0' Hamming distance' bin decreases (resulting in less correlation between the ground state and the spins producing the largest $\lambda_{\min}$). Outside of the "0 Hamming distance" bin, the heights of the bins increase as the Hamming distance increases. It is unclear whether this fact is relevant.

We also plotted histograms of the difference between the $\lambda_{\min}$ of the grid point that minimizes the Ising

Hamiltonian and the largest $\lambda_{\min}$ across all grid points (Figure 27). For all problem sizes shown, there is a large "zero" bin (the eigenvalues are identical), with a small Gaussian-looking tail pointing toward the left. The height of this Gaussian increases as the problem size increases, indicating increased differences in the two eigenvalues (the eigenvalues of the Lyapunov Hessian for the ground state deviate more from the largest $\lambda_{\min}$ for all grid points).
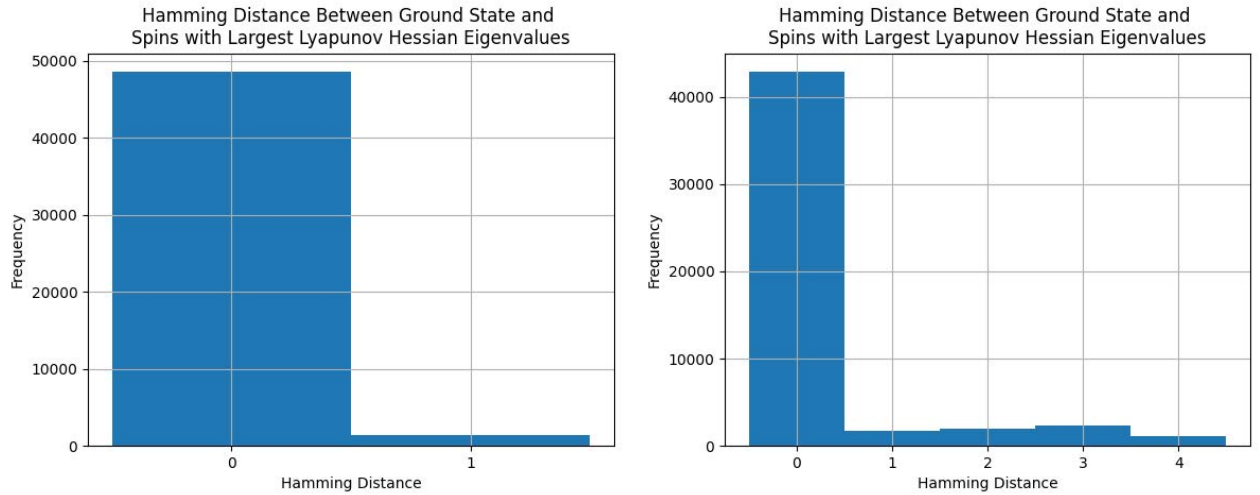
(a) Two non-fixed spins: 2.860% of 50,000 runs yielded a non-zero Hamming distance.

**(b)** Four non-fixed spins: 14.284% of 50,000 runs yielded a non-zero Hamming distance.

(c) Six non-fixed spins: 25.650% of 50,000 runs yielded a non-zero Hamming distance.

(d) Eight non-fixed spins: 34.686% of 50,000 runs yielded a non-zero Hamming distance.

Fig. 26: Histogram of Hamming distances for uniform random Ising coupling matrices, over 50,000 runs.

3.5.2 MU-MIMO Coupling Matrices

Results from random coupling matrices, however, do not necessarily apply to real-world problems. So, we also performed Monte Carlo experiments on small MU-MIMO problems with varying sizes and numbers of scatterers.[23] We chose MU-MIMO problems in particular because OIMs perform well on classes of these
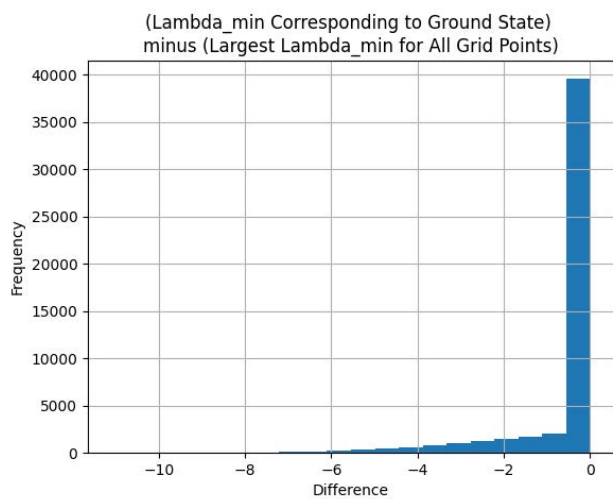
---

[23]The size of the problems we can consider is limited, as each Monte Carlo iteration involves doing a brute force search over $O(2^n)$ different spin configurations.
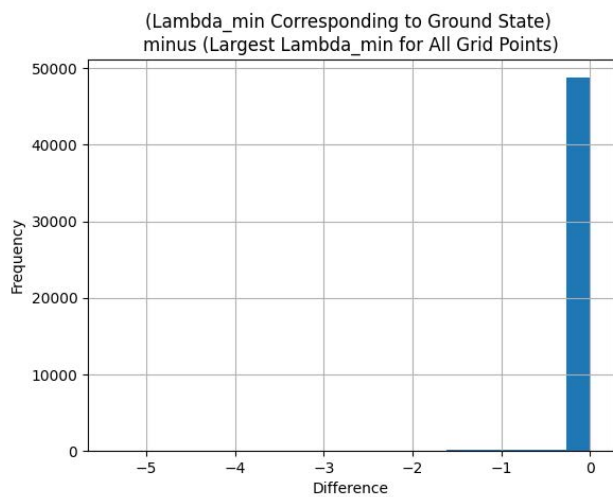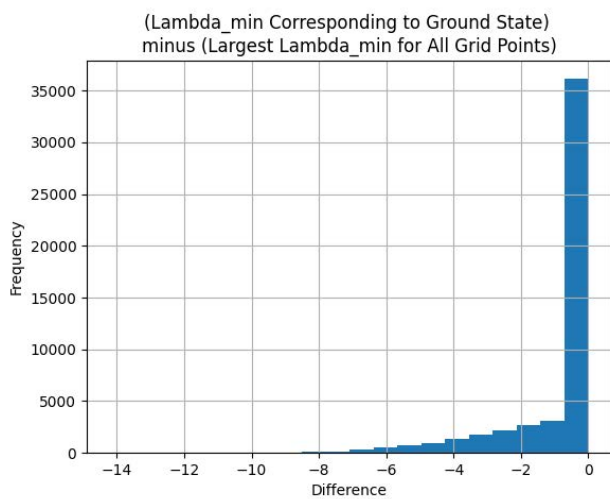
(a) Two non-fixed spins.
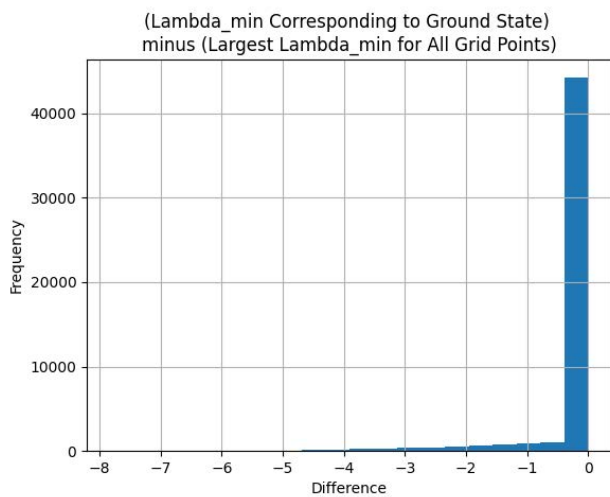
(b) Four non-fixed spins.

(c) Six non-fixed spins.

(d) Eight non-fixed spins.

Fig. 27: Histogram of $\{\lambda_{\min}$ of the ground state $-$ largest $\lambda_{\min}$ across all grid points$\}$, for the random coupling matrices from Figure 26.

problems [27], and it is of interest to investigate when and why this is the case. The process for generating coupling matrices, as well as the relevance of different problem parameters, is described in Section 2.5.1.

Transmitters	Receivers	Scatterers	% of Runs with Non-zero Hamming Distance
6	12	20	0.264%
6	12	12	0.768%
6	12	6	2.624%
6	12	3	7.740%
6	12	2	12.352%
10	20	100	0.038%
10	20	10	2.78%
10	20	5	9.46%
10	20	3	17.824%
10	20	2	25.342%
12	24	100	0.024%
12	24	20	0.956%
12	24	12	2.916%
12	24	6	10.836%
12	24	2	31.158%

Fig. 28: Percentage of 25,000 Monte Carlo runs that yielded a non-zero Hamming distance, for Ising coupling matrices derived from MU-MIMO problems, varying the problem size and number of scatterers.

Some results for problems with six, ten, and twelve non-fixed spins are in Figure 28. For a high enough number of scatterers (*e.g.*, twelve scatterers for the size-six problem and 20 for the size-twelve problem), the ground state matched the grid point with the highest λ_{min} for over 99% of Monte Carlo iterations. Then, as the number of scatterers decreases, so does the proportion of runs where the same set of phases yields the minimum Hamiltonian and largest λ_{min}. Each problem size appears to have a "threshold" number of scatterers, below which the probability of a non-zero Hamming distance increases rapidly. This threshold is positively correlated with problem size. These results indicate that OIM, under parameter cycling, is less likely to find global minima when the number of scatterers is below a threshold. In the future, we hope to verify this via simulation of the Kuramoto equations.

References

[1] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*. Springer, 1972, pp. 85–103.

[2] M. Garey and D. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W.H. Freeman, 1979.

[3] A. Lucas, "Ising formulations of many NP problems," *Frontiers in Physics*, vol. 2, p. 5, 2014.

[4] Z. Bian, F. Chudak, W. G. Macready, and G. Rose, "The Ising model: teaching an old problem new tricks," *D-Wave Systems*, vol. 2, 2010.

[5] M. W. Johnson, M. H. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A. J. Berkley, J. Johansson, P. Bunyk, *et al.*, "Quantum annealing with manufactured spins," *Nature*, vol. 473, no. 7346, pp. 194–198, 2011.

[6] K. Boothby, P. Bunyk, J. Raymond, and A. Roy, "Next-Generation Topology of D-Wave Quantum Processors," *arXiv:2003.00133*, 2020.

[7] Z. Bian, F. Chudak, R. Israel, B. Lackey, W. G. Macready and A. Roy, "Discrete optimization using quantum annealing on sparse Ising models," *Frontiers in Physics*, vol. 2, p. 56, 2014.

[8] A. Marandi, Z. Wang, K. Takata, R. L. Byer and Y. Yamamoto, "Network of time-multiplexed optical parametric oscillators as a coherent Ising machine," *Nature Photonics*, vol. 8, no. 12, pp. 937–942, 2014.

[9] T. Wang and J. Roychowdhury, "Oscillator-based Ising Machine," *arXiv:1709.08102*, 2017.

[10] T. Wang, L. Wu and J. Roychowdhury, "New Computational Results and Hardware Prototypes for Oscillator-based Ising Machines," in *Proc. IEEE DAC*, 2019, pp. 239:1–239:2.

[11] T. Wang, L. Wu, P. Nobel, and J. Roychowdhury, "Solving combinatorial optimisation problems using oscillator based Ising machines," *Natural Computing*, pp. 1–20, April 2021.

[12] R. Afoakwa, Y. Zhang, U. K. R. Vengalam, Z. Ignjatovic, and M. Huang, "BRIM: Bistable Resistively-Coupled Ising Machine," in *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE, 2021, pp. 749–760.

[13] J. Roychowdhury, "Bistable Latch Ising Machines," in *Proc. UCNC*, ser. LNCS sublibrary: Theoretical computer science and general issues, October 2021, Web link.

[14] A. K. Singh, K. Jamieson, D. Venturelli, and P. McMahon, "Ising Machines' Dynamics and Regularization for Near-Optimal Large and Massive MIMO Detection," *arXiv:2105.10535v1*, May 2021.

[15] M. Kim, D. Venturelli, and K. Jamieson, "Leveraging quantum annealing for large MIMO processing in centralized radio access networks," in *Proceedings of the ACM Special Interest Group on Data Communication*. ACM, 2019, pp. 241–255.

[16] J. Roychowdhury, J. Wabnig, and K. P. Srinath, "Performance of Oscillator Ising Machines on Realistic MU-MIMO Decoding Problems," *Research Square preprint (Version 1)*, 22 September 2021, Web link to preprint.

[17] J. Bisgard, *Analysis and Linear Algebra: The Singular Value Decomposition and Applications*. American Mathematical Society, 2021.

[18] L. O. Chua, J. Yu, and Y. Yu, "Negative resistance devices," *International Journal of Circuit Theory and Applications*, vol. 11, no. 2, pp. 161–186, 1983.

[19] "MATLAB Phased Array System Toolbox," Available at https://www.mathworks.com/help/phased/index.html.

[20] R. W. Heath, N. Gonzalez-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An overview of signal processing techniques for millimeter wave MIMO systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 3, pp. 436–453, 2016.

[21] T. F. Chan and P. C. Hansen, "Computing truncated singular value decomposition least squares solutions by rank revealing QR-factorizations," *SIAM Journal on Scientific and Statistical Computing*, vol. 11, no. 3, pp. 519–530, 1990.

[22] R. Horn and C. Johnson, *Matrix Analysis*.   Cambridge University Press, 1985.

[23] J. Roychowdhury, "Numerical simulation and modelling of electronic and biochemical systems," *Foundations and Trends in Electronic Design Automation*, vol. 3, no. 2-3, pp. 97–303, December 2009. [Online]. Available: http:/dx.doi.org/10.1561/1000000009

[24] E. Allgower and K. Georg, *Numerical Continuation Methods*.   New York: Springer-Verlag, 1990.

[25] J. Roychowdhury and R. Melville, "Delivering Global DC Convergence for Large Mixed-Signal Circuits via Homotopy/Continuation Methods," *IEEE Trans. CAD*, vol. 25, no. 1, pp. 66–78, Jan 2006.

[26] H. Goto, "Bifurcation-based adiabatic quantum computation with a nonlinear oscillator network," *Scientific Reports*, vol. 6, no. 1, feb 2016. [Online]. Available: https://doi.org/10.1038%2Fsrep21686

[27] J. Roychowdhury, J. Wabnig, and K. P. Srinath, "Performance of Oscillator Ising Machines on Realistic MU-MIMO Decoding Problems," *Science Advances*, submitted.