ML on the QCi quantum computer

Sebastian Grieninger and Dmitri Kharzeev

Stony Brook University

UNTERSTÜTZT VON / SUPPORTED BY

Alexander von HUMBOLDT STIFTUNG

C2QA Co-design Center for Quantum Advantage

In progress with Nick Chancellor, David Haycraft, Ali Miri, Lac Nguyen

Outline

1. Neural networks: an introduction

2. Hopfield network

3. Quantum neural networks

4. Current implementation on QCi hardware

Neural networks

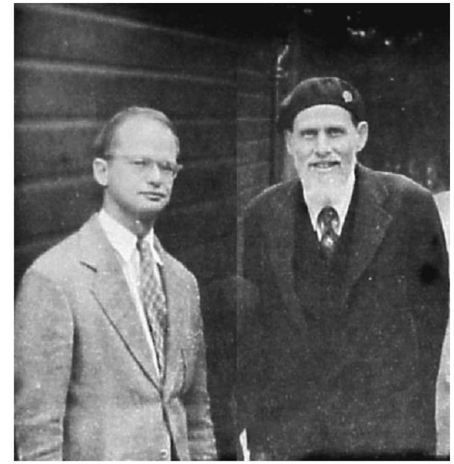
A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY

Warren S. McCulloch and Walter H. Pitts

Because of the "all-or-none" character of nervous activity, neural events and the relations among them can be treated by means of propositional logic. It is found that the behavior of every net can be described in these terms, with the addition of more complicated logical means for nets containing circles; and that for any logical expression satisfying certain conditions, one can find a net behaving in the fashion it describes. It is shown that many particular choices among possible neurophysiological assumptions are equivalent, in the sense that for every net behaving under one assumption, there exists another net which behaves under the other and gives the same results, although perhaps not in the same time. Various applications of the calculus are discussed.

$$\sigma_{\mathrm{i}}(t+1) = \mathrm{sgn}\left[\sum_{\mathrm{j}\neq \mathrm{i}} J_{\mathrm{ij}}\sigma_{\mathrm{j}}(t) - \theta_{\mathrm{i}}\right]$$

Bulletin of Mathematical Biophysics, Vol. 5, 1943, p. 115-133



McCulloch (right) and Pitts (left) in 1949

Active neuron: $\sigma_i = +1$

Inactive neuron: $\sigma_i = -1$

Activation threshold: θ

Weights (synapses): J_{ij}

Review: arXiv:2412.18030

Neural networks

Hopfield 1982:

Nobel prize in Physics, 2024,
with G. Hinton (“Boltzmann machines”)

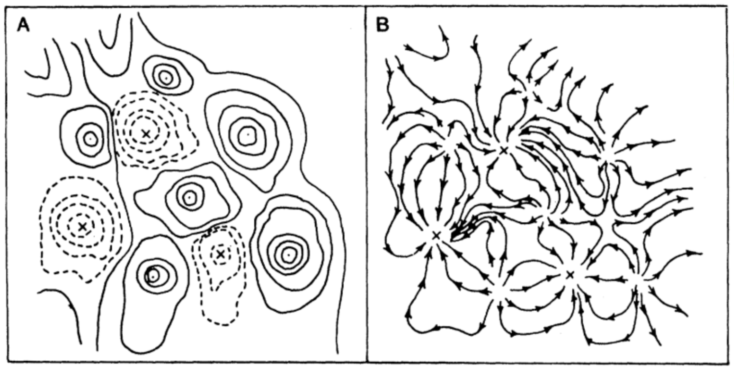
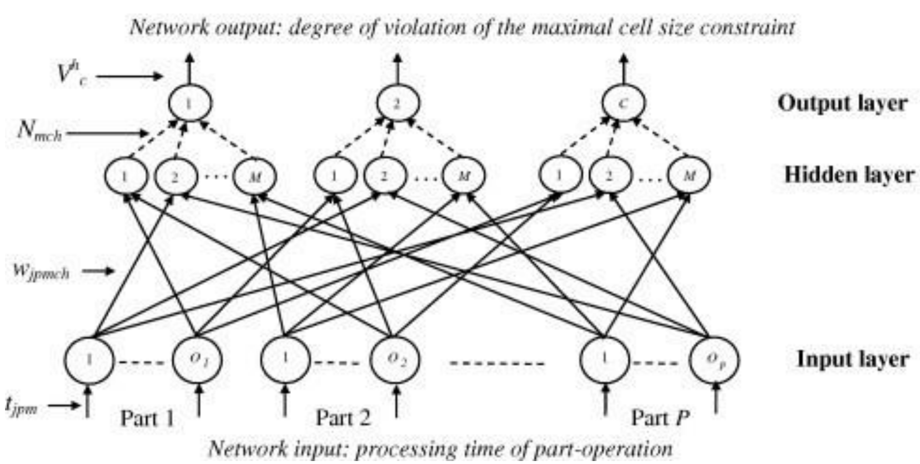


FIG. 2: Energy landscape and trajectories in a model of neural networks [39]. (A) Solid contours are above a mean level and dashed contours below, with X marking fixed points at the bottoms of energy valleys. (B) Corresponding dynamics, shown as a flow field.

The network is sliding down on a landscape, which is an effective energy function.



“Coming to a rest at the minimum of the energy is a computation, analogous to recalling a memory.”

Review: W. Bialek, arXiv:2412.18030

Standard Hopfield network

$$E(\boldsymbol{\sigma}) = -\frac{1}{2}\sum_{\mathrm{ij}} J_{\mathrm{ij}}\sigma_{\mathrm{i}}\sigma_{\mathrm{j}},$$

where the thresholds $\theta_{\mathrm{i}} = 0$ $J_{\mathrm{ij}} = J_{\mathrm{ji}}$

We can program the network so that its stable final states are close to some specific stored patterns by choosing

$$J_{\mathrm{ij}} = J\sum_{\mu=1}^{K} \xi_{\mathrm{i}}^{\mu}\xi_{\mathrm{j}}^{\mu},$$

for K binary patterns we'd like to store. Example: ferromagnet $\boldsymbol{\sigma} = \boldsymbol{\xi}^1$

Local learning rule: update the matrix J depending on the outcome (Hebbian, Storkey, …)

The local learning rule

If the system is in a state $\boldsymbol{\sigma}^t$ at some moment in time, and we would like to add this to list of stored patterns, then the synaptic strengths should be adjusted as

$$J_{\rm ij} \rightarrow J_{\rm ij} + J\sigma_{\rm i}^t\sigma_{\rm j}^t. \quad (5)$$

We notice that this is a local learning rule: what happens at the synapse between neurons i and j depends only on the states of those two neurons, and not on the rest of the network. This is surprising because ground states are a property of the network as a whole, yet they can be programmed without global knowledge.

Review: W. Bialek, arXiv:2412.18030

This is analogous to the universal quantum computing theorem:
all quantum computations can be performed by using only single- and primary two-qubit gates.

Modern Hopfield network

Replace the original energy functional

$$E(\boldsymbol{\sigma}) = -\frac{J}{2}\sum_{\mathrm{ij}}\sum_{\mu=1}^{K}\xi_{\mathrm{i}}^{\mu}\xi_{\mathrm{j}}^{\mu}\sigma_{\mathrm{i}}\sigma_{\mathrm{j}} = -\frac{J}{2}\sum_{\mu=1}^{K}\left(\boldsymbol{\xi}^{\mu}\cdot\boldsymbol{\sigma}\right)^{2}$$

by a more complicated function

$$E(\boldsymbol{\sigma}) = -J\sum_{\mu=1}^{K}F\left(\boldsymbol{\xi}^{\mu}\cdot\boldsymbol{\sigma}\right).$$

For example, for

$$F(x) = e^{x}$$

one can store a huge number of patterns,

$$\log \bar{K} \sim \alpha N$$

An ML algorithm

The proposed ML algorithm is as follows:

Perform minimization of fully connected Ising model with fixed boundary condition on one side (input neuron layer) + open boundary condition on other side (output neuron layer). Matrix J_{ij} is fixed by stored patterns; Define symmetric matrix from asymmetric **J** determined by stored p patterns by

$$\tilde{\mathbf{J}} \equiv \mathbf{J} \cdot \mathbf{J}^T$$

Once energy function minimized for given initial quantum state (describing the input neuron layer): find quantum state on other boundary.

Then compute overlap of this quantum state with stored patterns (next slide) and pick largest overlap (fidelity) – this provides output of network.

The minimization algorithm

Once patterns stored (matrix **J** determined): minimize energy of Ising system

$$E(\boldsymbol{\sigma}) = -\frac{1}{2}\sum_{\mathrm{ij}} J_{\mathrm{ij}}\sigma_{\mathrm{i}}\sigma_{\mathrm{j}},$$

providing fixed boundary condition on input layer (determined by particular input), and using open boundary condition on output layer.

Once minimization done: find corresponding state described by $\tilde{\xi}$.
Then: compute projections of $\tilde{\xi}$ on all stored patterns ξ^{μ} :

$$|\langle \xi^{\mu} | \tilde{\xi} \rangle|$$

Largest overlap defines the output of algorithm.

The Storkey Learning Rule

Introduces a local field correction that removes correlations between patterns, increasing network capacity

$$J_{ij}^{Storkey} = J_{ij}^{old} + {}^{1}/_{N}\xi_i^\mu\xi_j^\mu - {}^{1}/_{N}\xi_i^\mu h_{j\to i}^\mu - {}^{1}/_{N} h_{i\to j}^\mu \xi_j^\mu, \quad J_{ii} = 0$$

Local field at neuron *i* due to all neurons except *j* when pattern μ is stored:

$$h_{j\to i}^\mu = \sum_{k\neq i,\, k\neq j} J_{ik}^{old}\xi_k^\mu$$

1. Increased Capacity

Stores ~N/√(2logN) patterns vs 0.14N for standard Hebbian

2. Reduced Spurious Memories

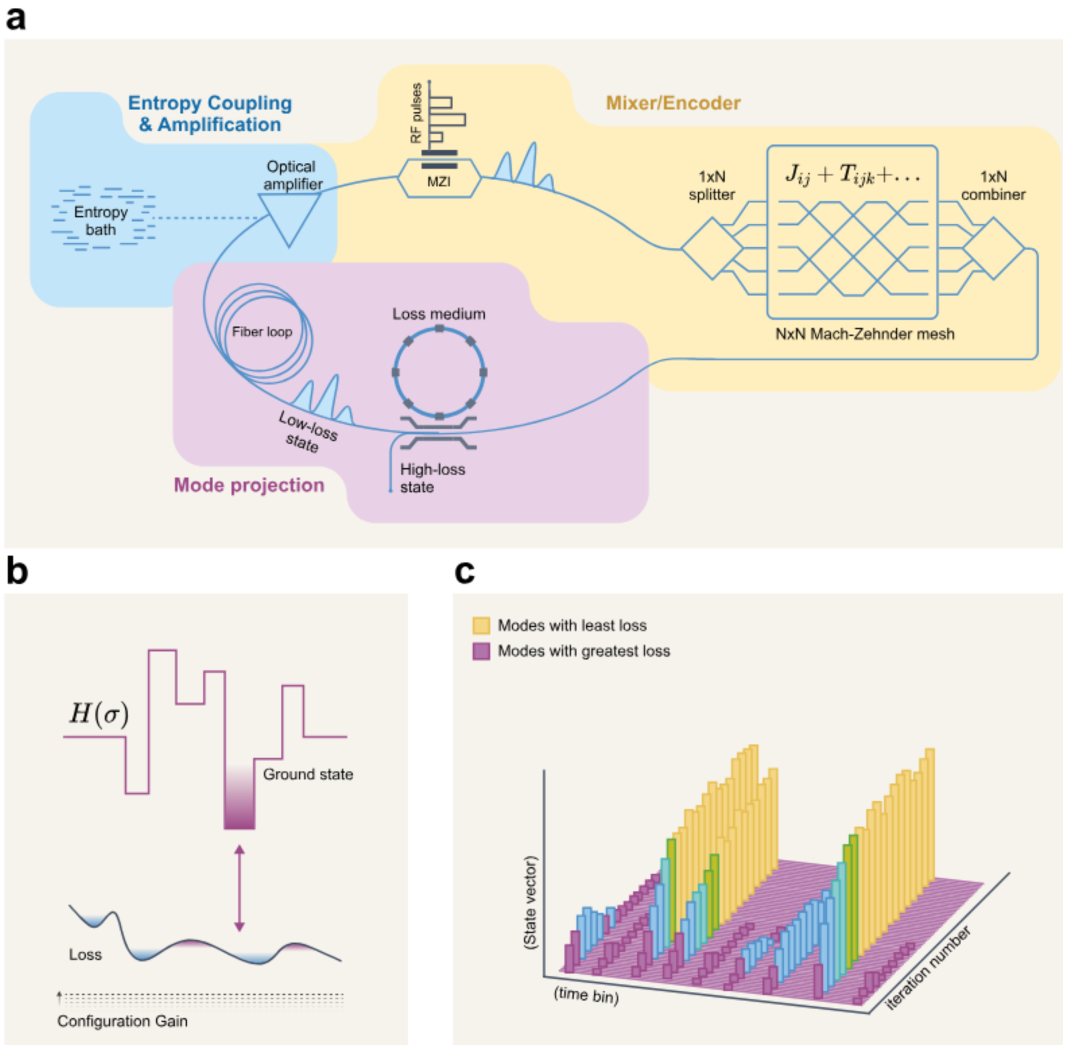
Correction terms reduce creation of spurious attractors

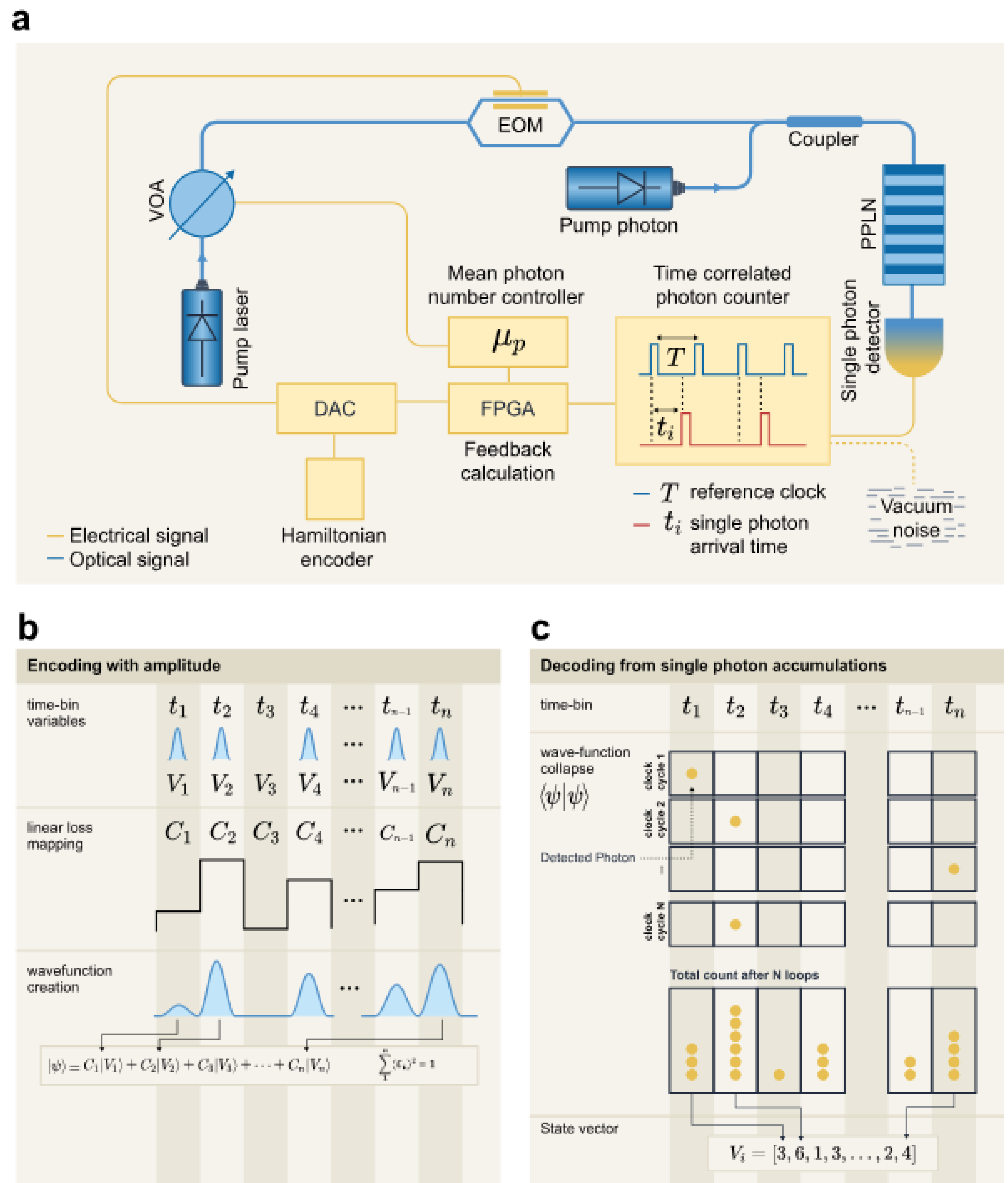
3. Improved Convergence

Networks converge to stored patterns more reliably

Key insight: Accounts for interactions between patterns by considering existing network weights when adding new patterns, creating better separation in the energy landscape

Dirac 3 "quantum computer"




a
EOM
Coupler
VOA
Pump photon
PPLN
Mean photon
number controller
Time correlated
photon counter
Pump laser
Single photon
detector
μ_p
T
t_i
DAC
FPGA
Feedback
calculation
T reference clock
t_i single photon
arrival time
Vacuum
noise
Electrical signal
Optical signal
Hamiltonian
encoder
b
Encoding with amplitude
time-bin
variables
t_1 t_2 t_3 t_4 ··· t_{n-1} t_n
V_1 V_2 V_3 V_4 ··· V_{n-1} V_n
linear loss
mapping
C_1 C_2 C_3 C_4 ··· C_{n-1} C_n
wavefunction
creation
$|\psi\rangle = C_1|V_1\rangle + C_2|V_2\rangle + C_3|V_3\rangle + \cdots + C_n|V_n\rangle$
c
Decoding from single photon accumulations
time-bin
t_1 t_2 t_3 t_4 ··· t_{n-1} t_n
wave-function
collapse
$\langle\psi|\psi\rangle$
clock cycle 1
clock cycle 2
Detected Photon
clock cycle N
Total count after N loops
State vector
$V_i = [3, 6, 1, 3, \ldots, 2, 4]$

Quantum Computing Inc.'s Entropy Quantum Computer

- Energy-efficient, robust, scalable, affordable
- Optimization problems

Key Features

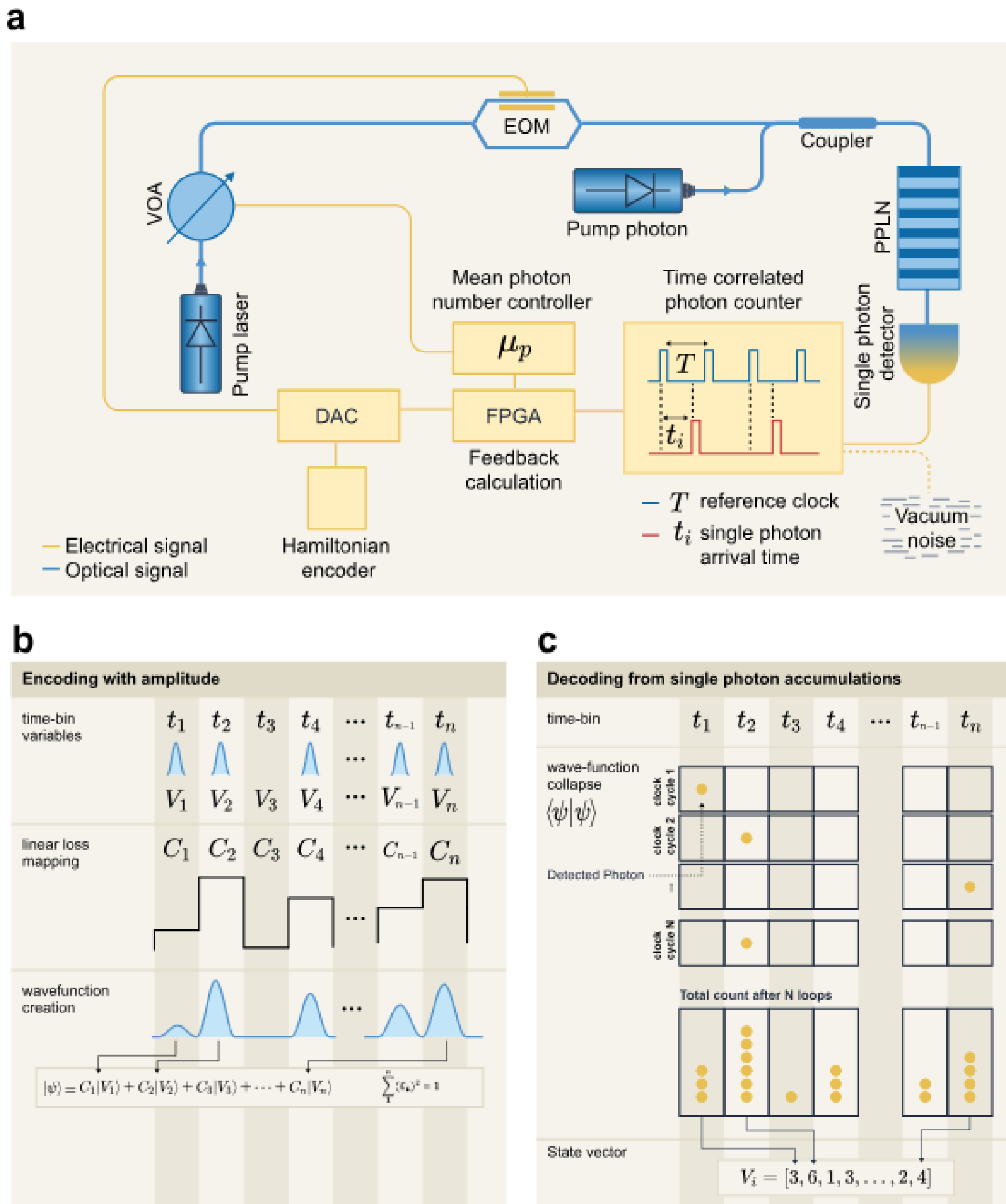
- Non-binary qudits 200 discrete modes per qudit
- Room temperature operation

Operation

- Problem → photonic architecture
- Optical feedback loops modulate variable interaction
- System settles into ground state (optimal solution)
- Solution readout

NASA:

- Radar image reconstruction (phase unwrapping)
- LiDAR spectral analysis from lower Earth orbit



Dirac-3 Energy Functional

Ising Hamiltonian Input

$$E = \sum_i C_i x_i + \sum_{i,j} J_{ij} x_i x_j + \text{higher-order terms}$$

- x_i: Variables (represented by qudits)
- C_i: Linear coefficients (positive, negative, or zero)
- J_{ij}: Coupling coefficients (any real number)
- Higher-order interactions supported (cubic, quartic, etc.)

Example Polynomial

$$E = 3x_4 + 2.1x_1^2 + 1.5x_2^2 + 7.9x_2x_3 + x_2x_4^2 + x_3^3$$

Dirac-3 natively handles polynomials beyond quadratic terms

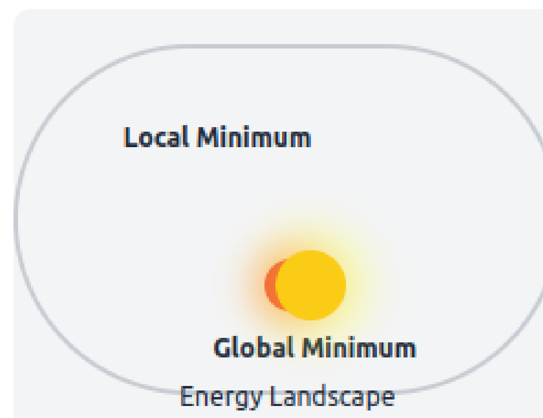
Key Properties

Variables: Integer values (qudits)

Sum Constraint: 1-10000

Relaxation Schedule: Controls quantum dissipation (1-4)

Objective: Find lowest energy state



Initial State Implementation

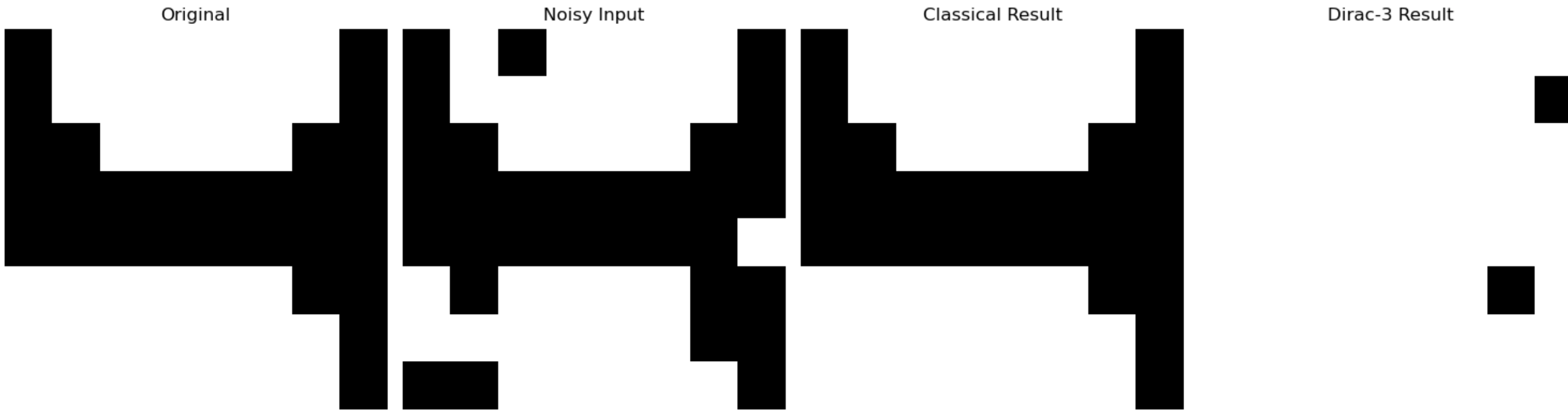
Problem: Dirac 3 starts minimization from random state instead of particular state (input pattern).

Solution: Encode input pattern as magnetic field (bias term) placing us on correct starting point in energy landscape.

$$E = -\frac{1}{2}\sum_{i,j} J_{ij}\, s_i s_j - h \sum_{i} h_i s_i$$

New parameter h: determines how strong bias term ”pulls” us to initial pattern. In practice, after training: determine which strength h works best.

Some first results




```
Weights loaded from trained_weights.npy
Dirac-3 time allocation: {'metered': True, 'seconds': 5

Running classical solver...

Submitting to Dirac-3...
2025-02-06 18:43:12 - Dirac allocation balance = 584 s
2025-02-06 18:43:13 - Job submitted: job_id='67a549106f
2025-02-06 18:43:13 - QUEUED
2025-02-06 18:43:16 - RUNNING
2025-02-06 18:44:57 - COMPLETED
2025-02-06 18:45:00 - Dirac allocation balance = 557 s

Performance Comparison:
--------------------------------------------------
Input Digit: 4 | Noise Level: 10.0%
Network Size: 8x8

Classical Solver:
Time: 0.0010 seconds
Accuracy: 100.00%
Identified Digit: 4

Dirac-3 Solver:
Total Time: 109.9398 seconds
Accuracy: 59.38%
Identified Digit: 7
Energy: -490.72
```

Free account limits us to 100 neurons (or images smaller than 10x10

Classical results are surprisingly fast (even for larger images)

Simulation on Dirac3 still needs some refinement

Toward Quantum Advantage

$$H = -J \sum_{i,j} \sigma_i^z \sigma_j^z - h \sum_i \sigma_i^x$$

- The key addition is the transverse field term ➡ spins can rotate
- Have to solve Quantum Ising model!
- There exists mapping to classical Ising model but in 1 dimension higher (Trotter-Suzuki decomposition in Path integral Monte Carlo)