EigenFold, a Variational Quantum Eigensolver applied to Protein Folding

Arham Aneeq[a], Aarush Bindod[b], Abhiroop Gohar[c], V Hemal[d]
[a]Department of Metallurgical Engineering and Material Science, IIT Indore
[b], [c], [d]Department of Engineering Physics, IIT Indore

Abstract

EigenFold explores quantum approaches to the protein folding problem by formulating lattice-based folding under the hydrophobic–polar (HP) model as a Quadratic Unconstrained Binary Optimization (QUBO) problem and solving it with a Variational Quantum Eigensolver (VQE). The framework encodes spatial adjacency, self-avoidance, and hydrophobic interactions directly into a binary Hamiltonian mapped to Pauli operators. Using Qiskit's EstimatorV2, EigenFold identifies low-energy quantum states corresponding to physically plausible folds, often revealing multiple near-degenerate conformations. Results are analyzed across lattice geometries and peptide sequences, yielding insights into how quantum variational solvers capture the energy landscape structure of biomolecular systems.

Introduction

Protein folding is a complex optimization problem where a sequence of amino acids must find its lowest-energy 3D structure. Classical algorithms such as Monte Carlo or molecular dynamics require large computational resources and often get trapped in local minima. Quantum computing offers a new approach by representing the folding landscape as a quantum energy function that can be explored more efficiently.

EigenFold reformulates the lattice-based Hydrophobic–Polar (HP) folding model as a *Quadratic Unconstrained Binary Optimization (QUBO)* problem and solves it using the *Variational Quantum Eigensolver (VQE)*. The workflow maps residues to binary variables, encodes physical rules like adjacency and self-avoidance as QUBO terms, converts the model into a Pauli Hamiltonian, and minimizes it using VQE. The resulting quantum state provides a probability distribution over possible folds, allowing multiple stable conformations to be visualized and compared across lattice geometries.

Methods

QUBO Formulation

We represent the protein folding problem as a Quadratic Unconstrained Binary Optimization (QUBO) problem, where each binary variable $b_i \in \{0,1\}$encodes part of the position of a residue on a discrete lattice. The total energy of the system is expressed as:

$$E(b_1, \dots, b_N) = \sum_i a_i b_i + \sum_{i<j} Q_{ij} b_i b_j + C$$

Here, a_i are linear coefficients, Q_{ij} are pairwise couplings, and c is a constant offset. The optimization objective is to find a bitstring $\boldsymbol{b}$ that minimises E, corresponding to the folded ground state.

Binary Encoding

In earlier one-hot formulations we used, each residue was assigned one binary variable per lattice site, requiring $R \times S$ bits for a system of R residues and S lattice sites. This grows exponentially with lattice size and becomes infeasible for variational solvers.

To improve efficiency, we employed a binary positional encoding. Instead of assigning one variable per site, we encode the lattice index in binary form. For each residue r, its position $p_r$is represented as:

$$p_r = \sum_{k=0}^{n_b-1} 2^k \, x_{r,k}$$

where $x_{r,k} \in \{0,1\}$and $n_b = \lceil \log_2 S \rceil$. This reduces the total number of binary variables to $R \times n_b$, achieving an exponential compression in qubit count while maintaining a continuous encoding of space.

Energy Components

The total folding energy comprises four terms:

$$E = E_{backbone} + E_{collision} + E_{contact} + E_{bias}$$

Backbone Adjacency

Sequential residues must be placed on adjacent lattice sites. This is enforced by minimizing the squared difference between their encoded positions:

$$E_{backbone}\,(r, r+1) = C(p_r + 1 - p_r)^2$$

In the QUBO form this expands to

$$E_{backbone} = C \sum_{\{i,j\}} \left(w_i^2 x_{r,i} + w_j^2 x_{r+1,j} - 2 w_i w_j x_{r,i} x_{r+1,j} \right)$$

Where $w_i = 2^i$ are binary weights.

Self-Collision Penalty

To prevent steric overlap, any two residues occupying the same lattice site are penalized through their positional difference:

$$E_{\text{collision}}(i,j) = B\,(p_i - p_j)^2$$

This term expands identically to the adjacency expression but applies to all distinct residue pairs (i, j), ensuring each residue has a unique lattice coordinate. A small cross-term penalty $B/10$is added between all bits of distinct residues to discourage partial overlaps.

Hydrophobic Contact Reward

Hydrophobic residues prefer to be spatially close. To model this, pairs of hydrophobic residues receive a reward when their positions are near each other:

$$E_{\text{contact}}(i,j) = -A\,(p_i - p_j)^2$$

This term mirrors the structure of the adjacency penalty but contributes negatively to the energy, thus favoring compact hydrophobic packing in the folded configuration.

Positional Bias

Finally, a small linear bias is applied to break symmetry and prevent the trivial all-zero configuration:

$$E_{\text{bias}}(r) = -D\sum_{k} x_{r,k}$$

This ensures that residues occupy distinct, non-degenerate positions on the lattice.

Mapping to Pauli Words

To solve the QUBO on a quantum computer, the binary variables $b_i$are mapped to qubit operators using the standard substitution:

$$b_i \mapsto \frac{1 - Z_i}{2}$$

where $Z_i$is the Pauli-Z operator acting on qubit i. Under this transformation, each QUBO term is converted as follows:

$$a_i b_i \mapsto \frac{a_i}{2} I - \frac{a_i}{2} Z_i$$
$$Q_{ij} b_i b_j \mapsto \frac{Q_{ij}}{4}(I - Z_i - Z_j + Z_i Z_j)$$

The resulting Hamiltonian is a weighted sum of Pauli words forming a SparsePauliOp, which directly represents the folding energy landscape in operator form.

Variational Quantum Eigensolver

The Variational Quantum Eigensolver (VQE) is used to approximate the ground-state energy of the Hamiltonian derived from the QUBO formulation. In EigenFold, the QUBO–Pauli Hamiltonian represents the discrete folding energy landscape, and the lowest eigenvalue corresponds to the most stable protein conformation.

The VQE framework operates by preparing a *parameterized quantum state* $|\psi(\theta)\rangle$through a chosen ansatz circuit, evaluating the expectation value

$$E(\theta) = \langle \psi(\theta) \mid H \mid \psi(\theta) \rangle$$

and iteratively updating the parameters θusing a classical optimizer until convergence to the minimal energy configuration.

Ansatz and Circuit Construction

EigenFold employs a TwoLocal ansatz with alternating single-qubit rotations and entangling layers:

$$TwoLocal(n_{qubits}, [R_y, R_z], CZ, reps = 2)$$

Each layer applies parameterized R_y and R_z rotations to all qubits, followed by controlled-Z entangling gates connecting neighboring qubits. This architecture balances expressivity and shallow circuit depth, making it well-suited to noisy intermediate-scale quantum (NISQ) hardware.

Optimization and Backend

The COBYLA (Constrained Optimization BY Linear Approximations) algorithm is used as the classical optimizer due to its robustness for smooth, low-dimensional variational landscapes. Expectation values are evaluated through Qiskit's Aer EstimatorV2 primitive, providing fast, simulator-based statevector sampling.

Each iteration of the VQE loop computes:

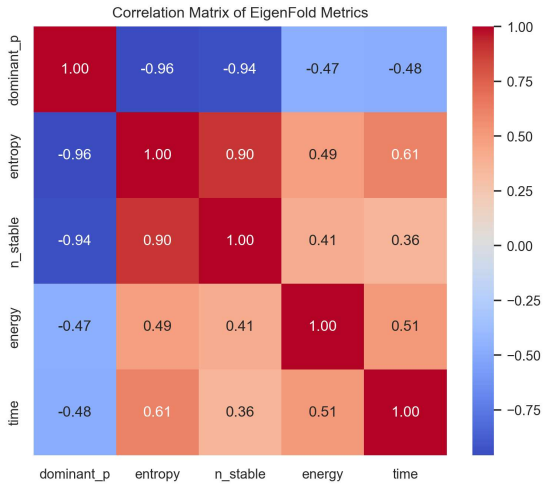
1. Prepare $|\psi(\theta)\rangle$using the parameterized circuit.
2. Measure the expectation value of the Hamiltonian H.
3. Update parameters $\theta$to minimize $E(\theta)$.

Convergence is achieved when successive energy updates fall below a numerical threshold. The final parameterized circuit represents the *quantum ground state*, whose probability amplitudes correspond to the relative likelihood of stable folded conformations.

Results and Analysis

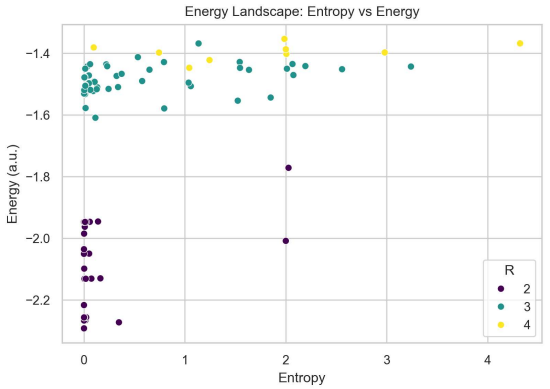
EigenFold was evaluated on three lattice geometries, face-centered cubic (fcc), body-centered cubic (bcc), and simple cubic (sc) — using a benchmark set of 35 peptide chains (16 dipeptides, 16 tripeptides, and 3 tetrapeptides). The QUBO formulation employed parameters $A = 5.0$, $B = 50.0$, $C = 5.0$, and $D = 1.0$ to balance backbone connectivity, collision avoidance, and contact energy. The resulting Hamiltonians were solved using Qiskit Aer's EstimatorV2 backend with a VQE implementation based on a $TwoLocal(ry, rz, cz)$ ansatz and the COBYLA optimizer. Each system used binary lattice encoding with $n_q = R \times \lceil \log_2 S \rceil$ qubits, ensuring qubit-efficient representation across all lattice types.

Correlation Analysis



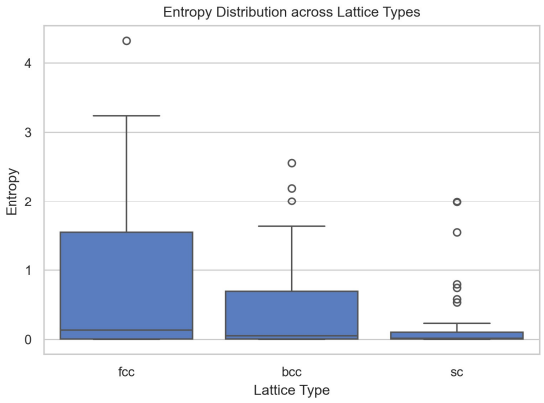
The correlation matrix reveals that the dominant conformation probability is strongly anti-correlated with both entropy and the number of stable states, indicating that as configurational diversity increases, probability distributes more evenly among multiple folds. Entropy and stability are positively correlated, confirming that multiple low-energy conformations contribute directly to higher configurational entropy. Moderate correlations between energy, entropy, and runtime suggest that more frustrated landscapes—those with higher entropy—stabilize at slightly higher energies and require longer optimization times. Overall, the results capture realistic folding trade-offs between dominance and degeneracy, with solver complexity driven by landscape diversity rather than system size.

Energy Landscape

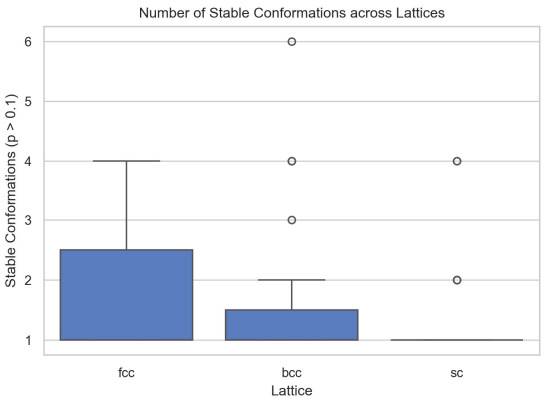


The energy–entropy plot shows two clear regimes: short peptides ($R = 2$) form deep, low-entropy minima, while longer chains ($R = 3$–4) exhibit higher entropy and shallower wells. This reflects the transition from simple, deterministic folding to frustrated landscapes with multiple accessible conformations. As residue count increases, the folding landscape flattens—mirroring realistic protein-like behavior where longer sequences exhibit competing low-energy configurations.

Lattice Expressivity

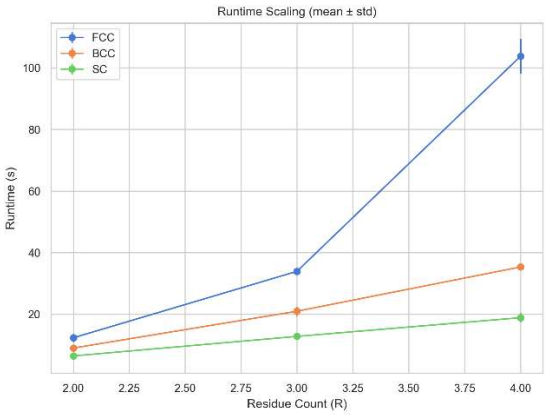


Entropy varies strongly with lattice geometry, following the order fcc > bcc > sc. The fcc lattice, having the highest coordination number, allows more spatial freedom and produces diverse conformations with high entropy. In contrast, the simple cubic lattice imposes rigid geometric constraints, yielding near-deterministic folding. This confirms that lattice connectivity directly governs configurational diversity in the model.



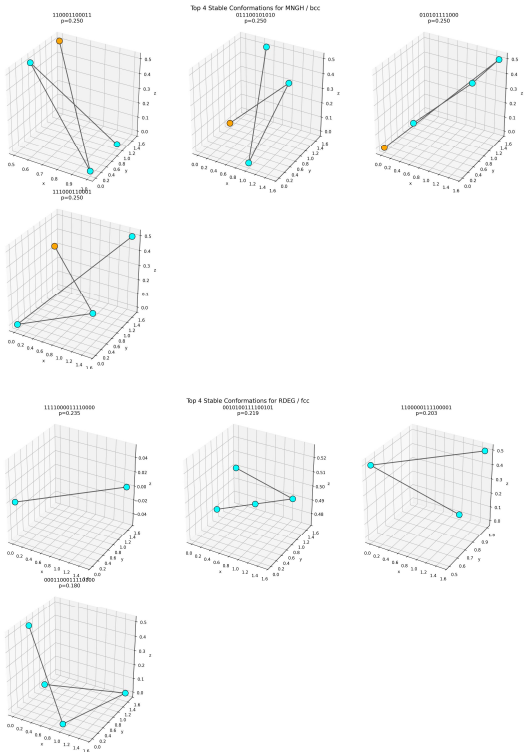
The number of stable conformations ($p > 0.1$) rises with lattice complexity, averaging around 2–4 for fcc, 1–2 for bcc, and nearly always a single state for sc. This trend mirrors the entropy distribution and supports the conclusion that topological richness promotes structural degeneracy. In simpler lattices, folding pathways converge to a unique low-energy configuration, whereas in dense geometries, multiple conformations coexist near the global minimum.

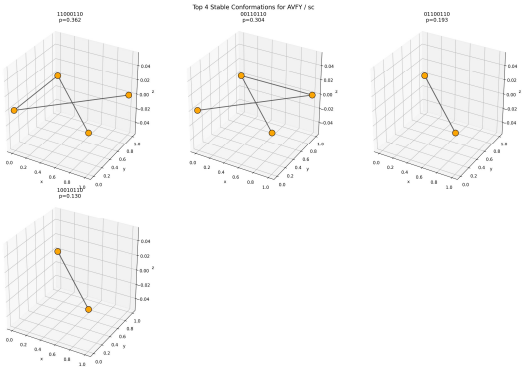
Runtime Scaling



Runtime increases with peptide length R, roughly linearly for sc and bcc lattices but superlinearly for fcc. The fcc lattice exhibits both the highest mean runtime and the largest variance, consistent with its denser adjacency graph and more complex QUBO coupling structure. This indicates that computational cost scales not only with system size but also with interaction density—an important consideration for scaling quantum simulations.

Selected Structures





We can also observe that degenerate structures do exist, suggesting that the QUBO weights can be tweaked further to emphasise constraints.

Conclusion

In summary, EigenFold successfully reproduces realistic folding behavior within a quantum-inspired framework. The model captures the trade-off between energetic stability and configurational entropy, showing that longer peptides and denser lattices yield more frustrated, degenerate energy landscapes. Runtime scaling aligns with interaction complexity, and lattice topology clearly governs folding diversity. Overall, the results validate the QUBO formulation and binary encoding as physically consistent and computationally efficient foundations for quantum protein-folding simulations.

Repository & Data

You can find all the source code at https://github.com/arhamaneeq/qc-EigenFold.git

Bibliography

- *Variational Quantum Eigensolver for Protein Folding using Neutral Atom Platforms*, Gefen Barnes, 2024
- *A perspective on protein structure prediction using quantum computers*, Doga et al, 2023
- *Resource-efficient quantum algorithm for protein folding*, Robert et al., 2021
- *Estimation of Effective Interresidue Contact Energies from Protein Crystal Structures: Quasi-Chemical Approximation*, Miyazawa & Jernigan, 1985