Latent Space Optimization for the 2D Ising Model

Name: Amanda Tartarotti Cardozo da Silva
Course: Optimization Techniques
Master Program: Master in Computacional Engineering and Smart Systems
Academic Year: 2025/2026

University of Basque Country (EHU/UPV)

Abstract. This report investigates the application of Latent Space Optimization to find the ground state of a 2D Ising Model. By training a Variational Autoencoder to compress grid configurations and a Surrogate Model to approximate the energy landscape, optimization is performed in a continuous low-dimensional space. This project compare Gradient-based methods against CMA-ES, demonstrating that LSO can successfully identify the global minimum energy configuration.

1 Problem Definition

The **Ising Model** is a fundamental mathematical model of ferromagnetism in statistical mechanics. It consists of discrete spins σ_i arranged on a lattice, where each spin can be in one of two states: $+1$ (up) or -1 (down).[1]

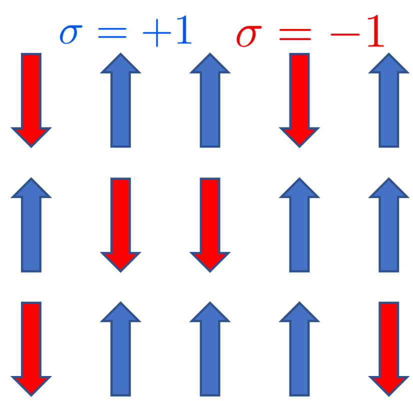

Fig. 1: Schematic representation of a configuration of the 2D Ising Model on a square lattice

1.1 Physical Context

In a physical system in thermal equilibrium with a heat bath at temperature T, the probability $P(\mu)$ of finding the system in a specific configuration μ with energy E_μ is given by the Boltzmann Distribution:

$$P(\mu) = \frac{1}{Z} e^{-\beta E_\mu} \tag{1}$$

Where $\beta = \frac{1}{k_B T}$ is the inverse temperature and $Z = \sum_\nu e^{-\beta E_\nu}$ is the partition function.

Standard numerical methods for simulating this system rely on satisfying the detailed balance condition [2] to ensure equilibrium is reached:

$$P(\mu)P(\mu \to \nu) = P(\nu)P(\nu \to \mu) \tag{2}$$

Where $P(\mu \to \nu)$ is the transition probability between states.

1.2 Optimization Objective

While standard simulations aim to sample states according to these probabilities, the objective in this project is distinct. The model aims to find the Ground State, which corresponds to the limit as $T \to 0$. In this limit, the probability mass concentrates entirely on the configuration with the minimum total energy.

Thus, the problem is framed not as a sampling task, but as a combinatorial optimization task to minimize the Hamiltonian:

$$E(\mathbf{x}) = -\sum_{\langle i,j \rangle} J_{ij} \sigma_i \sigma_j \tag{3}$$

Where $\langle i, j \rangle$ represents the summation over nearest neighbors and $J = 1$ represents ferromagnetic coupling. For a 20×20 grid, finding the minimum of this function involves searching a discrete space of size 2^{400}.

2 Steps for Solving the Problem

To avoid combinatorial optimization in the high-dimensional discrete space, a Latent Space Optimization (LSO) approach [3] was adopted. The methodology consists of four key steps:

2.1 Data Generation

The project starts by generating a training dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$ of Ising grids and their energies. A biased sampling strategy was emplyed, with probability thresholds set at $p = 0.65$ (bias towards -1) and $p = 0.35$ (bias towards $+1$).

A purely random initialization ($p = 0.5$) produces high-entropy noise with no distinct structure. By selecting these specific moderate biases, the dataset was populated with partially ordered configurations. These samples effectively cover the basins of attraction for both positive and negative magnetization, providing the necessary gradient information for the latent space to capture the transition from disorder to order.

2.2 Dimensionality Reduction

To enable continuous optimization, the discrete configuration space was mapped to a low-dimensional manifold using a Variational Autoencoder (VAE). The VAE consists of two probabilistic networks:

- **Encoder** ($q_\phi(\mathbf{z}|\mathbf{x})$): Compresses the high-dimensional grid $\mathbf{x}$ into a latent vector $\mathbf{z}$.
- **Decoder** ($p_\theta(\mathbf{x}|\mathbf{z})$): Reconstructs the grid configuration $\hat{\mathbf{x}}$ from the latent vector $\mathbf{z}$.

The model architecture consists of fully connected layers. The input 20×20 grid is flattened into a vector of size $N = 400$ and the implementation selects a latent dimension of $d = 8$, representing a 50x compression ratio. The decoder utilizes a Tanh activation function in the final layer: since Ising spins are binary $\{-1, 1\}$, the Tanh function provides a natural continuous relaxation onto the interval $[-1, 1]$, allowing for differentiable operations.

The network is trained by minimizing the Evidence Lower Bound (ELBO) loss function:

$$\mathcal{L} = \mathbb{E}_q[\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) \tag{4}$$

The first term minimizes the reconstruction error, while the second term - Kullback-Leibler divergence - regularizes the latent space towards a standard Gaussian prior, ensuring the space is easy to navigate for the optimizer.

2.3 Surrogate Modeling

Direct optimization in the latent space requires evaluating the energy of generated samples. However, repeatedly decoding points to evaluate their physical energy is computationally expensive inside an optimization loop.

To address this challenge, the model train a Surrogate Model $h : \mathbb{R}^d \to \mathbb{R}$. This model is a neural network designed to approximate the composite function of decoding and energy calculation:

$$h(\mathbf{z}) \approx E(\text{decode}(\mathbf{z})) \tag{5}$$

Implementation Details:

- **Architecture:** The surrogate consists of a feed-forward network with two hidden layers (128 units each) and ReLU activations.
- **Training:** Construct a dataset of latent vectors $\mathbf{z}_{train}$ by encoding the original training data. The corresponding energy values are standardized to ensure stable convergence.
- **Objective:** The network is trained to minimize the Mean Squared Error (MSE) between the predicted latent energy and the true physical energy.

The Surrogate's function $h(\mathbf{z})$ is fully differentiable, what allows us to compute the gradient with respect to the latent variables, and enables the use of efficient gradient-based algorithms.

2.4 Optimization Strategy

The objective of the optimization strategy is to find the latent vector $\mathbf{z}^*$ that minimizes the predicted energy:

$$\mathbf{z}^* = \operatorname{argmin} h(\mathbf{z}) \tag{6}$$

To ensure robustness and mitigate the risk of getting trapped in local minima, which is a common challenge in non-convex landscapes, a Multi-Start approach was employed using two complementary algorithms:

Gradient Descent (Adam) Using the differentiability of the neural network, the model computes the surrogate gradient with respect to latent variables $\nabla_{\mathbf{z}} h(\mathbf{z})$, through backpropagation [4]. In this process, iteratively update $\mathbf{z}$ to descend the energy surface. This method is highly efficient for local fine-tuning.

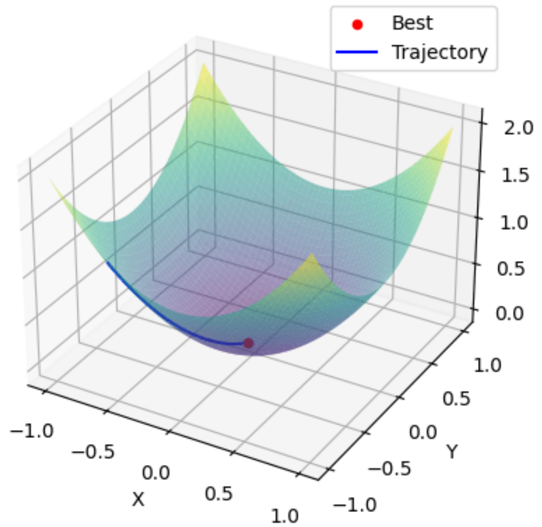

Fig. 2: Global Minima Using Adam Optimization Algorithm

The above image illustrates optimization trajectory of the candidate points in the Gradient Descent method [5]. The best solution found is shown as a red dot, and the trajectory is shown as a blue line connecting the candidate points.

Covariance Matrix Adaptation Evolution Strategy A population-based evolutionary algorithm [6]. It iteratively samples a distribution of candidate vectors and updates its covariance matrix to favor promising regions. This method is particularly effective for global exploration and escape from shallow local optima.

3 Experiments Proposed

Following the already described steps to solve the problem, the project experiments began by constructing a training dataset comprising 2,000 Ising grid samples. To ensure the model learned relevant physical structures rather than pure noise, it employed a biased sampling strategy with probability thresholds of $p = 0.35$ and $p = 0.65$. This approach ensures that the dataset cover regions of high quality or promising configurations.

Regarding the neural architectures, the VAE was implemented with symmetric Encoder and Decoder networks, each consisting of two fully connected hidden layers with 256 units. The high-dimensional input ($N = 400$) was compressed into a compact latent dimensionality of $d = 8$. The

Surrogate Model, responsible for approximating the energy landscape, was implemented as a Multi-Layer Perceptron with two hidden layers of 128 units. This network was trained to minimize the MSE between its predictions and the normalized true energy values.

Sequentially, was performed a comparative analysis of two distinct optimization algorithms within the learned latent space: Gradient Descent and CMA-ES. The success of these methods was evaluated and described in the next section of this report.

4 Results of the Experiments

4.1 Surrogate Performance

The surrogate model was trained for 22 epochs, a duration determined empirically to prevent overfitting. The validation MSE reached a minimum of ≈ 0.67, confirming that the model effectively captured the underlying topology and gradient directions of the energy landscape.

Figure 3 illustrates the surrogate's accuracy. The scatter plot demonstrates a strong correlation between the predicted values and the true physical energy, with the data points clustering closely around the ideal line represented by the red dashed diagonal.

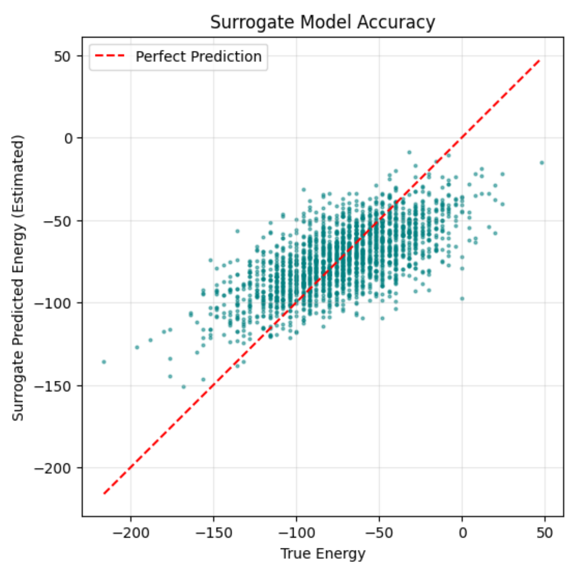

Fig. 3: Surrogate Model Accuracy: Comparison between True Energy (calculated via Hamiltonian) and Predicted Energy (estimated by the Neural Network). The red dashed line represents perfect prediction ($y = x$).

4.2 Optimization Results

10 restarts were performed for each optimization method. The results were as follows:

- **Gradient Descent:** Consistently converged to the ground state.
- **CMA-ES:** Also found the ground state, proving robustness.

In terms of computational efficiency, the optimization process demonstrated rapid convergence. For the Gradient Descent method, we allocated a fixed budget of 200 steps per restart. As evidenced by the trajectory analysis (Figure 4), the algorithm consistently reached the ground state energy ($E = -800$) in between this time limit. This indicates that the latent space topology constructed by the VAE provides a direct and smooth path to the solution, requiring relatively few updates to traverse from a random initialization to the optimal region.

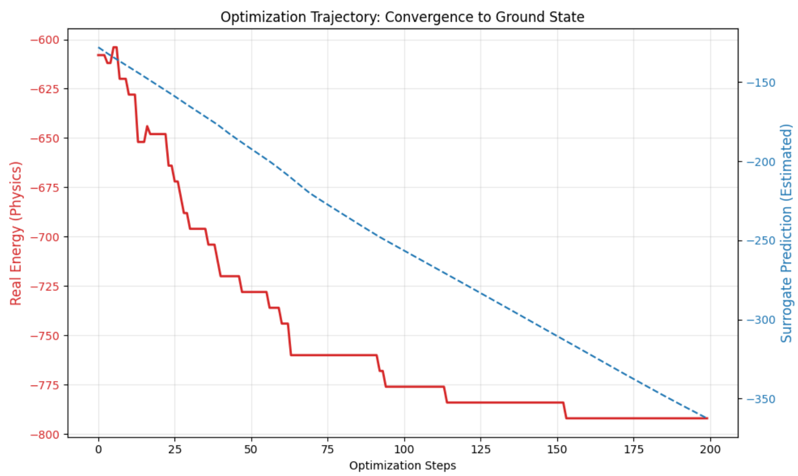

Fig. 4: Optimization Trajectory. The Red line shows the Real Energy converging to -800. The Blue dashed line shows the Surrogate prediction

To verify the physical consistency of the optimization pathway, Figure 5 visualizes the grid configurations obtained by linearly interpolating between a random starting point and the optimal solution, it demonstrates the continuity of the learned latent space.

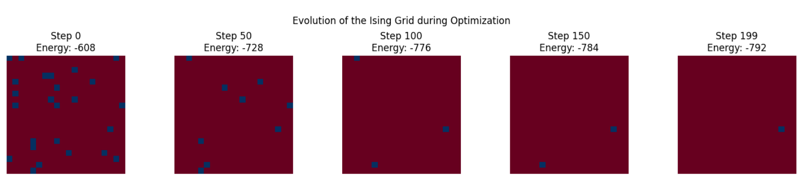

Fig. 5: Physical Evolution of the Ising Grid

5 Conclusions

This project demonstrated the efficacy of Latent Space Optimization for solving high-dimensional combinatorial problems like the Ising Model. By mapping discrete spin configurations to a continuous latent manifold, it was possible to successfully transform a discrete search problem into a differentiable optimization task.

The VAE successfully learned a smooth representation of the discrete configuration space and this well-structured manifold allowed the Surrogate model to effectively capture the global topology of the energy landscape. Consequently, both optimization algoritms implemented - Gradient Descent and CMA-ES - demonstrated robustness, successfully locating the theoretical global minimum ($E = -800.0$).

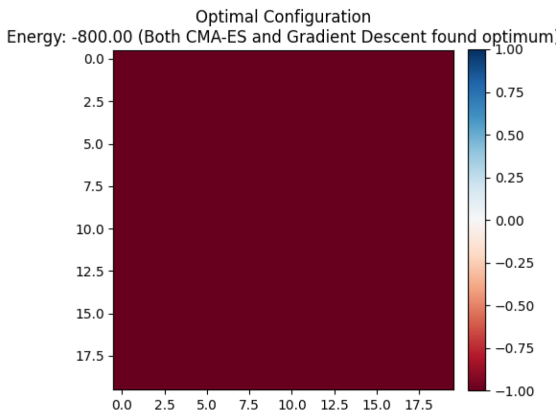

Fig. 6: Optimal Configuration found ($\mathbf{x}^*$). The grid shows a perfect ferromagnetic domain

The visual representation of the decoded solution (Figure 6) confirms the physical validity of the result. The grid exhibits a perfect ferromagnetic domain, verifying that the mathematical minimum found in the latent space corresponds to the true physical ground state.

References

1. Newell, G. F., & Montroll, E. W. (1953). On the theory of the Ising model of ferromagnetism. *Reviews of Modern Physics*, 25(2), 353.
2. [Mr. P Solver]. (2021). *The Ising Model in Python: Statistical Mechanics and Permanent Magnets.* [Video]. Available at: `https://www.youtube.com/watch?v=K--1hlv9yv0&t=891s`. (Accessed: December 2025).
3. Tripp, A., Daxberger, E., & Hernández-Lobato, J. M. (2020). Sample-efficient optimization in the latent space of deep generative models via weighted retraining. *Advances in Neural Information Processing Systems*, 33, 11259–11272.
4. Thiyaneshwaran, G. (2022). *Types of Optimizers in Deep Learning From Gradient Descent to Adam.* Medium. Available at: `https://medium.com/@ThiyaneshwaranG/types-of-optimizers-in-deep-learning-from-gradient-descent-to-adam-1572e657c8c5`. (Accessed: December 2025).

5. GeeksforGeeks. (n.d.). *How to implement Adam Gradient Descent from scratch using Python*. Available at: `https://www.geeksforgeeks.org/how-to-implement-adam-gradient-descent-from-scratch-using-python/`. (Accessed: December 2025).
6. Nikolaus Hansen, Youhei Akimoto, and Petr Baudis. Cma-es/pycma on github. Zenodo, DOI:10.5281/zenodo.2559634, 2019.