Report: 3D N × N × N queens problem project

Xinxian Ma, Jinghao Zheng, Yixiao Zhang

In this project, we adopt a unified modeling approach: the energy of a configuration is defined as the number of pairs of queens that attack each other. Based on this common energy function, we design three different state representations and their corresponding proposal moves, ranging from almost unconstrained 3D coordinates to more structured height-matrix and row-wise permutation representations. For each representation, we construct a Metropolis–Hastings Markov chain * and combine it with simulated annealing schedules so that the sampling process gradually focuses on low-energy configurations. Finally, we compare the three methods and different annealing strategies for several values of N, and study their convergence behavior, run time, and the minimal energies that can be reached.

1. PROBLEM FORMULATION

1.1 The 3D (N^2)-Queens problem

We study a three-dimensional version of the N-Queens problem. The board is an $N \times N \times N$ grid and we must place N^2 queens on distinct cells.

Let two queens be at positions (i_1, j_1, k_1) and (i_2, j_2, k_2), and define

$$d_i = i_1 - i_2, \quad d_j = j_1 - j_2, \quad d_k = k_1 - k_2.$$

If $(d_i, d_j, d_k) = (0, 0, 0)$, two queens occupy the same cell, which is not allowed. Meanwhile, the two queens will attack each other if and only if one of the following holds:

- **Axis-aligned:** exactly one of d_i, d_j, d_k is non-zero (the queens lie on a line parallel to a coordinate axis).
- **Planar diagonal:** the queens lie in the same coordinate plane and form a diagonal:

$$d_k = 0,\ |d_i| = |d_j| \neq 0 \quad \text{or} \quad d_j = 0,\ |d_i| = |d_k| \neq 0 \quad \text{or} \quad d_i = 0,\ |d_j| = |d_k| \neq 0.$$

- **Space diagonal:** all three coordinate differences have the same non-zero absolute value,

$$|d_i| = |d_j| = |d_k| \neq 0.$$

Our goal is to find one or more 3D configurations of N^2 queens such that no pair of queens satisfies any of these attack conditions as solution.

1.2 State space (high-level)

Abstractly, a state s is a configuration of N^2 queens placed on distinct cells of the $N \times N \times N$ grid. We can view s as an array of N^2 coordinates (i, j, k) with no duplicates.

The set of all such configurations defines the (very large) state space on which we build our Markov chain. In practice, we will not explore this full space directly. In Section 2 we will introduce three concrete state representations, each restricting this abstract space in a different way and inducing different proposal moves.

1.3 Energy function and objective

We associate an energy $E(s)$ to every state s, defined as the number of attacking pairs of queens in the configuration. Each unordered pairs that satisfies one of the attack conditions above contributes 1 to $E(s)$.

By construction, $E(s) = 0$ if and only if there are no attacking pairs, so zero-energy states are exactly valid solutions of the 3D (N^2)-Queens problem. Our objective is to find states with minimal energy, ideally $E(s) = 0$. In the following sections we use Markov Chain Monte Carlo and simulated annealing to search this state space and to compare different state representations and parameter choices.

*Hastings, W.K. (1970). Monte Carlo sampling methods using Markov chains and their applications. Biometrika, 57:97-109.

2. ALGORITHM

In this section we describe the Markov Chain Monte Carlo (MCMC) scheme used to minimize the energy function $E(s)$ defined above. All three methods share the same target distribution, acceptance rule and annealing schedules. They differ only in how a state is represented and how proposal moves are generated. We organize the description from the least constrained state space (Method A) to the most constrained one (Method C).

2.1 Common MCMC framework

Given the energy function $E(s)$, we sample configurations using a Metropolis–Hastings Markov chain at inverse temperature $\beta > 0$ with target distribution

$$\pi_\beta(s) \propto \exp(-\beta E(s)).$$

At each iteration t, given the current state s_t, we:

1. sample a proposed state s' from a symmetric proposal distribution $q(\cdot \mid s_t)$;
2. compute the energy difference $\Delta E = E(s') - E(s_t)$;
3. accept s' with probability

$$\alpha(s_t \to s') = \min\{1, \exp(-\beta_t \Delta E)\},$$

where β_t is the current inverse temperature; if the move is rejected, we keep $s_{t+1} = s_t$.

Because the proposal is symmetric, this acceptance rule ensures reversibility with respect to π_{β_t}. In practice, the main design choice is therefore the proposal distribution $q(\cdot \mid s)$, which determines how local each move is and how fast the chain can explore the state space.

Simulated annealing schedules

To escape local minima, we combine Metropolis sampling with simulated annealing. We start at a high temperature $1/\beta_0$ and gradually cool down to $1/\beta_{\text{end}}$, so the chain first explores widely and later focuses on low-energy states.

We experiment with several simple schedules for β_t:

- **Fixed:** $\beta_t = \beta_0$ for all t.
- **Exponential:** $\beta_t = \beta_0 \cdot c^t$ for a constant $c > 1$.
- **Geometric:**

$$\beta_t = \beta_0 \left(\frac{\beta_{\text{end}}}{\beta_0} \right)^{t/T_{\max}},$$

where $T_{\max}$ is the total number of iterations.

- **Logarithmic:**

$$\beta_t = \beta_0 \cdot \frac{\log(1+t)}{\log(1+T_{\max})}.$$

2.2 Method A: Free 3D coordinates

Method A uses the least constrained state space. A state is represented as an array of shape $(N^2, 3)$, where each row stores the coordinates (i, j, k) of one queen. All coordinates are distinct, so no two queens occupy the same cell, but there are no further structural constraints.

Proposal move. At each iteration, we:

1. Pick a queen index q uniformly at random from $\{1, \ldots, N^2\}$.
2. Sample a new cell (i', j', k') uniformly from all unoccupied cells of the board.
3. Propose a move that relocates queen q from (i, j, k) to (i', j', k').

Only one queen moves, so we can compute the energy difference ΔE by checking the conflicts involving that queen before and after the move, instead of recomputing $E(s)$ from scratch. This keeps the cost of each iteration linear in the number of queens that interact with the moved queen.

This representation explores the full configuration space of N^2 queens on N^3 cells. It is very flexible and, in principle, can reach any configuration that respects the "at most one queen per cell" constraint. However, the state space is extremely large and contains many configurations with trivial conflicts. The chain can thus require many iterations to move from a high-energy region to a low-energy one, and the runtime tends to be longer than for more constrained representations.

2.3 Method B: Column-height matrix

Method B restricts the state space by enforcing exactly one queen in each vertical column (i, j). A state is represented as an $N \times N$ matrix H of integers, where

$$H[i, j] = k \quad \Longleftrightarrow \quad \text{there is a queen at } (i, j, k).$$

This guarantees that there are exactly N^2 queens on the board, one in each column (i, j), and eliminates configurations with empty or doubly occupied columns.

Proposal move. At each iteration, we:

1. Pick a base cell (i, j) uniformly at random.
2. Let the current height be $k = H[i, j]$. Propose a new height k' sampled uniformly from $\{0, \ldots, N-1\} \setminus \{k\}$.
3. Update the queen position from (i, j, k) to (i, j, k') in the proposed state.

Again, only one queen moves, so we compute ΔE by looking at the conflicts of that queen before and after the height change. The per-iteration cost is similar to Method A, but the number of possible states is significantly smaller because the number of queens per column is fixed.

By removing the freedom to leave columns empty or place multiple queens in the same column, Method B reduces the effective size of the state space and avoids many obviously bad configurations. This typically leads to faster convergence and better runtime compared to Method A, especially for larger values of N. On the other hand, the chain can no longer explore configurations that violate the one-queen-per-column constraint, so some parts of the full configuration space become unreachable.

2.4 Method C: Row-wise permutation heights

Method C adds even more structure. The state is still an $N \times N$ height matrix H, but each row $H[i, :]$ is constrained to be a permutation of $\{0, 1, \ldots, N-1\}$. For each fixed row index i, all possible heights appear exactly once along that row. This implies that each row is "spread" across all height levels.

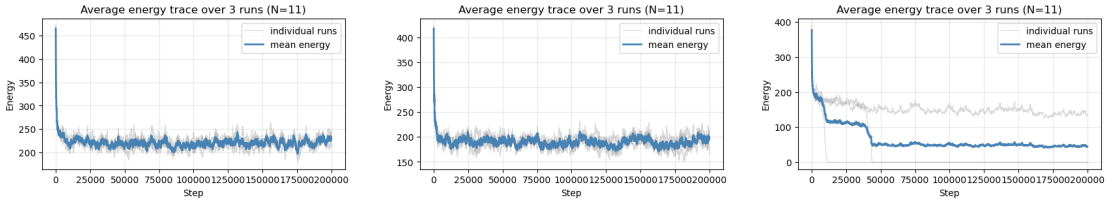
Proposal move. At each iteration, we:

1. Pick a row index i uniformly at random.
2. Pick two distinct column indices $j_1 \neq j_2$ uniformly at random.
3. Swap the two heights in that row:

$$H[i, j_1],\, H[i, j_2] = H[i, j_2],\, H[i, j_1].$$

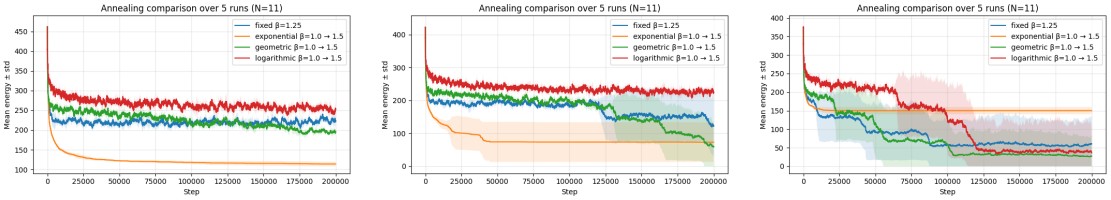
This swap move exchanges the heights of the two queens in row i, and therefore moves two queens in 3D space. We compute the energy difference ΔE by checking conflicts involving these two queens before and after the swap.

The permutation constraint further shrinks the state space and enforces a more regular structure on the configuration. The initial energies tend to be lower than in Methods A and B, and each proposal changes the configuration in a local and controlled way. This often leads to faster convergence. However, the chain is restricted to a much smaller subset of all possible configurations. In particular, some low-energy configurations that do not satisfy the row-wise permutation constraint may never be visited.



(a) Method A (b) Method B (c) Method C

Figure 1: Energy evolution for $N = 11$ for the three state representations at fixed inverse temperature $\beta = 1.25$.



(a) Method A (b) Method B (c) Method C

Figure 2: different annealing schedules for $N = 11$

3. EXPERIMENTS

3.1 Experimental setup

We mainly study the case $N = 11$ to analyse the energy evolution and the effect of simulated annealing, and then look at how the minimal energy scales with N.

Unless stated otherwise, we use `max_steps` $= 200000$ iterations. For the energy traces at fixed N we average over $R = 3$ runs per method, and for the annealing comparison we use $R = 5$ runs. Without annealing we fix the inverse temperature to $\beta = 1.25$. With annealing we use a geometric schedule that increases the inverse temperature from $\beta_0 = 1.0$ to $\beta_{\text{end}} = 1.5$.

For the scaling experiment in Section 3.3, we consider board sizes $N \in \{3, \ldots, 13\}$. For each N we run $R = 5$ chains with Method C and the geometric schedule, and record the minimal energy reached by each run.

3.2 Energy evolution for a fixed N

We first study how the energy evolves over time for a fixed board size. Figure 1 shows the average energy E_t for $N = 11$ and the three state representations, with each curve averaged over $R = 3$ runs.

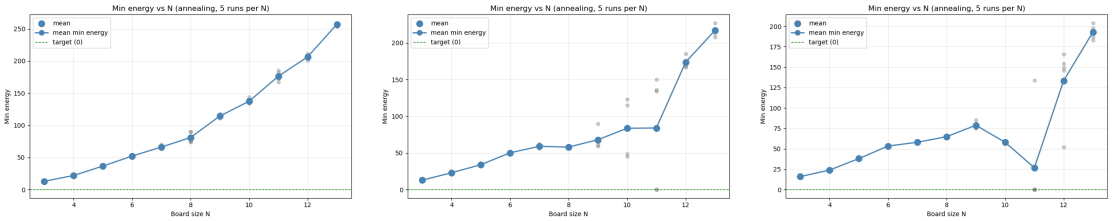
For all three methods the energy drops quickly at the beginning and then flattens. Methods A and B stabilise at a non-zero level, with Method B reaching slightly lower energies. Method C reaches much lower energies, and two of the three runs even reach zero, showing that this more structured representation has a higher chance of finding a valid solution within the same number of iterations.

As shown in the Figure 1, at a fixed temperature, all methods reduce the number of conflicts, but more structured state representations (in particular Method C) converge faster to low-energy configurations.

Then, we still fix $N = 11$ and compare four different temperature schedules. All runs share the same max steps, base seed and cooling rate. Figure 2 shows, at each step, the mean energy over the 5 runs. The shaded region indicates the mean $\pm$ one standard deviation across runs. The staircase-like decreases in the mean curve arise from a small number of runs that abruptly converge to zero energy, causing sudden downward shifts in the average.

Compared to the fixed schedule, exponential annealing is aggressive: it decreases energy quickly but is unstable. It rarely reaches zero energy and performs poorly for Method C due to early freezing.

Logarithmic annealing is conservative: stable but generally weak, often giving higher final energies than geometric and sometimes close to the fixed schedule.



(a) Method A (b) Method B (c) Method C

Figure 3: Minimal energy reached as a function of the board size N for the three methods.

The geometric schedule is never the worst and often the best. It behaves well for all three methods. For Method B it reaches zero energy in 3 out of 5 runs. For Method C it reaches zero energy in 4 out of 5 runs. These are the best results among all schedules.

In summary, simulated annealing is useful, but it needs a good schedule. A random choice of annealing schedule may not improve much over a fixed temperature. With a good schedule, such as the geometric one, we can reach lower energies in fewer steps than with the fixed schedule. Therefore, in the following experiments we use geometric annealing as our default annealing scheme.

3.3 Minimal energy as a function of N

We finally analyze how the minimal energy depends on the board size N. In this experiment, we focus on Method C and reuse the same geometric annealing schedule. For each $N \in \{3, \ldots, 13\}$, we run $R = 5$ chains and record the minimal energy reached by each run.

Figure 3 summarizes these results: gray points represent the minimal energy of each run, and the curve is their average. The average minimal energy tends to increase with N, but some sizes are noticeably easier than their neighbours. In particular, $N = 11$ stands out with much lower minimal energies, and several runs reach zero energy. This suggests that the difficulty of the 3D (N^2)-Queens problem is not a simple monotone function of N; for some values of N (such as $N = 11$), our MCMC algorithm finds very good configurations much more easily than for nearby sizes.

We found a detailed study on the N-Queens problem[†] that analyzes the 3D setting and provides an explanation for this result. It states the following: if $\gcd(n, 210) = 1$, then n^2 queens can be placed on an $n \times n \times n$ cube so that no two attack each other (a Latin square of order n exists). We can explain why the energy of 11 is smaller than other values because it can find a solution with energy 0. Theoretically, we should also find a smaller energy on 13 (but that may require different settings, e.g., more iterations or different hyper parameters).

4. CONCLUSION

We studied the 3D (N^2)-Queens problem using a unified MCMC framework with a common energy function and several state representations. Our experiments show that the choice of representation strongly affects the efficiency of the algorithm: the more structured the representation, the faster the chain converges and the lower the energies it can reach. In particular, the row-permutation model (Method C) consistently achieved the best performance and was often able to find zero-energy configurations for $N = 11$.

Simulated annealing further improves all three methods, with geometric schedules giving the most reliable results. Finally, the minimal energy does not scale monotonically with N, suggesting that some board sizes are intrinsically easier than others under our assumptions.

To draw the conclusion, our results illustrate the value of structured state spaces and annealing schedules when applying MCMC to high-dimensional combinatorial optimization problems.

[†]N-Queens URSS workbook, University of Warwick, available at `https://warwick.ac.uk/fac/sci/maths/general/outreach/urss/n-queens-urss-workbook_16.pdf`.