*Transactions of the Korean Nuclear Society Spring Meeting*
*Jeju, Korea, May 22-23, 2025*

# Quantum Hybrid Reinforcement Learning for Optimal Variable Ordering in Binary Decision Diagrams

Young Ho Chae [a*], Seo Ryong Koo [a]
[a] *Korea Atomic Energy Research Institute, 111 Daedeok-daero 989 beon-gil, Daejeon, 34057*
[*] *Corresponding author: yhchae@kaeri.re.kr*

***Keywords :*** Quantum-Inspired Optimization, Reinforcement Learning, Variable Ordering, Binary Decision Diagram

## 1. Introduction

In the nuclear power plant domain, the safety of highly complex systems is crucial. To ensure operational reliability, probabilistic safety assessment (PSA) is widely employed to quantify and manage the risks associated with potential faults. Given the intricate interdependencies in such systems, traditional deterministic methods often fall short, making probabilistic approaches essential for accurate risk evaluation.

A particularly powerful technique for fault analysis in complex systems is the Binary Decision Diagram (BDD) based method. BDDs provide a compact and canonical representation of Boolean functions, enabling efficient fault tree analysis and system reliability computations. They have been successfully applied in numerous safety-critical applications due to their ability to simplify and analyze the combinatorial structure of failure modes.

Despite the intuitive appeal of BDD-based methods, a significant challenge remains: the variable ordering problem. The size of a BDD, which directly impacts computational efficiency and memory usage, is highly sensitive to the order in which variables are arranged. Mathematically, if we denote the BDD size by $S$, and the variable ordering by $\pi$, then the objective is to minimize:

$$\min_{\pi} S(\pi)$$

However, finding the optimal ordering is known to be NP-hard, which has motivated the use of heuristic methods for practical applications [1].

To address this challenge, heuristic-based approaches have been developed to quickly approximate good variable orderings without exhaustive search. In this paper, we propose a novel method that leverages the strengths of both quantum-inspired techniques and reinforcement learning (RL). The Quantum Enhanced RL Hybrid Optimizer is designed to achieve two key objectives: - Fast Calculation Speed: By incorporating quantum-inspired tunneling strategies, the optimizer can efficiently escape local minima, mimicking quantum superposition and interference. - BDD Size Optimization: Reinforcement learning is employed to iteratively improve the variable ordering, ensuring that the resulting BDD remains as compact as possible.

The integration of these techniques offers a promising direction for overcoming the variable ordering problem in BDD-based fault analysis, potentially leading to more reliable and efficient safety assessments in nuclear power plants.

## 2. Quantum Hybrid RL Optimizer for Variable Ordering

The variable ordering problem is a central challenge when using Binary Decision Diagrams (BDDs) to represent Boolean functions. The size of a BDD, denoted as $S(\pi)$, is highly sensitive to the variable ordering $\pi$, and finding the optimal ordering is NP-hard:

$$\min_{\pi} S(\pi)$$

### *2.1 Quantum-Inspired Exploration and Tunneling*

Quantum mechanics provides powerful concepts that can be harnessed to navigate complex search spaces. In our optimizer, we adopt two key quantum-inspired ideas: superposition (with interference) and tunneling.

In quantum mechanics, a system exists in a superposition of states, allowing it to explore multiple configurations simultaneously. We mimic this behavior by generating a diverse set of candidate orderings from a base ordering $\pi$. For example, elementary operations such as the swap operation,

$$\pi' = \text{swap}(\pi, i, j),$$

and the reverse operation,

$$\pi' = \text{reverse}(\pi, i, l),$$

are applied in parallel to create many candidates. When these candidate orderings are combined in a manner analogous to quantum interference, the resulting ordering can incorporate the most promising features from each candidate. This approach is essential because the vast search space of variable orderings often leads classical methods to converge slowly or get trapped in local minima.

Complementing this, quantum tunneling enables a particle to traverse an energy barrier that it would not overcome by classical means. In our optimizer, tunneling strategies provide nonlocal moves that help the algorithm escape local optima. Examples include grouping variables based on parity,

$$\pi' = \text{group\_parity}(\pi),$$

or by modulo operations,

$$\pi' = \text{group\_mod}(\pi, m),$$

as well as performing rotations,

$$\pi' = \text{rotate}(\pi, k).$$

Each tunneling strategy $T_i$ is assigned a weight $w_i$, and the probability of selecting a particular strategy is given by

$$P(T_i) = \frac{w_i}{\sum_j w_j}.$$

This nonlocal search capability is crucial for bypassing regions where only incremental changes are insufficient to reduce the BDD size, thus accelerating convergence toward a smaller BDD.

### *2.2 Reinforcement Learning Framework*

To guide the search process more effectively, the optimizer incorporates reinforcement learning. Rather than considering the full variable ordering $\pi$—which is prohibitively large—we extract key features to form a compact state representation. For instance, the state $s$ may include the first and last elements of the ordering, $(\pi_1, \pi_n)$, as well as a measure of the parity grouping quality $q_p$, defined as

$$q_p = 1 - \frac{\text{span of even (or odd) indices}}{n-1},$$

along with other grouping metrics such as $q_{mod3}$, $q_{mod}$ , etc. In this way, the state is expressed as:

$$s = \left(\pi_1, \pi_n, q_p, q_{mod}\ , q_{mod4}, \dots\right).$$

The action space $\mathcal{A}$ includes elementary transformations that adjust the ordering, such as: - Swap: $a = \text{swap}(i, j)$, - Move: $a = \text{move}(i, j)$, - Reverse: $a = \text{reverse}(i, l)$, - and various pattern-based actions.

After taking an action, the optimizer evaluates the change in the BDD size to compute an immediate reward $r$, defined as:

$$r = \begin{cases} \lambda \dfrac{S(\pi_{\text{old}}) - S(\pi_{\text{new}})}{S(\pi_{\text{old}})} & \text{if } S(\pi_{\text{new}}) < S(\pi_{\text{old}}) \\ -\mu & \text{otherwise} \end{cases},$$

where $\lambda$ and $\mu$ are positive constants scaling the reward and penalty.

The RL agent maintains a Q-table that estimates the quality of each state–action pair. The Q-learning update is performed according to:

$$Q(s, a) \leftarrow Q(s, a) + \alpha\big(r - Q(s, a)\big),$$

with $\alpha$ being the learning rate. Actions are selected using an epsilon-greedy policy:

$$a^* = \begin{cases} \text{a random action} & \text{with probability } \epsilon, \\ \underset{a \in \mathcal{A}}{\text{argmax}}\, Q(s, a) & \text{with probability } 1 - \epsilon. \end{cases}$$

This framework enables the optimizer to adaptively favor actions that have historically led to significant reductions in BDD size, thereby improving search efficiency.

### *2.3 Integration of Quantum-Inspired and RL Methods*

The final optimizer interleaves quantum-inspired exploration with RL-based exploitation in an adaptive, multi-phase framework:

1. Exploration Phase: Candidate orderings are generated using quantum-inspired methods, effectively creating a superposition of solutions that enhances global search.
2. Exploitation Phase: The RL agent applies actions with high expected rewards to fine-tune the ordering locally.
3. Tunneling Phase: When local improvements stagnate, the optimizer employs tunneling strategies to perform nonlocal moves, allowing it to overcome local minima.
4. Pattern-Based Phase: The algorithm utilizes learned grouping patterns (such as parity or modulo clustering) to propose new orderings that align with structures known to reduce BDD size.

This integrated approach leverages the benefits of both global exploration (through quantum-inspired concepts) and local refinement (via reinforcement learning), resulting in a robust method for the variable ordering problem.

In summary, the Quantum Enhanced RL Hybrid Optimizer directly addresses the variable ordering problem by combining nonlocal, quantum-inspired search methods with an adaptive RL framework. The mathematical foundations—from the objective function

$$\min_{\pi} S(\pi)$$

to the Q-learning update rule,

$$Q(s, a) \leftarrow Q(s, a) + \alpha\big(r - Q(s, a)\big),$$

This hybrid approach enables a rapid and robust search in the high-dimensional space of variable orderings.

## 3. Experiment

### *3.1 Experimental Setup*

We compare the Quantum Hybrid Reinforcement Learning (QHRL) Optimizer against classical heuristic methods, including:

1. Sifting Optimizer [1]: A local search algorithm that iteratively improves the ordering via systematic sifting.
2. Simulated Annealing [2]: A probabilistic method that utilizes temperature-based acceptance criteria to avoid local minima.
3. Quantum Enhanced RL Hybrid Optimizer: Our proposed method that integrates quantum-inspired global search and RL-based local refinement.

The experiments are conducted under controlled parameters such as a fixed time limit per run, number of repetitions (runs), and varying problem sizes. This controlled setup allows us to isolate the effects of the optimization strategies on performance.

We utilize Deceptive Local Minima benchmark to simulate different characteristics of the variable ordering problem.

- **Deceptive Local Minima [3]:** Designed to trap classical local search methods by presenting non-intuitive optimal orderings.

### *3.2 Results*

Convergence plots provide insights into how algorithms progress toward optimal solutions over time. Figures 1-4 (Quantum Hybrid Reinforcement Learning, Sifting, Simulated Annealing, Comparison of three algorithms) show the convergence behavior for the three algorithms on the Deceptive Local Minima benchmark with n = 64, a problem size that presents significant challenges for conventional optimization methods. It is worth noting that while we refer to 712 as the "ground truth" value, this was established through extensive optimization rather than exhaustive enumeration of all permutations (which would be computationally infeasible for n=64).

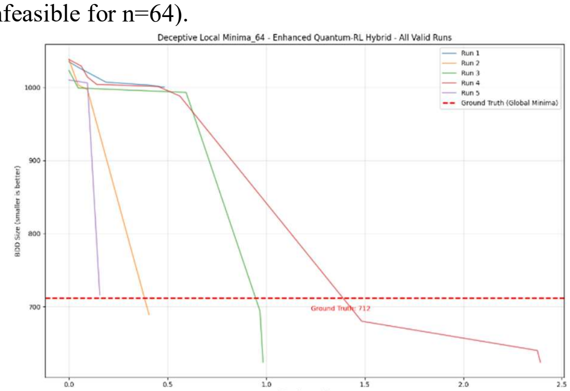


Fig. 1. Deceptive Local Minima 64 – Enhanced QRL Hybrid

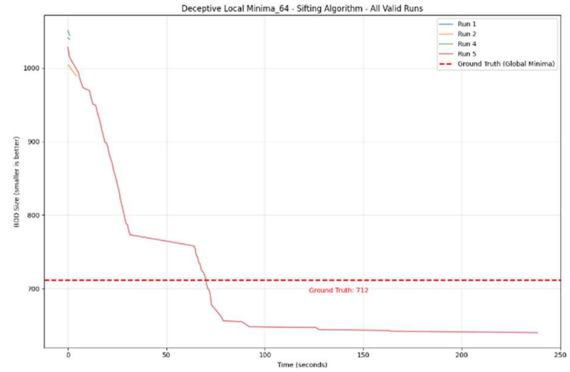


Fig. 2. Deceptive Local Minima 64 – Sifting Algorithm

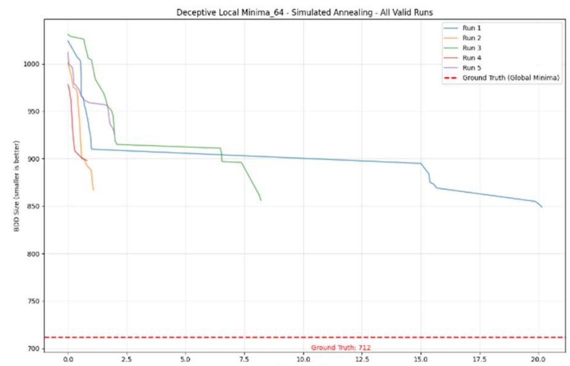


Fig. 3. Deceptive Local Minima 64 – Simulated Annealing

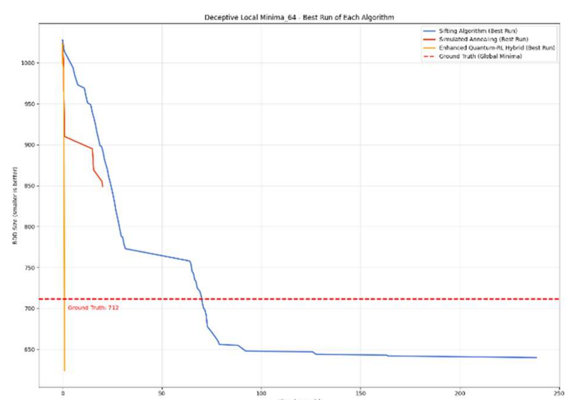


Fig. 4. Comparison of Three Algorithms

1. **Quantum Hybrid Reinforcement Learning (QHRL)** demonstrates remarkably rapid convergence, reaching the ground truth value (712) almost instantaneously. This exceptional performance suggests that the quantum-inspired tunneling mechanism effectively bypasses the deceptive local minima that trap classical methods.
2. **Sifting Algorithm** shows a much slower convergence pattern, requiring approximately 75-100 seconds to approach its best solution. While it

eventually achieves reasonable BDD sizes (around 640), it never reaches the ground truth value despite substantial computation time.

3. **Simulated Annealing** exhibits the poorest performance, with its best run plateauing around 850, significantly higher than the ground truth. This indicates that the temperature-based escape mechanism is insufficient for the deceptive landscape of this benchmark.

## 4. Conclusion

This paper introduced the Quantum Hybrid Reinforcement Learning (QHRL) optimizer, a novel approach to the variable ordering problem in Binary Decision Diagrams. By integrating quantum-inspired tunneling strategies with reinforcement learning techniques, we created a method that effectively balances global exploration and local exploitation in the high-dimensional space of variable orderings.

## Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (Ministry of Science and ICT) (No. RS-2022-00144150)

## REFERENCES

[1] Rudell, R. (1993, November). Dynamic variable ordering for ordered binary decision diagrams. In *Proceedings of 1993 International Conference on Computer Aided Design (ICCAD)* (pp. 42-47). IEEE.

[2] Aarts, E., Korst, J., & Michiels, W. (2005). Simulated annealing. *Search methodologies: introductory tutorials in optimization and decision support techniques*, 187-210.

[3] Felt, E., York, G., Brayton, R., & Sangiovanni-Vincentelli, A. (1993, September). Dynamic variable reordering for bdd minimization. In Proceedings of EURO-DAC 93 and EURO-VHDL 93-European Design Automation Conference (pp. 130-135). IEEE.