Optimizing Concentrated Liquidity Management: A Synthetic-to-Historical Deep Reinforcement Learning Strategy

Ricardo Arcifa
SRI. Engineering
TUS: Midlands Midwest
Athlone, Ireland
a00279376@student.tus.ie

Yuhang Ye
SRI. Engineering
TUS: Midlands Midwest
Athlone, Ireland
yuhang.ye@tus.ie

Yuansong Qiao
SRI. Engineering
TUS: Midlands Midwest
Athlone, Ireland
yuansong.qiao@tus.ie

Brian Lee
SRI. Engineering
TUS: Midlands Midwest
Athlone, Ireland
brian.lee@tus.ie

***Abstract*—Automated market makers (AMMs) have revolutionized decentralized finance (DeFi), enabling trustless asset exchange through algorithmic liquidity pools. One notable AMM is Uniswap, which improved capital efficiency by introducing concentrated liquidity, allowing Liquidity Providers (LPs) to allocate capital within specific price ranges. However, this flexibility requires active management to maximize fee generation while effectively addressing impermanent loss (IL) and gas-costs. We formulate concentrated liquidity management (CLM) as a stochastic control problem and propose deep reinforcement learning (RL) strategies, specifically Proximal Policy Optimization (PPO) and Deep Q-Network (DQN), to optimize liquidity positioning. Unlike heuristics, RL agents adapt dynamically to market conditions. We adopt a two-stage synthetic-to-historical evaluation strategy, involving RL model training on synthetic data and evaluation on historical data. Results show that training on synthetic data and testing historic data allows RL-based strategies to perform well across regimes, capturing more fees while keeping net performance ahead of a passive buy-and-hold strategy. These findings highlight the robustness and practical viability of such strategies for CLM.**

***Index Terms*—Decentralized Finance, Reinforcement Learning, Automated Market Makers, Uniswap v3, Concentrated Liquidity, Deep Q-Network, Proximal Policy Optimization, Impermanent Loss.**

I. Introduction

Automated market makers (AMMs) have transformed decentralized finance (DeFi) by enabling trustless trading and liquidity provision. Uniswap introduced the concept of concentrated liquidity in v3, which allows liquidity providers (LPs) to concentrate capital in specific price ranges [1]. Compared to Uniswap v2, where liquidity spreads uniformly over all possible prices, v3 has improved capital efficiency by introducing price ranges for liquidity provision. However, concentrated liquidity requires LPs to adjust their positions regularly to remain profitable while minimizing impermanent loss (IL) and gas expenditures.

Heuristic approaches remain the predominant method for Concentrated Liquidity Management (CLM), whereby LPs place fixed price intervals and re-balance based on triggers such as certain price deviations, moving averages, or regular intervals. While these methods are computationally less expensive and straightforward, they are often suboptimal in highly dynamic markets as they cannot adapt quickly to price fluctuations or fee generation.

Current research suggests that RL-based strategies can enhance market environments by continuously learning and adjusting strategies in response to market changes. Algorithms such as Proximal Policy Optimization (PPO) and Deep Q-Network (DQN) are employed across different financial settings, such as algorithmic trading and portfolio management [2]. While recent research comparing heuristics and PPO to optimize CLM has been proposed [3], the application using synthetic-to-historical market data remains relatively unexplored in this context. Synthetic-to-historical strategies include training RL models on generated synthetic data and evaluating them with real historical market data to enhance predictive accuracy and decision-making.

This paper acknowledges this gap and proposes a simulation-based evaluation of synthetic-to-historical deep RL-based strategies for CLM. The principal contributions are proposed as follows:

- **Synthetic-to-historical RL Strategies**: We develop an open source framework that trains on synthetic geometric Brownian motion (GBM) price/volume paths and evaluates the model on one year of historical market data from Uniswap (ETH/USDC), enabling reproducible synthetic-to-historical testing
- **Multi-Model Analysis**: We evaluate heuristic liquidity management strategies alongside RL-based strategies (PPO and DQN) under different market regimes, providing explicit insights into adaptive behaviors and gas-cost trade-offs.
- **Performance Metrics**: We examine net profitability, IL, and gas efficiency across all strategies, demonstrating the practical advantages of RL-based strategies in dynamic liquidity positioning and volatility responsiveness.

This paper proposes a foundation to advance research in CLM and demonstrates the effectiveness of using synthetic-to-historical market data to train and evaluate RL-based agents.

Paper Organization: Section II surveys the background, III explains our methods, IV–VI report and analyze the results, and VII offers the conclusions.

II. Background

A. Automated Market Makers

AMMs are decentralized apps (dApps) that provide digital assets for trading automatically and in a permissionless way. A well-known AMM is Uniswap, which employs a constant product market maker (CPMM) model formulated as [4]:

$$x \cdot y = k, \tag{1}$$

where:

- x and y are the reserves of two assets in a pool,
- k is a constant preserved during trades,
- x corresponds to the quantity of token X,
- y corresponds to the quantity of token Y,
- The product k remains constant when traders exchange tokens, determining how the reserves adjust.

In early iterations of AMMs, liquidity was distributed uniformly across every possible price, which led to shortcomings in capital efficiency. Uniswap v3 addresses these shortcomings by introducing the concept of CLM, allowing LPs to focus their capital in a specific price range [1].

B. Challenges in CLM

Although CLM improves capital efficiency, it also introduces new challenges:

- **Impermanent Loss (IL)**: Price movements outside the active price range can result in losses compared to buy-and-hold strategies.
- **Active Re-balancing**: LPs must periodically readjust their price range position to generate LP fees.
- **Gas-Costs**: Re-balancing actions incur gas fees, impacting fee earnings if done too often.

These challenges underscore the need for robust and adaptive liquidity management practices.

C. Heuristics for CLM

Heuristics remain the prevalent method in CLM because of their simplicity and computational efficiency. Three widely adopted heuristics are:

- **Fixed Price Range Bands**: LPs establish a static price range $[P-\delta, P+\delta]$ and typically do not re-balance unless manually intervened. While computationally efficient, this approach may underperform in volatile markets due to limited adaptability [5], [6].
- **Price Deviation-Based Re-balancing**: Liquidity is re-balanced when the market price deviates significantly from the active range, typically exceeding a threshold before a reset occurs. This approach allows for periodic recentering but may incur higher gas-costs during volatile periods [5].
- **Volatility-Based Re-balancing**: The liquidity range is adjusted dynamically based on recent price swings, often widening during volatile markets to reduce out-of-range risk [7].

D. Reinforcement Learning for CLM

RL-based strategies employ a data-driven approach to learning optimal policies over time. The CLM task can be formalised as a Markov decision process (MDP) because it involves sequential decision-making, satisfies the Markov property, and optimises a reward that balances fee accrual against re-balancing costs. Following Sutton and Barto [8], we write the MDP as $\langle \mathcal{S}, \mathcal{A}, p, \gamma \rangle$ with $0 \le \gamma < 1$. At each step

$$\begin{aligned} (S_{t+1}, R_{t+1}) &\sim p(\cdot \mid S_t = s_t,\ A_t = a_t), \\ G_t &= \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}. \end{aligned} \tag{2}$$

where

- $S_t \in \mathcal{S}$ is the state at time t,
- $A_t \in \mathcal{A}$ is the action chosen by the policy π,
- $p(s', r \mid s, a) = \Pr\{S_{t+1} = s',\ R_{t+1} = r \mid S_t = s, A_t = a\}$ is the joint transition-and-reward kernel,
- R_{t+1} is the scalar reward obtained one step after acting,
- G_t is the return the agent seeks to maximise,
- γ is the discount factor that trades off short- and long-term rewards.

III. Related Work

A. Reinforcement Learning in AMMs and Finance

RL-based strategies have shown promising outcomes in several areas of finance, such as portfolio management, market making, and high frequency trading. In traditional finance, research has demonstrated that RL-based strategies outperform classical approaches in terms of return and risk-adjusted metrics [2]. In the context of market making, RL-based strategies have improved profitability relative to Avellaneda–Stoikov baselines by dynamically adjusting risk variables [9].

The rise of DeFi has motivated research on RL-based strategies in AMMs. Recent studies employing RL for AMM liquidity provisioning have demonstrated robust performance in capturing fees while mitigating IL [3], [10], [11].

Building on these foundations, recent empirical and theoretical work has turned to CLM in AMMs. Current lines of inquiry include training deep RL agents on live Uniswap v3 order flow [12], quantifying how range width and holding period shape IL-adjusted returns [13], and modelling the LP payoff as equivalent to selling a covered call without collecting the time premium of an option [14].

B. RL vs. Heuristic Strategies

Comparative studies show that RL-based strategies often outperform static rules in complex financial settings. In equity trading, for instance, DQN out-earned buy-and-hold and technical indicator methods [15], [16]. Within CLM, RL policies have generated higher fee income and lower IL compared to heuristic range rules [3].

Our work extends these studies by benchmarking PPO and DQN against two heuristic re-balancers: (i) a price threshold rule (HP) that re-positions when the range exits its current

$\pm\lambda$ band, and (ii) a volatility threshold rule (HV) that triggers on elevated seven day price variability. To ensure the triggers fire with realistic frequency on the historical market data, we analyze its rolling seven day coefficient of variation (σ/μ): the 50$^{\text{th}}$ and 75$^{\text{th}}$ percentiles are 2.7% and 4.1%, respectively. We define the thresholds as:

$$\lambda = 0.05, \qquad \sigma_{\text{threshold}} = 0.03, \tag{3}$$

where

- λ is the half width of the price-band ($\pm 5\%$ around the mid price), and
- $\sigma_{\text{threshold}}$ is the volatility level above which the HV rule re-ranges.

IV. Methodology

A. Environment Assumptions

We developed an evaluation framework to investigate RL-based strategies and heuristics to improve CLM on AMMs. The environment employs synthetic-to-historical market data to validate the potential of using synthetic data to train models for real-world CLM strategies.

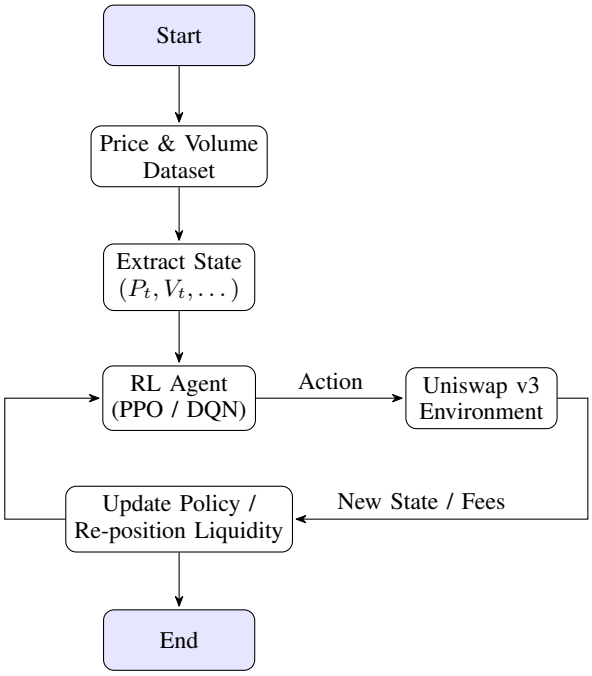

Fig. 1. Workflow of the proposed synthetic-to-historical RL framework for CLM on Uniswap v3.

Price Dynamics: Price movements are modeled with a geometric Brownian motion (GBM):

$$dP_t = \mu P_t\, dt + \sigma P_t\, dW_t, \tag{4}$$

where:

- P_t is the price at time t,
- μ is the drift parameter,
- σ is the volatility,
- W_t is a Wiener process (Brownian motion).

During our simulations, time is discretized (e.g., one step - one day). Prices are logged at each time step to allow the RL agent to learn from sequential data.

Volume Model: Trading volume significantly affects accrued fees. We model trading volume using a synthetic stochastic process for initial training and historical data from Uniswap for evaluation. Each trade updates the reward for the LP.

Concentrated Liquidity Equation: Concentrated liquidity utilizes the tick-based model proposed in Uniswap v3. [1]:

$$L = \min\left(\frac{X}{\frac{1}{\sqrt{P_{min}}} - \frac{1}{\sqrt{P_{max}}}},\ \frac{Y}{\sqrt{P_{max}} - \sqrt{P_{min}}}\right), \tag{5}$$

where:

- X and Y are the token quantities allocated to the position,
- P_{min} and P_{max} denote the lower and upper price bounds,
- $\sqrt{P}$ indicates the square root of the asset price in appropriate units for Uniswap v3.

The objective of the LP is to select ranges that balance fee generation against IL risk.

Gas-Cost Calculations: Daily gas fees are taken from an Etherscan base fee export (16 Mar 2024 – 15 Mar 2025). For every day d we convert the base fee to a dollar figure per Uniswap v3 transaction as:

$$\text{usd\_tx}_d \;=\; \text{b}_d \times 10^{-9}\,(\text{ETH/gwei}) \times G \times \text{ETHUSD}_d, \tag{6}$$

where:

- b_d is the median base fee on day d (in gwei),
- 10^{-9} is the conversion factor from gwei to ETH,
- $G = 260\,000$ denotes the gas units consumed by a typical mint/burn on Uniswap v3 [1] ,
- ETHUSD_d is the ETH–USD close price for the same day (taken from the pool snapshot),
- usd_tx_d is the resulting transaction cost in USD ($).

Table I summarizes the resulting distribution ($0.99–42.84, mean $11.64, standard deviation $8.42; abbreviated as SD). During evaluation this daily series is looked up and perturbed with log normal noise ($\sigma_{\text{rel}} = 0.2$) to mimic intraday variability.

TABLE I
DISTRIBUTION OF 260 000 GAS UNIT TRANSACTION COSTS (MAR 2024 – MAR 2025, 366 OBSERVATIONS; 29 FEB 2024 INCLUDED)

Statistic	USD ($) per tx
Min	0.99
1st Quartile	5.65
Median	9.28
Mean	11.64
3rd Quartile	15.95
Max	42.84
SD	8.42

[1]Etherscan: `mint`, `decreaseLiquidity`, `increaseLiquidity`.

B. Observation and Action Space

As mentioned in Section II-D we cast the CLM problem as a Markov decision process. At every decision step the agent observes

$$s_t \;=\; \big\{P_t,\; O_t,\; L_t,\; V_t,\; F_t^{\mathrm{cum}}\big\}, \tag{7}$$

where

- P_t is the current ETH/USDC mid-price,
- O_t is the number of consecutive steps the position has been out of range (resets to 0 once P_t re-enters the band),
- L_t is the liquidity currently supplied to the AMM position,
- V_t is the on-chain trading volume on day t,
- F_t^{cum} is the cumulative LP fees earned up to (and including) t.

The Action space implementation exposes a two-choice discrete control. At every decision point the agent picks $a_t \in \mathcal{A}$ with

$$\mathcal{A} = \{a_0,\, a_1\} \tag{8}$$

where:

- a_0 represents the decision to keep the current liquidity range unchanged;
- a_1 represents the decision to withdraw the existing position, pay the applicable gas fee, and supply a new symmetric band centred on the current price, whose half-width is drawn uniformly from $[0.04P_t,\, 0.08P_t]$.

C. Reward Function

$$\begin{aligned} R_t \;&= \big(V_t^{\text{port}} - V_{t-1}^{\text{port}}\big) - \lambda_{\text{gas}}\,\text{GasCost}_t \\ &\quad + \alpha\, F_t - \lambda_{\text{oor}}\,\mathbf{1}_{\{O_t>5\}}\, V_{t-1}^{\text{port}}. \end{aligned} \tag{9}$$

where:

- V_t^{port} is the mark-to-market value of the entire wallet *and* pool position at step t,
- F_t are the LP fees accrued during $(t-1, t]$,
- GasCost_t is the gas paid if a_1 is taken at t,
- $\mathbf{1}_{\{O_t>5\}}$ is the indicator that turns on once the position has spent more than five consecutive steps out of range,
- λ_{gas} is the gas-cost weight, default 1,
- α is the fee-amplification factor, default 10,
- λ_{oor} is the out-of-range penalty weight, default 0.01.

IL is already captured in the ΔV^{port} term, so no extra IL penalty is applied.

D. Re-Balancing Strategies

We benchmark two deterministic re-balancers, each using a relative price-band half width of $\lambda = 0.05$ and a volatility trigger of $\sigma_{\text{threshold}} = 0.03$ computed over a rolling 7-day window.

- **Price-band heuristic (HP):** Re-positions whenever the spot exits its current band,

$$\big|P_t - P_{\text{mid}}\big| \;>\; \lambda\, P_{\text{mid}}, \tag{10}$$

  where:
  - P_t is the spot price at time t,
  - P_{mid} is the midpoint of the active band (set at the previous liquidity placement),
  - λ is the relative half width of the band (we use $\lambda = 0.05$, giving a $\pm 5\%$ range).
- **Volatility heuristic (HV):** Re-positions when the 7-day coefficient of variation exceeds the threshold,

$$\sigma_{P,\tau} \;>\; \sigma_{\text{threshold}}, \quad \tau = 7 \text{ days}. \tag{11}$$

  where:
  - $\sigma_{P,\tau}$ is the rolling SD of prices over a τ-day window, divided by the window mean,
  - $\sigma_{\text{threshold}}$ is the trigger level (set to 0.03, i.e. 3%).

Throughout the experiments we therefore apply equation 3, which captures approximately the noisier half of weeks in the historical data.

E. Training and Hyperparameters

We train and evaluate PPO and DQN policies using `stable-baselines3` package [17] with key hyperparameters:

- **PPO**: learning rate $\alpha = 2.5 \times 10^{-4}$, discount factor $\gamma = 0.93$, entropy coefficient $= 0.05$.
- **DQN**: learning rate $\alpha = 1 \times 10^{-4}$, replay buffer size $= 10^6$, target update frequency $= 1000$ steps.
- **Multiple random seeds**: Each RL method is trained for 500 000 steps with three seeds $(0, 1, 2)$ to reduce optimization variance. After training, every policy is evaluated on the 1-year ETH/USDC dataset under 50 independent seeds $(0 : 49)$. Each seed re-draws the stochastic gas cost and volume shocks, so that statistical dispersion can be reported.

Gas fees and other trading assumptions are applied consistently, and every policy is trained solely on synthetic data. Each agent is trained with three random seeds (0, 1, 2) and evaluated over 50 independent seeds per strategy, capturing variability in gas-cost and volume shocks. A paired Wilcoxon signed rank test on annual percentage returns (APR) (Table VI) confirms advantage of PPO over DQN ($p = 2.36 \times 10^{-3}$). PPO and DQN outperformed the heuristic strategies based on HP ($p = 2.12 \times 10^{-9}$) and HV. ($p < 10^{-14}$). The alignment of synthetic training results with one year of historical market data supports the robustness and practical viability of the proposed strategies.

TABLE II
ILLUSTRATIVE HYPERPARAMETER SETTINGS FOR PPO AND DQN

Parameter	PPO Value	DQN Value
Learning Rate α	2.5×10^{-4}	1.0×10^{-4}
Discount Factor γ	0.93	0.99
Entropy Coefficient	0.05	–
Replay Buffer Size	–	1×10^6
Target Update	–	1000 steps
Training Steps	500 000 (per seed)	

F. Volatility Conditions

During training using synthetic market data, we applied volatility levels (σ) as follows:

- **Low Volatility** ($\sigma < 0.02$): Small, steady price fluctuations.
- **Medium Volatility** ($0.02 \leq \sigma < 0.05$): Noticeable price movements with moderate swings.
- **High Volatility** ($\sigma \geq 0.05$): Frequent and substantial price swings, significantly impacting LP positions.

TABLE III
STRATEGY PROFITABILITY UNDER DIFFERENT VOLATILITY REGIMES

Volatility Regime	PPO ($)	DQN ($)	Heuristic ($)
Low ($\sigma < 0.02$)	3200	3100	2900
Medium ($0.02 \leq \sigma < 0.05$)	5000	4600	3700
High ($\sigma \geq 0.05$)	6000	5100	3400

Table III summarizes the average profit earned by each strategy under the three synthetic volatility regimes.

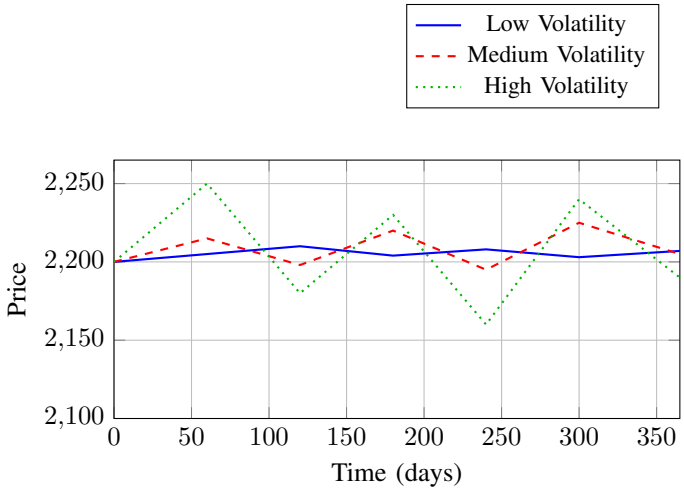

Fig. 2. Representative price paths under three volatility regimes used for synthetic market simulations; actual fluctuations may differ.

G. Reproducibility Assets

The complete simulation framework, CSV result file (`results/seed_runs.csv`) and all other relevant files and scripts are openly released at https://github.com/rarcifa/clm-rl-s2h.

V. RESULTS

A. Simulation Setup

We conduct evaluations on 50 seeds for each strategy, using identical initial capital and gas-cost assumptions. Each simulation started with the same initial capital allocation, employing consistent assumptions regarding reward function weights, gas-costs, and re-balancing parameters. The simulation setup allowed us to rigorously evaluate the effectiveness of each strategy in terms of profitability, risk management, gas efficiency, and overall adaptability under various market scenarios.

B. Evaluation Metrics

The following metrics are employed to evaluate each strategy:

- **Profitability**: Net fee returns minus gas-costs.
- **IL**: Extent to which strategy performance diverges from a simple hold strategy.
- **Gas Efficiency**: Frequency and expense of re-balancing actions.

C. Historical Back-test

To assess the generalizability of models trained on synthetic data, we evaluate their performance using historical ETH/USDC market data from Uniswap v3, covering 365 days from March 2024 to March 2025. Over the course of the evaluation, the price of ETH declined from $3525.13 at the start of the simulation to $1926.93 at the end. We retained the same initial capital, reward function weights, and gas assumptions across all strategies. This approach clarifies how each strategy performs when confronted with realistic market dynamics.

The Final Portfolio Value includes both wallet and LP-held ETH/USDC converted to USDC using the final ETH price. Net Profitability is calculated as:

$$\Pi_{\text{net}} = V_{\text{final}} - C_{\text{gas}} - I_{\text{init}} \tag{12}$$

where:

- Π_{net} is the net profit realized by the strategy.
- V_{final} is the final portfolio value (in $), inclusive of LP fees held in wallet or active position.
- C_{gas} is the total gas-cost accumulated from re-positioning events.
- I_{init} is the initial investment amount (e.g., $10 000).

LP Fees are the cumulative fees earned by the agents throughout the simulation and are already included in the Final Portfolio Value via USDC balance increases. Total Gas Fees reflect the cumulative cost of re-positioning events.

D. Profit and Cost Analysis

Across the 50-seed evaluation (Table V), PPO produced the highest average APR (46%), followed by DQN (18.6%). HP averaged $-30.6\% \pm 33$ percentage points (pp), whereas HV clustered around $-82.4\% \pm 17$ pp. Both heuristics underperform compared to the RL agents, and HV is negative for every seed. SD figures confirm the need to report variability, but PPO still outperforms DQN in 38 of the 50 seeds. Aggregated statistics (Table V) show PPO exceeds DQN by 27.4 pp in mean APR, and the paired Wilcoxon test (Table VI) confirms this gap is relevant ($p = 2.36 \times 10^{-3}$).

These results indicate that the RL-based strategies were more robust in adapting to changing market conditions. Figure 4 visualizes the same dispersion, highlighting that the entire inter-quartile range of PPO strategies lies well above zero whereas DQN straddles.

The APR for each strategy was calculated based on the final portfolio value, accounting for LP fees earned and gas-costs incurred, relative to the initial investment. Specifically, we define APR as:

$$\text{APR} = \left(\frac{V_{\text{final}} - C_{\text{gas}} - I_{\text{init}}}{I_{\text{init}}} \right) \times 100 \tag{13}$$

where:

- V_{final} is the final portfolio value (in \$), including both wallet and in-range LP assets at the final ETH price.
- C_{gas} denotes the total gas-costs incurred from re-positioning liquidity.
- I_{init} is the initial investment amount (e.g., \$10 000).

E. Benchmark and IL Comparison

Consistent with the standard definition [18], IL is measured against a passive buy-and-hold position holding the same initial assets, 1.4184 ETH and \$5 000 USDC (exactly half of the \$10 000 stake at $t = 0$), and by construction satisfies IL ≤ 0, with larger magnitudes indicating greater underperformance. We report the median IL across the 50 evaluation seeds to reduce the influence of outliers.

PPO and DQN outperform the buy-and-hold benchmark on 44/50 and 32/50 seeds respectively, despite larger IL values, while the two heuristic baselines fail to beat the buy-and-hold strategy once gas fees are deducted (Fig. 3). This is because (i) their many short in-range windows accumulate substantial swap fees and (ii) the large ETH-to-USDC re-balancing triggered by the year-long down-move converts most ETH to the stable-coin leg.

The median portfolio of the PPO seeds stays above the buy-and-hold curve for the entire year. DQN also outperforms the benchmark, though its spread across seeds is wider. Both heuristics end the year only slightly ahead of buy-and-hold after gas fees, remaining well below the RL agents.

The final portfolio value of PPO exceeds DQN strategies in 38 cases, confirming the aggregate APR results reported earlier. DQN occasionally wins in high fee, low volatility paths where its tighter ranges harvest more fees.

F. Gas-Cost Robustness

Table I shows that 260 000 gas unit costs during our evaluation year ranged from \$0.99 to \$42.84 (mean \$11.64, $\sigma = \$8.42$). We therefore stress test the agents by halving and doubling every daily fee. The resulting APRs are reported in Table IV. The APR of PPO declines by 11.2 pp as gas-costs move from the 0.5× band to the 2× band. DQN stays positive but gives up 13 pp over the same span. HP deteriorates from -13.5% to -64.7%, while HV falls from -38.9% to -169.5%.

G. Policy Behavior Analysis

The re-balancing and liquidity management behaviors observed across strategies reveal distinct patterns:

- PPO exhibited more stable portfolio trajectories than DQN strategies. Its re-balancing decisions were more aligned with periods of volatility.

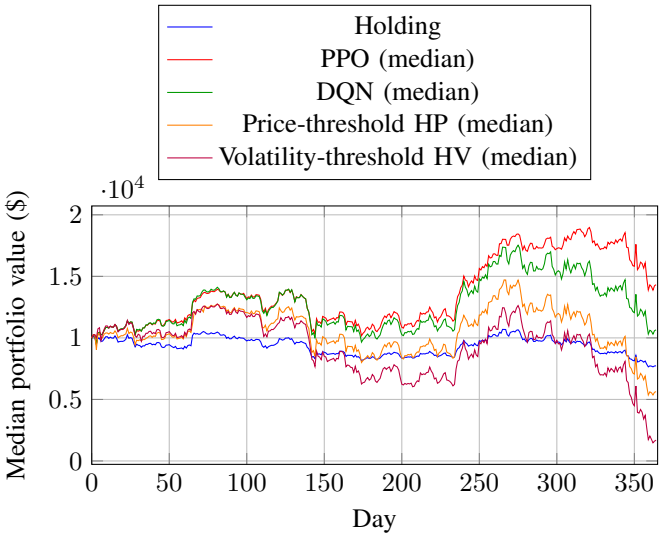

Fig. 3. Median *net-of-gas* portfolio value (\$) across 50 seeds (starting from \$10 000). The buy-and-hold benchmark is shown for reference.

TABLE IV
MEAN APR (%) UNDER THREE GAS-COST SCENARIOS (AVERAGED OVER THE SAME 50 SEEDS USED IN TABLE V).

Strategy	0.5× Gas	1× Gas	2× Gas
PPO	49.7	46.0	38.5
DQN	22.9	18.6	9.9
HP	-13.5	-30.6	-64.7
HV	-38.9	-82.4	-169.5

- DQN re-balanced at a comparable frequency to PPO, but tended to accumulate heavier ETH exposure in some scenarios. While it earned more LP fees in several simulations, this sometimes came at the cost of greater portfolio volatility and sensitivity to ETH price declines.
- HP and HV rely on fixed rule triggers (price deviation and volatility spike, respectively). Both are fully deterministic, but their gas usage diverges sharply. HP re-balances only when the spot price exits its ±5% band, incurring high gas-costs, whereas HV fires on almost every high volatility day. Neither baseline adapts its range width or timing. The resulting fee capture is modest and the strategies remain unprofitable once gas costs are deducted.

By training the RL agents first on synthetic data and then validating on historical market data, we were able to expose these cost–benefit trade-offs and show why PPO and DQN policies outperform heuristic strategies under real market conditions.

VI. DISCUSSION

This section examines the key observations further, emphasizing how volatility, gas-costs, and trade-offs affect the outcomes. It also outlines possible directions for further development.

TABLE V
MEAN ± SD ACROSS 50 SEEDS (FEES IN K$; APR AND IL IN %)

	PPO	DQN	HP	HV
APR (%)	46 ± 40	18.6 ± 44.2	-30.6 ± 33	-82.4 ± 17
IL (%)	-66 ± 25	-64.3 ± 12	-52.9 ± 26	-64.1 ± 10
LP Fees	12.7 ± 3.7	9.9 ± 4.4	6.7 ± 3.27	7.7 ± 1.9
Gas Fees	0.7 ± 0.08	0.8 ± 0.1	3.4 ± 0.751	8.7 ± 0.2

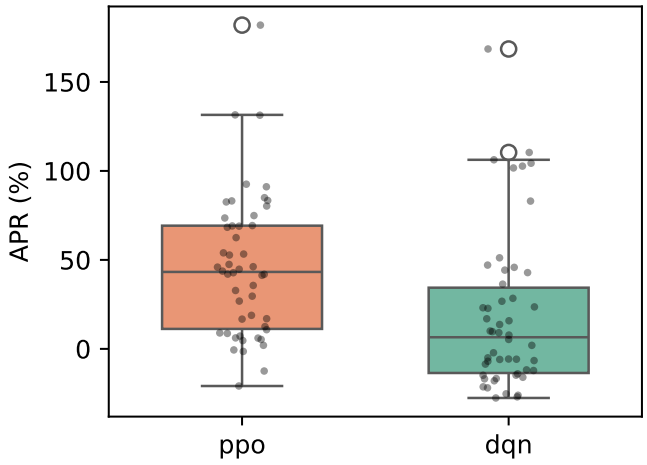

Fig. 4. APR distribution of the two RL agents over 50 evaluation seeds. Boxes show the inter-quartile range and median; black dots are per-seed outcomes. PPO attains the highest median APR while DQN exhibits both lower typical performance and wider dispersion.

A. Portfolio Composition and Exposure Bias

While LP fees are often seen as the primary driver of LP profitability, our results show that the final asset composition plays a significant role. PPO frequently maintained a more balanced ETH/USDC ratio, which helped preserve value during ETH price declines. In contrast, DQN sometimes ended with ETH heavy portfolios that were more sensitive to price movements, leading to a lower final value in some simulations.

B. Fee Capture vs. Risk Exposure

Our results demonstrate that higher LP fee income does not necessarily translate to higher profitability. In a few simulations, DQN performed better than PPO in raw fees but ended with lower portfolio value due to unhedged ETH exposure.

C. Behavioral Robustness Under Market Conditions

Despite being trained solely on synthetic price data, both PPO and DQN adapted reasonably well to historical market conditions. The fact that both models generalized across price regimes suggests that key behavioral patterns, such as rebalancing and position symmetry, can transfer from synthetic to real-world conditions. Unlike RL agents, both heuristic strategies apply a fixed rule, but their realized returns vary with the seed specific gas-cost and volume shocks. Hence we report their full dispersion across the 50 seeds.

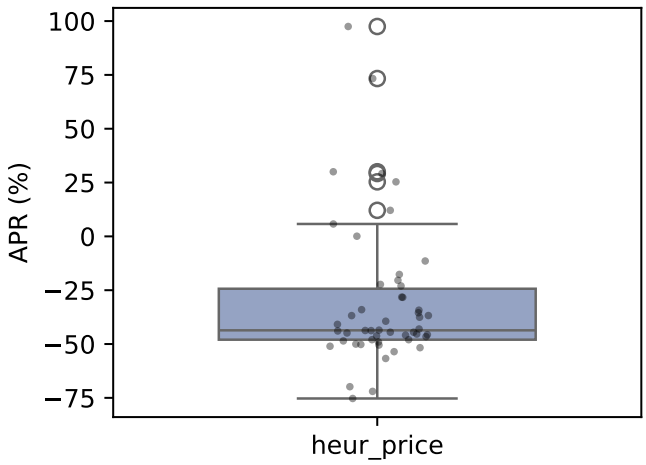

Fig. 5. APR distribution for the *price threshold heuristic* (HP) across the same 50 seeds. Although the rule occasionally achieves modest gains, its median APR is negative, highlighting the difficulty of using a fixed price-band in a volatile market.

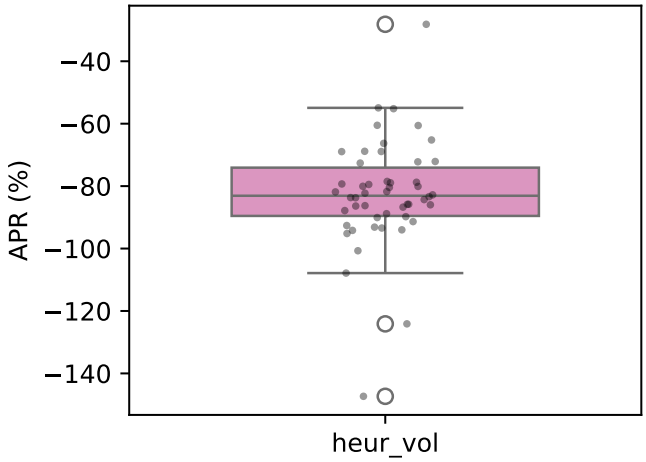

Fig. 6. APR distribution for the *volatility threshold heuristic* (HV). The strategy consistently underperforms, clustering around -82.4% APR.

D. Future Research Directions

While the presented results offer valuable insights into current strategies, several promising directions remain unexplored. Future research can deepen our understanding and enhance the robustness of these strategies. Specifically, these directions include:

- Investigating Partially Observable Markov Decision Processes (POMDPs) to address uncertainty and partial observability inherent in market environments, potentially leading to more resilient decision-making frameworks.
- Exploring synthetic-to-real data simulations across different use cases, explicitly focusing on improving generalization from synthetic data training to historic market scenarios, thereby enhancing the practical applicability of RL strategies.

TABLE VI
PAIRED WILCOXON SIGNED–RANK p-VALUES ON APR (50 SEEDS)

Comparison	p-value
PPO vs. DQN	2.36×10^{-3}
PPO vs. HP	7.66×10^{-12}
PPO vs. HV	1.78×10^{-15}
DQN vs. HP	6.92×10^{-9}
DQN vs. HV	1.78×10^{-15}
HP vs. HV	$< 10^{-14}$

- Studying alternative RL algorithms (e.g., Soft Actor Critic or Transformer-based RL) that might further enhance policy robustness and sample efficiency.
- Integrating additional market data sources, such as on-chain analytics and macroeconomic indicators, to enrich RL decision-making.
- Testing RL-based CLM strategies in live DeFi environments to evaluate performance under diverse liquidity conditions.

VII. CONCLUSION

This paper investigates deep RL-based and heuristic strategies for CLM, showcasing the transition from synthetic data training to evaluation on historical market data. The synthetic-to-historical strategy provided key insights into the strengths and limitations of each approach:

- PPO yields the highest mean APR (46%) and the lowest gas usage, outperforming DQN by 27.4 pp (paired $p = 2.36 \times 10^{-3}$).
- DQN outperformed the heuristic baselines in most simulations but exhibited higher variance in returns due to greater ETH exposure and sensitivity to market conditions.
- HP and HV finish the year with negative average APRs (-30.6% and -82.4%), reflecting heavy gas usage and capital inefficiency.
- Market volatility and re-balancing costs were primary performance drivers, underscoring the importance of adaptive and cost-aware policy design.

In conclusion, the synthetic-to-historical evaluation approach confirms the practical viability of RL-based strategies for CLM, particularly in volatile or unpredictable markets. While no strategy guarantees optimal results in every scenario, deep RL strategies like PPO offer a compelling balance of adaptability, fee generation, and gas efficiency, making them well-suited for real-world deployment.

ACKNOWLEDGMENT

This work was supported by the President's Doctoral Scholarship from the Technological University of the Shannon (TUS), Ireland, awarded in 2021. The authors wish to thank TUS for its financial support and continued assistance, which facilitated this research.

REFERENCES

[1] H. Adams, N. Zinsmeister, M. Salem, R. Keefer, and D. Robinson, "Uniswap v3 core whitepaper," https://uniswap.org/whitepaper-v3.pdf, 2021.

[2] F. Espiga-Fernández, Álvaro García-Sánchez, and J. Ordieres-Meré, "A systematic approach to portfolio optimization: A comparative study of reinforcement learning agents, market signals, and investment horizons," *Algorithms*, vol. 17, no. 12, p. 570, 2024.

[3] H. Xu and A. Brini, "Improving defi accessibility through efficient liquidity provisioning with deep reinforcement learning," *arXiv preprint*, 2024. [Online]. Available: https://arxiv.org/abs/2501.07508

[4] Uniswap, "How uniswap works — uniswap v2 protocol overview," https://docs.uniswap.org/contracts/v2/concepts/protocol-overview/how-uniswap-works, documentation page (Accessed: Jan 12, 2025).

[5] R. Fritsch, "Concentrated liquidity in automated market makers," in *Proceedings of the 2021 ACM CCS Workshop on Decentralized Finance and Security (DeFi'21)*. Association for Computing Machinery, 2021, pp. 15–20.

[6] F. Spinoglio, "Concentrated liquidity in uniswap v3: A new strategy to optimize the capital bear market," *Journal of New Finance*, vol. 3, no. 2, p. Article 1, 2024.

[7] Álvaro Cartea, F. Drissi, and M. Monga, "Decentralized finance and automated market making: Predictable loss and optimal liquidity provision," *SIAM Journal on Financial Mathematics*, vol. 15, no. 3, pp. 931–959, 2024.

[8] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA: MIT Press, 2018, see Chapter 3 for the formal MDP and return definitions. [Online]. Available: http://incompleteideas.net/book/the-book-2nd.html

[9] J. F. Marin, D. D. P. de Vera, and E. L. Gonzalo, "A reinforcement learning approach to improve the performance of the Avellaneda–Stoikov market-making algorithm," *PLoS ONE*, vol. 17, no. 12, 2022, article e0277042.

[10] T. Lim, "Predictive crypto-asset automated market maker architecture for decentralized finance using deep reinforcement learning," *Financial Innovation*, vol. 10, no. 1, p. 144, 2024.

[11] Óscar Fernández Vicente, F. Fernández, and J. García, "Automated market maker inventory management with deep reinforcement learning," *Applied Intelligence*, vol. 53, no. 19, pp. 22 249–22 266, 2023.

[12] H. Zhang, X. Chen, and L. F. Yang, "Adaptive liquidity provision in uniswap v3 with deep reinforcement learning," *arXiv preprint*, 2023. [Online]. Available: https://arxiv.org/abs/2309.10129

[13] T. Drossos, D. Kirste, N. Kannengiesser, and A. Sunyaev, "Automated market makers: Toward more profitable liquidity provisioning strategies," in *Proceedings of the 40th ACM/SIGAPP Symposium on Applied Computing (SAC '25)*. ACM, 2025, p. 8 pp.

[14] J. Hasbrouck, T. J. Rivera, and F. Saleh, "An economic model of a decentralized exchange with concentrated liquidity," *SSRN Electronic Journal*, 2024, sSRN 4529513. [Online]. Available: https://ssrn.com/abstract=4529513

[15] V. M. Ngo, H. H. Nguyen, and P. V. Nguyen, "Does reinforcement learning outperform deep learning and traditional portfolio optimization models in frontier and developed financial markets?" *Research in International Business and Finance*, vol. 65, p. 101936, 2023.

[16] J. Liu and Y. Kang, "Automated cryptocurrency trading approach using ensemble deep reinforcement learning: Learn to understand candlesticks," *Expert Systems with Applications*, vol. 237, no. Part A, p. 121373, 2024.

[17] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, and R. Traore, "Stable-baselines3: Reliable reinforcement learning implementations," https://github.com/DLR-RM/stable-baselines3, 2021, software library, commit v1.7.0 (accessed 12 Jan 2025).

[18] Uniswap, "Understanding returns—why is my liquidity worth less than i put in?" https://docs.uniswap.org/contracts/v2/concepts/advanced-topics/understanding-returns#why-is-my-liquidity-worth-less-than-i-put-in, documentation page (Accessed: Jan 12, 2025).