Reinforcement Learning for Perpetual Futures Market Making

Martin Cekal

FFA Working Paper 4/2026

UNIVERSITY OF ECONOMICS AND BUSINESS PRAGUE

VŠE

FACULTY OF FINANCE AND ACCOUNTING

Bibliographic information:

Čekal M. (2026). Reinforcement Learning for Perpetual Futures Market Making. FFA Working Paper 4/2026, FFA, Prague University of Economics and Business, Prague.

This paper can be downloaded at: wp.ffu.vse.cz

Contact e-mail: ffawp@vse.cz

Reinforcement Learning for Perpetual Futures Market Making

Martin Cekal
Department of Banking and Insurance, University of Economics
martin.cekal@vse.cz

May 2026

Abstract

This paper studies market making in BTC and ETH perpetual futures with a simple event-driven reinforcement-learning setup. PPO chooses quote skew and quote width, while the environment tracks execution gains and inventory losses separately. A temporary reduction of the adverse-fill penalty makes early training more stable. Across simulated and historical data, PPO quotes more selectively, carries smaller inventory, and usually ends with better portfolio value than a zero-action baseline. BTC is tighter than ETH, so the same control logic requires market-specific calibration, yet the qualitative policy behavior remains consistent across both markets. The results are therefore encouraging in a narrow sense: a compact and interpretable RL policy can improve inventory control and overall portfolio outcomes in two different crypto perpetual environments, even though robust profitability claims remain out of scope.

1 Introduction

Perpetual futures market making balances spread capture, adverse selection, and inventory risk on an event-driven stream of book and trade updates. Classical inventory-based models from Ho and Stoll [1] to Avellaneda and Stoikov [3] give the right intuition, but in a simulator the trade-off is pathwise: a quote can earn immediate edge, lose value at fill time, and create later mark-to-market risk through the inventory it leaves behind.

Five ingredients matter most. First, the control is written compactly through reservation adjustment and reservation spread in log-price space. Second, step-wise portfolio value decomposes into trade edge and inventory mark-to-market fluctuation, while maker trade edge splits into immediate quote edge and fill-time fluctuation. Third, scheduled damping of adverse fill fluctuation through ϕ_{fluc} reduces variance in early PPO training. Fourth, the same accounting decomposition is examined on both simulated and historical perpetual data. Fifth, the BTC–ETH comparison shows that a crypto perpetual simulator cannot be transferred across markets by rescaling only the price level: spread clustering, event-arrival intensity, and short-horizon volatility priors all change materially. The main empirical pattern is simple: PPO behaves mainly as a conservative inventory-control rule, especially on ETH, where the zero-action baseline accumulates much larger exposure.

Table 1: Notation used throughout the paper.

Symbol	Meaning
t	Decision-step index.
m_t	Mid price at step t.
q_t	Inventory before the next market update.
c_t	Cash position.
PV_t	Portfolio value, $PV_t = c_t + q_t m_t$.
ra_t	Reservation adjustment in log-price space.
rs_t	Reservation spread in log-price space.
$b_{quote,t}$	Bid-side quote.
$a_{quote,t}$	Ask-side quote.
p_t	Execution price.
s_t	Executed size.
d	Buy/sell direction indicator, $d \in \{+1, -1\}$.

2 Related Background

Classical market-making models balance spread capture against inventory risk, from dealer-pricing models such as Ho and Stoll [1] to the reservation-price framework of Avellaneda and Stoikov [3]. The broader microstructure setting follows O'Hara [2]. On the learning side, PPO [6] is a practical policy-gradient method for event-driven control, and the general RL framing follows Sutton and Barto [4]. Learning-based trading problems beyond market making include optimal execution; see Nevmyvaka et al. [5]. The accounting split used here is also close in spirit to the distinction between execution edge and inventory-driven valuation changes in Cartea et al. [7].

Within RL market making, Spooner et al. [8] show that the task can be learned directly in a limit-order-book simulator with inventory-aware reward shaping. Guéant and Manziuk [9] extend deep RL to high-dimensional dealer inventories in corporate bonds. Spooner and Savani [10] focus on robustness under simulator misspecification. Gasperov and Kostanjcar [11] add predictive signals, Gasperov and Kostanjcar [12] use a Hawkes-based order-flow model, and Haider et al. [13] study multi-asset transfer through multi-task learning.

The contribution here is narrower. The study stays with perpetual futures, keeps the state and action design compact, and makes the accounting decomposition central. BTC and ETH are used to test whether the same inventory-aware control logic remains interpretable across two crypto markets with different spread and order-flow structure.

Table 2: Interpretation of the compact state-action design.

Component	Role in the control problem
$pfolio\_delta_t$	Inventory-risk signal; pushes the policy to skew quotes toward reducing exposure.
ols_t	Local liquidity signal; indicates whether the market is currently tight or wide.
rv_fast_t	Short-horizon uncertainty signal; encourages more cautious quoting when recent mid-price movement is unstable.
ra_t	Moves the quote center; redistributes fill pressure between bid and ask.
rs_t	Changes quote width; controls overall willingness to trade.

3 Perpetual Market-Making Environment

Each environment instance observes the top of book for one active perpetual market at a time, and the empirical study applies the same setup to both BTC and ETH. At each decision step, the state is

$$s_t = (pfolio\_delta_t,\ ols_t,\ rv\_fast_t)\,, \tag{1}$$

where $pfolio\_delta_t$ is portfolio delta, ols_t is a log spread observation, and rv_fast_t is a fast annualized realized-volatility estimate based on perpetual mid returns.

3.1 State Interpretation

The three state components have clear roles. $pfolio\_delta_t$ is the inventory signal and drives quote skew. ols_t indicates whether the market is currently tight or wide. rv_fast_t captures short-horizon uncertainty and encourages more cautious quoting after unstable price moves. The state is intentionally small: it separates exposure, liquidity, and local volatility without adding weakly interpretable inputs.

The action is a two-dimensional continuous vector,

$$a_t = (ra_t, rs_t), \tag{2}$$

where ra_t shifts the quote center and rs_t controls quote width. In the perpetual configuration studied here, $ra_t \in [-5\times10^{-5}, 5\times10^{-5}]$ and $rs_t \in [0, 5\times10^{-4}]$ in log-price units.

The control pair is equally compact. ra_t moves the quote center and rs_t sets quote width, so the learned behavior can be read directly as skew and aggressiveness. Table 2 summarizes the role of each state and action component.

3.2 Quote Construction

For perpetuals, the quoting rule is written in log-price form as

$$\log a_{quote,t} = \log m_t + ra_t + \frac{rs_t}{2}, \tag{3}$$

$$\log b_{quote,t} = \log m_t + ra_t - \frac{rs_t}{2}. \tag{4}$$

After exponentiation,

$$a_{quote,t} = \exp\Big(\log m_t + ra_t + \frac{rs_t}{2}\Big)\,, \tag{5}$$

$$b_{quote,t} = \exp\Big(\log m_t + ra_t - \frac{rs_t}{2}\Big)\,. \tag{6}$$

In the reported experiments, the spread control is centered relative to the observed market spread before exponentiation; for exposition, however, the direct formulation above is sufficient. Final price-space quotes are tick-aligned using the market tick size, which is \$0.5 for BTC and \$0.05 for ETH.

3.3 Fill Conditions

The model considers only the best ask and best bid. A maker fill can occur in two ways. First, an order-book touch crosses the standing quote: a buy fill occurs when the new ask is at or below the quoted bid, and a sell fill occurs when the new bid is at or above the quoted ask. Second, an incoming trade print is treated as execution against the standing quote: a sell print can trigger a buy fill when its price is below the quoted bid, and a buy print can trigger a sell fill when its price is above the quoted ask. The execution size is the minimum of available quote size and available trade size.

This is a top-of-book execution model, not a full exchange-matching model. It preserves only what is needed for the accounting problem: quote placement, possible execution against the observed market, and the later mark-to-market effect of the resulting inventory. That is enough to separate immediate execution quality from later inventory revaluation.

4 PnL Decomposition and Reward

The central accounting point is that maker execution quality and inventory mark-to-market movement are distinct objects. Let m_{quote} denote the prevailing mid when the quote is placed, and let m_{exec} denote the mid when the fill is processed.

4.1 Maker PnL

Define the immediate maker quote edge as

$$\psi_{quote} = sd(m_{quote} - p), \tag{7}$$

where s is executed size, $d = +1$ for a buy fill and $d = -1$ for a sell fill, so sd is signed executed size. Define the fill-time fluctuation term as

$$fluc = sd(m_{exec} - m_{quote}). \tag{8}$$

Its positive and negative parts are

$$\psi_{fluc} = \max(fluc, 0), \qquad \phi_{fluc} = \min(fluc, 0). \tag{9}$$

Hence the total maker contribution is

$$\psi_{quote} + \psi_{fluc} + \phi_{fluc} = sd(m_{exec} - p). \tag{10}$$

By construction, $\psi_{quote} \geq 0$, $\psi_{fluc} \geq 0$, and $\phi_{fluc} \leq 0$ under this sign convention.

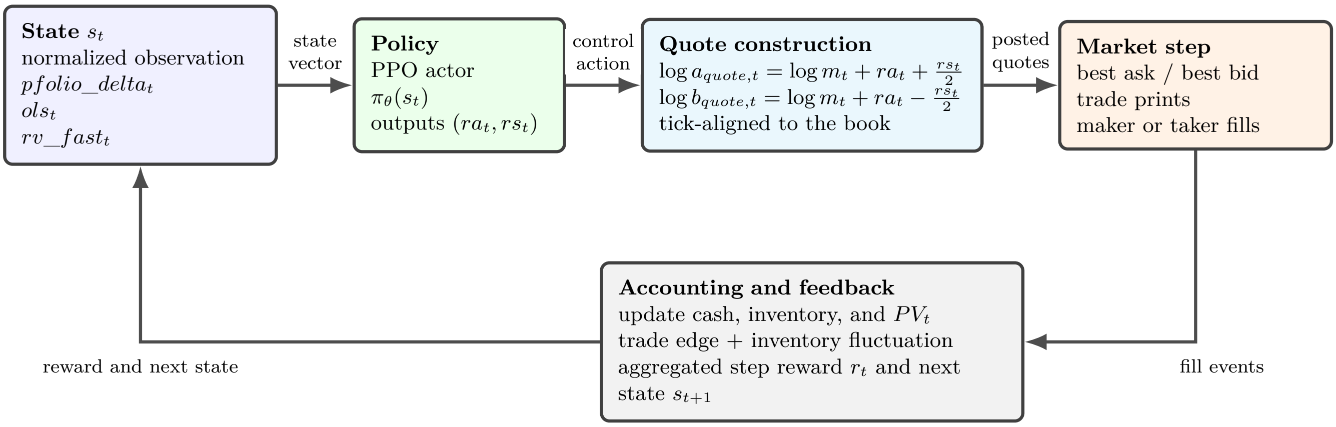

Figure 1: Perpetual control loop used in the paper. The agent observes normalized state, outputs (ra_t, rs_t), quotes both sides, and then receives fills, inventory fluctuation, and the aggregated step reward.

4.2 Taker PnL

If the strategy crosses the spread, the taker term is

$$\phi_{take} = sd(m_{exec} - p_{take}). \tag{11}$$

For the perpetuals used here, maker fees are zero and taker fees are five basis points, so the execution price already embeds the fee adjustment.

4.3 Inventory Mark-to-Market Fluctuation

For already-held inventory, the separate mark-to-market fluctuation term is

$$step\_fluctuations_t = q_{t^-}(m_t - m_{t^-}), \tag{12}$$

where q_{t^-} is the inventory before processing the new market row. This term is the pathwise revaluation of inventory that is already on the book. It is not a fill-quality term and is kept separate from $\psi_{fluc,t}$ and $\phi_{fluc,t}$ throughout the paper.

4.4 Step-Wise Portfolio Value Decomposition

For the single perpetual instrument considered here, $\psi_{quote,t}$, $\psi_{fluc,t}$, $\phi_{fluc,t}$, and $\phi_{take,t}$ denote the step-t realizations of the quantities defined above.

Writing the signed executed size as $x_t = s_t d_t$, a fill updates inventory and cash by

$$q_t = q_{t^-} + x_t, \qquad c_t = c_{t^-} - x_t p_t. \tag{13}$$

The immediate portfolio-value effect of the new fill, evaluated at the fill-processing mid, is therefore $x_t(m_{exec,t} - p_t)$. This is precisely the maker or taker trade-edge term defined above. Any later mark-to-market movement acts on inventory that is already being carried and therefore belongs to $step\_fluctuations_t$. The accounting split is exact because new execution and previously held inventory enter portfolio value through different state variables.

In this setting, the step-wise decomposition is

$$\Delta PV_t = trade\_edge\_net_t + step\_fluctuations_t, \tag{14}$$

Table 3: Reward constituents with non-zero terminal or scheduled weight in the current training setup.

Term	Sign	Interpretation
ψ_{quote}	+	Immediate maker quote edge.
ψ_{fluc}	+	Favorable fill-time fluctuation.
ϕ_{fluc}	−	Adverse fill-time fluctuation.
ϕ_{take}	−	Taker execution loss relative to the mid.
Risk aversion	mixed	Improvement in squared portfolio delta, $\delta_0^2 - \delta_1^2$.
Delta aversion	−	Direct penalty on squared end-of-step portfolio delta.
Spread aversion	−	Temporary penalty for quoting wider than the observed market spread.
Inventory alignment	mixed	Temporary action-sensitive term aligning reservation adjustment with portfolio delta.

with

$$trade\_edge\_net_t = \psi_{quote,t} + \psi_{fluc,t} + \phi_{fluc,t} + \phi_{take,t}. \tag{15}$$

This is the exact decomposition used in this exposition. An equivalent form is

$$\begin{aligned}\Delta PV_t = {} & \psi_{fluc,t} + \phi_{fluc,t} \\ & + (\psi_{quote,t} + \phi_{take,t} + step\_fluctuations_t).\end{aligned} \tag{16}$$

which is useful for discussing adverse selection at fill time.

The scalar reward is a weighted sum of interpretable constituents,

$$r_t = \sum_k w_k R_{k,t}. \tag{17}$$

The reported configuration uses terminal weights on ψ_{quote}, ψ_{fluc}, ϕ_{fluc}, ϕ_{take}, risk aversion, and delta aversion. Early training additionally applies temporary shaping through spread aversion and reservation-adjustment alignment, while the adverse fluctuation term ϕ_{fluc} is ramped up from zero to its terminal weight. Randomized initial inventory is enabled only in the early training phase and then switched off, so later updates are made closer to the evaluation distribution.

Because each reward constituent has a direct economic interpretation, the reward can be audited step by step rather than treated as a black-box scalar. A positive reward increment does not by itself imply a positive portfolio-value increment, and conversely a negative reward increment can coexist with favorable mark-to-market movement if earlier inventory happens to revalue in the strategy's favor. That distinction is useful for training and for evaluation. During learning, the reward can discourage unstable inventory growth before the policy has learned to generate consistently positive realized edge. During evaluation, the same decomposition makes it possible to report raw portfolio value and shaped reward side by side without conflating them.

5 Reinforcement Learning Formulation

The environment is an event-driven MDP. The state is the normalized observation vector. The action is the continuous pair (ra_t, rs_t). Time advances when the next order-book or trade event is processed, so the transition kernel is defined by the incoming market event, the quote-matching logic, and the portfolio accounting rules above rather than by a separate clock variable in the state. The reward is the weighted sum in Table 3.

PPO is used because it supports continuous control with clipped policy updates and does not require an analytic transition model. In the setup studied here, training uses five-minute episodes, a 1000 ms decision interval, and repeated passes over the same training horizon. Randomized initial inventory is present only in the early 20% of training and is then disabled. The adverse fill-fluctuation term ϕ_{fluc} is also started in a damped state and ramped toward its terminal penalty, which reduces early reward variance while the policy is still learning the quoting geometry. With a five-minute horizon, this yields 300 decision steps per episode and 60 rollout steps per PPO update when five updates are taken per episode. The calibration used below runs for 1000 episodes, for a total of 300,000 environment steps.

Two design choices are especially important for interpreting the learning problem. First, the observation intentionally compresses the market state to a small set of short-horizon signals: current portfolio exposure, a normalized spread observation, and a fast realized-volatility estimate. The agent is therefore not asked to reconstruct queue position, hidden liquidity, or deep-book shape. Instead, it learns whether a compact state is already sufficient to recover an inventory-aware quoting rule. This makes the economic interpretation of the learned behavior sharper. If the policy succeeds, it does so by reacting to a deliberately small amount of information rather than by exploiting a high-dimensional latent state representation.

Second, the two action coordinates play different economic roles and must be learned jointly. Changing ra_t primarily shifts fill pressure between the bid and ask, whereas changing rs_t changes the overall willingness to trade. A wider spread with no skew can reduce fill intensity but leave an existing inventory imbalance unresolved. A strong skew at too narrow a spread can liquidate inventory quickly but may do so at the cost of repeated adverse selection. PPO therefore has to learn a geometry of joint quote placement rather than a one-dimensional aggressiveness knob. The policy quality depends on whether the spread and skew heads become coordinated enough to reduce exposure without systematically giving away too much edge.

Table 4: Reference hyperparameters for the perpetual training exposition used in the article.

Item	Value
Training window	5 minutes per episode
Decision interval	1000 ms
Episode length	300 steps
Updates per episode	5
PPO rollout length *n_steps*	60
Episodes	1000
Total training steps	300,000
Parallel environments	12 on CPU, 18 on CUDA
γ	0.999
λ_{GAE}	0.95
Clip range	0.1
Policy initialization	continuous Gaussian, `log_std_init` $= -0.2$
Action bounds	$ra_t \in [-5, 5] \times 10^{-5}$, $rs_t \in [0, 5] \times 10^{-4}$
Training data source	simulated perpetual pipeline
Evaluation data sources	simulated and historical perpetual pipelines

The policy is parameterized in normalized action space and then mapped back to quote controls. This keeps optimization well scaled while preserving economic meaning. The reservation-adjustment range behaves as a local skew of the quoted center, and the reservation-spread range moves the strategy from near-market quoting to a defensive stance. The action space stays compact without losing price-space interpretation.

These hyperparameters reflect a short-horizon control objective rather than a directional forecasting objective. A five-minute episode is long enough for inventory to build up, be partially unwound, and create visible mark-to-market consequences, but short enough that the policy cannot rely on slow price drift. The high discount factor pushes the agent to respect the delayed cost of carrying exposure, while the small clip range prevents abrupt policy changes in a setting where a few adverse fills can otherwise dominate one batch of experience.

The weight design induces a clear set of priorities. Immediate quote edge rewards favorable passive execution, adverse fill fluctuation penalizes being picked off, and the risk-based terms suppress persistent inventory growth. The temporary shaping terms guide early exploration toward plausible quote placements. The training problem is therefore not to maximize fill count, but to learn when lower trading intensity is preferable.

6 Data

The data cover the six-month period from 2021-10-01 to 2022-03-31. For each market, the sample contains 182 daily perpetual files and the full top-of-book event stream. Spread is measured in market ticks, using \$0.5 for BTC and \$0.05 for ETH. This choice matters because the same dollar spread corresponds to very different tick counts across the two markets.

Table 5 shows three broad features of the six-month sample. First, ETH is more active at the top of book, producing about 61.63 million snapshots versus 44.38 million for BTC over the same 182-day span. Second, BTC is tighter in tick space: its median spread is 3 ticks and 45.02% of snapshots are one tick wide, whereas ETH has a median of 5 ticks and only 35.88% of snapshots are one tick wide. Third, ETH is more concentrated in intermediate spread states, especially 6–9 ticks, while BTC puts relatively more mass into the far upper tail.

This descriptive section is informative on its own because it characterizes the background fill environment across the full sample, rather than only on the shorter calibration or evaluation windows. The tail quantiles make the difference especially clear. BTC reaches 20 ticks by the 90th percentile and 55 ticks by the 99th percentile, while ETH reaches 16 and 41 ticks at the same two quantiles. BTC is therefore tighter at the center but still exhibits a thicker extreme right tail in the six-month sample, whereas ETH spends more of its time in medium-width spread regimes.

These differences matter directly for the control problem. A one-tick or few-tick widening action represents a different change in aggressiveness across the two markets because the central spread distributions are not the same. BTC spends more time in very tight states, so narrow quoting can still imply relatively frequent execution. ETH spends more time in medium-width states, so the same nominal action may move the policy into a much less active trading regime. Reporting both central quantiles and upper-tail mass is therefore important: the center of the distribution governs routine quoting conditions, while the right tail describes the broader-spread states in which inventory can become difficult to unwind quickly.

7 Experimental Design

7.1 Simulated Pipeline

The simulated study uses a synthetic perpetual market with a Heston-like mid-price process and a discrete Markov spread model. The spread model is calibrated separately for BTC and ETH from 250 ms last-quote samples. For BTC this yields about 11.53 million calibration snapshots and a dominant one-tick regime at 64.44%. For ETH it yields about 16.35 million snapshots, but the one-tick regime drops to 49.37%, already indicating a broader spread process. These shorter-horizon calibration numbers are consistent with the six-month descriptive picture in Table 5.

Table 6 shows why ETH cannot be treated as a scaled copy of BTC. BTC spends much more time in the tightest spread regime and also stays there longer once it arrives. ETH, by contrast, is more diffuse both in the unconditional distribution and in the transition dynamics. The updated ETH chain still uses six states for parsimony, but the representative tick levels shift to 1, 4, 7, 11, 16, and 31. This keeps the compact Markov construction while acknowledging that ETH spends substantially more time in broader-spread conditions than BTC.

The spread model is deliberately low dimensional. It captures how often the book is tight, how persistent tight states are, and how often the process moves into wider regimes. This keeps the BTC–ETH comparison interpretable because the main calibration differences remain transparent.

The training configuration also separates short-horizon order-flow and volatility priors by market instead of reusing one synthetic setting everywhere. One-second print-arrival rates are estimated on the 2021-10-03 17:49–18:09 window and then collapsed to a symmetric per-side simulator rate within each market for parsimony. The resulting market-specific priors are summarized in Table 7. The shared choice of $\mu = 0$ keeps the simulator centered on short-horizon inventory management rather than drift forecasting.

These numbers are pragmatic simulator priors rather than structural claims about the true data-generating process. The simulator also imposes an exchange-tick-aware minimum market spread so that early PPO updates do not collapse into unrealistically tight quoted books. The recalibrated BTC and ETH blocks were validated at the model-input level: the initial probabilities and each transition row normalize, and the spread sampler reproduces the intended state supports. The calibration is therefore treated as an improved input layer for the learning problem rather than as a separate model-selection exercise.

7.2 Historical Pipeline

For the historical experiments, the same cleaning logic is applied to BTC and ETH. Raw book-ticker and trade records are filtered to the selected currency. Timestamps are normalized from microseconds to milliseconds. Missing best quotes are forward-filled within symbol, and consecutive rows with no quote change are dropped. Trades are aggregated by timestamp, symbol, side, and price. If several prices occur within the same timestamp-side batch, the batch is split into sequential components across consecutive milliseconds so that the event stream remains ordered when fills are reconstructed. After the trade and quote streams are joined, best bid and ask are kept non-crossed, quote sizes are forward-filled when needed, and the volatility-index stream is joined backward in time. Small spillovers from adjacent calendar days are removed so that each daily sample keeps only rows whose normalized timestamp belongs to the intended processing date.

This procedure is meant to preserve the ordering needed for pathwise fill reconstruction while removing artifacts that would otherwise dominate the accounting. The goal is not to infer hidden exchange-side state, but to maintain an internally consistent top-of-book and trade stream on which quote placement, execution, and inventory revaluation can be evaluated in the same order as they would occur in the environment. That consistency is what makes the historical screen informative for the same decomposition that is used in simulation.

The analysis remains focused on the perpetual stream

Table 5: Six-month descriptive statistics for BTC and ETH perpetual spreads. The final six columns give the proportions of snapshots whose market spread falls into each tick bin.

Market	Tick [USD]	Days	Snapshots [M]	q50	q75	q90	q95	q99	1	2–5	6–9	10–13	14–20	21+
BTC	0.50	182	44.38	3	11	20	28	55	45.02%	13.91%	12.09%	9.60%	9.78%	9.60%
ETH	0.05	182	61.63	5	10	16	22	41	35.88%	17.46%	18.51%	12.82%	9.59%	5.74%

Table 6: Market-specific spread-state calibrations used by the current simulator. The persistence numbers are the diagonal entries of the one-step 250 ms transition matrix for the tightest and widest spread regimes.

Market	Sampled 250 ms snapshots	One-tick share	Representative spread states [ticks]	Tight-state persistence	Tail-state persistence
BTC	11.53M	64.44%	1, 4, 8, 11, 16, 29	0.756	0.493
ETH	16.35M	49.37%	1, 4, 7, 11, 16, 31	0.670	0.490

alone, but the ETH historical screen still serves as a useful robustness check on whether the accounting decomposition remains interpretable once the market changes. It does: the ETH baseline again loses mainly through inventory fluctuation rather than through immediate trade edge.

7.3 Evaluation Protocol and Baseline

The evaluation compares one PPO checkpoint per market with a zero-action baseline on both simulated and historical pipelines. Because the reservation-adjustment bounds are symmetric and the spread head is pivoted around the observed market spread, the baseline is not a no-op; it applies zero reservation skew while quoting near the prevailing market spread. The simulated screens summarize six rollouts with seeds 55–60 on a 46-minute window; in the simulated figures, each line is the per-step mean across seeds and the shaded band is ± 1 sample standard deviation. The historical screens use matched 46-minute windows with no randomized initial inventory.

The two evaluation pipelines play complementary roles. The simulated screens are the better tool for assessing whether the learned policy has a stable directional effect under repeated realizations of the calibrated market model. The historical screens are more diagnostic than statistical: they expose the same policy to one realized event path with no simulator averaging. Simulation therefore addresses the internal coherence of the learned control rule, while historical evaluation tests whether the same control logic still behaves as intended once it is exposed to an observed path with its own realized order flow and realized inventory shocks.

8 Results

Table 8 compares BTC and ETH side by side. The cross-market view matters because it shows that the same accounting decomposition and the same broad PPO behavior remain informative in two perpetual markets with visibly different spread structure.

8.1 BTC Results

BTC provides the clearest case in which PPO improves both mean terminal portfolio value and the shaped reward in simulation. Over the six 46-minute synthetic rollouts, the PPO checkpoint achieves a modestly positive mean ΔPV of 4.3 ± 10.2 USD, whereas zero-action averages -19.9 ± 40.4 USD. The mechanism is conservative inventory control rather than aggressive spread capture. PPO reduces mean absolute inventory from 0.200 ± 0.082 to 0.038 ± 0.004 contracts and turns cumulative trade edge positive (3.0 ± 0.7 versus -16.1 ± 0.6), but it does so by quoting much wider than the baseline (12.19 versus 5.77 USD) and accepting less than half as many maker fills on average (851 versus 1961).

That simulated gain is small relative to pathwise variance, so it should be read as directional evidence rather than a profitability claim. The important pattern is stable across seeds: PPO keeps inventory tighter, lowers fill count, and avoids the large negative trade-edge realizations of the aggressive baseline.

Figure 2 makes the simulated mechanism visible. The PPO trace stays close to flat inventory throughout the window, and both its cumulative trade edge and cumulative portfolio value remain near zero to mildly positive. The zero-action baseline trades much more frequently at a narrower quoted spread, but this produces persistently negative trade edge and much larger inventory excursions. In the simulated setting, wider quoting is therefore not just a cost; it is the main way the policy avoids adverse selection.

On historical BTC perpetual data, both policies are loss-making over the same 21:00–21:46 interval, but PPO still materially improves the final result: \$-62.0 versus \$-222.7 for zero-action. Table 8 and Figure 3 show why. Zero-action actually earns more cumulative trade edge (15.9 versus 3.1) and also achieves a higher shaped reward (20.6 versus 11.1), but it carries substantially larger inventory and suffers much larger inventory fluctuation (-238.6 versus -65.1). PPO sacrifices fill count and quotes much wider (\$6.65 versus \$1.84) to keep mean absolute inventory low (0.067 versus 0.189 contracts), and that reduction in inventory risk dominates the loss in immediate trade edge.

The BTC historical screen therefore marks the limit of what can be concluded from simulation alone. Positive average simulated trade edge does not guarantee positive realized historical PnL on a single path. What survives the move from synthetic to historical data is not profitability, but the structure of the control rule. PPO continues to reduce inventory exposure, and once that happens the terminal loss shrinks substantially even though the baseline occasionally records higher direct trade edge.

Table 7: Market-specific simulator priors used in the current training configuration. The measured one-second buy/sell print rates come from the real-data calibration window; the simulator then uses one symmetric per-side print probability within each market. The Heston-style tuples are $(\kappa, \theta, \xi, \rho, v_0)$ with $\mu = 0$ in both markets.

Market	Measured buy-print prob.	Measured sell-print prob.	Simulator per-side prob.	$(\kappa, \theta, \xi, \rho, v_0)$	s_0
BTC	0.6515	0.5483	0.6515	(5.4, 3.8, 5.3, 0.1, 0.24)	48147
ETH	0.3600	0.3832	0.3832	(6.5, 2.6, 4.1, 0.4, 0.3)	3426

Table 8: Cross-market evaluation summary for the PPO checkpoints and the zero-action baseline. Simulated values are mean ± standard deviation across six seeds. Historical values are single matched screens.

Market	Pipeline	Policy	ΔPV [USD]	Trade edge [USD]	Fluctuations [USD]	Reward	Mean $\|q\|$	Maker fills	Mean quote spread [USD]
BTC	Simulated	PPO	4.3 ± 10.2	3.0 ± 0.7	1.3 ± 10.2	6.8 ± 0.5	0.038 ± 0.004	851 ± 13	12.19 ± 0.05
BTC	Simulated	Zero-action	-19.9 ± 40.4	-16.1 ± 0.6	-3.7 ± 40.0	-3.7 ± 1.3	0.200 ± 0.082	1961 ± 16	5.77 ± 0.03
BTC	Historical	PPO	-62.0	3.1	-65.1	11.1	0.067	650	6.65
BTC	Historical	Zero-action	-222.7	15.9	-238.6	20.6	0.189	1049	1.84
ETH	Simulated	PPO	4.2 ± 4.3	3.1 ± 0.5	1.1 ± 4.7	2.2 ± 2.2	0.330 ± 0.078	248 ± 18	1.34 ± 0.01
ETH	Simulated	Zero-action	-14.5 ± 41.0	-14.4 ± 1.2	0.0 ± 40.3	-114.2 ± 170.3	1.463 ± 0.964	1364 ± 26	0.48 ± 0.00
ETH	Historical	PPO	-13.5	-1.0	-12.4	-13.6	0.564	356	1.06
ETH	Historical	Zero-action	-218.6	-1.3	-217.3	-2851.2	8.127	763	0.46

That persistence across pipelines is one of the stronger arguments that the learned behavior is not merely a simulator artifact.

8.2 ETH Results

ETH shows that the positive simulated PPO-versus-baseline gap is not unique to BTC. Over six ETH simulated seeds, PPO attains 4.2 ± 4.3 USD while zero-action averages -14.5 ± 41.0 USD. The mechanism again runs through much tighter inventory control: mean absolute inventory drops from 1.463 ± 0.964 to 0.330 ± 0.078, maker fills fall from about 1364 to 248, and the quoted spread widens from roughly \$0.48 to \$1.34. The ETH simulated screen is therefore qualitatively consistent with BTC, but it operates at a lower quote-spread scale and with a much smaller fill count, which is exactly what one would expect from the market-specific calibration in Tables 6 and 7.

Figure 4 shows that the ETH simulated PPO path stays much closer to flat inventory than zero-action. Unlike BTC, where PPO also improves the mean shaped reward substantially, the ETH simulated reward remains more fragile (2.2 ± 2.2 versus -114.2 ± 170.3 for zero-action). The reason is scale mismatch between the reward constituents and the more frequent inventory shocks in the baseline. Even so, the ranking by terminal portfolio value is clear: once inventory is kept under control, the policy avoids the extreme downside realizations that dominate the ETH zero-action seeds.

The ETH historical screen is even more informative. PPO ends at -13.5 USD, while zero-action loses -218.6 USD. Here the immediate trade-edge terms are nearly identical (-1.0 versus -1.3), so the difference comes almost entirely from inventory fluctuation (-12.4 versus -217.3). Put differently, ETH historical data produce a near-pure demonstration of the decomposition in (14): the policies do not differ much in direct trade edge, but they differ enormously in the pathwise mark-to-market cost of the inventory they carry.

Figure 5 makes that interpretation visible. Zero-action spends long stretches with large directional exposure, which is reflected in the very high mean absolute inventory of 8.127 contracts. PPO still loses money, but it constrains mean absolute inventory to 0.564 and quotes roughly twice as wide as the baseline. The resulting portfolio path is much less volatile and finishes far closer to flat PnL. This is a stronger within-market inventory-control result than the BTC historical screen, even though direct cross-market comparisons of inventory in contracts should be interpreted with care.

ETH is especially useful because the broader-spread environment is less forgiving to passive narrow quoting. A baseline that stays close to the prevailing market without active inventory-sensitive skew control can accumulate very large exposure before it has a chance to unwind. PPO does not eliminate losses, but it changes their source. The residual loss that remains under PPO is small enough to be interpreted mainly as ordinary pathwise revaluation of a controlled position, rather than as the consequence of runaway inventory accumulation.

8.3 Cross-Market Comparison

The cross-market comparison yields three practical observations. First, the same qualitative PPO behavior survives the move from BTC to ETH: wider quoting, materially fewer maker fills, much tighter inventory, and a better terminal portfolio value than zero-action. Second, the ETH calibration makes the market-specific nature of the simulator impossible to ignore. ETH has a broader spread-state distribution, lower one-second observed print rates, and different short-horizon volatility priors, so a BTC-tuned simulator would misstate the fill environment even before training begins. Third, the ETH historical screen strengthens the interpretation of the learned policy as a conservative inventory filter. On BTC historical data, PPO gives up direct trade edge to limit fluctuation. On ETH historical data, the difference in trade edge is almost zero, so the entire performance gap is explained by lower inventory fluctuation.

This cross-market interpretation is consistent with the descriptive data section. BTC is tighter at the center, so the baseline can trade aggressively and still remain active almost continuously. ETH spends more time in intermediate spread regimes, so narrow quoting becomes a comparatively stronger bet on execution. That differ-

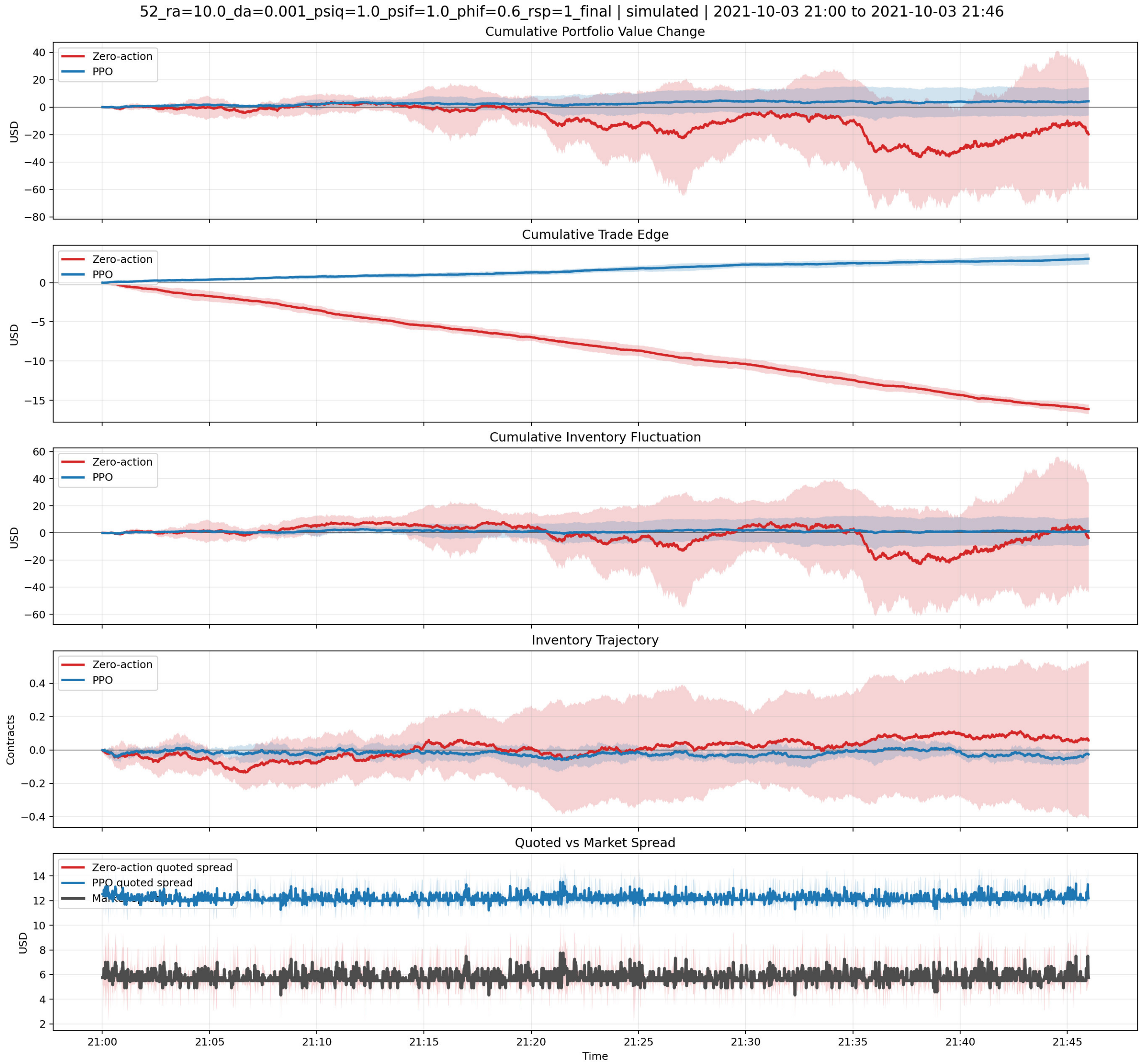

Figure 2: Simulated evaluation summary for the stored checkpoint over 2021-10-03 21:00–21:46. Each line is the mean across seeds 55–60 and the shaded band is ± 1 sample standard deviation. The five panels show cumulative portfolio value, cumulative trade edge, cumulative inventory fluctuation, inventory trajectory, and quoted versus market spread. PPO quotes materially wider and trades less often than zero-action, but it keeps inventory much closer to flat and finishes with a higher average portfolio value.

ence helps explain why the ETH baseline produces such large historical inventory. The environment is not merely a rescaled BTC problem; the same control action changes trading intensity differently once the spread distribution and print-arrival process shift.

The comparison also clarifies what should and should not be treated as market-invariant. The invariant part is the qualitative control logic: the learned policy widens quotes, reduces fill count, narrows the inventory distribution, and improves terminal portfolio value relative to zero-action. The non-invariant part is the numerical scale on which this happens. BTC and ETH differ in spread levels, fill intensity, and inventory measured in contracts, so raw numbers are not directly interchangeable. What can be compared across markets is the causal pattern by which lower trading aggressiveness leads to lower inventory fluctuation.

Table 9 compresses the same comparison into a few operational contrasts. This summary is useful because the cross-market argument depends on several linked differences at once: the center of the spread distribution, the persistence of tight states, the print-arrival environment, and the way baseline inventory behaves once the policy becomes too aggressive.

8.4 Reward Versus Portfolio Value

The combined BTC and ETH evidence is most informative when shaped reward and terminal portfolio value are read together rather than treated as interchangeable criteria. In BTC simulation, PPO improves both objects at the same time, which is the cleanest case for the intended learning mechanism. In BTC historical data, the ranking diverges: zero-action records higher trade edge and

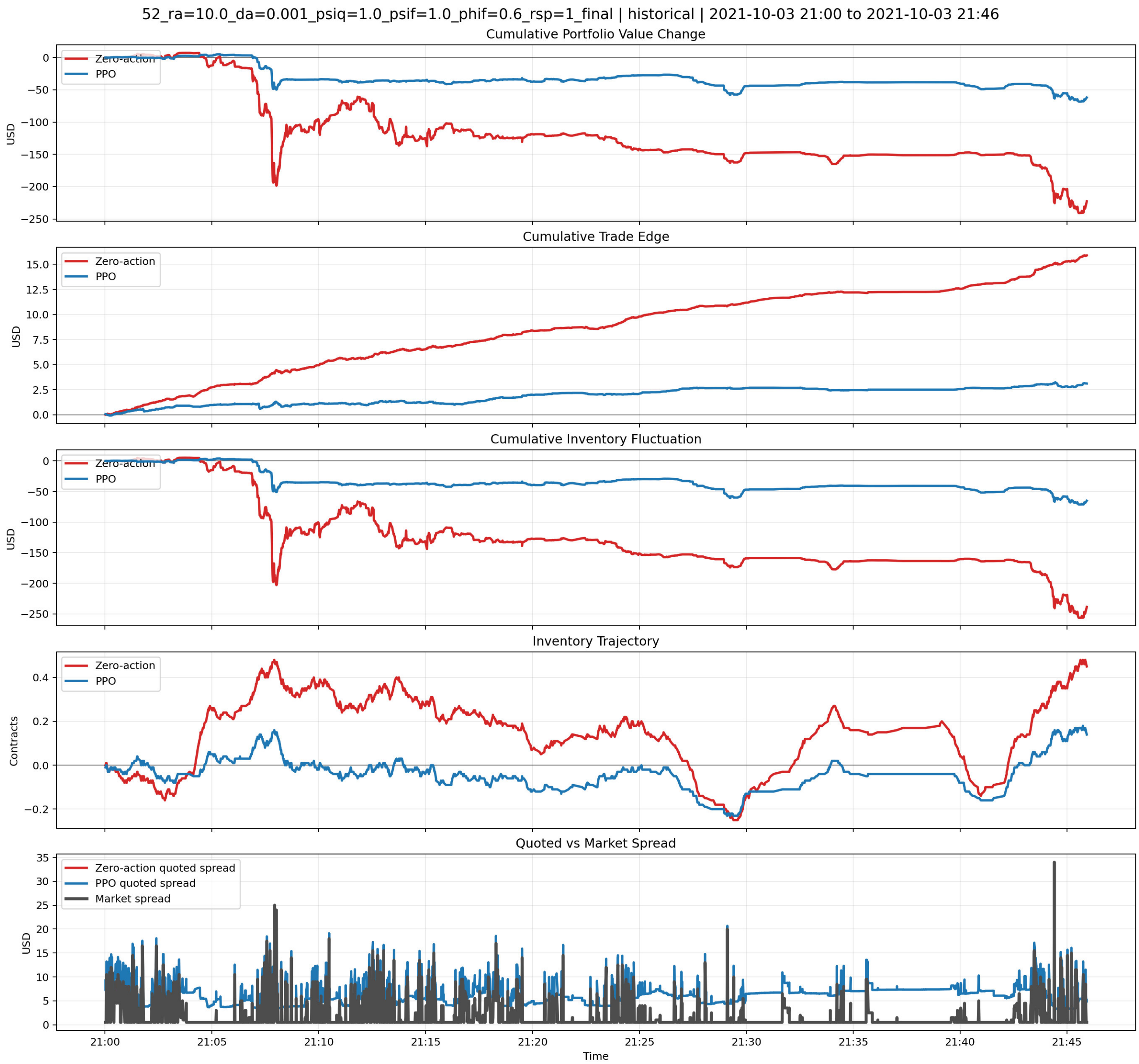

Figure 3: Historical evaluation summary for the stored checkpoint over 2021-10-03 21:00–21:46. The same five panels are shown as in the simulated figure. Zero-action earns higher cumulative trade edge, but much larger inventory swings dominate and produce a substantially worse final portfolio value. PPO loses less by quoting wider and containing directional exposure.

higher cumulative reward, yet still finishes with a much worse terminal portfolio value because inventory fluctuation overwhelms the gain in immediate execution terms. ETH historical data sharpen the same point even further because the trade-edge terms are nearly equal while the portfolio-value gap remains very large.

This divergence is not a defect of the decomposition. It reflects the fact that shaped reward is a local training signal whereas terminal portfolio value is an aggregate economic outcome. A policy can improve local reward components for long stretches and still finish with worse PnL if it repeatedly carries too much inventory into unfavorable price moves. Conversely, a policy that sacrifices some immediate trade edge can finish with a better terminal result if it keeps exposure small enough that inventory revaluation never dominates the path.

Evaluation therefore needs two views. Terminal portfolio value answers which policy ends in a better position. The decomposition explains why. Here, PPO usually benefits from tighter inventory control and lower exposure to adverse fill sequences.

9 Discussion

Two points matter most. First, fill-time fluctuation and inventory revaluation are different objects. $\psi_{fluc} + \phi_{fluc}$ measures the mid-price move between quote placement and fill, while $step\_fluctuations_t$ measures the revaluation of inventory already being held. Keeping them separate is necessary for interpreting why a policy wins or loses.

Second, the training schedule on ϕ_{fluc} acts as variance control. Early in training, adverse fills can dominate gradients before the policy learns basic skew and width behavior. Damping that term early does not change the evaluation accounting; it makes learning less noisy.

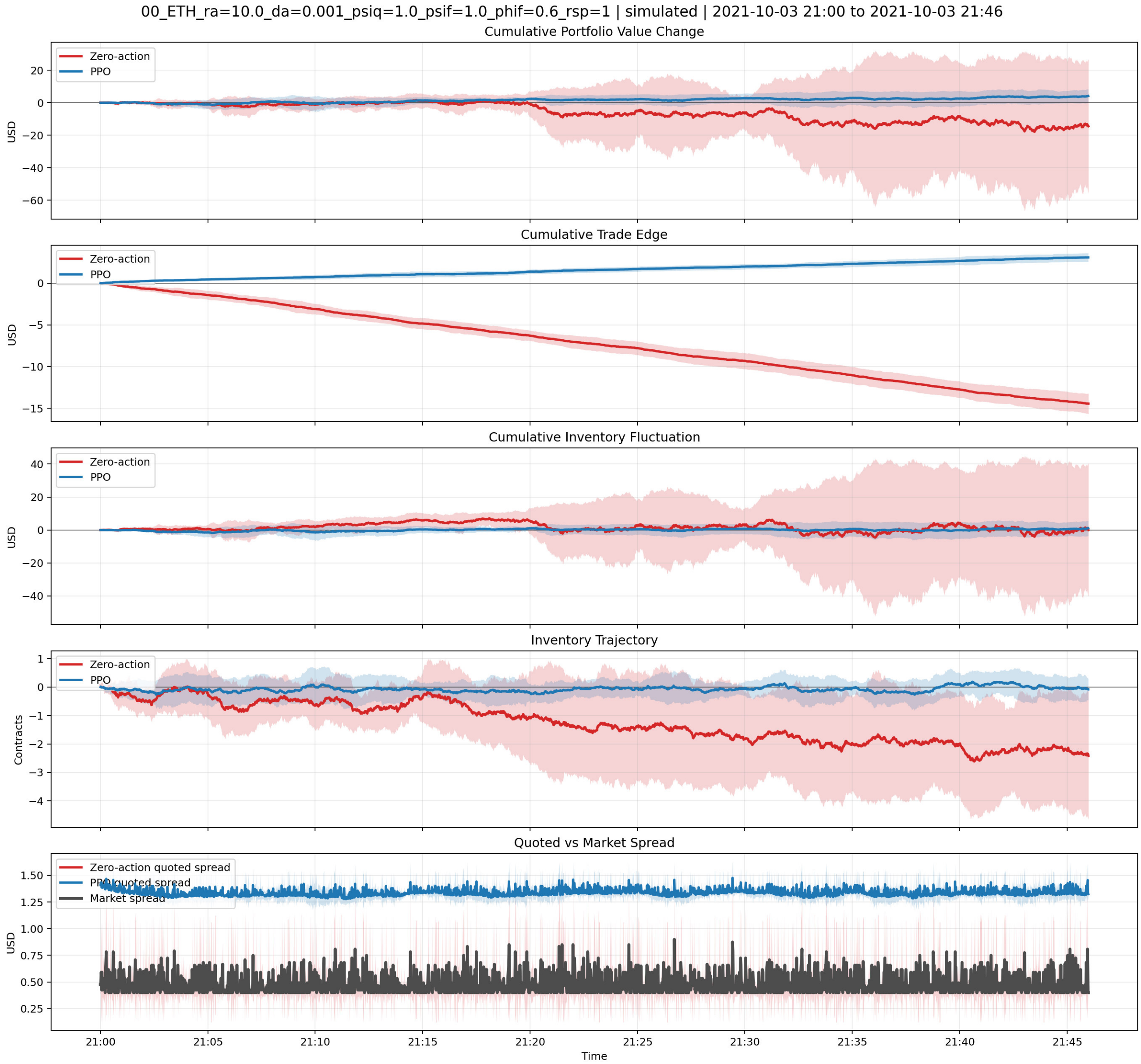

Figure 4: ETH simulated evaluation summary. Each line is the mean across six seeds and the shaded band is ± 1 sample standard deviation. The same five panels are shown as in the BTC simulated screen. PPO quotes materially wider than zero-action and trades much less often, but the inventory path is far tighter and the final average portfolio value is higher.

Reward and terminal portfolio value can still diverge. In BTC historical data, zero-action earns more immediate trade edge yet finishes with worse portfolio value because inventory fluctuation dominates. In ETH historical data, trade edge is nearly the same across PPO and zero-action, but the baseline still loses heavily through large inventory excursions. The learned policy is therefore better interpreted as an inventory filter than as a robust profit engine.

The evidence also remains limited. ETH requires different spread, arrival, and volatility calibration than BTC, so market-specific simulation is part of the problem definition. The historical windows are still narrow, and the spread model compresses the upper tail into one state. Broader historical evaluation and a finer tail model are the next obvious extensions.

10 Implications for Inventory-Aware Market Making

The results support three practical implications. Explicit accounting improves interpretability because terminal portfolio value can be linked to trade edge, fill-time fluctuation, and inventory revaluation instead of being reported as one aggregate number. Compact controls are still informative when they map directly to economic objects such as skew and quote width. Market-specific calibration is part of the model, not a minor implementation detail, because BTC and ETH respond differently to the same nominal quoting rule.

The same decomposition is also useful for diagnosis. Poor performance can come from bad passive execution, excessive taker activity, oversized inventory, or a simulator mismatch. That makes the framework useful even when profitability remains limited, because it shows

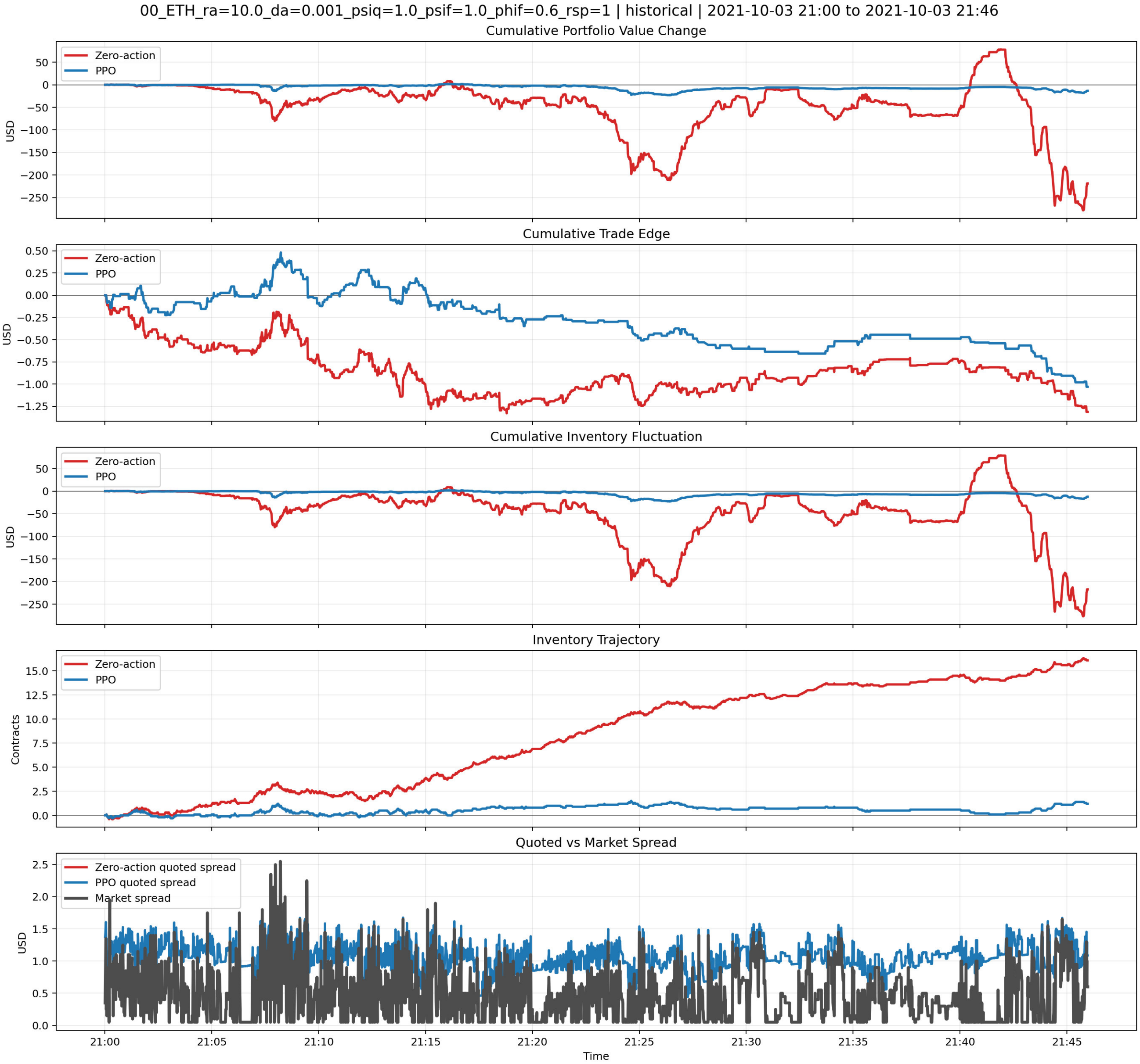

Figure 5: ETH historical evaluation summary. The same five panels are shown as in the other article figures. The key visual difference is that zero-action accumulates very large directional exposure, whereas PPO keeps inventory far tighter and therefore avoids most of the historical mark-to-market loss.

where the next iteration should change the state, reward, or calibration.

11 Conclusion

A perpetual market-making problem can be written as a compact RL control task in which PPO sets quote skew and width while the environment tracks trade edge and inventory fluctuation separately. Across BTC and ETH, the learned policy improves results mainly by reducing inventory risk, not by maximizing immediate spread capture. That pattern appears in both simulated and historical data, although the historical evidence remains limited and robust profitability claims remain premature. The main value of the framework is interpretability: it makes it clear when the policy is controlling exposure, when it is earning execution edge, and when calibration differences across markets matter. Broader historical testing and finer spread-tail modeling are the next natural extensions.

References

[1] Thomas Ho and Hans R. Stoll. Optimal dealer pricing under transactions and return uncertainty. *Journal of Financial Economics*, 9(1):47–73, 1981.

[2] Maureen O'Hara. *Market Microstructure Theory*. Blackwell, 1995.

[3] Marco Avellaneda and Sasha Stoikov. High-frequency trading in a limit order book. *Quantitative Finance*, 8(3):217–224, 2008.

[4] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, second edition, 2018.

Table 9: Qualitative comparison of BTC and ETH perpetual environments and their control implications.

Aspect	BTC	ETH	Implication for control
Central spread	Tighter center; median 3 ticks and 45.02% one-tick states.	Broader center; median 5 ticks and 35.88% one-tick states.	The same nominal widening action changes trading intensity differently across markets.
Spread dynamics	Tight states are more persistent in the simulated spread chain.	Spread states are more diffuse and less concentrated in the tightest regime.	ETH requires a more cautious calibration of quoting aggressiveness.
Observed print arrivals	Higher one-second per-side simulator rate.	Lower one-second per-side simulator rate.	Passive narrow quoting is less uniformly rewarded in ETH.
Historical baseline behavior	Zero-action accumulates meaningful but still moderate inventory.	Zero-action accumulates very large inventory on the historical screen.	Inventory-sensitive skew becomes especially important in ETH.
PPO effect	Lowers inventory and improves terminal portfolio value.	Same qualitative effect, with a stronger historical inventory-control contrast.	The qualitative control logic transfers even though the numerical scales differ.

[5] Yuriy Nevmyvaka, Yi Feng, and Michael Kearns. Reinforcement learning for optimized trade execution. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 673–680, 2006.

[6] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[7] Alvaro Cartea, Sebastian Jaimungal, and Jose Penalva. *Algorithmic and High-Frequency Trading.* Cambridge University Press, 2015.

[8] Thomas Spooner, John Fearnley, Rahul Savani, and Andreas Koukorinis. Market making via reinforcement learning. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 434–442, 2018.

[9] Olivier Guéant and Iuliia Manziuk. Deep reinforcement learning for market making in corporate bonds: Beating the curse of dimensionality. *Applied Mathematical Finance*, 26(5):387–452, 2019.

[10] Thomas Spooner and Rahul Savani. Robust market making via adversarial reinforcement learning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pages 4541–4547, 2020.

[11] Bruno Gasperov and Zvonko Kostanjcar. Market making with signals through deep reinforcement learning. *IEEE Access*, 9:61611–61622, 2021.

[12] Bruno Gasperov and Zvonko Kostanjcar. Deep reinforcement learning for market making under a Hawkes process-based limit order book model. *IEEE Control Systems Letters*, 6:2480–2485, 2022.

[13] Abbas Haider, Glenn I. Hawe, Hui Wang, and Bryan Scotney. Multi-asset market making via multi-task deep reinforcement learning. In *Machine Learning and Knowledge Discovery in Databases. Research Track*, pages 418–433. Springer, 2022.

FFA Working Paper Series

2019

1. Milan Fičura: Forecasting Foreign Exchange Rate Movements with k-Nearest-Neighbor, Ridge Regression and Feed-Forward Neural Networks.

2020

1. Jiří Witzany: Stressing of Migration Matrices for IFRS 9 and ICAAP Calculations.
2. Matěj Maivald, Petr Teplý: The impact of low interest rates on banks' non-performing loans.
3. Karel Janda, Binyi Zhang: The impact of renewable energy and technology innovation on Chinese carbon dioxide emissions.
4. Jiří Witzany, Anastasiia Kozina: Recovery process optimization using survival regression.

2021

1. Karel Janda, Oleg Kravtsov: Banking Supervision and Risk-Adjusted Performance in the Host Country Environment.
2. Jakub Drahokoupil: Variance Gamma process in the option pricing model.
3. Jiří Witzany, Oľga Pastiranová: IFRS 9 and its behaviour in the cycle: The evidence on EU Countries.

2022

1. Karel Janda, Ladislav Kristoufek, Binyi Zhang: Return and volatility spillovers between Chinese and U.S. Clean Energy Related Stocks: Evidence from VAR-MGARCH estimations.
2. Lukáš Fiala: Modelling of mortgage debt's determinants: The case of the Czech Republic.
3. Oľga Jakubíková: Profit smoothing of European banks under IFRS 9.
4. Milan Fičura, Jiří Witzany: Determinants of NMD Pass-Through Rates in Eurozone Countries.
5. Sophio Togonidze, Evžen Kočenda: Macroeconomic Responses of Emerging Market Economies to Oil Price Shocks: An Analysis by Region and Resource Profile.
6. Jakub Drahokoupil: Application of the XGBoost algorithm and Bayesian optimization for the Bitcoin price prediction during the COVID-19 period.

7. Karel Janda, Binyi Zhang: Econometric estimation of green bond premiums in the Chinese market.
8. Shahriyar Aliyev, Evžen Kočenda: ECB monetary policy and commodity prices.
9. Sophio Togonidze, Evžen Kočenda: Macroeconomic implications of oil-price shocks to emerging economies: a Markov regime-switching approach.

2023

1. Jiří Witzany, Milan Fičura: Machine Learning Applications for the Valuation of Options on Non-liquid Option Markets.
2. David Mazáček, Jiří Panoš: Key determinants of new residential real estate prices in Prague.
3. Milan Fičura: Impact of size and volume on cryptocurrency momentum and reversal.
4. Masood Tadi, Jiří Witzany: Copula-Based Trading of Cointegrated Cryptocurrency Pairs.
5. Pavel Jankulár, Zdeněk Tůma: Changes to Bank Capital Ratios and their Drivers Prior Integrating Flood Risk into House Price Models Using Expected Discounted Loss: Evidence from the Czech Republic in 2024
6. Teona Shugliashvili: The words have power: the impact of news on exchange rates.
7. Jiří Witzany, Milan Fičura: A Comparison of Neural Networks and Bayesian MCMC for the Heston Model Estimation *(Forget Statistics – Machine Learning is Sufficient!)*

2026

1. Marek Folprecht: Measuring Flood Risk in Czechia with Stress Testing and a Gumbel copula-based VaR
2. Marek Folprecht: Integrating Flood Risk into House Price Models Using Expected Discounted Loss: Evidence from the Czech Republic in 2024
3. Tadi M., Fičura M., and Witzany J. (2026). Natural Gas Storage Valuation Using Deep Reinforcement Learning.
4. Čekal M. (2026). Reinforcement Learning for Perpetual Futures Market Making.

All papers can be downloaded at: wp.ffu.vse.cz

Contact e-mail: ffawp@vse.cz

Faculty of Finance and Accounting, Prague University of Economics and Business, 2023

Winston Churchill Sq. 1938/4, CZ-13067 Prague 3, Czech Republic, ffu.vse.cz