ALLOCFLOW: END-TO-END PORTFOLIO TRADING ACTION GENERATION FROM RAW MARKET DATA

Anonymous authors
Paper under double-blind review

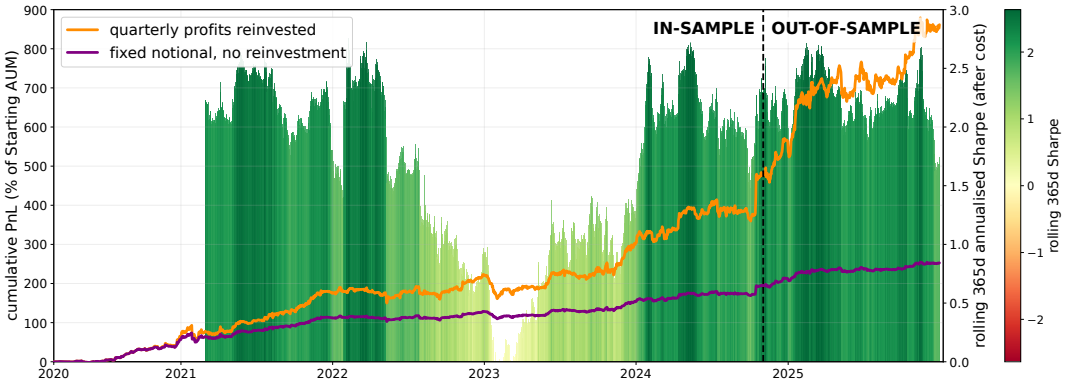


Figure 1: **After-cost trading performance** from 01 Jan 2020 to 31 Dec 2025. The dashed line marks the out-of-sample start. Purple shows cumulative PnL on fixed notional, orange shows a view with profits reinvested quarterly, and color indicates rolling 365-day annualized Sharpe.

ABSTRACT

Real-world portfolio management requires converting noisy market observations into coordinated, simultaneous allocations across multiple assets under shared capital constraints. Advances in forecasting models and alpha discovery help identify opportunities. However, traditional pipelines often decouple signal forecasting from portfolio construction, potentially misaligning learning with allocation and execution objectives. We present AllocFlow, an end-to-end flow-matching framework that directly generates joint portfolio allocations from raw multi-asset market histories. To capture dependencies among portfolio positions, AllocFlow employs a cross-asset flow Transformer conditioned on temporal market representations to generate continuous, signed position weights across all instruments jointly. To address the sample inefficiency and instability of learning continuous allocation policies from scratch, we establish a two-stage training scheme: strategy-supervised imitation shapes a structured distribution of candidate portfolios, followed by group-relative reinforcement learning (RL) that refines joint decisions using portfolio-level financial rewards and a frozen policy reference. The primary value of imitation is to provide a coherent starting region for optimization, rather than a standalone performance target. In retrospective cryptocurrency evaluation after modeling trading fees, AllocFlow achieves an annualized arithmetic net return of 51.38% and Calmar 4.043, establishing the value of our design.

1 INTRODUCTION

Algorithmic trading has evolved from explicit rules and statistical portfolio construction toward learned signals, policies, and market representations. Classical portfolio construction formalizes the balance between expected return and risk (Markowitz, 1952), while modern machine learning extracts richer predictive signals (Gu et al., 2020). Recent financial foundation models learn reusable representations of price histories (Shi et al., 2025), and language-model agents and reinforcement learning (RL) increasingly automate alpha discovery (Yu et al., 2023; Tang et al., 2025; Li et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2025; Chen et al., 2026). Yet an improved forecast or alpha does not by itself determine a better trade. Its economic value depends on how it is converted into positions, combined with other opportunities, and implemented through rebalancing. Optimizing these intermediate signals separately can therefore leave a gap between what a model learns and what a trading account needs to achieve (Elmachtoub & Grigas, 2022).

An account must choose *how much to allocate to each instrument, in which direction*, under shared capital and exposure considerations. A correct directional forecast can still lead to a poor decision if the position is oversized, duplicates another exposure, or costs too much to change. Conversely, coordinated long and short positions can exploit relative opportunities without relying on a large directional market bet. Each position must therefore be considered in relation to the others. Direct trading and portfolio policies have a substantial history (Moody & Saffell, 2001; Jiang et al., 2017; Li et al., 2023). We investigate how to realize this joint decision through an end-to-end generative model and a training procedure that produces profitable portfolio behavior.

Our goal is to make joint trading decisions for the entire portfolio directly from raw market observations. To realize this challenging goal, the action representation must express coordinated positions, while training must discover profitable allocations within a large continuous space. Historical observations provide noisy outcomes and no unique optimal portfolio label. Temporal pretraining supplies useful market representations (Zhou et al., 2023; Jin et al., 2024; Shi et al., 2025), but does not itself determine a trading policy. RL can optimize financial outcomes directly (Moody & Saffell, 2001; Deng et al., 2017; Zhang et al., 2020); however, uninformed exploration must discover meaningful position relationships and sizing while learning from noisy returns. Trading costs can further favor inactivity when candidate portfolios have insufficient expected benefit. Financial RL frameworks have tried to address parts of these challenges (Liu et al., 2020; Jiang et al., 2024), and generative policies offer expressive continuous-action models (Chi et al., 2023). The remaining design question is how to combine an appropriate joint action model with an informative starting point and effective portfolio-level optimization.

We propose AllocFlow, a cross-asset flow Transformer that generates continuous, signed position weights conditioned on pretrained temporal market representations. Together with the model, we designed a two-stage training recipe that separates acquiring trading structure from improving portfolio outcomes. We train the randomly initialized action head through imitation learning (IL) of an alpha-based strategy, teaching relative position preferences, direction, and scale to yield a structured distribution of candidate portfolios. This gives RL a coherent starting region for search without making the teacher's standalone performance the final objective. Group-relative RL then compares candidate portfolios under shared market outcomes and improves their financial rewards, while a frozen policy reference guides the updates. The encoder supplies market information; the action module learns how to convert that information into coordinated allocations and refine them through reward feedback.

We evaluate AllocFlow on hourly cryptocurrency data with modeled trading fees. The complete model reaches 51.38% annualized arithmetic net return, 2.192 Sharpe, and 4.043 Calmar in the reported retrospective test. More importantly for our design claim, the controlled IL-to-RL comparison improves both return and drawdown, while the action-head and encoder ablations identify which components make that refinement effective.

Our contributions are threefold: (i) **Joint action generation from market histories.** A frozen financial encoder compresses each asset's history, and a cross-asset flow Transformer produces the signed target book in one decision; (ii) **Strategy-guided refinement.** Imitation supplies coherent portfolio proposals, then group-relative RL with a frozen reference improves their portfolio-level reward; (iii) **After-fee retrospective evidence.** The reported policy attains 51.38% annualized arithmetic net return and 4.043 Calmar, while head, encoder, teacher, and reward ablations examine the design choices behind this result.

2 Related Works

From market information to decisions. Forecasting and financial foundation models learn market representations (Gu et al., 2020; Shi et al., 2025; Liu et al., 2023; Song et al., 2026a), while AlphaGen, AlphaSAGE, AlphaAgent, and R&D-Agent-Quant mine predictive signals (Yu et al.,

2023; Chen et al., 2026; Tang et al., 2025; Li et al., 2025). Their outputs can inform a portfolio, but require a further rule to choose signed positions. AllocFlow instead learns the complete joint allocation and evaluates its realized outcome.

Direct portfolio learning. Trading and portfolio RL already map observations to actions (Moody & Saffell, 2001; Jiang et al., 2017; Li et al., 2023), and imitation-assisted portfolio learning is established (Dong & Zheng, 2024). Recent approaches combine learned information with mean-variance allocation, executable programs, Black-Litterman scenarios, differentiable sparse optimization, or ranked allocation (Yu et al., 2024; Song et al., 2026b; Soun et al., 2026; Jeon et al., 2026; Zhang et al., 2025b). Thus, portfolio-level learning is prior art. Our contribution is a flow-generated continuous joint action and a tested strategy-guided refinement procedure. The operative difference is what the policy emits: an alpha score, an optimizer input, a trading program, and a vector of executable target weights define different decision boundaries. Our experiments therefore compare complete deployed portfolios while using within-model controls to examine the action generator itself.

Generative action refinement. Flow matching transports noise to continuous samples (Lipman et al., 2022); diffusion and flow policies apply generative models to actions (Chi et al., 2023; Black et al., 2024). DPPO, Flow-GRPO, and ReinFlow demonstrate reinforcement updates for such policies (Ren et al., 2025; Liu et al., 2025; Zhang et al., 2025a). These methods are widely adopted in continuous-space dense generation tasks, e.g., image generation and robotic action generation. We adapt this connection to portfolio trading actions, using imitation for initialization and a frozen reference during reward refinement. Appendix D discusses execution-aware trading and other generative finance work.

3 METHOD

A market representation can provide a compact market context without deciding how it should be traded alongside other assets. AllocFlow aims to generate a complete target portfolio from raw market histories. As Figure 2 shows, a frozen encoder first condenses those histories, imitation gives the joint action generator a useful starting distribution, and reward-guided refinement improves the resulting portfolio decisions.

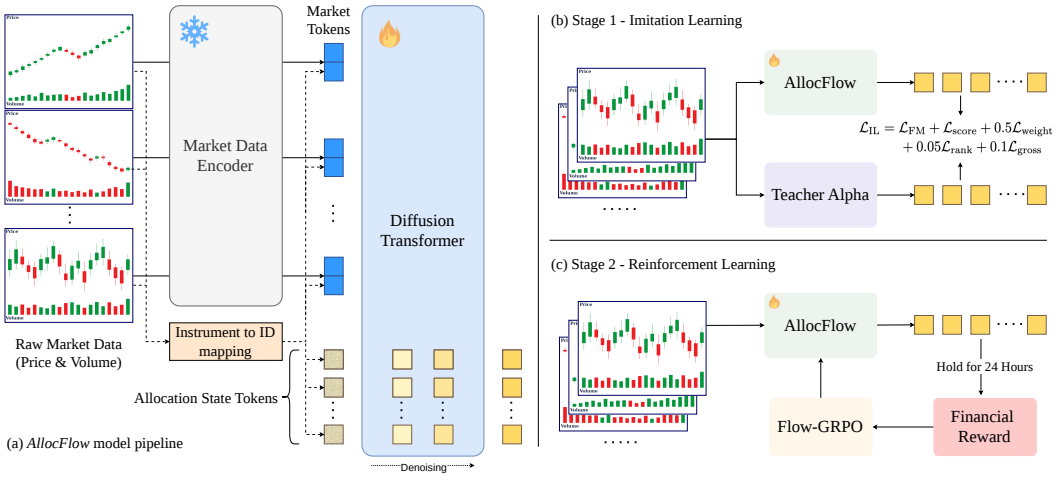


Figure 2: **AllocFlow from market histories to portfolio decisions.** (a) Frozen Kronos yields a coarse/fine market-token pair per asset; the instrument-ID mapping connects each pair to its allocation-state token. The DiT-style flow head generates a joint target book. (b) Causal strategy allocations initialize the head through imitation. (c) A 24-hour portfolio reward and a frozen IL reference guide Flow-GRPO refinement.

3.1 GENERATING A PORTFOLIO FROM COMPACT MARKET CONTEXT

Portfolio weights should be decided together: a position's value depends on the other exposures and the neutrality of the complete book. At market time t, let $\mathcal{U}_t$ contain N_t eligible assets and let $X_t \in \mathbb{R}^{N_t \times L \times C}$ contain their available histories. The policy produces signed target weights $w_t \in \mathbb{R}^{N_t}$ relative to a fixed account notional. Positive and negative weights denote long and short positions. Execution rounds the target to $\widehat{w}_t$ and applies it after one bar. In the fixed-notional ledger, the trading action is simply $\Delta w_{t+1} = \widehat{w}_t - \widehat{w}_{t-1}$, with zero weight assigned to names outside each target's asset list. Thus the model learns the desired book, while the ledger obtains orders and fees from changes to that book (Appendix C.1).

Direct attention over every bar of every asset would make the action head's context long. We therefore freeze Kronos (Shi et al., 2025) and use its temporal representation rather than its next-bar forecast. For each asset, normalized OHLCV and derived amount become coarse and fine token IDs. A single forward pass provides two hidden states before their logit heads: the final normalized coarse-decoder context and the fine dependency-layer output conditioned on that context and a coarse-token query. We mean-pool the last eight steps of each stream, obtaining one coarse and one fine market token per asset. This path neither samples a next-bar token nor feeds a predicted price to the policy (Appendix B.1).

We interleave the N_t token pairs and append their mean as a shared market token, yielding $2N_t + 1$ tokens for the DiT-style flow head (Peebles & Xie, 2022). The instrument-to-ID mapping associates each pair with the allocation-state token whose output is that asset's target weight (Figure 2a). The head can therefore compare opportunities across assets while generating their positions together. At inference, both token sets are assembled for the current $\mathcal{U}_t$: membership, size, and order may change for instruments with mapped IDs, without fixed asset-specific output slots. "End-to-end" refers to this bars-to-weights inference path; Kronos remains frozen. The frozen-encoder comparison in Table 5 examines the representation choice.

The allocation-state tokens form a continuous action vector, so a flow model can generate alternative complete books instead of assigning positions independently. Starting from $z_0 \sim \mathcal{N}(0, I)$, the market-conditioned velocity field evolves the vector according to $\mathrm{d}z_\tau/\mathrm{d}\tau = v_\theta(z_\tau, \tau; X_t)$. Flow time $\tau \in [0, 1]$ indexes generation and is distinct from market time t. Each update changes the whole vector; four Euler steps are used at inference. The final state is decoded as

$$w_t = \frac{1}{c} P_t z_1, \qquad P_t = I - \frac{\mathbf{1}\mathbf{1}^\top}{N_t}, \qquad c = 10. \tag{1}$$

Projection removes the common component, imposing $\mathbf{1}^\top w_t = 0$ before execution rounding, while the model still chooses relative directions, sizes, and variable gross exposure $\|w_t\|_1$. There is no additional hard gross cap in the evaluated decoder; dollar neutrality alone does not imply beta neutrality or bounded risk (Appendix A.2). Current holdings enter the trading ledger, not the policy input. The action-head and depth ablations in Tables 2 and 6 test this joint generator.

3.2 IMITATION LEARNING

A continuous generator trained only from financial rewards would have to discover coordinated books through noisy outcomes and trading costs. We therefore initialize the action head randomly and train it by imitation learning on causal strategy decisions $w_t^\star$; only the frozen Kronos encoder starts from pretrained weights. We design our teacher method, named AgenticAlpha, by aggregating effective alphas mined by AlphaGen (Yu et al., 2023) and AlphaSAGE (Chen et al., 2026) to supply the reference demonstrations, while also employing other common teachers to test whether refinement depends on this particular teacher as discussed in Section 5.2 and Table 4. These are historical strategy actions, not hindsight-optimal portfolios: imitation supplies plausible directions and scale for exploration, without making the teacher's return the final objective.

Conditional flow matching moves noise toward each demonstrated book. With $a_t = c w_t^\star$, $\epsilon \sim \mathcal{N}(0, I)$, and sampled flow time τ, define $z_\tau = (1 - \tau)\epsilon + \tau a_t$ and minimize

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}\left[\frac{1}{N_t} \left\| v_\theta(z_\tau, \tau; X_t) - (a_t - \epsilon) \right\|_2^2\right]. \tag{2}$$

This teaches a distribution over complete allocations. Because useful search proposals also require reliable relative preferences and nontrivial sizing, we supplement flow matching with a direct-score readout and losses on the estimated terminal action $\widehat{a} = z_\tau + (1-\tau)v_\theta(z_\tau, \tau; X_t)$. The imitation objective is

$$\mathcal{L}_{\mathrm{IL}} = \mathcal{L}_{\mathrm{FM}} + \mathcal{L}_{\mathrm{score}} + 0.5\mathcal{L}_{\mathrm{weight}} + 0.05\mathcal{L}_{\mathrm{rank}} + 0.1\mathcal{L}_{\mathrm{gross}}. \tag{3}$$

The ranking term preserves which assets the teacher favors; the weight and gross terms preserve direction, relative size, and book scale. Appendix A.1 discusses the uninformative no-trade case. Imitation shapes where RL begins without requiring it to reproduce a teacher action. The teacher ablation in Table 4 examines this initialization choice.

3.3 Refining joint decisions with portfolio rewards

Imitation makes the generated books meaningful, but it cannot tell the policy which deviations improve portfolio outcomes. Flow-GRPO (Liu et al., 2025) therefore compares complete candidate books under the same future market path. For each historical context, the old policy samples $G = 8$ candidates through $K = 10$ stochastic flow transitions. Each is held fixed over the same $H = 24$ forward hourly bars. To compare composition rather than reward a mere increase in leverage, we normalize each decoded candidate to gross exposure $g_0 = 0.85$ before scoring:

$$\widetilde{w}_i = g_0 \frac{w_i}{\|w_i\|_1 + \varepsilon}, \quad p_{i,h} = \widetilde{w}_i^\top r_{t,h}, \quad R_i = \sqrt{8760}\frac{\mathrm{mean}_{h=1}^{H} p_{i,h}}{\mathrm{std}_{h=1}^{H} p_{i,h} + \varepsilon}. \tag{4}$$

Here $r_{t,h}$ is the vector of returns h bars ahead. This gross Sharpe reward provides a local comparison among alternative exposures, not a simulation of the deployed trading sequence. Deployment generates a new, variable-gross target each hour and charges fees on target changes; the after-fee ledger therefore remains the test of financial value.

Financial rewards are noisy, so unrestricted updates could discard the structure learned from demonstrations. We keep a copy of the imitation-learned policy θ_{IL} frozen as a soft reference while updating only the action head. Following clipped policy optimization (Schulman et al., 2017; Shao et al., 2024) and stochastic flow-policy refinement (Liu et al., 2025; Zhang et al., 2025a), let $A_i = \mathrm{clip}((R_i - \overline{R})/(s_R + \varepsilon), -5, 5)$ be the group-relative advantage. For mean per-asset transition log density $\ell_\theta^{i,k}$, set $q_{i,k} = \exp(\ell_\theta^{i,k} - \ell_{\mathrm{old}}^{i,k})$. The update minimizes

$$\mathcal{L}_{\mathrm{RL}} = \mathbb{E}\left[\mathcal{C}(q_{i,k}, A_i) + \lambda\mathcal{C}(q_{i,k}, B_{i,k}) + \beta\mathcal{K}_{i,k}\right]. \tag{5}$$

Here $\mathcal{C}$ is the negative clipped surrogate. The detached $B_{i,k}$ favors transitions scored more highly by this frozen reference than by the old policy, following the on-policy distillation idea of Wang et al. (2026); $\mathcal{K}_{i,k}$ penalizes deviation of transition means from that reference. We use $\lambda = 0.1$, $\beta = 0.004$, and ratio clipping $\delta = 10^{-3}$; Appendix B.3 specifies the sampler and dimension normalization. The reference guides exploration without forcing exact imitation. After training, the same head generates target books with four deterministic Euler steps. The reward ablation in Table 7 examines the choice of financial objective.

4 Experiments

The main experiment asks whether generating a joint target book improves realized, after-fee portfolio outcomes. We first compare imitation with its RL-refined continuation under the same model and ledger, then place the complete policy alongside published alpha-mining methods. Section 5 examines which design choices account for the observed differences.

4.1 Model Training

We train the randomly initialized action head in two stages while keeping the pretrained Kronos encoder frozen. First, AgenticAlpha produces a target portfolio for each historical market context using only information available at that time. The head learns from these examples through flow matching; auxiliary losses help it preserve the teacher's position sizes, asset preferences, and total exposure. This imitation learning (IL) stage gives the policy plausible complete portfolios before it is asked to improve financial outcomes.

In the second stage, the policy generates eight alternative portfolios for each training context. We scale each to 0.85 gross exposure, hold it fixed over the same next 24 hourly bars, and compute its gross Sharpe. This controls for leverage and market path when comparing candidate allocations. Flow-GRPO compares the eight scores and updates the head toward better candidates, while a frozen copy of the imitation-learned policy keeps updates anchored. Demonstrations and reward windows exclude the test period. This fixed-book reward guides training; the backtest below evaluates changing target books after fees. At test time, four Euler steps generate each target portfolio. Appendix B.3 gives the sampler details.

4.2 Evaluation design

We use cryptocurrencies because granular historical trading records are freely and publicly available over long periods. Policy training and test samples are strictly separated in time. Kronos pretraining ends in June 2024 (Shi et al., 2025), before the hourly test from November 2, 2024, to December 31, 2025; thus the frozen encoder has no test-period information in its weights. Every week, the eligible list is recomputed as the 20 assets with the highest cumulative trading volume in the preceding 60 days. This causal reconstitution tests a changing opportunity set rather than a list selected with knowledge of the full test period.

All targets enter a fixed-notional ledger after one bar. Orders are changes from current to target weights, including entries and exits when the eligible list changes, and incur 5 basis points per unit of absolute traded weight. We report arithmetic annualized net return, additive maximum drawdown (MDD), net daily Sharpe, Calmar, and average daily turnover (TO), as defined in Appendix C.1. These measures jointly test profitability, downside exposure, and the trading needed to obtain it.

Published alpha-mining baselines use a common multi-asset signal-to-allocation pipeline, whereas AllocFlow generates weights directly. The IL-to-RL comparison keeps the frozen encoder and action-head architecture the same, making it the more direct test of reward refinement. Teacher strategies and native long-only Kronos deployment are examined separately in Section 5.

4.3 Quantitative Results

Reward refinement improves the book on both sides of the return-risk trade-off. In Table 1, net annualized return rises from 27.79% after IL to 51.38% after RL, while MDD falls from 29.10% to 12.81%. The resulting Calmar increase is thus supported by both a larger numerator and a smaller drawdown, rather than by return alone. Because both stages use the same frozen representation and action-head architecture, this comparison is the main evidence that the reward stage changes the quality of the generated portfolio. The AllocFlow (RL) results in Table 1 are means with standard deviations over AllocFlow trained with five random seeds; each metric is computed for each seed before averaging.

Table 1: **Retrospective performance after modeled fees.** Return and MDD are percentages; TO is average daily turnover. Published alpha-mining baselines share a multi-asset signal-to-allocation pipeline, while AllocFlow generates positions directly.

Policy	Return ↑	MDD ↓	Sharpe ↑	Calmar ↑	TO
AlphaGen (Yu et al., 2023)	35.42	20.60	1.242	1.720	0.248
AlphaSAGE (Chen et al., 2026)	31.30	23.36	1.396	1.340	1.255
R&D-Agent-Quant (Li et al., 2025)	20.69	18.53	0.967	1.117	0.397
AlphaAgent (Tang et al., 2025)	29.12	26.93	1.048	1.081	0.651
AllocFlow (IL)	27.79	29.10	0.954	0.955	1.143
AllocFlow (RL)	**51.38** ±2.04	**12.81** ±1.51	**2.192** ±0.086	**4.043** ±0.379	1.976 ±0.047

The improvement also survives a more demanding trading profile. Average daily turnover increases from 1.143 to 1.976, so the refined policy trades more weight; the reported return already deducts the associated modeled fees. Net daily Sharpe also rises, indicating improvement under a volatility-adjusted measure as well as the drawdown-based Calmar ratio. Figure 1 adds a temporal view of one run: over a test window longer than a year, after-cost PnL trends upward and the plotted rolling 365-day Sharpe stays near 2 for much of the out-of-sample period, supporting sustained profitability

in this retrospective evaluation. To have a more detailed understanding of the trading behavior change during the RL process, we also present two cases with zoomed-in concrete trading actions in Figure 3. The reward ablation in Section 5 examines behavior and objective choice more directly.

Against external methods, the final policy achieves the highest reported net return and Calmar in Table 1; the strongest listed alpha baseline reaches 1.720 Calmar. This comparison evaluates complete signal-to-position pipelines, which is relevant because AllocFlow is designed to decide positions jointly instead of converting independent alpha scores afterward. Differences in baseline coverage and trading rules prevent attributing that gap to the action head alone. The fixed-backbone head comparisons in Section 5 provide the narrower test of that design choice.

5 Ablations

The ablations study the proposed AllocFlow design from three sides: market features must carry useful context information, the action head must turn them into a coherent book, and reward refinement must improve on imitation. We compare heads, teachers, encoders, depth, and rewards using the after-fee ledger of Table 1; the native Kronos forecast deployment is identified separately.

5.1 The profitability comes from the Joint Action Generation or Kronos

If frozen Kronos features alone explained the portfolio gain, simpler heads using those features should benefit similarly from reward refinement. We therefore compare the flow head with an MLP, a one-pass Transformer, and a sequential autoregressive categorical head under the reported IL and RL configurations (Table 2). The first three reach similar Calmar after imitation learning, but the flow head improves from 0.955 to 4.043 after RL, while the MLP and one-pass Transformer show mixed changes. The distinction emerges primarily during refinement, which supports modeling a distribution of complete continuous books as a useful match for the portfolio-level search.

The categorical head supplies a further contrast: its imitation-learned book loses money, and its tested RL continuation produces no trades. This is consistent with poor proposals making group-relative improvement difficult, but it does not show that discrete policies must fail. The heads also use different RL training kernels, so Table 2 supports the effectiveness of the full flow-head design in this setup rather than isolating flow sampling as the sole cause.

Table 2: **Action-head comparison with frozen Kronos.** "NT" denotes an observed no-trade outcome, for which risk-adjusted ratios are not meaningful.

	After IL			After RL		
Head	Return (%)	Sharpe	Calmar	Return (%)	Sharpe	Calmar
Flow Transformer	27.79	0.954	0.955	51.38	2.192	4.043
MLP	27.31	0.935	0.912	20.12	0.856	1.307
One-pass Transformer	27.78	0.949	0.878	20.85	0.786	0.960
Autoregressive	-35.51	-1.189	-0.480		NT	

We also ask whether Kronos's next-bar forecasts could be traded directly, without learning a joint allocation policy. Table 3 deploys zero-shot and LoRA-adapted forecasts through their native long-only top-K ranking rule; both lose money after fees. This illustrates the gap between extracting market information and converting it into a profitable book. Because these forecast portfolios differ in exposure and construction from AllocFlow, they are contextual end-to-end comparisons; the fixed-backbone head experiment provides tighter evidence for learning the allocation action.

5.2 Demonstrations initialize action space; RL creates value

Teacher variation probes the role of imitation: does it merely copy a profitable expert, or does it provide structured candidate books that RL can improve? For each of four causal strategies, we train a new randomly initialized action head by imitation learning: AgenticAlpha, which combines AlphaGen and AlphaSAGE signals developed on training data; MACD (12h, 26h); MACD (48h, 104h); and Alpha101 #9 (Kakushadze, 2016). Table 4 compares each resulting IL policy with its RL

continuation on the same ledger. Calmar improves in every reported pair, including short-horizon MACD, whose IL policy is loss-making. A profitable teacher imitation is therefore not required for the observed RL improvement.

Initialization still matters: AgenticAlpha yields the strongest refined policy among these teachers. The combined pattern fits our two-stage design that demonstrations supply nontrivial directions and sizes, while reward feedback selects better portfolios from that starting region. It does not establish that imitation is strictly necessary, because none of the reported RL policies began reward training directly from random head weights.

Table 3: Kronos forecasts with native long-only deployment; these portfolios use a different allocation rule.

Kronos-direct	Return (%)	Sharpe	Calmar
Zero-shot	−44.502	−0.469	−0.438
LoRA	−34.570	−0.391	−0.312

Table 4: Policy Calmar by teacher signal.

Teacher	After IL	After RL
AgenticAlpha	0.955	4.043
MACD (12h, 26h)	−0.559	1.856
MACD (48h, 104h)	0.538	2.114
Alpha101 #9	0.401	1.068

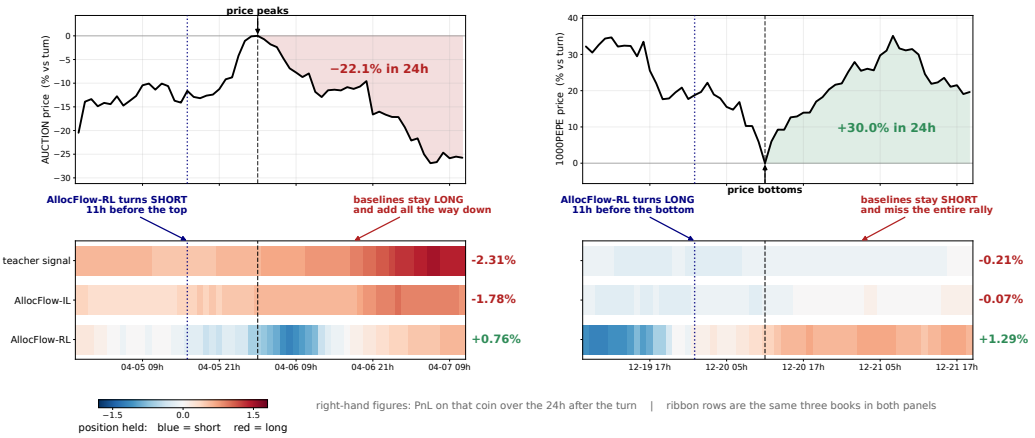


Figure 3: **Opposite asset-level reversals.** Left: AUCTION falls from a local peak. Right: 1000PEPE rebounds from a local bottom. Black dashed lines mark price turns; blue dotted lines mark RL's earlier change in position direction. Blue/red positions are short/long. The labels beside each panel give asset-specific PnL over the subsequent 24 hours.

Zoom-in to local decision traces. Aggregate ratios cannot reveal what changed in an individual trade. Figure 3 aligns asset prices with the teacher, IL, and RL position ribbons around two opposite turning points. This makes the relevant question visible: whether RL changes direction in time to participate in the subsequent move, rather than merely rescaling the demonstrated book.

On AUCTION, RL turns short about 11 hours before the peak, while the teacher and IL policy stay long through much of the decline. As the asset falls 22.1% over the next 24 hours, RL records $+0.76\%$ asset-specific PnL versus -1.78% for IL.

1000PEPE shows the converse: RL covers its short and turns long about 11 hours before the bottom, while the teacher and IL policy remain short around the turn. The asset rises 30.0% over the following 24 hours; RL records $+1.29\%$ versus -0.07% for IL. The pair illustrates that refinement can alter the sign and timing of decisions in either direction. These selected windows explain a mechanism, not how often it occurs across the full portfolio.

5.3 Market Representation

The flow head can learn a useful book only if its compact market tokens preserve information relevant to trading. To probe that dependency, we replace frozen Kronos with a frozen iTransformer

while retaining the flow action head and evaluating both IL and RL policies on the same ledger (Table 5). The replacement loses heavily at both stages; reward refinement does not recover the performance of the Kronos-conditioned policy. Together with the poor Kronos-direct forecast deployment, this points to a complementary design: well-represented market features are valuable when a suitable head learns how to allocate from them, but features alone are not the trading policy.

Table 5: Encoder replacement with a flow head. Return is annualized arithmetic net percent.

Encoder / stage	Return	Sharpe	Calmar
Kronos / IL	27.79	0.954	0.955
Kronos / RL	51.38	2.192	4.043
iTransformer / IL	−141.58	−5.359	−0.850
iTransformer / RL	−119.29	−7.119	−0.846

Table 6: Flow-head depth with frozen Kronos. After-fee return is in percent.

Depth / stage	Return	Sharpe	Calmar
12L / IL	29.37	0.999	1.033
12L / RL	46.74	2.109	3.728
32L / IL	27.79	0.954	0.955
32L / RL	51.38	2.192	4.043

5.4 The Effect of Flow-head Size

We also study the influence of the diffusion head size. Therefore, we compare 12-layer and 32-layer flow heads with the same frozen Kronos representation and the reported two-stage training procedure (Table 6). The shallower head is slightly stronger after IL, yet both heads improve markedly under RL; the 12-layer version reaches 3.728 Calmar after refinement. The deeper head achieves the higher final return and Calmar. It would be intriguing to study the scaling of the AllocFlow when resources are available.

5.5 Reward choice and RL objective

The group-relative update can only prefer the behavior its reward distinguishes, so we experiment with various common financial metrics as reward to find the influence. Table 7 compares Sharpe, cumulative return, mean return, and a risk-adjusted-return variant against the IL starting policy, using the same after-fee evaluation. Cumulative return gives the highest annualized return, but the Sharpe-trained policy has the smallest MDD and highest Calmar. Its turnover exceeds that of the return-focused variants, so the downside improvement is not simply the result of trading less. The risk-adjusted-return variant loses money, showing that nominal risk adjustment does not by itself ensure a useful training signal in this setup. The Sharpe ratio is the best reward we find that balances return and downside risk better than alternatives.

Table 7: After-fee reward ablation. Return and MDD are percentages.

Training reward	Return	MDD	Sharpe	Calmar	TO
IL initialization	27.79	29.10	0.954	0.955	1.143
Sharpe	51.38	12.81	2.192	4.043	1.976
Cumulative return	60.62	16.79	2.150	3.611	1.658
Mean return	58.36	17.19	2.051	3.395	1.613
Risk-adjusted return	−10.63	28.60	−0.480	−0.372	2.070

6 Conclusion

AllocFlow turns raw multi-asset histories into a complete set of signed portfolio positions through a cross-asset flow model. Strategy-supervised imitation first establishes structured, nontrivial trading actions; group-relative RL then improves their portfolio performance while a frozen reference guides refinement. The reported after-fee results and improvements across four demonstration priors support this progression. Architecture comparisons explain why useful market representations must be paired with an action model that can both express coordinated allocations and support reward-based updates. Together, the design and training recipe realize a profitable instance of direct joint portfolio decision-making. Matched initialization controls, independent forward evaluation, and closer alignment to execution can strengthen and extend this evidence.

AI USE STATEMENT

We used generative AI tools to help us understand relevant background material, improve the clarity and wording of the manuscript, and assist with code implementation. LLMs are also used to construct the teacher signals. The authors reviewed the AI-assisted work and take responsibility for the final paper and code.

REPRODUCIBILITY STATEMENT

We describe the model, training procedure, evaluation protocol, and implementation details in Sections 3 and 4 and the appendix. We will publicly release the code.

ETHICS STATEMENT

This study does not involve human subjects or sensitive personal data. Nevertheless, generative trading models could be misused, and their deployment requires appropriate safeguards.

REFERENCES

Ahmad Aghapour, Erhan Bayraktar, and Fengyi Yuan. Solving dynamic portfolio selection problems via score-based diffusion models, 2025. URL https://arxiv.org/abs/2507.09916.

Boris Belyakov. AlphaZeroBeta: Deep reinforcement learning for market-neutral portfolios, 2026. URL https://arxiv.org/abs/2607.18001.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2024. URL https://arxiv.org/abs/2410.24164.

Hans Bühler, Lukas Gonon, Josef Teichmann, and Ben Wood. Deep hedging, 2018. URL https://arxiv.org/abs/1802.03042.

Binqi Chen, Hongjun Ding, Ning Shen, Taian Guo, Jinsheng Huang, Luchen Liu, and Ming Zhang. AlphaSAGE: Structure-aware alpha mining via GFlowNets for robust exploration. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/06115011841a8d500050261c0ac4b1e4-Abstract-Conference.html.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion, 2023. URL https://arxiv.org/abs/2303.04137.

Yue Deng, Feng Bao, Youyong Kong, Zhiquan Ren, and Qionghai Dai. Deep direct reinforcement learning for financial signal representation and trading. *IEEE Transactions on Neural Networks and Learning Systems*, 28(3):653–664, 2017. doi: 10.1109/TNNLS.2016.2522401. URL https://doi.org/10.1109/TNNLS.2016.2522401. First published online in 2016.

Li Dong and Haichao Zheng. Soft imitation reinforcement learning with value decomposition for portfolio management. *Applied Soft Computing*, 151:111108, 2024. doi: 10.1016/j.asoc.2023.111108. URL https://www.sciencedirect.com/science/article/pii/S1568494623011262.

Adam N. Elmachtoub and Paul Grigas. Smart "predict, then optimize". *Management Science*, 68 (1):9–26, 2022. URL https://arxiv.org/abs/1710.08005.

Nicolae Gârleanu and Lasse Heje Pedersen. Dynamic trading with predictable returns and transaction costs. *The Journal of Finance*, 68(6):2309–2340, 2013. doi: 10.1111/jofi.12080. URL https://docs.lhpedersen.com/DynamicTrading.pdf.

Shihao Gu, Bryan Kelly, and Dacheng Xiu. Empirical asset pricing via machine learning. *The Review of Financial Studies*, 33(5):2223–2273, 2020. doi: 10.1093/rfs/hhaa009. URL https://dachxiu.chicagobooth.edu/download/ML_BKP.pdf.

Haeun Jeon, Seunghoon Choi, Hyunglip Bae, Yongjae Lee, and Woo Chang Kim. Decision-focused sparse tangent portfolio optimization. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://arxiv.org/abs/2607.00581.

Yifu Jiang, Jose Olmo, and Majed Atwi. Deep reinforcement learning for portfolio selection. *Global Finance Journal*, 62:101016, 2024. doi: 10.1016/j.gfj.2024.101016. URL https://www.sciencedirect.com/science/article/pii/S1044028324000887.

Zhengyao Jiang, Dixing Xu, and Jinjun Liang. A deep reinforcement learning framework for the financial portfolio management problem, 2017. URL https://arxiv.org/abs/1706.10059.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-LLM: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.01728.

Zura Kakushadze. 101 formulaic alphas, 2016. URL https://arxiv.org/abs/1601.00991.

Jiahao Li, Yong Zhang, Xingyu Yang, and Liangwei Chen. Online portfolio management via deep reinforcement learning with high-frequency data. *Information Processing & Management*, 60(3):103247, 2023. doi: 10.1016/j.ipm.2022.103247. URL https://www.sciencedirect.com/science/article/pii/S030645732200348X.

Yuante Li, Xu Yang, Xiao Yang, Minrui Xu, Xisen Wang, Weiqing Liu, and Jiang Bian. R&D-Agent-Quant: A multi-agent framework for data-centric factors and model joint optimization, 2025. URL https://arxiv.org/abs/2505.15155.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling, 2022. URL https://arxiv.org/abs/2210.02747.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-GRPO: Training flow matching models via online RL, 2025. URL https://arxiv.org/abs/2505.05470.

Xiao-Yang Liu, Hongyang Yang, Qian Chen, Runjia Zhang, Liuqing Yang, Bowen Xiao, and Christina Dan Wang. FinRL: A deep reinforcement learning library for automated stock trading in quantitative finance, 2020. URL https://arxiv.org/abs/2011.09607.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting, 2023. URL https://arxiv.org/abs/2310.06625.

Harry Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952. doi: 10.1111/j.1540-6261.1952.tb01525.x. URL https://doi.org/10.1111/j.1540-6261.1952.tb01525.x.

John Moody and Matthew Saffell. Learning to trade via direct reinforcement. *IEEE Transactions on Neural Networks*, 12(4):875–889, 2001. doi: 10.1109/72.935097. URL https://people.idsia.ch/~juergen/rnnaissance2003talks/MoodySaffellTNN01.pdf.

Arishi Orra, Himanshu Choudhary, and Manoj Thakur. Learning stock trading policies via barycenter-based adversarial inverse reinforcement learning. *arXiv preprint arXiv:2608.15770*, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers, 2022. URL https://arxiv.org/abs/2212.09748.

Allen Z. Ren, Justin Lidard, Lars L. Ankile, Anthony Simeonov, Pulkit Agrawal, Anirudha Majumdar, Benjamin Burchfiel, Hongkai Dai, and Max Simchowitz. Diffusion policy policy optimization. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2409.00588.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Yu Shi, Zongliang Fu, Shuo Chen, Bohan Zhao, Wei Xu, Changshui Zhang, and Jian Li. Kronos: A foundation model for the language of financial markets, 2025. URL https://arxiv.org/abs/2508.02739.

Chen-Hui Song, Shuoling Liu, and Liyuan Chen. The label horizon paradox: Rethinking supervision targets in financial forecasting. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026a. URL https://arxiv.org/abs/2602.03395.

Zifan Song, Kaitao Song, Guosheng Hu, Ding Qi, Junyao Gao, Xiaohua Wang, Dongsheng Li, and Cairong Zhao. Trade in minutes! rationality-driven agentic system for quantitative financial trading. In *International Conference on Learning Representations*, 2026b. URL https://arxiv.org/abs/2510.04787.

Yejun Soun, Hosung Lee, Suyoung Park, and U Kang. STABLE: Shift-tolerant allocation via black-litterman using conditional diffusion estimates. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/4d4dbc5c955c5d273eed12d565335068-Abstract-Conference.html.

Ziyi Tang, Zechuan Chen, Jiarui Yang, Jiayao Mai, Yongsen Zheng, Keze Wang, Jinrui Chen, and Liang Lin. AlphaAgent: LLM-driven alpha mining with regularized exploration to counteract alpha decay, 2025. URL https://arxiv.org/abs/2502.16789.

Yinjie Wang, Xuyang Chen, Xiaolong Jin, Mengdi Wang, and Ling Yang. OpenClaw-RL: Train any agent simply by talking, 2026. URL https://arxiv.org/abs/2603.10165.

Jize Xie, Haomiao Wu, Qiang Chen, Xiu Su, and Yi Chen. Decision-weighted flow matching for contextual stochastic optimization, 2026. URL https://arxiv.org/abs/2606.16790.

Shuo Yu, Hongyan Xue, Xiang Ao, Feiyang Pan, Jia He, Dandan Tu, and Qing He. Generating synergistic formulaic alpha collections via reinforcement learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 5476–5486, 2023. doi: 10.1145/3580305.3599831. URL https://arxiv.org/abs/2306.12964.

Yangyang Yu, Zhiyuan Yao, Haohang Li, Zhiyang Deng, Yuechen Jiang, Yupeng Cao, Zhi Chen, Jordan W. Suchow, Zhenyu Cui, Rong Liu, Zhaozhuo Xu, Denghui Zhang, Koduvayur Subbalakshmi, Guojun Xiong, Yueru He, Jimin Huang, Dong Li, and Qianqian Xie. FinCon: A synthesized LLM multi-agent system with conceptual verbal reinforcement for enhanced financial decision making. In *Advances in Neural Information Processing Systems*, 2024. URL https://proceedings.neurips.cc/paper/2024/hash/f7ae4fe91d96f50abc2211f09b6a7e49-Abstract-Conference.html.

Tonghe Zhang, Chao Yu, Sichang Su, and Yu Wang. ReinFlow: Fine-tuning flow matching policy with online reinforcement learning. In *Advances in Neural Information Processing Systems*, 2025a. URL https://arxiv.org/abs/2505.22094.

Zihao Zhang, Stefan Zohren, and Stephen Roberts. Deep reinforcement learning for trading. *The Journal of Financial Data Science*, 2(2):25–40, 2020. doi: 10.3905/jfds.2020.1.030. URL https://arxiv.org/abs/1911.10107.

Ziliang Zhang, Tianming Zhao, and Albert Y. Zomaya. Online portfolio selection with ML predictions. In *Advances in Neural Information Processing Systems*, 2025b. URL https://doi.org/10.52202/085713-2762.

Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One fits all: Power general time series analysis by pretrained LM. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2302.11939.

A Mathematical Clarifications

A.1 A no-trade condition under uninformative expected returns

Consider a one-period decision from a zero book with conditional expected return vector $\mu \in \mathbb{R}^N$, proportional trading cost $\kappa\|w\|_1$, and risk penalty $\Psi(w) \geq 0$ satisfying $\Psi(0) = 0$. Let the feasible set contain zero, and define

$$J(w) = \mu^\top w - \kappa\|w\|_1 - \Psi(w). \tag{6}$$

If $\|\mu\|_\infty \leq \kappa$, zero is a global maximizer of this objective. Indeed, Hölder's inequality gives

$$J(w) \leq (\|\mu\|_\infty - \kappa)\|w\|_1 - \Psi(w) \leq 0 = J(0). \tag{7}$$

This elementary observation also holds after restricting the feasible set to dollar-neutral books. It motivates biasing initial portfolio proposals toward useful trading behavior, but is not a convergence theorem or evidence that the reported flow policy collapses without imitation. It assumes a zero current book; with existing holdings, costs depend on position changes instead. It also does not directly describe the selected gross-Sharpe reward in Equation 4.

A related optimization issue is that a group of identical zero books receives identical rewards, yielding zero group-relative advantages. The reward-driven surrogate then provides no directional signal for that group. This does not make zero an absorbing state in general: stochastic exploration, nonzero categorical probability mass, and reference regularization can still change a policy. The reported zero-book outcomes concern alternative action heads refined after IL. They neither prove a general failure of discrete-action RL nor supply the missing control without imitation.

A.2 Dollar neutrality, factor neutrality, and execution rounding

For $P = I - \mathbf{1}\mathbf{1}^\top/N$, $P^2 = P$ and $\mathbf{1}^\top P = 0$. Consequently, the continuous decoder in Equation 1 maps every latent to the dollar-neutral subspace. If asset returns follow $r = \beta f + u$, however, portfolio factor exposure is $w^\top\beta$. This need not vanish when $w^\top\mathbf{1} = 0$. For example, $w = (a, -a)$ and $\beta = (2, 1)$ give $w^\top\beta = a$. Dollar neutrality therefore cannot certify beta neutrality, sector neutrality, or bounded drawdown.

Rounding each position to a grid of width q in allocation units can also introduce a residual net position. With fixed notional A, nearest-grid rounding gives the loose bound $|\mathbf{1}^\top\widehat{w}| \leq Nq/(2A)$, assuming no additional clipping. The recorded $A = 10$ and $q = 0.1$ correspond to 0.01 increments in normalized portfolio weights. Single-name clipping in the teacher or factor pipeline may additionally alter neutrality. Exact deployed neutrality would require a final feasible projection, which is not assumed in the reported implementation.

B Implementation and Training Details

B.1 Market context and demonstration construction

The reference universe chooses the top 20 assets by trailing 60-day traded amount and reconstitutes weekly on Sunday. The reference strategy first computes a signal for each asset, then applies its allocation pipeline: temporal smoothing over 24 hourly bars, volatility scaling with target 0.05, a gross allocation cap of 10 units, a per-name clip of ± 2 units, and a 0.1-unit execution grid. Demonstrations are the resulting allocations divided by fixed notional $A = 10$.

The model encoder uses window-normalized OHLCV and derived amount channels, with normalized values clipped to $[-5, 5]$. The recent eight coarse and eight fine token states are pooled separately for each asset. Their summaries, a cross-asset market summary, asset identity embeddings, and calendar information condition the action head. The implementation reads encoder width from the loaded checkpoint configuration. We use the Kronos-base checkpoint in our implementation.

B.2 Supervised auxiliary objectives

Let $h(u)$ denote the unit-threshold Huber loss: $h(u) = u^2/2$ for $|u| < 1$ and $|u| - 1/2$ otherwise. For a nonempty demonstrated book, define $\widehat{w} = P\widehat{a}/c$ and let $d_\theta(X)$ be the auxiliary direct-score

output, with $w^{\mathrm{dir}} = Pd_\theta(X)/c$. The portfolio auxiliary terms are

$$\mathcal{L}_{\mathrm{weight}} = \mathbb{E}\left[\frac{1}{N}\sum_j h(c\widehat{w}_j - cw_j^\star)\right], \tag{8}$$

$$\mathcal{L}_{\mathrm{gross}} = \mathbb{E}\left[h(\|c\widehat{w}\|_1 - \|cw^\star\|_1)\right], \tag{9}$$

$$\mathcal{L}_{\mathrm{score}} = \mathbb{E}\left[\frac{1}{N}\sum_j h(cw_j^{\mathrm{dir}} - cw_j^\star)\right]. \tag{10}$$

They supervise action-scaled weights, so their magnitude is comparable to the flow loss. Empty demonstrated books are excluded from these three auxiliary averages in the allocation-target path. The ranking term averages over pairs whose teacher weight gap exceeds 0.01:

$$\mathcal{L}_{\mathrm{rank}} = \mathbb{E}_{(i,j):|w_i^\star - w_j^\star|>0.01}\left[\log\big(1+\exp\{-\operatorname{sign}(w_i^\star - w_j^\star)(\widehat{w}_i - \widehat{w}_j)\}\big)\right]. \tag{11}$$

Flow time is sampled as $\tau = 0.999(1-U)$ with $U \sim \mathrm{Beta}(1.5, 1.0)$, and the velocity network uses 1,000 time buckets. These objectives encourage fidelity but do not force the neural policy to reproduce the teacher's temporal smoothness or executed turnover.

B.3 Stochastic flow transitions and loss normalization

For K transitions, let $\Delta = 1/K$, $\tau_k = k/K$, and $s_k = 1 - \tau_k$, for $k = 0, \ldots, K-1$. Set $d_k = \tau_k$ for $k > 0$, and $d_0 = 1/K$. With noise coefficient $\nu = 0.7$, the implemented stochastic flow sampler uses

$$\gamma_k = \nu\sqrt{s_k/d_k}, \tag{12}$$

$$m_\theta(z_k, \tau_k) = z_k\left(1 - \frac{\gamma_k^2\Delta}{2s_k}\right) + v_\theta(z_k, \tau_k; X)\left(1 + \frac{\gamma_k^2\tau_k}{2s_k}\right)\Delta, \tag{13}$$

$$z_{k+1} = m_\theta(z_k, \tau_k) + \gamma_k\sqrt{\Delta}\,\xi_k, \qquad \xi_k \sim \mathcal{N}(0, I). \tag{14}$$

All evaluated transitions occur before $\tau = 1$, avoiding the terminal singularity. Setting $\nu = 0$ recovers an Euler update. The training sampler uses ten stochastic transitions; deployed inference uses four deterministic Euler transitions conditional on a seeded initial Gaussian latent. Thus deterministic inference refers to repeatability for a given timestamp seed, not zero initial noise.

The scalar transition statistic is

$$\ell_\theta = \frac{1}{N}\sum_{j=1}^{N}\left[-\frac{(z_{k+1,j} - m_{\theta,j})^2}{2\gamma_k^2\Delta} - \log(\gamma_k\sqrt{\Delta}) - \frac{1}{2}\log(2\pi)\right]. \tag{15}$$

Consequently, $q = \exp(\ell_\theta - \ell_{\mathrm{old}})$ is the Nth root of the full joint transition likelihood ratio. This dimension normalization changes the clipped surrogate and must be retained in a reproduction. It should not be interpreted as an exact untempered joint-policy likelihood-ratio estimator.

The two clipped surrogates in Equation 5 share the definition

$$\begin{aligned}\mathcal{C}(q, a) &= -\min\{qa, \mathrm{clip}(q, 1-\delta, 1+\delta)a\},\\ B_{i,k} &= \mathrm{sg}\Big[\mathrm{clip}(\ell_{\mathrm{S}}^{i,k} - \ell_{\mathrm{old}}^{i,k}, -5, 5)\Big], \qquad \delta = 10^{-3}.\end{aligned} \tag{16}$$

Here sg stops gradients through the reference advantage. Both surrogates use the same sampled transition and dimension-normalized ratio, but their advantages express different preferences: realized portfolio reward for A_i, and relative reference-policy support for $B_{i,k}$.

The separate reference term in Equation 5 is

$$\mathcal{K}_k = \frac{\|m_\theta - m_{\mathrm{S}}\|_2^2}{2N\gamma_k^2} = \frac{\Delta}{N}D_{\mathrm{KL}}\big(\mathcal{N}(m_\theta, \gamma_k^2\Delta I)\,\|\,\mathcal{N}(m_{\mathrm{S}}, \gamma_k^2\Delta I)\big). \tag{17}$$

It is a scaled Gaussian divergence, because the denominator uses the diffusion coefficient rather than the full transition variance. The OPD surrogate uses a detached difference of teacher and old-policy log densities on the same sampled transition. This log-density-difference advantage follows

the on-policy distillation pattern used in OpenClaw-RL (Wang et al., 2026); here the reference is a frozen IL trading policy, without hindsight textual hints. The reference variant does not use an EMA teacher or noise-only optimization. Dropout is disabled during both stochastic rollout and likelihood recomputation, while gradients update the action head.

Training begins with historical contexts and causal strategy allocations. The frozen backbone encodes each context, and Equation 3 trains the action head before a copy is retained as θ_{S}. Refinement samples eight old-policy stochastic paths per context and stores their states and mean log densities. Terminal portfolios are decoded, rescaled to gross 0.85, and scored on the forward return window. After computing group-relative advantages, the trainable and reference heads recompute the sampled transition densities. Equation 5 averages over contexts, candidates, and ten transitions to update the action head.

Deployment retains this head but uses the shorter deterministic integration path. A timestamp seeds the initial Gaussian latent; four Euler steps generate the action, which is centered and divided by $c = 10$. The rounded target enters the next ledger bar. Distinguishing the scoring and deployment paths matters: the former compares fixed candidate books over a local horizon, whereas the latter produces a sequence whose position changes incur fees.

B.4 Recorded configuration and reproducibility boundaries

Table 8: Training and inference configuration.

Component	Recorded setting
Market cadence / universe	Hourly; dynamic top 20; weekly reconstitution
Backbone	Frozen Kronos-base with its base tokenizer
Main flow head	32 layers; 32 attention heads; head dimension 48; about 1.03B parameters
IL dropout / time sampling	0.2; $0.999(1-U), U \sim \mathrm{Beta}(1.5, 1.0)$; 1,000 time buckets
Action scale / execution grid	$c = 10$; grid 0.1 allocation units for notional 10
IL auxiliary coefficients	Score 1.0; weight 0.5; ranking 0.05; gross 0.1
RL sampler	8 candidates; 10 stochastic transitions; noise coefficient 0.7
RL objective	Ratio clip 10^{-3}; advantage clip 5; reference coefficient 0.004; OPD 0.1
Inference	4 Euler steps; timestamp-seeded initial noise
Execution cost	5 bps per absolute traded weight; net daily metrics

C Accounting and Comparison Details

C.1 Ledger metrics and timing

Let $\widehat{w}_t$ denote the grid-rounded target generated from bars available through t, with zero weight outside the eligible universe. In the reported lagged target-weight ledger, the target becomes the position for the following bar. A normalized expression for the accounting is

$$g_{t+1} = \widehat{w}_t^{\top} r_{t+1}, \qquad u_{t+1} = \|\widehat{w}_t - \widehat{w}_{t-1}\|_1, \qquad n_{t+1} = g_{t+1} - 0.0005 u_{t+1}. \tag{18}$$

This describes the fixed-notional target-change convention. It does not assert that turnover includes continuously drifted holdings, realistic order fills, or market impact. For each day d, aggregate $n_d = \sum_{t \in d} n_t$, $u_d = \sum_{t \in d} u_t$, and let D be the number of evaluated days. Then

$$\mathrm{AnnRet} = \frac{365}{D} \sum_{d=1}^{D} n_d, \qquad \mathrm{Sharpe} = \sqrt{365}\, \frac{\overline{n}}{s(n)}, \qquad \mathrm{TO} = \frac{1}{D} \sum_{d=1}^{D} u_d, \tag{19}$$

$$C_d = \sum_{j=1}^{d} n_j, \qquad \mathrm{MDD} = \max_{0 \le a \le b \le D} (C_a - C_b), \qquad \mathrm{Calmar} = \frac{\mathrm{AnnRet}}{\mathrm{MDD}}, \tag{20}$$

with $C_0 = 0$. Metrics use zero as the risk-free benchmark. A zero-book run has no economically meaningful Sharpe or Calmar; our tables report both as No Trade (NT). All such entries are explicitly described as no-trade outcomes.

Native data alignment yields 423–425 evaluation days across methods. A common arithmetic metric convention does not equalize fills, rebalancing, or asset availability. Additive drawdown can exceed 100% in this bookkeeping, as seen for some negative baselines; those values should not be read as feasible continuation of a funded account after insolvency.

Training rewards use fixed candidate weights over H forward hourly returns. The code's after-cost variants compare the candidate with the model's deterministic previous-bar portfolio, computed without gradients, and penalize this change. This is different from replaying an entire future sequence of hourly decisions and their rebalancing costs.

D Additional Related Context

The decision to generate a portfolio does not remove the established connections between forecasting, control, and execution. Formulaic alphas (Kakushadze, 2016) remain useful interpretable signals and can supply strategy demonstrations. Dynamic trading theory explicitly models the cost of adjusting toward a target (Gârleanu & Pedersen, 2013), while deep hedging learns actions under trading constraints (Bühler et al., 2018). These connections highlight a distinction within direct action learning: choosing a target book and choosing the cost-aware path to that book are related but different problems. Our evaluated policy learns the former, and its sequence is assessed through a fee-charging ledger. Explicit holdings-conditioned control is a natural extension, rather than a capability established by the present model.

Recent preprints further narrow the novelty claim. Adversarial inverse-RL trading uses demonstrations from multiple behaviors (Orra et al., 2026), and AlphaZeroBeta studies market-neutral portfolio RL (Belyakov, 2026). Generative portfolio research also uses diffusion to model market environments (Aghapour et al., 2025), while decision-weighted flow matching targets distributions useful for downstream stochastic optimization (Xie et al., 2026). These works motivate separating what is generated from how it is used. In AllocFlow, a sample is the joint signed action itself, and the same action model participates in both imitation and reinforcement. Our empirical analysis consequently examines which action model and strategy-guided search configuration realize profitable joint decisions. It does not claim a new general-purpose flow objective, a new concept of portfolio neutrality, or priority for combining demonstrations with financial RL.