Volt: Searching the Option Strategy Space

research@strategynet.ai

July 2026

Abstract

An option chain supports many more portfolios than the named templates usually used to search it. Direct construction from listed contracts makes that wider space available. American marks, exercise, assignment, and pathwise Greeks are costly, but portfolios sharing the same contracts can reuse those leg calculations; direct enumeration is then a strong baseline. Volt trains order-invariant surrogates only where the remaining candidate or sequential evaluation cost may justify them. Deep Sets supplies the set baseline; a parameter-matched typed graph network consumes the exported portfolio, position, underlying, and expiry relations. This paper specifies the financial assumptions, mathematical formulation, and empirical criterion: recovery of high-ranked portfolios after all shared computation is separated from portfolio-specific work. QUBO batch selection and sequential policy learning follow as separate experiments.

Contents

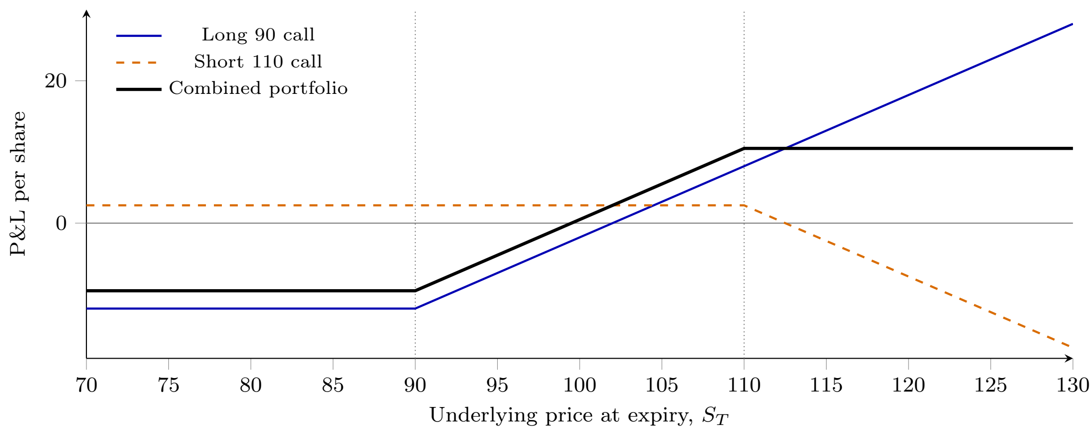

Figure 1: Expiry profit and loss for one long 90 call and one short 110 call, using illustrative entry prices of 12 and 2.50 before fees. The combined position has a net debit of 9.50, a break-even price of 99.50, a maximum loss of 9.50, and a maximum profit of 10.50 per share. The pathwise evaluator additionally requires pre-expiry marks, exercise, assignment, fills, and management actions.

1 Introduction

A common way to make option-strategy search tractable is to begin with a template: choose a vertical, straddle, butterfly, calendar, or another familiar shape, then search over its strikes, expiries, and quantities. This approach reduces the number of portfolios under consideration, but the reduction occurs before their behavior is evaluated. The number of legs, option rights, and sign pattern have already been chosen by the template.

The alternative considered here starts with the listed contracts themselves. A candidate portfolio assigns a signed quantity to each eligible contract, subject to limits on exposure, liquidity, capital, and leg count. Conventional strategy names can be applied afterward for interpretation. They play no role in candidate generation, so the search includes structures outside a predefined catalogue.

The resulting search space grows quickly. With d eligible contracts and quantities from a finite set $\mathcal{Q}$, the unfiltered space contains $|\mathcal{Q}|^d$ quantity vectors. Most applications cannot evaluate every vector across every relevant market path. Evaluation is especially costly for American options because portfolio behavior depends on the path of option values and on decisions or events that occur before expiry. A terminal payoff diagram supplies the expiration geometry; path evaluation supplies the pre-expiry behavior.

Payoff diagrams remain a useful first description of the economic position and are standard in options reference material [1]. Figure 1 shows how two contract positions combine. The portfolio can be recognized as a bull call spread after it has been constructed; candidate generation uses the two signed positions.

The practical objective is to recover high-utility portfolios without applying the full evaluator to the entire combinatorial set. A small evaluated sample is used to fit a surrogate; the surrogate ranks the remaining pool; an acquisition rule selects another batch for the financial evaluator; and the process repeats. Success is measured against exhaustive evaluation on small problems and by held-out performance on later decision dates.

This paper develops that problem in three stages. Section 2 describes the trading and derivatives assumptions. Sections 3 and 4 give the mathematical formulation and the first learning model. Sections 5 and 6 define candidate acquisition and the empirical comparison. The initial study searches entry portfolios under a fixed management rule. Sequential management is discussed separately in Section 9.

2 Trading and derivatives setting

2.1 Portfolio construction and strategy management

An entry portfolio specifies the contracts and signed quantities held at the decision time. A management rule specifies later decisions such as closing, resizing, rolling, exercising, or waiting. Assignment and other external events can change the position independently of the holder's choices.

The first experiment searches entry portfolios while holding the management rule and external-event model fixed. This isolates the portfolio-construction question: given the same information, paths, execution assumptions, and management rule, which initial combinations warrant further evaluation? Changing the management rule defines another experiment.

2.2 A concrete decision state

Suppose the point-in-time chain shows spot $S_t = 100$, a 90 call marked at 12, and a 110 call marked at 2.50, with both calls expiring in 30 days. Spot and marks are market observations. The listed strikes, exercise terms, expiry, and contract multiplier are contract data. The portfolio decision is the signed quantity on each contract: $+1$ for the 90 call and -1 for the 110 call in Figure 1.

The marks would ordinarily come from a timestamped option chain under a declared quote rule, such as mid, executable bid or ask, or a fitted surface. Synthetic examples set these values directly. The same applies to spot, interest rates, dividends, and the initial volatility state.

Changing a strike, expiry, option right, or quantity produces another candidate portfolio. Both leg orders map to the same candidate. A new date or a new set of contemporaneous marks produces another decision state. In the graph comparison, one model input represents a candidate–state pair, so the same contract quantities observed on two dates give two examples with different market features.

Candidate portfolios are generated by software from the eligible chain and the declared quantity, leg-count, liquidity, exposure, and capital limits. Manual entry remains useful for fixtures such as Figure 1.

2.3 Where market paths come from

The evaluator accepts three kinds of path evidence:

- historical paths assembled from point-in-time underlying and option observations;
- generated paths sampled from a stated stochastic or stress model; and
- forecast-conditioned paths produced by a model fitted before the decision time.

Historical replay measures behavior on recorded observations. Generated paths measure behavior under the specified distribution or stress construction. Forecast-conditioned paths measure behavior under a forecast that was available when the portfolio was chosen.

A path of spot, volatility, rates, and dividends still requires option valuation at each observation. Historical studies may use recorded option quotes. Generated studies require a pricing method that maps the factor state and contract terms to an option mark and, when execution is modelled, bid and ask prices. The pricing method, quote-selection rule, and missing-data policy are part of the experiment.

2.4 Early exercise and assignment

Let

$$\mathcal{T} = \{\tau_0, \tau_1, \ldots, \tau_H\}, \qquad \tau_0 < \tau_1 < \cdots < \tau_H,$$

be the observation grid. Every option has an exercise-opportunity set $\mathcal{E} \subseteq \mathcal{T}$. European exercise is the singleton $\mathcal{E} = \{T\}$. Bermudan exercise uses a finite declared subset. American exercise uses every grid point between the first exercise time and expiry:

$$\mathcal{E} = \{\tau_j \in \mathcal{T} : t_{\text{first}} \leq \tau_j \leq T\}.$$

European and Bermudan exercise are restricted cases of the same exercise-opportunity representation. The value and realized outcome of an American option depend on the observation grid and the holder's exercise rule. The treatment of a short position also requires an assignment rule or a supplied assignment schedule. The experiment therefore records:

- the observation grid;
- the holder's exercise rule;
- the short-assignment model;
- the option-valuation method;
- the fill and transaction-cost rules; and
- settlement and liquidation conventions.

Least-squares Monte Carlo is one established approach to estimating continuation values for American options [3]. It is one optional valuation component of the search method.

3 Mathematical formulation

3.1 Contracts and portfolios

At decision time t, let

$$\mathcal{U}_t = \{o_1, \ldots, o_d\}$$

be the eligible listed contracts. Contract o_i contains its identifier, underlying, currency, option right, strike, multiplier, exercise set, expiry, and settlement rule.

A portfolio is a quantity vector

$$q = (q_1, \ldots, q_d) \in \mathcal{Q}^d, \qquad \mathcal{Q} \subset \mathbb{Z},$$

where $q_i > 0$ is long, $q_i < 0$ is short, and $q_i = 0$ omits the contract. Portfolio constraints define the candidate set

$$\mathcal{C}_t = \{q \in \mathcal{Q}^d : g_j(q, c_t) \leq 0,\ j = 1, \ldots, J\},$$

where c_t is the information available at the decision time. The constraints may include gross quantity, leg count, liquidity, capital, margin, and exposure bounds.

The vector q contains no construction order. A sequence of trades is converted to q by summing quantities for each contract. Reordering the same trades leaves q unchanged. Any portfolio model used here must therefore be invariant to the order in which its nonzero positions are serialized.

3.2 Paths and portfolio outcomes

One path is

$$\xi_m = (z_{m,0}, z_{m,1}, \ldots, z_{m,H}),$$

where $z_{m,j}$ contains the market observation, contract prices, and external events at time τ_j. Let ν denote the option-valuation or quote-selection rule, μ the management and exercise rule, α the assignment rule, and ω the fill, cost, accounting, and settlement rules.

The financial evaluator returns

$$R(q, \xi_m; c_t, \nu, \mu, \alpha, \omega) \in \mathbb{R}^p.$$

The result contains leg-level records and portfolio aggregates. Components may include net P&L, pathwise equity, drawdown, turnover, transaction costs, cash, delivered underlying, exercise effects, assignment effects, and terminal inventory.

For a cohort of M paths,

$$\Xi_t = \{\xi_1, \ldots, \xi_M\},$$

an aggregation rule A creates the model target

$$y_t(q) = A\left(\{R(q, \xi_m; c_t, \nu, \mu, \alpha, \omega)\}_{m=1}^{M}\right) \in \mathbb{R}^r.$$

The target can retain several outcome and risk measures. A stated utility function $U(y_t(q))$ supplies a ranking when the experiment requires one ordering.

Model inputs contain q, c_t, and forecast artifacts available at time t. Future observations in Ξ_t are used to compute outcomes. Training, validation, and test periods follow the decision clock, with a purge interval where label horizons overlap.

3.3 Budgeted search

Evaluating every candidate on every path costs approximately

$$|\mathcal{C}_t|\, M\, C_R,$$

where C_R is the average cost of evaluating one candidate on one path. Let B be the number of candidates that can receive full path evaluation, with $B < |\mathcal{C}_t|$. The search procedure chooses an evaluated subset $\mathcal{L}_t \subset \mathcal{C}_t$, $|\mathcal{L}_t| \leq B$, and returns the best evaluated candidate according to U.

On a small candidate pool, exhaustive evaluation identifies

$$q_t^\star = \arg\max_{q \in \mathcal{C}_t} U(y_t(q)).$$

This gives a reference for simple regret,

$$U(y_t(q_t^\star)) - \max_{q \in \mathcal{L}_t} U(y_t(q)),$$

and for recovery of the top k candidates. These quantities make the cost and benefit of learned acquisition observable.

4 Permutation-invariant portfolio models

4.1 Deep Sets baseline

For each nonzero position, construct a feature vector

$$x_i = x(o_i, q_i, c_t).$$

Features can include option right, signed quantity, strike relative to spot, time to exercise boundaries and expiry, multiplier, settlement, mark, bid–ask spread, implied-volatility coordinates, and liquidity measures.

The first nonlinear model is

$$h_i = \phi_\theta(x_i),$$
$$h(q, c_t) = \sum_{i:q_i \neq 0} h_i,$$
$$\widehat{y}_t(q) = \rho_\theta(h(q, c_t), c_t).$$

The shared leg map ϕ_θ and the sum make the output invariant to position order. The signed quantity remains part of each position feature, and the sum can retain information about gross scale and number of positions.

Zaheer et al. characterize permutation-invariant set functions under their stated domain conditions and motivate this form [2]. The application here follows from the financial definition of a portfolio as a quantity assignment without a leg ordering.

The model predicts the outcome vector $y_t(q)$. Training may combine component-wise regression with a pairwise ranking loss among candidates from the same decision state. Calibration of the outcome components and quality of the induced ranking are reported separately.

4.2 Typed graph comparison

For a candidate q at decision state c_t, the implemented graph is

$$G(q, c_t) = (V, E), \qquad V = V_{\text{pos}} \cup V_{\text{underlying}} \cup V_{\text{expiry}} \cup \{v_{\text{portfolio}}\}.$$

Each nonzero quantity creates one position node with feature vector x_i. Typed edges connect that position to its portfolio, underlying, and expiry. The learning adapter adds the three reverse directions so messages travel both ways. Implied volatility and liquidity fields are joined to position nodes by Volt contract identity.

Two relation-specific message-passing layers update the node representations. The portfolio node supplies an order-invariant readout and a prediction head for $\widehat{y}_t(q)$. Graph construction is deterministic for a given candidate–state pair. Serializing its position nodes in another order gives the same financial input. A change to a contract quantity changes the candidate; a change to the contemporaneous market data changes the state.

The fixed comparison uses 8,020 trainable parameters for Deep Sets and 7,404 for the graph network. Path generation and option valuation remain responsibilities of the financial evaluator, which produces the same training outcomes for both models. Their common candidate split, evaluation budget, optimizer budget, and targets isolate the contribution of typed adjacency [4].

The current topology has one underlying and one expiry, so every candidate is a small star graph. Position-to-position, account, inventory, and action relations enter later ablations and the sequential state/action graph.

5 Selecting candidates for evaluation

After an initial evaluated sample, the model predicts outcomes for the remaining pool. An acquisition score can combine predicted utility, uncertainty, coverage of sparsely sampled regions, and path-evaluation cost. The simplest rules select candidates at random, by predicted utility, by uncertainty, or by a greedy utility–diversity score. These establish the baseline for a batch optimizer.

5.1 QUBO batch selection

Suppose the frontier contains n candidates. Let $b_i \in \{0, 1\}$ indicate selection of candidate i, let a_i be its acquisition value, and let $s_{ij} \geq 0$ measure similarity between candidates i and j. For an equal-cost batch of size k, consider

$$\min_{b \in \{0,1\}^n} -\sum_i a_i b_i + \gamma \sum_{i<j} s_{ij} b_i b_j + \rho \left(\sum_i b_i - k \right)^2.$$

The linear term favors candidates with high acquisition value. The quadratic term discourages a batch concentrated in one region of the portfolio space. The final term encodes the batch size.

QUBO is a binary optimization formulation with classical and quantum-inspired solution methods [5]. Its role here is candidate allocation. The financial evaluator supplies outcomes and the constraint system supplies admissibility. The empirical comparison charges matrix construction and solver time to the acquisition method and includes exhaustive solutions on small frontiers, classical heuristics, greedy diversity, and random selection.

5.2 Learning loop

For each decision state, the search proceeds as follows:

1. generate and constrain the candidate pool;
2. evaluate an initial sample over the path cohort;
3. fit the portfolio model;
4. score the remaining candidates;
5. select another batch using one acquisition rule;
6. evaluate the selected batch and update the training data; and
7. stop at the declared evaluation or wall-clock budget.

Every round records the candidate pool, evaluated subset, model checkpoint, scores, selection rule, path cohort, financial assumptions, and results.

6 Empirical design

6.1 Initial study

The initial study uses listed American-style options on one underlying. The candidate generator selects contracts and quantities directly, subject to declared bounds on moneyness or delta, expiry, quantity, spread, volume, open interest, and leg count. Analysis may add strategy labels after generation.

The study fixes one entry time and management rule for every candidate at a decision date. The exercise rule, assignment process, observation grid, valuation method, and transaction-cost model are shared across candidates. Historical and generated path cohorts are analysed separately because they support different interpretations.

A small universe is evaluated exhaustively. It provides the target ranking and shows where portfolio symmetry and offsetting positions reduce duplicate constructions. Later experiments expand the number of contracts, legs, paths, and decision dates one dimension at a time.

6.2 Comparisons

Implemented prediction models are ridge regression over summed leg features, Deep Sets, and a parameter-matched typed graph network. The implemented one-batch acquisition ranks predicted utility and compares with repeated random batches. Stratified, uncertainty, greedy-diversity, and QUBO acquisition remain comparisons on the same candidate pool, initial evaluated sample, path cohort, and compute accounting.

Primary measurements are top-k recovery, simple regret, rank correlation, outcome calibration, path-evaluator calls, total paths evaluated, and elapsed time. Secondary measurements include coverage across exercise and assignment states, diversity of the evaluated portfolios, transaction-cost sensitivity, and stability under alternative exercise and assignment rules.

6.3 Time and reproducibility

Decision dates are divided chronologically into training, validation, and test periods. Overlapping outcome horizons are assigned to one split or separated by a purge interval. Feature scaling and model selection use the training and validation periods. The final configuration is applied once to the test period.

Each result records the listed universe, decision timestamp, candidate constraints, path source, pricing and quote rules, exercise and assignment rules, execution assumptions, target definition, model version, random seeds, and selection history. Altering any of these inputs creates another experiment record while the portfolio quantity vector remains available for comparison.

7 Pilot integration results

The pilot tests the complete identity–evaluation–learning path on one fixed seed. It uses twelve American contracts, unit quantities, at most three legs, and a bound excluding portfolios whose expiration loss diverges as spot rises. Software generates 1,335 canonical portfolios. Each quantity vector passes through the Volt contract validator, portfolio canonicalizer, payoff projection, and graph exporter. The research tensor joins implied volatility and liquidity fields to the resulting Volt portfolio hash.

The generated decision state has spot 100, 30 calendar days to expiry, a 22-point grid, and 512 GBM paths. A CRR lattice supplies American marks and holder exercise decisions; the same decisions generate short assignment. The target contains mean P&L, lower-tail mean, P&L standard deviation, and profit probability. Utility is mean P&L plus one quarter of the lower-tail mean. The model sees 20% of the exhaustive labels initially and acquires another 10%. This requires 205,312 candidate–path evaluations, versus 683,520 for exhaustive evaluation.

On the remaining candidates, Deep Sets attains Spearman rank correlation 0.972, recovers all of the exhaustive top 20 after acquisition, and finds the exhaustive optimum. The typed graph network attains 0.953, recovers 18 of the top 20, and also finds the optimum. Ridge regression on summed leg features attains 0.386 rank correlation and recovers 18 of the top 20. The selected generated portfolio is long one 105 call and long one 110 call. It has no exact match in the deliberately small reference catalogue used by the explanation generator. That observation identifies a candidate for further comparison; it supplies no novelty or robustness result.

Across three candidate-split and initialization seeds on the same generated path cohort, mean rank correlation is 0.968 for Deep Sets and 0.951 for the graph network. Mean top-20 recall is 1.00 and 0.933; the exhaustive optimum is recovered in three and two runs, respectively. Deep Sets is stronger under the current fixed-expiry star relations.

The first market-data check uses the SPY chain at the 14 July 2026 close and options expiring 24 July. Daily option closes are proxy marks. Six calls and six puts nearest spot pass the declared volume, transaction-count, intrinsic value, and American implied-volatility filters. A 512-path cohort uses the eight validated daily observation times and the median selected-contract implied volatility. The 22 July equity partition is absent and is recorded in the manifest; no imputation is applied.

Deep Sets reaches rank correlation 0.996, full top-20 recovery, and zero simple regret under the generated cohort. The typed graph network reaches 0.994, full recovery, and zero regret. The corresponding ridge values are 0.189, four of 20, and 0.0447 per share. Both nonlinear models select short one 750 call and long one 753 call, an ordinary bear call spread identified after generation. SPY closes at 738.93 on expiry; under the proxy-mark, funding, exercise, assignment, and cost rules, the position earns 1.989 per share. Across all candidates, cohort utility and this one realized-path P&L have rank correlation 0.801. This run checks the pipeline; return estimation requires executable quotes and multiple decision dates.

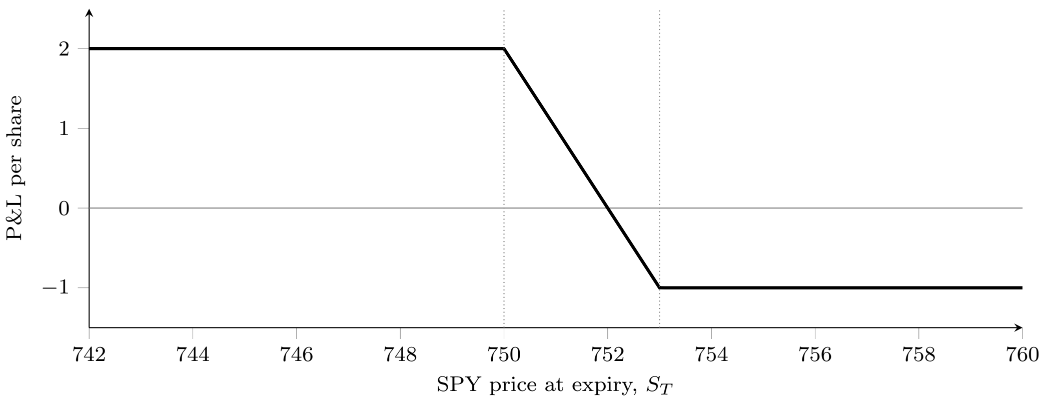

Figure 2: Expiration P&L for the SPY pilot selection: short one 750 call and long one 753 call for a 2.00 proxy credit, before financing and transaction costs. The maximum profit is 2.00, the maximum loss is 1.00, and the break-even spot is 752.

Three energy continuation checks use USO, XLE, and XOM chains over the same 14–24 July interval with lower declared liquidity thresholds. The selected positions are respectively a 119/121 bull call spread, a 56/57 bull call spread, and a 141/142 bull put spread. All three are conventional verticals despite catalogue-free generation. They earn positive P&L on the single observed path, while the cross-candidate rank correlations between generated-cohort utility and realized P&L are −0.530, −0.635, and −0.623. Deep Sets rank correlations against the generated labels are 0.996, 0.968, and 0.967; graph-network correlations are 0.983, 0.938, and 0.899. The models learn the supplied cohort labels; the negative realized correlations direct the next work toward path calibration and multiple decision dates.

Two earlier decision cohorts are registered for that multi-date test. The tariff cohort fixes information at the 1 April 2025 close and observes the 2–9 April response window described in the corresponding market study [6]. The initial US–Iran cohort fixes information at the 27 February 2026 close and begins its response window on 2 March [7]. These dates define the information sets and evaluation windows; causal attribution requires a separate design. The tariff archive currently lacks the completion certificates required for a primary result. The Iran option aggregates are recoverable from source files, but their derived partitions still require reconciliation.

Table 1: Fixed-seed pilot ranking diagnostics. Surrogate correlations compare predicted and exhaustive generated-cohort utility on candidates outside the initial evaluated set. Realized correlation compares exhaustive cohort utility with P&L on one later observed path.

Cohort	Set ρ	Graph ρ	Ridge ρ	Top-20 set/graph	Realized ρ
Generated	0.972	0.953	0.386	20/18	—
SPY	0.996	0.994	0.189	20/20	0.801
USO	0.996	0.983	0.221	20/20	-0.530
XLE	0.968	0.938	0.231	20/20	-0.635
XOM	0.967	0.899	0.277	20/17	-0.623

The fixed-horizon management check separates evaluation from expiration. It enters SPY options on 14 July, closes live positions on 24 July, and uses contracts expiring 31 July. Seven dates have a common stock close, all selected option closes, and valid American implied-volatility inversions. The 22 July stock partition is absent; the 23 July option cross-section fails an intrinsic-value check. Both dates are excluded with recorded reasons.

An entry-only gamma screen selects short one 751 put and long one 752 put for a 0.51 proxy debit. The rule maximizes entry gamma relative to absolute daily theta and bounded loss, subject to positive gamma, negative theta, absolute delta below 0.35, a debit entry, and maximum expiration loss below 15 per share. The selection depends only on entry data. On the observed path, unhedged P&L is 0.6734 per share. Daily delta hedging produces 0.4660; the 0.10 delta band makes no hedge trades because net delta remains inside the band.

For every path and policy, the accounting identity is

$$\begin{aligned}\Pi = \Pi_\Delta + \Pi_\Gamma + \Pi_\Theta + \Pi_{\text{vega}} + \Pi_{\text{residual}} + \Pi_{\text{financing}} \\ + \Pi_{\text{hedge}} - C_{\text{option}} - C_{\text{hedge}}.\end{aligned}$$

The residual contains discrete moves, lattice approximation, mark/model basis, and higher-order effects. The analytics record expiration geometry; delta, gamma, theta, vega, rho, vanna, volga, charm, and speed; distributional and tail measures; drawdown and recovery; exercise, assignment, closes, turnover, liquidity; and a common spot–volatility–time stress surface.

Table 2: Management results for the selected 751/752 bear put spread on a separate 128-path, fixed-seed model cohort. Values are per share. Expected shortfall is reported as a positive loss.

Policy	Mean P&L	P&L sd	ES 95%	Mean drawdown	Cost
Unhedged	-0.0798	0.5873	1.7088	0.3358	0.0260
Daily delta	-0.0554	0.1377	0.4461	0.1101	0.0347
Delta band 0.1	-0.0042	0.3244	0.5214	0.2746	0.0309

The generated cohort uses fixed contract implied volatilities and legwise CRR exercise decisions. Tail loss can exceed the spread's expiration maximum loss when legs settle at different dates under that policy. The observed mark-to-market drawdown also exceeds the expiration bound, exposing the effect of non-synchronous close proxies. These diagnostics motivate portfolio-level exercise sensitivity and executable quote paths before the management labels are used for strategy claims.

7.1 Generated volatility search

A volatility forecast changes the search target. Directional terminal P&L alone cannot distinguish a view that realized volatility will exceed implied volatility from a view on the terminal price. The generated volatility study therefore evaluates every candidate under scheduled aggregate-delta hedging. Its entry exposure vector is

$$g(q) = \sum_i q_i \left(\Delta_i, \Gamma_i, \Theta_i, \mathcal{V}_i\right),$$

where theta is measured per calendar day and vega per volatility point. These signed leg exposures are supplied to both invariant models. The graph also stores their sums on the portfolio node.

The decision state has spot 100, flat 20% entry implied volatility, ten American options at strikes $90, 95, 100, 105, 110$, 45 days to expiry, and a 21-day evaluation horizon. Canonical enumeration produces 2,890 one- through four-leg portfolios with quantities in $\{-1, +1\}$. The compiler retains 283 with finite expiration loss no greater than 5 per share, absolute entry delta no greater than 0.20, and nonzero gamma and vega. Contract ordering cannot create a second candidate.

Two equal 48-path cohorts have zero median terminal return and annualized realized volatilities of 10% and 35%. Implied volatility remains at 20%, isolating realized volatility from a separate vega forecast. The observation grid has 22 points. A 40-step CRR lattice supplies aligned American marks and Greeks; scheduled hedging resets aggregate delta at every observation. Net P&L includes option and hedge costs. Conditional utility is mean net P&L plus one quarter of the lower-decile mean.

Under the low-realized-volatility cohort, the exhaustive optimum is short the 100 call and put, with a long 90 put and 110 call. Its entry delta, gamma, daily theta, and one-point vega are -0.0338, -0.4762, 0.0396, and -0.1757. Mean P&L is 0.7234 per share and the lower-decile mean is 0.4336. The same portfolio has mean P&L -1.1435 in the high-realized-volatility cohort. Its low-volatility attribution includes 1.0307 of theta P&L, -0.2907 of gamma P&L, and 0.0639 of transaction costs.

Under the high-realized-volatility cohort, the exhaustive optimum is long the calls and puts at both 95 and 105. Its entry delta, gamma, daily theta, and one-point vega are 0.0987, 0.0498, -0.0928, and 0.4078. Mean P&L is 3.4094 and the lower-decile mean is 0.9750. The same portfolio has mean P&L −1.8156 in the low-volatility cohort. Its high-volatility attribution includes 5.2579 of gamma P&L, −1.8950 of theta P&L, and 0.1103 of transaction costs.

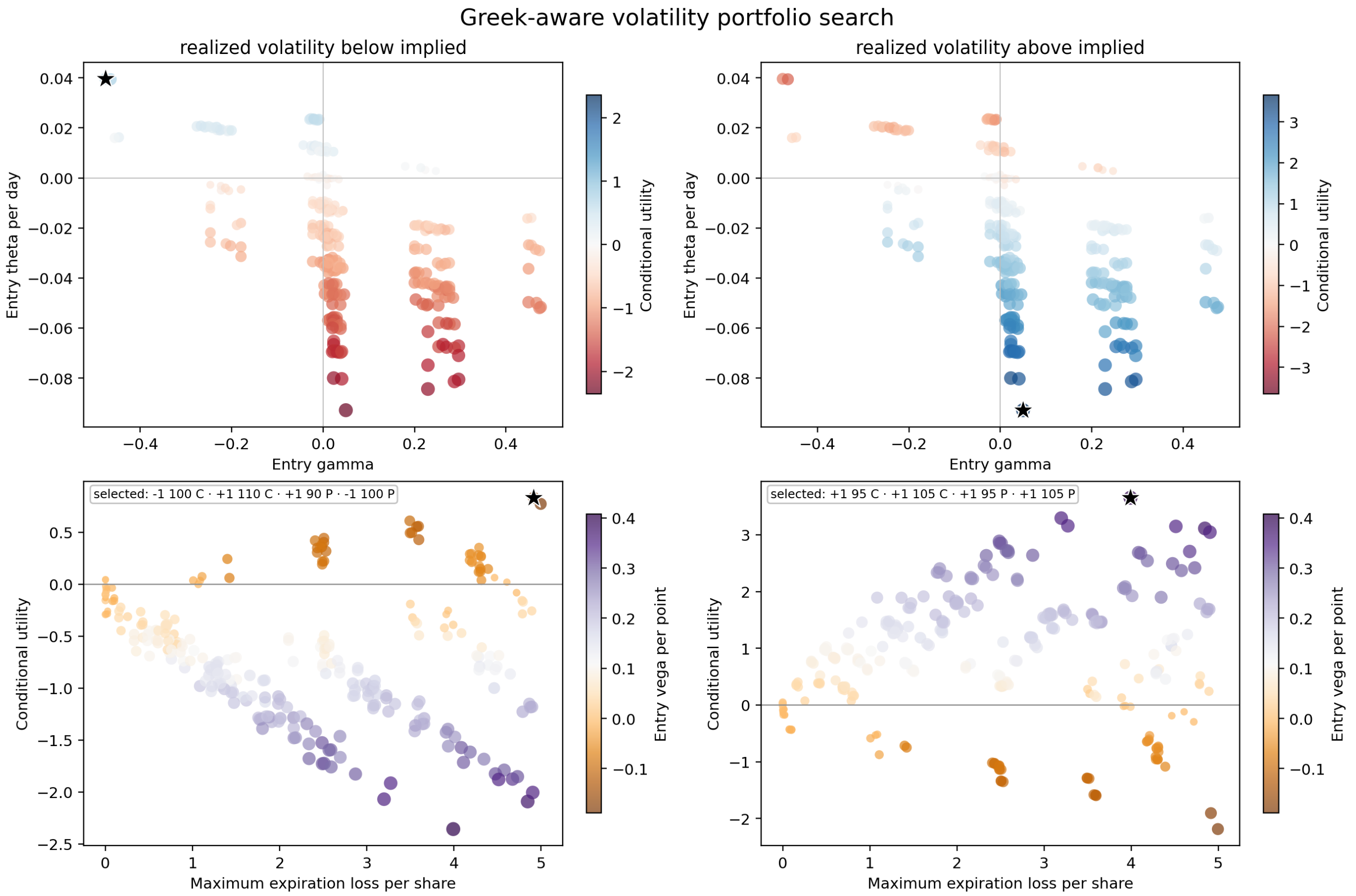

Figure 3: The 283 compiled portfolio graphs under the two conditional objectives. The upper panels place each candidate by entry gamma and daily theta; marker area increases with absolute vega and colour records conditional utility. The lower panels compare maximum expiration loss with utility and colour points by entry vega. A star marks the exhaustive conditional optimum.

Table 3: Generated volatility ranking and compute results. Correlations and top-20 recall use candidates outside the initial evaluated set. Every model recovers the exhaustive optimum after acquisition.

Realized vol.	Model	Spearman ρ	Top-20 recall	Simple regret	Train/search s
10%	Deep Sets	0.966	0.95	0	0.521
	Typed graph	0.976	0.95	0	0.536
	Ridge	0.914	0.90	0	0.003
35%	Deep Sets	0.986	1.00	0	0.165
	Typed graph	0.985	1.00	0	0.782
	Ridge	0.955	1.00	0	0.002

Candidate compilation takes 0.242 seconds, graph export 0.137, scenario construction 6.230, pathwise labels 72.647, and model encoding 0.052; the full run takes 81.809 seconds. American path valuation and Greek attribution account for most compute. The learned models consume less than one second per regime at this scale. The 48 paths per regime and accelerated lattice make this a generated mechanism check. More paths, lattice-resolution sensitivity, volatility-surface dynamics, and empirical quote paths precede a trading-performance claim.

A later compiler-aware benchmark separates the common calculation from portfolio aggregation. Fourteen American contracts, 64 paths, and nine observations require 33.317 seconds for chain construction, scenario marks, and contract-leg compilation. Aggregating 4,000 daily-hedged portfolios then takes 0.152 seconds. A Deep Sets search with a 30% label budget takes 1.145 seconds after the common work; the relation GNN takes 13.763 seconds. Their rank correlations are 0.9945 and 0.9920, but neither is faster than direct enumeration. Because the optimum is already in the initial evaluated sample, the acquisition step receives no credit for the zero-regret result.

The first three-date event transfer is also a development result. The GNN has mean final-date rank correlations 0.7486 for the tariff state and 0.5476 for the conflict state, but no model recovers an optimum. The first execution exposed a constant-column scaling defect on the final date; the corrected preprocessing is now locked and requires a new date for a prospective test.

Validated NYMEX session data supplies contemporaneous CL, RB, and BZ evidence for the July energy checks. Within a decision state this evidence is constant across candidates. It supports explanation and future multi-date conditioning but cannot explain the within-state ranking. The available research inventory contains outright futures and no identified options-on-futures series. Futures options would require contract, exercise, delivery, margin, and settlement semantics beyond the current equity-option pilot.

Every selected portfolio produces a versioned evidence packet containing its exact positions, mark authority, payoff geometry, entry Greeks, path statistics, leg attribution, exercise and assignment counts, and limited catalogue comparison. Human-facing text is rendered from those computed fields. An LLM explanation is required to bind every numerical claim to the packet and to reserve claims about execution, causal event attribution, robustness, and innovation for the corresponding evidence.

8 Current implementation

Volt currently supplies the financial identity and accounting components needed by this formulation. Contracts carry European, Bermudan, or American exercise schedules. Position netting converts repeated positions into a single quantity per contract. The runtime handles close, exercise, assignment, expiry, cash, delivered underlying, and journaled state changes. The market-data component gives historical and generated option observations a shared versioned interface with declared price authority. Scenario paths retain ordered points, source identity, and reproducibility seeds. The scenario evaluator consumes supplied option marks or quotes and reports fills, costs, pathwise P&L, terminal P&L, and leg attribution. Candidate actions resolve strike, moneyness, delta, and expiry buckets to listed contracts.

The supervised graph example evaluates a bounded terminal-payoff fixture. A general graph-export command now accepts declared marked contracts and quantity vectors, then returns Volt portfolio hashes and graph samples. The pilot research driver adds a generated cohort builder, CRR American marks, fixed exercise and assignment policies, constrained candidate enumeration, reusable contract-leg compilation, path-target aggregation, Deep Sets, typed graph, and ridge models, one batch-acquisition step, market-data adapters, and evidence packets. The graph model consumes Volt nodes and typed edges and saves a transferable portfolio–state embedding. The management implementation adds fixed-horizon option closes, unhedged and delta-hedged policies, self-financing hedge accounting, full P&L attribution, path and distribution risk measures, higher-order entry Greeks, and stress surfaces.

The runtime research driver extends the current implementation to multiple expiries, cash, delivered underlying, completed option lifecycles, account limits, compiler action nodes, and a composed legal mask. It trains state-relative action scorers and records graph construction, path evaluation, training, and inference time. Section 9 and the sequential-control paper report that experiment separately from static candidate search.

The next integration work is:

1. add enough chronologically separated decision dates to test the locked preprocessing beyond the three-date development cohorts;
2. replace close proxies with executable quote rules where quote coverage permits;
3. compare legwise exercise, portfolio exercise, and stochastic assignment policies;
4. run iterative random, uncertainty, utility, and diversity acquisition baselines;
5. run edge, relation, account-limit, and action-feature ablations on the implemented runtime graph; and
6. test QUBO batch selection after the classical acquisition baselines.

9 Extension to sequential control

Static discovery chooses an entry portfolio under a fixed management rule. Sequential control chooses actions as market observations, fills, exercise, assignment, and inventory change the state. Its data consist of state–action–reward trajectories.

The Volt state/action graph and legal-action mask provide part of that environment interface. A reinforcement-learning study additionally requires a reward definition, transition scheduler, trajectory store, policy and value models, and out-of-time policy evaluation. The set or graph encoder developed for static portfolios may initialize the policy representation, while control performance determines the value of that transfer.

QUBO has a separate possible role in control: allocating a limited rollout or replay budget among frontier states. That experiment follows the static candidate-allocation comparison and uses its own baselines.

10 Discussion

The candidate pool sets the discovery domain. The path cohort, valuation method, management rule, and outcome measure set the preference represented by the labels. Reported results are conditional on these choices.

American options make this dependence visible. Early exercise changes the cash and inventory path; short assignment can create states the holder did not choose; coarse observation grids change the opportunities available to the management rule. Reported results therefore require sensitivity to the exercise policy, assignment process, grid, quotes, and execution costs.

The empirical test compares the portfolios found by a learned invariant model with those found by simple search rules at the same evaluation budget. Graph and QUBO components are measured by incremental improvement over their set-model and acquisition baselines.

11 Conclusion

Searching directly over listed option contracts expands the candidate set beyond predefined strategy templates. Volt compiles common American-option leg paths before evaluating portfolios. A learned surrogate enters only when portfolio-specific or sequential work remains costly after that reuse. An acquisition rule then decides which candidates receive the next evaluations.

The current experiments compare Deep Sets with conventional portfolio models, measure ranking quality, and charge shared and candidate-specific computation separately. Direct enumeration wins the present compute comparison, and the chronological development cohort recovers no exact optimum. Graph structure, sequential management, and QUBO allocation show no performance benefit in the current evidence.

References

[1] Options Industry Council, *Options Strategies Quick Guide*, https://www.theocc.com/getcontentasset/f34f8a0d-806f-4f1a-adf7-d49d8d94b16e/dfc3d011-8f63-43f6-9ed8-4b444333a1d0/option-strategies-quick-guide.pdf.

[2] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Póczos, R. Salakhutdinov, and A. Smola, *Deep Sets*, Advances in Neural Information Processing Systems, 2017, https://arxiv.org/abs/1703.06114.

[3] F. A. Longstaff and E. S. Schwartz, *Valuing American Options by Simulation: A Simple Least-Squares Approach*, The Review of Financial Studies, 14(1), pp. 113–147, 2001, https://doi.org/10.1093/rfs/14.1.113.

[4] P. W. Battaglia et al., *Relational Inductive Biases, Deep Learning, and Graph Networks*, 2018, https://arxiv.org/abs/1806.01261.

[5] G. Kochenberger, J.-K. Hao, F. Glover, M. Lewis, Z. Lü, H. Wang, and Y. Wang, *The Unconstrained Binary Quadratic Programming Problem: A Survey*, Journal of Combinatorial Optimization, 28, pp. 58–81, 2014, https://doi.org/10.1007/s10878-014-9734-0.

[6] StrategyNet, *Liberation Day Tariffs: Market Response to Announcement and Court Ruling*, https://strategynet.ai/insights/liberation-day-tariffs-factor-evolution.

[7] StrategyNet, *US–Iran Conflict: Factor Evolution*, https://strategynet.ai/insights/us-iran-conflict-factor-evolution-2026.