DECORRELATED IS NOT DIVERSIFIED: BEHAVIORAL DIVERSITY IN THE SEARCH FOR FORMULAIC ALPHA FACTORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Formulaic alpha factors are short, interpretable expressions over market data whose cross-sectional values forecast returns. Automated miners now produce thousands of candidates per run, yet the final set is still ranked by the information coefficient (IC) and de-duplicated by the correlation between factor values (mutual-IC). This criterion misses a form of redundancy: factors with nearly uncorrelated values can earn their IC on the same stretch of the capitalization spectrum and on the same rhythm. We introduce the Exposure–Cycle Residual (ECR), which describes a factor by where on the capitalization spectrum its ranking has been effective and by the period of that tilt. During search, ECR scales each candidate's fitness by the share of its behavior that an online archive does not yet cover, so rediscovering a known behavior earns only a fraction of the fitness. We re-run eight miners in one formula space over 344 tick-derived daily fields of the Chinese A-share market, with a shared screen and combiner. Adding ECR to the reward raises the out-of-sample label Sharpe of the combined factor while its IC barely changes. The same pattern appears for tournament GP on RPN, where the reward is the only change. Behavioral diversity is most useful inside the search reward, while selection should keep the IC order.

1 INTRODUCTION

Formulaic alpha factors, short expressions over market data whose cross-sectional values forecast forward returns, remain the interpretable workhorse of quantitative equity pipelines (Kakushadze, 2016). The search for them has moved from genetic programming (Zhang et al., 2020; Cui et al., 2021) to reinforcement-learning policies over reverse-Polish (RPN) tokens (Yu et al., 2023) and to generative flow networks (Chen et al., 2026), and a single run now yields thousands of candidates. The rule that decides which candidates survive has not moved. Candidates are ranked by the information coefficient (IC), the average cross-sectional correlation between factor values and forward returns (Grinold & Kahn, 2000), and duplicates are removed by the pairwise correlation of factor values (mutual-IC). A set chosen this way is decorrelated in its values, yet it can still be concentrated in its behavior. This paper asks where that distinction should enter the pipeline; our experiments point to the search reward rather than the final screen.

Mutual-IC removes redundancy only among factor values. Two factors with nearly uncorrelated values can still earn their IC on the same stretch of the capitalization spectrum and switch on the same rhythm, and factor books with such shared exposures are a known source of regime risk (Asness et al., 2017; Arnott et al., 2019). This concentration is visible in our runs (Figure 1). When the candidates of batched search, an IC-driven batched REINFORCE miner, are admitted in decreasing IC order, the mutual-IC screen has to relax its cap before the set fills, and the kept factors sit in a narrow band of exposure half-lives. The same miner trained with our reward fills the set at a stricter cap, and its picks spread along the half-life axis. The caps are reported with the other miners in Section 5.2.

The blind spot is also built into the objective on which miners are trained. Miners maximize a formula backtest (Section 3.3) that pays again for any behavior the training window rewards, so the pool fills with formulas that differ in syntax and in values but not in where their ranking works.

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.

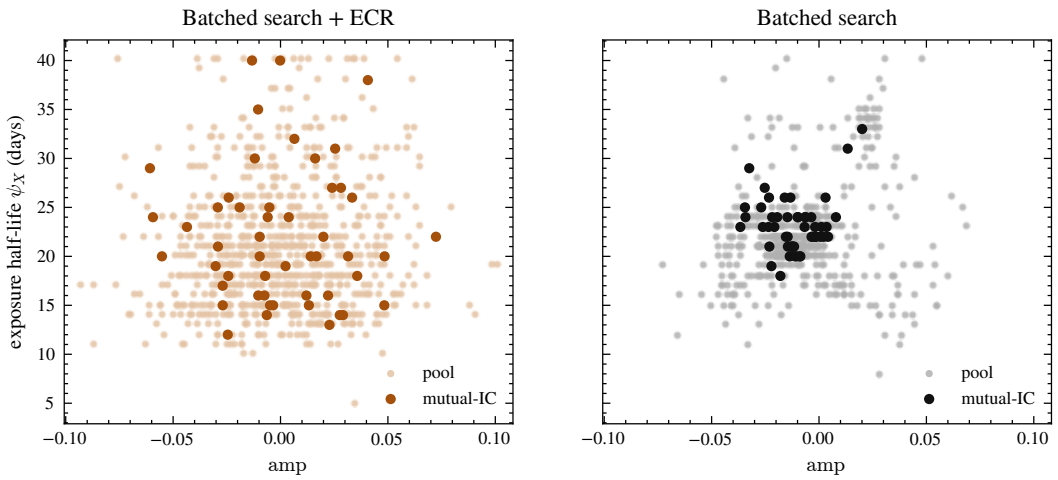

Figure 1: Behavioral concentration under IC-driven search (training window). Each point is a candidate formula placed by two coordinates of its exposure path (Section 4.3): amp summarizes where on the capitalization spectrum its ranking has been effective, and ψ_X is the half-life of that exposure. Filled markers are the factors kept by the mutual-IC screen. With ECR in the search reward (left), they spread along the half-life axis; without it (right), they sit in a narrow band. Both clouds are pools after the shared eligibility filters; the right-hand pool is further truncated by training $|\overline{\mathrm{IC}}|$ (Section 5.1).

Existing diversity terms compare formulas by their values (Zhang et al., 2020; Chen et al., 2026) or by their structure (Chen et al., 2026), and neither can tell whether a new formula tilts with the same period as those already found. We therefore describe each formula by its Exposure–Cycle Residual (ECR) signature, which records where on the capitalization spectrum its ranking has been effective and the rhythm of that tilt, and we scale its fitness by the residual energy of this signature against an online archive. A candidate is paid in full when its exposure shape or its tilt rhythm is new, and only a fraction once both are covered. Under the standard mutual-IC screen, the term raises IC, top-100 label return, and label Sharpe for all three miners it is added to. The gains are large for batched search and for tournament GP on RPN, where the reward is the only change, and small for an AlphaGen-style REINFORCE miner. In both large cases, return rises far more than IC (Section 5.2).

Where diversity enters matters as much as whether it is used. Farthest-point sampling on the ECR coordinates selects a spread-out set without looking at IC. It raises label Sharpe only for the controls mined without the residual, and it lowers the label Sharpe of the pools mined with it. Once the reward has shaped the pool, selection should keep the IC order.

Comparing miners is itself difficult, because each system has been evaluated on its own footing (a few price–volume fields, its own pool reward, and its own combiner), and none at the scale at which a factor pipeline operates. We therefore re-run eight miners in one formula space over 344 tick-derived daily fields, with the same eligibility filters, selection rules, dynamic linear combiner, and formula backtest, and with the same number of factors in every portfolio. Section 5.1 and Appendix F state what each re-implementation changes relative to the system it follows; we make no claim against published results.

Our contributions are as follows.

- **A behavioral reward.** We propose ECR, a descriptor of where and on what rhythm a factor's ranking works along the capitalization spectrum, and use its residual energy against an online archive as a multiplicative term in the search reward. Under mutual-IC screening, the term improves all three miners it is added to (Sections 4.2 and 5.2).
- **A diagnosis of concentration.** We use two coordinates of the same descriptor as a diagnostic plane rather than as a selection rule, and show that IC-driven search concentrates

the candidate pool, that the gains from ECR appear in return far more than in IC, and that IC-blind selection on behavior helps only pools mined without the residual (Section 5.2).

- **A controlled comparison.** We compare eight miners from four families (genetic programming, policy gradient, generative flow networks, and uniform RPN sample) under one pipeline over 344 fields, with shared filters, selection rules, combiner, and formula backtest (Section 5).

2 Related work

Formulaic alpha mining. Automated search for formulaic factors began with genetic programming over expression trees (Zhang et al., 2020; Cui et al., 2021). AlphaGen casts factor construction as sequential decisions over RPN tokens and rewards the IC increment of a growing linear combination (Yu et al., 2023); later reinforcement-learning miners bound the variance of the policy gradient (Zhao et al., 2025) or learn distributional value estimates (Zhu & Zhu, 2025). AlphaForge pairs a generative–predictive proposer with a dynamic linear combiner (Shi et al., 2025), AlphaSAGE samples formulas with a generative flow network conditioned on a structure-aware graph encoder (Chen et al., 2026), and language-model miners propose and refine formulas from natural-language hypotheses (Wang et al., 2025; Tang et al., 2025). These systems differ mainly in how they propose formulas. Which formulas survive is still decided by IC and by the correlation of factor values.

Diversity in alpha mining. Diversity has entered alpha search mainly through factor values. AutoAlpha brings quality-diversity search to genetic programming: it penalizes the fitness of a candidate whose values, summarized by their first principal component, are too similar to alphas already recorded (Zhang et al., 2020). AlphaGen's combination reward discourages redundant values implicitly (Yu et al., 2023). AlphaSAGE adds a novelty term, one minus the largest absolute correlation with an online library, and a structure-aware term that aligns expression embeddings with the distance between factor outputs, which it calls a behavioral distance (Chen et al., 2026). In our terms, each of these devices, like mutual-IC at selection, compares factors through their values. None of them records where on the capitalization spectrum a ranking has been effective, or how persistent that tilt is. ECR adds this information to the reward.

Behavioral diversity in search. Novelty search and quality-diversity (QD) algorithms describe candidates by what they do rather than by how they are encoded, and keep an archive of the behaviors already found (Lehman & Stanley, 2011; Mouret & Clune, 2015; Pugh et al., 2016). ECR follows this idea with a behavior descriptor built for cross-sectional factors. It departs from standard QD in two ways. Behavior enters as a soft multiplicative residual on a predictive fitness, not as a separate objective or a partition into niches. And the final factor set is chosen in IC order instead of being sampled across niches, a choice that our experiments test directly (Section 5.2).

3 Preliminaries

3.1 Formulaic factors

We consider N stocks and T trading dates. On date t, stock i has a feature vector drawn from a fixed set of market fields, a capitalization $C_{i,t}$, and a label $y_{i,t}$: the return from the close on date t to the next close, standardized within capitalization buckets (Appendix D). The label shares the date index with the factor. Every information coefficient, top-100 mean, and Sharpe ratio in this paper is a moment of this label, not a price return. A formulaic factor is a short expression f over the fields; evaluating it yields a panel $f_{i,t}$, with larger values on the long side. Entries may be missing, so every correlation below uses only the stocks on which both arguments are finite, and skips any date on which fewer than two such stocks remain. All miners draw expressions from the same stack machine, operator set, and 344 fields, so two factors differ only in their expressions. The field concepts and the operator categories are in Appendix C.

3.2 IC AND MUTUAL-IC

The predictive score of a factor is its information coefficient. On date t it is the cross-sectional Pearson correlation between factor values and the label,

$$\mathrm{IC}_t(f) = \mathrm{corr}_i(f_{\cdot,t},\, y_{\cdot,t})\,, \tag{1}$$

and the ranking score $\overline{\mathrm{IC}}(f)$ averages it over the dates on which Equation 1 is defined. Mutual-IC applies the same correlation to a pair of factors. With $\mathcal{T}(f,g)$ the dates on which both panels are defined,

$$\mathrm{mutIC}(f,g) = \frac{1}{|\mathcal{T}(f,g)|}\sum_{t\in\mathcal{T}(f,g)} \mathrm{corr}_i(f_{\cdot,t},\, g_{\cdot,t})\,. \tag{2}$$

Screens compare $|\mathrm{mutIC}(f,g)|$ with a threshold. Both quantities are averages of cross-sectional correlations, so neither records where on the capitalization spectrum a factor's ranking works, or how that changes over time.

3.3 MINING, SCREENING, AND COMBINATION

A miner searches the space of expressions for a high fitness $F(f)$, the formula backtest. It takes the end of the ranking indicated by the sign of the training IC (the high scores if the sign is positive, the low scores otherwise), computes the mean label $r_{\mathrm{book}}(f)$ of the names on that end, and combines it with $|\overline{\mathrm{IC}}(f)|$. Batched search uses $F(f) = 8|\overline{\mathrm{IC}}(f)| + 4\,r_{\mathrm{book}}(f)$. Tournament GP on RPN and AlphaGen-style REINFORCE use this same formula backtest. F is a training score, not a tradable net value. For the reinforcement-learning miners, the reward is F of one formula, not the combination-IC increment of AlphaGen (Yu et al., 2023).

The search returns a candidate pool. The mutual-IC screen keeps K factors from it: walking the candidates in decreasing $|\overline{\mathrm{IC}}|$, it admits a candidate when its absolute mutual-IC with every admitted factor is at most a cap τ, and it relaxes τ along a fixed ladder until K factors are admitted. Farthest-point selection (Section 4.3) ignores IC and draws K factors that are spread out in behavior. Admitted factors are standardized cross-sectionally on each date and passed to one dynamic linear combiner. All experiments hold the eligibility filters and the combiner fixed, and vary only how F enters the miner's objective and which selection rule is used.

4 METHOD

Mutual-IC compares what two factors output; it cannot tell whether they earn their IC on the same part of the market. We therefore describe each factor by how its ranking efficacy moves along the capitalization spectrum over time (Section 4.1). This description is used in two places with different roles (Figure 2). Its full signature enters the search reward as a residual against an archive of behaviors already found (Section 4.2); this is our method. Two scalar coordinates compressed from the same exposure path place factors on a plane, which we use to diagnose concentration and to build an IC-blind selection baseline (Section 4.3).

4.1 A BEHAVIORAL DESCRIPTOR

On each date, the stocks with finite capitalization are sorted from small to large and cut into five equal-count layers, layer 1 holding the smallest names; a date with fewer than five such stocks is skipped. Inside each layer, the factor ranking forms a long leg and a short leg. The lagged strength of each layer's long–short spread and the layer's capitalization centroid, both demeaned across layers, give weights $w_{t,k}$ and centers $\tilde{c}_{t,k}$ that compose the net exposure

$$X_t = \sum_{k=1}^{5} w_{t,k}\,\tilde{c}_{t,k}, \tag{3}$$

defined on dates with at least one finite strength and one finite centroid. Because the label is already standardized within capitalization buckets, X_t measures on which stretch of the spectrum the ranking is effective: it is positive when recent efficacy sits in the large-cap layers and negative when it sits in the small-cap layers.

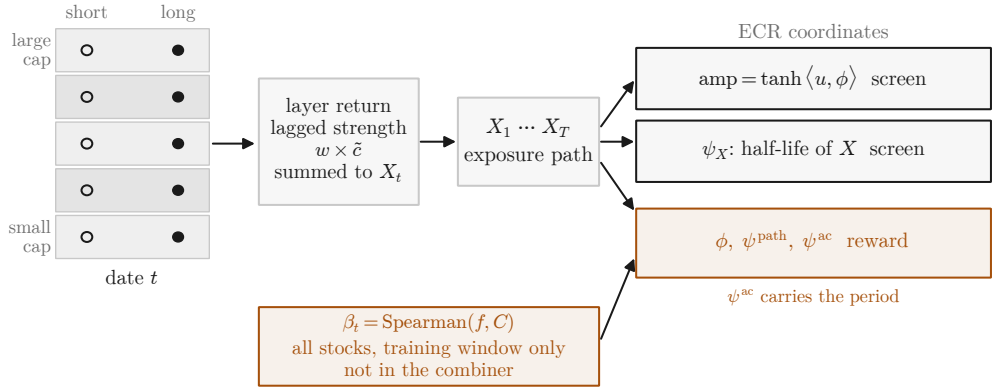

Figure 2: One date's cross-section and the resulting exposure path. Open circles are the short leg and filled circles the long leg; the bottom row is layer 1, the smallest names. The ECR signature $(\phi, \psi^{\text{path}}, \psi^{\text{ac}})$ enters the search reward: ϕ is the shape of $\{X_t\}$, and ψ^{path} and ψ^{ac} are read from β_t on the training window only. The ECR coordinates (amp, ψ_X) are used only for diagnosis and for the IC-blind selection baseline.

From the training window we read three unit-norm signatures. The amplitude signature ϕ is the demeaned path $\{X_t\}$ scaled to unit length, so it records the shape of the exposure over time. The other two come from the size loading β_t, the Spearman correlation between the factor and capitalization on date t, after smoothing. The path signature ψ^{path} is the demeaned, normalized path of β_t, and the autocorrelation signature $\psi^{\text{ac}} \in \mathbb{R}^{40}$ collects its normalized autocorrelations at lags 1 to 40; a factor that swings between small- and large-cap tilts every p days has large entries near lag p. Leg sizes, the smoothing recursions, and exact definitions are in Appendix A.

4.2 Residual energy in the search reward

A behavior that the archive already covers should no longer be paid in full. The archive $\mathcal{A}$ stores the signatures of the formulas admitted so far, together with an orthonormal basis of their amplitude signatures. For a new formula, the amplitude residual res_{amp} is the fraction of $\|\phi\|_2^2$ that remains after projecting ϕ off that basis: near 0 when ϕ lies in the span of stored signatures, and near 1 when it points in a new direction. The cycle residual compares the new formula with each stored formula j. Phase coverage is the positive part of the cosine between the two ψ^{path} vectors, so a tilt in the opposite phase does not count as covered; frequency coverage is the absolute cosine between the two ψ^{ac} vectors. When both are available, cover_j is the smaller of the two, and

$$\text{res}_{\text{cyc}} = 1 - \max_j \text{cover}_j. \tag{4}$$

A formula counts as new when either its exposure shape or its tilt rhythm is new, so the reward uses $\text{fill} = \max(\text{res}_{\text{amp}}, \text{res}_{\text{cyc}})$. The fitness is kept and rescaled,

$$r(f) = F(f) \cdot \big(\alpha + (1 - \alpha)\,\text{fill}(f)\big), \qquad \alpha = 0.3. \tag{5}$$

A covered behavior therefore earns the fraction α of its fitness, and a new one earns nearly all of it. The multiplicative form has two further consequences. Novelty never substitutes for predictive power, because the term only rescales F. And among equally new candidates, the ordering by fitness is unchanged.

Tournament GP on RPN + ECR and AlphaGen-style REINFORCE + ECR optimize Equation 5 alone (Algorithm 1): their updates have no other term, and the final screen sees every valid formula in the search log. Batched search + ECR uses the same residual but changes more of the loop. A formula enters the archive only if it is new on at least one axis, with thresholds 0.15 on res_{amp} and 0.20 on res_{cyc} (Appendix A); the screen sees only admitted formulas; and the reward keeps the fill factor in Equation 5, then adds a shape term and subtracts a batch penalty (Appendix E). The

Algorithm 1 One search step with the ECR reward (tournament GP on RPN + ECR and AlphaGen-style REINFORCE + ECR).

Require: archive $\mathcal{A}$ (stored signatures and amplitude basis), weight α
1: Propose a formula f and evaluate its fitness $F(f)$ on the training window
2: Compute the signatures ϕ, ψ^{path}, and ψ^{ac} of f ▷ Appendix A
3: fill $\leftarrow \max(\text{res}_{\text{amp}}, \text{res}_{\text{cyc}})$ against $\mathcal{A}$
4: $r \leftarrow F(f)\big(\alpha + (1-\alpha)\,\text{fill}\big)$ ▷ Equation 5
5: Update the miner with r
6: **if** f passes the admission test **then**
7: Add f to $\mathcal{A}$
8: **end if**

tournament-GP and REINFORCE pairs therefore isolate the reward term, while the batched pair measures the full loop.

4.3 A DIAGNOSTIC PLANE AND IC-BLIND SELECTION

To inspect a pool, we compress the exposure path into two scalar coordinates. The amplitude coordinate projects ϕ onto one fixed direction u, drawn once and shared by every formula (Appendix A),

$$\text{amp} = \tanh(\langle u, \phi \rangle). \tag{6}$$

The half-life ψ_X is the smallest lag, in days and capped at 40, at which the absolute autocorrelation of the demeaned path X_t falls below $1/2$. Because the two axes have different units, the distance z-scores each axis on the candidate pool before taking the Euclidean norm,

$$d_{\text{ECR}}(f,g) = \sqrt{\Big(\frac{\text{amp}_f - \text{amp}_g}{s_{\text{amp}}}\Big)^2 + \Big(\frac{\psi_{X,f} - \psi_{X,g}}{s_\psi}\Big)^2}, \tag{7}$$

where s_{amp} and s_ψ are the standard deviations of the two coordinates on the pool. A factor with undefined ϕ, or with an exposure path too short for a half-life, is left out of the plane. Figure 1 shows this plane for the two batched-search pools.

Farthest-point selection draws K factors on this plane without reading IC. It starts from the candidate farthest from the pool centroid and repeatedly adds the candidate whose distance to its nearest selected point is largest. We use it only as an IC-blind baseline for selecting on behavior (Section 5.2); it is not part of our method.

5 EXPERIMENTS

We ask four questions.

- **RQ1 (diagnosis).** Do IC-driven miners produce pools that mutual-IC screening cannot decorrelate, and does the IC of the selected set convert into return?
- **RQ2 (search).** Does adding the ECR residual to the reward raise IC, top-100 label return, and label Sharpe under mutual-IC screening?
- **RQ3 (placement).** Does behavioral diversity help at selection time, and does the answer depend on whether the pool was mined with it?
- **RQ4 (comparison).** Under one pipeline, how do the miner families rank, and how much of the ranking follows the objective?

5.1 SETUP

Data and protocol. The universe is a panel of $4{,}806$ A-share stocks with 344 tick-derived daily fields. Mining and both selection rules use data from 2018-01-02 to 2021-08-16; all reported metrics are out of sample, from 2021-08-17 to 2024-03-29 (634 trading days). Every run uses seed 0.

Table 1: Eight miners under one pipeline (seed 0). A name with "+ ECR" has the ECR residual in its search reward; the same name without it is its control. Pool: candidates after the shared filters (batched search is then truncated to 1,000). Cap: mutual-IC threshold at which fifty factors filled. Ov.: formulas shared by the mutual-IC and farthest-point sets. IC, Top-100, and Sharpe are out-of-sample moments of the combined score on the standardized label under the mutual-IC screen; $\text{Sharpe}_{\text{FP}}$ is the label Sharpe under farthest-point selection. Best values are in bold.

				Mutual-IC screen			FP
Miner	Pool	Cap	Ov.	IC	Top-100	Sharpe	$\text{Sharpe}_{\text{FP}}$
Batched search + ECR	886	0.30	2	0.085	**0.092**	**8.86**	3.63
Batched search	1,000	0.90	2	0.083	0.064	4.58	5.75
Tournament GP on RPN + ECR	771	0.50	8	**0.089**	0.084	8.64	**7.01**
Tournament GP on RPN	824	0.50	9	0.075	0.048	4.29	5.78
AlphaGen-style REINFORCE + ECR	460	0.50	8	0.074	0.071	6.42	6.17
AlphaGen-style REINFORCE	732	0.50	19	0.069	0.069	6.28	5.93
Uniform RPN sample	296	0.30	6	0.076	0.070	6.79	5.50
AlphaSAGE	196	0.50	12	0.064	0.055	5.69	4.33

Pipeline. Three eligibility filters (Appendix B) are applied to every pool before selection. After them, the batched-search control keeps 7,939 formulas, which we truncate to the top 1,000 by training $|\overline{\text{IC}}|$; every other miner, including batched search + ECR, keeps fewer than 1,000 and is not truncated. Each selection rule keeps $K = 50$ factors, and Table 1 reports the mutual-IC cap at which fifty were admitted. The combiner is the dynamic linear rule of AlphaForge (Shi et al., 2025): the weights used on a date are fitted on earlier dates only, and at most twenty of the fifty factors are used on any date (Appendix B).

Miners. All eight miners share one stack machine, operator set, and field set; Appendix F details each of them. Three search procedures are run with and without ECR, and each run without ECR is the control of its "+ ECR" counterpart. *Batched search* extends the open-source AlphaGPT framework (imbue-bit, 2025), a Transformer policy trained with REINFORCE, and evaluates candidates in GPU batches; batched search + ECR is our main method. *Tournament GP on RPN* evolves RPN formulas by tournament selection and mutation. *AlphaGen-style REINFORCE* keeps a masked token policy but scores one formula at a time by F instead of by AlphaGen's combination reward (Yu et al., 2023). For tournament GP on RPN and AlphaGen-style REINFORCE, ECR changes only the reward; for batched search, it also changes archive admission and the update (Section 4.2). We further run *uniform RPN sample*, which reads no score, and *AlphaSAGE* (Chen et al., 2026), reimplemented in our formula space with F as its terminal score. The other miners score one formula at a time; proposal counts are in Appendix F. Every portfolio contains fifty factors.

Metrics. We report the out-of-sample IC of the combined score, the mean label of its top 100 names (top-100 label return), and the annualized Sharpe ratio of that daily series (label Sharpe). All three are moments of the standardized label defined in Appendix D; they are not returns of a tradable portfolio and do not account for trading costs.

5.2 Results

Table 1 reports all eight miners under both selection rules, Table 2 isolates the ECR term in matched pairs, and Figure 3 plots IC against return for all sixteen miner–selection pairs.

RQ1: IC-driven search concentrates the pool. The batched-search control fills fifty factors only at a mutual-IC cap of 0.90, against 0.30 for batched search + ECR, and its fifty picks sit in a narrow band of exposure half-lives (Figure 1). Mutual-IC screening cannot repair this, because all it can do is relax the cap. Decorrelation and behavioral spread are also different criteria: across the eight miners, the mutual-IC and farthest-point sets share only 2 to 19 of their fifty formulas. Finally, IC converts into return unevenly. Under mutual-IC, the batched-search and tournament-GP-on-RPN controls fall below the IC–return trend in Figure 3.

Table 2: Matched pairs (seed 0). Each entry is the miner with ECR minus the same miner without it. "Changed" lists what ECR modifies: R, the reward; A, archive admission; U, the policy update. Only the batched pair changes more than the reward. Levels are in Table 1.

Miner	Changed	Screen	ΔIC	ΔTop-100	ΔSharpe
Batched search	R, A, U	Mutual-IC	+0.002	+0.028	+4.28
		Farthest point	−0.026	−0.031	−2.12
Tournament GP on RPN	R	Mutual-IC	+0.014	+0.036	+4.35
		Farthest point	−0.008	+0.008	+1.23
AlphaGen-style REINFORCE	R	Mutual-IC	+0.005	+0.002	+0.14
		Farthest point	+0.002	+0.011	+0.24

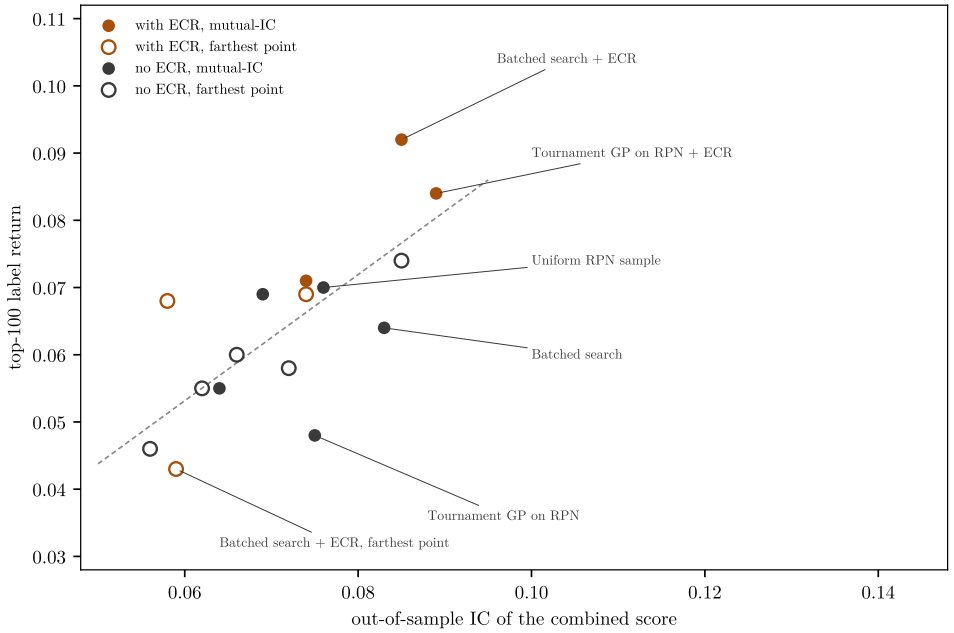

Figure 3: Out-of-sample IC of the combined score against top-100 label return for the sixteen miner–selection pairs. Filled markers: mutual-IC screen; open markers: farthest-point selection; orange: ECR in the search reward. Names match Table 1. The dashed least-squares line through the sixteen points is a visual reference only. Under mutual-IC, the batched-search and tournament-GP-on-RPN controls fall below it.

RQ2: The residual improves every miner it enters. Under mutual-IC screening, all three metrics rise for all three matched pairs (Table 2). The gains are large for batched search (+4.28 label Sharpe) and for tournament GP on RPN (+4.35), where ECR changes only the reward, and small for AlphaGen-style REINFORCE (+0.14). In both large cases, return rises far more than IC: for batched search, IC rises by 2% while top-100 label return rises by 44%; for tournament GP on RPN, the two figures are 19% and 75%. Figure 4 draws that daily series as a running sum. Under mutual-IC the two large gains pull away from their controls and stay smooth.

RQ3: Diversity belongs in the reward, not in selection. Selecting on behavior helps only pools that were mined without it. Farthest-point selection raises label Sharpe for the batched-search and tournament-GP-on-RPN controls by 1.17 and 1.49, but lowers it for all three ECR-trained pools, most sharply for batched search + ECR (from 8.86 to 3.63; Figure 4, right). A plausible reading is that once the reward has spread the pool across behaviors, an IC-blind rule mainly discards predictive factors. Diversity should enter once, in the reward, and selection should keep the IC order.

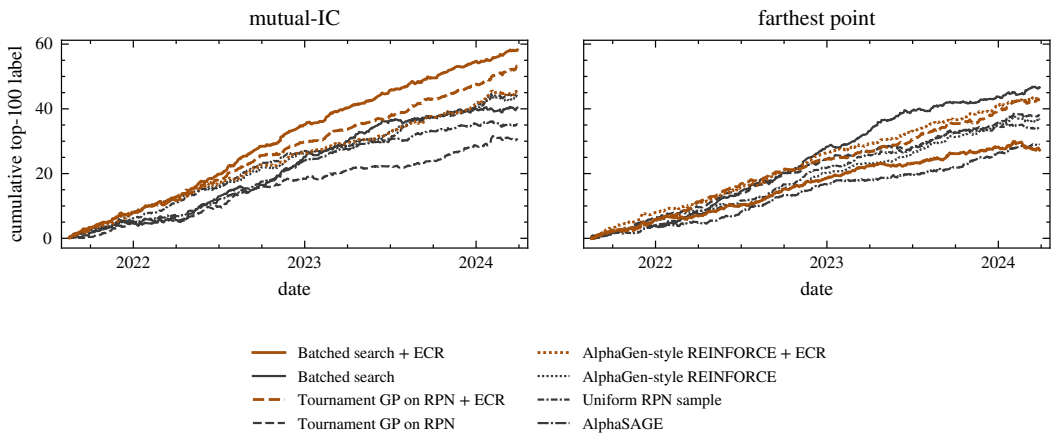

Figure 4: Running sum of the daily top-100 label mean over the out-of-sample window. Left: mutual-IC screen. Right: farthest-point selection. Orange marks a miner trained with ECR in the reward. The sum is the path of the top-100 column in Table 1; it is not a compounded portfolio value and does not include trading costs. Names match Table 1.

RQ4: The ranking follows the objective. Under one pipeline, the ranking of miners tracks whether the objective preserves behavioral diversity. Uniform RPN sample, which reads no score, reaches a label Sharpe of 6.79 and beats both the batched-search (4.58) and tournament-GP-on-RPN (4.29) controls. AlphaSAGE, whose novelty term is a correlation of factor values, reaches 5.69; value novelty does not substitute for the behavioral residual.

6 Conclusion

We introduced the Exposure–Cycle Residual to record the behavior that mutual-IC does not see: where on the capitalization spectrum a factor's ranking has been effective, and the rhythm of that tilt. In the search reward it scales fitness by the share of this behavior an online archive has not yet covered, so a repeated capitalization tilt is paid only in part.

Under one pipeline the residual raises out-of-sample label Sharpe for every miner it enters, and the rise in return is larger than the rise in IC. The gain is large for batched search and for tournament GP on RPN, where the reward is the only change. The same diversity at selection, through an IC-blind farthest-point rule, helps only the pools mined without the residual. Across the eight miners, the ranking follows whether the objective preserves that behavior. Behavioral diversity therefore belongs in the search reward, and the screen that follows should keep the IC order.

AI USE STATEMENT

In this work, generative AI was used to interpret the logged seed-0 summaries and to draft and edit the manuscript, including the tables and the scatter of IC against top-100 label return. It was not used to generate data, to run the miners, to formulate the definitions in Section 4, or to write proofs. Synthetic data, survey instruments, and qualitative coding are not part of this work. Code edits by the tool were limited to the figure script and the LaTeX source. The numbers in the tables were checked against the logged summaries. The authors reviewed the AI-assisted text and take responsibility for the final content.

REPRODUCIBILITY STATEMENT

Appendix A lists the descriptor constants, the smoothing recursions, and the draw of u. Appendix B lists the eligibility filters, the cap ladder, and the combiner. Appendix C gives the field concepts and the operator categories. Appendix D gives the label and the reported metrics. Appendix E lists the objective of each procedure. Appendix F records what each baseline changes and how many formulas it proposed. All reported tables use seed 0 and one shared pipeline. An anonymized code archive is submitted as supplementary material.

REFERENCES

Rob Arnott, Campbell R. Harvey, Vitali Kalesnik, and Juhani Linnainmaa. Alice's adventures in factorland: Three blunders that plague factor investing. *The Journal of Portfolio Management*, 45 (4):18–36, 2019.

Clifford S. Asness, Swati Chandra, Antti Ilmanen, and Ronen Israel. Contrarian factor timing is deceptively difficult. *The Journal of Portfolio Management*, 43(5):72–87, 2017.

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.

Can Cui, Wei Wang, Meihui Zhang, Gang Chen, Zhaojing Luo, and Beng Chin Ooi. AlphaEvolve: A learning framework to discover novel alphas in quantitative investment. In *Proceedings of the 2021 International Conference on Management of Data*, pp. 2208–2216, 2021.

Richard C. Grinold and Ronald N. Kahn. *Active Portfolio Management*. McGraw-Hill, 2nd edition, 2000.

imbue-bit. AlphaGPT: An open-source automated factor factory based on deep reinforcement learning. GitHub repository, commit 04a6a04b, 2025. URL `https://github.com/imbue-bit/AlphaGPT`.

Zura Kakushadze. 101 formulaic alphas. *Wilmott*, 2016(84):72–81, 2016.

Joel Lehman and Kenneth O. Stanley. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary Computation*, 19(2):189–223, 2011.

Jean-Baptiste Mouret and Jeff Clune. Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909*, 2015.

Justin K. Pugh, Lisa B. Soros, and Kenneth O. Stanley. Quality diversity: A new frontier for evolutionary computation. *Frontiers in Robotics and AI*, 3:40, 2016.

Hao Shi, Weili Song, Xinting Zhang, Jiahe Shi, Cuicui Luo, Xiang Ao, Hamid Arian, and Luis Angel Seco. AlphaForge: A framework to mine and dynamically combine formulaic alpha factors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12524–12532, 2025.

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. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2813–2822, 2025.

Saizhuo Wang, Hang Yuan, Leon Zhou, Lionel M. Ni, Heung-Yeung Shum, and Jian Guo. AlphaGPT: Human-AI interactive alpha mining for quantitative investment. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 196–206, 2025.

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.

Tianping Zhang, Yuanqi Li, Yifei Jin, and Jian Li. AutoAlpha: An efficient hierarchical evolutionary algorithm for mining alpha factors in quantitative investment. *arXiv preprint arXiv:2002.08245*, 2020.

Junjie Zhao, Chengxi Zhang, Min Qin, and Peng Yang. QuantFactor REINFORCE: Mining steady formulaic alpha factors with variance-bounded REINFORCE. *IEEE Transactions on Signal Processing*, 73:2448–2463, 2025.

Zhoufan Zhu and Ke Zhu. AlphaQCM: Alpha discovery in finance with distributional reinforcement learning. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 80463–80479, 2025.

A DESCRIPTOR DETAILS AND DEFAULT PARAMETERS

The descriptor, the residual reward, and farthest-point selection use the defaults below. Farthest-point selection uses no threshold. The fitness weight α is the coefficient in Equation 5, and the two admission thresholds are the gates of Section 4.2.

Quantity	Default
Capitalization layers L	5
Leg fraction	0.2
Minimum names per leg m	3
Volatility half-life	10
Strength half-life	20
Observations before an average is emitted	20
Stocks required for β_t	10
Centered window for β_t	20
Autocorrelation lags	40
Half-life threshold for ψ_X	$1/2$
Norm floor for ϕ	10^{-12}
Fitness weight α	0.3
Amplitude admission threshold	0.15
Cycle admission threshold	0.20
Amplitude admission cosine	0.85

On date t, let $S_{t,k}$ be the stocks in layer k whose factor value and label are both finite, and write $n_{t,k} = |S_{t,k}|$. The layer is left undefined when $n_{t,k} < \max(4, 2m)$ with $m = 3$. The leg size is

$$n^{\pm}_{t,k} = \min\big(\max(1, \lfloor 0.2\, n_{t,k} \rfloor),\ \lfloor n_{t,k}/2 \rfloor\big). \tag{8}$$

Before ranking inside a layer, the factor is standardized across stocks on that date. Sorting $S_{t,k}$ by the standardized factor, the layer return and the layer centroid are

$$r_{t,k} = \frac{1}{n^{\pm}_{t,k}} \sum_{i \in \text{top}} y_{i,t} - \frac{1}{n^{\pm}_{t,k}} \sum_{i \in \text{bottom}} y_{i,t}, \qquad c_{t,k} = \frac{1}{n_{t,k}} \sum_{i \in S_{t,k}} C_{i,t}, \tag{9}$$

where $C_{i,t} = \ln M_{i,t}$ and $M_{i,t}$ is the free-float market capitalization of stock i on date t, in yuan. The series M is the one in Appendix C. Because the logarithm is strictly increasing, sorting on C is the same order as sorting on capitalization. The same calculation also forms a nonlinear midcap residual, the part of the cubed cross-sectional z-score of C that is orthogonal to C; that residual is not the centroid and does not enter β_t. The centroid is then demeaned across the defined layers, giving $\tilde{c}_{t,k}$.

For a series x_t and half-life h, set $\lambda_h = 1 - 2^{-1/h}$ and $E_t = (1 - \lambda_h)E_{t-1} + \lambda_h x_t$ when x_t is finite, leaving E unchanged otherwise, and initialize E at the first finite observation. The value used on date t is E_{t-1}, emitted only after 20 finite observations; we write $\mathrm{LE}_h[x]_t$ for this lagged average. Each layer return is scaled by its trailing volatility and then smoothed,

$$\sigma^2_{t,k} = \mathrm{LE}_{10}[r^2_{\cdot,k}]_t, \qquad s_{t,k} = \mathrm{LE}_{20}\left[\frac{r_{\cdot,k}}{\sigma_{\cdot,k}}\right]_t. \tag{10}$$

Demeaning $s_{t,k}$ across layers, with undefined entries set to zero after the demeaning, gives the weights $w_{t,k}$ in Equation 3. Because both the weights and the centroids are demeaned, X_t is a positive multiple of the cross-layer covariance between strength and centroid.

Let $\bar{X}$ be the vector of X_t over the training dates, with undefined dates set to zero. The amplitude signature is

$$\phi = \frac{\bar{X} - \langle \bar{X} \rangle \mathbf{1}}{\|\bar{X} - \langle \bar{X} \rangle \mathbf{1}\|_2}, \tag{11}$$

and it is left undefined when the norm is below 10^{-12}. The direction u in Equation 6 is drawn once from a standard normal generator with seed 17 and then normalized; it is shared by every formula and does not depend on the candidate.

The size loading β_t is the Spearman correlation between the factor and capitalization over the stocks where both are finite; it is undefined when fewer than ten stocks qualify. A centered moving average of window 20, requiring at least ten finite values inside the window, smooths β_t. These signatures are computed on the training window only and do not enter the combiner. Let v be the finite smoothed values with their mean removed; at least 40 of them are required. The path signature ψ^{path} is this demeaned series, with missing dates set to zero, divided by its L_2 norm. The autocorrelation signature $\psi^{\mathrm{ac}} \in \mathbb{R}^{40}$ has entries

$$\psi^{\mathrm{ac}}_{\ell} = \frac{\sum_t v_t v_{t+\ell}}{\sum_t v_t^2}, \qquad \ell = 1, \ldots, 40, \tag{12}$$

and is divided by its own L_2 norm. A factor that swings between a small-cap and a large-cap tilt on a p-day rhythm has its large entries of ψ^{ac} near lag p. The reward uses $(\phi, \psi^{\mathrm{path}}, \psi^{\mathrm{ac}})$; the diagnostic plane uses (amp, ψ_X), which come from the same exposure path.

Admission to the archive uses, beyond the thresholds 0.15 and 0.20, the cosine in the table. The amplitude axis counts as new only when $\mathrm{res}_{\mathrm{amp}} > 0.15$ and the absolute cosine to the nearest stored signature is below 0.85; the cycle axis counts as new when $\mathrm{res}_{\mathrm{cyc}} > 0.20$. If the cycle axis is confidently covered, it vetoes an amplitude-only claim. A formula that is new on either axis is appended and, when its amplitude claim is reliable, extends the basis. A formula covered on both axes replaces the nearest stored formula only when its fitness is higher, and it does not extend the basis. The numeric gates behind a confidently covered cycle axis and a reliable amplitude claim are in the supplementary code.

B Pipeline

Three filters are applied to every miner before either selection. (i) A formula may contain the operator `TS_CORR_20` at most three times. (ii) On eight probe dates, we compute the share of names lying within 0.05 cross-sectional standard deviations of the cross-sectional median; the median of these eight shares must be below 0.10. The probe dates are 20210225, 20210317, 20210407, 20210427, 20210520, 20210609, 20210630, and 20210720. (iii) Training $|\overline{\mathrm{IC}}|$ must be at least 0.01. The released code uses the same cutoffs. After these filters, the batched-search control has 7,939 formulas and is truncated to the top $1{,}000$ by training $|\overline{\mathrm{IC}}|$. Batched search + ECR retains 886 formulas and is not truncated, and neither is any other miner, since each has fewer than $1{,}000$ left.

The mutual-IC screen uses one ladder for every miner: 0.30, 0.50, 0.70, 0.90, 0.95. It stops at the first cap that yields fifty formulas, which is the cap reported in Table 1. Farthest-point selection draws fifty formulas from the same filtered pool and does not read IC.

The combiner is a dynamic linear combination of the fifty selected formulas, shared by every row. The estimation window is expanding, and the weight vector used on a date is estimated from earlier dates only. Formulas with historical $|\mathrm{RankIC}| > 0.015$ and $|\mathrm{RankICIR}| > 0.15$ are eligible; they are ranked by $|\mathrm{RankICIR}|$, and at most twenty are retained. Retained values are standardized in the cross-section, and the linear model includes an intercept. The reported IC, top-100 mean, and Sharpe are computed from the resulting score on the label of Appendix D.

C Fields and operators

The 344 fields are daily reductions of the limit-order book and the trade tape, plus a chip block. Two inputs are used, and neither is released with this paper. The book and the tape are the paid five-level tick feed from Juejin; a reader obtains that feed separately. Free-float capitalization $M_{i,t}$ is the circulating market value, in yuan, from the AKShare series `stock_value_em`. The logarithm $C_{i,t}$ and the midcap residual are computed from this series, and the label buckets use the same C. Operator implementations are not released. Table 3 records what each block measures. Table 4 records the 44 operators by kind. A window is fixed inside the operator, as in `TS_CORR_20`, rather than being a separate token.

Table 3: Daily fields by concept. Each count is a number of fields.

Block	What it measures	Count
Tape, ten families	Spread 16, depth 21, price impact 17, illiquidity 21, order flow 24, realized volatility 24, money flow 23, microstructure 18, intraday timing 21, comovement 15	200
Tape, further	Execution cost 7, price discovery 6, order-flow resilience 6, refinements of depth, volatility, flow, impact, and illiquidity 11, dollar volume and aggregate depth 2	32
Handcrafted reductions	Book structure 16, quote changes and cancellations 10, signed trades and trade size 13, impact 7, volatility 12, volume–price 12, session returns 12, limit moves 5	87
Chip atoms	Close, volume-weighted price, high, low, volume, and a large-versus-small trade split of price and volume	9
Chip state	Cost, profit share, and concentration on a 60-day window, with the same quantities for large trades	16
		344

Table 4: Operators by kind. The search uses this set and no others.

Kind	What an operator of this kind does	Count
Elementwise arithmetic	Add, subtract, multiply, divide, or rescale one panel by another	5
Elementwise nonlinear	Negate, or apply a smooth sign, bound, or absolute value	4
Gates	Pass a signal when a condition is high or low, or blend two signals	3
Cross-sectional	Rescale, standardize, or rank within the day	3
Time-series	Lag, decay, or take a rolling mean, sum, scale, change, skew, correlation, or short-horizon reversal	25
Turnover-weighted recursion	Update a reference price or a gap with turnover, or mark a smooth profit	4
		44

D LABEL AND REPORTED METRICS

The label on date t is the close-to-close return from that close to the next. Stocks are cut into five equal-count buckets of $C_{i,t}$, the log free-float capitalization of Appendix A. Inside each bucket the return is winsorized at the 1st and 99th percentiles, the bucket mean is removed, and the remainder is divided by the within-bucket standard deviation. A bucket with fewer than twenty finite names is left undefined. The factor panel and the label file share this date index. Mining and both selections use dates through 2021-08-16; the reported window $\mathcal{D}$ runs from 2021-08-17 through 2024-03-29 (634 trading days).

Let $z_{i,t}$ be the combined score and $\mathrm{Top}_{100}(t)$ the 100 names with the highest score on day t. We report

$$\mathrm{IC} = \frac{1}{|\mathcal{D}|}\sum_{t\in\mathcal{D}} \mathrm{corr}_i(z_{\cdot,t}, y_{\cdot,t}), \qquad m_t = \frac{1}{100}\sum_{i\in\mathrm{Top}_{100}(t)} y_{i,t}, \qquad \mathrm{Sharpe} = \sqrt{252}\,\frac{\mathrm{mean}_{t\in\mathcal{D}}\, m_t}{\mathrm{std}_{t\in\mathcal{D}}\, m_t}, \tag{13}$$

with the risk-free rate set to zero. The top-100 label return is the mean of m_t over $\mathcal{D}$. All three numbers are moments of the label, not returns of a tradable portfolio.

E WHAT EACH PROCEDURE OPTIMIZES

The eight procedures do not optimize one expression. Equation 5 is the objective of tournament GP on RPN + ECR and AlphaGen-style REINFORCE + ECR. The batched-search control keeps $F = 8|\overline{\mathrm{IC}}| + 4\, r_{\mathrm{book}}$. Batched search + ECR keeps that score inside the same fill factor, then adds a shape term and subtracts a batch penalty,

$$r_{\mathrm{batch}}(f) = F(f) \cdot \big(\alpha + (1-\alpha)\,\mathrm{fill}(f)\big) + s(f) - p(f), \tag{14}$$

with $\alpha = 0.3$. For this miner the two residuals inside fill are taken against the archive and against the higher-fitness formulas already counted in the same batch.

The shape term reads the token sequence. Let $U(f)$ be the number of distinct field tokens, and let $\mathcal{R}(f)$ be the lengths of the maximal runs of consecutive operator tokens. Then

$$s(f) = 0.1\, U(f) - 0.05 \sum_{L \in \mathcal{R}(f)} L^2. \tag{15}$$

A field token ends the current operator run. The batch penalty is computed inside one GPU batch. The amplitude penalty is the largest absolute cosine between ϕ_f and the amplitude signature of any other formula in the batch, and the cycle penalty is the largest positive cosine between ψ_f^{path} and the path signature of any other formula in the batch, so an opposite phase is not penalized. With those two quantities,

$$p(f) = 0.3\,\mathrm{pen}_{\mathrm{amp}}(f) + 0.15\,\mathrm{pen}_{\mathrm{cyc}}(f). \tag{16}$$

A formula with an undefined signature, or a batch that contains no other valid signature, has $p(f) = 0$. Equation 14 is the scalar reward of batched search + ECR. Tournament GP on RPN and AlphaGen-style REINFORCE do not add s or p.

Procedure	Objective
Batched search + ECR	Equation 14, with s in Equation 15 and p in Equation 16
Batched search	$F = 8\|\overline{\mathrm{IC}}\| + 4\, r_{\mathrm{book}}$
Tournament GP on RPN + ECR, AlphaGen-style REINFORCE + ECR	Equation 5
Tournament GP on RPN, AlphaGen-style REINFORCE	F
Uniform RPN sample	none
AlphaSAGE	F, plus value-correlation novelty and a structure term

Here r_{book} is the mean label on the chosen end in the formula backtest of Section 3.3. AlphaSAGE does not use Equation 5; its novelty term is one minus the largest absolute mutual-IC against its online library. Every procedure uses the same stack machine, operator set, and 344 fields. The concepts and the counts are in Appendix C; operator implementations are not released.

F BASELINE IMPLEMENTATIONS

This appendix records how each miner was run. Every reported portfolio contains fifty factors under the screens of Appendix B. Batched search proposes formulas in GPU batches, 400 iterations of 256; the other miners score one formula at a time, which is why they propose fewer formulas.

Procedure	Proposals scored	Eligible pool
Batched search + ECR	102,400	886
Batched search	102,400	1,000
Tournament GP on RPN + ECR	2,000	771
Tournament GP on RPN	2,000	824
AlphaGen-style REINFORCE + ECR	1,907	460
AlphaGen-style REINFORCE	2,000	732
Uniform RPN sample	2,000	296
AlphaSAGE	1,136	196

The two shortfalls, $1{,}907$ and $1{,}136$, are the valid formulas produced before the target of $2{,}000$ was reached. The eligible pool is the list after the filters of Appendix B; only the batched-search control is then truncated, from $7{,}939$ formulas to $1{,}000$.

Batched search. Batched search extends the open-source AlphaGPT framework (imbue-bit, 2025) at commit `04a6a04b`, under the Apache-2.0 license. That framework trains a Transformer policy with REINFORCE to emit formula tokens for a stack machine, originally to time single crypto assets. We keep the policy, the REINFORCE update, and the stack executor, with masked sampling on the 344-field panel, and replace the timing reward with the cross-sectional formula backtest $F = 8|\overline{\mathrm{IC}}| + 4\, r_{\mathrm{book}}$. Batched search + ECR keeps this loop and adds the archive: admission uses the thresholds 0.15 on the amplitude residual and 0.20 on the cycle residual (Appendix A), the screen sees the pool after admission, and the reward is Equation 14 (Appendix E). The framework is unrelated to Alpha-GPT, the language-model miner of Wang et al. (2025).

Tournament GP on RPN. A hand-written tournament with reverse-Polish mutation, with population size 32 and tournament size 4. The objective is F. Tournament GP on RPN + ECR optimizes Equation 5 alone, and the screen sees every valid formula in the log.

AlphaGen-style REINFORCE. A masked policy over the same tokens. The reward is F of one formula, so it stays separate from the combination-IC increment of Yu et al. (2023), and the combiner stays outside the reward. AlphaGen-style REINFORCE + ECR optimizes Equation 5 alone, and the screen sees every valid formula in the log.

Uniform RPN sample. Draws valid reverse-Polish formulas from the shared expression space and reads no score.

AlphaSAGE. We keep the generative flow network, the relational graph encoder, the entropy bonus, and the decaying novelty and structure terms of Chen et al. (2026). The novelty term is one minus the largest absolute mutual-IC against the online library, the same value correlation that mutual-IC applies at selection. The terminal information coefficient of the original is replaced by F, expressions are drawn from the shared stack machine rather than from the original data environment, and Equation 5 is not added. The reported portfolio is the shared screen over the logged formulas.

AlphaForge. AlphaForge supplies the dynamic linear combiner used after selection (Shi et al., 2025); we do not rerun its generator.