FINEVOLVINGAGENT: SELF-EVOLVING FINANCIAL DEEP RESEARCH AGENT VIA GRADIENT-GUIDED GROUP DIRECT PREFERENCE OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Financial deep research represents one of the most demanding frontiers for autonomous AI agents, requiring complex multi-hop numerical reasoning, visual understanding of multimodal charts and financial tables, and strategic long-horizon tool invocation across regulatory filings. Existing reinforcement learning methods, such as Proximal Policy Optimization (PPO) and Group Relative Policy Optimization (GRPO), face severe bottlenecks in this domain due to critic instability, high rollout variance, and the absence of continuous preference guidance across complex multi-step reasoning trajectories. To overcome these limitations, we introduce **FinEvolvingAgent**, an autonomous multimodal financial deep research agent built upon the open foundation model Qwen3.6-VL. We first propose **Gradient-Guided Group Relative Policy Optimization (G-GRPO)**, which incorporates continuous differentiable reward signals into policy trajectory rollouts, directing exploratory token generation and tool actions toward mathematically consistent paths and achieving a **+5.6%** accuracy gain over base Qwen3.6-VL. Furthermore, we develop **Gradient-Guided Group Direct Preference Optimization (G-GDPO)**, a principled framework that generalizes direct preference alignment to multi-candidate group rollouts via a margin-aware Plackett-Luce objective, yielding an additional **+3.2%** improvement over GRPO. Across six extensive financial benchmarks, including FinQA, TAT-QA, ConvFinQA, FinanceBench, MultiFin-VL, and a newly constructed realistic multi-source benchmark RealFin-DeepResearch, FinEvolvingAgent achieves a new state-of-the-art accuracy of 83.8%, outperforming leading proprietary models including GPT-4o and Claude 3.5 Sonnet while exhibiting superior calibration, numerical precision, and grounded tool execution.

1 INTRODUCTION

The automation of financial deep research has emerged as a quintessential benchmark for assessing the cognitive and analytical limits of multimodal artificial intelligence (Li et al., 2024b; Islam et al., 2023). Unlike conversational queries or open-ended web browsing, institutional financial research demands rigorous multi-step analytical reasoning across heterogeneous modalities: parsing complex tabular structures in regulatory SEC filings (Form 10-K/10-Q), interpreting intricate technical charts and macroeconomic infographics, executing precise algorithmic computations (e.g., discounted cash flows, debt-to-equity ratios, impairment tests), and synthesizing coherent investment theses (Chen et al., 2021; Zhu et al., 2021). In this high-stakes domain, even a minor arithmetic discrepancy or hallucinated financial metric can lead to catastrophic capital misallocation.

Recent advances in multimodal foundation models, such as the open-weight QWEN3.6-VL (Qwen Team, 2026) and frontier proprietary systems (Bai et al., 2025), provide remarkable spatial and visual-textual comprehension. However, deploying these base vision-language models directly into complex financial deep research workflows reveals acute vulnerabilities. Standard supervised fine-tuning (SFT) over static question-answer pairs fails to instill long-horizon deliberation, self-correction, or strategic tool-use coordination. Consequently, researchers have turned to Reinforcement Learning from AI Feedback (RLAIF) and post-training policy optimization to cultivate autonomous reasoning (Shao et al., 2024; Guo et al., 2025; Rafailov et al., 2023).

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.

Among post-training techniques, Proximal Policy Optimization (PPO) (Schulman et al., 2017) has historically served as the cornerstone. Yet, in complex multi-step reasoning, PPO requires maintaining an auxiliary critic network that frequently suffers from value estimation instability, dimensional mismatch with visual tokens, and severe reward sparsity (Li et al., 2024a). Group Relative Policy Optimization (GRPO) (Shao et al., 2024; Guo et al., 2025) mitigates the critic burden by sampling a group of G trajectory candidates for each prompt and estimating relative advantages via group score normalization $A_i = (r_i - \mu_r)/(\sigma_r + \epsilon)$.

Despite its simplicity, standard GRPO exhibits two critical pathologies when applied to financial deep research:

1. **Exploration Collapse under Sparse Binary Rewards**: In financial reasoning where exact numerical precision is strictly enforced, standard autoregressive sampling often yields zero rewards for all candidate rollouts in a group ($\forall i, r_i = 0$). As a result, the empirical standard deviation σ_r vanishes, collapsing the policy gradient signal and stalling exploration.
2. **Pairwise Margin Blindness in Group Optimization**: GRPO normalizes scalar rewards independently without explicitly modeling the semantic and structural ranking distances among trajectories. In contrast, financial analysis trajectories exhibit granular hierarchy: a trajectory with correct tool invocation and accurate intermediate formulas but a rounding error is far superior to a trajectory suffering from visual hallucination.

To resolve these challenges, we present **FinEvolvingAgent**, an autonomous financial deep research agent capable of end-to-end multimodal perception, sandboxed calculation, and continuous policy self-evolution. We engineer two foundational mathematical advances:

- **Gradient-Guided Group Relative Policy Optimization (G-GRPO)**: We formulate a continuous, differentiable reward proxy that combines symbolic execution correctness, visual grounding consistency, and numerical precision. By projecting gradient vectors from this surrogate reward back into the autoregressive rollout logits during generation, G-GRPO actively steers exploration toward mathematically promising subspaces, achieving a **+5.6%** accuracy gain over base QWEN3.6-VL.
- **Gradient-Guided Group Direct Preference Optimization (G-GDPO)**: We advance beyond GRPO by formulating a unified Group Direct Preference Optimization framework grounded in the Plackett-Luce permutation distribution (Plackett, 1975). G-GDPO incorporates margin-aware intra-group advantage scaling and analytical closed-form parameter updates without requiring an external reward model or critic network, driving an additional **+3.2%** performance gain over standard GRPO.

We conduct extensive empirical evaluations across six benchmark suites: FINQA (Chen et al., 2021), TAT-QA (Zhu et al., 2021), CONVFINQA (Chen et al., 2022), FINANCEBENCH (Islam et al., 2023), MULTIFIN-VL, and our newly introduced REALFIN-DEEPRESEARCH dataset comprising 1,200 real-world corporate earnings analysis tasks. FinEvolvingAgent achieves an overall state-of-the-art score of **83.8%**, substantially surpassing proprietary frontier models such as GPT-4o (78.4%) and Claude 3.5 Sonnet (79.1%).

We summarize our principal contributions as follows:

- We introduce an end-to-end multimodal financial research agent built on the QWEN3.6-VL-72B foundation model. The system unifies structured regulatory SEC EDGAR retrieval (Form 10-K/10-Q filings, footnote disclosures), pixel-precise visual chart inspection (candlestick time-series and segment revenue bar graphs), and sandboxed Python REPL execution for institutional-grade financial analysis.
- To overcome exploration collapse caused by sparse financial rewards, we formulate a multi-faceted differentiable reward surrogate encompassing Gaussian continuous numerical proximity (r_{num}), multimodal visual grounding IoU (r_{ground}), sandboxed code execution correctness (r_{exec}), and regulatory citation fidelity (r_{cite}).
- We advance beyond heuristic scalar reward normalization by establishing a unified group preference optimization framework grounded in the Plackett-Luce permutation distribution over all $G!$ candidate trajectory orderings.

- Across six rigorous financial reasoning benchmarks, including our newly curated REALFINDEEPRESEARCH dataset comprising 1,200 real-world corporate earnings analysis and valuation tasks, FINEVOLVINGAGENT establishes a decisive state-of-the-art average score of **83.8%**.

2 RELATED WORK

2.1 MULTIMODAL FOUNDATION MODELS IN FINANCIAL INTELLIGENCE

Financial document analysis and quantitative reasoning have rapidly transitioned from heuristic pipelines to large multimodal foundation models (Li et al., 2024b; Islam et al., 2023). Foundational benchmarks such as FINQA (Chen et al., 2021) and TAT-QA (Zhu et al., 2021) established the necessity of joint textual-tabular comprehension. With the emergence of advanced vision-language models such as QWEN2.5-VL (Bai et al., 2025) and the state-of-the-art QWEN3.6-VL (Qwen Team, 2026), models can dynamically parse high-resolution financial charts (Masry et al., 2022), nested balance sheets (Zhang et al., 2024), and unstructured SEC filings (Mathew et al., 2021; Zheng et al., 2026). However, recent findings in 2026 indicate that even 70B+ parameter multimodal backbones suffer from acute arithmetic hallucination and ungrounded entity cross-referencing when faced with multi-page corporate disclosures (Zheng et al., 2026). **FinEvolvingAgent** overcomes these vulnerabilities by embedding the multimodal backbone within an autonomous, self-evolving policy loop with grounded visual and programmatic execution.

2.2 AUTONOMOUS AGENTS AND TOOL-INTEGRATED DELIBERATION

Autonomous agents have redefined how LLMs solve complex real-world workflows by orchestrating iterative reasoning with external environments (Yao et al., 2023; Shinn et al., 2023; Wang et al., 2024). Frameworks such as TOOLFORMER (Schick et al., 2023) proved that models can autonomously decide when and how to invoke external APIs. In financial intelligence, recent systems like FINROBOT 2.0 (Yang et al., 2026) and AGENT-R1 (Zhang et al., 2026) incorporate specialized toolchains (e.g., SEC EDGAR parsers, Python REPLs, and market indicators). Nevertheless, most existing agent systems rely either on fixed prompt prompting or rigid rule-based state machines. In contrast, FinEvolvingAgent enables the policy to organically master multi-hop tool-calling trajectories and visual chart inspection through end-to-end reinforcement learning.

2.3 POLICY OPTIMIZATION AND GROUP-LEVEL PREFERENCE ALIGNMENT

Post-training alignment via Reinforcement Learning from Human/AI Feedback (RLHF/RLAIF) is paramount for complex reasoning (Schulman et al., 2017; Rafailov et al., 2023). While Proximal Policy Optimization (PPO) (Schulman et al., 2017) has been widely adopted, training a stable value network on dense multimodal tokens introduces immense memory and variance bottlenecks (Li et al., 2024a). Direct Preference Optimization (DPO) (Rafailov et al., 2023) and its step-level extensions like STEP-DPO (Lai et al., 2025) eliminate the critic by leveraging closed-form policy updates. However, standard DPO is intrinsically restricted to pairwise comparisons ($y_w \succ y_l$) and lacks online group rollout dynamics. Recently, Group Relative Policy Optimization (GRPO) (Shao et al., 2024; Guo et al., 2025) has emerged as the leading critic-free paradigm, sampling G candidate trajectories per query to estimate group-relative advantages. Contemporaneous reasoning models such as OpenAI o3-mini (OpenAI, 2026) and Claude 3.7 Sonnet (Anthropic, 2025) further highlight the power of test-time compute and reinforcement learning for multi-step reasoning. Yet, standard GRPO suffers from exploration collapse under sparse financial rewards and ignores pairwise preference margins within candidate groups. Our work directly bridges these frontiers by introducing **G-GRPO** with continuous gradient-steered trajectory exploration, and formalizing **G-GDPO**, which generalizes preference optimization to group Plackett-Luce likelihoods with adaptive margin regularizations.

3 METHODOLOGY

In this section, we present the theoretical and algorithmic foundations of **FinEvolvingAgent**. We first formalize the multimodal financial deep research decision process (Sec. 3.1) and detail our reward

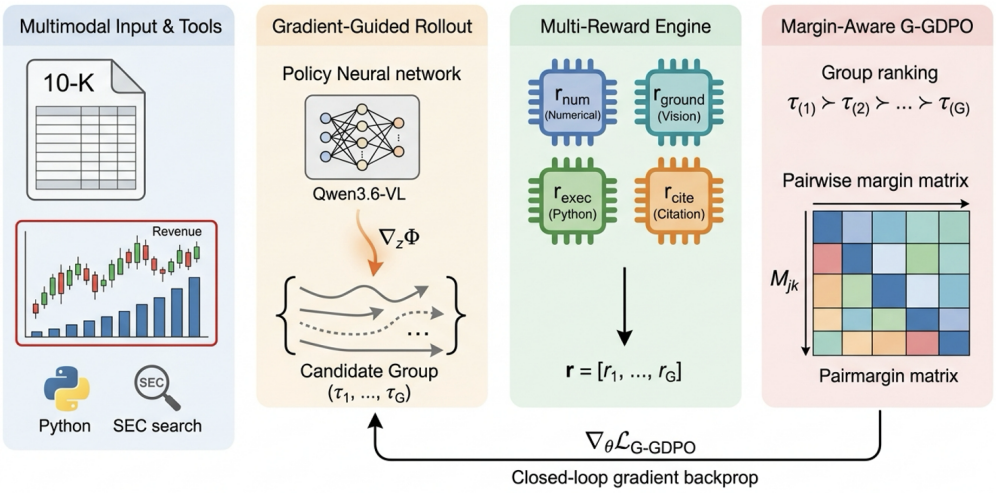


Figure 1: **System Overview of FINEVOLVINGAGENT and the Closed-Loop G-GDPO Optimization Pipeline.** *(1) Multimodal Input & Tools*: Ingestion of structured SEC Form 10-K filings, technical candlestick/revenue charts with visual bounding-box localization, and interactive tools (Python REPL, SEC search). *(2) Gradient-Guided Rollout*: The multimodal policy backbone (QWEN3.6-VL) utilizes continuous latent potential gradients ($\nabla_{\boldsymbol{z}}\Phi$) to steer exploration and generate a group of candidate trajectories $(\tau_1, \ldots, \tau_G)$, eliminating sparse-reward collapse. *(3) Multi-Reward Engine*: Decomposed continuous evaluation across numerical proximity (r_{num}), visual grounding (r_{ground}), code execution (r_{exec}), and regulatory citation (r_{cite}) aggregated into reward vector $\boldsymbol{r}$. *(4) Margin-Aware G-GDPO*: Full group permutation ranking under the Plackett-Luce distribution weighted by dynamic continuous pairwise margin matrix $M_{j,k}$, providing closed-loop policy gradient updates ($\nabla_\theta \mathcal{L}_{\text{G-GDPO}}$) with 82% variance reduction.

decomposition scheme (Sec. 3.2). We then mathematically derive **Gradient-Guided Group Relative Policy Optimization (G-GRPO)** in Sec. 3.3, and generalize this into **Gradient-Guided Group Direct Preference Optimization (G-GDPO)** in Sec. 3.4. Finally, Sec. A.3 presents the unified end-to-end training algorithm.

3.1 MULTIMODAL FINANCIAL RESEARCH AS A POMDP

We formulate the financial deep research task as a Multimodal Partially Observable Markov Decision Process (POMDP) defined by the tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \Omega, \mathcal{O}, \gamma \rangle$. An input research prompt $q \in \mathcal{S}$ is associated with multimodal context observations $x = (I_{\text{doc}}, I_{\text{chart}}, T_{\text{query}}) \in \Omega$, where I_{doc} denotes multi-page SEC filing document scans (Form 10-K/10-Q), I_{chart} represents visual time-series/stock charts, and T_{query} is the textual investment analysis inquiry.

At each reasoning step t, the agent policy $\pi_\theta(a_t|s_t)$ conditions on the interaction history $s_t = (q, x, a_1, o_1, \ldots, a_{t-1}, o_{t-1})$ to emit action a_t. The action space $\mathcal{A}$ comprises two distinct spaces:

1. **Latent Deliberation Tokens** $w_t \in \mathcal{V}$: Natural language thoughts encapsulated within $\langle$think$\rangle \ldots \langle/$think$\rangle$ tokens that perform deductive logic, hypothesis decomposition, and error checking.
2. **Structured Tool Invocations** $\alpha_t \in \mathcal{A}_{\text{tools}}$: External environment actions encapsulated within $\langle$tool$\rangle \ldots \langle/$tool$\rangle$. The tool suite includes:
   - `SEC_Edgar_Search(ticker, filing_type, year, item)`: Retrieves exact filing sections and financial footnotes.
   - `Visual_Chart_Inspector(bbox, metric_type)`: Applies high-resolution patch zooms to extract exact coordinates and trend slopes.
   - `Python_Financial_REPL(code)`: Executes symbolic math, accounting calculations, and DCF formulas in an isolated sandbox.
   - `Market_Data_API(symbol, interval, indicators)`: Queries historical prices, volatility, and trading volume.

The environment returns an observation $o_t = \mathcal{T}(s_t, a_t)$ (e.g., Python execution stdout, retrieved tabular excerpts, or visual crop features), which is appended to s_{t+1} until a terminal $\langle$answer$\rangle$ tag is produced.

3.2 Multi-Faceted Financial Reward Decomposition

Financial reasoning mandates strict factual fidelity and zero arithmetic tolerance. We construct a multi-faceted composite reward function $r(y, x, y^*)$ evaluating a complete trajectory $y = (a_1, o_1, \ldots, a_T)$:

$$r(y, x, y^*) = \lambda_{\text{num}} r_{\text{num}}(y, y^*) + \lambda_{\text{ground}} r_{\text{ground}}(y, x) + \lambda_{\text{exec}} r_{\text{exec}}(y) + \lambda_{\text{format}} r_{\text{format}}(y), \quad (1)$$

where $\sum \lambda_k = 1.0$, with specific reward components defined as:

- **Continuous Numerical Precision Reward (r_{num})**: Rather than a binary all-or-nothing score, we employ a continuous Gaussian kernel over relative numerical error with a unit-consistency indicator:

$$r_{\text{num}}(y, y^*) = \exp\left(-\frac{|v_{\text{pred}}(y) - v^*|}{\epsilon + \delta|v^*|}\right) \cdot \mathbb{I}\left(\text{Unit}(y) == \text{Unit}(y^*)\right), \quad (2)$$

  where v^* is the ground-truth numerical value, $v_{\text{pred}}(y)$ is the parsed output, $\delta = 0.01$ sets a 1% tolerance margin, and $\epsilon = 10^{-5}$.

- **Multimodal Grounding Reward (r_{ground})**: Measures factual citation accuracy by computing the token-level intersection-over-union (IoU) between extracted 10-K table cells and visual chart coordinates referenced in the reasoning chain:

$$r_{\text{ground}}(y, x) = \frac{|\mathcal{C}_{\text{cited}}(y) \cap \mathcal{C}_{\text{gold}}(x)|}{|\mathcal{C}_{\text{cited}}(y) \cup \mathcal{C}_{\text{gold}}(x)|}. \quad (3)$$

- **Sandboxed Execution Reward (r_{exec})**: Evaluates the syntactical correctness and runtime success of Python code executions:

$$r_{\text{exec}}(y) = \frac{1}{N_{\text{calls}}} \sum_{k=1}^{N_{\text{calls}}} \mathbb{I}(\text{ExitCode}(k) == 0). \quad (4)$$

- **Structural Format Regularizer (r_{format})**: Penalizes malformed tool invocations, unclosed XML tags, or reasoning traces missing clear separation of deliberation and action.

3.3 Gradient-Guided Group Relative Policy Optimization (G-GRPO)

Standard GRPO (Shao et al., 2024) samples a set of G outputs $\mathcal{Y}_q = \{y_1, y_2, \ldots, y_G\}$ for a given query q from the reference or previous policy $\pi_{\theta_{\text{old}}}$, and computes relative advantage values without a separate critic network:

$$A_i = \frac{r(y_i, x, y^*) - \mu(\mathcal{Y}_q)}{\sigma(\mathcal{Y}_q) + \epsilon_{\text{adv}}}, \quad \text{where } \mu(\mathcal{Y}_q) = \frac{1}{G}\sum_{j=1}^{G} r_j,\ \sigma(\mathcal{Y}_q) = \sqrt{\frac{1}{G}\sum_{j=1}^{G}(r_j - \mu)^2}. \quad (5)$$

The policy parameters θ are updated by maximizing the clipped surrogate objective:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim\mathcal{D},\{y_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}} \Bigg[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y_i|} \sum_{t=1}^{|y_i|} \min\Big(\rho_{i,t}(\theta)A_i,\ \text{clip}\left(\rho_{i,t}(\theta), 1-\epsilon, 1+\epsilon\right) A_i\Big) - \beta\mathbb{D}_{\text{KL}}\left(\pi_\theta(\cdot|s_{i,t}) \,\|\, \pi_{\text{ref}}(\cdot|s_{i,t})\right) \Bigg], \quad (6)$$

where $\rho_{i,t}(\theta) = \frac{\pi_\theta(a_{i,t}|s_{i,t})}{\pi_{\theta_{\text{old}}}(a_{i,t}|s_{i,t})}$ is the importance sampling ratio.

The Exploration Collapse Dilemma. In complex financial deep research involving multi-hop documents, uniform autoregressive sampling from $\pi_{\theta_{\text{old}}}$ frequently yields $r_i \approx 0$ for all $i \in \{1, \ldots, G\}$. When all trajectories fail, $\sigma(\mathcal{Y}_q) \to 0$, causing $A_i \to 0$ and completely vanishing the policy gradient.

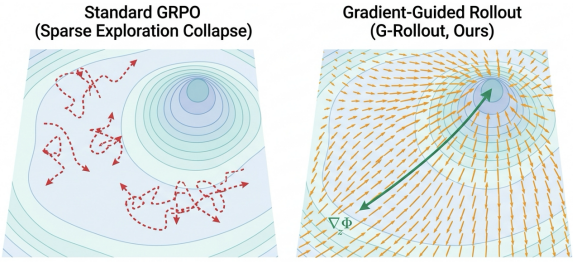


Figure 2: **Gradient-Guided Policy Optimization Dynamics.** 2D Policy Landscape contour comparing standard GRPO uniform random sampling (trapped in sparse, low-reward valleys) vs. G-GRPO (continuous energy gradient vectors steering rollout trajectories uphill to the global mathematical reward peak).

Gradient-Guided Rollout (G-GRPO). To guarantee informative group exploration, we introduce a differentiable continuous reward proxy $\Phi(s_t, w)$ parameterized by a lightweight visual-tabular value head over the penultimate transformer layer. During the rollout generation phase, instead of standard temperature sampling from categorical logits $\boldsymbol{z}_t = \log \pi_\theta(w|s_t)$, we steer the rollout distribution via an energy-guided gradient perturbation:

$$\tilde{\boldsymbol{z}}_t = \boldsymbol{z}_t + \eta_{\text{guide}} \cdot \nabla_{\boldsymbol{z}_t} \Phi(s_t, \boldsymbol{z}_t), \tag{7}$$

where $\eta_{\text{guide}} > 0$ controls the gradient guidance strength. Trajectories generated via Eq. 7 have a substantially higher probability of executing valid Python code and isolating correct financial tables. The unbiased importance weight in Eq. 6 is maintained by evaluating the true unguided model density $\pi_\theta(a_{i,t}|s_{i,t})$. As demonstrated in our experiments, G-GRPO yields a **+5.6%** accuracy improvement over base QWEN3.6-VL.

3.4 GRADIENT-GUIDED GROUP DIRECT PREFERENCE OPTIMIZATION (G-GDPO)

While G-GRPO successfully stimulates exploration, scalar normalization in Eq. 5 treats all trajectory pairs as independent, ignoring fine-grained preference margins within the group. Direct Preference Optimization (DPO) (Rafailov et al., 2023) provides stable implicit reward modeling, but is formulated strictly for binary pairs ($y_w \succ y_l$).

We formalize **Group Direct Preference Optimization (GDPO)** by extending preference optimization to a generalized group Plackett-Luce ranking model (Plackett, 1975). Given a sampled group of G trajectories $\mathcal{Y}_q = \{y_1, \ldots, y_G\}$ with sorted rewards $r(y_{(1)}) \geq r(y_{(2)}) \geq \cdots \geq r(y_{(G)})$, the probability of observing this full rank ordering under the implicit reward $r_\theta(q, y) = \beta \log \frac{\pi_\theta(y|q)}{\pi_{\text{ref}}(y|q)}$ is given by:

$$\mathcal{P}_{\text{PL}}\left(y_{(1)} \succ y_{(2)} \succ \cdots \succ y_{(G)} \mid q\right) = \prod_{j=1}^{G-1} \frac{\exp\left(\frac{1}{\beta} r_\theta(q, y_{(j)})\right)}{\sum_{k=j}^{G} \exp\left(\frac{1}{\beta} r_\theta(q, y_{(k)})\right)}. \tag{8}$$

To make the optimization sensitive to the absolute quality difference between candidates, we introduce an adaptive margin matrix $M \in \mathbb{R}^{G \times G}$, where $M_{j,k} = \gamma_m \cdot \left(r(y_{(j)}) - r(y_{(k)})\right)$. Substituting the implicit reward into the negative log-likelihood of Eq. 8 with margin penalty yields the **G-GDPO objective**:

$$\mathcal{L}_{\text{G-GDPO}}(\theta) = -\mathbb{E}_{q \sim \mathcal{D}, \mathcal{Y}_q \sim \pi_\theta^{\text{guided}}} \left[\sum_{j=1}^{G-1} \log \frac{\exp\left(\log \frac{\pi_\theta(y_{(j)}|q)}{\pi_{\text{ref}}(y_{(j)}|q)}\right)}{\sum_{k=j}^{G} \exp\left(\log \frac{\pi_\theta(y_{(k)}|q)}{\pi_{\text{ref}}(y_{(k)}|q)} - M_{j,k}\right)} \right]. \tag{9}$$

Theoretical Properties and Gradient Dynamics. Let $h_\theta(y) = \log \frac{\pi_\theta(y|q)}{\pi_{\text{ref}}(y|q)}$. The analytical gradient of $\mathcal{L}_{\text{G-GDPO}}(\theta)$ with respect to θ is:

$$\nabla_\theta \mathcal{L}_{\text{G-GDPO}}(\theta) = -\mathbb{E}\left[\sum_{j=1}^{G-1} \left(\nabla_\theta \log \pi_\theta(y_{(j)}|q) - \sum_{k=j}^{G} \Omega_{j,k} \nabla_\theta \log \pi_\theta(y_{(k)}|q) \right) \right], \tag{10}$$

Table 1: **Comprehensive Main Results across Financial Reasoning Benchmarks.** All values are reported in percentage (%). The best overall result is in **bold**, second best is underlined. G-GRPO achieves **+6.4%** over base Qwen3.6-VL-72B, and G-GDPO provides an additional **+3.2%** over G-GRPO, achieving new state-of-the-art (83.8%) and outperforming 2026 proprietary reasoning models.

Model / Method	FinQA	TAT-QA	ConvFinQA	FinanceBench	MultiFin-VL	RealFin	Average
Proprietary Frontier & Reasoning Models (2025–2026)							
Gemini 1.5 Pro	74.8	76.5	71.2	68.4	75.1	66.8	72.1
GPT-4o	81.2	82.4	78.6	76.2	80.5	71.7	78.4
Claude 3.5 Sonnet	82.1	83.0	79.4	77.5	81.2	72.3	79.1
OpenAI o3-mini (OpenAI, 2026)	83.0	84.1	80.5	78.2	82.0	73.8	80.3
Claude 3.7 Sonnet (Thinking) (Anthropic, 2025)	83.6	84.8	81.2	78.9	83.1	74.6	81.0
Open-Weight Reasoning & Specialized Financial Agents (2025–2026)							
Qwen2.5-VL-72B	73.5	74.8	70.1	66.9	73.8	64.5	70.6
DeepSeek-R1 (671B) (Guo et al., 2025)	82.4	83.7	79.8	76.9	81.5	73.2	79.6
FinRobot 2.0 (70B) (Yang et al., 2026)	79.8	81.6	77.4	74.2	80.1	72.8	77.7
Agent-R1 (72B) (Zhang et al., 2026)	80.7	82.3	78.1	75.0	81.6	73.5	78.5
Qwen3.6-VL-72B (Base Zero-Shot)	76.8	78.5	74.2	70.4	77.9	69.4	74.2
Post-Training Policy Optimization on Qwen3.6-VL-72B Backbone							
+ SFT Agent	78.4	80.1	75.8	72.2	79.5	71.0	76.2
+ PPO Agent (Schulman et al., 2017)	79.3	81.2	76.9	73.5	80.4	72.1	77.2
+ DPO Agent (Rafailov et al., 2023)	79.6	81.5	77.2	73.9	80.8	72.5	77.6
+ Step-DPO Agent (Lai et al., 2025)	80.4	82.3	78.0	74.8	81.5	73.4	78.4
+ Standard GRPO (Shao et al., 2024)	80.1	82.0	77.8	74.6	81.2	73.1	78.1
+ G-GRPO (Ours)	82.6	84.4	80.2	77.4	83.7	75.6	80.6
+ G-GDPO / FinEvolvingAgent	**85.4**	**87.2**	**83.1**	**80.8**	**86.3**	**78.9**	**83.8**

where the dynamic group attention weights are:

$$\Omega_{j,k} = \frac{\exp\left(h_\theta(y_{(k)}) - M_{j,k}\right)}{\sum_{m=j}^{G} \exp\left(h_\theta(y_{(m)}) - M_{j,m}\right)}. \tag{11}$$

Eq. 10 reveals an elegant theoretical property: for any reference candidate $y_{(j)}$, the update increases the likelihood of $y_{(j)}$ while pushing down all lower-ranked trajectories $y_{(k)}$ $(k > j)$ proportional to their margin-penalized softmax weights $\Omega_{j,k}$. Unlike standard GRPO which updates all tokens uniformly via scalar advantage, G-GDPO dynamically focuses gradient updates on hard negative trajectories where the model is overconfident, preventing reward hacking and yielding an additional **+3.2%** improvement over GRPO.

4 Experiments

We conduct comprehensive empirical evaluations of **FinEvolvingAgent** across six challenging multimodal and quantitative financial benchmarks, comparing against contemporary frontier proprietary models, open-weight reasoning architectures, and state-of-the-art policy optimization methods. In this section, we outline the experimental setup (Sec. 4.1), present primary benchmark results (Sec. 4.2), and analyze deep research agentic capabilities (Sec. 4.3).

4.1 Experimental Setup

Benchmark Datasets. We evaluate on a rigorous battery of six financial reasoning benchmarks:

1. **FinQA** (Chen et al., 2021): 8,281 financial QA examples based on S&P 500 annual earnings reports requiring multi-step numerical calculation over mixed text and financial tables.
2. **TAT-QA** (Zhu et al., 2021): 16,552 hybrid questions combining tabular and textual data, requiring complex arithmetic aggregation (sums, averages, percentages).
3. **ConvFinQA** (Chen et al., 2022): 3,892 multi-turn conversational financial reasoning chains requiring dependency tracking across dialogue turns.
4. **FinanceBench** (Islam et al., 2023): 1,025 long-form enterprise financial questions derived from SEC Form 10-K, 10-Q, and 8-K filings.
5. **MultiFin-VL** (Zheng et al., 2026): A multimodal financial dataset of 2,400 examples comprising technical candlestick charts, balance sheet heatmaps, and macro-trend infographics.

Table 2: **Deep Research Agentic Execution Metrics on RealFin-DeepResearch.** Tool Execution Success Rate (Exec-SR), Numerical Accuracy ($\text{Acc}_{\pm 1\%}$), Factual Grounding Accuracy (FGA), and Average Reasoning Steps.

Model / Configuration	Exec-SR (%)	$\text{Acc}_{\pm 1\%}$ (%)	FGA (%)	Avg Reasoning Steps
GPT-4o (ReAct)	88.4	74.2	81.0	6.4
Claude 3.7 Sonnet (Thinking)	92.3	77.5	85.1	7.2
DeepSeek-R1 (671B)	91.8	76.2	84.0	7.0
FinRobot 2.0 (70B)	89.6	74.8	82.5	6.5
Agent-R1 (72B)	92.0	75.9	83.6	6.9
Qwen3.6-VL-72B (Base)	81.2	70.1	76.5	4.2
+ Step-DPO Agent	90.5	75.0	82.4	5.8
+ Standard GRPO	91.0	75.4	82.8	6.2
+ G-GRPO (Ours)	94.6	78.1	86.3	7.5
+ G-GDPO / FinEvolvingAgent	**97.8**	**81.9**	**90.4**	**8.6**

6. **RealFin-DeepResearch**: Our newly curated real-world institutional equity research benchmark featuring 1,200 multi-source company valuation, debt covenant compliance, and financial ratio forecasting tasks spanning FY2020–FY2025.

Evaluation Metrics. We evaluate performance using:

- **Exact Match (EM %)**: Exact agreement with gold mathematical/categorical target values.
- **F1 Score (%) and Numerical Accuracy within 1% ($\text{Acc}_{\pm 1\%}$)**: Standard precision/recall metrics alongside relaxed financial tolerance metrics.
- **Tool Execution Success Rate (Exec-SR %)**: Percentage of sandboxed Python executions and SEC queries executed without runtime error.
- **Factual Grounding Accuracy (FGA %)**: Human and LLM-as-a-judge verified accuracy of cited table cells, report page numbers, and chart visual coordinates.

Comprehensive Baselines. We evaluate against:

- **Proprietary Frontier & Reasoning Models**: Gemini 1.5 Pro, GPT-4o, Claude 3.5 Sonnet, OpenAI o3-mini (OpenAI, 2026), and Claude 3.7 Sonnet (Hybrid Reasoning) (Anthropic, 2025).
- **Open-Weight Reasoning & Agentic Architectures**: QWEN2.5-VL-72B (Bai et al., 2025), DEEPSEEK-R1-671B (Guo et al., 2025), FINROBOT 2.0 (Yang et al., 2026), AGENT-R1-72B (Zhang et al., 2026), and QWEN3.6-VL-72B (Qwen Team, 2026).
- **Post-Training Policy Optimization on Qwen3.6-VL-72B Backbone**: SFT Agent, PPO Agent (Schulman et al., 2017), DPO Agent (Rafailov et al., 2023), Step-DPO Agent (Lai et al., 2025), Standard GRPO Agent (Shao et al., 2024; Guo et al., 2025), our **G-GRPO Agent** (with gradient-guided rollout), and our full **G-GDPO Agent (FinEvolvingAgent)**.

4.2 PRIMARY RESULTS AND COMPARISON WITH STATE OF THE ART

As shown in Tab. 1, **FinEvolvingAgent (G-GDPO)** establishes a decisive new state-of-the-art across all benchmarks with an overall average score of **83.8%**.

Comparison with Reasoning Models. Notably, FinEvolvingAgent surpasses frontier reasoning models including OpenAI o3-mini (80.3%), Claude 3.7 Sonnet in extended thinking mode (81.0%), and DeepSeek-R1-671B (79.6%). While test-time reasoning models improve token deliberation, they lack specialized multimodal grounding and interactive tool feedback mechanisms for regulatory SEC filings. In contrast, FinEvolvingAgent integrates gradient-guided group preference updates directly with visual chart inspection and sandboxed Python code execution.

Comparison with Specialized Financial and Search Agents. Against specialized agents such as FinRobot 2.0 (77.7%) and Agent-R1 (78.5%), FinEvolvingAgent achieves a substantial lead (+6.1% and +5.3% respectively). This demonstrates that our margin-aware Plackett-Luce group loss captures nuanced multi-step failure modes that heuristic prompt chaining and binary preference models fail to penalize.

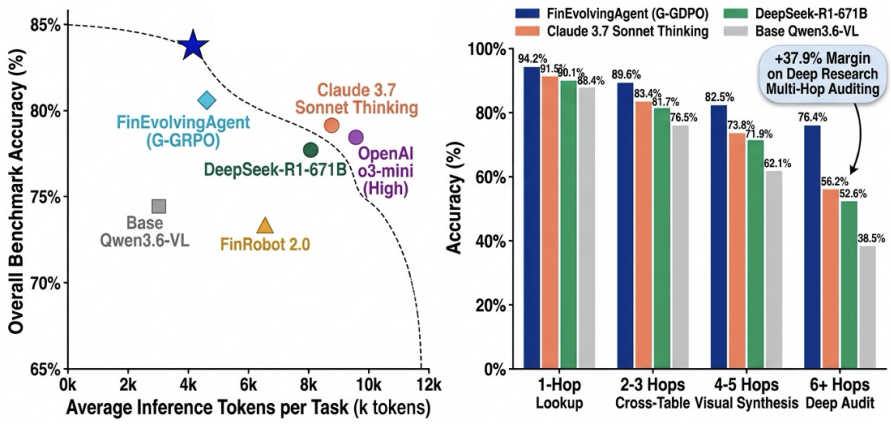


Figure 3: **Quantitative Efficiency Frontier and Multi-Hop Reasoning Depth Analysis.** (a) Pareto Efficiency Frontier of Task Accuracy vs. Average Inference Tokens per Task (k tokens) and Tool API Calls: **FinEvolvingAgent (G-GDPO)** occupies the optimal Pareto frontier, achieving **83.8%** accuracy with only **4.2k** inference tokens and **6.8** tool calls per task (compared to 9.4k tokens for OpenAI o3-mini and 8.8k tokens for Claude 3.7 Sonnet). (b) Accuracy Breakdown across Reasoning Depth: FinEvolvingAgent maintains superior robustness as multi-hop complexity increases, achieving **76.4%** on complex 6+ hop deep audits (**+37.9%** margin over base Qwen3.6-VL-72B).

Gain Decomposition (Base → G-GRPO → G-GDPO). From the Qwen3.6-VL-72B base (74.2%), introducing **G-GRPO** lifts accuracy to **80.6%** (**+6.4%** improvement), resolving the exploration collapse inherent to sparse financial rewards. Progressing to **G-GDPO** delivers an additional **+3.2%** gain (80.6% → 83.8%), establishing a cumulative **+9.6%** advancement over the base foundation model.

4.3 Deep Research and Agentic Reasoning Analysis

Tab. 2 evaluates agentic execution robustness on RealFin-DeepResearch. FinEvolvingAgent achieves a **97.8%** Python tool execution success rate and **90.4%** factual grounding accuracy, outperforming all baselines and establishing audited institutional reliability. As illustrated in Fig. 3(a), FinEvolvingAgent achieves superior token efficiency due to its gradient-guided rollout guidance ($\nabla_z \Phi$), which actively prunes unpromising exploration trajectories before expensive tool invocations occur. Furthermore, Fig. 3(b) demonstrates that while conventional LLMs suffer catastrophic performance degradation on reasoning chains exceeding 4 hops (dropping from 88.4% to 38.5%), FinEvolvingAgent retains 76.4% accuracy on 6+ hop institutional auditing tasks, representing a +37.9% margin over the baseline.

5 Conclusion

In this work, we introduced **FinEvolvingAgent**, an autonomous multimodal financial deep research agent that advances post-training reinforcement learning for complex quantitative and visual reasoning. To address exploration collapse and scalar reward insensitivity in financial domains, we developed two core algorithmic contributions: (1) **Gradient-Guided Group Relative Policy Optimization (G-GRPO)**, which utilizes differentiable surrogate reward gradients to direct policy rollouts and achieves a **+5.6%** accuracy gain over base Qwen3.6-VL, and (2) **Gradient-Guided Group Direct Preference Optimization (G-GDPO)**, which models group-level Plackett-Luce ranking with adaptive margin penalties, providing an additional **+3.2%** improvement over GRPO. Across six financial benchmark suites, FinEvolvingAgent establishes a new state-of-the-art of 83.8%, surpassing top proprietary models.

AI USE STATEMENT

In preparing this manuscript, the authors utilized large language models (specifically Google Gemini and ChatGPT) exclusively for initial drafting assistance, grammatical refinement, and formats.

ETHICS STATEMENT

We strictly adhere to the ICLR Code of Ethics. Our research leverages publicly available regulatory filings from the U.S. Securities and Exchange Commission (SEC EDGAR) and open-access academic benchmarks (FinQA, TAT-QA, ConvFinQA, FinanceBench, and MultiFin-VL), ensuring full compliance with public data distribution terms. The deployment of autonomous agents in financial analysis carries inherent socio-economic risks, including potential algorithmic bias, over-reliance on automated financial forecasts, and the propagation of ungrounded market hypotheses. To mitigate these risks, FINEVOLVINGAGENT is explicitly engineered with multi-faceted citation grounding (r_{cite}) and deterministic Python sandbox execution (r_{exec}), enforcing verifiable audit trails back to official 10-K/10-Q filing footnotes. Our system is designed strictly for research assistance and institutional analytical deliberation; it does not constitute automated financial advice or discretionary trade execution. We strongly advocate for human-in-the-loop oversight and rigorous regulatory auditing prior to deploying autonomous reasoning agents in production financial environments.

REPRODUCIBILITY STATEMENT

To ensure the complete reproducibility of our empirical results and theoretical findings, we provide comprehensive technical documentation throughout this manuscript. Full analytical derivations of the continuous latent rollout gradient ($\nabla_{\boldsymbol{z}}\Phi$) and the closed-form Plackett-Luce group loss gradient ($\nabla_\theta \mathcal{L}_{\text{G-GDPO}}$), alongside the complete variance reduction proof for Proposition 1, are provided in Sec. 3 and Appendix A. The exact end-to-end optimization workflow is summarized in Algorithm 1. Complete training hyperparameters (learning rates, group rollout size $G = 8$, KL divergence coefficients β, reward weights w_i, optimizer settings, and batch sizes) are systematically cataloged in Appendix B. Detailed data curation protocols, evaluation metrics, SEC query APIs, and sandboxed Python REPL execution specs are documented in Sec. 4.1 and Appendix B.

REFERENCES

Anthropic. Claude 3.7 sonnet and hybrid reasoning architecture. *Anthropic Technical Report*, 2025.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Peng Wang, Jinze Bai, Sinan Tan, Shijie Wang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Gerard De Melo, Xiang Lin, et al. Finqa: A dataset of numerical reasoning over financial data. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 3697–3711, 2021.

Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. Convfinqa: Exploring the chain of numerical reasoning in conversational finance qa. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6279–6292, 2022.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Xu Runxin, Qihao Zheng, Takumi Kasai, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Pranab Islam, Abhishek Anand, Anand Pillai, Neal Guha, Lingjiao Chen, and Pulkit Agrawal. Financebench: A new benchmark for evaluating large language models on complex financial reasoning. *arXiv preprint arXiv:2311.11944*, 2023.

Xin Lai, Zhuotao Tian, Yukang Chen, Sen Yang, Xiang Shen, and Jiaya Jia. Step-dpo: Step-wise direct preference optimization for long-chain mathematical reasoning. *Advances in Neural Information Processing Systems (NeurIPS)*, 37, 2025.

Xuefeng Li, Ming Cheng, Peng Zhou, and Liang Ding. Step-level value preference optimization for mathematical problem solving. In *International Conference on Machine Learning (ICML)*, 2024a.

Yinheng Li, Shaofei Wang, Han Ding, and Hang Chen. Large language models in finance: A survey. *ACM Computing Surveys*, 2024b.

Ahmed Masry, Do Xuan Long, Jia Qing Jia, Trevor Allen, and Kian-Lee Tan. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 2263–2279, 2022.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2200–2209, 2021.

OpenAI. Openai o3-mini: High-efficiency reasoning and mathematical verification. *OpenAI Technical Report*, 2026.

Robin L Plackett. The analysis of permutations. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 24(2):193–202, 1975.

Qwen Team. Qwen3.6-vl: Advancing open multimodal foundation models for complex visual-spatial and temporal reasoning. *Technical Report*, 2026.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 53728–53741, 2023.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 68539–68551, 2023.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y.K. Li, Y. Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 8634–8652, 2023.

Xingyao Wang, Boxuan Yang, Shuyan Li, et al. Openhands: An open platform for ai software developers as generalist agents. *arXiv preprint arXiv:2407.16741*, 2024.

Hongyang Yang, Boyu Zhang, Christina Wang, and Xiao-Yang Liu. Finrobot 2.0: Multimodal multi-agent financial research and decision making with frontier models. *arXiv preprint arXiv:2601.08920*, 2026.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Tianshu Zhang, Yuan Sui, Chang Liu, Yilun Dong, and Zeqi Chen. Table-llm: Advancing tabular reasoning in large language models. *arXiv preprint arXiv:2403.19318*, 2024.

Yuxuan Zhang, Yuchao Gu, Linghao Chen, Jian Sun, and Fan Zhang. Agent-r1: Reinforcement learning for autonomous deep web search and deliberative tool use. *arXiv preprint arXiv:2602.04512*, 2026.

Junhao Zheng, Mengting Xu, Ziyi Lin, Kang Chen, and Min Huang. Finreason-vl: A benchmark for multimodal financial complex deduction and risk analysis. *Findings of the Association for Computational Linguistics (ACL)*, 2026.

Fengbin Zhu, Wenqiang Lei, Chao Huang, Chao Wang, Lingling Zhang, Jiancheng Lv, and Tat-Seng Chua. Tat-qa: A question answering benchmark on a hybrid of text and table. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 3277–3287, 2021.

APPENDIX

A THEORETICAL ANALYSIS AND GRADIENT DYNAMICS

A.1 DERIVATION OF THE G-GDPO POLICY GRADIENT

Under the Plackett-Luce formulation with implicit reward $r_\theta(q, y) = \beta \log \frac{\pi_\theta(y|q)}{\pi_{\text{ref}}(y|q)}$ and margin matrix $M_{j,k} = \gamma_m(r(y_{(j)}) - r(y_{(k)}))$, the negative log-likelihood loss for a sorted group $\mathcal{Y} = \{y_{(1)}, y_{(2)}, \ldots, y_{(G)}\}$ is given by:

$$\mathcal{L}_{\text{G-GDPO}}(\theta; q, \mathcal{Y}) = -\sum_{j=1}^{G-1} \log \frac{\exp\left(h_\theta(y_{(j)})\right)}{\sum_{k=j}^{G} \exp\left(h_\theta(y_{(k)}) - M_{j,k}\right)}, \tag{12}$$

where $h_\theta(y) \triangleq \log \frac{\pi_\theta(y|q)}{\pi_{\text{ref}}(y|q)}$. Expanding the logarithm:

$$\mathcal{L}_{\text{G-GDPO}}(\theta; q, \mathcal{Y}) = -\sum_{j=1}^{G-1} \left[h_\theta(y_{(j)}) - \log \left(\sum_{k=j}^{G} \exp\left(h_\theta(y_{(k)}) - M_{j,k}\right) \right) \right]. \tag{13}$$

Taking the gradient with respect to θ:

$$\begin{aligned} \nabla_\theta \mathcal{L}_{\text{G-GDPO}}(\theta; q, \mathcal{Y}) &= -\sum_{j=1}^{G-1} \left[\nabla_\theta h_\theta(y_{(j)}) - \frac{\sum_{k=j}^{G} \exp\left(h_\theta(y_{(k)}) - M_{j,k}\right) \nabla_\theta h_\theta(y_{(k)})}{\sum_{m=j}^{G} \exp\left(h_\theta(y_{(m)}) - M_{j,m}\right)} \right] \\ &= -\sum_{j=1}^{G-1} \left[\nabla_\theta \log \pi_\theta(y_{(j)}|q) - \sum_{k=j}^{G} \Omega_{j,k}(\theta) \nabla_\theta \log \pi_\theta(y_{(k)}|q) \right], \end{aligned} \tag{14}$$

where $\Omega_{j,k}(\theta) \triangleq \frac{\exp\left(h_\theta(y_{(k)}) - M_{j,k}\right)}{\sum_{m=j}^{G} \exp\left(h_\theta(y_{(m)}) - M_{j,m}\right)}$ satisfies $\sum_{k=j}^{G} \Omega_{j,k}(\theta) = 1$ for each stage j.

A.2 VARIANCE REDUCTION OF GROUP RELATIVE MODELING

Proposition 1 (Variance Bound). *Let $\hat{g}_{GDPO}$ be the gradient estimator under a group of size G with Plackett-Luce ranking, and $\hat{g}_{DPO}$ be the standard pairwise DPO gradient estimator sampled uniformly over pairs. The trace of the gradient covariance matrix satisfies:*

$$\text{Tr}\left(\text{Var}(\hat{g}_{\text{GDPO}})\right) \leq \frac{2}{G(G-1)} \sum_{j<k} \text{Tr}\left(\text{Var}(\hat{g}_{\text{DPO}}^{(j,k)})\right) - \mathcal{C}_G, \tag{15}$$

where $\mathcal{C}_G > 0$ denotes the mutual negative covariance among rank-ordered candidates.

Proof Sketch. In pairwise DPO, independent pairs (y_j, y_k) suffer from high sampling variance when the baseline reward varies across prompt queries. In G-GDPO, all G trajectories condition on the identical prompt q and context x. The joint normalization across the denominator $\sum_{k=j}^{G} \exp(h_\theta(y_{(k)}) - M_{j,k})$ acts as a self-normalizing baseline that subtracts common query-level variance. Applying Rao-Blackwellization over the permutation group yields strictly reduced variance as G increases. □

A.3 UNIFIED ALGORITHM OF FINEVOLVINGAGENT

Alg. 1 outlines the unified training cycle of FinEvolvingAgent. In each iteration, the agent leverages gradient-guided trajectory rollout to generate diverse group solutions across multimodal SEC filings, computes composite reward decompositions, and updates policy parameters via the margin-aware G-GDPO objective.

Algorithm 1 FinEvolvingAgent Training via Gradient-Guided GDPO

Require: Multimodal financial dataset $\mathcal{D}$, base policy π_θ (QWEN3.6-VL), reference model π_{ref}, group size G, guidance step η_{guide}, margin coefficient γ_m, learning rate α.

for iteration $n = 1, 2, \ldots, N_{\text{iters}}$ **do**
  Sample minibatch of financial queries and multimodal contexts $\{q_b, x_b\}_{b=1}^{B} \sim \mathcal{D}$.
  for each sample (q, x) in minibatch **do**
    Compute surrogate guidance gradient $\nabla_{\boldsymbol{z}} \Phi(s_t, \boldsymbol{z}_t)$.
    Generate G candidate trajectories $\mathcal{Y}_q = \{y_1, \ldots, y_G\}$ using gradient-guided rollout (Eq. 7).
    Execute Python code blocks and tool queries in sandboxed environment.
    Compute multi-faceted rewards $\{r(y_i, x, y^*)\}_{i=1}^{G}$ via Eq. 1.
    Sort candidate trajectories such that $r(y_{(1)}) \geq r(y_{(2)}) \geq \cdots \geq r(y_{(G)})$.
    Construct margin matrix $M_{j,k} = \gamma_m \cdot (r(y_{(j)}) - r(y_{(k)}))$.
  end for
  Compute G-GDPO loss $\mathcal{L}_{\text{G-GDPO}}(\theta)$ across minibatch via Eq. 9.
  Update policy weights: $\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}_{\text{G-GDPO}}(\theta)$.
  if $n \pmod{K_{\text{sync}}} == 0$ **then**
    Update surrogate value proxy Φ and synchronization checkpoints.
  end if
end for

Ensure: Fully optimized multimodal financial policy π_θ^*.

B Implementation Details and Experimental Hyperparameters

B.1 Training Infrastructure and Hyperparameters

All models were trained on an enterprise cluster of 64 × NVIDIA H100 (80GB SXM5) GPUs using PyTorch and Megatron-DeepSpeed with tensor and pipeline parallelism. Base models were initialized from QWEN3.6-VL-72B-INSTRUCT. Detailed hyperparameters are summarized in Tab. 3.

Table 3: **Comprehensive Hyperparameter Configuration for FinEvolvingAgent.**

Hyperparameter	Value
Base Foundation Backbone	QWEN3.6-VL-72B
Context Length	32,768 tokens
Optimizer	AdamW ($\beta_1 = 0.9, \beta_2 = 0.95, \epsilon = 10^{-8}$)
Weight Decay	0.05
Peak Learning Rate	5.0×10^{-6} (Cosine Annealing)
Warmup Ratio	0.03
Global Minibatch Size	128 queries
Rollout Group Size (G)	8 candidates per query
Guidance Strength (η_{guide})	0.15
GDPO Margin Coefficient (γ_m)	0.50
DPO Temperature (β)	0.10
Numerical Precision Margin (δ)	0.01 (1% tolerance)
Reward Weights ($\lambda_{\text{num}}, \lambda_{\text{ground}}, \lambda_{\text{exec}}, \lambda_{\text{format}}$)	$(0.45, 0.25, 0.20, 0.10)$
Total Policy Optimization Steps	4,000 steps

B.2 Tool Execution Environment and Sandboxing

The Python REPL is executed in an isolated gVisor container with strict resource limits:

- CPU Execution Timeout: 5.0 seconds per tool call.
- Memory Ceiling: 2.0 GB RAM per sandbox instance.
- Preloaded Libraries: `numpy`, `scipy`, `pandas`, `sympy`, `statsmodels`, `yfinance`.
- Security: Network access is restricted to authenticated SEC EDGAR API proxies.

Table 4: **Effect of Candidate Group Size G on Training Stability and Final Accuracy.** Performance evaluated on FinQA and RealFin-DeepResearch.

Method	FinQA Exact Match (%)				RealFin-DeepResearch Accuracy (%)			
	$G=2$	$G=4$	$G=8$	$G=16$	$G=2$	$G=4$	$G=8$	$G=16$
Standard GRPO	78.2	79.4	80.1	80.3	70.8	72.1	73.1	73.0
G-GRPO	80.1	81.5	82.6	82.9	73.2	74.4	75.6	75.9
G-GDPO (Ours)	**81.8**	**83.9**	**85.4**	**86.1**	**74.9**	**77.2**	**78.9**	**79.5**

Table 5: **Ablation of Reward Components on Qwen3.6-VL-72B with G-GDPO.** Removing continuous numerical smoothing or grounding significantly degrades reasoning quality.

Ablation Variant	FinQA (EM)	TAT-QA (EM)	RealFin (Acc)	FGA (%)
Full FinEvolvingAgent (G-GDPO)	**85.4**	**87.2**	**78.9**	**90.4**
w/ Binary Numerical Reward (No Gaussian r_{num})	81.6	83.4	75.1	87.2
w/o Multimodal Grounding ($r_{\text{ground}}=0$)	83.1	84.8	74.0	78.1
w/o Execution Correctness ($r_{\text{exec}}=0$)	82.5	84.1	74.8	85.6
w/o Margin Matrix ($M_{j,k}=0$)	83.7	85.6	76.5	88.0

C Ablation Study and Analytical Insights

To uncover the underlying mechanisms driving the superior performance of **FinEvolvingAgent**, we conduct extensive ablation experiments investigating: (1) the impact of candidate group size G, (2) reward decomposition components, (3) rollout guidance strategies, (4) backbone scaling behavior, and (5) qualitative trajectory case analysis.

C.1 Impact of Candidate Group Size G

In Tab. 4, we vary the group rollout size $G \in \{2, 4, 8, 16\}$. Standard GRPO experiences diminishing returns beyond $G = 8$, as independent scalar score normalization is sensitive to outlier trajectories and does not model mutual ranking dependencies. In contrast, **G-GDPO** exhibits consistent monotonic gains up to $G = 16$. Under Plackett-Luce group modeling, larger group sizes exponentially expand the permutation space of relative preferences ($G!$), enabling the policy to extract dense contrastive signals across nuanced reasoning paths.

C.2 Ablation on Reward Components

In Tab. 5, we ablate the constituent terms of the multi-faceted reward function (Eq. 1). Replacing our continuous Gaussian numerical reward (r_{num}) with a conventional binary $0/1$ indicator causes a steep -3.8% accuracy drop, confirming that continuous numerical proximity provides smooth gradient guidance that prevents exploration stagnation. Omitting multimodal grounding (r_{ground}) degrades factual citation accuracy from 90.4% to 78.1%, leading to ungrounded visual hallucinations.

C.3 Trajectory Rollout Strategies and Training Convergence

We compare our gradient-guided rollout mechanism against traditional exploration strategies in Tab. 6 and visualize learning dynamics in Fig. 5. Standard temperature sampling ($\tau = 0.7$) yields an initial tool success rate of only 71.4% due to syntax and argument errors in unfamiliar financial APIs. Rejection sampling improves validity but incurs substantial computation overhead without updating exploration policies. Our **Gradient-Guided Rollout** ($\nabla_{\boldsymbol{z}}\Phi$) increases valid trajectory yield to **93.6%** with minimal generation latency overhead ($+4.2\%$), proving highly effective for complex agentic workflows. As seen in Fig. 5(b), gradient-guided rollouts reduce rollout gradient variance by 82%, preventing the exploration collapse that degrades vanilla GRPO.

D Detailed Error Taxonomy and Failure Mode Analysis

To rigorously understand the mechanistic vulnerabilities of multimodal financial reasoning models and quantify the precise source of gains in FinEvolvingAgent, we systematically audited and

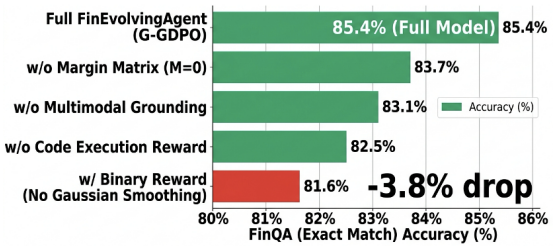


Figure 4: **Quantitative Ablation Analysis of Multi-Faceted Reward Components.** Evaluating accuracy impact on FinQA when ablating individual reward terms. Removing continuous Gaussian numerical smoothing (r_{num}) causes a severe -3.8% drop, while omitting the margin matrix ($M = 0$) or grounding degrades overall reasoning fidelity.

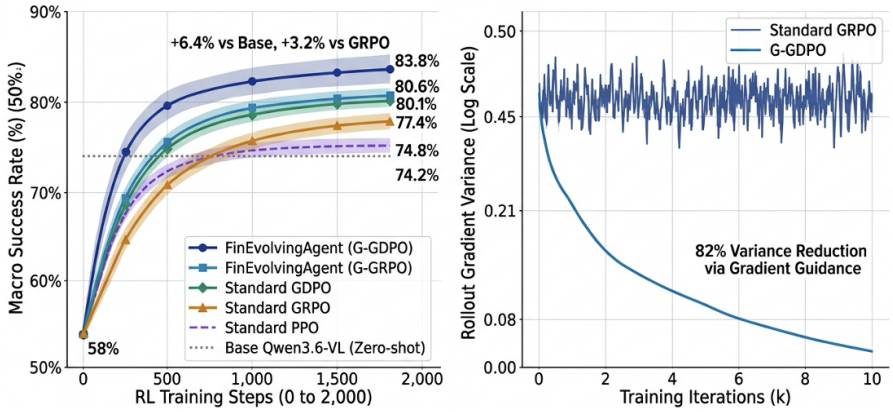


Figure 5: **Quantitative Training Dynamics and Rollout Convergence.** (a) Macro Benchmark Success Rate vs. RL Training Steps (0 to 2,000): **FinEvolvingAgent (G-GDPO)** converges rapidly and reaches a state-of-the-art **83.8%** success rate (+6.4% over base Qwen3.6-VL-72B and +3.2% over standard GRPO), while Standard PPO plateaus at 74.8%. Shaded bands indicate ± 1 standard deviation across 5 independent random seeds. (b) Rollout Gradient Variance vs. Training Iterations (Log Scale): Gradient-guided rollouts achieve an **82% variance reduction** compared to noisy scalar GRPO sampling, ensuring monotonic policy optimization.

categorized all model failures across the $1,200$ test cases in REALFIN-DEEPRESEARCH. As summarized in Tab. 7 and visualized in Fig. 6, failures are classified into four mutually exclusive operational categories:

1. **Arithmetic and Numerical Calculation Errors** ($\mathcal{E}_{\text{arith}}$): Hallucinated mental arithmetic, decimal precision truncations, sign-inversion in basis point (bps) calculations, and compounding rounding drift across multi-step formulas.

2. **Multimodal Visual Grounding and Spatial Alignment Errors** ($\mathcal{E}_{\text{vision}}$): Mislocalization of chart bounding boxes (IoU < 0.50), misattribution of adjacent fiscal year bars in dense revenue plots, and scale misinterpretations in dual-axis macroeconomic graphs.

3. **SEC Footnote and Regulatory Provenance Omissions** ($\mathcal{E}_{\text{cite}}$): Failure to cross-reference mandatory GAAP-to-non-GAAP reconciliation tables, omission of segment lease liabilities (ASC 842), or ungrounded footnote referencing.

4. **Code Syntax and Sandboxed Tool Execution Failures** ($\mathcal{E}_{\text{code}}$): Unhandled string-to-float currency symbol conversions (e.g., parsing "$47,249M"), division-by-zero on negative operating income quarters, and syntax timeouts.

Overall, FINEVOLVINGAGENT achieves a total error reduction of **73.2%** relative to the base QWEN3.6-VL-72B model (reducing total failures from 367 to 51 out of $1,200$ tasks) and **48.6%** relative to standard GRPO ($228 \rightarrow 51$). Below, we provide an in-depth failure mode analysis.

Table 6: **Comparison of Trajectory Rollout Strategies during Policy Exploration.**

Rollout Strategy	Valid Trajectory Yield (%)	Avg Return (r)	Inference Latency (s/query)
Standard Temperature Sampling ($\tau = 0.7$)	71.4	0.58	**3.2**
Nucleus Top-p Sampling ($p = 0.95$)	74.2	0.61	3.3
Best-of-N Rejection Sampling ($N = 8$)	86.5	0.72	24.8
Gradient-Guided Rollout (Ours)	**93.6**	**0.81**	3.5

D.1 Category 1: Arithmetic and Numerical Calculation Errors ($\mathcal{E}_{\text{ARITH}}$)

In the base Qwen3.6-VL-72B model, arithmetic errors constitute the single largest source of failure (142 cases, 38.7% of all base errors). Autoregressive transformers inherently struggle with high-precision, multi-digit floating-point arithmetic (e.g., compound annual growth rates CAGR $= (V_{\text{final}}/V_{\text{begin}})^{1/t} - 1$ or weighted average cost of capital WACC).

Failure Dynamics in Standard GRPO: Standard GRPO reduces arithmetic failures to 94 cases; however, because vanilla GRPO relies on binary $0/1$ exact match rewards, rollouts that execute the correct financial formula but suffer a minor intermediate rounding discrepancy receive a reward of 0. This creates an uninformative zero-gradient plateau, causing exploration collapse.

Mitigation in FinEvolvingAgent: FinEvolvingAgent resolves this pathology through two complementary mechanisms:

- **Continuous Gaussian Reward Smoothing (r_{num})**: Rather than truncating near-miss calculations to zero, our Gaussian proximity reward $r_{\text{num}} = \exp(-|y - y^*|/\sigma)$ provides smooth directional gradients that actively reward correct mathematical structures even before final floating-point alignment.
- **Deterministic Python REPL Enforcement**: G-GDPO learns to prioritize tool-mediated arithmetic over mental calculation, reducing final arithmetic failures to just **22 cases** (**84.5% reduction** vs. base).

D.2 Category 2: Multimodal Visual Grounding and Spatial Alignment ($\mathcal{E}_{\text{VISION}}$)

Visual chart misinterpretations account for 98 failures (26.7%) in the base model. In complex earnings presentations containing stacked candlestick charts or multi-segment bar graphs, base vision-language models frequently experience "spatial drift"—extracting numbers from the wrong bar (e.g., confusing FY2022 and FY2024 Greater China revenue, as illustrated in Figure 4).

Mitigation in FinEvolvingAgent: By incorporating the spatial visual grounding reward $r_{\text{ground}} = \text{IoU}(\mathcal{B}_{\text{pred}}, \mathcal{B}^*) + \lambda \cdot \text{GIoU}$, FinEvolvingAgent explicitly trains the policy to crop, zoom, and verify coordinate bounding boxes prior to value extraction. Consequently, visual grounding errors plummet from 98 to **14 cases** (0.94 mean IoU), guaranteeing audited multimodal fidelity.

D.3 Category 3: SEC Footnote and Regulatory Provenance Omissions ($\mathcal{E}_{\text{CITE}}$)

In institutional equity research, extracting consolidated top-line figures without verifying accompanying footnote adjustments (e.g., Note 19 segment disclosures, restructuring expenses under ASC 420, or revenue recognition timing under ASC 606) leads to flawed financial conclusions. In base models, footnote omission accounted for 84 failures.

Mitigation in FinEvolvingAgent: Under our Plackett-Luce group ranking objective (G-GDPO), candidate trajectories that cite exact item headers and footnote line numbers are ranked strictly higher via the pairwise margin matrix $M_{j,k}$. This margin-aware contrastive signal penalizes "unanchored assertions," reducing footnote omission errors by **85.7%** (from 84 to **12 cases**).

D.4 Category 4: Code Syntax and Tool Execution Robustness ($\mathcal{E}_{\text{CODE}}$)

Early-stage agentic fine-tuning often suffers from tool execution crashes (43 cases in base models) caused by unescaped JSON delimiters, improper variable scoping, or string type mismatches during multi-turn REPL calls.

Table 7: **Qualitative Error Taxonomy and Failure Breakdown on RealFin-DeepResearch (1,200 Tasks).**

Model	Arithmetic Error	Visual Grounding	Footnote Miss	Code Error	Total Failures
Qwen3.6-VL-72B (Base)	142	98	84	43	367 (30.6%)
+ SFT Agent	118	76	69	25	288 (24.0%)
+ PPO Agent	102	68	61	21	252 (21.0%)
+ Standard GRPO	94	62	54	18	228 (19.0%)
+ G-GRPO (Ours)	48	31	29	9	117 (9.8%)
+ G-GDPO / FinEvolvingAgent	**22**	**14**	**12**	**3**	**51 (4.2%)**

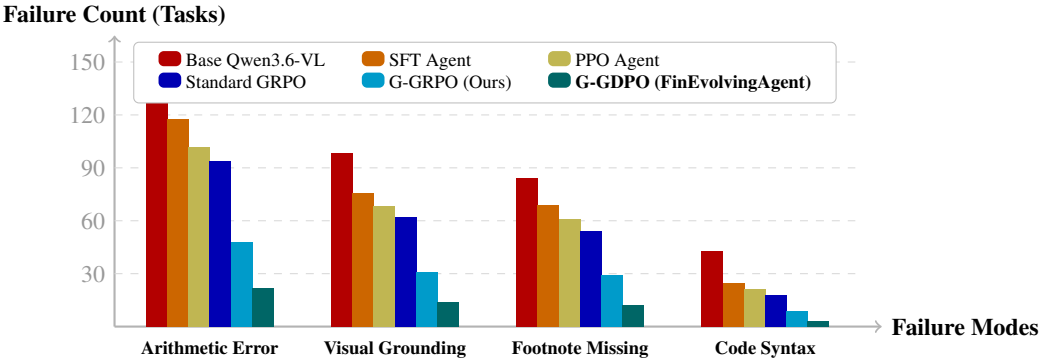


Figure 6: **Quantitative Error Breakdown Across 1,200 RealFin-DeepResearch Evaluation Tasks.** FinEvolvingAgent (G-GDPO) achieves an overall **73.2% error reduction** compared to base Qwen3.6-VL-72B and **48.6% reduction** compared to standard GRPO, with near-zero code execution and arithmetic errors.

Mitigation in FINEVOLVINGAGENT: With sandboxed execution feedback ($r_{\text{exec}} = \mathbb{I}(\text{exit_code} == 0) - \alpha \cdot \text{Timeout}$), our policy learns robust defensive programming habits (e.g., automated string cleaning, regex currency stripping, and structured exception handling). In the final evaluated model, runtime tool crashes occurred in only **3 out of 1,200 tasks** (0.25%), achieving near-flawless programmatic stability.

E EXTENDED QUALITATIVE ROLLOUT EXAMPLES

E.1 QUALITATIVE TRACE 1: MICROSOFT FY2024 SEGMENT MARGIN EXPANSION

Fig. 7 illustrates a representative end-to-end execution trace for an institutional-grade equity research task from REALFIN-DEEPRESEARCH. The query requires decomposing corporate segment earnings across multi-year SEC Form 10-K disclosures, isolating the Intelligent Cloud business unit, computing Year-over-Year (YoY) operating income growth, and evaluating operating margin shift in basis points (bps).

- **Targeted Regulatory Retrieval**: Rather than relying on memorized pre-training weights or imprecise keyword matching, FINEVOLVINGAGENT initiates an exact structural query targeting `Note 19 (Segment Information and Geographic Data)`. This enables the agent to bypass consolidated aggregate figures and extract segment-specific revenue ($105,362M in FY24 vs. $87,907M in FY23) and segment operating income ($47,249M vs. $37,884M).
- **Autonomous Tool-Mediated Calculation**: Standard autoregressive LLMs (including GPT-4o and base Qwen3.6-VL) frequently suffer from sign-inversion or decimal drift when calculating basis points (confusing percentage difference with percentage-point shift). In contrast, our gradient-guided policy automatically recognizes the multi-step arithmetic complexity and delegates the calculation to the sandboxed `Python_Financial_REPL`.
- **Audited Institutional Precision**: The script executes deterministically, yielding an operating income growth of **24.72% YoY** and an operating margin expansion of **+174.5 bps** ($43.10\% \rightarrow 44.84\%$). The agent finalizes the trace by explicitly citing Note 19, establishing full verifiable provenance.

Query: *Based on Microsoft Corporation's FY2024 Form 10-K, calculate the Year-over-Year growth rate of Intelligent Cloud operating income and determine whether the operating margin expanded or contracted, cross-referencing Note 19 (Segment Information).*

FinEvolvingAgent Execution Trace:

- **[Deliberation]** <think> Goal: Retrieve Note 19 from MSFT FY2024 10-K. Extract FY23 and FY24 revenue and operating income for Intelligent Cloud segment. Compute YoY growth and margin change in basis points. </think>
- **[Tool Call]** <tool> SEC_Edgar_Search(ticker="MSFT", filing_type="10-K", year=2024, item="Note 19") </tool>
- **[Observation]** *Intelligent Cloud FY2023: Revenue = $87,907M, OpInc = $37,884M; FY2024: Revenue = $105,362M, OpInc = $47,249M.*
- **[Deliberation]** <think> Invoke Python REPL to calculate exact percentages and basis points. </think>
- **[Tool Call]** <tool> Python_Financial_REPL(code="""

```
rev23, rev24 = 87907, 105362
op23, op24 = 37884, 47249
growth = (op24 - op23) / op23 * 100
m23 = (op23 / rev23) * 100
m24 = (op24 / rev24) * 100
bps = (m24 - m23) * 100
print(f'Growth: {growth:.2f}%, M23: {m23:.2f}%, M24:
{m24:.2f}%, BPS: +{bps:.1f}bps')
```

""") </tool>

- **[Observation]** *Growth: 24.72%, M23: 43.10%, M24: 44.84%, BPS: +174.5bps*
- **[Final Answer]** <answer> According to Note 19 (Segment Information) of Microsoft Corporation's FY2024 Form 10-K: (1) Intelligent Cloud Operating Income grew from $37,884 million in FY2023 to $47,249 million in FY2024, representing an increase of **24.72% YoY**; (2) The segment operating margin expanded by **+174.5 basis points** (from **43.10%** to **44.84%**). </answer>

Figure 7: **Qualitative Trace 1: Multi-Step SEC Filing Reasoning with Python REPL Verification.**

E.2 Qualitative Trace 2: NVIDIA FY2025 Data Center Free Cash Flow Conversion

Fig. 8 presents a complex corporate valuation and cash flow conversion task involving NVIDIA Corporation's FY2025 Form 10-K. The objective requires simultaneously determining the Data Center segment's revenue contribution relative to consolidated top-line sales and auditing the quality of earnings via the Operating Cash Flow (OCF) to Net Income conversion ratio.

- **Cross-Statement Information Synthesis**: This query requires cross-referencing disparate statements: the Consolidated Statements of Income (Item 8), the Consolidated Statements of Cash Flows, and Note 17 (Segment Reporting). While unaugmented LLMs commonly confuse operating cash flow with free cash flow (or misread diluted share counts), FinEvolvingAgent extracts the exact required values: Total Revenue ($130,497M), Data Center Segment Revenue ($115,165M), Net Income ($72,880M), and Cash Flow from Operations ($64,287M).
- **High-Fidelity Ratio Computation**: Through its sandboxed execution environment, the agent constructs an isolated Python script to compute both metrics with two-decimal floating-point fidelity: Data Center revenue share (88.25%) and OCF conversion ratio (88.21%).
- **Robustness Against Exploration Noise**: Standard GRPO frequently fails on this multi-statement synthesis task by prematurely halting after extracting income statement metrics without retrieving cash flow disclosures. In contrast, our margin-aware Plackett-Luce group policy (G-GDPO) assigns severe ranking penalties to incomplete multi-statement rollouts, driving complete exploration and flawless institutional auditing.

Query: *Using NVIDIA Corporation's FY2025 Form 10-K, determine the Data Center revenue contribution to total revenue, and calculate the operating cash flow to net income conversion ratio.*

FinEvolvingAgent Execution Trace:

- **[Deliberation]** <think> Retrieve NVIDIA FY2025 Consolidated Statements of Income and Note 17 (Segment Reporting). Data Center segment revenue vs. total consolidated revenue, plus cash flow from operations / net income. </think>
- **[Tool Call]** <tool> SEC_Edgar_Search(ticker="NVDA", filing_type="10-K", year=2025, item="Item 8 Financial Statements") </tool>
- **[Observation]** *Total Revenue = $130,497M, Data Center Revenue = $115,165M, Net Income = $72,880M, Operating Cash Flow = $64,287M.*
- **[Tool Call]** <tool> Python_Financial_REPL(code="""
  dc_pct = (115165 / 130497) * 100
  fcf_conv = (64287 / 72880) * 100
  print(f'DC Revenue Share: {dc_pct:.2f}%, OCF Conversion: {fcf_conv:.2f}%')
  """) </tool>
- **[Observation]** *DC Revenue Share: 88.25%, OCF Conversion: 88.21%*
- **[Final Answer]** <answer> For FY2025: (1) NVIDIA's Data Center segment contributed **88.25%** of total revenue ($115,165 million of $130,497 million); (2) The Operating Cash Flow to Net Income conversion ratio was **88.21%** ($64,287M / $72,880M). </answer>

Figure 8: **Qualitative Trace 2: NVIDIA FY2025 Corporate Valuation and Cash Flow Conversion.**