ICML
International Conference
On Machine Learning

Towards Feedback-to-Plan Decisions for Self-Evolving LLM Agents in CUDA Kernel Generation

Yee Hin Chong, Jiaming Wu, Youhui Zhang, Peng Qu*

https://github.com/yuxuan-z19/cudanalyst

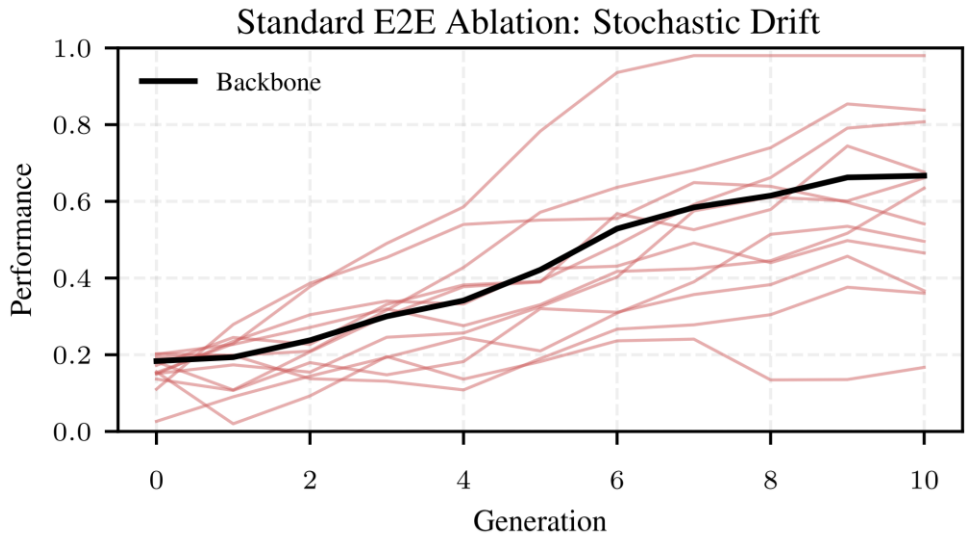
For self-evolving LLM agents, end-to-end ablation conflates analysis with trajectory-dependent drift.

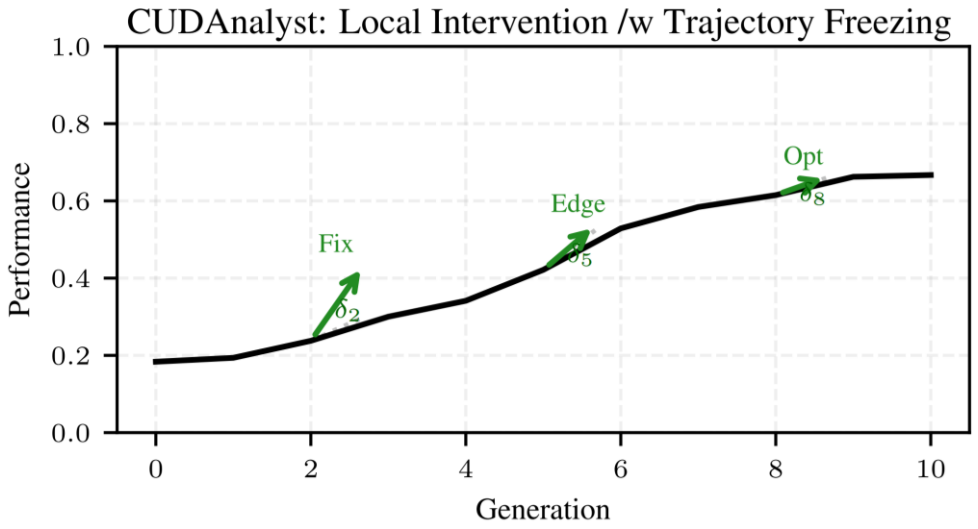
For HPC kernel generation, with heterogeneous diagnostic feedback injecting into the context,

How to isolate feedback utility from stochastic trajectory drift?

Which feedback signals truly drive planning decisions?

How do feedback signals interact at the generation level?

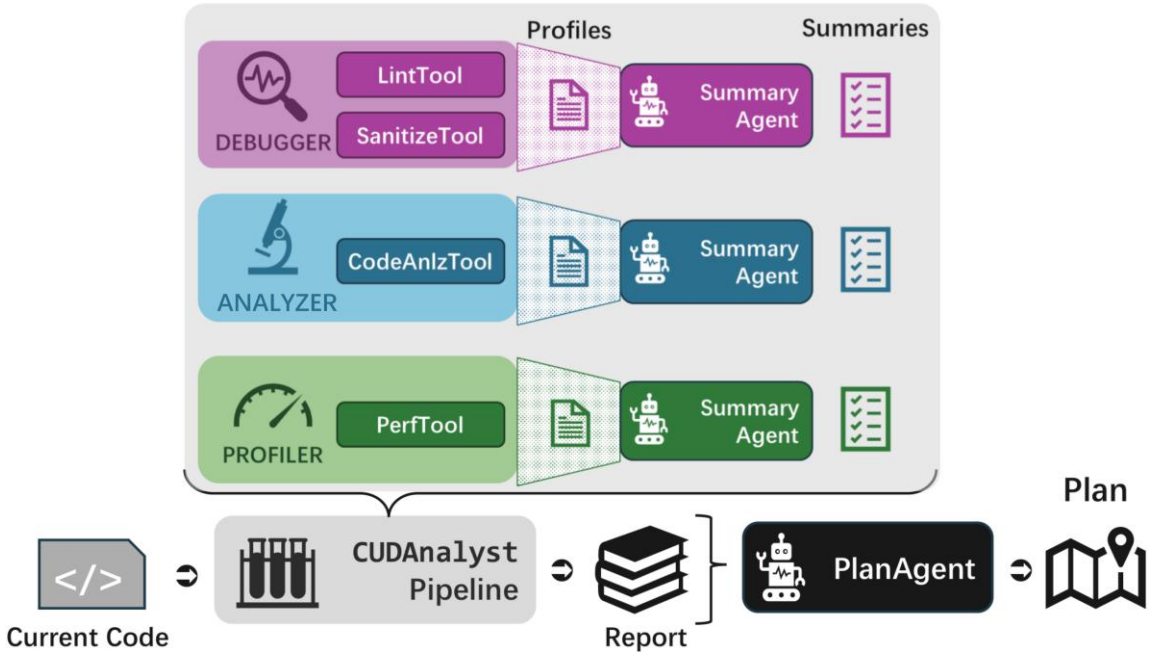




Feedback attribution must be performed at fixed generations to avoid confounding from potential drift.

CUDAnalyst: Unified analysis layer decoupling feedback from planning via frozen-trajectory intervention.

- Comprehensive CUDA analysis modules
- Supports various form of feedback profiles
- Clear evidence-decision boundary



Frozen-Trajectory Intervention

- Freezes program states at specific generation G
- Eliminates trajectory drift; Changes in planning are strictly caused by intervention

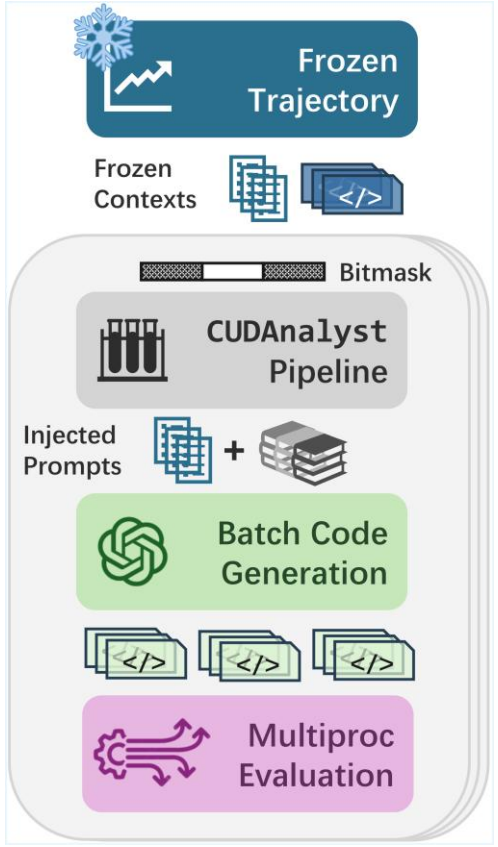
Selective Input Manipulation

- Systematically masks or modifies individual feedback components
- Creates a controlled environment for precise causal attribution

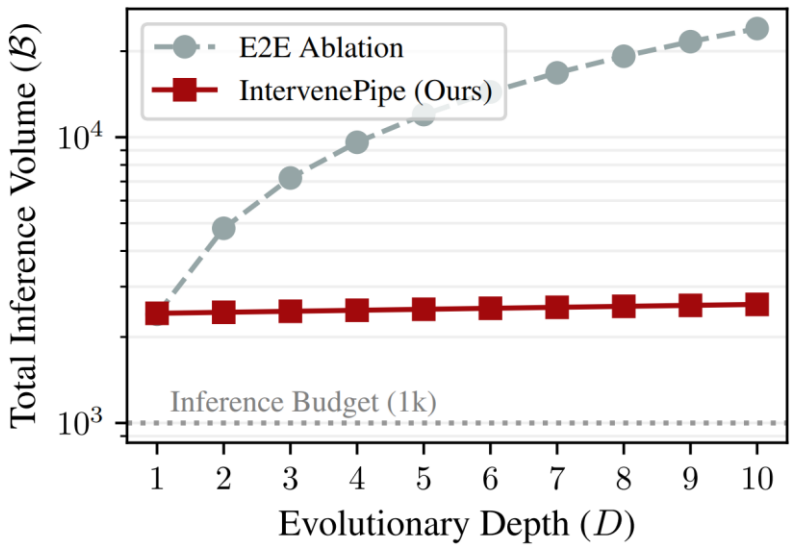
Coalitional-style Attribution

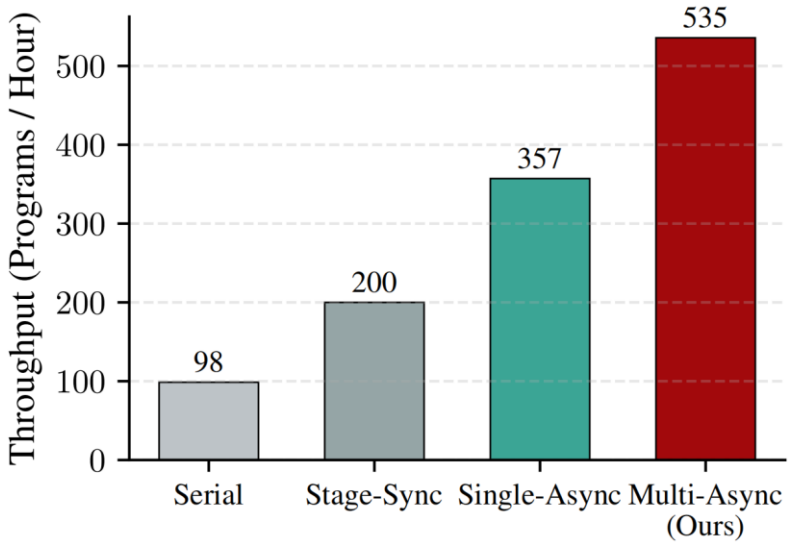
- Calculates marginal contributions and interaction effects via Banzhaf analysis.
- Moves beyond "all-or-nothing" ablation to quantifiable utility of each feedback signal

IntervenePipe: Sample-centric, event-driven execution framework for massive kernel sample evaluation



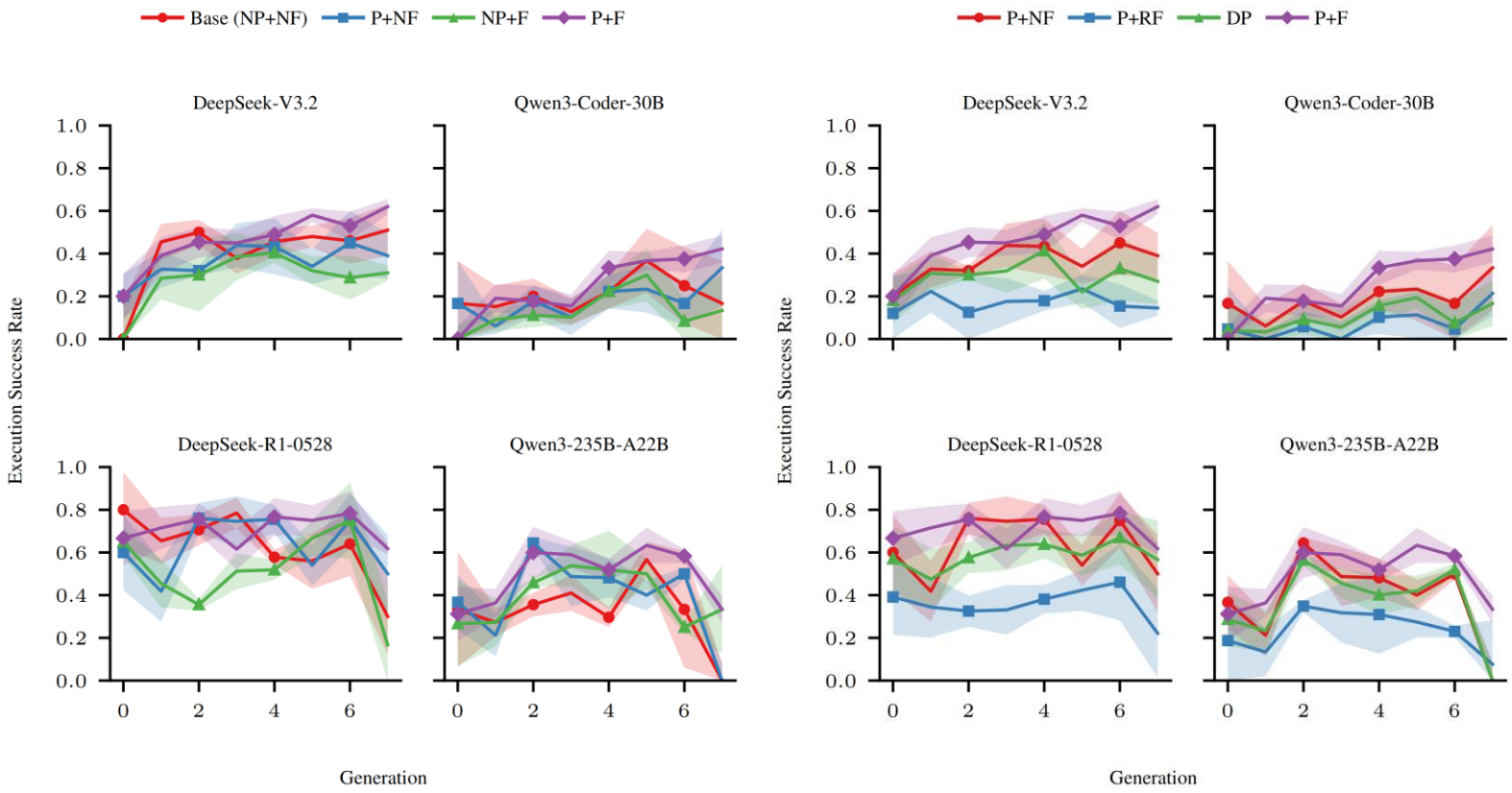
- ✓ Event-driven feedback injection and LLM prompting
- ✓ Fan-out parallel evaluation with consistent state management
- ✓ Online aggregation with event-driven fan-in





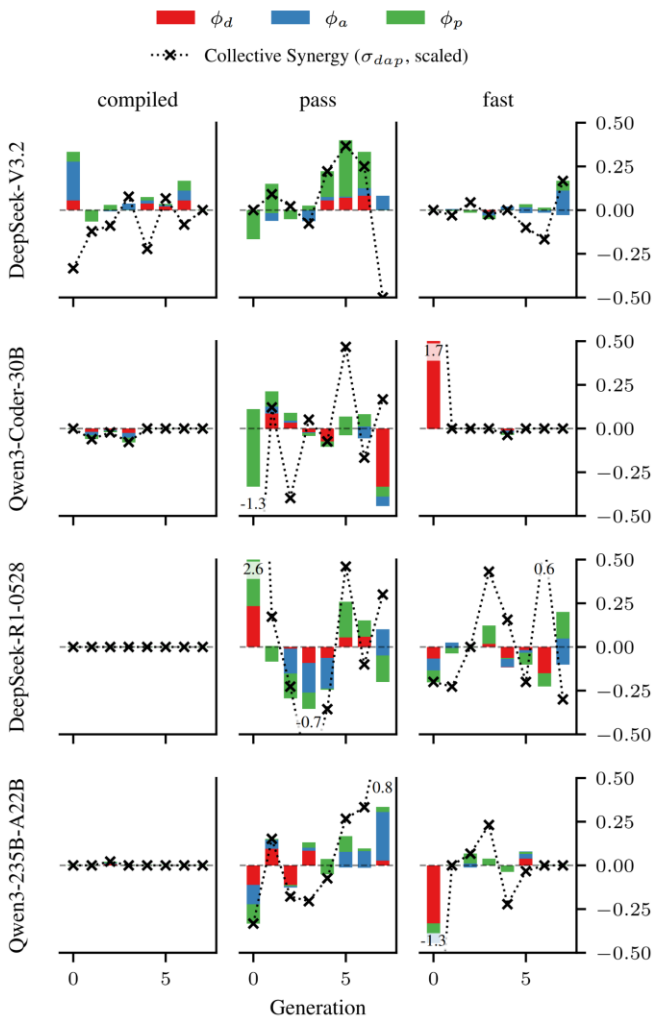
RQ0
Planning

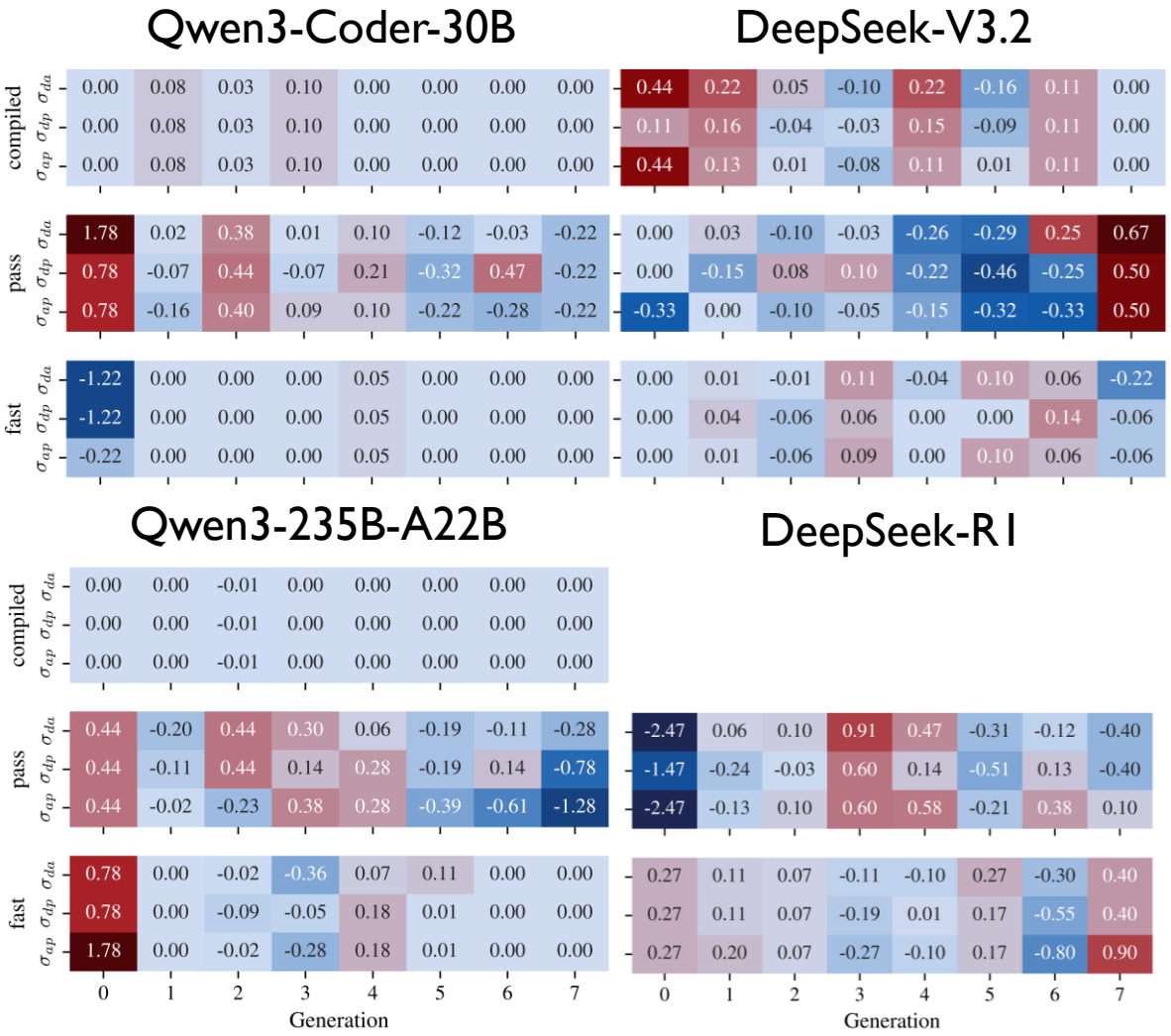
Explicit planning is effective only when grounded in feedback, with feedback-aligned planning yielding stable improvements.



RQ1
Tool Feedback

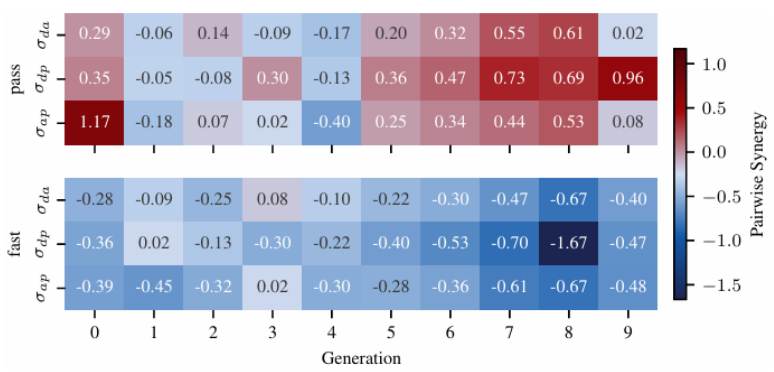
Planning effectiveness arises from interactions among multiple feedback components and joint feedback availability.



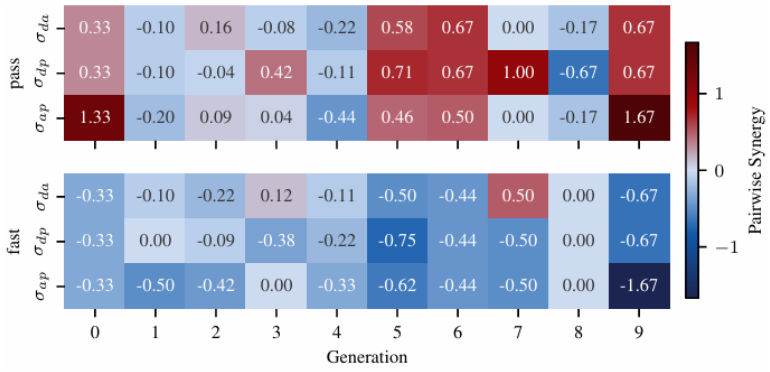


Gen0 Backbone

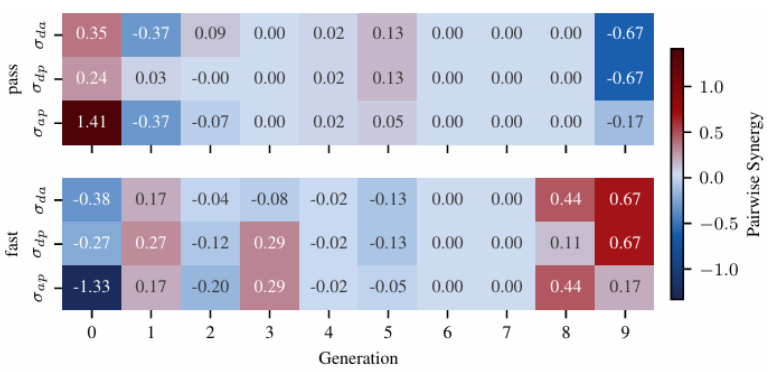
Tool synergy is largely architecture-invariant,
while the nature of synergy evolves with backbone capability.



(a) Kimi-K2 backbone



(b) MiniMax-M2.5 backbone



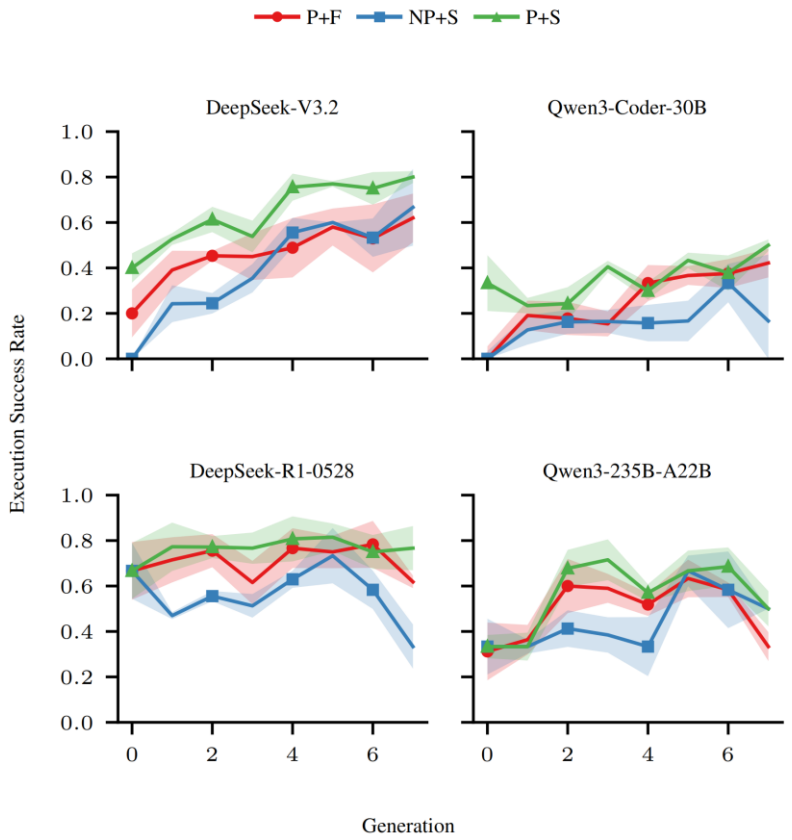
(c) Gemini-2.5-Pro backbone

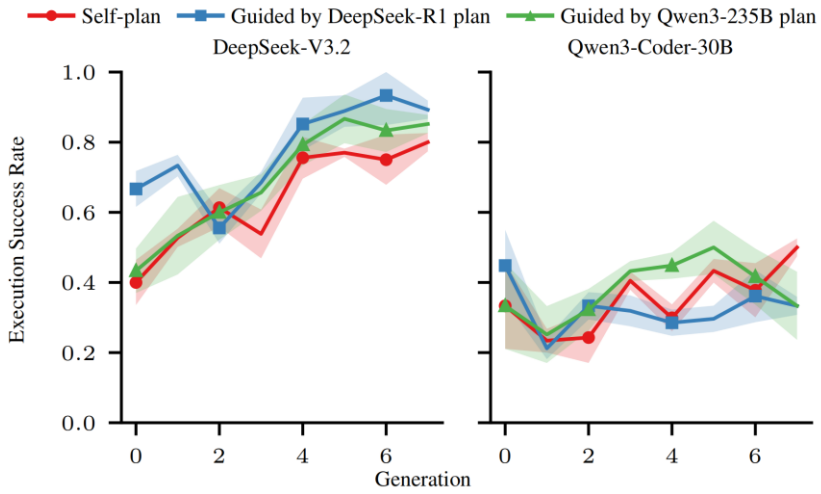
RQ2
Tool Summary

Feedback summarization facilitates but does not replace explicit planning, particularly benefiting weaker models.

RQ3
Distillation

Plans generated by stronger models partially transfer to weaker models within the same model family.

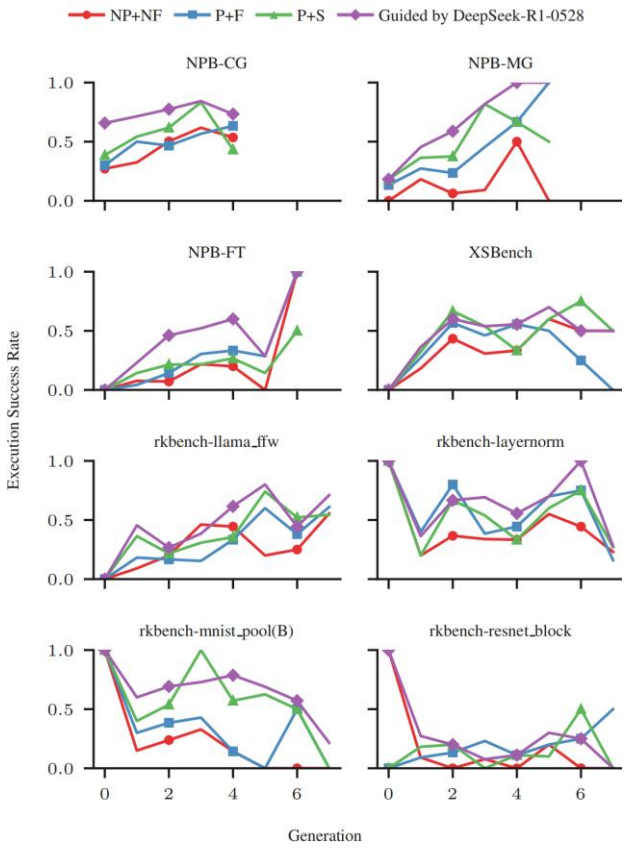


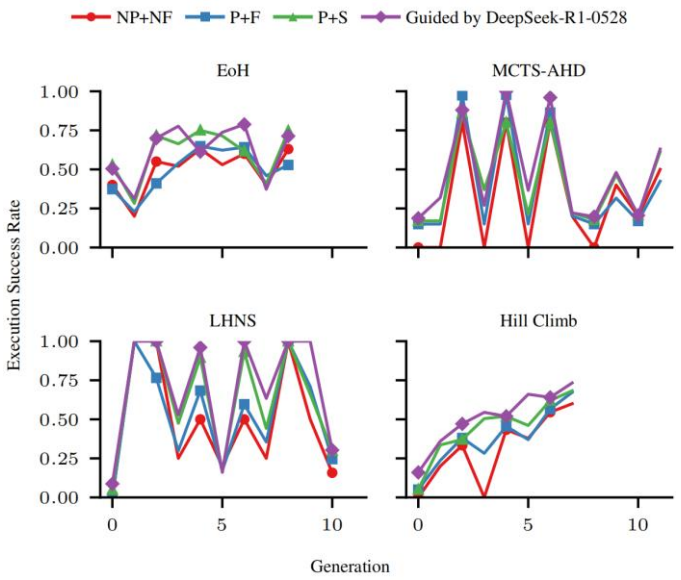


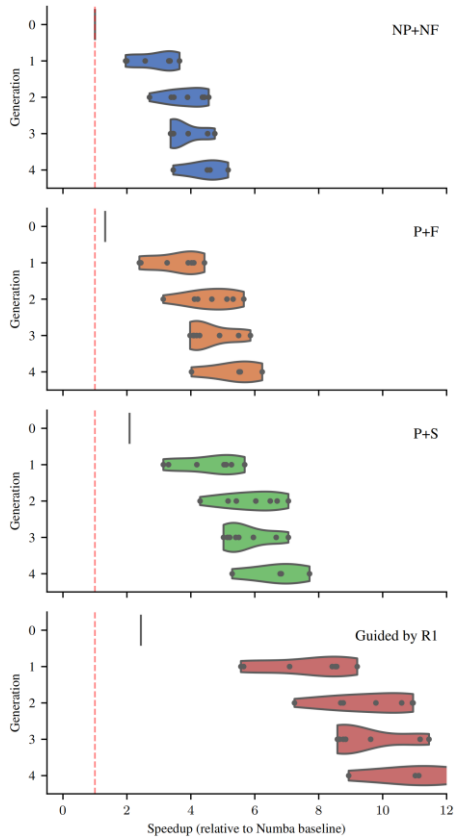
Gen1 Workload Consistent qualitative patterns across workloads are observed, while the magnitude of improvements varies.

Gen2 Regimes Execution dynamics exhibit strong structural invariance across all regimes by different evolutionary operators.

Gen3 Domain Strategic planning mechanism identified holds tool-agnostic and generalizes across CPU-based Numba tasks.







Long-term evolution transitions from an explore-dominant phase to an exploit-dominant phase where code similarity stabilizes.

CuGEdit: A modular plugin that leverages kernel-similarity-aware gating mechanism for feedback selection, guiding DeepSeek-V3.2's kernel generation with DeepSeek-R1's plans.