Benchmarking the Capability and Behavior of AI Agents

Paolo Calafiura[1], Jay Chan[1], Joshua Ho[1,2], Xiangyang Ju[1], Dongwon Kim[1,2], Haichen Wang[1,2], Chengxi Yang[1], Haoran Zhao[1,3]

[1]Lawrence Berkeley National Laboratory [2]UC Berkeley [3]NERSC

CLARIPHY Collaboration Agentic Analysis Meeting

October 27, 2026

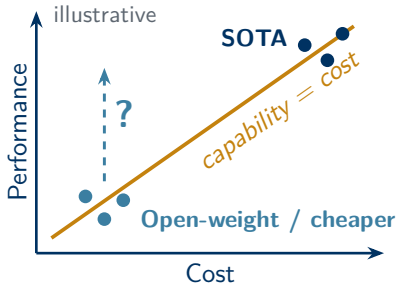
Four reasons to benchmark AI agents on HEP tasks.

1 SOTA model benchmarks draw attention from industry, encouraging the AI/ML community to incorporate HEP programs as part of their evaluations.

2 Lower-end model benchmarks inform the cost–performance trade-off in production systems.

a) Open-weight or cheaper models may be cost-effective for mass production systems with high throughput.

b) To what extent can skills/knowledge bases enhance the capability of lower-end models?



3 Benchmarks produce trajectories for RL.

4 Benchmarks inform the review of research results produced by AI agents.

Premise: bet on general agents and models

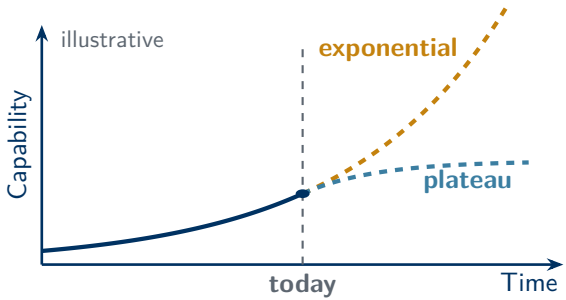
Models get cheaper and more capable every year. Today's SOTA model will be one of the cheaper models we can use next year.

Our premise

- Focus on general agentic frameworks and models, not customized frameworks.
- Safeguards, steering, and routing built into customized wrappers will be internalized into the models.
- We test directly whether agents behave like human scientists, not whether a customized framework can reproduce that behavior.

Caveat

- Based on only two to three years of experience: we could be at the beginning of an exponential, or approaching a plateau.
- In the plateau case, much of what we need will have to be built into customized frameworks.

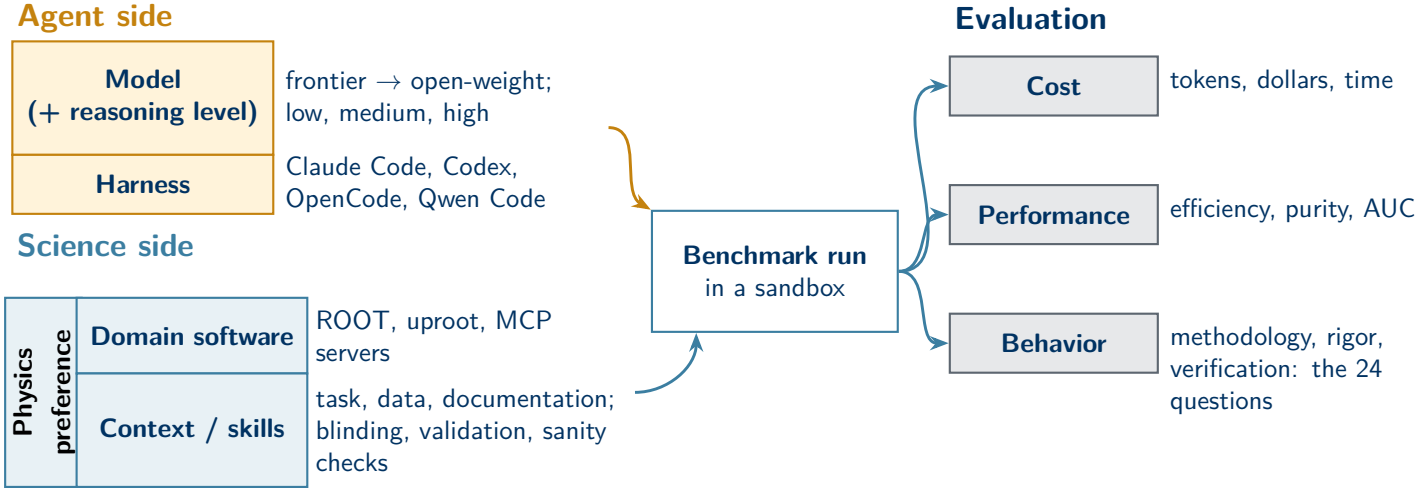


- Price for a fixed capability: ~ 5–$10\times$ cheaper per year (Epoch AI; Gundlach et al. 2025).
- Open GPT-OSS-120B (Aug 2025, one 80 GB GPU) matches o1 (best model, Dec 2024) on GPQA Diamond at $\sim 1\%$ of the price.
- Claude Opus 4 $\rightarrow$ 4.5 (2025): stronger model, $3\times$ cheaper.
- Our benchmark: free GLM-5.3 Flash scores 12–15/19, like Opus 5.5; GPT-OSS-120B scores 0–4.

Next: what do we benchmark?

What do we benchmark?

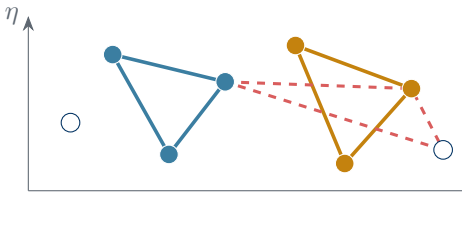
Three sets of dimensions: the agent, the science setup, and the evaluation.



We are carrying out a benchmarking program in which we identify representative computational HEP tasks. We will show one such example.

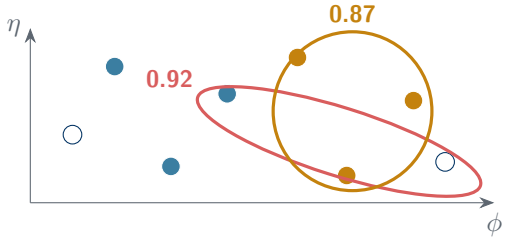
Example physics problem

In collision events where a top quark pair is produced, each top quark can decay into three jets (a triplet). Jets in data are not labeled, so identifying the triplet of jets from the same top quark decay is a combinatorial problem to be solved with physics information.



Combinatorics: many triplets per event; only two are true (solid).

→ *motivates ML classifier training*



Hypothetical scores: the top-scored triplet is fake; the second is true, but they share a jet.

→ *motivates smart use of the scores*

A general solution: use simulated events to train a classifier to separate true triplets from fake triplets, make inference on data, and then decide how to use the classifier scores when more than one triplet is present.

Build true and fake triplets (simulation) → ML classifier training → Inference on data → Select triplets

The higher the reconstruction efficiency, the better.

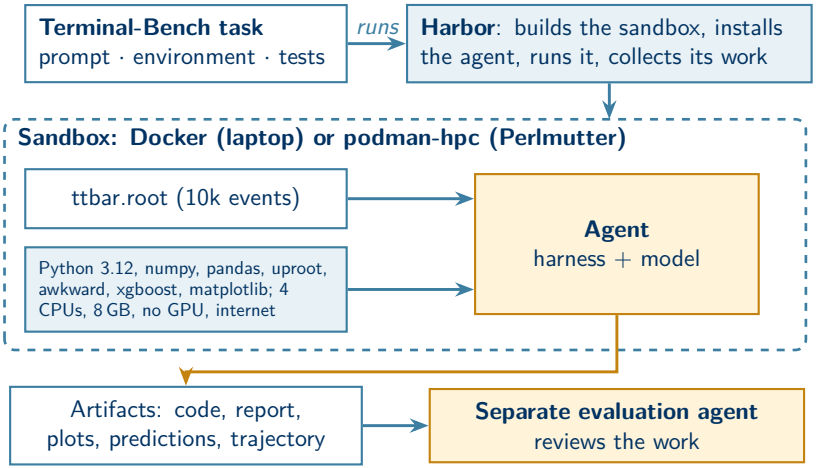
Source: ATLAS HIGG-2020-16; Ho *et al.*, arXiv:2412.10665.

The setup: how an agent is tested

The only input is a 10k-event $t\bar{t}$ Monte Carlo sample; the agent works on its own in a sandbox.

The prompt (non-prescriptive, excerpt)

> You are given /root/data/ttbar.root, a 10k-event Monte Carlo $t\bar{t}$ ROOT ntuple. [branch list]
>
> Your task is to reconstruct hadronically decaying top quarks from the generator-level jets in held-out events.
>
> For each event, your final reconstruction should contain zero, one, or two reconstructed top candidates. … Two selected candidates in the same event may not share any jet.
>
> Develop your own end-to-end reconstruction strategy. …
>
> Optimize the reconstruction efficiency while ensuring that the selected candidates remain physically sensible. …
>
> Document the main approaches you tried, the diagnostics used to compare them, and the reasoning behind the final method.



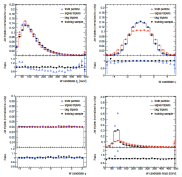
The agent's work is judged by a separate evaluation agent (more on this later).

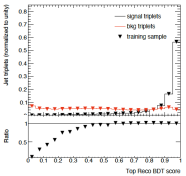
Human and agent: results that look alike

The same top quark reconstruction task, presented by a human physicist and by an AI agent.

Human physicist (PhD thesis)

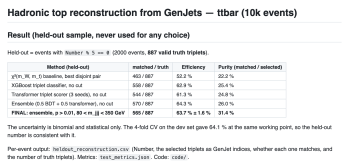
	b candidate	W candidate
Signal	b_1	$q_1 + q_1'$
Signal	b_2	$q_2 + q_2'$
Background	b_1	$q_2 + q_2'$
Background	b_1	$q_1 + q_2'$
Background	b_1	$q_2 + q_1'$
Background	b_1	b_2 + anything
Background	b_2	$q_1 + q_1'$
Background	b_2	$q_1 + q_2'$
Background	b_2	$q_2 + q_1'$
Background	b_2	b_1 + anything
Background	q_1, q_1', q_2 or q_2'	anything
Background	QCD/pileup	anything
Background	anything	QCD/pileup + anything





Training	t-candidate	W-candidate	b-candidate
Nominal	**37%**	**46%**	**57%**
Nominal w/o m_t	35%	45%	57%
Nominal w/o m_W, m_t	35%	43%	57%
Boosted	25%	33%	53%
Boosted w/o m_t	17%	26%	50%
Boosted w/o m_W, m_t	17%	25%	51%
Boosted + p_T^t w/o m_W, m_t	19%	28%	51%

AI agent (Claude Code, Opus 5.5)


Hadronic top reconstruction from GenJets — ttbar (10k events)

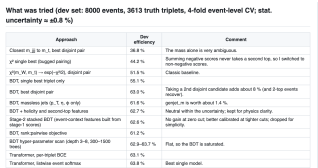
Result (held-out sample, never used for any choice)

Held-out = events with `Number % 5 == 0` (2000 events, **887 valid truth triplets**).

Method (held-out)	matched / truth	Efficiency	Purity (matched / selected)
χ^2(m_W, m_t) baseline, best dijoint pair	463 / 887	52.2 %	22.2 %
XGBoost triplet classifier, no cut	599 / 887	62.9 %	35.4 %
Transformer triplet scorer (3 seeds), no cut	644 / 887	61.3 %	34.8 %
Ensemble (0.5·BDT + 0.5·transformer), no cut	570 / 887	64.3 %	29.0 %
FINAL: ensemble, p > 0.01, 80 < m_jjj < 350 GeV	**565 / 887**	**63.7 % ± 1.6 %**	**31.4 %**

The uncertainty is binomial and statistical only. The 4-fold CV on the dev set gave 64.1 % at the same working point, so the held-out number is consistent with it.

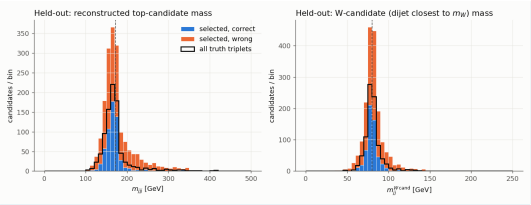
Per-event output: `heldout_reconstruction.csv` (Number, the selected triplets as GenJet indices, whether each one matches, and the number of truth triplets). Metrics: `test_metrics.json`. Code: `code/`.


What was tried (dev set: 8000 events, 3613 truth triplets, 4-fold event-level CV; stat. uncertainty ≈ ±0.8 %)

Approach	Dev efficiency	Comment
Closest m_{jjj} to m_t, best dijoint pair	38.8 %	The mass alone is very ambiguous.
χ^2 single best (bugged pairing)	44.2 %	Summing negative scores never takes a second top, so I switched to non-negative scores.
$\chi^2(m_t, m_W)$ → exp(−χ²/2), dijoint pair	51.5 %	Classic baseline.
BDT, single best triplet only	55.1 %	
BDT, best dijoint pair	63.0 %	Taking a 2nd dijoint candidate adds about 8 % (and 2-top events matches).
BDT, massless jets (p, τ, φ only)	61.8 %	gentjet_m is worth about 1.4 %.
BDT + helicity and second-top features	62.7 %	Neutral within the uncertainty; kept for physics clarity.
Stage-2 stacked BDT (event-context features built from stage-1 scores)	62.6 %	No gain at zero cut; better calibrated at tighter cuts; dropped for simplicity.
BDT, soft pseudo-log objective	61.2 %	
BDT hyper-parameter scan (depth 3–6, 300–1500 trees)	62.8–63.7 %	Flat, so the BDT is saturated.
Transformer, per-triplet BCE	63.1 %	
Transformer, listwise event softmax	63.8 %	Best single model.
BDT + transformer (listwise), 55/50	**64.8 %**	Best overall (weights 0.3/0.5/0.7 gave 64.4/64.6/64.1 %).
Ablation: BDT + gentjet_ktag	70.0 %	Not used: truth-flavour, excluded (see above).

Ranking diagnostic for the BDT: the truth triplet is ranked 1st in 55 % of cases, in the top 2 in 73 %, top 3 in 81 %, top 5 in 88 % and top 10 in 96 %. The loss is dominated by b-versus-light and ISR/FSR-jet ambiguity, which kinematics alone can't resolve. The BDT, the transformer and the stacked model all saturate at about 63–65 %, which points to an information limit rather than a model limit.

36 min, $1.96, non-prescriptive instructions



Source: Left: J. E. Dickinson, PhD thesis, UC Berkeley (2020), repository.cern/records/srrjy-07s16. Right: Claude Code with Opus 5.5, run 20260928T155559Z-…-claude-code-70311 (Harbor-recorded).

All runs: score, physics performance, and cost

#	CLI + LLM + effort	Served via	Score /19	Efficiency	Purity	Cost (USD)	Agent time (min)
1	Codex + GPT-6 Astra + medium	OpenAI	16	0.660	0.324	2.49	9.6
2	OpenCode + Claude Opus 5.5 + high	CBORG	15	0.644	0.296	n/a	12.7
3	Qwen Code + Claude Opus 5.5 + high	CBORG	15	0.637	0.306	n/a	11.0
4	OpenCode + GLM-5.3 Flash + default	CBORG	15	0.678	0.307	0 (free)	36.6
5	Codex + GPT-6 Astra + high	OpenAI	15	0.619	0.276	2.84	12.8
6	Claude Code + Claude Opus 5 + medium	Anthropic	14	0.614	0.248	5.02	37.9
7	Qwen Code + Claude Opus 5.5 + medium	CBORG	14	0.630	0.278	n/a	7.5
8	Claude Code + Claude Opus 5.5 + high	Anthropic	13	0.637	0.314	1.96	36.3
9	OpenCode + Claude Opus 5.5 + medium	CBORG	13	0.782	0.444	n/a	5.5
10	Claude Code + Claude Opus 5.5 + high	CBORG	13	0.644	0.295	3.19	52.7
11	Codex + GPT-6 Sol + high	OpenAI	13	0.608	0.318	0.38	6.6
12	Qwen Code + GLM-5.3 Flash + default	CBORG	13	0.634	0.282	0 (free)	23.2
13	Codex + GLM-5.3 Flash + high	CBORG	12	0.710	0.290	0 (free)	16.7
14	Codex + GPT-6 Luna + high	OpenAI	12	0.739	0.405	0.019	8.6
15	Codex + GPT-6 Sol + medium	OpenAI	12	0.609	0.261	0.39	6.8
16	Codex + GPT-6 Luna + medium	OpenAI	11	0.732	0.514	0.012	8.4
17	Codex + Gemma 4 31B Thinking + high	CBORG	6	0.489	–	0 (free)	6.2
18	OpenCode + Gemma 4 31B Thinking + default	CBORG	5	0.576	–	0 (free)	14.7
19	Claude Code + Gemma 4 31B Thinking + default	CBORG	4	0.555	–	0 (free)	4.1
20	Claude Code + GPT-OSS 120B + high	CBORG	4	0.231	–	0 (free)	8.2
21	Qwen Code + GPT-OSS 120B + high	CBORG	3	0.398	–	0 (free)	2.2
22	Codex + GPT-OSS 120B + high	CBORG	0	–	–	0 (free)	0.5
23	OpenCode + GPT-OSS 120B + high	CBORG	0	–	–	0 (free)	0.7
–	Oracle (reference script)	no model	9	0.623	0.258	no model	0.2

The presentation looks human-like. Is the work behind it?

Source: HQ review files (frontier-review.json and review.json). Same order and labels as the behavioral-ranking bar chart.

Evaluating the agent's top quark reconstruction

Like physicists reviewing a colleague's work, we ask 24 questions about each run.

Q1 How long did the run take (trial wall-clock and agent execution time)?
Q2 What is the cost of the agent run in USD?
Q3 Did the agent perform BDT classification for top decay triplet candidates?
Q4 How many BDT features were engineered and used in the model?
Q5 Did the agent optimize or select a refined feature list?
Q6 Did the agent implement a proper train / validation / test partition?
Q7 Did the agent explore ML beyond XGBoost binary classification (e.g. ranking losses)?
Q8 Which candidate selection algorithm picked the final top candidates per event?
Q9 Did the agent compare multiple candidate selection algorithms or thresholds?
Q10 How many selection algorithms or score thresholds were evaluated?
Q11 Was the constraint $N_{\text{top}} \leq 2$ candidates per event strictly imposed?
Q12 What is the achieved BDT classifier ROC AUC?
Q13 Did the agent evaluate or avoid mass sculpting in candidate selection?
Q14 Was top quark invariant mass used in candidate selection or feature engineering?
Q15 How many optimization iterations or pipeline tuning steps were performed?
Q16 Was the BDT score distribution plotted or saved as a diagnostic?
Q17 Were mass distribution comparison plots made for top candidate invariant masses?
Q18 What is the overall top candidate selection efficiency?
Q19 What is the efficiency with only the first 2 non-overlapping candidates per event?
Q20 Did the selection produce any overlapping (shared-jet) triplets?
Q21 Jet multiplicity: efficiency and purity vs. jet multiplicity, separating no-truth-triplet events from missed ones?
Q22 Efficiency–purity: purity reported, and gains checked to come from better assignments, not more candidates?
Q23 Significance: uncertainties, and gains over a simple baseline on the same events, with event-level correlations?
Q24 False candidates: rate of selections in no-truth-triplet events, with their scores and masses inspected?

The answers come from the run's code, logs, reports, and plots.

Evaluating the agent's top quark reconstruction

We evaluate agents on performance, and on behavior and alignment with scientists.

Evaluating the agent's research work

Performance (unscored quantities)

D. Reconstruction performance

Q12 ROC AUC
Q18 efficiency
Q19 eff., 2 disjoint

E. Runtime and cost

Q1 time
Q2 cost

Behavior and alignment with scientists

A. Exploration and method development /9

Q3 BDT classification
Q4 features
Q5 feature selection
Q7 other algorithms
Q8 selection algorithm
Q9 compare selections
Q10 thresholds explored
Q14 top mass used
Q15 tuning iterations

B. Robustness and scientific rigor /4

Q6 train/val/test split
Q11 $N_{\text{top}} \leq 2$
Q13 mass sculpting
Q20 no shared jets

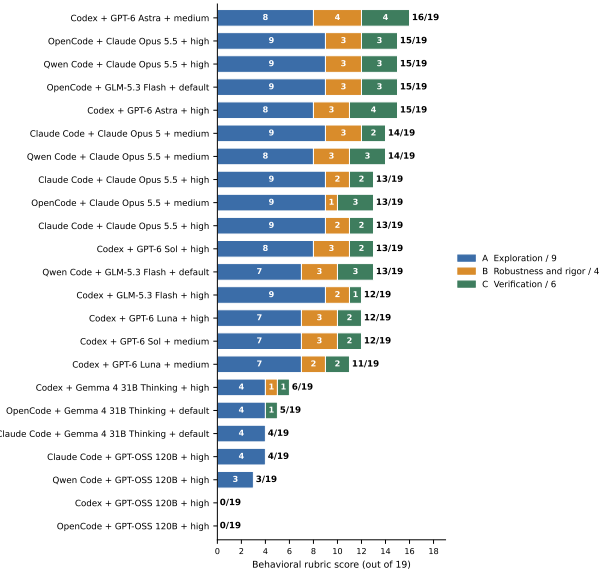
C. Verification /6

Q16 score plot
Q17 mass plots
Q21 jet multiplicity
Q22 efficiency–purity
Q23 significance
Q24 false candidates

A–C: Y = 1 or N = 0, no partial credit; every N is marked *not performed* or *insufficient evidence*. Total /19; D–E and Harbor reward are reported, not scored.

These dimensions are how we evaluate the work of agents.

Behavioral ranking

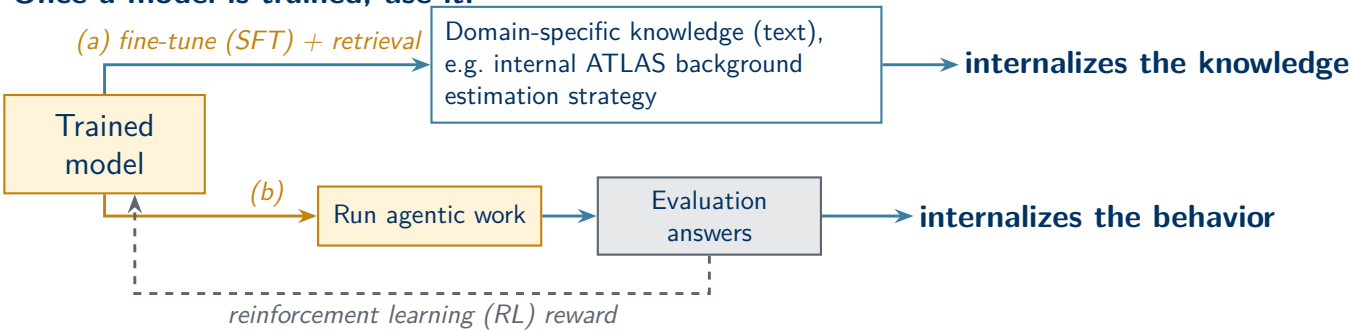


	Proprietary & SOTA LLM	**Open weights**
Frontier lab CLI Codex, Claude Code	Codex + GPT-6 Astra, Sol, Luna; Claude Code + Claude Opus 5, 5.5 scores 11–16/19	Codex + GLM-5.3 Flash, Gemma 4 31B, GPT-OSS 120B; Claude Code + Gemma 4 31B, GPT-OSS 120B scores 0–12/19
Free, open-source CLI OpenCode, Qwen Code	OpenCode, Qwen Code + Claude Opus 5.5 scores 13–15/19	OpenCode + GLM-5.3 Flash, Gemma 4 31B, GPT-OSS 120B; Qwen Code + GLM-5.3 Flash, GPT-OSS 120B scores 0–15/19

Caveats: just a few trajectories, so no statistical conclusion; the plot illustrates the method of capturing behavior.

These evaluation answers are good rewards for reinforcement learning.

Once a model is trained, use it:



RL and SFT are increasingly accessible for non-CS domains, as compute and intelligence become abundant.

The case for small customized models:

Sovereignty over data, knowledge, and access

Economic at high token volume

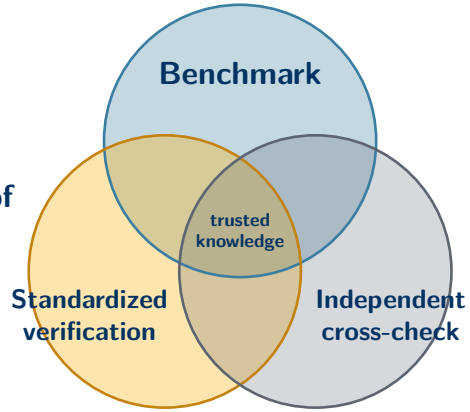
Agility: respond promptly to experiment and organization changes

A future framework of establishing knowledge produced by agentic AI should have at least three components.

1. **Benchmark**
   Basic understanding of AI capabilities and behavior. Establishes basic trust and confidence in agentic results, so we don't have to manually check every piece of results.

2. **Standardized verification, embedded into the production of science**
   Like ISO standards governing quality control in manufacturing: results produced by agents come with immediate verification built into the agentic system.

3. **Independent cross-check**
   Major science results must be independently cross-checked, as has always happened (nine loops, follow-up experiments, etc.).



Benchmarks are the first step: they build the trust agentic science needs.

Where our HEP benchmarking program is heading.

1. We are working on a benchmark that tests a broad range of agents (CLI harness + LLM + reasoning level).

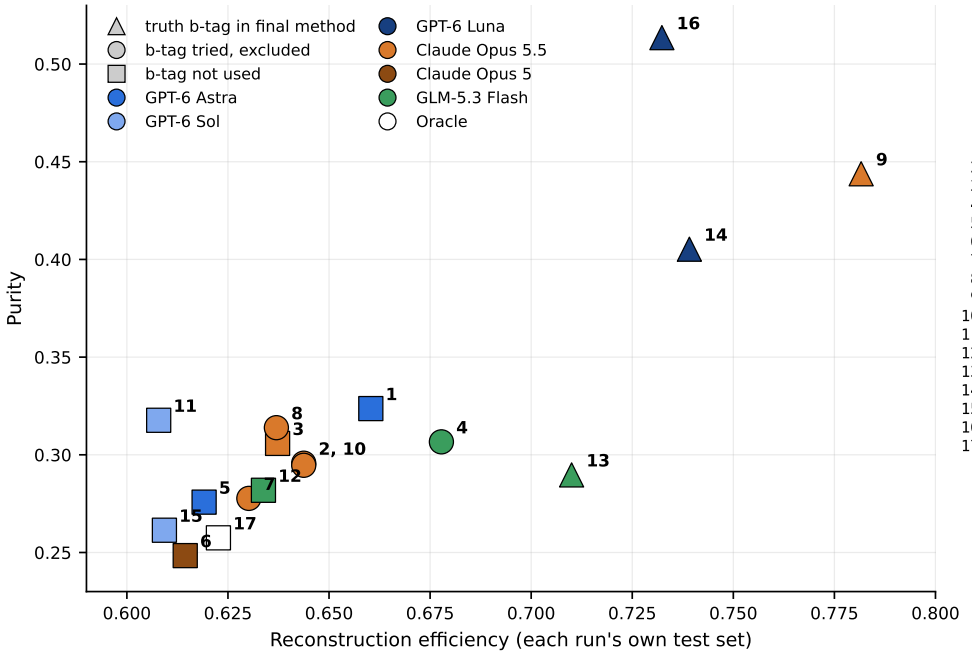
2. The upcoming benchmark will inform us about the performance and cost of those agents in the context of HEP.

3. These tests produce trajectories and rewards for reinforcement learning, which we are using to enhance the performance of agents powered by small open-weight models.

Thank you.

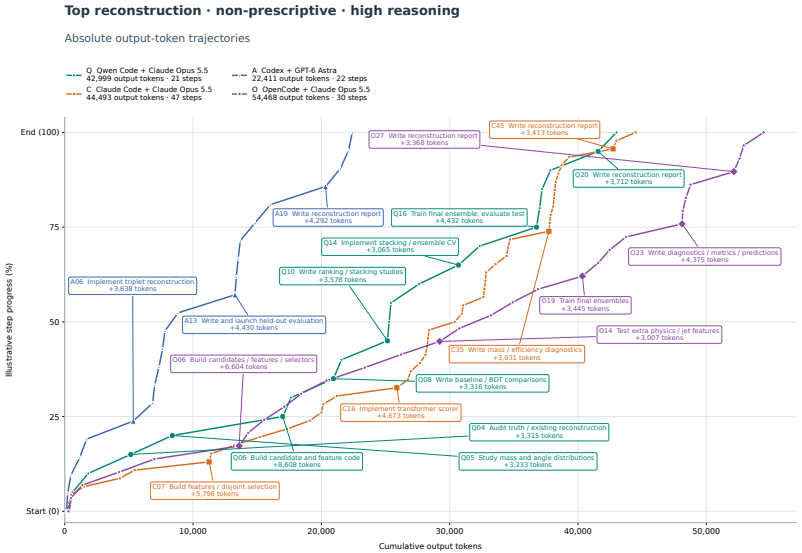
Backup

Efficiency vs purity



1 Codex + GPT-6 Astra + medium (16/19)
2 OpenCode + Claude Opus 5.5 + high (15/19)
3 Qwen Code + Claude Opus 5.5 + high (15/19)
4 OpenCode + GLM-5.3 Flash + default (15/19)
5 Codex + GPT-6 Astra + high (15/19)
6 Claude Code + Claude Opus 5 + medium (14/19)
7 Qwen Code + Claude Opus 5.5 + medium (14/19)
8 Claude Code + Claude Opus 5.5 + high (13/19)
9 OpenCode + Claude Opus 5.5 + medium (13/19)
10 Claude Code + Claude Opus 5.5 + high (13/19)
11 Codex + GPT-6 Sol + high (13/19)
12 Qwen Code + GLM-5.3 Flash + default (13/19)
13 Codex + GLM-5.3 Flash + high (12/19)
14 Codex + GPT-6 Luna + high (12/19)
15 Codex + GPT-6 Sol + medium (12/19)
16 Codex + GPT-6 Luna + medium (11/19)
17 Oracle (reference script) (9/19)

Where do the output tokens go?



- Most output tokens go to writing code and reports.
- Claude Code, Qwen Code, and OpenCode generate about twice as many output tokens as Codex.
- Large single steps show where each harness appears to make more effort.