WATERFALL

Workflow for Adaptive Training with Evolutionary Reward Formulation and Automated Learning Loops

DIRECTED BY **Eleftherios Triantafyllidis**[1] Filippos Christianos[2] Zhibin Li[3] Bernd Bickel[1]

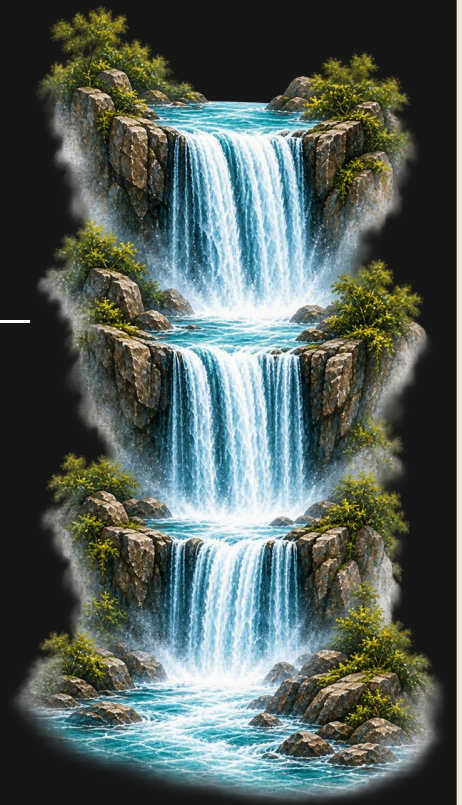
STUDIOS [1]ETH Zurich [2]Independent Researcher, London [3]UCL

PREMIERE **NeurIPS 2026**, Main Track · poster

RUNTIME 5 minutes · seven settings · one rule: observability parity

CONTACT eleftherios.triantafyllidis@gmail.com

CODE github.com/etriantafyllidis/waterfall



OBSERVABILITY · WHAT THE AGENT CAN SEE

Fully vs partially observable worlds

Fully observable **Full Information Readily Available**



Door open: everything in view, the goal included.

Partially observable **Partial Information**



A keyhole: a fragment; the goal is out of sight.

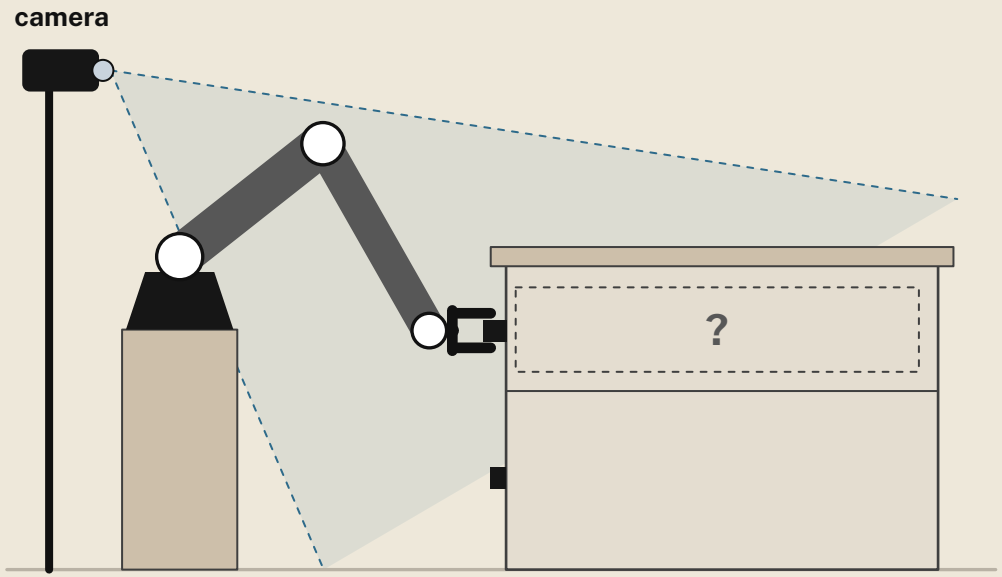
IN REWARD DESIGN Most reward designers are handed the open door: the goal, full state, even true success.

NOTES: FOMDP / POMDP: fully / partially observable Markov decision process (Kaelbling, Littman & Cassandra, 1998)

OBSERVABILITY · VISUAL ROBOTICS EXAMPLE

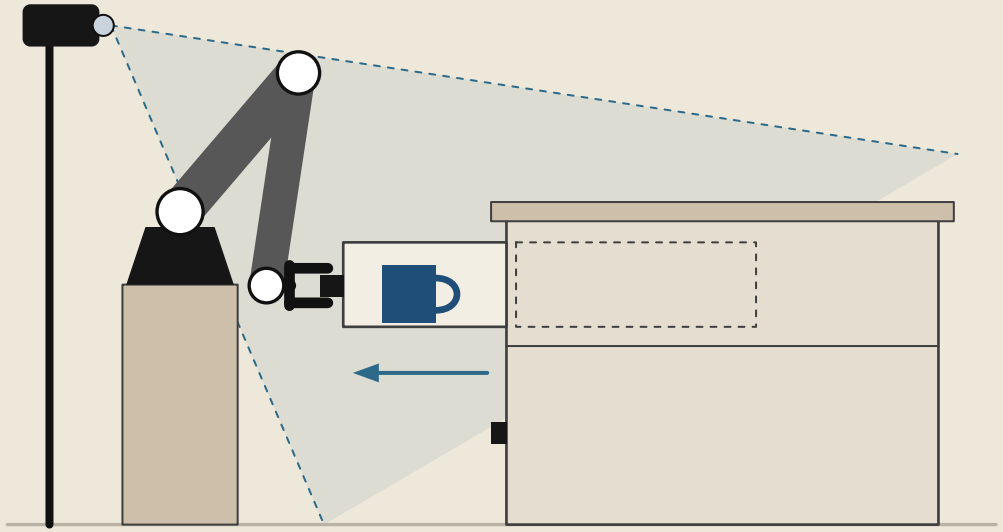
Robotics: Some of the world is only seen by acting

Before PARTIALLY OBSERVABLE



Drawer closed: what is inside is hidden.

After REVEALED BY ACTING

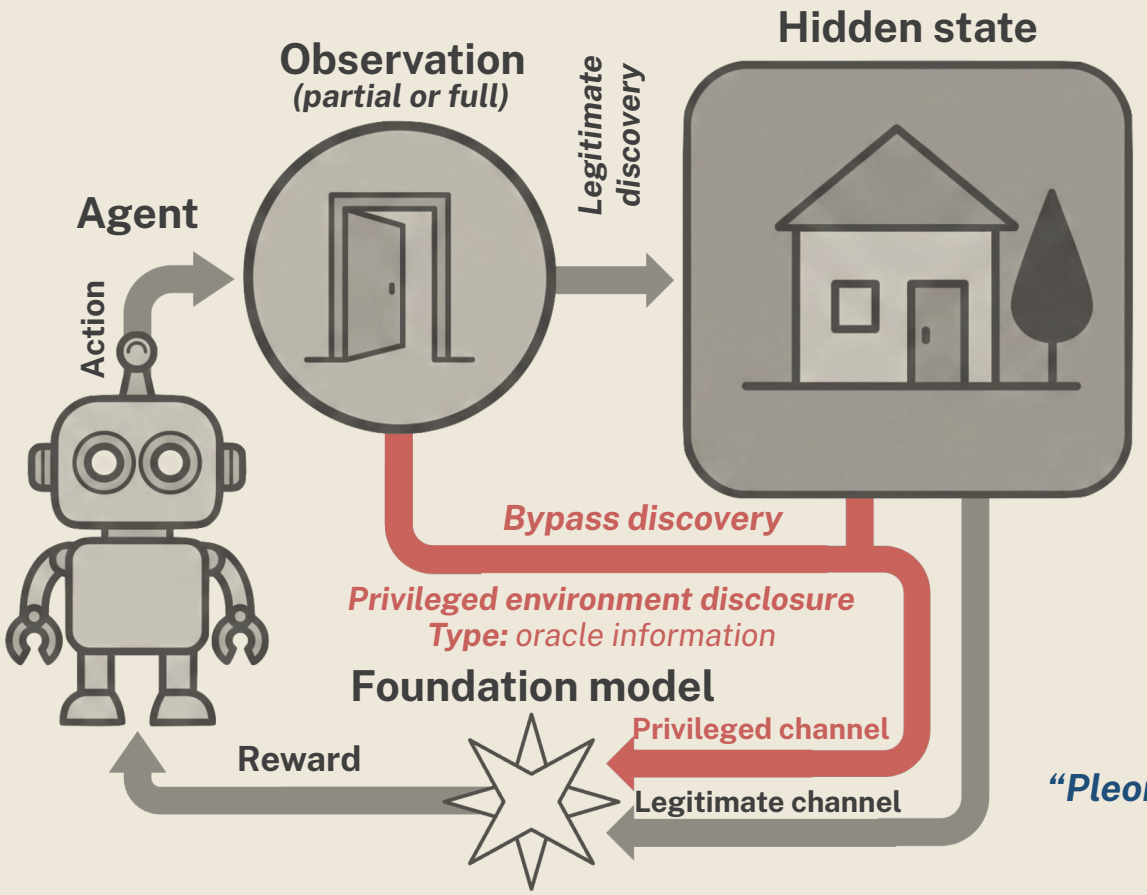


Open it, and the camera sees inside.

NOTES: POMDP: hidden state is revealed only through the agent's own actions and observations (Kaelbling et al., 1998)

PLEONECTIC INFORMATION

Genuine discovery vs privileged information



LEGITIMATE DISCOVERY

The agent acts and observes; only then does it know what lies behind the door.

PLEONECTIC INFORMATION

A privileged channel bypasses the discovery and hands in data that are unknown. Easy in simulation, impossible in real use.

WHY IT MATTERS

Results look better in training than in deployment, where only the agent's own observations remain.

***"Pleonexia"**: (Ancient Greek: πλεονεξία) For wanting more than one's share.*

NOTES: Legitimate Discovery vs Pleonectic Information

OUR APPROACH

Foundation models as guides, not oracles for AI

Principle I: Guided Learning
Trial and error, with a guide

STEP 1
The AI agent acts

Principle II: Learning from Partial Data
Adapt to what can be seen

STEP 2
It sees only part of the world

Principle III: Balanced Learning
Guide, do not control

STEP 3
New data is revealed; the agent must re-adapt

Repeated millions of times, the agent does more of whatever scores well.

ORACLE · TAKEN AT ITS WORD

Whatever it outputs is used as given and at "face value". The ***key*** is to ***guide*** instead of control.

GUIDE · TESTED BY LEARNING

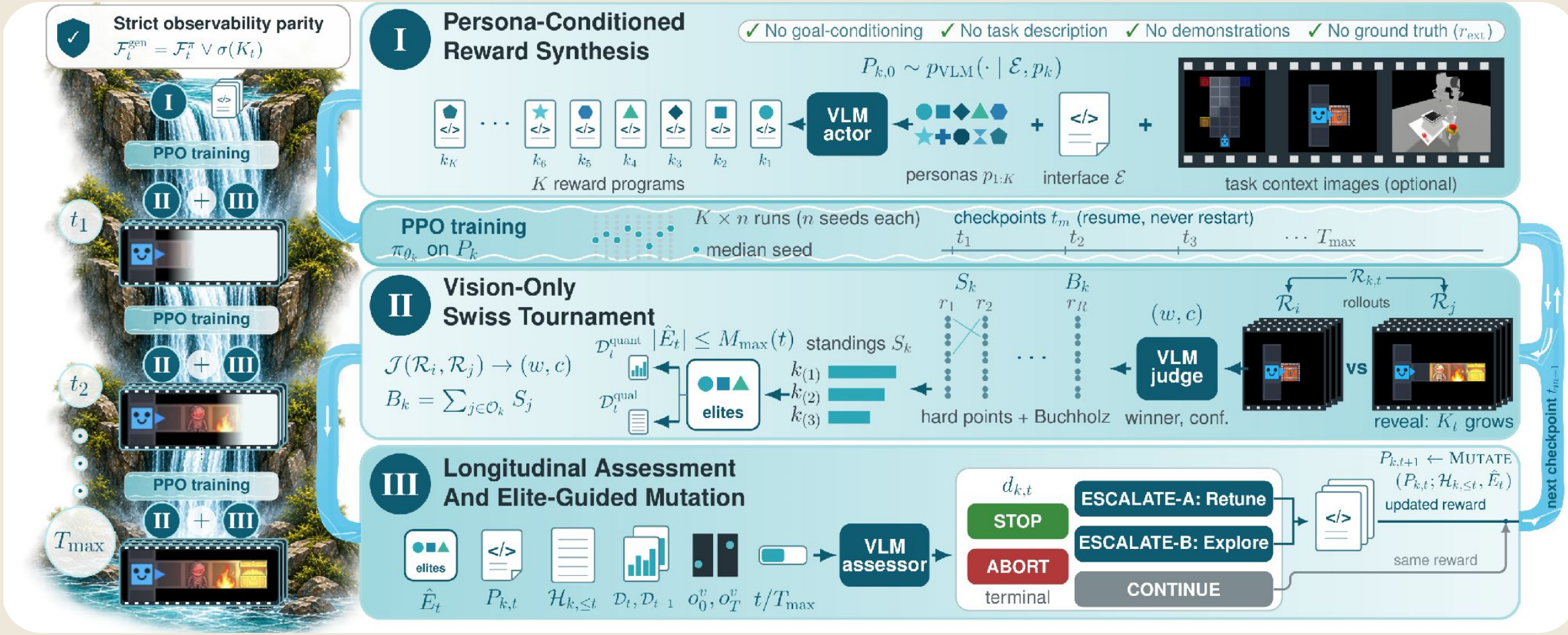
It proposes rewards and judgements; learning tests them against what the agent can see.

THE PAPER · NEURIPS 2026, MAIN TRACK · TRIANTAFYLLIDIS, CHRISTIANOS, LI & BICKEL

WATERFALL

Workflow for **A**daptive **T**raining with **E**volutionary **R**eward **F**ormulation and **A**utomated **L**earning **L**oops

NEURAL INFORMATION PROCESSING SYSTEMS



THE METHOD · HOW WATERFALL WORKS

WATERFALL in four, end-to-end steps, each with its rule

STRICT OBSERVABILITY PARITY

The reward-writer may use only the agent's history plus what acting has revealed.

$$F_t^{\text{gen}} = F_t^{\pi} \vee \sigma(K_t)$$

STAGE I · WRITE

Write K rewards

A vision-language model writes ***K*** reward programs, one per persona.

$$P_{k,0} \sim p_{\text{VLM}}(\cdot \mid E, p_k)$$

TRAIN

Train one agent each

Each reward trains its own agent with PPO, checkpoint by checkpoint.

$$\pi_{\theta k}(a_t \mid F_t^{\pi})$$

STAGE II · JUDGE

Judge the footage

A vision-only judge compares rollouts in pairs; a Swiss tournament ranks them.

$$J(R_i, R_j) \rightarrow (w, c)$$

STAGE III · ASSESS

Assess and evolve

An assessor reads each history: stop, abort, continue, or rewrite.

$$P_{k,t+1} \leftarrow \text{MUTATE}(P_{k,t}; H_{k,\le t}, \hat{E}_t)$$

repeat at each checkpoint $t_1, t_2, \ldots, T_{max}$, until the assessor calls STOP

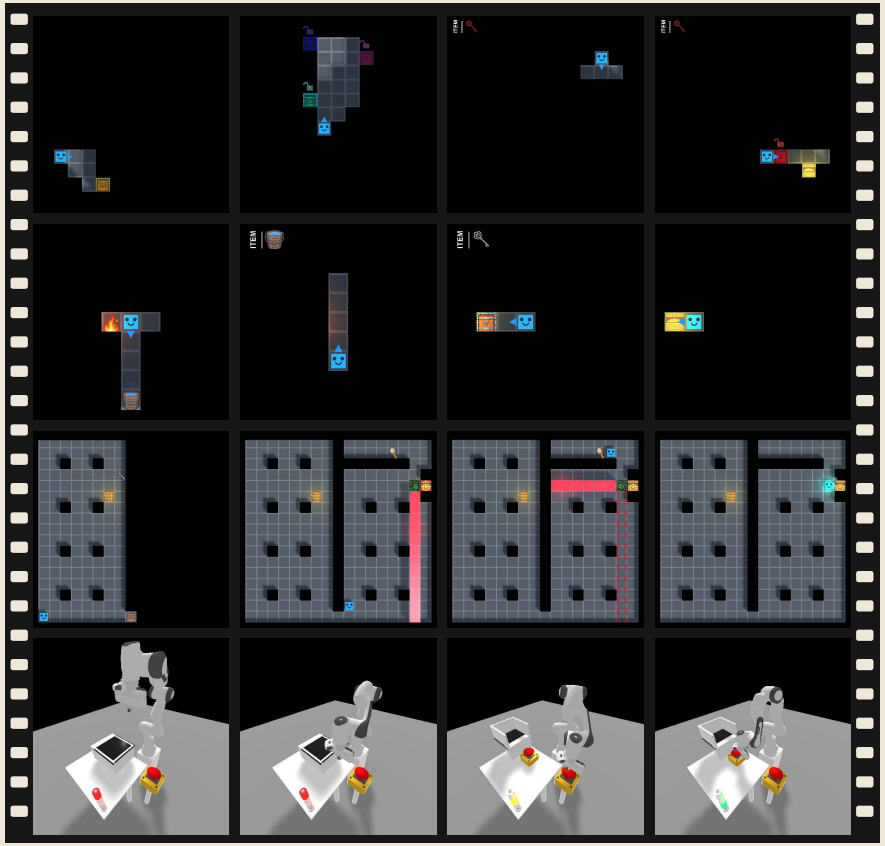
The model proposes; learning and the judge decide.

NOTES: Triantafyllidis, Christianos, Li & Bickel, WATERFALL, NeurIPS 2026 · notation as in the paper

THE TEST ENVIRONMENTS · PARTIALLY OBSERVABLE

What the agent sees: four environments, revealed only by acting

start → end · success



Env I
Locked rooms

Env II
Dark corridor

Env III
Medusa's Revenge

Env IV
Robot arm (Franka)

WHAT MUST BE DISCOVERED

key → door → chest
which door the key opens, and where the chest is

tools → blockers → goal
which tool clears which blocker

key → door → artefact → guardian → goal
a second chamber, behind a locked door

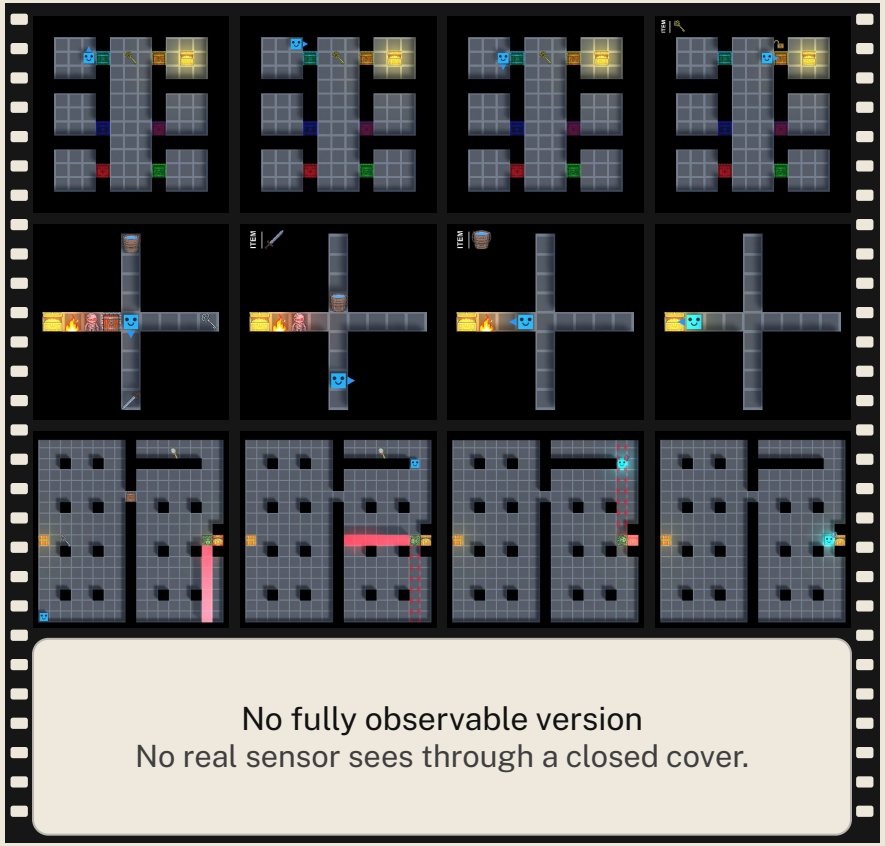
button → cover → hidden button
a second button, under a closed cover

NOTES: Frames: representative successful rollouts from the paper's appendix (episode 1 of 3 per environment)

THE TEST ENVIRONMENTS · FULLY OBSERVABLE

The same worlds, fully visible from the start

start → end · success



Env I
Locked rooms

Env II
Dark corridor

Env III
Medusa's Revenge

Env IV
Robot arm (Franka)

WHAT IS VISIBLE FROM THE START

key → door → chest
every room, key and door

tools → blockers → goal
every tool and every blocker

key → door → artefact → guardian → goal
both chambers, the guardian and the goal

button → cover → hidden button
tested only with hidden information

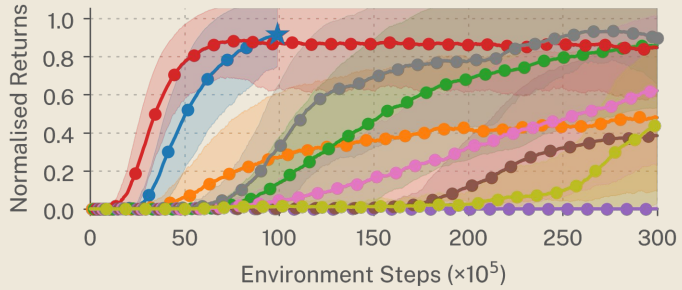
NOTES: Frames: representative successful rollouts from the paper's appendix (episode 1 of 3 per environment)

EVIDENCE · PARTIALLY OBSERVABLE

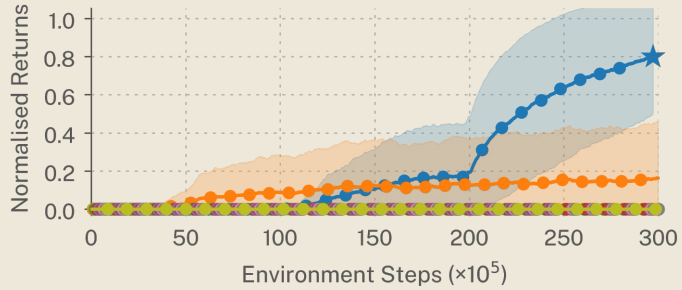
Higher returns, with less information (partially observable)

Normalised task return, mean of 10 training runs (5 in Env IV) · the star marks where WATERFALL's judge called STOP

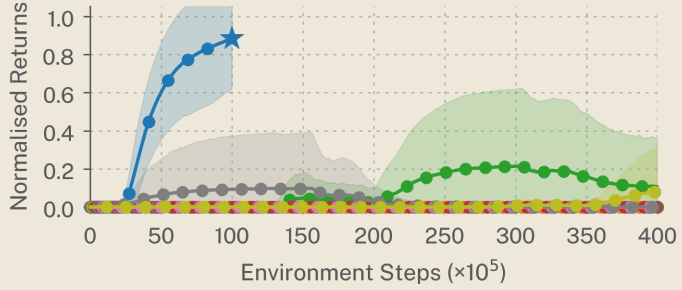
Env. I, partially observable



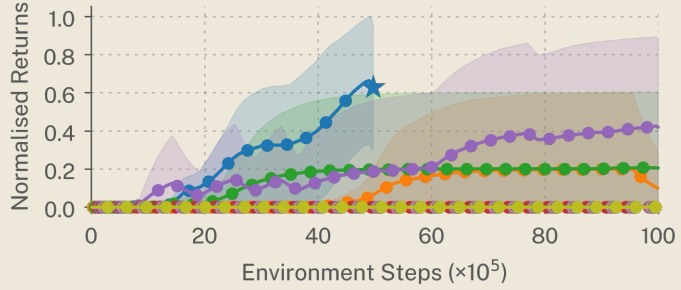
Env. II, partially observable



Env. III, partially observable



Env. IV, partially observable



IN ALL FOUR HIDDEN-INFORMATION SETTINGS

Return

WATERFALL ends highest in all four at equal training steps*

Information

Never told the task, unlike every baseline that writes rewards

Selection

Reward-writing baselines: best of 10 candidates, picked using true success

*In Env I, L2R learned faster early on, and one RL baseline caught up after 25M steps.

- **WATERFALL (Ours)**
- Code as Reward
- EUREKA
- L2R
- T2R
- RL
- RL wCBE
- RL wICM
- RL wRND

PRIVILEGED INPUTS

- Goal-conditioned
- Pre-revealed state
- Ground-truth refinement

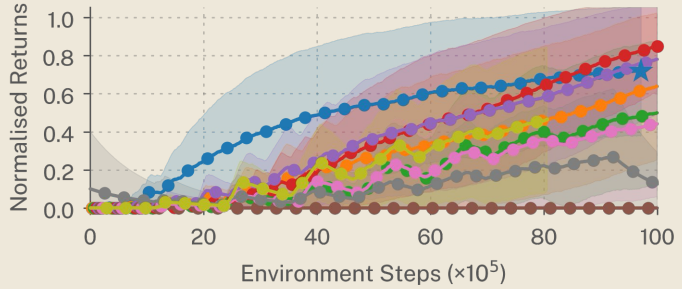
NOTES: Restyled from WATERFALL (NeurIPS 2026), Fig. 3, panels A.1–A.3 and A.7

EVIDENCE · FULLY OBSERVABLE

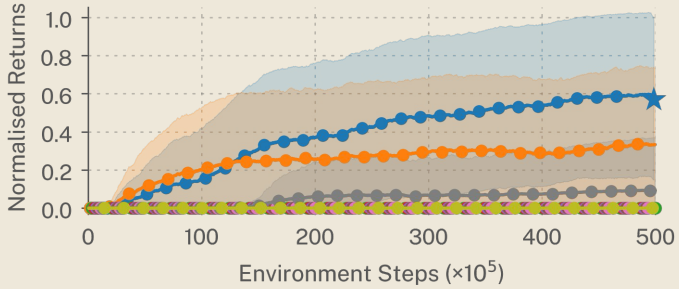
Competitive returns, with more information (fully observable)

Normalised task return, mean of 10 training runs · the star marks where WATERFALL's judge called STOP

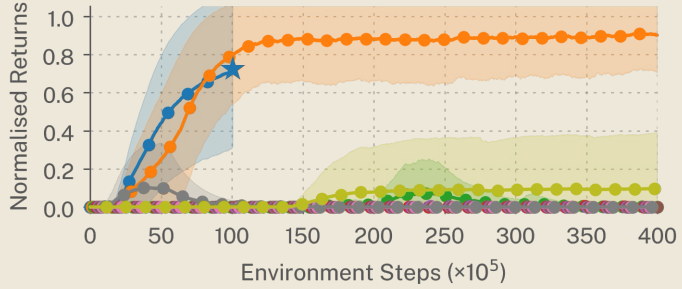
Env. I, fully observable



Env. II, fully observable



Env. III, fully observable



NO ENV IV HERE

Seeing everything would mean seeing through a closed cover, which no real sensor can. Env IV is tested only with hidden information.

IN ALL THREE FULLY OBSERVABLE SETTINGS

Visibility

Everything is visible from the start, so there is less to discover

Return

Baselines told the task catch up. WATERFALL leads in Env II, and led most of the way in Env I and III

Learner

Plain RL stays at zero in Env I: visibility helps the reward-writers, not plain RL

- **WATERFALL (Ours)**
- Code as Reward ⚑
- EUREKA ⚑ ✓
- L2R ⚑
- T2R ⚑
- RL
- RL wCBE
- RL wICM
- RL wRND

PRIVILEGED INPUTS

⚑ Goal-conditioned

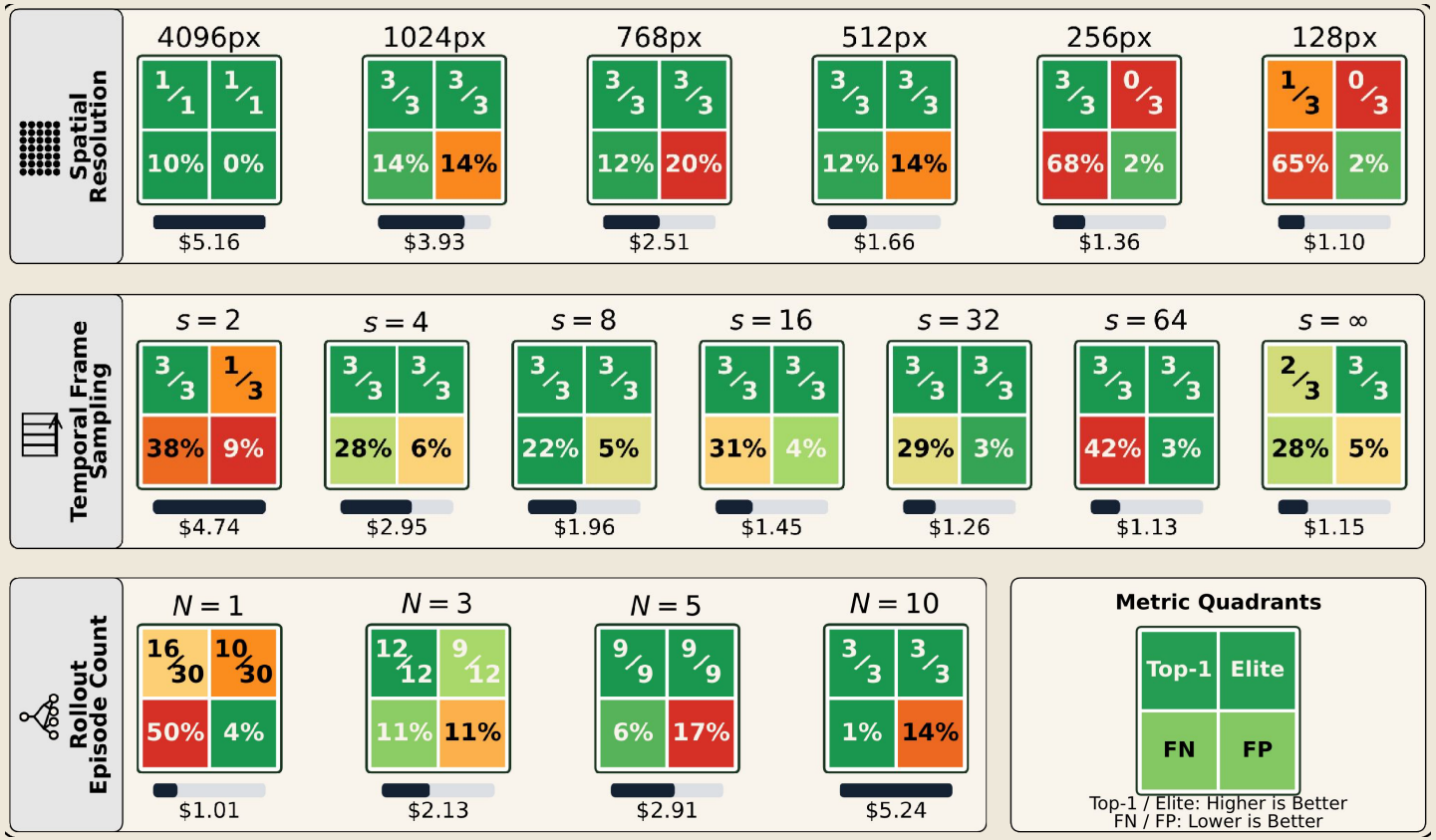
✓ Ground-truth refinement

NOTES: Restyled from WATERFALL (NeurIPS 2026), Fig. 3, panels A.4–A.6 · Env I corrected as in the camera-ready

ABLATIONS · VISUAL EVIDENCE

How much the judge needs to see

Env III, frozen at 10M steps, where the two solved candidates are known



WHAT THE JUDGE NEEDS

Spatial resolution

512 px is enough; at 128 px it fails

Temporal sampling

Every 4th to 64th frame works, and costs less

Episodes per candidate

One is brittle; five always keep the solved pair

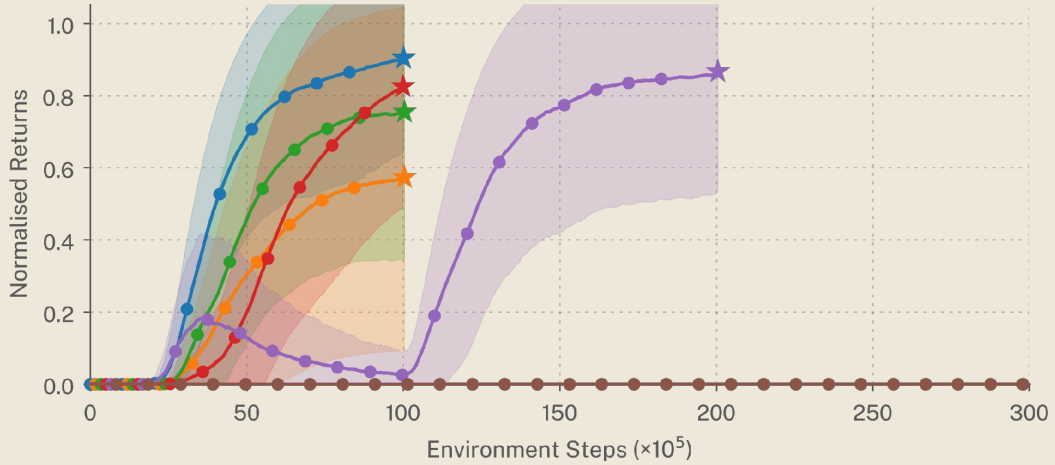
NOTES: From WATERFALL (NeurIPS 2026), Fig. 3.E and Results VI

ABLATIONS · FOUNDATION MODELS AND TOURNAMENTS

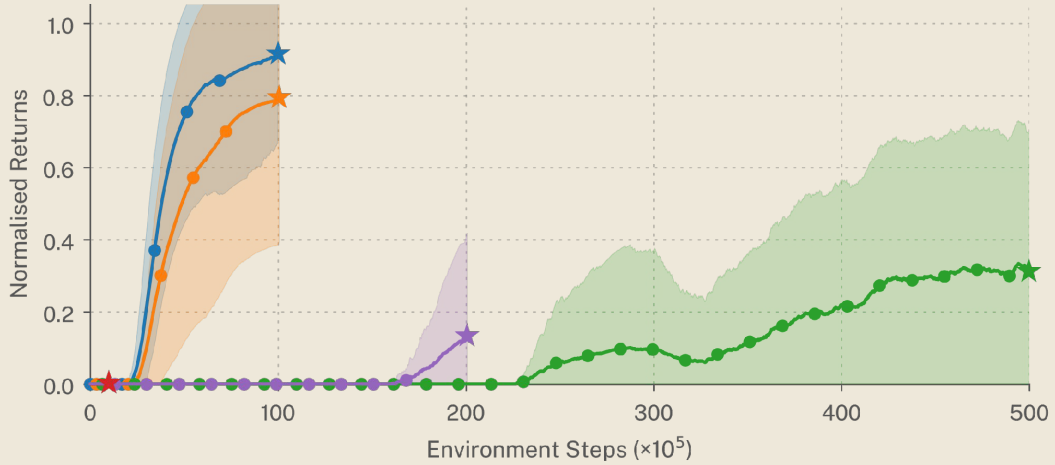
Which foundation model, and which tournament

Env III, partially observable · normalised task return, mean ± 1 SD over 10 training runs · stars mark STOP calls · $ = API cost until STOP

Env. III, vision and foundation models



Env. III, tournaments



- **GPT-5.2 high (default)**
- GPT-5.2 medium
- GPT-5.4 high
- GPT-5.4 medium
- No images
- Zero-shot

Foundation model: not tied to one; without images, far slower; zero-shot never learns.

- **Swiss (ours) · $11.43**
- Round-robin · $19.27
- King-of-the-hill · $16.79
- No evolution, no tournaments · $1.26
- Evolution, no tournaments · $5.49

Tournament: Swiss (ours) learns best at moderate cost; without tournaments it stops early, unsolved.

NOTES: Restyled from WATERFALL (NeurIPS 2026), Fig. 3, panels B and C · Results III–V

HIGHLIGHTS

Substantial learning performance despite strict information constraints

PERFORMANCE

No Info Privilege

Never told the task, it still beat baselines that were, wherever information was hidden.

TRAINING AND SAMPLE EFFICIENCY

2.8–4.6×

Fewer training steps than other reward-writing methods, as it stops or drops weak candidates early.

TRUST IN THE JUDGE

94%

Its AI judge picked the truly better candidate in 169 of 180 head-to-head comparisons.

TRUSTWORTHY STOPS

10 of 11

When it called a task solved, the call held up ten times out of eleven.

A SMALL JUDGE IS ENOUGH

$0.52

Per tournament with a small judge model that picked the best candidates as well as costlier ones.

SMARTER COMPARISONS

Half the cost

At 10M steps, a Swiss tournament picked the same top three as comparing every pair.

NOTES: Source: WATERFALL (NeurIPS 2026)

IN SHORT

Agents learn what they are rewarded for.

WATERFALL lets foundation models write and judge those rewards ***from only what the agent can see***: no goals, no task text, no demonstrations, no ground truth.

Thank you. See you at the poster.

github.com/etriantafyllidis/waterfall
eleftherios.triantafyllidis@gmail.com

THAT'S A WRAP