ALIGNING TO NEW WORLD MODELS USING CONSTITUTIONAL AI

Anonymous authors
Paper under double-blind review

ABSTRACT

As large language models (LLMs) have scaled, so has the quality of their *implicit world models*: an internal understanding of an environment. However, there are environments in which even frontier LLMs don't work correctly out of the box, which we call *new world models*. We ask how to align a frozen LLM with a new world model using a small set of solved training problems and no weight updates. We build five such environments from Manhattan routing and orbital mechanics, on which a frontier LLM fails to get more than half of the evaluation correct. We use *sleep-time compute*: before test time, the model solves training problems, checks its answers against a simulator, and revises a *constitution*, a list of single-clause rules that we place in its prompt. While prior work uses constitutions to align models with human values, we repurpose them to instead state the governing rules of an environment. We add a fixed slot budget to makes rules compete for space, and an ensemble procedure to account for variance across independent runs. Ensembled constitutions improve accuracy by 1.8× over zero-shot and outperform few-shot prompting, an RL-trained advisor, and Inverse Constitutional AI. An ensembled constitution also slightly outperforms unconstrained prompt optimization (GEPA) with a 3.9× shorter prompt. Taken together with the advantages of constitutions, such as interpretability and editability, our results show that constitutions are an effective method of aligning frozen LLMs for new world models.

1 INTRODUCTION

Large Language Models (LLMs) have become strong zero-shot performers on tasks they did not train on directly (Bommasani et al., 2021; Radford et al., 2019). One explanation for this is that pretraining at scale induces an implicit *world model*: an internal representation of an environment's states and transition rules, as opposed to a memorization of familiar trajectories (Vafa et al., 2024; 2025). Tests of this explanation find that transformers trained on turn-by-turn Manhattan navigation and planetary orbits predict familiar trajectories accurately, but lose track of state under detours and fit local curves in place of Newton's force law (Vafa et al., 2024; 2025; Liu et al., 2026).

These findings raise a question for frontier LLMs: can we align a model to an environment whose rules it does not apply correctly on its own, i.e. a *new world model*? Studying this question requires tasks that frontier LLMs still fail. Frontier LLMs now solve unmodified Manhattan routing almost perfectly (Table 1), so we change Manhattan's rules in three ways (for example, by banning left turns or closing central avenues) and anonymize its street grid. For orbits, we keep the physics of prior work (Vafa et al., 2025; Liu et al., 2026), which frontier LLMs still fail to infer from trajectories (Vafa et al., 2025). On these five environments (§4.1), a frontier LLM answers only about half of the held-out problems correctly zero-shot.

Given that frontier LLMs struggle on these new world models zero-shot, we want to align a frozen LLM, whose weights we cannot change, using a small set of solved training problems. Standard approaches fall short (§4.2). Few-shot in-context learning helps Brown et al. (2020), but its accuracy depends on which examples appear in the prompt. An advisor model trained with reinforcement learning (Ouyang et al., 2022) to write hints for the frozen LLM (Asawa et al., 2025) would get no learning signal at training if all of its sampled hints fail, which is common when rewards are sparse. Inverse Constitutional AI (Findeis et al., 2025) ranks rules by how often they apply, favoring generic

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.

advice over rare, specific rules. None of these three baselines revises its rules from the model's errors on training problems.

To learn from these errors, we propose using *sleep-time compute* (Lin et al., 2025): offline compute spent before test time in which the model works on training problems, reflects on mistakes, and revises its findings in natural language, all without changing its weights. We store these rules as a *constitution*. In prior work, a constitution aligns a LLM with human values: its principles are usually guided by human preference and govern the model's behavior, steering it away from harmful outputs (Bai et al., 2022; Huang et al., 2024; He et al., 2025b; Findeis et al., 2025; Clifford et al., 2026). Here the constitution holds a new world model: the model itself writes the environment's rules into at most P single-clause slots, which we place in the prompt at test time. Prior agents also learn rules that align an LLM world model with environment dynamics (Zhou et al., 2024) and cap or compress the learned rule set (Chen et al., 2024; Gao et al., 2026). Like them, and unlike prompt optimizers that place no limit on prompt length (Agrawal et al., 2026), we fix the budget at P slots, so a new rule can enter only by replacing or merging with an existing one. Because rules compete for space, training tends to keep general rules and drop fixes for individual problems (Rissanen, 1978; Gao et al., 2026). To account for training variance, we also develop a simple ensemble method that can aggregate independent runs without using labels.

Across the five environments, an ensembled constitution reaches 93.25% average accuracy, $1.8\times$ the zero-shot accuracy and higher than every baseline (§4.1, §4.2). It scores 20.11 pp above the best of few-shot in-context learning, the RL advisor, and Inverse Constitutional AI. It also averages higher than GEPA (Agrawal et al., 2026), given the training budget of one constitution run (93.25% vs. 91.71%), and scores higher on four of five environments with a $3.9\times$ shorter prompt. Part of this margin comes from ensembling, which depends on the constitution's structure: by being a list of discrete rules, we can easily pool the rules of five runs and select a non-redundant subset, while five free-form GEPA prompts have no such units to combine. A single constitution averages 86.67%, above the other baselines but below GEPA.

The remaining experiments study what constitutions trade for this accuracy and how they train. Compared with GEPA, a constitution leaves more work to test time (§4.3): GEPA stores step-by-step procedures in its prompt, while the model derives them from a constitution's rules, spending $1.6\times$ more reasoning tokens for nearly the same total tokens per query. The effect of the slot budget P depends on the environment (§4.4). Neo-Manhattan follows eight rules, so once $P \geq 10$ the model leaves extra slots empty and accuracy stays flat. We similarly find that after the first quarter of training on the three changed Manhattan environments, 70% of rule updates edit an existing rule in place, even if there are empty slots. On orbits, extra slots hurt performance, as they fill with narrow fixes that lower accuracy.

2 Related Work

World models in sequence models. Sequence models trained on next-token prediction encode space and time (Gurnee & Tegmark, 2024) and board states (Li et al., 2022), yet they lose track of state under detours (Vafa et al., 2024), and transformers trained on orbits tend to fit orbit shapes instead of deriving Newton's laws (Vafa et al., 2025; Liu et al., 2026). Changes to training, such as restricting attention to the recent past (Liu et al., 2026) or predicting the next latent state (Teoh et al., 2025), yield more faithful world models. We keep the weights frozen, as black-box frontier LLMs require.

Learning rules for a frozen model. Constitutional AI and its extensions align models with principles that people write in advance (Bai et al., 2022; Huang et al., 2024; He et al., 2025b). Other work learns rules from data for a frozen model's prompt. ICAI infers principles from preferences (Findeis et al., 2025; Clifford et al., 2026), HtT keeps rules that often yield correct training answers (Zhu et al., 2023), IDEA refines hypotheses as an agent explores (He et al., 2025a), and ExpeL and AutoManual build insights and manuals from past episodes (Zhao et al., 2024; Chen et al., 2024). WALL-E aligns an LLM world model with environment dynamics using learned rules that it translates into code (Zhou et al., 2024; 2025). Some also keep the rule set small: AutoManual caps its rules at 12 and merges or deletes the excess, WALL-E prunes rules by maximum coverage, and RIMRULE consolidates rules under a minimum description length objective (Gao et al., 2026). HtT and RIMRULE also transfer learned rules across models. We adopt a fixed rule budget and study its effect: extra slots stay empty

on discrete rule changes but fill with narrow fixes on continuous dynamics (§4.4). We also ensemble runs without labels and compose constitutions across rule changes and domains (§F.1).

Prompt optimization and evolving contexts. Advisor models train a second model with reinforcement learning to write per-query hints (Asawa et al., 2025), but on our environments RL often gets no gradient because all sampled hints fail (§4.2). Reflexion and Self-Refine revise answers to the same problem across attempts (Shinn et al., 2023; Madaan et al., 2023), and Dynamic Cheatsheet carries strategies across test queries (Suzgun et al., 2026); we learn before test time and evaluate on held-out problems. Prompt optimizers revise instructions from scores or feedback (Yang et al., 2024; Yuksekgonul et al., 2024; Agrawal et al., 2026), and GEPA selects them on a validation set. ACE argues that concise contexts drop domain detail and grows long playbooks instead (Zhang et al., 2026). Our results partly agree: GEPA wins when the model must derive a long procedure from bare rules (§4.2, §F.1).

3 Problem Setup and Constitutional Induction

3.1 Problem Setup: Learning a Constitution for a New World

We study how to adapt a frozen, black-box language model to an environment whose transition rules differ from its pretraining data (Vafa et al., 2024; 2025; Liu et al., 2026). On each problem, the model observes a state (such as a current intersection and destination) and chooses the next action from a fixed option set, while a deterministic simulator verifies whether the chosen step is legal and shortest-path optimal.

Drawing on Constitutional AI (Bai et al., 2022; Findeis et al., 2025) and statutory interpretation (He et al., 2025b), we represent the environment's rules as a *constitution*: a list of at most P short natural-language principles in the model's prompt. Following prior work that caps or compresses learned rule sets (Chen et al., 2024; Gao et al., 2026), and unlike prompt optimizers that place no limit on prompt length (Agrawal et al., 2026), our method enforces a hard budget of P slots so rules compete for space: once all P slots are full, a new rule can enter only by merging with or replacing an existing rule (§4.4). Adaptation proceeds in two phases:

1. **Offline learning via sleep-time compute (Lin et al., 2025):** The model tests the current constitution on training batches, revises slots whenever the simulator marks a prediction wrong, and pools rules across independent runs via a single label-free merge call (§3.3).
2. **Test-time evaluation:** We freeze the P-slot constitution in the prompt and evaluate zero-shot on 400 held-out test problems spanning 4 extrapolation tasks of 100 problems each never seen during training (Appendix A).

3.2 Four-Stage Induction Loop and Single-Clause Guardrails

Our offline update loop follows the iterative evaluate-reflect-edit cycle of reflective prompt optimizers (Yuksekgonul et al., 2024; Agrawal et al., 2026). Whereas modern reflective optimizers rewrite a single prompt block (Agrawal et al., 2026) and agent memories edit lists of free-form insights or rules (Zhao et al., 2024; Chen et al., 2024), our four stages (Figure 1; Appendix B) enforce per-slot credit assignment, fixed-P slot replacement, and a single-clause syntax across a discrete array of P slots:

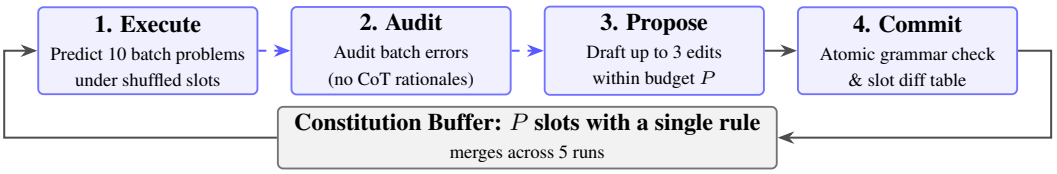


Figure 1: **Four-stage offline induction loop.** On each training batch of $B = 10$ problems, the model predicts actions using the current P-slot constitution. When all B predictions succeed, the step stops early and skips Stages 2–4. When any prediction fails, an auditor LLM checks which rules helped or failed, an architect LLM proposes up to $b = 3$ edits, and a committer verifies that every new rule is a single clause before updating the slots.

1. **Stage 1 (Execute):** The frozen student model answers all $B = 10$ problems in the mini-batch in parallel using the current constitution. We shuffle the presentation order of the P slots on each problem so the model cannot rely on rule position, and the deterministic simulator marks each prediction as `CORRECT` or `FAILED`.
2. **Stage 2 (Audit):** If all B answers are correct and cite active rules, the step stops early and skips Stages 2–4 (§4.4). Otherwise, an auditor LLM inspects the current constitution, problem prompts, and simulator verdicts (without access to the student's internal reasoning traces) and marks each rule as `OPERATIVE`, `MISAPPLIED`, or `FAULTY`, or flags a `MISSING_PRINCIPLE`.
3. **Stage 3 (Propose):** Conditioned on the Stage 2 audit report, an architect LLM proposes up to $b = 3$ slot operations (`ADD`, `MODIFY`, or `CLEAR`) to fix the failed problems within the P-slot limit; once all P slots are occupied, a new rule can enter only by replacing or merging into an existing slot.
4. **Stage 4 (Commit & Single-Clause Check):** A committer applies the proposed edits across the P slots. Because an unconstrained proposer could bypass a P-slot cap by packing multiple rules into one compound sentence, a deterministic regex filter (Appendix B.2) checks every new or modified rule before saving and rejects clause-splitting punctuation (`;`, `:`), compound conjunctions, multi-clause exception phrases, and rules with more than two prepositions.

3.3 Label-Free Merging Across Independent Runs

Because Stages 2–4 run only when a training batch contains an error or audit flag, a single run that passes early batches with partial shortcuts can stop updating before recording every boundary rule (§4.4). Across $K = 5$ independent runs that encounter training batches in different orders, most runs discover the complete rule set with distinct wording (pairwise content-word Jaccard overlap of 0.142–0.270 in §4.1), while an occasional low-scoring run stops at an incomplete intermediate set. Whereas prompt optimizers such as OPRO (Yang et al., 2024) and GEPA (Agrawal et al., 2026) require held-out validation scores to select among candidate prompts, Stage 4's single-clause rule format lets us merge independent runs $\mathcal{C}^{(1)}, \ldots, \mathcal{C}^{(K)}$ without validation data by pooling their occupied slots into $M \leq K \cdot P$ candidate rules (≤ 50 candidates for $P = 10$). In a single prompt call (Appendix B.3), the frozen model reads the numbered candidates and selects up to P rule identifiers that cover the distinct rules in the pool without repetition. This selection uses no test labels, validation scores, or citation counts, and it copies the chosen rules directly without editing their wording. A post-filter drops any candidate with > 0.75 word Jaccard overlap with an already accepted slot and fills the slot from the next ranked identifier.

4 Evaluating Constitutional World Models

We are now ready to describe the new world models that we use to empirically evaluate the efficacy of constitutions against other baselines. Our primary dataset for evaluation is the Manhattan road network: a prominent feature in pretraining corpora and a common choice in the literature Vafa et al. (2024). We study three modifications to Manhattan (described below), and also build an anonymized *Neo-Manhattan*. Finally, inspired by prior work Vafa et al. (2025); Liu et al. (2026), we also study orbital dynamics, where there exist a small set of laws that govern the motion dynamics.

4.1 Experimental Setup and Main Perturbation Results

Environments, tasks, and baselines. We evaluate constitutional induction across five different environments in which the standard rules governing the world model have been modified, grouped into three families (simulator specifications and dataset samples in Appendix A):

- **Manhattan Rule Modifications (3 environments):** Starting from a Midtown/Upper-Manhattan simulator (14th–110th Streets, 1st–12th Avenues), we introduce three global rule modifications that conflict with pretrained map priors: *No Left Turns* (interior left turns prohibited, requiring a clockwise three-right-turn loop detour), *Prohibited Central Avenues* (turns onto Fifth, Madison, and Park Avenues are banned), and *Arterial Boulevards Only* (90% of minor one-way cross streets closed, restricting crosstown turns to twelve two-way boulevards).

- **Anonymous Manhattan-like Grid (*Neo-Manhattan*):** A 100×12 Cartesian grid ('Row 1..100, Column 1..12') removing street names (to prevent lookups by the model) and enforcing directional parity (even rows East, odd rows West; odd interior columns North, even interior columns South; perimeter Rows $1, 100$ and Columns $1, 12$ are bidirectinoal).
- **Continuous Orbital Dynamics (*Kepler to Newton*):** Following Liu et al. (2026), the model observes (x, y) coordinates under a central inverse-square gravitational field ($\vec{a} = -\mu\vec{r}/\|\vec{r}\|^3, \mu = 120{,}000$) and predicts the next discretized 8-sector compass heading. As in past work on world models, this tests the model's ability to simulate forward update rules in a continuum, in contrast to the discrete path-finding set-up of the other environments.

In each environment, the model trains on $N_{\text{train}} = 200$ problems (2 training tasks of 100 problems each) for 40 steps (2 epochs of $B = 10$ batches) and is evaluated zero-shot on $N_{\text{test}} = 400$ held-out extrapolation problems (4 held-out tasks of 100 problems each; task descriptions, examples, and literature origins in Appendix A) using `Gemini 3.5 Flash` with `MEDIUM` thinking effort and $K = 5$ seeds. We compare against six baselines (Table 1):

- **Zero-Shot:** The frozen student model queried directly without a prepended constitution. This should trivially fail, as the model has never seen the world before.
- **10-Shot In-Context Learning (ICL):** Ten solved training problems (5 per training task) prepended to the query across 5 random exemplar seeds.
- **RL-based Advisor (Asawa et al. 2025):** A `Gemma 4 26B-A4B` advisor trained with GRPO on the same 200 training problems to emit per-instance hints to the student (1,213 chars/query; training details in Appendix C.1).
- **Inverse Constitutional AI (ICAI; Findeis et al. 2025):** Generates candidate rules from 100 trajectories, clusters them into 100 semantic groups, and selects the top $P = 10$ rules by net validation win count.
- **GEPA (Agrawal et al., 2026):** A sleep-time prompt optimizer that evolves a free-form prompt with no length or grammar limit. After each mini-batch of training problems, a reflection step rewrites the prompt, which comes to hold step-by-step routing and physics procedures (Appendix E.3). GEPA splits the 200 training problems into 150 for reflection and 50 for validation, and each run makes 400 student calls, as many as one constitution run (40 steps $\times$ 10 problems).
- **GEPA-Axiomatic:** GEPA with the same search loop, but with its reflection step restricted to writing short single-sentence rules, as in a constitution.

Table 1: Held-out accuracy (%, $N = 400$ test problems per environment) across the five rule-modified environments and the unmodified Manhattan control (`MEDIUM` thinking effort). Values after $\pm$ are sample standard deviations over 5 seeds (exemplar seeds for 10-Shot, runs otherwise; per-seed 10-Shot results in Appendix D.2). The single constitution uses $P = 10$; the ensembled constitution merges the 5 runs (§4.5). Prompt tokens give the mean prepended prefix per query (`chars/4`). Per-run spread (median, minimum, peak, and content-word overlap) is in Appendix D.3.

	Baselines						Constitutions (ours)	
Environment	Zero-Shot	10-Shot (5 seeds)	Advisor	ICAI	GEPA-Axiomatic (5 seeds)	GEPA (5 seeds)	Single (5 seeds)	**Ensembled**
Unmodified Manhattan (Control)	**97.50**	—	—	—	—	—	—	—
No Left Turns	56.75	75.90 ± 2.73	49.25	71.00	91.95 ± 2.06	91.35 ± 3.87	83.50 ± 20.31	**97.25**
Prohibited Avenues	43.50	74.55 ± 3.59	42.75	37.25	95.30 ± 1.44	94.35 ± 1.65	88.80 ± 10.96	**95.75**
Arterial Boulevards	75.00	80.80 ± 3.50	32.50	75.25	90.80 ± 5.89	**93.40 ± 6.00**	87.10 ± 17.63	85.75
Neo-Manhattan	38.00	70.80 ± 8.28	47.00	85.00	91.65 ± 0.29	90.75 ± 0.47	91.65 ± 1.08	**93.00**
Kepler to Newton	49.00	63.65 ± 8.11	13.25	56.00	85.15 ± 14.38	88.70 ± 9.40	82.30 ± 19.03	**94.50**
5-Environment Mean	52.45	73.14 ± 2.68	36.95	64.90	90.97	91.71	86.67	**93.25**
Prompt tokens	0	794	303	419	299	899	221	232

4.2 How do the baselines compare?

Table 1 (plotted in Figure 5, Appendix D.1) shows that the ensembled constitution (§4.5) reaches 93.25% on average, 20.11 pp above the best non-GEPA baseline (10-Shot ICL, 73.14%). GEPA averages 91.71%; the ensembled constitution scores higher on four of five environments and lower on

Arterial Boulevards, with a prompt 3.2–5.5× shorter than GEPA's. The ensembled constitution pools five runs, and GEPA has no equivalent step: a constitution is a list of discrete rules that the merge can pool and select from, while a GEPA prompt is free-form text with no such units, so five GEPA prompts cannot be combined the same way. At the budget of one run, a single constitution averages 86.67%, below both GEPA variants, because some runs score 10/10 on batches before finding every rule and then stop editing, or delete a correct rule late in training (§4.5). The next paragraphs explain why each baseline falls short and where GEPA's procedures help.

The RL Advisor gets no gradient when all rollouts follow pretraining priors: For GRPO to update the advisor on a problem, at least one of its 8 sampled rollouts must guess a legal move so that successful hints receive higher reward than failed ones. When a hidden rule contradicts pretraining priors (such as banning all left turns or closing Fifth Avenue), all 8 rollouts follow the standard map and fail together. Across all 1,000 training steps (200 per environment), 70.6% of steps produce zero reward variance across the 8 rollouts, giving zero policy gradient (per-environment counts in Table 3, Appendix C.1). As a result, it usually outputs unfinished coordinate arithmetic that guides the student into illegal move, dropping accuracy below Zero-Shot (36.95% vs. 52.45%).

ICAI favors generic pathfinding advice over rare boundary bans: ICAI scores each candidate rule in isolation by counting how many validation problems it agrees with (#correct − #incorrect), then keeps the top $P = 10$ rules. Since general advice (*"move toward the destination"*) applies on almost every step, whereas a specific ban (*"never turn onto Park Avenue"*) applies only when a route crosses that avenue, generic advice always outscores specific bans. On Prohibited Avenues, 7 of ICAI's 10 chosen slots repeat *"reduce distance to the destination"* and 3 repeat *"obey one-way streets"*, while 0 out of 10 slots mention Fifth, Madison, or Park Avenues. On Kepler to Newton, ICAI learns 4 separate rules banning turns sharper than 45° because training orbits are smooth, which drops accuracy to 24.00% on held-out *Sudden Mid-Course Deflections*.

10-Shot ICL misses unrepresented streets and cannot teach looking ahead: Ten solved examples show only the direction chosen at ten intersections, leaving the student to infer the underlying rule at test time. This fails in two ways. First, ten random examples rarely cover every restricted corridor: on Prohibited Avenues, when a seed's ten examples happen to show detours around Fifth and Madison Avenues but none at Park Avenue, the student checks the examples and concludes that Park Avenue is open. This sensitivity to which ten examples are sampled creates a 20.0 pp accuracy spread across seeds on Neo-Manhattan (61.25% to 81.25%) and a 21.25 pp spread on Kepler to Newton (50.00% to 71.25%; Table 4, Appendix D.2). Second, examples at the current intersection do not teach the student to check the rule one block ahead: on No Left Turns, even when examples avoid an immediate left turn, the student still takes routes that force an illegal left turn on the next block (68.0% on *Different Paths to the Same Intersection*).

GEPA's cached procedures are redundant when the model can derive them: On four of five environments, the ensembled constitution states only the rules and still scores higher than GEPA (Table 1); indeed, §4.3 shows that the model spends more reasoning tokens to derive the procedures itself. GEPA's procedures are also not error-free: reflection on mini-batches writes wrong street directions on No Left Turns, omits the two-way perimeter rows and columns on Neo-Manhattan, and overfits constants to the 50-problem validation split on Kepler to Newton (Appendix E.3). Restricting GEPA to atomic rules does not help: GEPA-Axiomatic scores within one point of GEPA on No Left Turns, Prohibited Avenues, and Neo-Manhattan, lower on Kepler to Newton and Arterial Boulevards, and lower on average (90.97% vs. 91.71%). The single-clause format alone therefore does not explain the constitution's advantage, which likely comes from the ensembling.

GEPA's single checklist keeps a rule that the merge drops: GEPA beats the ensembled constitution on Arterial Boulevards (93.40% vs. 85.75%) because the ensembling weights all runs equally and drops the side-street ban. Run 1's constitution states the boulevard list and the ban in separate slots (*Slot 6* and *Slot 9*; Figure 7 in Appendix D.5), and the ensembled constitution keeps only the boulevard list). GEPA's prompt avoids this loss by storing both rules in one checklist that scans north and south for the nearest open corridor before any East or West step (Appendix E.3). When rule sets are combined, this checklist matters more (Appendix F.1).

4.3 SLEEP-TIME VS. TEST-TIME TOKENS

Compared with GEPA, a constitution leaves more work to test time: GEPA writes procedures into the prompt during sleep-time compute, while a constitution states only the rules. We measure this split with two counts per query: *prompt tokens*, the prefix that sleep-time compute writes, and *reasoning tokens*, the tokens the model spends on chain-of-thought (Wei et al., 2022) at test time (Figure 2; per-environment breakdown in Table 7, Appendix E.1).

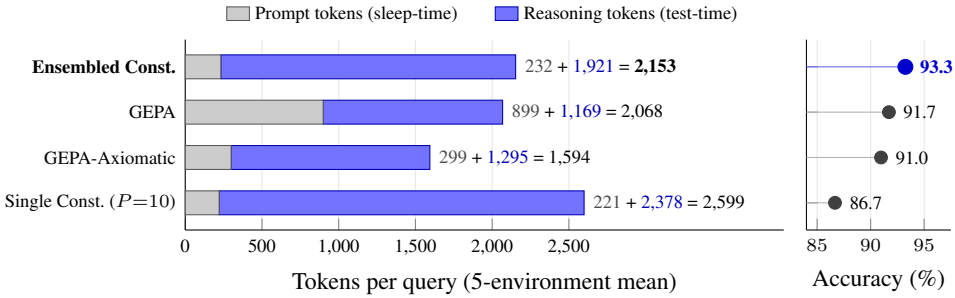


Figure 2: **Prompt and reasoning tokens per query.** Left: each bar splits the total tokens per query into prompt tokens (`chars / 4`) and reasoning tokens, averaged over the five environments. Right: 5-environment mean accuracy for the same methods. We log reasoning tokens only for the constitutions and GEPA variants; baseline accuracies are in Table 1.

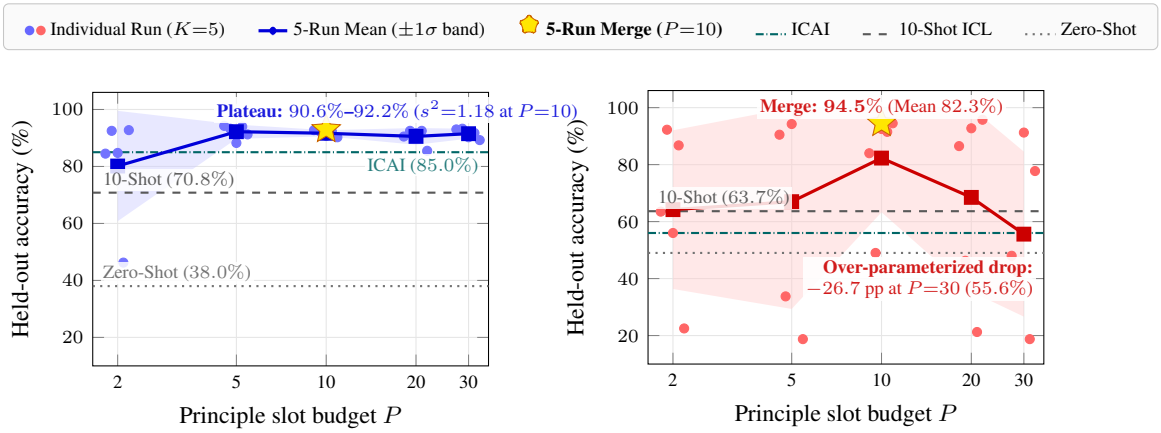
The ensembled constitution uses $3.9\times$ fewer prompt tokens than GEPA on average (232 vs. 899) and $1.6\times$ more reasoning tokens (1,921 vs. 1,169), so the two use nearly the same total tokens per query (2,153 vs. 2,068; Figure 2). GEPA stores step-by-step procedures in the prompt, such as a clockwise *"Right Loop Strategy"* for No Left Turns and a 7-step numeric procedure ending in an `atan2` sector lookup for Kepler to Newton, so the model mostly follows a checklist at test time. With a constitution, we attribute the extra reasoning tokens to the model deriving these detours and vector updates from the rules.

4.4 PRINCIPLE CAPACITY AND LEARNING DYNAMICS

Table 1 demonstrates that constitutions are quite effective in learning the rules for new world models from data. Now, we look to understand how the structure of the constitution aids in this learning. In this section, we study how the rule budget P impacts constitutional induction. Here, we report the results of an experiment sweeping $P \in \{2, 5, 10, 20, 30\}$ across 5 seeds per budget on the discrete Neo-Manhattan and continuous Kepler to Newton environments (Figure 3 and Table 6 in Appendix D.4). Note that these cover discrete and continuous domains, so it isn't apriori clear that the effect of increasing the budget for number of principles would be the same across these environments. Indeed, Figure 3 shows that the slot budget P behaves very differently depending on whether the environment has a small set of discrete rules or a continuous physical law:

For Neo-Manhattan, we observe that finding eight grid rules is sufficient as the entire Neo-Manhattan map is governed by eight rules: four one-way parity rules for interior streets (even rows go East, odd rows go West, odd columns go North, even columns go South) and four two-way highway exceptions on the outer border (Rows 1 and 100, Columns 1 and 12). With only $P = 2$ slots, the constitution lacks the space to state all four interior directions, dropping mean accuracy to 80.15%. Once $P \geq 5$, the model captures the grid rules and begins scoring $10/10$ on training batches. Because the induction loop skips rule edits whenever a training batch reaches 100% accuracy, the model leaves extra capacity unused: even at $P = 30$, it fills only 8.8 slots on average, and held-out accuracy stays flat between 90.55% and 92.20%.

In the Kepler to Newton setting, we find that more slots lead to overfitting. Unlike a street grid, continuous (x, y) orbits never repeat exact coordinates, so the model continues to encounter subtle angle boundary errors across training batches. When the budget is tight ($P = 10$), slots initially fill up quickly (7.4 active slots) with such exceptions. However, to add a rule after a mistake, stage 3 of the induction must overwrite a weaker rule, thus forcing it to replace the rough approximations with



(a) Neo-Manhattan (100×12 discrete grid)

(b) Kepler to Newton (continuous $1/r^2$ field)

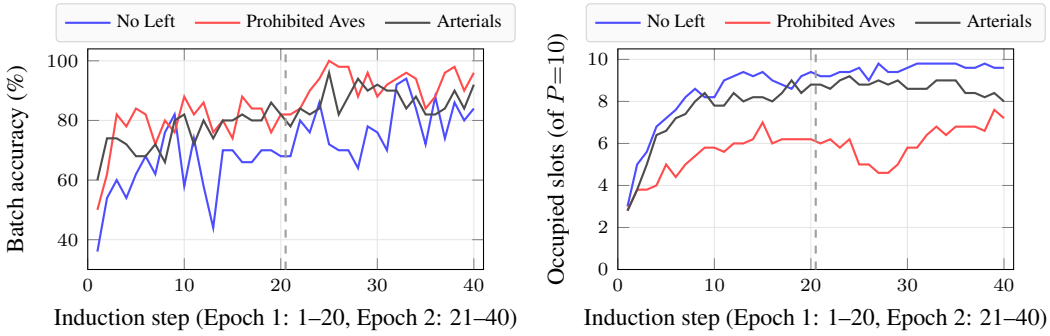
Figure 3: **Principle capacity scaling across $P \in \{2, 5, 10, 20, 30\}$ (25 runs per domain, $N = 400$).** On discrete Neo-Manhattan (a), accuracy plateaus for all $P \geq 5$ (90.55%–92.20%) because slot occupancy naturally saturates at $8.8/30$ rules once the 8 grid parity and boundary laws are discovered. On continuous Kepler to Newton (b), accuracy peaks at $P = 10$ (82.30% mean; 94.50% ensembled) and degrades by -26.70 pp at $P = 30$ (55.60%) as excess slots accumulate batch-specific tangent heuristics ($14.0/30$ occupied slots).

a single $1/r^2$ gravity law. When the budget is raised to $P = 30$, open slots are always available (14.0 occupied slots), so the architect LLM never has to delete earlier guesses. Instead of consolidating into a $1/r^2$ formula, it appends a new patch after every failed batch: Run 2 (42.25% accuracy) accumulates 16 separate coordinate-based rules, while Run 1 (48.00%) keeps a linear rule (*"divide the coordinate difference by eight"*) that only works when radius $r \approx 100$. These narrow patches fail on new orbits, dropping mean accuracy by -26.70%.

P and learning dynamics To see how the constitution uses this $P = 10$ budget during training, Figure 4 tracks batch accuracy and occupied slots across all 40 induction steps on No Left Turns, Prohibited Avenues, and Arterial Boulevards. During Steps 1–10, each run starts with an empty constitution and fills 6 to 9 slots. After Step 10, the slot count in Figure 4b stays flat while batch accuracy in Figure 4a rises by +13.3 pp from Epoch 1 to Epoch 2. During Steps 1–10, each run starts with an empty constitution and fills 6 to 9 slots. Over Steps 11–40, the slot count averages 6.1 on Prohibited Avenues, 8.6 on Arterial Boulevards, and 9.4 on No Left Turns, leaving at least one slot open on 69% of steps. This is a surprising observation as naively one would expect the LLM to use all available space.

Based on log analysis for Stage 2–4 logs across all 15 runs (5 runs per task), we notice that after Step 10 the Stage 2 auditor LLM prefers editing existing rules rather than rewriting news ones. This is perhaps because by Step 10, we find that the learned constitution usually covers the basic street rules. After this, the Stage 2 auditor LLM tends to attribute new batch mistakes to imprecise rules already in the table. Indeed, the architect LLM then edits those existing rules in-place on 70% of its updates after Step 10, even when empty slots remain open.

Another reason why the auditor LLM prefers editing existing rules over writing new ones is that in-place edits and warm starts lift Epoch 2 accuracy by +13.3 pp. At Step 10, many occupied slots hold vague reminders or single-street rules: in No Left Turns (Run 2), five slots repeat variations of *"avoid left turns"* while Slot 9 states only *"treat 66th Street as westbound."* Over Steps 11–40, the architect LLM refines those slots with the full northbound and southbound avenue lists, the odd/even street rules, and the ten two-way crosstown streets. These edits raise mean batch accuracy from 72.4% in Epoch 1 to 85.8% in Epoch 2.



(a) Batch accuracy over 40 training steps

(b) Active slot occupancy ($P = 10$)

Figure 4: **Learning dynamics across the** 40 **training steps** ($K = 5$ **runs per Manhattan modification,** $P = 10$**).** Slot occupancy (b) fills 6–9 slots within the first 10 steps and remains bounded while Stage 4 replaces local batch rules with general invariants, lifting mean batch accuracy (a) by $+13.33$ pp from Epoch 1 to Epoch 2.

4.5 Ensembling Multiple Runs for Improved Performance

In the initial stages our experiments, we noticed that there are occasional runs of the constitution learning pipelines that incur a significant performance dip. We suspect this is related to how the constitution principles evolve over time. And indeed, inspecting intermediate constitutions saved at Steps $1, 10, 20$, and 40 (Figures 6 and 7 in Appendix D.5) shed light on the dramatic variance observed in our experiments ($\sigma = 10.96$–20.31 pp in Table 1).

First, in some settings, the principles evolve in a multi-stage manner, which is correlated with high accuracy. As an example, on Kepler to Newton ($P = 5$, Run 4, 97.75% test accuracy), Step 1 writes a memoryless origin heuristic (*minimize distance to the origin*), Step 3 introduces velocity vector addition ($\vec{v} + \vec{a}$), Step 10 fits a constant radial pull (0.5) for orbits near $r \approx 490$, and Steps 30–40 refine the law to the inverse-square formula μ/r^2 with $\mu = 120{,}000$ after high-eccentricity perigee flybys fail at Step 20.

Second, the audior LLM sometimes make dramatic mistakes that are hard to recover from. Recall that all five runs train on the same 200 examples, differing only in the random seed that shuffles those examples into twenty 10-question batches per epoch. On No Left Turns, Run 4 (48.50% test accuracy) actually learns the left-turn ban by Step 20 and scores 90%–100% on Steps 30–33. At Step 36, however, the auditor LLM spuriously concludes that left turns are legal and deletes the left-turn ban, with few steps left to recover. On Prohibited Avenues, Run 5 (70.50%) fails for the opposite batch-order reason: banning Park Avenue (Slot 3) alongside a generic reminder happens to score $10/10$ on 25 of its 40 batches, skipping Stages 2–4 on 62.5% of steps so the run never adds Fifth or Madison Avenue.

Given the above, it seems inevitable to have some low scoring runs, and we observe that 4 of 5 runs finish with the complete rule set (86%–98% accuracy). Thus, by pooling the occupied slots across all 5 runs ($M \leq 50$ candidates) and selecting $P = 10$ non-redundant rules verbatim (**Select-Only Merge**, §3.3; prompt in Appendix B.3) can keep the strong rules and drop the single-run aberrations. This is reminiscent of *ensembles*, and provide a substantial performance lift for constitutions.

5 Conclusion

We align a frozen LLM with new world models by having it write an environment's rules into a fixed-size constitution during sleep-time compute, then ensembling runs label-free. On five environments, ensembled constitutions reach 93.25% accuracy ($1.8\times$ zero-shot), beat few-shot prompting, an RL advisor, and ICAI, and slightly beat GEPA with a $3.9\times$ shorter prompt. Appendix F motivates future work on composing constitutions, which suffers when one rule's detour breaks another, and distilling them to smaller models, which improves when the small model joins training.

AI Disclosure Statement

In this work, we used generative AI tools for generating and cleaning synthetic data sets, and supporting qualitative and thematic data analysis. We have not used generative AI tools to propose or refine hypotheses, design or provide feedback on research methodology or experiments, and helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, and assisting in the writing of proofs are not applicable to this work. Additionally, we used generative AI tools to create or modify scientific figures or images, create or edit software code, and to create artifacts. All AI-generated code and figures have been reviewed for accuracy/correctness to the best of our ability. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

References

Lakshya A Agrawal, Shangyin Tan, Dilara Soylu, Noah Ziems, Rishi Khare, Krista Opsahl-Ong, Arnav Singhvi, Herumb Shandilya, Michael J Ryan, Meng Jiang, et al. Gepa: Reflective prompt evolution can outperform reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 8479–8565, 2026.

Parth Asawa, Alan Zhu, Abigail O'Neill, Matei Zaharia, Alexandros G Dimakis, and Joseph E Gonzalez. How to train your advisor: Steering black-box llms with advisor models. *arXiv preprint arXiv:2510.02453*, 2025.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Minghao Chen, Yihang Li, Yanting Yang, Shiyu Yu, Binbin Lin, and Xiaofei He. Automanual: Constructing instruction manuals by llm agents via interactive environmental learning. *Advances in Neural Information Processing Systems*, 37:589–631, 2024.

Eleanor Clifford, Michael Amir, Arduin Findeis, Aaron Zhao, and Robert Mullins. Open problems in constitutional preference reconstruction. *arXiv preprint arXiv:2606.30116*, 2026.

Arduin Findeis, Timo Kaufmann, Eyke Hüllermeier, Samuel Albanie, and Robert Mullins. Inverse constitutional ai: Compressing preferences into principles. In *International Conference on Learning Representations*, volume 2025, pp. 52687–52730, 2025.

Xiang Gao, Yuguang Yao, Qi Zhang, Kaiwen Dong, Avinash Baidya, Ruocheng Guo, Hilaf Hasson, and Kamalika Das. Rimrule: Improving tool-using language agents via mdl-guided rule learning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 34631–34646, 2026.

Wes Gurnee and Max Tegmark. Language models represent space and time. In *International Conference on Learning Representations*, volume 2024, pp. 2483–2503, 2024.

Kaiyu He, Mian Zhang, Shuo Yan, Peilin Wu, and Zhiyu Zoey Chen. Idea: Enhancing the rule learning ability of large language model agent through induction, deduction, and abduction. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 13563–13597, 2025a.

Luxi He, Nimra Nadeem, Michel Liao, Howard Chen, Danqi Chen, Mariano-Florentino Cuéllar, and Peter Henderson. Statutory construction and interpretation for artificial intelligence. *arXiv preprint arXiv:2509.01186*, 2025b.

Saffron Huang, Divya Siddarth, Liane Lovitt, Thomas I Liao, Esin Durmus, Alex Tamkin, and Deep Ganguli. Collective constitutional ai: Aligning a language model with public input. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, pp. 1395–1417, 2024.

Kenneth Li, Aspen K Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Emergent world representations: Exploring a sequence model trained on a synthetic task. *arXiv preprint arXiv:2210.13382*, 2022.

Kevin Lin, Charlie Snell, Yu Wang, Charles Packer, Sarah Wooders, Ion Stoica, and Joseph E Gonzalez. Sleep-time compute: Beyond inference scaling at test-time. *arXiv preprint arXiv:2504.13171*, 2025.

Ziming Liu, Sophia Sanborn, Surya Ganguli, and Andreas Tolias. From kepler to newton: Inductive biases guide learned world models in transformers. *arXiv preprint arXiv:2602.06923*, 2026.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *Advances in neural information processing systems*, 36:46534–46594, 2023.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

J. Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978. ISSN 0005-1098. doi: https://doi.org/10.1016/0005-1098(78)90005-5. URL https://www.sciencedirect.com/science/article/pii/0005109878900055.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652, 2023.

Mirac Suzgun, Mert Yuksekgonul, Federico Bianchi, Dan Jurafsky, and James Zou. Dynamic cheatsheet: Test-time learning with adaptive memory. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7080–7106, 2026.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, et al. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

Jayden Teoh, Manan Tomar, Kwangjun Ahn, Edward S Hu, Tim Pearce, Pratyusha Sharma, Akshay Krishnamurthy, Riashat Islam, Alex Lamb, and John Langford. Next-latent prediction transformers learn compact world models. *arXiv preprint arXiv:2511.05963*, 2025.

Keyon Vafa, Justin Y Chen, Ashesh Rambachan, Jon Kleinberg, and Sendhil Mullainathan. Evaluating the world model implicit in a generative model. *Advances in Neural Information Processing Systems*, 37:26941–26975, 2024.

Keyon Vafa, Peter G Chang, Ashesh Rambachan, and Sendhil Mullainathan. What has a foundation model found? using inductive bias to probe for world models. *arXiv preprint arXiv:2507.06952*, 2025.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *International Conference on Learning Representations*, volume 2024, pp. 12028–12068, 2024.

Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Zhi Huang, Carlos Guestrin, and James Zou. Textgrad: Automatic" differentiation" via text. *arXiv preprint arXiv:2406.07496*, 2024.

Qizheng Zhang, Changran Hu, Shubhangi Upasani, Boyuan Ma, Fenglu Hong, Vamsidhar Kamanuru, Jay Rainton, Chen Wu, Mengmeng Ji, Hanchen Li, et al. Agentic context engineering: Evolving contexts for self-improving language models. In *International Conference on Learning Representations*, volume 2026, pp. 86069–86100, 2026.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19632–19642, 2024.

Siyu Zhou, Tianyi Zhou, Yijun Yang, Guodong Long, Deheng Ye, Jing Jiang, and Chengqi Zhang. Wall-e: World alignment by rule learning improves world model-based llm agents. *arXiv preprint arXiv:2410.07484*, 2024.

Siyu Zhou, Tianyi Zhou, Yijun Yang, Guodong Long, Deheng Ye, Jing Jiang, and Chengqi Zhang. Wall-e 2.0: World alignment by neurosymbolic learning improves world model-based llm agents. *arXiv preprint arXiv:2504.15785*, 2025.

Zhaocheng Zhu, Yuan Xue, Xinyun Chen, Denny Zhou, Jian Tang, Dale Schuurmans, and Hanjun Dai. Large language models can learn rules. *arXiv preprint arXiv:2310.07064*, 2023.

A Task Definitions, Literature Origins, and Dataset Samples

A.1 Overview of Training and Held-Out Extrapolation Tasks

In every environment, the model trains on $N_{\text{train}} = 200$ problems (2 training tasks of 100 problems each) and is evaluated zero-shot on $N_{\text{test}} = 400$ held-out extrapolation problems (4 test tasks of 100 problems each). On the unmodified Manhattan grid ($N = 600$ problems across all six task categories in `plain_manhattan_600.json`), `Gemini 3.5 Flash` (`models/gemini-v4p1s-rev24-ajax` at `MEDIUM` thinking effort) achieves 97.50% zero-shot accuracy on the $N = 400$ held-out test split ($390/400$) and 97.67% ($586/600$) overall (98.0% on short routes, 100.0% on detours around closed streets, 97.0% on different paths to the same intersection, 98.0% on single-intersection rule checks, 94.0% on cross-town routes, and 99.0% on blocked direct moves), confirming that the frozen model already encodes standard Manhattan street conventions.

Grid Navigation Environments (*No Left Turns*, *Prohibited Avenues*, *Arterial Boulevards*, *Neo-Manhattan*):

- **Training Tasks ($2 \times 100 = 200$ problems):**
  1. **Short Routes** (dataset ID `1a`): Given a starting intersection, current heading, and a nearby destination (2–8 blocks away), select the first legal move (`North`, `South`, `East`, `West`) along the shortest legal route. Inspired by the base next-step shortest-path setup in Vafa et al. (2024).
  2. **Single-Intersection Rule Checks** (dataset ID `1d`): Given an intersection and an adjacent 1-block destination where the direct move violates a turn restriction or one-way flow rule, select the legal alternative move. Inspired by local transition-validity probing in Vafa et al. (2024).
- **Held-Out Test Tasks ($4 \times 100 = 400$ problems):**
  1. **Detours Around Closed Streets** (dataset ID `1b`): A street segment along the normal route is closed (`Closure: North along 10th Ave from W 33rd St to W 34th St is closed`), requiring the model to choose the first step of a valid detour that respects both the road closure and the environment's traffic rules. Directly adapted from the detour test in Vafa et al. (2024), which showed that sequence models trained on shortest paths achieve high accuracy on normal routes but collapse when a single closed street forces a novel detour.
  2. **Different Paths to the Same Intersection** (dataset ID `1c`): The prompt shows two distinct 2-step arrival paths (`Path 1` and `Path 2`) that end at the same intersection, gives a destination, and asks for the next move. Adapted from the Myhill-Nerode prefix-equivalence test in Vafa et al. (2024), which evaluates whether a model merges distinct path histories that reach the same state while keeping states separate when the arrival heading constrains legal outgoing turns (as in *No Left Turns*).
  3. **Cross-Town Routes** (dataset ID `1e`): Origins and destinations are separated by 15–35 blocks across quadrants (compared to 2–8 blocks during training), testing whether the model plans ahead to legal arterial or avenue corridors over longer horizons (Vafa et al., 2024; Li et al., 2022).
  4. **Blocked Direct Moves** (dataset ID `1f`): The destination lies straight ahead along the current street or avenue, but moving directly toward it is illegal due to one-way flow or a turn ban. Every legal first move initially increases or maintains Manhattan distance ($|\Delta x| + |\Delta y|$), isolating states where greedy distance-reducing heuristics oppose true graph distance (Vafa et al., 2024).

Orbital Physics Environment (*Kepler to Newton*):

- **Training Tasks ($2 \times 100 = 200$ problems):**
  1. **Standard Orbit Steps** (dataset ID `0a`): Given an observed sequence of (x, y) coordinates along a stable elliptical orbit around $(0, 0)$, predict the 8-way compass direction (`N, S,`

`E, W, NE, NW, SE, SW`) of the next step. Inspired by the orbital trajectory prediction benchmark in Vafa et al. (2025) and Liu et al. (2026).

2. **Gravity Direction Checks** (dataset ID `0d`): Given an object's coordinates on the 2D plane, identify the compass direction of the inward gravitational pull toward the central mass at $(0, 0)$ (Liu et al., 2026).

- **Held-Out Test Tasks** ($4 \times 100 = 400$ **problems):**

1. **Sudden Mid-Course Deflections** (dataset ID `1b`): The trajectory follows a smooth orbit until the final step, where an impulse abruptly shifts the velocity vector. Predicting the next direction requires updating the post-deflection velocity with gravity toward $(0, 0)$ rather than continuing the smooth curve of earlier points. Inspired by perturbed-velocity counterfactuals in Vafa et al. (2025) and Liu et al. (2026), which distinguish geometric curve-fitting (Keplerian heuristics) from Newtonian state transitions ($\mathbf{v}_{t+1} = \mathbf{v}_t + \mathbf{a}_t$).
2. **Two Trajectories Meeting at One Point** (dataset ID `1c`): Two objects (`Track Alpha` and `Track Beta`) arrive at the same final coordinate (x_t, y_t) via different histories—50 pairs share the same last two coordinates (same velocity $\rightarrow$ `Same` next direction) and 50 pairs arrive with different velocities (`Different` next directions). Adapted from state-equivalence probing (Vafa et al., 2024; 2025) for second-order Newtonian mechanics, where the minimal physical state is the pair $(\mathbf{x}_t, \mathbf{v}_t)$ (the last two positions) rather than the full history.
3. **Slow-Moving Inward Fall** (dataset ID `1g`, `damped_suborbital_collapse`): Given a 2-point trajectory where tangential speed is low, gravitational attraction toward $(0, 0)$ overpowers inertia and pulls the next step sharply inward rather than along a circular tangent. Inspired by low-momentum sub-orbital test regimes in Vafa et al. (2025) and Liu et al. (2026).
4. **Close Flyby Slingshots** (dataset ID `1h`, `pericenter_slingshot`): Given a 2-point trajectory passing close to the central mass at $(0, 0)$, the $1/r^2$ gravitational acceleration spikes and whips the trajectory around the origin, testing whether the learned rule accounts for distance-dependent acceleration ($\|\mathbf{a}\| \propto 1/r^2$) (Vafa et al., 2025; Liu et al., 2026).

A.2 Manhattan perturbations (verbatim dataset samples)

No Left Turns, train split (`no_left_train_200.json`, $N = 200$)

Short Routes (`1a`)

```
Current Position: W 59th St & 6th Ave (arrived traveling North)
Destination: W 56th St & 7th Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `West`. Record id: `nyc_no_left_1a_077`.

Single-Intersection Rule Checks (`1d`)

```
Current Position: W 65th St & 8th Ave (arrived traveling North)
Destination: W 64th St & 8th Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `nyc_no_left_1d_014`.

No Left Turns, test split (`no_left_test_400.json`, $N = 400$)

Detours Around Closed Streets (`1b`)

```
Current Position: W 33rd St & 10th Ave (arrived traveling West)
```

```
Destination: E 39th St & Madison Ave
Closure: North along 10th Ave from W 33rd St & 10th Ave to W 34th St &
    10th Ave is closed

What direction does the driver choose towards the destination along the
    detour?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `West`. Record id: `nyc_no_left_1b_012`.

Different Paths to the Same Intersection (`1c`)

```
Path 1: W 55th St & 8th Ave -> W 56th St & 8th Ave -> W 56th St & 7th
    Ave
Path 2: W 56th St & 9th Ave -> W 56th St & 8th Ave -> W 56th St & 7th
    Ave
Destination: E 52nd St & 2nd Ave

From the current position (W 56th St & 7th Ave, arrived traveling East),
    what direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `nyc_no_left_1c_083`.

Cross-Town Routes (`1e`)

```
Current Position: W 19th St & 5th Ave (arrived traveling South)
Destination: E 37th St & Madison Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `West`. Record id: `nyc_no_left_1e_051`.

Blocked Direct Moves (`1f`)

```
Current Position: W 52nd St & 11th Ave (arrived traveling East)
Destination: W 58th St & 11th Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `South`. Record id: `nyc_no_left_1f_067`.

PROHIBITED AVENUES, TRAIN SPLIT (`PROHIBITED_AVES_TRAIN_200.JSON`, $N = 200$)

Short Routes (`1a`)

```
Current Position: W 34th St & 5th Ave (arrived traveling East)
Destination: W 21st St & 6th Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `nyc_prohibited_aves_1a_000`.

Single-Intersection Rule Checks (`1d`)

```
Current Position: W 79th St & 5th Ave (arrived traveling West)
Destination: W 61st St & 12th Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `West`. Record id: `nyc_prohibited_aves_1d_000`.

PROHIBITED AVENUES, TEST SPLIT (`PROHIBITED_AVES_TEST_400.JSON`, $N = 400$)

Detours Around Closed Streets (`1b`)

```
Current Position: E 46th St & Park Ave (arrived traveling East)
Destination: E 30th St & 1st Ave
Closure: North along Park Ave from E 46th St & Park Ave to E 47th St &
    Park Ave is closed

What direction does the driver choose towards the destination along the
    detour?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `nyc_prohibited_aves_1b_000`.

Different Paths to the Same Intersection (`1c`)

```
Path 1: E 44th St & 3rd Ave -> E 45th St & 3rd Ave -> E 45th St &
    Lexington Ave
Path 2: E 45th St & 2nd Ave -> E 45th St & 3rd Ave -> E 45th St &
    Lexington Ave
Destination: E 48th St & 3rd Ave

From the current position (E 45th St & Lexington Ave, arrived traveling
    West), what direction does the driver choose towards the
    destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `South`. Record id: `nyc_prohibited_aves_1c_000`.

Cross-Town Routes (`1e`)

```
Current Position: W 16th St & 5th Ave (arrived traveling East)
Destination: W 48th St & 7th Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `nyc_prohibited_aves_1e_000`.

Blocked Direct Moves (`1f`)

```
Current Position: E 48th St & Park Ave (arrived traveling East)
Destination: E 47th St & 1st Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `nyc_prohibited_aves_1f_000`.

ARTERIAL BOULEVARDS, TRAIN SPLIT (ARTERIALS_ONLY_TRAIN_200.JSON, $N = 200$)

Short Routes (1a)

```
Current Position: E 86th St & Park Ave (arrived traveling North)
Destination: E 91st St & 3rd Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: East. Record id: nyc_arterials_only_1a_079.

Single-Intersection Rule Checks (1d)

```
Current Position: E 23rd St & Park Ave (arrived traveling South)
Destination: W 18th St & 7th Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: West. Record id: nyc_arterials_only_1d_043.

ARTERIAL BOULEVARDS, TEST SPLIT (ARTERIALS_ONLY_TEST_400.JSON, $N = 400$)

Detours Around Closed Streets (1b)

```
Current Position: E 71st St & Park Ave (arrived traveling South)
Destination: W 68th St & 10th Ave
Closure: North along Park Ave from E 71st St & Park Ave to E 72nd St &
    Park Ave is closed

What direction does the driver choose towards the destination along the
    detour?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: South. Record id: nyc_arterials_only_1b_063.

Different Paths to the Same Intersection (1c)

```
Path 1: W 23rd St & 9th Ave -> W 23rd St & 8th Ave -> W 24th St & 8th
    Ave
Path 2: W 22nd St & 8th Ave -> W 23rd St & 8th Ave -> W 24th St & 8th
    Ave
Destination: E 27th St & 1st Ave

From the current position (W 24th St & 8th Ave, arrived traveling
    North), what direction does the driver choose towards the
    destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: North. Record id: nyc_arterials_only_1c_014.

Cross-Town Routes (1e)

```
Current Position: W 28th St & 10th Ave (arrived traveling North)
Destination: E 44th St & 2nd Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: North. Record id: nyc_arterials_only_1e_030.

Blocked Direct Moves (`1f`)

```
Current Position: E 93rd St & 2nd Ave (arrived traveling South)
Destination: W 85th St & 9th Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `South`. Record id: `nyc_arterials_only_1f_064`.

A.3 CAPACITY SWEEP ENVIRONMENTS

NEO-MANHATTAN, TRAIN SPLIT (`TRAIN_200.JSON`, $N = 200$)

Short Routes (`1a`)

```
Cartesian Grid: North increases Row (+Row), South decreases Row (-Row),
    East increases Column (+Column), West decreases Column (-Column).

Current Position: Row 76, Column 11 (arrived traveling South)
Destination: Row 78, Column 1

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `North`. Record id: `grid_extrap_1a_051`.

Single-Intersection Rule Checks (`1d`)

```
Cartesian Grid: North increases Row (+Row), South decreases Row (-Row),
    East increases Column (+Column), West decreases Column (-Column).

Current Position: Row 16, Column 6

What direction is legally permitted along Row 16 from this position?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `grid_extrap_1d_039`.

NEO-MANHATTAN, TEST SPLIT
(`GRID_VANILLA_EXTRAPOLATION_FIRST_ORDER_TEST_400.JSON`, $N = 400$)

Detours Around Closed Streets (`1b`)

```
Cartesian Grid: North increases Row (+Row), South decreases Row (-Row),
    East increases Column (+Column), West decreases Column (-Column).

Current Position: Row 47, Column 7 (arrived traveling West)
Destination: Row 45, Column 6
Closure: West along Row 47 from Row 47, Column 7 to Row 47, Column 6 is
    closed

What direction does the driver choose towards the destination along the
    detour?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `North`. Record id: `grid_firstorder_1b_045`.

Different Paths to the Same Intersection (`1c`)

```
Cartesian Grid: North increases Row (+Row), South decreases Row (-Row),
    East increases Column (+Column), West decreases Column (-Column).

Path 1: Row 8, Column 11 -> Row 11, Column 10
Path 2: Row 10, Column 12 -> Row 11, Column 10
Destination: Row 53, Column 1

From the current position (Row 11, Column 10, arrived traveling South),
    what direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `West`. Record id: `grid_firstorder_1c_050`.

Cross-Town Routes (`1e`)

```
Cartesian Grid: North increases Row (+Row), South decreases Row (-Row),
    East increases Column (+Column), West decreases Column (-Column).

Current Position: Row 42, Column 11 (arrived traveling East)
Destination: Row 7, Column 3

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `grid_firstorder_1e_025`.

Blocked Direct Moves (`1f`)

```
Cartesian Grid: North increases Row (+Row), South decreases Row (-Row),
    East increases Column (+Column), West decreases Column (-Column).

Current Position: Row 94, Column 2 (arrived traveling North)
Destination: Row 78, Column 10

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `grid_firstorder_1f_015`.

KEPLER TO NEWTON, TRAIN SPLIT (`TRAIN_0A_0D_200_V8.JSON`, $N = 200$)

Standard Orbit Steps (`0a`)

```
2D Grid:
- Reference center is located at (0, 0).
- Permitted next-step directions: North, South, East, West, North-East,
    South-East, North-West, South-West.

Observed Trajectory History:
(-157, 92) -> (-162, 104)

What is the direction of the next coordinate step from the current
    position (-162, 104)?
Options: North, South, East, West, North-East, South-East, North-West,
    South-West
ANSWER: [Direction]
```

Ground truth: `North`. Record id: `traj_train_0a_000`.

Gravity Direction Checks (`0d`)

```
2D Grid:
- Reference center is located at (0, 0).
- Permitted next-step directions: North, South, East, West, North-East,
    South-East, North-West, South-West.

Observed Trajectory History:
(112, -129) -> (104, -120)

What is the direction of the next coordinate step from the current
    position (104, -120)?
Options: North, South, East, West, North-East, South-East, North-West,
    South-West
ANSWER: [Direction]
```

Ground truth: `North-West`. Record id: `traj_train_0d_motion_063`.

KEPLER TO NEWTON, TEST SPLIT (`TEST_FIRST_ORDER_400_V8.JSON`, $N = 400$)

Sudden Mid-Course Deflections (`1b`)

```
2D Grid:
- Reference center is located at (0, 0).
- Permitted next-step directions: North, South, East, West, North-East,
    South-East, North-West, South-West.

Observed Trajectory History:
(-179, -3) -> (-178, 23) -> (-172, 48) -> (-164, 73) -> (-151, 95) ->
    (-136, 116) -> (-118, 135) -> (-79, 100)

What is the direction of the next coordinate step from the current
    position (-79, 100)?
Options: North, South, East, West, North-East, South-East, North-West,
    South-West
ANSWER: [Direction]
```

Ground truth: `South-East`. Record id: `traj_test_1b_000`.

Two Trajectories Meeting at One Point (`1c`)

```
2D Grid:
- Reference center is located at (0, 0).
- Consider two objects arriving at the final observation:
  * Track Alpha: (492, -120) -> (394, -96) -> (315, -77) -> (236, -58)
    -> (191, -72) -> (197, -48)
  * Track Beta:  (79, -19) -> (108, -26) -> (138, -34) -> (167, -41) ->
    (203, -24) -> (197, -48)

Do both tracks predict the Same next direction or Different next
    directions from position (197, -48)?
Options: Same (North), Same (South), Same (East), Same (West), Same
    (North-East), Same (South-East), North-West, Same (South-West),
    Different
ANSWER: [Choice]
```

Ground truth: `Different`. Record id: `traj_test_1c_neg_082`.

Slow-Moving Inward Fall (`1g`)

```
2D Grid:
- Reference center is located at (0, 0).
- Permitted next-step directions: North, South, East, West, North-East,
    South-East, North-West, South-West.

Observed Trajectory History:
```

```
(-79, 118) -> (-86, 109)

What is the direction of the next coordinate step from the current
    position (-86, 109)?
Options: North, South, East, West, North-East, South-East, North-West,
    South-West
ANSWER: [Direction]
```

Ground truth: `South`. Record id: `traj_test_1g_reworked_000`.

Close Flyby Slingshots (`1h`)

```
2D Grid:
- Reference center is located at (0, 0).
- Permitted next-step directions: North, South, East, West, North-East,
    South-East, North-West, South-West.

Observed Trajectory History:
(-11, 59) -> (-35, 22)

What is the direction of the next coordinate step from the current
    position (-35, 22)?
Options: North, South, East, West, North-East, South-East, North-West,
    South-West
ANSWER: [Direction]
```

Ground truth: `South-East`. Record id: `traj_test_1h_reworked_000`.

A.4 Unmodified Manhattan control and compound composition suites

1. Unmodified Manhattan Control, train split (`plain_manhattan_train_200.json`, $N = 200$)

Short Routes (`1a`)

```
Current Position: W 30th St & 9th Ave (arrived traveling East)
Destination: W 38th St & 8th Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `nyc_plain_1a_077`.

Single-Intersection Rule Checks (`1d`)

```
Current Position: W 79th St & 8th Ave (arrived traveling East)
Destination: E 79th St & Lexington Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `nyc_plain_1d_003`.

2. Unmodified Manhattan Control, test split (`plain_manhattan_test_400.json`, $N = 400$)

Detours Around Closed Streets (`1b`)

```
Current Position: W 22nd St & 12th Ave (arrived traveling North)
Destination: E 23rd St & Madison Ave
Roadblock: East along 22nd St from W 22nd St & 12th Ave is closed.
```

```
What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `North`. Record id: `nyc_plain_1b_069`.

Different Paths to the Same Intersection (`1c`)

```
Current Position: W 18th St & 5th Ave (arrived traveling East)
Destination: E 21st St & 1st Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `East`. Record id: `nyc_plain_1c_065`.

Cross-Town Routes (`1e`)

```
Current Position: E 81st St & Park Ave (arrived traveling South)
Destination: E 36th St & 2nd Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `South`. Record id: `nyc_plain_1e_050`.

Blocked Direct Moves (`1f`)

```
Current Position: E 42nd St & 1st Ave (arrived traveling North)
Destination: W 51st St & 6th Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth: `West` (`legal_optimal_turns: ['West', 'North']`). Record id: `nyc_plain_1f_045`.

3. COMPOUND MULTI-PERTURBATION SUITE (`COMPOUND_ALL_THREE_TEST_300.JSON`, $N = 300$ PER COMBINATION)

True Compound Conflict (`No Left` $\wedge$ `Prohibited Aves` $\wedge$ `Arterials Only`)

```
Current Position: W 56th St & 7th Ave (arrived traveling East)
Destination: E 52nd St & 2nd Ave

What direction does the driver choose towards the destination?
Options: North, South, East, West
ANSWER: [Direction]
```

Ground truth under conjunction: `South` (whereas single-policy `No Left Turns` optimal turn is `East`, which violates `Arterials Only` on minor 56th Street). Record id: `nyc_no_left_1c_083`.

B CONSTITUTIONAL INDUCTION PROMPTS, GUARDRAILS, AND COMPOSITION SPECIFICATIONS

This section documents the exact prompt templates, structured output schemas, deterministic grammar guardrails, and multi-seed merging / composition protocols used by the four-stage constitutional induction harness (§3).

B.1 Stage-by-Stage Prompt Templates and Output Schemas

Stage 1: Batched Execution under Shuffled Slots. For each problem s_b ($b \in \{1, \ldots, B\}$, $B = 10$), the harness draws an independent uniform random permutation σ_b over the currently occupied slots $\mathcal{C} = \{d_1, \ldots, d_{|\mathcal{C}|}\}$ so that slot index carries no positional priority. The student model π_θ receives the shuffled directives as a numbered markdown list followed by the problem query and emits a step-by-step trace terminating in an explicit action choice and cited slot list:

```
### Active Constitution (World-Model Directives)
1. {d_{\sigma_b(1)}}
...
k. {d_{\sigma_b(k)}}

### Problem
{problem_query_s_b}

Apply the active constitution directives to determine the legal and optimal
action. Cite the directive numbers that govern your step, then output:
CITED_PRINCIPLES: [comma-separated slot numbers or NONE]
ANSWER: [chosen action from Options]
```

Stage 2: Deterministic Verification and CoT-Free Diagnostic Audit. First, the environment simulator evaluates whether $\hat{a}_b \in A^*(s_b)$ (`IS_CORRECT: TRUE | FALSE`), and a lightweight trace check verifies whether the student derived $\hat{a}_b$ from an occupied slot (`REASONING_SOUND: TRUE | FALSE`, flagging runs that guess correctly while declaring the constitution silent). If all B problems satisfy `IS_CORRECT = TRUE` and `REASONING_SOUND = TRUE`, the revision gate terminates the step immediately and Stages 2–4 remain idle.

Whenever at least one batch problem fails verification or relies on an unsupported guess, the Stage 2 diagnostic auditor inspects the active constitution, problem queries, and ground-truth verifier verdicts while *withholding* the student's chain-of-thought rationale (preventing the auditor from rationalizing hallucinated student derivations):

```
### Active Constitution ({|C|} / {P} slots occupied)
[Slot 1] {d_1}
...
[Slot |C|] {d_{|C|}}

### Batch Verification Ledger (B = 10 problems; student CoT withheld)
Problem 1: {query_1} | Student: {a_hat_1}
  Verifier Verdict: {CORRECT | FAILED (Legal: A*(s_1))}
...
Problem B: {query_B} | Student: {a_hat_B}
  Verifier Verdict: {CORRECT | FAILED (Legal: A*(s_B))}

Diagnose each occupied slot and any unrepresented constraints:
- OPERATIVE: Directive states a valid law and governed a valid step.
- MISAPPLIED: Directive is valid, but its boundary scope was misapplied.
- FAULTY: Directive states an incorrect rule that caused a failure.
- MISSING_PRINCIPLE: Identify any environmental law required by the
  failed problems that is absent from Slots 1..|C|.
```

Stage 3: Architect Revision Proposal. Stage 3 receives the Stage 2 diagnostic audit together with an explicit capacity ledger stating the total slot budget P, the occupied slot count $|\mathcal{C}|$, and the remaining free slots $P - |\mathcal{C}|$. To prevent a single noisy mini-batch from overwriting the entire world model, the architect LLM is restricted to at most $k = \min(3, P)$ conceptual slot edits per batch:

```
### Capacity Ledger
Total Budget: P = {P} | Occupied: {|C|} | Free Capacity: {P - |C|}
Maximum Permitted Edits This Step: k = min(3, P) = {min(3, P)}

### Stage 2 Diagnostic Audit
{stage_2_audit_report}
```

```
Propose at most k = {min(3, P)} conceptual edits (ADD, MODIFY, or CLEAR)
to resolve the audited failures within the P-slot ceiling. If Free
Capacity is 0, any new invariant must either MODIFY an existing slot
(consolidating compatible sub-rules) or CLEAR a redundant/faulty slot.
Every proposed directive must state one atomic subject-predicate clause.
```

Stage 4: Slot-Indexed Diff Commitment. The Stage 4 committer translates the architect LLM's conceptual proposals into a structured, slot-indexed diff table over slots $1 \dots P$ and submits every `ADD` or `MODIFY` candidate to the deterministic grammar validator before updating the constitution buffer:

```
Emit the updated P-slot diff table using one operation per line:
SLOT 1: KEEP | MODIFY: "<new_atomic_directive>" | CLEAR
...
SLOT |C|: KEEP | MODIFY: "<new_atomic_directive>" | CLEAR
SLOT |C|+1: ADD: "<new_atomic_directive>" | NONE
```

B.2 Stage 4 Atomic Grammar Guardrail Specification

Before Stage 4 commits any `ADD` or `MODIFY` directive to the P-slot buffer, a deterministic regular-expression validator (`validate_atomic_directive`) enforces that the text encodes a single elementary subject-predicate clause:

1. **No clause-splitting punctuation or delimiters:** Rejects any candidate matching `[,;:()\[\]/]`, preventing semicolon-chained rules, comma-separated lists of street exceptions, or parenthetical side conditions.
2. **No coordinating or subordinating conjunctions:** Rejects word-boundary matches for `\b(and|or|nor|but|while|whereas|unless|because)\b` (case-insensitive), blocking compound rules of the form "Do X on avenues and do Y on streets unless Z."
3. **No multi-clause exception connectives:** Rejects phrase matches for `\b(instead of|except when|as well as)\b`.
4. **No participial clause stacking:** Rejects trailing participial modifiers matching `\bwith\b.*\b(permitting|forbidding|allowing|prohibiting|requiring|restricting)\b`.
5. **Preposition ceiling (≤ 2 prepositions):** Counts matches against standard spatial and relational prepositions (*in, on, at, to, from, by, with, for, of, into, onto, along, across, through, between, among, under, over, before, after, during, without, within*) and rejects any candidate exceeding two prepositions.

When a candidate directive triggers rule (1)–(5), Stage 4 returns the rule violation to the committer for at most two rewrite retries (`MAX_GUARDRAIL_RETRIES = 2`); if the second retry still violates the grammar check, the pipeline discards the edit and retains the prior slot state.

B.3 Label-Free Multi-Run Merging and World-Model Composition Prompts

Label-free multi-run merging (direct selection and clustered synthesis). Given $K = 5$ independent induction runs $\mathcal{C}^{(1)}, \dots, \mathcal{C}^{(K)}$ trained with slot budget P, we pool all occupied directives into a flat candidate bank $\{(\texttt{ID}_m, d_m)\}_{m=1}^{M}$ ($M \leq K \cdot P$) without reading any validation or test trajectories. In our primary label-free merge (direct selection), a single call to the frozen model selects up to P identifiers that form a complete, non-redundant basis:

```
### Candidate Pool Across K = 5 Independent Runs ({M} total directives)
[D01] {directive_1}
...
[D{M}] {directive_M}

Select at most P = {P} directive IDs that together cover every distinct
environmental invariant in the pool without redundancy. Do not rewrite or
synthesize new text; output only the selected IDs:
SELECTED_IDS: [Dxx, Dyy, ...]
```

To guarantee zero near-duplicate slots in the final buffer, we iterate through `SELECTED_IDS` in order and filter any candidate whose content-token Jaccard similarity with an already accepted slot exceeds 0.75, backfilling from the next ranked identifier if needed. Because every pooled directive d_m already passed `validate_atomic_directive` during Stage 4 of its source run, direct identifier selection preserves 100% grammar compliance and introduces zero unvetted claims. Our clustered synthesis variant instead asks the model to cluster the M pooled directives into P semantic groups and synthesize one consensus atomic sentence per cluster, gated by the same Stage 4 validator.

Intra-domain conjunction ($\mathcal{C}^{(A)} \wedge \mathcal{C}^{(B)}$) and cross-domain union ($\mathcal{C}^{(\textbf{Grid})} \oplus \mathcal{C}^{(\textbf{Physics})}$). For intra-domain conjunction (§F.1), where an agent must satisfy multiple concurrent perturbations on the same map, we concatenate the constituent slot arrays and retain each corridor rule's negative exclusion guard so permissive turn directives in one constitution cannot override minor-street prohibitions in another. For cross-domain union, where distinct environments share (x, y) coordinate notation with incompatible physical dynamics, we partition the combined slot buffer using explicit 41-character ontology headers:

```
### Domain: Cartesian Grid Navigation
1. {grid_directive_1}
...
P. {grid_directive_P}

### Domain: Gravitational Orbital Dynamics
1. {physics_directive_1}
...
P. {physics_directive_P}
```

C Baseline Implementation Details

C.1 RL-based Advisor (GRPO) Training Details

We follow the two-step formulation of Asawa et al. (2025): the advisor receives the problem and writes advice that must not state the answer, and the frozen student (`Gemini 3.5 Flash`, `MEDIUM` thinking effort, temperature 0) then answers the problem with the advice appended. We train one advisor per environment on the same 200 training problems used for constitutional induction, taking one problem per GRPO step in a single pass (200 steps). The advisor never sees labels: a rollout receives reward 1 if the student's next move is accepted as legal and optimal by the environment simulator and 0 otherwise, and any `ANSWER:` line is stripped from the advice before it reaches the student. Advantages are normalized within each group of 8 rollouts, so a step on which all 8 rollouts receive the same reward produces no policy gradient. Table 2 lists the full configuration.

Table 3 reports, for each environment, the number of the 200 training steps on which all 8 rollouts received the same reward, which underlies the 70.6% figure in §4.2, together with the mean advice length and held-out accuracy at test time.

D Extended Constitutional Induction Analysis

D.1 Non-GEPA Baselines vs. Induced Constitutions Bar Chart

Figure 5 displays the visual comparison of Zero-Shot, 26B GRPO Advisor, ICAI, 10-Shot ICL (5-seed mean), Single Constitution ($P = 10$, with all 5 individual seed runs plotted as yellow markers), and ensembled constitutions across all five environments (§4.1).

D.2 10-Shot In-Context Learning Multi-Seed Variance Breakdown

Table 4 reports the complete per-seed results, aggregate variance statistics, and held-out task subfamily performance for the 10-shot in-context learning (ICL) baseline across all five environments ($N = 400$ per run, 5 exemplar seeds: Seeds 42–46). In each environment, ten solved training problems (five per training task) are prepended to the query prompt.

Table 2: GRPO training configuration for the RL-based Advisor (identical across all five environments).

Setting	Value
Advisor model	`Gemma 4 26B-A4B IT`, LoRA rank 16 (base weights frozen)
Reference policy π_{ref}	Frozen base model
Advisor prompt	Two-step template of Asawa et al. (2025), thinking enabled
Training data	200 training problems, one per step, single pass
GRPO steps	200
Group size G	8 rollouts per problem
Rollout sampling	Temperature 0.7, at most 384 new tokens
Reward	Binary: student's next move legal and optimal under the simulator
Advantage	$(r_i - \bar{r})/(\sigma_r + 10^{-6})$; 0 if all rewards are equal
Objective	PPO-clipped surrogate ($\epsilon = 0.2$) $- \beta\cdot$ per-token KL to π_{ref}
KL coefficient β	0.004
Loss normalization	Mean over completion tokens per rollout, then mean over the group
Updates per step	1 gradient step per group
Optimizer	AdamW, learning rate 10^{-5} (constant), weight decay 0.01
Hardware	32 TPU devices (4 hosts), FSDP
Student (reward and test)	`Gemini 3.5 Flash`, `MEDIUM` thinking effort, temperature 0
Test-time advice	One advice string per test problem from the final checkpoint

Table 3: Per-environment Advisor training signal and test-time statistics ($N = 400$ held-out problems per environment).

Environment	Zero-variance steps (of 200)	Mean advice length (chars)	Test accuracy (%)
No Left Turns	148	1,254	49.25
Prohibited Avenues	146	1,273	42.75
Arterial Boulevards	128	1,256	32.50
Neo-Manhattan	130	1,244	47.00
Kepler to Newton	154	1,037	13.25
Total / Mean	706 of 1,000 (70.6%)	1,213	36.95

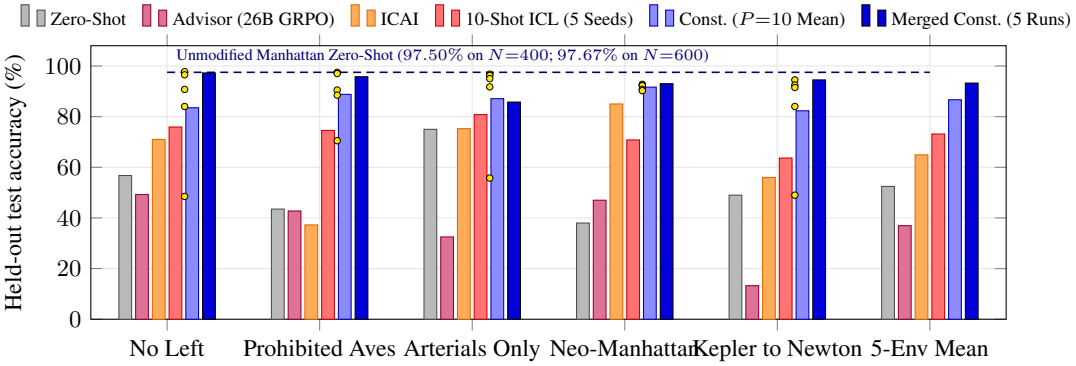


Figure 5: Held-out test accuracy ($N = 400$ per environment) across four non-GEPA baselines, single-run constitutions ($P = 10$, with individual runs plotted as yellow dots), and ensembled constitutions. On unmodified Manhattan, zero-shot accuracy is 97.50% on the $N = 400$ extrapolation slice (97.67% on $N = 600$, top dashed line), confirming that errors on the five rule-modified environments stem from unknown transition laws rather than output formatting.

The results reveal marked sample-selection instability. On discrete Neo-Manhattan and continuous Kepler to Newton, test accuracy varies by 20.00 to 21.25 percentage points across exemplar seeds ($s^2 = 68.54$ and $s^2 = 65.71$, respectively). The failure mechanism differs between domains:

- **Manhattan street corridor coverage:** Ten random exemplars rarely cover every prohibited avenue or turn boundary. When a sampled seed omits demonstrations along Park Avenue or Fifth

Avenue, the student assumes the corridor remains unrestricted. On No Left Turns, exemplars display legal moves at local intersections but cannot teach the student to plan two steps ahead, yielding poor performance (68.0%) on Myhill-Nerode state compression (Subfamily 1C).

- **Continuous velocity and gravity coverage:** In Kepler to Newton, exemplar trajectories that sample mostly circular orbits fail to expose the $1/r^2$ acceleration law under extreme step shifts (Subfamily 1B, 51.2%) or mid-course deflections, producing accuracy fluctuations from 50.00% (Seed 45) to 71.25% (Seed 43).

Table 4: Complete 5-seed held-out accuracy breakdown (%, $N = 400$ per seed) and task subfamily performance for 10-shot in-context learning across Seeds 42–46. All $\pm$ values indicate sample standard deviation ($n - 1$ denominator).

	Exemplar Seeds (%)					Aggregate Statistics			Subfamily Means (%)			
Environment	Seed 42	Seed 43	Seed 44	Seed 45	Seed 46	Mean $\pm s$	Min / Max	s^2	1B	1C	1E	1F / 1G / 1H
No Left Turns	79.00	73.50	76.25	72.75	78.00	75.90 ± 2.73	72.75 / 79.00	7.46	92.4	68.0	72.6	70.6 (1F)
Prohibited Avenues	74.25	80.50	72.75	74.25	71.00	74.55 ± 3.59	71.00 / 80.50	12.86	76.6	68.4	75.0	78.2 (1F)
Arterial Boulevards	79.25	84.00	85.00	77.00	78.75	80.80 ± 3.50	77.00 / 85.00	12.23	72.8	76.4	90.6	83.4 (1F)
Neo-Manhattan	64.00	71.50	81.25	76.00	61.25	70.80 ± 8.28	61.25 / 81.25	68.54	70.2	70.4	70.2	72.4 (1F)
Kepler to Newton	64.75	71.25	64.50	50.00	67.75	63.65 ± 8.11	50.00 / 71.25	65.71	51.2	65.8	69.2 (1G)	68.4 (1H)
5-Environment Mean	72.25	76.15	75.95	70.00	71.35	**73.14 ± 2.68**	70.00 / 76.15	7.19	72.6	69.8	75.5	74.6

D.3 Per-Run Spread of Single Constitutions

Table 5 reports the spread across the 5 single-constitution runs ($P = 10$) summarized in Table 1, together with the accuracy of choosing uniformly among legal moves (among the 8 headings for Kepler to Newton), which explains why this floor differs by environment. Jaccard is the mean pairwise overlap of content words across the 5 induced constitutions before merging.

Table 5: Per-run spread of single constitutions ($P = 10$, 5 runs, $N = 400$ held-out problems per environment, `MEDIUM` thinking effort) and the random-move floor.

Environment	Random	Mean $\pm \sigma$	Median	Min	Peak	Jaccard
Unmodified Manhattan (Control)	28.12	—	—	—	—	—
No Left Turns	26.44	83.50 ± 20.31	90.75	48.50	97.75	0.270
Prohibited Avenues	25.69	88.80 ± 10.96	90.50	70.50	97.50	0.252
Arterial Boulevards	25.38	87.10 ± 17.63	95.00	55.75	96.75	0.190
Neo-Manhattan	34.94	91.65 ± 1.08	92.25	90.25	92.75	0.218
Kepler to Newton	12.50	82.30 ± 19.03	91.50	49.00	94.50	0.142
5-Environment Mean	24.99	86.67	92.00	62.80	95.85	0.214

D.4 Complete Principle Capacity Sweep Table ($P \in \{2, 5, 10, 20, 30\}$)

Table 6 provides the complete numerical breakdown corresponding to Figure 3 (§4.4) across all 50 independent induction runs (5 seeds per budget $P \in \{2, 5, 10, 20, 30\}$ on discrete Neo-Manhattan and continuous Kepler to Newton).

D.5 Step-by-Step Rule Evolution Across 40 Training Steps and Run Variance Analysis

Figures 6 and 7 document the verbatim intermediate rules across Steps $1, 5$–$10, 20$, and 40 (§4.5) and compare the highest- and lowest-scoring runs on each Manhattan rule modification ($P = 10$).

E Extended GEPA, Token Efficiency, and Composition Details

E.1 Per-Environment Single-Domain Token Efficiency Breakdown

Table 7 reports the full single-domain prompt prefix and test-time reasoning token breakdown across all five environments and the four sleep-time world-model methods (§4.3, corresponding to Figure 2).

Table 6: Principle capacity sweep on Neo-Manhattan and Kepler to Newton (5 runs per setting, $N = 400$), compared against non-GEPA baselines at `MEDIUM` thinking effort. Min reports the lowest-scoring run, s^2 the sample variance across runs, and Occupancy the average number of active slots at Step 40.

	Neo-Manhattan (100×12 discrete grid)				Kepler to Newton (continuous $1/r^2$ field)			
Configuration	Mean $\pm$ σ	Min	s^2	Occupancy	Mean $\pm$ σ	Min	s^2	Occupancy
$P = 2$	80.15 ± 19.37	46.25	375.14	2.0 / 2	64.20 ± 27.85	22.50	775.48	2.0 / 2
$P = 5$	$\mathbf{92.20 \pm 2.50}$	88.25	6.26	5.0 / 5	67.00 ± 37.66	18.75	1418.50	4.8 / 5
$P = 10$	91.65 ± 1.08	**90.25**	**1.18**	7.4 / 10	$\mathbf{82.30 \pm 19.03}$	**49.00**	**362.33**	7.4 / 10
$P = 20$	90.55 ± 2.94	85.50	8.64	8.2 / 20	68.50 ± 33.10	21.25	1095.56	9.4 / 20
$P = 30$	91.50 ± 1.73	89.25	3.00	8.8 / 30	55.60 ± 28.96	18.75	838.86	14.0 / 30
Ensembled ($P = 10$)	**93.00** (715 chars / 528 clean)				**94.50** (1,116 chars / 930 clean)			
Zero-Shot	38.00				49.00			
10-Shot ICL (5 Seeds)	70.80 ± 8.28	61.25	68.54	—	63.65 ± 8.11	50.00	65.71	—
Advisor (26B GRPO)	47.00				13.25			
ICAI ($P = 10$)	85.00				56.00			

Run (Test Acc.)	**Step 1 (Initial Shortcut)**	**Steps 5–10 (Local Rule)**	**Step 20 (Mid-Run Error)**	**Step 40 (Final Rule Set)**
Kepler to Newton P=5, Run 4 (**97.75%** test)	*Batch acc: 50%* *Slot 1:* "Select the compass direction that minimizes the distance to the origin." *Slot 2:* "Ignore prior trajectory history when determining the optimal direction."	*Batch acc: 50% → 80%* *Slot 1:* "Compute the next displacement by adding the previous velocity vector to a radial acceleration vector directed toward the origin." *Slot 2:* "Scale the inward radial unit vector by 0.5 to determine the acceleration vector."	*Batch acc: 50% (flyby fail)* Constant-magnitude rule ($0.5\,\hat{r}$) passes circular orbits at Step 10 (80%) but fails eccentric flybys at Step 20. *Slot 4 added:* "Increase the inward radial pull strength when the craft is closer to the origin."	*Batch acc: 90%, Test: 97.75%* *Slot 1:* "Calculate the target displacement vector by adding the current velocity vector to the acceleration vector." *Slot 2:* "Calculate the acceleration vector as the unit vector toward the origin multiplied by 120000 divided by the squared distance to the origin."
Prohibited Aves P=10, Run 3 (**97.50%** test)	*Batch acc: 50%* *Slot 1:* "Do not turn onto an avenue if that turn is prohibited for the destination." (circular restatement)	*Batch acc: 90%* Adds base-map street parity (*Slots 4–5*) while keeping the circular Slot 1 rule; passes batches that lack central avenue turns.	*Batch acc: 80%* *Slot 8:* "Proceed straight along the current street at intersections where turning onto the destination avenue is prohibited."	*Batch acc: 90%, Test: 97.50%* *Slot 1:* "Never turn East or West from Fifth Avenue onto a cross street." *Slot 9:* "The vehicle must never turn onto Fifth Avenue Madison Avenue Park Avenue."
No Left Turns P=10, Run 1 (**97.75%** test)	*Batch acc: 30%* Writes 3 base-map rules (avenue flow, street parity, Manhattan distance) and omits turn bans entirely.	*Batch acc: 90%* *Slot 4:* "Left turns are prohibited at interior intersections." *Slot 6:* "Three consecutive right turns execute a directional reversal when left turns are prohibited."	*Batch acc: 70% (regression)* Misinterprets a boundary avenue turn and adds *Slot 8:* "Left turns onto avenues override distance minimization," contradicting Slot 4.	*Batch acc: 100%, Test: 97.75%* Epoch 2 narrows Slot 8 to legal boundary avenues (*"Right turns onto avenues..."*) and pairs *Slot 4* with *Slot 9* clockwise loop detours.

Figure 6: Rule evolution from Step 1 to Step 40 across three high-scoring training trajectories.

Both exact character counts and token equivalents (`chars / 4`, alongside regex word/subword token counts) are reproduced from `single_domain_details.json`.

	Highest-Scoring Run	Lowest-Scoring Run
No Left Turns	**Run 1, 97.75%** *Slot 4:* "Left turns are prohibited at interior intersections"	**Run 4, 48.50%** Nine occupied slots, none prohibiting a turn. *Slot 7:* "Vehicles must minimize turns for equal distance paths"
Prohibited Aves	**Run 3, 97.50%** *Slot 9:* "The vehicle must never turn onto Fifth Avenue Madison Avenue Park Avenue."	**Run 5, 70.50%** *Slot 3:* "Travel on Park Avenue is prohibited." *Slot 2:* "Travel East of 1st Avenue is prohibited."
Arterial Boulevards	**Run 1, 95.00%** (**Run 2, 96.75%**) *Slot 6:* "Vehicles must turn onto intersecting major crosstown corridors toward the destination." *Slot 9:* "Vehicles must avoid minor intermediate cross streets except the destination street."	**Run 3, 55.75%** *Slot 8:* "Every routing step must prioritize turns onto major crosstown corridors toward the destination cardinal direction." No slot forbids minor streets.

Figure 7: Comparison of rules from the highest- and lowest-scoring runs on each Manhattan modification ($P = 10$).

Table 7: Single-domain prompt prefix and test-time reasoning token breakdown across all five environments ($N = 400$ per environment, `MEDIUM` thinking effort). Prompt tokens measure the prepended prefix (`chars / 4`); reasoning tokens measure internal chain-of-thought (`thoughts`) per query in both `chars / 4` (`~Tok`) and regex subword tokens (`Regex Tok`); total tokens sum `Prompt ~Tok + Reasoning ~Tok`.

Environment	Method	Held-Out Acc (%)	Raw Prefix (Chars / ~Tok)	Clean Directives (Chars / ~Tok)	Reasoning ~Tok (`chars/4`)	Reasoning (Regex Tok)	Total ~Tok / Query
No Left Turns	Single Constitution ($P = 10$, 5 seeds)	83.50 ± 20.31	1,105 (276)	769 (192)	2,305	2,290	2,581
	Ensembled Constitution	**97.25**	803 (201)	615 (154)	2,270	2,309	**2,471**
	GEPA-Axiomatic (5 seeds)	91.95 ± 2.06	1,533 (383)	1,533 (383)	1,891	1,860	2,274
	GEPA (5 seeds)	91.35 ± 3.87	4,441 (1,110)	4,441 (1,110)	1,599	1,588	2,709
Prohibited Avenues	Single Constitution ($P = 10$, 5 seeds)	88.80 ± 10.96	786 (197)	509 (127)	1,327	1,237	1,524
	Ensembled Constitution	**95.75**	858 (215)	672 (168)	1,220	1,158	**1,435**
	GEPA-Axiomatic (5 seeds)	95.30 ± 1.44	959 (240)	959 (240)	1,109	1,047	1,349
	GEPA (5 seeds)	94.35 ± 1.65	2,921 (730)	2,921 (730)	985	956	1,715
Arterial Boulevards	Single Constitution ($P = 10$, 5 seeds)	87.10 ± 17.63	1,012 (253)	716 (179)	1,428	1,358	**1,681**
	Ensembled Constitution	85.75	1,150 (288)	977 (244)	2,117	2,050	2,405
	GEPA-Axiomatic (5 seeds)	90.80 ± 5.89	1,122 (281)	1,122 (281)	1,012	979	1,293
	GEPA (5 seeds)	**93.40 ± 6.00**	3,951 (988)	3,951 (988)	980	971	1,968
Neo-Manhattan	Single Constitution ($P = 10$, 5 seeds)	91.65 ± 1.08	591 (148)	551 (138)	3,778	3,995	3,926
	Ensembled Constitution	**93.00**	715 (179)	528 (132)	2,649	2,870	2,828
	GEPA-Axiomatic (5 seeds)	91.65 ± 0.29	883 (221)	883 (221)	1,432	1,750	1,653
	GEPA (5 seeds)	90.75 ± 0.47	2,994 (749)	2,994 (749)	1,524	2,068	2,273
Kepler to Newton	Single Constitution ($P = 10$, 5 seeds)	82.30 ± 19.03	923 (231)	886 (221)	3,050	3,813	3,281
	Ensembled Constitution	**94.50**	1,116 (279)	930 (232)	1,350	1,760	**1,629**
	GEPA-Axiomatic (5 seeds)	85.15 ± 14.38	1,489 (372)	1,489 (372)	1,032	1,521	1,404
	GEPA (5 seeds)	88.70 ± 9.40	3,665 (916)	3,665 (916)	756	1,120	1,672
5-Environment Mean	Single Constitution ($P = 10$, 5 seeds)	86.67 (Peak **95.85**)	883 (**221**)	686 (**172**)	2,378	2,539	2,599
	Ensembled Constitution	**93.25** (Peak **95.85**)	928 (232)	744 (186)	1,921	2,029	**2,153**
	GEPA-Axiomatic (5 seeds)	90.97 (Peak 94.60)	1,197 (299)	1,197 (299)	1,295	1,431	1,594
	GEPA (5 seeds)	91.71 (Peak 95.45)	3,595 (899)	3,595 (899)	**1,169**	**1,341**	2,068

E.2 Cross-Domain World-Model Union and Missing-Principle Diagnostic

Table 8 reports the cross-domain composition evaluation combining the discrete Neo-Manhattan ($P = 5$, Run 1, 94.25% alone) and continuous Kepler to Newton ($P = 5$, Run 4, 97.75% alone) constitutions under two-line domain headers (`[Domain A: Grid Navigation]` and `[Domain B: Orbital Physics]`) across $N = 800$ held-out queries (400 per domain; §F.1). Table 9 reports the missing-principle diagnostic on the 1,200 Manhattan composition queries when we append the single omitted side-street exclusion principle from the Run 1 *Arterial Boulevards* candidate pool (*Slot 9*: *"Vehicles must avoid minor intermediate cross streets except the destination street"*) to our ensembled constitution.

E.3 Excerpts of Evolved GEPA Prompts, Cached Algorithms, and Student-Adapted Constitutions

We document the verbatim prompt clauses from `gepa_report.json`, `compound_composition_details.json`, and `gemma4_transfer_campaign` cited in §4.2–§F.2:

Table 8: Cross-domain union combining Neo-Manhattan ($P = 5$) and Kepler to Newton ($P = 5$) under two-line domain headers ($N = 400$ per domain, 800 total queries at `MEDIUM` thinking effort). Composed Constitutions outperform Composed GEPA across both domains (94.50% vs. 93.00%) at a 10.5× smaller prefix (253 vs. 2,660 prompt tokens) and 1.5× lower total tokens on Kepler (2,466 vs. 3,761).

	Prompt Prefix		Held-Out Accuracy (%)			Reasoning ¯Tok (`chars/4`)		Total ¯Tok / Query	
Evaluation Condition	Chars	Prompt ¯Tok	Neo-Manhattan	Kepler to Newton	2-Domain Mean	Grid (1B Slice) / Kepler	2-Domain Mean	Kepler Total	2-Domain Mean
Separate Single-Domain ($P = 5$, Run 1 / Run 4)	465	116	94.25	97.75	96.00	5,684 (3,026†) / 1,463	3,573 (2,759†)	1,579	3,690
Composed Constitution ($P = 5 \oplus 5$)	1,013	**253**	**93.00**	**96.00**	**94.50**	6,248 (2,798) / 2,213	4,231	**2,466** (**1.5**× lower)	4,484
Composed GEPA	10,638	2,660	90.00	**96.00**	93.00	**2,313** (1,513) / **1,101**	**1,707**	3,761	**4,367**

†Parenthetical values under Separate Single-Domain denote the 5-run $P = 5$ mean across Seeds 42–46 (3,026 tok on Neo-Manhattan; 2,492 tok on Kepler; 2,759 mean).

Table 9: Diagnostic ablation on the 1,200 Manhattan composition routes (§F.1). Appending the single side-street exclusion principle omitted by the unweighted *Arterial Boulevards* merge (*Slot 9* from Run 1: *"Vehicles must avoid minor intermediate cross streets except the destination street"*) raises conflict-route accuracy by +12.33 pp (65.67% → 78.00%) and overall accuracy by +8.67 pp (78.50% → 87.17%), while the remaining 11.67 pp gap to Composed GEPA on conflict routes (78.00% vs. 89.67%) reflects test-time multi-block search.

	Overall Accuracy by Rule Conjunction ($N = 300$ each)				Macro Accuracy (1,200)		
Composed World Model	No Left ∧ Prohib. Aves	No Left ∧ Arterials	Prohib. Aves ∧ Arterials	All Three Rules	Compatible (600)	Conflict (600)	Overall (1,200)
Ensembled Constitution	89.67	66.33	81.00	77.00	91.33	65.67	78.50
Ensembled (+ omitted Run 1 *Slot 9* principle)	89.67	79.67	88.67	89.67	96.33	78.00	87.17
Composed GEPA	**90.67**	**92.33**	**96.67**	**91.00**	95.67	**89.67**	**92.67**

- **Cached Algorithms in GEPA Prompts vs. Single-Clause Constitutional Slots (§4.2–§4.3):** Whereas Stage 4 restricts each constitutional slot to one subject-predicate clause (§B.2), all 25 evolved GEPA prompts cache multi-step decision algorithms directly in the prompt prefix:
  1. *No Left Turns (GEPA Run 2,* 6,198 *chars):* Includes a dedicated procedural section (`#### The "Right Loop" Strategy (Replacing a Left Turn):` *"When you need to travel in the direction that requires a prohibited Left Turn, execute a 3-turn loop around a block using only Right Turns and Straight movements: Turn Right at the current intersection, travel 1 block, turn Right onto the parallel street, travel 1 block, and turn Right onto the target corridor."*) followed by `### 4. Step-by-Step Decision Algorithm`. In the ensembled constitution (803 chars), the rule is a single clause (*"Left turns are prohibited at interior intersections"*), requiring the student to derive the clockwise block detour in its test-time chain of thought (2,270 vs. 1,599 reasoning tokens; Table 7).
  2. *Arterial Boulevards (GEPA Run 1,* 3,951 *chars):* Includes `# Step-by-Step Reasoning Guide` (*"Step 3. Apply Arterial & Destination Rules: Minor cross streets cannot be used for intermediate crosstown travel. If your current cross street is closed or minor, continue North or South along the current avenue until you reach the nearest two-way arterial boulevard (*14*th,* 23*rd,* 34*th,* 42*nd,* 57*th,* 72*nd,* 79*th,* 86*th,* 96*th,* 106*th,* 116*th, or* 125*th Street).*"*).
  3. *Kepler to Newton (GEPA Run 1,* 3,665 *chars):* Embeds a 7-step numerical script (`Step 1: Compute Baseline Velocity` → `Step 2: Compute Distance r` → `Step 3: Compute Gravitational Acceleration` → `Step 4: Next Step Vector` → `Step 5: Map to 8-Sector Compass` using `atan2` degree boundaries $[-22.5°, 22.5°]$, $[22.5°, 67.5°]$, etc.) alongside two worked numerical examples.
- **Over-Specified Map Tables and Constants in GEPA Prompts (§4.2 and §F.1):**
  1. *Neo-Manhattan Closed Parity Table (GEPA Run 3,* 91.25% *test accuracy): "At any given intersection (`Row, Column`), you have at most two legal directions of travel: one vertical (along the Column) and one horizontal (along the Row)... Odd-numbered Columns (*$1, 3, 5, 7, 9, 11$*): Northbound only (`+Row`). Even-numbered Columns (*$2, 4, 6, 8, 10, 12$*): Southbound only (`-Row`)."* Because this table omits perimeter exceptions, GEPA fails test queries requiring Southbound travel along two-way Column 1 or Northbound travel along two-way Column 12.
  2. *Kepler to Newton Validation-Tuned Constant μ and Slope Override (GEPA Run 1,* 74.00% *test accuracy): "Compute the gravitational pull toward the attractor at* $(0, 0)$ *using* $\mu \approx 250,000$*: $a_x = -250000 \cdot x_{curr}/r^3$, $a_y = -250000 \cdot y_{curr}/r^3$... Step 6: Initial Rest State Special Rules: If $|dx| > 2|dy|$ (shallow slope, $R > 2$), rotate* $45°$ *counter-clockwise from the radial direction."* On close flybys near the origin (1H), this 2.08× inflated constant and single-origin slope patch drop accuracy to 38.00%.

3. *Conflicting Street Tables Under Multi-Prompt Composition (§F.1):* On No Left Turns, GEPA Run 1 writes *"Major Crosstown Arterials: 34th St (Two-way), 42nd St (Two-way), 57th St (Two-way), 66th St (Westbound), 72nd St (Two-way)"* (mislabeling two-way 66th Street as Westbound), whereas Run 4 writes *"Madison Ave: Southbound (V)"*. When concatenated with the Prohibited-Avenues and Arterial-Boulevards GEPA prompts across the 1,200 composition queries in `compound_composition_details.json`, the model executes sequential multi-prompt checklists on $1{,}143/1{,}200$ queries (95.3%) vs. $0/1{,}200$ (0%) for composed constitutions, raising reasoning tokens from $1{,}504$ to $2{,}395$ per query (Table 10).

- **Teacher-Transferred vs. Student-in-the-Loop Constitutions on `Gemma 4 12B` (§F.2):** Comparing `track3a_teacher/select_only_slots.json` ($P = 10$, direct transfer from `Gemini 3.5 Flash`) against `track2_merged/select_only_slots.json` ($P = 15$, induced with `Gemma 4 12B` in the loop) shows why student-in-the-loop induction raises accuracy from 59.50% to 74.30% (Table 11):
  1. *Prohibited Avenues (23.25% direct teacher transfer vs. 77.45% GEPA):* The transferred teacher constitution includes *Slot 5: "The vehicle must never turn onto Fifth Avenue Madison Avenue Park Avenue"*, and on Arterial Boulevards (65.50% direct transfer vs. 85.00% GEPA) it includes *Slot 6* (the twelve major two-way corridors) and *Slot 8* (*"Vehicles must turn onto intersecting major crosstown corridors toward the destination"*). Despite having the exact declarative bans, the 12B model fails to plan multi-block detours around the prohibited avenues without GEPA's step-by-step routing checklist.
  2. *No Left Turns (40.25% direct teacher transfer $\rightarrow$ 75.75% student-in-the-loop merge):* In direct transfer, the label-free teacher merge retains ten avenue and street flow rules ending with *Slot 10: "U turns at intersections are prohibited"*, omitting the left-turn clause and leaving compass turn transitions implicit. When `Gemini 3.5 Flash` inspects `Gemma 4 12B`'s training errors in Stage 2, the student-in-the-loop merge (`no_left/track2_merged/select_only_slots.json`) induces explicit compass transition definitions and East-to-West avenue ordering: *Slot 11: "Vehicles must never turn left at any intersection with turning left defined as turning South from West also East from South also North from East also West from North"*, *Slot 14: "Avenues from East to West are First Second Third Lexington Park Madison Fifth Sixth Seventh Eighth Ninth Tenth Eleventh Twelfth"*, and *Slot 15: "Going West increases numbered avenue values while going East decreases numbered avenue values"*.

F Exploring Limits of Constitutions

We also present some preliminary results that explores the limits of constitutional representation for learning. Specifically, we explore the usefulness of the constitutional representation in two use cases: composing rules for multiple new worlds together, and for distilling learnings for new worlds from larger models to smaller ones. These are early results, so we include them in the appendix for completeness.

F.1 Prompt Composition

In deployed settings, an agent that has learned separate prompts for individual modifications (A and B) may face a combined environment ($A \wedge B$) where both rule sets apply at once. We define *prompt composition* as concatenating the prompts learned independently on separate environments under two-line module headers (such as `[Domain A: Grid Navigation]` and `[Domain B: Orbital Physics]`) and evaluating the agent on the combined environment without new sleep-time rollouts. Because a constitution states only minimal rules and relies on the model to derive the routing algorithm at test time (§4.3), concatenating constitutions requires `Gemini 3.5 Flash` to satisfy multiple constraints at once in a single chain-of-thought. We test this in two setups:

1. **Within-domain intersection** (Table 10): We combine our ensembled constitutions across the three Manhattan modifications in all three pairs and all three together (300 routes per combination; 1,200 total). To isolate when rules interfere at test time, each 300-route set is split evenly into 150 *compatible* routes, where the detour required by one rule stays legal under the other rule, and 150 *conflict* routes, where the standard detour for the first rule directly violates the second rule

(for example, a clockwise 3-right-turn loop around a banned left turn that lands on a closed minor cross street or a banned southbound avenue).

2. **Cross-domain union** (Table 8 in Appendix E.2): We concatenate the 5-rule *Neo-Manhattan* constitution and the 5-rule *Kepler* constitution under their two-line domain headers into a 10-rule prompt and evaluate across 400 grid queries and 400 orbital queries.[1]

Table 10: Zero-shot composition across four Manhattan rule combinations ($N = 300$ routes each: 150 compatible + 150 conflict; 1,200 total) for our ensembled constitution vs. Composed GEPA.

	Overall Accuracy by Rule Conjunction ($N = 300$ each)					Macro Accuracy (1,200)			Mean Tokens / Query		
Composed World Model	No Left ∧ Prohib. Aves	No Left ∧ Arterials	Prohib. Aves ∧ Arterials	All Three Rules	Compatible (600)	Conflict (600)	Overall (1,200)	Prompt	Reasoning	Total
Ensembled Constitution	89.67	66.33	81.00	77.00	91.33	65.67	78.50	**428**	**1,464**	**1,892**
Composed GEPA	**90.67**	**92.33**	**96.67**	**91.00**	**95.67**	**89.67**	**92.67**	3,152	2,395	5,547

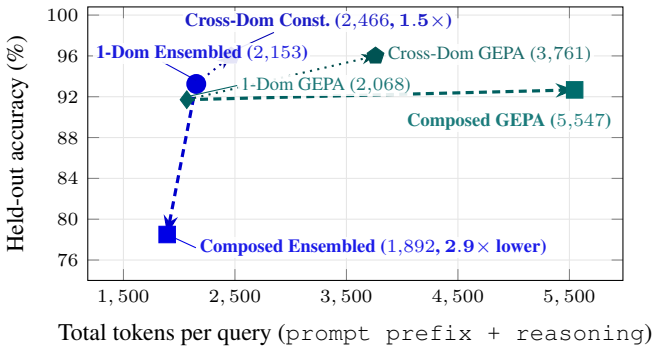


Figure 8: **Accuracy vs. total tokens per query under composition.** Arrows start at the single-environment means over five environments (Figure 2) and end at the composed settings: the four Manhattan rule combinations (squares, 1,200 routes) and the cross-domain union evaluated on Kepler to Newton (pentagons, 400 queries). On the Manhattan combinations, Composed GEPA reaches 92.67% at 5,547 total tokens per query vs. 78.50% at 1,892 for the ensembled constitution. On the cross-domain union, both reach 96.00%, and the constitution uses 2,466 tokens vs. 3,761.

- **Compatible and cross-domain rules compose cleanly, whereas conflict routes overload test-time search:** When the composed rules do not interfere with each other's local detours, the ensembled constitution matches or beats Composed GEPA: it reaches 91.33% on the 600 compatible Manhattan routes (Table 10) and 94.50% vs. 93.00% on the cross-domain Neo-Manhattan ∪ Kepler union at 1.5× lower total tokens on Kepler (Table 8). On the 600 conflict routes, however, ensembled constitution accuracy drops to 65.67%. When a 3-right-turn loop around a banned left turn lands on a minor cross street closed by *Arterial Boulevards*, a legal route requires simulating candidate loops and backtracking in chain-of-thought. Instead of expanding its search depth, `Gemini 3.5 Flash` spends 24% *fewer* reasoning tokens under composition than in single environments (1,464 vs. 1,921) and halts after satisfying the first rule. Composed GEPA avoids joint search by pre-computing each subgoal in the prompt and verifying the prompts sequentially one checklist after another (Appendix E.3), at 2.9× the total token cost (5,547 vs. 1,892; Figure 8).
- **Restoring the omitted side-street principle recovers half the conflict gap, while the rest reflects test-time search:** Because our unweighted *Arterial Boulevards* merge dropped the side-street exclusion principle (§4.2, Figure 7), we can separate the effect of that missing rule from the test-time search bottleneck by appending the omitted principle from the Run 1 candidate pool (*Slot 9*: *"Vehicles must avoid minor intermediate cross streets except the destination street"*; Table 9 in Appendix E.2). Restoring this single principle raises conflict-route accuracy by +12.33 pp (65.67% → 78.00%, and 78.50% → 87.17% overall across all 1,200 routes), while the remaining 11.67 pp gap to Composed GEPA on conflict routes (78.00% vs. 89.67%) isolates the cost of unguided multi-block search at test time.

[1]Table 8 in Appendix E.2 provides more details on the $P = 5$ principles.

F.2 Distilling Prompts to a Smaller Model

Whereas composition holds the student fixed and increases the difficulty of test-time search, *prompt distillation* holds the environment fixed and reduces the student's test-time reasoning capacity: a frontier teacher (`Gemini 3.5 Flash`) learns a prompt during sleep-time for a smaller student (`Gemma 4 12B IT`; Team et al. (2026)). We evaluate two regimes across all five environments (Table 11): (1) **direct teacher transfer**, where we prepend the 10-slot ensembled constitutions and GEPA prompts learned on `Gemini 3.5 Flash` (§4.2) directly to the frozen 12B student, and (2) **student-in-the-loop distillation**, where `Gemma 4 12B` executes the training rollouts (Stage 1) on 50 training queries across 20 steps ($P = 15$, 5 runs) while the `Gemini 3.5 Flash` teacher audits the student's errors and proposes rule updates (Stages 2–3), and we then apply our constitutional merge procedure.

Table 11: Held-out test accuracy (%, $N = 400$ per task) for a frozen `Gemma 4 12B IT` student under student baselines, direct transfer of teacher prompts (`Flash` → `12B`), and student-in-the-loop distillation (`12B` + `Flash`).

	`Gemma 4 12B` Baselines		Direct Teacher Transfer (`Flash` → `12B`)		Student-in-the-Loop (`12B` + `Flash`)
Environment	Zero-Shot	10-Shot (5 seeds)	Ensembled Constitution	GEPA (5 seeds)	**Ensembled Constitution**
Neo-Manhattan	48.25	45.00 ± 2.33	89.50	80.50 ± 11.25 (89.50)	**92.50**
No Left Turns	28.00	34.45 ± 3.41	40.25	68.70 ± 9.47 (**76.25**)	**75.75**
Kepler to Newton	50.75	47.05 ± 2.48	79.00	82.80 ± 9.32 (**90.75**)	**84.50**
Arterial Boulevards	56.25	39.50 ± 9.75	65.50	**85.00 ± 6.75** (**89.50**)	72.25
Prohibited Avenues	15.50	40.50 ± 12.99	23.25	**77.45 ± 5.02** (**83.50**)	46.50
5-Environment Mean	39.75	41.30	59.50	**78.89** (**85.90**)	74.30

- **Direct teacher transfer works for 1-step checks, but fails when rules require spatial inversion:** Directly transferring the teacher's ensembled constitution to `Gemma 4 12B` trails transferred GEPA on average (59.50% vs. 78.89%; Table 11), with the gap being most apparent in the three Manhattan environments. On *Neo-Manhattan* (89.50%) and *Kepler to Newton* (79.00%), each rule specifies a direct 1-step check on (x, y) and transfers cleanly. On the Manhattan tasks, however, teacher rules state negative exclusions (*"Do not travel South on Broadway, Park Ave, or Amsterdam Ave"*). While `Gemini 3.5 Flash` can invert a negative ban in chain-of-thought by recalling the East-to-West avenue order to pick an adjacent legal corridor, `Gemma 4 12B` cannot perform that spatial inversion without GEPA's street directory and cached detour steps (Appendix E.3) and drops to 23.25% on *Prohibited Avenues*.
- **Student-in-the-loop distillation recovers local turn accuracy by writing explicit transitions into the slots:** Putting `Gemma 4 12B` inside the induction loop raises the ensembled constitution from 59.50% to 74.30% and outperforms transferred GEPA on three of five environments (Table 11). When the teacher inspects the 12B student's Stage 1 rollout errors in Stages 2–3, it replaces bare bans with explicit transition tables that the teacher did not need: on *No Left Turns*, the student-in-the-loop constitution spells out each forbidden compass pair (*North→West, West→South, South→East, East→North*) and the avenue order, which lifts accuracy from 40.25% to 75.75% (Appendix E.3). Because $P = 15$ single-clause rules still cannot store a full Manhattan street directory, however, the 12B student remains behind GEPA on *Arterial Boulevards* and *Prohibited Avenues*, where multi-block corridor planning exceeds 12B test-time search.