PROMPT-ATTRIBUTABLE FEASIBILITY VIOLATIONS OF MEDIUM-WEIGHT LLM-BASED CONTROL OF A CASCADED FIVE-RESERVOIR SYSTEM

Anonymous authors
Paper under double-blind review

ABSTRACT

When an LLM is tasked to control a complex real-world problem, a physically infeasible control action is usually projected onto the feasible operating range by the environment, allowing the simulation to continue. Yet, this can mask repeated constraint violations behind strong aggregate performance. Here, we investigate prompt-attributable feasibility failures in a twenty-year, non-resettable, multi-reservoir hydrological testbed, using two LLM families – DeepSeek-R1-Distill and Gemma 4 – as controllers making sequential operational decisions under physical constraints. Empirically, despite achieving near-optimal aggregate performance, the LLM agents violate physical constraints in 10 to 28% of decisions. We show that these violations are systematic rather than stochastic and arise from two reproducible, prompt-attributable reading failures. The first, constraint anchoring, occurs when a field specifies a maximum output limit while explicitly excluding an incoming quantity. DeepSeek-R1-Distill interprets this limit as total availability and systematically under-commits, whereas Gemma 4 does not. The second, forecast presence bias, occurs when a predictive field is introduced into an observation whose other fields describe currently available quantities, causing agents to spend the predicted quantity as if it was immediately available. The effect is driven by forecast presence, rather than magnitude: scaling the forecast to an unambiguous underestimate still triggers violation rates similar to the unscaled forecast case, while removing the forecast drops up to 87% of the violations. We investigate both mechanisms with a five-step attribution protocol – open-loop replay, single-element perturbation with the remainder byte-identical, failure-rate comparison, domain-stripped simplified replay, and token-preserved identity substitution. The results isolate structural misreading from domain reasoning. Constraint anchoring survives complete removal of domain vocabulary, and is eliminated by a single arithmetic clarification. Conversely, forecast presence bias disappears in domain-stripped simplified replay, but returns intact under identity substitution, so neither violation is semantic. Our findings therefore show that the observed feasibility violations arise not from hydrology semantics or the specific benchmark, but from a specific model family's anchoring tendency and a general disposition to remain optimistic even with an underestimated forecast.

Keywords: LLM agents · physical feasibility · closed-loop control · prompt anchoring · prompt attribution · constraint satisfaction · agentic evaluation

1 INTRODUCTION

Large language models (LLMs) are increasingly moving from conversational interfaces to the agentic control of systems (Wang et al., 2024). However, the dominant paradigm for evaluating agentic reliability employs resettable digital sandboxes and test environments. In these settings, an infeasible action is either fundamentally inexpressible or gracefully absorbed without permanent operational consequence (Rabanser et al., 2026). Because these environments can be trivially reset, they fail to capture the compounding effect of accumulating errors characteristic of continuous closed-loop execution. This creates a gap: evaluating aggregate task performance independently of strict constraint satisfaction hides whether an agent can safely operate over long horizons where every action

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.

permanently impacts the future state. In environmental control, operations often cannot simply be reset. At the same time, while traditional methods like dynamic programming guarantee constraint satisfaction by construction, deploying unconstrained agents – such as those trained via standard reinforcement learning or LLMs – requires a safety filter to project infeasible actions onto a certified safe set before execution (Yuan et al., 2022; Zhao et al., 2021). Unlike reinforcement learning agents that undergo continuous gradient-based weight updates, LLMs operate via in-context inference. Although it is possible to engineer an iterative feedback loop – where the LLM proposes an action, hits the environmental boundary, and receives error-correction prompts until it generates a feasible command – we explicitly evaluate these models as zero-shot controllers to isolate models' raw capacity to synthesize physical constraints. Here, we evaluate general-purpose open-weight LLMs as zero-shot, stateless controllers within a long-horizon physical closed loop using the Zambezi River multi-reservoir cascade (Giuliani & Castelletti, 2013). This diverse basin features both storage-dominated dams and run-of-river plants operated in a 20-year framework with monthly decisions, aimed at meeting hydroelectric demand while minimizing shortfalls and respecting physical limits. We evaluated our control framework across six model configurations, specifically open-weight models from the DeepSeek-R1-Distill and Gemma 4 families, using evolutionary multiobjective direct policy search (EMODPS) as the optimal performance benchmark (Arnold & Castelletti, 2023). We identify two systemic failures of the LLM controller, *constraint anchoring* and *forecast presence bias*, attribute them to specific prompt elements, mitigate them through targeted interventions, and assess their dependence on the hydrological domain particular to this control problem.

In summary, our main contribution is threefold. First, we introduce a comprehensive testbed for evaluating physical agentic control alongside an attribution protocol, enabling us to test whether an agentic failure depends on its semantic subject matter or on the numerical syntax of the decision. Second, we demonstrate that near-optimal aggregate metrics mask constraint violations. While our LLM controllers achieve 93.7–97.2% of the performance of the EMODPS benchmark, the LLM also outputs physically infeasible decisions in 10% to 28% of operational decisions. Through this analysis, we characterize two distinct recurring operational errors: (i) a constraint-anchoring mechanism at the run-of-river plant (exclusive to the DeepSeek family) and (ii) a forecast-induced over-commitment at the storage reservoir. Finally, we demonstrate that these structurally embedded failures can be attributed, isolated, and addressed. At the run-of-river plant, the constraint-anchoring failure can be repaired by inserting an arithmetic formula into the prompt. At the storage reservoir, we show that forecast-induced over-commitment is triggered by the mere *presence* of forecast data rather than its *magnitude*, and that this response persists even when the forecast is scaled to highly conservative levels.

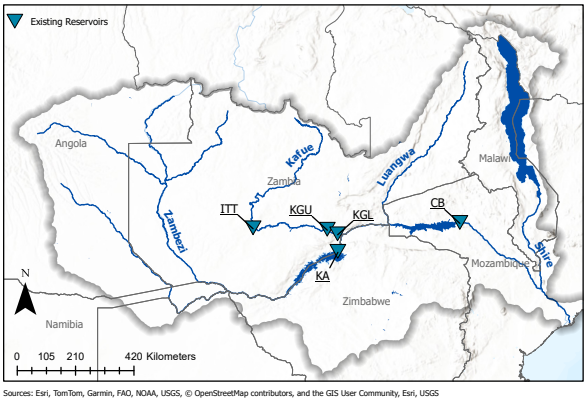
2 Related Work

The current dominant paradigm for evaluating agentic reliability relies on digital sandboxes, including multi-environment suites (Liu et al., 2024b), web tasks (Zhou et al., 2024), and software repositories (Jimenez et al., 2024). Such environments are typically resettable and reward iterative error correction (Rabanser et al., 2026). While embodied vision-language-action models demonstrate physical interaction capabilities, they are predominantly evaluated in open-loop or short-horizon settings that fail to capture the effect of accumulating errors and covariate shifts characteristic of closed-loop execution (Huo et al., 2026; Wang et al., 2026; Zou et al., 2026). In these autonomous regimes, evaluating aggregate task performance independently of constraint satisfaction (Xie et al., 2024; Qin et al., 2025) reveals a critical gap: near-optimal aggregate performance metrics can conceal substantial underlying feasibility violations. Unlike classical safe reinforcement learning (Altman, 1999; García & Fernández, 2015; Achiam et al., 2017), which can guarantee mathematical feasibility by construction, LLM controllers process constraints and observations through unstructured text susceptible to systematic biases, such as prompt anchoring and mid-context attention degradation (Takenami et al., 2025; Macmillan-Scott & Musolesi, 2024; Liu et al., 2024a)), even when utilizing chain-of-thought reasoning ((Wei et al., 2022). We evaluate LLM controllers in a long-horizon closed loop to jointly assess task performance and constraint violations. Using the Zambezi River cascade, we demonstrate that near-optimal aggregate performance masks systemic, prompt-attributable feasibility violations.

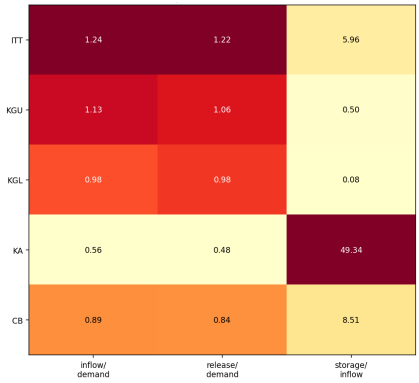
3 MATERIALS AND METHODS

3.1 CASE STUDY

We investigate the five-reservoir Zambezi River cascade, comprising Itezhi-Tezhi (ITT), Kafue Gorge Upper (KGU), Kafue Gorge Lower (KGL), Kariba (KA), and Cahora Bassa (CB), as illustrated in the map in Figure 1a. The system is simulated at a daily time step, using historical hydrological data from 1986 to 2005, together with corresponding demand series and operational constraints. Each reservoir is controlled by an independent LLM agent operating in a top-down self-interested manner, making monthly decisions. Agents do not communicate directly with one another, but downstream agents receive information on upstream release decisions. Each LLM agent configuration is run in three independent simulation runs.



(a) Map of the Zambezi River Basin showing the locations of the five existing reservoirs (ITT, KGU, KGL, KA, and CB) that form the cascade.



(b) Reservoirs characteristics: comparing average monthly ratios of inflow/release to demand, and storage to inflow (EMODPS as benchmark).

Figure 1: Overview of the Zambezi River basin case study.

The simulation environment relies on a calibrated and previously validated Zambezi model used here without modification, maintaining identical network topology, storage–elevation–release relationships, historical inflows, hydropower demand profiles and operational constraints((Giuliani & Castelletti, 2013; Arnold & Castelletti, 2023). Only the control layer is modified, with LLM agents replacing the EMODPS benchmark control policies.

The LLM agents make their monthly operational decisions by outputting a target allocation percentage $p \in [0, 100]$ representing the fraction of the current month's release compared to the hydropower goal set. To ensure physical feasibility, the environment converts this percentage into an intended release volume $R_{\text{target}} \in \mathbb{R}_{\geq 0}$ based on monthly demand $D \in \mathbb{R}_{\geq 0}$: $R_{\text{target}} = \frac{p}{100} \cdot D$. Per each reservoir, let $S \in [S_{\min}, S_{\max}] \subset \mathbb{R}_{\geq 0}$ denote the current reservoir storage volume (in m^3), where $S_{\min}$ and $S_{\max}$ denote the minimum admissible and maximum physical storage levels, respectively, and let $Q_{\max}, Q_{\min} : [S_{\min}, S_{\max}] \to \mathbb{R}_{\geq 0}$ denote the maximum and minimum admissible release rates as functions of storage. Although the LLM acts as the operator, a physical wrapper intercepts the agent's target release and enforces strict operational safety limits (as also done in the EMODPS benchmark). Given the current storage S and the incoming inflow I, this wrapper clips the agent's requested release R_{target} to lie within $[Q_{\min}(S, I), Q_{\max}(S, I)]$, thereby preventing unfeasible storage depletion or overflow. The actual release R_{actual} is thus computed as:

$$R_{\text{actual}} = \min\left(Q_{\max}(S, I), \max\left(Q_{\min}(S, I), R_{\text{target}}\right)\right). \tag{1}$$

The Zambezi cascade incorporates structurally heterogeneous reservoir types, as evidenced by the structural fingerprint in Figure 1b. For example, KA is heavily storage-dominated (with a high storage-to-inflow ratio), whereas KGL operates under near run-of-river conditions (with a near-zero storage-to-inflow ratio). This diversity allows us to evaluate multi-agent dynamics under complex heterogeneous physical interactions. Furthermore, the three-month routing delay between ITT and

KGU across the Kafue Flats introduces lagged physical interactions, testing the controllers' ability to manage delayed downstream effects. For these reasons, we consider the Zambezi basin as a representative case study to understand how multi-agent LLM controllers plan and perform within a larger cascading system, where each operational decision directly impacts downstream infrastructure operations.

Finally, aggregate system performance is benchmarked against an independently validated reference policy rather than an artificially constructed internal reward function (Arnold & Castelletti, 2023).

3.2 LLM Agent Interface

The control framework is stateless: each decision is an independent, single-turn exchange with no conversational memory carried over. Each agent is initialized with a fixed system message that defines the identity, position of the cascade, and objectives of the reservoir, including meeting the demand for hydropower, minimizing shortfalls, and respecting structural limits (Table 1; full text available in Appendix B). At each turn, the agent receives the current environmental state and must return a formatted JSON object, containing the target release allocation (0-100%), a reasoning string, and a dictionary with the importance ranking, scoring the influence of each observation field (Table 1, second and third blocks) on its decision from 0 to 4 (see Appendix B for more details).

Table 1: State representation supplied to agents and dynamic observations (volumes in m^3).

Field	Description	Scope
Static & Contextual Block (repeated at every decision – System Prompt)		
Task definition	Objective to minimize shortfalls and respect limits. Shortfall = demand × (100% − allocation percentage).	All
Storage bounds	Maximum and minimum operable storage.	All
Demand	Water-year total and all twelve monthly values.	All
Historical inflow	Average inflow by month (available after the first iteration).	All
Distractor	Irrelevant string inserted to set the baseline of the importance ranking to zero.	All
Per-Decision State		
Month index	Current month of the water year.	All
Current storage	Reservoir storage volume at decision time.	All
Max safe release	Maximum physical release, *explicitly excluding* incoming inflow.	All
Min safe release	Minimum physical release, *explicitly excluding* incoming inflow.	All
Min environmental release	Minimum ecological flow standard.	ITT
Upstream inflow	Release committed by upstream agents for the current month.	All
Monthly demand	Hydropower demand volume for the current month.	All
Previous allocation	The agent's own allocation percentage from the previous step.	All
Seasonal Forecast Block (1993 onward)		
Local ensemble mean	GloFAS 1-month mean inflow at the local gauge.	ITT, KA, CB
Basin ensemble mean	GloFAS 1-month mean inflow for the whole basin.	All
Local low-flow prob.	GloFAS 3-month probability of lowest-quintile flow.	ITT, KA, CB
Basin low-flow prob.	GloFAS 3-month probability of lowest-quintile flow, whole basin.	All

Models We evaluate our framework using open-weight models from two families: DeepSeek-R1-Distill 14B and 32B parameters (Guo et al., 2025) and Gemma 4 12B and 31B parameters; (Gemma Team et al., 2024; 2026). We utilize self-hosted models and targeted 10B–30B models to enable execution on a single NVIDIA H100 80GB GPU—ensuring broad reproducibility. DeepSeek models are evaluated with and without reasoning enabled. Gemma 4 is evaluated solely without reasoning, as initial trials demonstrated that its reasoning configuration stalled indefinitely, terminating within the first simulated months despite the use of retry wrappers (further detailed in Appendix C). All models employ a decoding temperature of 0.1 to increase the deterministic control (Eagal et al., 2025).

3.3 Feasibility and Error Taxonomy

To systematically evaluate the physical realism of each LLM agents' decisions, we classify operational violations into three categories. *Under-release (over-commitment)* occurs when the agent allocates a water volume exceeding physical availability, causing delivery to fall short of intent. *Over-release (under-commitment)* occurs when the agent commits less than the mandatory minimum, forcing the system to release more than intended. *Natural over-release* occurs when water

delivery exceeds the intended target due to unavoidable physical constraints–such as forced spills or turbinable flows surpassing demand–independent from the agent's actions. These last events (which we exclude from the violation metrics) occur for 11% to 13% of all simulated operating months. They are predominantly concentrated at ITT, KGU, and KGL, as these locations periodically experience inflows that significantly exceed both current demand and available storage capacity.

3.4 The Intervention Protocol

Initial closed-loop evaluations revealed that operational violations were highly localized at two structurally distinct nodes: KGL and KA (details in Section 4). To diagnose the drivers behind these localized violations, we compile a dataset of all physically unfeasible decisions (over-release for KGL and under-release for KA) alongside a control group of feasible decisions, which are randomly sampled to exactly match the monthly distribution of the failures, up to an equal total count or the maximum available. Our protocol systematically compares the verbatim deployed prompt (Arm A) against a targeted, minimal intervention (Arm B), with and without domain intervention (by removing the hydrology context). To balance robust error reproduction with computational feasibility, we regenerate each query three times. The attribution protocol is composed of five steps:

1. **Replay the decision:** Regenerate the query in open-loop three times to verify baseline error reproduction.
2. **Change the potential cause:** Adjust a single candidate prompt element while holding the remainder byte-identical, and replay the decision three times.
3. **Track the violation:** Compare violation rates between the baseline and the modified prompt to establish prompt-component implication.
4. **Repeat domain-free:** Reformulate the problem as a heavily simplified abstract resource-allocation task to test the failure outside of the hydrological context (Appendix D).
5. **Identity substitution:** If the condensed formulation resolves the violation, isolate whether the fix was due to the removal of the hydrology terms or simply the shortening of the prompt. We test this by swapping in generic vocabulary while retaining the exact original prompt length and syntax (Appendix D).

Applying this framework to our specific vulnerability nodes, Arm B targets each hypothesized failure mechanism. At KGL, Arm B introduces an explicit arithmetic formula to aid availability deduction. At KA, Arm B completely removes the seasonal forecast field. Furthermore, because KA's baseline forecasts systematically overestimated inflows (Hirpa et al., 2018), we must untangle whether the LLM over-commits due to the forecast's sheer *magnitude* or its mere *presence*. To isolate this, we introduce three additional Step 2 variations applied exclusively at KA: *Arm B-KA3* and *Arm B-KA2* downscale the originally supplied forecast by a fixed 0.5x and 0.1x factor respectively to test magnitude, while *Arm B-KA1* explicitly states that forecasts are absent to test presence.

4 Results and Discussion

4.1 Aggregate performance and regime-based evaluation

Evaluated strictly on aggregate hydropower generation relative to the optimal EMODPS reference policy, the LLM controllers recover between 93.7% and 97.2% of the twenty-year cascade potential (Figure 2a). The configurations that achieve the highest recovery rates are the non-reasoning models–specifically the 32B and 14B DeepSeek variants (recovering 97.2% and 96.9%, respectively)– however, this apparent efficiency is misleading, as the non-reasoning variants simultaneously have the highest rates of the under- and over-commitment violations.

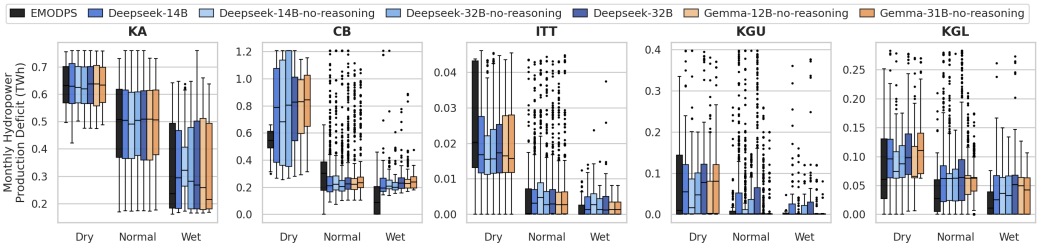
To evaluate the models' robustness to varying hydrologic conditions, we stratify the 1986–2005 evaluation period into dry, normal, and wet hydrological regimes (the temporal mapping of these regimes is detailed in Appendix A). This classification is derived by calculating the z-score of the sum of natural annual inflows at the KA and ITT reservoirs and mapping each simulated year to its corresponding climatic severity according to the amount of standard deviations from the historical mean. Analyzing deficit volumes relative to EMODPS (Figure 2c), LLM controllers generally

follow the optimal benchmark more closely during normal years, while their performance slightly degrades during extreme (dry or wet) conditions. Analyzing in detail the performance across varying hydrological regimes provides insight into the implicit policies adopted by the LLM controllers. At CB, during normal periods, all LLM configurations exhibit higher production levels (lower deficits) compared to the EMODPS solution, but suffer amplified deficits during wet and dry years. This behavior suggests that the LLMs maintain a lower baseline storage level than the EMODPS reference, which limits dry-year buffering capacity in favor of short-term generation. Conversely, at ITT, the LLM controllers' performance suffers specifically during dry periods, indicating the adoption of a more conservative, higher-storage policy that buffers against dry periods. It is interesting to note that all tested configurations reproduce the same overall storage strategy. This consistency suggests that the preference for short-term generation over long-term buffering is an intrinsic behavioral bias of zero-shot LLMs, rather than a mere artifact of the prompt.



(a) System-wide performance and violation rates.

(b) The distribution of over-release, under-release, and natural over-release actions across the five individual dams.



(c) Hydropower deficit volume compared to EMODPS across dry, normal, and wet regimes (lower is better).

Figure 2: Aggregate performance, regime classification, and distribution of physical feasibility violations across the Zambezi River Basin cascade.

4.1.1 Distribution of feasibility violations

Beyond illustrating the aggregate under- and over-commitment rates, Figure 2b decomposes these violations across the individual dams of the reservoir cascade and by model configuration. It is crucial to distinguish agent-driven errors from *natural over-releases*. The latter case occurs when the actual delivery exceeds the model's intended target, such as forced spills (or higher turbinable water volume) driven by inflows that substantially outpace both local demand and available storage. These events are tracked but categorically excluded from the agent's intentional violation count, if the agent has already picked the best decision possible when more water volume than the hydropower demand arrives (100% release as action). Looking closely at the distribution of the LLM agent's errors, we can observe a structural consistency:

- *Systematic under-release:* KA consistently fails by overcommitting more water than is available, forcing an actual under-release.
- *Systematic Deepseek over-release:* KGL acts as the exclusive site of systematic over-release across the cascade, but only for the Deepseek configurations.

The remaining dams are significantly less prone to agent-driven errors, though CB displays a moderate, persistent tendency to under-release across both model families. ITT and KGU remain remarkably robust to intentional agent errors, maintaining near-zero intentional violation rates across all tested models.

Although systematic over-releases at KGL do not occur in the Gemma models, we retain this violation mode because it remains rigidly homogeneous across all four DeepSeek configurations. Since our primary objective is to investigate the generalized interaction between language model generation and physical constraints, we strictly filter for errors that have cross-configurational consistency, as these are most likely to expose underlying architectural limitations rather than stochastic noise. Consequently, our analysis and intervention protocol focuses exclusively on KA and KGL, also considering the intrinsic vulnerability described in Section 3.1. While other localized anomalies emerge, such as the moderate under-releases at CB, these patterns lack the severe, structural homogeneity of the primary violations identified at KA and KGL. In fact, CB anomalies are driven by actual physical scarcity, with the vast majority occurring during drought years, where commitments simply outrun what a depleted system can deliver.

4.2 Constraint anchoring at Kafue Gorge Lower

KGL represents the cascade's most constrained operational storage profile, characterized by a near-zero storage-to-inflow ratio (Figure 1b): it has 51.9 Mm^3 of operable storage while receiving monthly inflows an order of magnitude larger ($\approx$603 Mm^3 in a representative month). It is the sole site exhibiting systemic, non-stochastic *over-release* failures. However, this failure mode is exclusively DeepSeek-specific: all four DeepSeek configurations over-release in 17%–32% of issued decisions, whereas both Gemma configurations maintain near-zero failure rates ($\leq$0.6%). Here we diagnose the underlying causes, following the intervention protocol introduced in Section 3.4.

Step 1 – Replay: Verifying Violation Reproducibility. Before attributing the failure to specific prompt elements, we must verify that the error is systemic and not stochastic noise; thus we replay the exact simulated decisions in an independent, open-loop setting. The violation replicates consistently (see Supplementary Table 6). The "flagged" prompts—those that caused over-releases during the simulation—fail again 30.3–61.9% of the time when replayed. Interestingly, even the control decisions (prompts from months where the simulation originally succeeded) still produce over-releases in 6.7–24.7% of the open-loop replays. The control decisions are still DeepSeek decisions at KGL, and they carry the same vulnerability in a milder form: they were simply not severe enough during those months. Thus, the entire DeepSeek family reads this prompt unreliably, with flagged decisions representing only the severe tail of a continuous distribution.

Two readings of one number. The prompt states a maximum safe release, specifying multiple times that it excludes incoming inflow (the complete prompt is reported in the Appendix B). The outcome therefore admits two different readings: let $R_{\max} \in \mathbb{R}_{\geq 0}$ denote the maximum release stated in the prompt, excluding the incoming inflow; $I \in \mathbb{R}_{\geq 0}$ the incoming inflow; and $D \in \mathbb{R}_{\geq 0}$ the monthly demand (all in m^3):

$$\text{anchor} = 100 \cdot \frac{R_{\max}}{D}, \qquad \text{correct} = \min\left(100,\ 100 \cdot \frac{R_{\max} + I}{D}\right). \tag{2}$$

The *anchor* reading erroneously treats $R_{\max}$ as the agent's total available water for the month, while the *correct* reading integrates the incoming inflow I before normalizing against demand D. Moreover, since the instruction to consider incoming inflow is structurally distributed across both the initial system message[1] and the dynamic observations, this persistent anchoring confirms a structural parsing limitation rather than spatial contextual memory loss (Hsieh et al., 2024).

Steps 2–3 – Intervention. To test this causal hypothesis, we insert a single disambiguating line: *"Total available water this month = (current storage − minimum storage) + incoming inflow"*, holding the rest of the prompt fixed. The intervention leads to consistent improvement (as detailed in Supplementary Table 6): over-release on flagged decisions drops by 25.2–36.8% across all configurations, including a 6.0$\times$ reduction (30.3% to 5.0%) in the 32B-reasoning model.

Step 4 – Domain-Free Replay. Re-posing the identical decisions stripped of hydrological context (Appendix D) confirms the failure is structural rather than semantic. The baseline failure persists (47.9–77.4% over-release; see Supplementary Table 6). This structural persistence proves the misreading is driven by the numerical syntax, and since the error is absent in Gemma configurations,

[1] Extracts of the DeepSeek reasoning traces at KGL are reported in Appendix C.3.

it suggests a DeepSeek-specific anchoring problem. However, the intervention's efficacy is not completely uniform. All reasoning models consistently improve, by dropping the failure rate on average of 31.8%. While non-reasoning models have an average drop of 24.7% apart from the Deepseek-32B non-reasoning model that continues to misinterpret the explicit formula, causing its over-release rates to increase by 7.6%. This behavior means that interventions do not uniformly generalize, and because all models processed byte-identical prompts, the discrepancy points to differences in base architecture, alignment (e.g., RLHF), or distillation. DeepSeek-R1-Distill's training to emulate larger reasoning trajectories (Guo et al., 2025) may inadvertently cause rigid anchoring to explicit numerical limits—a known risk where alignment processes over-optimize for specific safety constraints (Gao et al., 2023). Because the simplified task in this step exhibits the same misreading mechanism, proceeding to step 5 is unnecessary (see Section 3.4).

4.3 Forecast-induced over-commitment at Kariba

KA is the cascade's opposite pole from KGL: 77 Gm^3 of operable storage, being a storage-dominated extreme (Figure 1b). Its failure runs in the opposite direction: KA never over-releases in any configuration. Rather, it often *under*-releases, i.e., it commits to higher releases than the system can physically deliver. The candidate cause is a single conditional field: from 1993 the observation carries a seasonal inflow forecast, and the problem is architecture-agnostic.

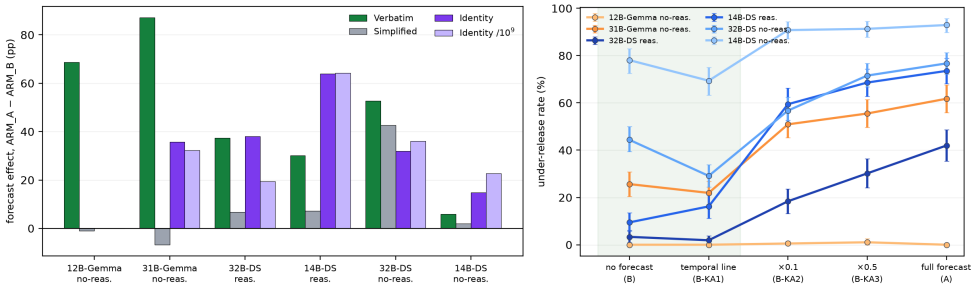
Step 1 – Replay. Flagged decisions are re-posed open-loop three times against calendar-matched controls. The failure reproduces at 47.6–92.0% on cases against 18.4–39.5% on controls. Similarly to KGL, the controls are not a clean population: unflagged KA decisions still over-commit in roughly a fifth to two fifths of replays (see full results in Supplementary Table 7). The correct reading is again a continuum – the flagged decisions are where a pervasive over-commitment tendency crosses into infeasibility – and the intervention below acts on both groups. The forecast predicts the total inflow expected over the course of the current month, yet it sits in an observation whose other fields describe guaranteed quantities available at the exact moment of decision. The natural hypothesis is that the agent treats this predicted, time-distributed volume as instantaneously fungible with present storage, committing to releases based on water that has not yet physically arrived.

Steps 2–3 – Intervention. Removing the forecast block entirely (Arm B) while holding every other line fixed produces the largest single effect (Supplementary Table 7): under-release falls by 5.9 to 87.1% in every configuration, in both cases and controls. The only configuration where removal barely helps, 14B-deepseek-no-reasoning (-5.9 %), is the one whose baseline without forecasts is already 85.5%: its over-commitment is largely forecast-independent.

Forecast presence versus its magnitude. The forecast handed to the Kariba agent is a GloFAS seasonal ensemble mean for the Victoria Falls gauge, known to carry substantial regional bias over the Zambezi (Hirpa et al., 2018) with a mean ratio of close to $4\times$ compared to the real incoming inflow. This suggests that the model adds an inflated number, so shrinking the number should roughly shrink the failure in proportion. Arm B-KA2 tests it by scaling the supplied forecast to $0.1\times$ – given the $\approx 4\times$ bias, about $0.4\times$ realized inflow on average, and so nominally a *conservative* figure. It is important to note that in this case the $0.1\times$ forecast is higher than the realized inflow just about 10% of the time (Appendix E.2). It does not behave like a pure scaling effect: under-release stays well above the no-forecast baseline in every susceptible configuration (Supplementary Table 7). For instance, in the 31B non-reasoning Gemma model, the failure rate on case decisions drops drastically from 90.8% to 3.7% when the forecast is removed entirely, yet rebounds to 41.5% when presented with the highly conservative $0.1\times$ forecast. A forecast that should now be an underestimate still drives over-commitment, so forecast presence carries much of the effect.

Step 4 – Domain-Free Replay: The Failure Itself Disappears. In the simplified domain-free rendering, the forecast effect largely vanishes in five of six configurations (Figure 3a). Taken in isolation, this suggests a strictly domain-bound failure. However, this abstraction introduces a severe methodological confounding feature: it simultaneously strips hydrological semantics, truncates prompt length, and removes structural complexity. Consequently, the nullified effect cannot be uniquely attributed to the loss of domain. To explicitly disambiguate these variables, we proceed with Step 5.

Step 5 – Identity Substitution: The Failure Returns. Rewriting only the domain vocabulary brings the failure back to verbatim-like levels (Arm A of 42.0–92.9% in five of six configurations) and brings the forecast effect back with it: 14.8 to 64.0 pp, matching or exceeding the verbatim effect in three of them (32B-reasoning, 14B-reasoning, 14B-no-reasoning; Figure 3a). The failure is *not* domain-bound for five of six configurations. 12B-gemma is an exception: its effect is 0.0% under identity substitution, and every arm sits at $\leq$1.0%. Furthermore, analyzing the entire intervention layer (Figure 3b), it suggests that the presence of the forecast has more impact than the magnitude itself. Arm B-KA1, which replaces the forecast block with a single stated temporal rule rather than removing it, tracks closely (or even decreases) the Arm B's no-forecast baseline in five of six configurations (Supplementary Table 9). Transitioning from this purely temporal baseline (Arm B-KA1) to a highly conservative forecast ($\times 0.1$, Arm B-KA2, representing on average $0.4\times$ of the realized inflow) triggers a dramatic spike in the failure rate for the five susceptible models. However, scaling the forecast to much larger, threatening volumes – $\times 0.5$ (Arm B- KA3, $2\times$ the realized inflow) and the full forecast (Arm A, $4\times$ the realized inflow) – produces comparatively modest incremental increases. This suggests that the presence of a forecast triggers a non-negligible baseline violation, with increased severity based on the magnitude.



(a) Change in under-release rate when the forecast is removed (Arm A − Arm B, %), per prompt rendering: verbatim, domain-free simplified, identity-substituted, and identity with volumes divided by 10^9.

(b) Under-release rate across forecast intensities in the identity-substituted prompt, from no forecast (Arm B) to the full forecast (Arm A); the shaded region has no forecast values.

Figure 3: Attribution of forecast-induced over-commitment at the Kariba reservoir, on flagged case decisions. Arms are defined in Section 3.4.

5 Conclusion

Where a physical environment silently clips infeasible commands, aggregate objective recovery brings limited information for operational safety. Although all evaluated LLM controllers achieved acceptable top-line generation metrics in controlling the multi-reservoir cascade of the Zambezi River, they committed physically infeasible actions in 10% to 28% of their decisions. Using our intervention protocol, we isolated two systemic, prompt-driven violations. At the run-of-river extreme (Kafue Gorge Lower), systemic over-release stems from a Deepseek-specific anchoring tendency. At the storage-dominated extreme (Kariba), systemic over-commitment is driven by the mere presence of forecast data, persisting even when the forecast is heavily scaled down. Beyond this specific hydrological cascade, these vulnerabilities carry critical implications for autonomous LLM control. Unlike traditional safe reinforcement learning, unconstrained LLMs process constraints as unstructured text, leaving them vulnerable to syntactic illusions. The ability of prompt anchoring and forecast biases to systematically bypass explicit safety limits demonstrates that aggregate performance in resettable sandboxes is a dangerous proxy for operational safety. Evaluating LLM agents for generalized physical control must therefore transition from pure performance benchmarking toward rigorous, long-horizon attribution protocols that test how models synthesize interdependent constraints over time. While these findings are currently bounded to a five-reservoir system with self-interested coordination policies, future work should explore collaborative multi-objective setups to better understand the structural vulnerabilities of LLM controllers in increasingly complex settings.

REPRODUCIBILITY STATEMENT

All controllers are open-weight models (DeepSeek-R1-Distill 14B/32B and Gemma 4 12B/31B) self-hosted on multiple NVIDIA H100 80GB GPUs, with decoding temperature 0.1 and three independent simulation runs per configuration (Section 3). The simulation environment is the previously validated Zambezi model (Giuliani & Castelletti, 2013; Arnold & Castelletti, 2023), used without modification over 1986–2005 with GloFAS seasonal forecasts from 1993 onward. The verbatim prompts, including the domain-free and identity-substituted renderings, are given in Appendices B and D; the replay and intervention protocol, including case and control selection and the three regenerations per query, is specified in Section 3.4. Source code[2].

ETHICS STATEMENT

This study involves no human subjects, personal data or user-generated content. No LLM output was connected to real reservoir operations. Our findings argue against deploying zero-shot LLM controllers on physical infrastructure without an independent safety layer, and we report violations that environmental clipping would otherwise hide.

AI USE STATEMENT

LLMs are the object of study in this paper: DeepSeek-R1-Distill and Gemma 4 act as the reservoir controllers, and their outputs are analyzed as reported. Beyond this, the authors used AI tools for three purposes. For writing, Claude (Anthropic) was used to polish the text and captions and to check cross-references and reported numbers. For retrieval and discovery, Elicit and Consensus were used to find related work, and the authors read all cited papers themselves. For research execution, Gemini and Claude were used to assist with coding and drafting and to brainstorm implementation options. The study design, attribution protocol, experiments, interpretation of results and all claims are the authors' own, and the authors take full responsibility for the paper.

REFERENCES

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International Conference on Machine Learning (ICML)*, pp. 22–31, 2017.

Eitan Altman. *Constrained Markov Decision Processes*. Chapman and Hall/CRC, 1999.

Wyatt Arnold and Andrea Castelletti. Operations eclipse sequencing in multipurpose dam planning. *Earth's Future*, 11(5), 2023. doi: 10.1029/2022EF003186.

Azeeza Eagal, Kathryn T Stolee, and John-Paul Ore. Analyzing the dependability of large language models for code clone generation. *Journal of Systems and Software*, 230:112548, 2025.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International conference on machine learning*, pp. 10835–10866. PMLR, 2023.

Javier García and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(42):1437–1480, 2015.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

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.

Matteo Giuliani and Andrea Castelletti. Assessing the value of cooperation and information exchange in large water resources systems by agent-based optimization. *Water Resources Research*, 49(7):3912–3926, 2013. doi: 10.1002/wrcr.20287.

[2]Anonymous Github Repository: `https://github.com/anonymousdelta18-commits/zambezi-llm-feasibility-anon`

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Feyera A Hirpa, Peter Salamon, Hylke E Beck, Valerio Lorini, Lorenzo Alfieri, Ervin Zsoter, and Simon J Dadson. Calibration of the global flood awareness system (glofas) using daily streamflow data. *Journal of Hydrology*, 566:595–606, 2018.

Cheng-Yu Hsieh, Yung-Sung Chuang, Chun-Liang Li, Zifeng Wang, Long Le, Abhishek Kumar, James Glass, Alexander Ratner, Chen-Yu Lee, Ranjay Krishna, et al. Found in the middle: Calibrating positional attention bias improves long context utilization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14982–14995, 2024.

Dongjie Huo, Haoyun Liu, Guoqing Liu, Dekang Qi, Zhiming Sun, Maoguo Gao, Jianxin He, Yandan Yang, Xinyuan Chang, Feng Xiong, et al. Abot-claw: A foundation for persistent, cooperative, and self-evolving robotic agents. *arXiv preprint arXiv:2604.10096*, 2026.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *International Conference on Learning Representations (ICLR)*, 2024.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024a. doi: 10.1162/tacl_a_00638.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. Agentbench: Evaluating LLMs as agents. In *International Conference on Learning Representations (ICLR)*, 2024b.

Olivia Macmillan-Scott and Mirco Musolesi. (ir)rationality and cognitive biases in large language models. *Royal Society Open Science*, 11(6):240255, 2024. doi: 10.1098/rsos.240255.

Tian Qin, Felix Bai, Ting-Yao Hu, Raviteja Vemulapalli, Hema Swetha Koppula, Zhiyang Xu, Bowen Jin, Mert Cemri, Jiarui Lu, Zirui Wang, et al. Compass: Benchmarking constrained optimization in llm agents. *arXiv preprint arXiv:2510.07043*, 2025.

Stephan Rabanser, Sayash Kapoor, Peter Kirgis, Kangheng Liu, Saiteja Utpala, and Arvind Narayanan. Towards a science of ai agent reliability. *arXiv preprint arXiv:2602.16666*, 2026.

Yoshiki Takenami, Yin Jou Huang, Yugo Murawaki, and Chenhui Chu. How does cognitive bias affect large language models? a case study on the anchoring effect in price negotiation simulations. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 4481–4498, 2025.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. A survey on large language model based autonomous agents. *Frontiers of computer science*, 18(6):186345, 2024.

Qiuyue Wang, Mingsheng Li, Jian Guan, Jinhui Ye, Sicheng Xie, Yitao Liu, Junhao Chen, Zhixuan Liang, Jie Zhang, Xintong Hu, et al. Qwen-vla: Unifying vision-language-action modeling across tasks, environments, and robot embodiments. *arXiv preprint arXiv:2605.30280*, 2026.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 24824–24837, 2022.

Jian Xie, Kai Zhang, Jiangjie Chen, Tinghui Zhu, Renze Lou, Yuandong Tian, Yanghua Xiao, and Yu Su. TravelPlanner: A benchmark for real-world planning with language agents. In *International Conference on Machine Learning (ICML)*, 2024.

Z. Yuan, A. W. Hall, S. Zhou, L. Brunke, M. Greeff, J. Panerati, and A. P. Schoellig. Safe-control-gym: A unified benchmark suite for safe learning-based control and reinforcement learning in robotics. *IEEE Robotics and Automation Letters*, 7:11142–11149, 2022. doi: 10.1109/lra.2022.3196132.

Weiye Zhao, Tairan He, and Changliu Liu. Model-free safe control for zero-violation reinforcement learning. In *5th Annual Conference on Robot Learning*, 2021.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations (ICLR)*, 2024.

Henry Peng Zou, Wei-Chieh Huang, Yaozu Wu, Jizhou Guo, Yankai Chen, Chunyu Miao, Hoang Nguyen, Yue Zhou, Weizhi Zhang, Liancheng Fang, Hanrong Zhang, Fangxin Wang, Pengfei Zhang, Huacan Wang, Langzhou He, Yangning Li, Dongyuan Li, Renhe Jiang, Xue Liu, and Philip S. Yu. LLM-based human-agent collaboration and interaction systems: A survey. In *Findings of the Association for Computational Linguistics: ACL 2026*, 2026.