Lunar-R1: Towards Efficient Reasoning of Large Language Models for Lunar Science via Reinforcement Learning

Xin-Yu Xiao[♠,♣] **Zhixian He**[♠,♢] **Shiqi Wang**[♠] **Ye Tian**[♠] **Qianchen Xia**[♠,⋆]
[♠]Tsinghua University [♣]University of Chinese Academy of Sciences
[♢]Sun Yat-sen University [⋆]National Key Laboratory of Human Factors Engineering
EMail: qianchenxia@tsinghua.edu.cn

Abstract

We present **Lunar-R1**, an 8B-parameter reasoning model tailored for lunar science and mission operations, developed through continued pre-training, supervised fine-tuning, and reinforcement learning. To optimize the accuracy-efficiency trade-off, we introduce **Latent Difficulty Perception (LDP)**, a lightweight mechanism that leverages final-layer hidden states to estimate query difficulty and regulates Chain-of-Thought generation length via difficulty-aware reward modeling. We construct **Lunar-QA**, an expert-verified benchmark derived from authentic lunar mission materials, and further evaluate on domain-specific tasks spanning astronomy and astrodynamics. Experimental results demonstrate that LDP reduces average token consumption by 38.9% while improving reasoning accuracy. Profiling on an RK3588 NPU yields 0.54 J/token energy efficiency and 380 ms time-to-first-token latency, validating feasibility for lunar surface edge deployment.

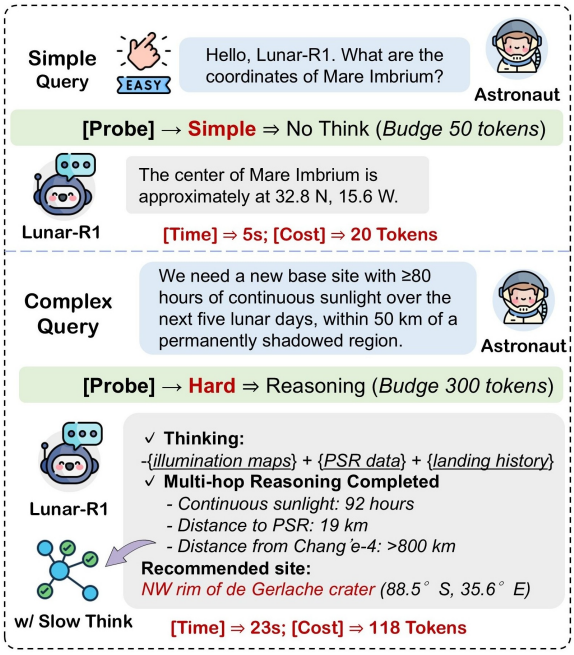

Figure 1: LDP enables the model to adaptively regulate reasoning length based on query difficulty, reducing token costs while maintaining domain reasoning accuracy.

1 Introduction

Early lunar exploration missions focused on short-duration objectives, including orbital surveys, landings, and sample returns, and provided limited capacity for sustained scientific observation (Lin et al., 2024b). In contrast, contemporary missions increasingly prioritize long-term lunar presence and semi-autonomous surface operations. NASA's Artemis program envisions persistent lunar research infrastructure (Smith et al., 2020), the European Space Agency has proposed the Moon Village initiative (Athanasopoulos, 2019), and the International Lunar Research Station targets extended autonomous operations with intermittent human involvement (Pei et al., 2020). As lunar infrastructure expands and mission workflows grow in complexity, exploration systems must deliver enhanced onboard analysis and scientific decision-support under strict computational constraints (Xiaoqiang et al., 2025; Gaddis et al., 2023; Zhang et al., 2023).

LLMs, including DeepSeek (Guo et al., 2025) and GPT-4 (Achiam et al., 2023), demonstrate strong language understanding and multi-step reasoning capabilities across domains spanning medical diagnosis (Thirunavukarasu et al., 2023), mathematical problem solving (Liu et al., 2023), and code generation (Zheng et al., 2023). These advances motivate recent studies on LLM applications in lunar mission operations (Zhao and Song, 2024; Navarro et al., 2024). However, lunar deployment introduces critical constraints: surface facilities operate with limited onboard computation and intermittent communication links (Moreira et al., 2025). Lightweight models support edge deployment but lack the reasoning depth required for complex scientific analysis. In contrast, advanced reasoning LLMs provide robust multi-step inference, yet their long-form Chain-of-Thought (CoT)

generation introduces substantial token overhead and latency. This challenge is amplified by ***overthinking***, where models generate redundant reasoning traces for low-difficulty queries (Chen et al., 2025c). Recent analyses indicate this inefficiency is partly driven by training objectives implicitly incentivizing verbose sequences (Sui et al., 2025).

To address these limitations, we introduce the **L**atent **D**ifficulty **P**erception (**LDP**) mechanism. Probing studies suggest that task difficulty can be decoded from the terminal hidden states of LLMs through linear probing (Lee et al., 2025; Shojaee et al., 2026). Motivated by this finding, LDP estimates query difficulty from model activations and uses this signal to guide reasoning-budget allocation during reinforcement learning. Specifically, we integrate the difficulty score into Group Relative Policy Optimization (GRPO) (Guo et al., 2025) through reward modeling, encouraging concise reasoning for low-difficulty queries while preserving sufficient CoT length for complex scientific tasks. This training-time control differs from fixed thinking budgets (Lin et al., 2025) and post-hoc truncation (Chen et al., 2025b) , as it enables the model to internalize difficulty-appropriate reasoning patterns. Based on this mechanism, we develop **Lunar-R1**, a reasoning model designed for lunar exploration, and evaluate its accuracy-efficiency trade-off under both domain benchmarks and edge hardware constraints. To support domain evaluation, we construct **Lunar-QA**, an expert-verified benchmark derived from authentic lunar mission materials and aligned with representative lunar exploration scenarios. We further evaluate on external benchmarks to examine cross-task generalization.

Our contributions are summarized as follows:

- **(Methodology)** We propose LDP to guide reward modeling and allocate the computational budget, ensuring that reasoning granularity matches task difficulty via linear probing.
- **(Performance)** We develop Lunar-R1 and achieve superior performance, outperforming LLMs with comparable parameter counts while optimizing accuracy and efficiency.
- **(Deployment)** We validate on-chip deployment through profiling on a consumer-level RK3588-class NPU. The model yields 0.54 J/token energy efficiency and a 380 ms time-to-first-token latency, demonstrating its feasibility for lunar surface edge deployment.

2 Related Works

LLMs for Space Exploration. Recent progress in knowledge reasoning motivates the application of LLMs to lunar and space exploration. Lunar Twins (Xiao et al., 2025) develop domain-specific LLMs and instruction data for lunar exploration. Concurrently, knowledge-graph-enhanced systems target lunar knowledge question answering (Wang et al., 2025) and scientific literature mining (Pekala et al., 2025). In parallel, multimodal LLMs and datasets designed for space exploration expand the scope of planetary perception. Space-LLaVA (Foutter et al., 2024) adapts vision-language models to Mars imagery using synthetic annotations. Furthermore, LLaVA-LE (Inal et al., 2026) constructs a lunar image-captioning and visual question-answering dataset for terrain analysis. Planetary benchmarks, including Mars-Bench (Purohit et al., 2025), LuSNAR (Liu et al., 2024), Lunar-Bench (Xiao et al., 2026), and AstroMM-Bench (Shi et al., 2025), establish evaluations for scene understanding. Despite these advances, current applications remain ground-based, as onboard resource constraints preclude direct deployment and necessitate edge-compatible reasoning.

Efficient Reasoning of LLMs. Although explicit reasoning improves problem-solving ability, it also introduces substantial inference costs. Budget-forcing methods such as s1 (Muennighoff et al., 2025) constrain or extend reasoning length under prescribed compute budgets, while Token-Budget-Aware Reasoning (Han et al., 2025) and Plan-and-Budget (Lin et al., 2025) allocate reasoning tokens according to problem complexity. SelfBudgeter (Li et al., 2025b) and BudgetThinker (Wen et al., 2025) further improve control over reasoning length by learning budget estimation or using explicit budget-control tokens during generation. Difficulty-adaptive methods provide another route to reducing redundant reasoning. DiffAdapt (Liu et al., 2025b) predicts task difficulty from hidden states and routes queries to different decoding strategies without fine-tuning the model, while DAST (Shen et al., 2025) and DIET (Chen et al., 2026) integrate difficulty-aware length control into training to reduce excessive reasoning. Truncation and self-regulation methods such as DART (Zhang et al., 2025b) and Draft-Thinking (Cao et al., 2026) also reduce redundant reasoning by learning when to stop or how to retain compact reasoning steps.

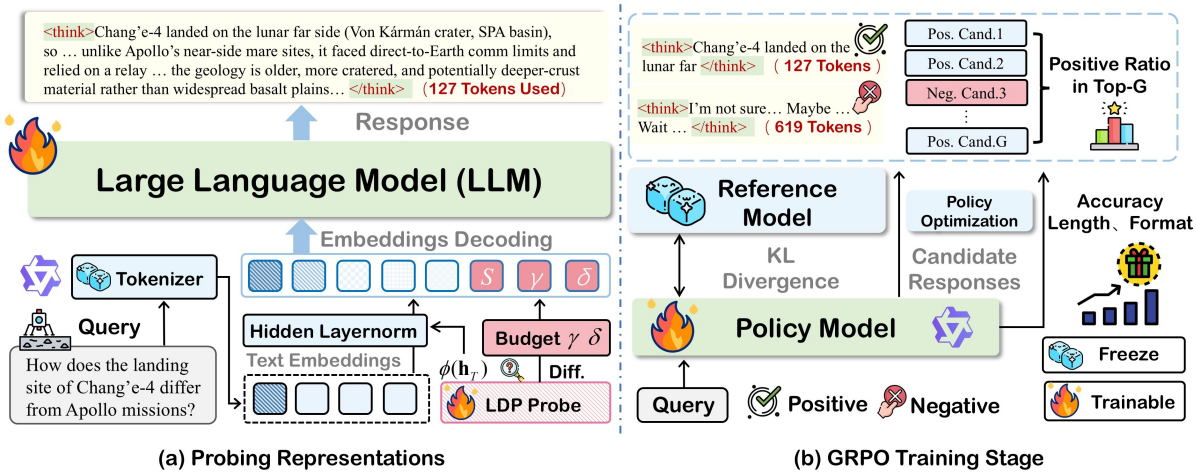

Figure 2: **Illustration of LDP.** LDP trains a lightweight probe mapping the terminal hidden state of a query to a scalar difficulty score, which establishes a reasoning budget via $\mathcal{T}_{\text{tok}}(s) = \delta + \gamma B s$. During GRPO training, the budget penalizes redundant CoT for easy queries while preserving sufficient reasoning length for complex tasks.

3 Methodology

For an input query $q = (x_1, \ldots, x_T)$, we model difficulty as a latent distribution over K ordered difficulty states. For each query in the probe-training set $\mathcal{D}p$, difficulty annotations are aggregated into a soft distribution $p(z \mid q)$, where $z \in 1, \ldots, K$ denotes an ordered difficulty state. The difficulty entropy is then computed as

$$\mathcal{H}(q) = -\sum z = 1^{K} p(z \mid q) \log p(z \mid q) \quad (1)$$

We normalize this empirical entropy into a continuous difficulty label $y(q) \in [0, 1]$:

$$y(q) = \text{clip}\left(\frac{\mathcal{H}(q) - \mathcal{H}\text{min}}{\mathcal{H}\text{max} - \mathcal{H}\text{min} + \epsilon}, 0, 1\right) \quad (2)$$

where $\mathcal{H}\text{min}$ and $\mathcal{H}\text{max}$ are the entropy bounds computed on $\mathcal{D}_p$, and ϵ ensures numerical stability. We parameterize a linear probe ϕ to project the hidden state onto a scalar score:

$$s(q) = \phi(\mathbf{h}T(q)) = \sigma(\mathbf{w}^{\top}\mathbf{h}T(q) + b) \quad (3)$$

where $\sigma(\cdot)$ constrains the prediction to $[0, 1]$. The probe is optimized via mean squared error:

$$\mathcal{L}\text{probe} = \mathbb{E}q \sim \mathcal{D}_p \left[|\phi(\mathbf{h}_T(q)) - y(q)|_2^2\right] \quad (4)$$

Post-training, ϕ is frozen. A stop-gradient operator $\text{sg}(\cdot)$ isolates the difficulty estimator from policy gradients to guarantee stability.

As shown in Figure 2, the predicted difficulty score s is converted into an adaptive reasoning budget $\mathcal{T}_{\text{tok}}(s) = \delta + \gamma B s$, where γ controls sensitivity to difficulty and δ sets the minimum budget, and B is a token-scale budget factor. For a sampled response o_i, let $c_i = \mathbb{I}(o_i \in \mathcal{Y}^*)$ indicate correctness, $f_i = \mathbb{I}_{\text{fmt}}(o_i)$ indicate format validity, and $\ell(o_i)$ denote token length. The penalty is:

$$p_i(s) = \text{ReLU}\left(\frac{\ell(o_i) - \mathcal{T}_{\text{tok}}(\text{sg}(s))}{\mathcal{T}_{\text{tok}}(\text{sg}(s)) + \epsilon}\right) \quad (5)$$

The reward function integrates correctness, format validity, and the length penalty:

$$r_i(o_i, s) = c_i + \mu f_i - \lambda c_i p_i(s) \quad (6)$$

The gate c_i ensures that the length penalty applies only to correct responses, preventing reward exploitation through trivial or incorrect outputs.

We further integrate the LDP reward into GRPO. For each query q, the policy samples G responses $\{o_1, \ldots, o_G\}$. Each response receives reward $r_i(o_i, s)$, and the group-normalized advantage is:

$$\hat{A}_i = \frac{r_i - \text{mean}(\{r_j\}_{j=1}^{G})}{\text{std}(\{r_j\}_{j=1}^{G}) + \epsilon} \quad (7)$$

The clipped surrogate objective is:

$$\mathcal{L}_i^{\text{clip}}(\theta) = \min\left(\rho_i(\theta)\hat{A}_i, \text{clip}(\rho_i(\theta), 1 \pm \epsilon_{\text{clip}})\hat{A}_i\right) \quad (8)$$

The group policy is optimized by maximizing the objective:

$$\mathcal{J}(\theta) = \mathbb{E}_q\left[\frac{1}{G}\sum_{i=1}^{G}\left(\mathcal{L}_i^{\text{clip}}(\theta) - \beta\mathbb{D}_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})\right)\right] \quad (9)$$

where β modulates the KL divergence penalty against the reference policy π_{ref} to mitigate policy drift. Extended empirical analyses substantiating the LDP are provided in Appendix C and D.

4 Experiments

4.1 Experimental Setup

Baselines. We initialize our policy model from the Qwen3-8B backbone, utilizing its hybrid reasoning capabilities via <Think> and </No_Think> tokens. We benchmark Lunar-R1 against three categories of models. GPT-5.2 (OpenAI, 2025), Gemini-3-Pro (Google DeepMind, 2025) and Claude Opus 4.6 (Anthropic, 2026) are adopted as reference upper bounds for performance. Large-scale open-weights models, including DeepSeek-R1 (Liu et al., 2025a) and Qwen3-235B (Yang et al., 2025), benchmark complex reasoning capacities. Small language models provide efficiency comparisons. All evaluations are conducted through the OpenRouter API with a 16K context window and same fixed sampling hyperparameters ($T = 0.6$, top-$p = 0.9$). Training details and benchmarks are deferred to Appendix A and B, respectively.

Benchmarks and Metrics. Lunar-QA serves as our primary evaluation testbed. To validate cross-domain generalization, we deploy a suite of external astronomical benchmarks stratified by cognitive complexity. Specifically, AstroBench (Ting et al., 2025) and Astro-QA (Li et al., 2025a) assess foundational domain knowledge. AstroReason-Bench (Wang et al., 2026) and APBench (Wu et al., 2025a) evaluate multi-step deductive reasoning. IOAA (Pinheiro et al., 2025) benchmarks olympiad-level scientific problem-solving. We quantify model performance across three strict dimensions. *Scientific Accuracy* is measured via Pass@1 exact-match rates. *Inference Efficiency* is defined by Average Token Consumption (ATC), isolating the computational overhead of correct generations. *Reasoning Quality* is adjudicated through a double-blind protocol that combines domain-expert evaluation with an LLM-as-a-Judge prompt (presented in Table 6).

Table 1: **Main results on benchmarks.** We report the average accuracy (in %, ↑) computed over 5 independent runs, and the efficiency measured by Average Token Consumption (ATC, ↓). **Bold** and underlined values denote the best and second-best results among non-proprietary models, respectively; closed-source LLMs are included only as reference baselines. The LDP probe is trained on the MMLU (Hendrycks et al., 2020) for difficulty calibration. Extended comparisons are provided in Appendix C, and a representative case study is shown in Figure 11.

Models	Scientific Accuracy (%) ↑						Inference Efficiency (ATC) ↓					
	Lunar	AstroQ	AstroB	AstroR	APB	IOAA	Lunar	AstroQ	AstroB	AstroR	APB	IOAA
Proprietary Reference Models												
GPT-5	64.8	83.2	94.2	49.7	71.2	85.5	1218	1103	969	2191	2353	1493
Gemini-3-Pro	62.6	86.0	92.1	49.8	70.3	86.6	1235	1083	1001	2058	2444	1401
Claude Opus 4.6	60.4	77.1	90.3	41.7	50.3	59.3	1005	932	917	1824	1972	1255
Open-Weight LLMs												
DeepSeek-R1	61.2	**82.5**	**90.5**	40.2	**66.0**	**78.5**	3451	2893	2657	4823	5121	3893
DeepSeek-V3.2	55.5	71.7	89.4	35.5	55.0	65.0	953	881	849	1763	1951	1213
Qwen3-235B-A22B	58.1	74.8	89.4	38.1	62.1	68.4	983	921	887	1847	2103	1273
Qwen3-32B	54.8	70.2	83.5	30.8	41.5	40.2	857	793	761	1603	1753	1099
ChatGLM-Z1-32B	53.9	74.0	84.1	31.5	40.2	42.5	843	779	751	1583	1719	1079
Qwen3-14B	50.2	74.6	87.7	28.5	35.2	38.5	721	679	649	1349	1499	947
Small Language Models (<10B)												
ChatGLM-Z1-9B	44.5	72.3	68.3	22.8	25.5	28.2	647	611	579	1249	1451	919
Qwen3-8B	38.5	48.9	65.1	15.2	12.8	15.5	453	419	397	853	981	649
Mistral-8B	42.8	53.4	49.2	18.5	19.4	22.8	613	569	541	1181	1319	859
Llama-3.1-8B	41.2	52.1	48.5	16.6	18.4	21.5	607	563	531	1151	1299	839
Gemma-2-9B	44.1	57.5	52.8	19.9	21.6	25.4	681	643	611	1283	1481	941
Lunar-R1 (Ours, 8B)												
No_Think	34.9	39.5	35.2	12.5	15.6	11.2	**227**	**199**	**185**	**313**	**347**	**289**
+ CPT Training	48.2	58.4	61.2	27.0	30.7	24.5	728	646	660	1231	1449	963
+ SFT Training	58.8	67.6	75.1	33.8	40.3	34.7	1196	1051	1079	2145	2446	1441
+ GRPO (w/o LDP)	59.1	70.4	79.8	35.6	38.4	35.8	947	893	863	1841	1919	1249
+ GRPO w/ LDP	**64.0**	76.6	79.7	**42.8**	46.5	45.2	442	415	391	785	891	635

4.2 Main Results

Overall Performance. Table 1 details the quantitative results across Lunar-QA and domain-specific benchmarks. Relative to the Qwen3-8B backbone, Lunar-R1 simultaneously elevates accuracy and compresses ATC, which confirms that LDP probe enables highly effective reasoning token allocation. Furthermore, Lunar-R1 approaches the performance of DeepSeek-V3.2 on scientific reasoning tasks with a markedly lower token footprint. These outcomes demonstrate that adapting reasoning length to query difficulty inherently optimizes both deductive capability and inference efficiency.

Output Preference Evaluation. As shown in Figure 3, against Qwen3-8B, Lunar-R1 is consistently preferred, achieving win rates of 62.0% under GPT-5 judging and 58.5% under human evaluation, with tie rates of 21.5% and 24.0%. Against larger DeepSeek-R1, Lunar-R1 remains competitive, with human evaluation reporting a 45.0% tie rate and a 22.5% win rate, while GPT-5 judging reports a 47.5% tie rate and a 28.0% win rate.

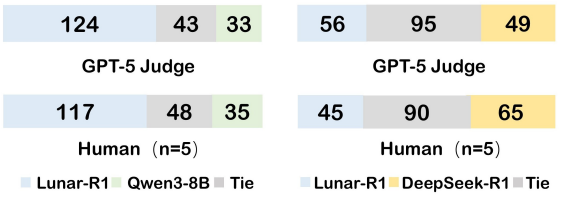

Figure 3: Pairwise preference evaluation of Lunar-R1 with Fleiss' $\kappa = 0.89$ inter-annotator agreement (n = 5).

4.3 Ablation Studies

Impact of Training Ablation. Figure 4(a) summarizes the performance across different training methods. The No_Think baseline uses the fewest tokens (227) but only achieves 34.9% accuracy, showing that direct generation cannot handle complex reasoning tasks. Adding PPO improves accuracy to 52.7% (+17.8%) but nearly triples the token consumption (612), which indicates inefficient exploration. GRPO (w/o LDP) further raises accuracy to 59.1% but causes the model to generate excessive text to maximize length-based rewards, pushing usage to a peak of 947 tokens. In contrast, GRPO w/ LDP achieves the highest accuracy of 64.0% while reducing consumption to 442 tokens.

Budget Parameters. We analyze the budget parameters δ and γ where δ determines the minimum token count and γ controls the sensitivity to difficulty. As shown in Figure 4 (b), increasing δ from 0 to 100 raises accuracy by 3.5% from 60.5% to 64.0%. However, further increasing δ to 200 yields a negligible gain of 0.1% while inflating ATC by 31% to 580. This indicates that excessive buffers lead to wasted computation. Similarly, setting γ to 1.5 optimizes the resource allocation. A higher setting of 2.0 reduces to 63.2% despite a higher consumption of 515 ATC. This suggests that an unrestrained budget encourages the model to generate irrelevant reasoning steps. Therefore, the configuration of $\delta = 100$ and $\gamma = 1.5$ achieves the optimal balance with 64.0% accuracy and 442 ATC.

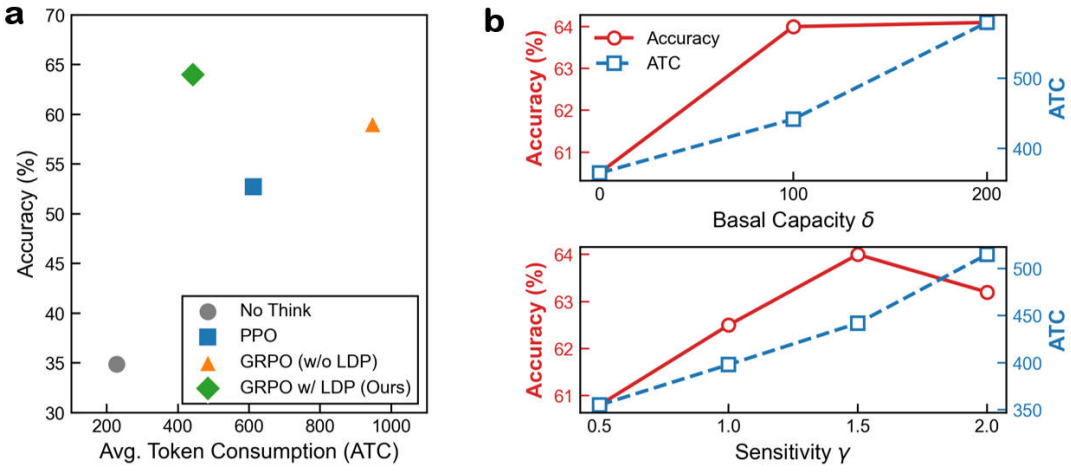

Figure 4: **Ablation study of training strategy and budget hyperparameters. (a)** Performance trade-off under different policy optimization settings. GRPO with LDP achieves a better balance between task accuracy and generation efficiency than PPO and GRPO without LDP. **(b)** Sensitivity analysis of the budget parameters δ and γ. Increasing the base budget δ beyond 100 brings limited accuracy improvement while substantially increasing ATC.

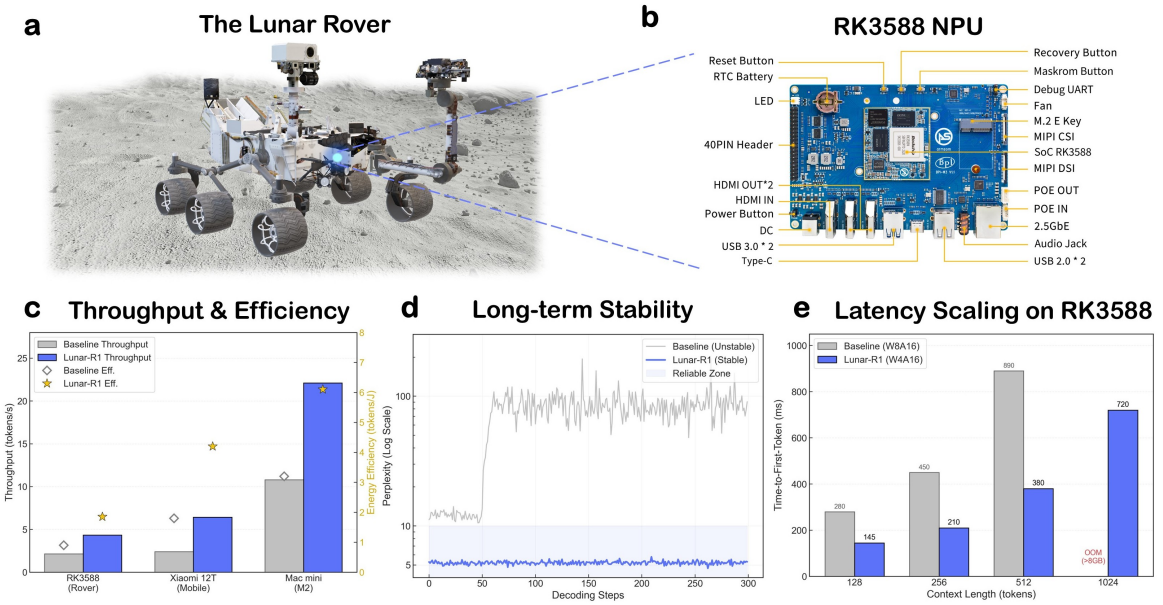

Figure 5: **Edge deployment profiling.** **(a)** The lunar rover. **(b)** RK3588 NPU. **(c)** Throughput and energy efficiency under different quantization settings. Lunar-R1 with W4A16 quantization improves decoding speed and energy efficiency over the W8A16 baseline. **(d)** Numerical stability during long-sequence generation. The baseline exhibits abrupt perplexity increases under low-precision offloading, while the stabilized implementation maintains stable perplexity over 300 decoding steps. **(e)** Latency scaling with input context length. Lunar-R1 maintains near-linear latency growth up to 1024 input tokens, whereas the baseline fails under the same memory constraint.

5 Edge Deployment

We evaluate Lunar-R1 across edge environments using the ELIB benchmarking framework (Chen et al., 2025a). The hardware testbed includes a rover-grade Rockchip RK3588 with 34 GB/s memory bandwidth, a Xiaomi Redmi Note 12 Turbo mobile platform, and a Mac mini M2 with 100 GB/s unified memory bandwidth. The Mac mini sustains interactive generation rates due to its higher memory bandwidth, whereas the RK3588 and mobile platforms are primarily limited by Model Bandwidth Utilization (MBU). On these resource-constrained devices, the standard W8A16 saturates DRAM bandwidth and keeps decoding throughput below 2.5 tok/s. We also observe that default OpenCL offloading on ARM-based SoCs can introduce numerical instability during long-sequence generation, leading to abrupt perplexity increases (PPL > 50) and reduced generation reliability.

To address these deployment constraints, Lunar-R1 adopts a stabilized W4A16 quantization strategy (Lin et al., 2024a) together with custom computing kernels. This configuration reduces memory traffic, improves MBU, and mitigates precision overflow during long decoding. Under tuned performance profiles, Lunar-R1 maintains stable perplexity below 6.5 and achieves 6.42 tok/s on the mobile platform and 22.10 tok/s on the Mac mini M2. Notably, as shown in Figure 5, on the target RK3588 rover platform, Lunar-R1 reaches 4.35 tok/s with a time-to-first-token latency of 380 ms and a specific energy consumption of 0.54 J/tok. These results indicate that Lunar-R1 can support practical lunar-domain reasoning on low-power edge hardware with improved throughput and energy efficiency.

6 Conclusion

This work presents Lunar-R1 to address the computational bottlenecks of deployment on lunar edge devices. We propose a LDP mechanism that exploits the linear separability of task complexity in terminal hidden states to allocate inference budgets. By integrating this difficulty signal into GRPO training, the model eliminates redundant reasoning steps without compromising reasoning depth. Experimental results show Lunar-R1 achieves SOTA performance among SLMs and outperforms larger LLMs like QwQ-32B on domain-specific tasks. Deployment on NPUs confirms the model satisfies latency and energy constraints. These findings demonstrate the feasibility of scientific reasoning for resource-limited space exploration missions.

Limitations

LDP estimates difficulty through a latent distribution over ordered difficulty states. The entropy-based difficulty signal shows a high correlation with expert assessments on Lunar-QA ($\rho = 0.82$), indicating that terminal hidden states provide useful cues for difficulty perception before response generation. However, its cross-domain validity depends on the alignment between the probe-training distribution and target tasks. A lunar-trained probe applied to mathematical deduction shows reduced correlation ($\rho = 0.63$ on MATH), whereas broader training on MMLU restores performance ($\rho = 0.73$ on MATH, $\rho = 0.79$ on Lunar-QA). Thus, cross-domain generalization requires either task-specific recalibration or diverse difficulty-state supervision.

Ethics Statement

All training data are sourced from publicly available repositories, with proprietary and personally identifiable information (PII) excluded. Human annotators and domain experts involved in data construction and evaluation were compensated above local statutory minimums. To ensure reproducibility, we commit to releasing model weights, datasets, and training code upon publication under open-source licenses. Lunar-R1 is designed exclusively for peaceful lunar exploration and mission support, and is not intended for military or adversarial use. Finally, we thank all the reviewers and area chair for contributions made to the review process.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Anthropic. 2026. Claude opus 4.6 system card. https://www.anthropic.com/system-cards. Accessed: 2026-02-14.

Daman Arora and Andrea Zanette. 2025. Training language models to reason efficiently. *arXiv preprint arXiv:2502.04463*.

Harald Köpping Athanasopoulos. 2019. The moon village and space 4.0: The 'open concept'as a new way of doing space? *Space Policy*, 49:101323. Publisher: Elsevier.

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

Jie Cao, Tianwei Lin, Zhenxuan Fan, Bo Yuan, Ziyuan Zhao, Rolan Yan, Wenqiao Zhang, and Siliang Tang. 2026. Draft-thinking: Learning efficient reasoning in long chain-of-thought llms. *arXiv preprint arXiv:2603.00578*.

Hao Chen, Cong Tian, Zixuan He, Bin Yu, Yepang Liu, and Jialun Cao. 2025a. Inference performance evaluation for llms on edge devices with a novel benchmarking framework and metric. *arXiv preprint arXiv:2508.11269*.

Weize Chen, Jin Tailin, Ning Ding, Huimin Chen, Zhiyuan Liu, Maosong Sun, et al. 2026. The overthinker's diet: Cutting token calories with difficulty-aware training. *Advances in Neural Information Processing Systems*, 38:27013–27038.

Xiao Chen, Sihang Zhou, Ke Liang, Xiaoyu Sun, and Xinwang Liu. 2025b. Skip-thinking: Chunk-wise chain-of-thought distillation enable smaller language models to reason better and faster.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, and others. 2025c. Do NOT think that much for 2+ 3=? on the overthinking of long reasoning models. In *Forty-second International Conference on Machine Learning*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Valerio Cosentino, Javier Luis, and Jordi Cabot. 2016. Findings from github: methods, datasets and limitations. In *Proceedings of the 13th international conference on mining software repositories*, pages 137–141.

Liang Ding, Ruyi Zhou, Tianyi Yu, Huaiguang Yang, Ximing He, Haibo Gao, Juntao Wang, Ye Yuan, Jia Wang, Zhengyin Wang, et al. 2024. Lunar rock investigation and tri-aspect characterization of lunar farside regolith by a digital twin. *Nature Communications*, 15(1):2098.

Matthew Foutter, Daniele Gammelli, Justin Kruger, Ethan Foss, Praneet Bhoj, Tommaso Guffanti, Simone D'Amico, and Marco Pavone. 2024. Space-LLaVA: A vision-language model adapted to extraterrestrial applications. *arXiv preprint arXiv:2408.05924*.

Lisa R Gaddis, Katherine H Joy, Ben J Bussey, James D Carpenter, Ian A Crawford, Richard C Elphic, Jasper S Halekas, Samuel J Lawrence, and Long Xiao. 2023. Recent exploration of the moon: Science from lunar missions since 2006. *Reviews in Mineralogy and Geochemistry*, 89(1):1–51. Publisher: Mineralogical Society of America.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Google DeepMind. 2025. Gemini 3. https://deepmind.google/models/gemini/. Accessed: 2026-02-08.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and others. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638. Publisher: Nature Publishing Group UK London.

Mandy Guo, Zihang Dai, Denny Vrandečić, and Rami Al-Rfou. 2020. Wiki-40b: Multilingual language model dataset. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2440–2452.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. 2025. Token-budget-aware llm reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 24842–24855.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. 2025. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Gokce Inal, Pouyan Navard, and Alper Yilmaz. 2026. Llava-le: Large language-and-vision assistant for lunar exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10251–10261.

Abhinav Kumar, Jaechul Roh, Ali Naseh, Marzena Karpinska, Mohit Iyyer, Amir Houmansadr, and Eugene Bagdasarian. 2026. Overthink: Slowdown attacks on reasoning llms.

Sunbowen Lee, Qingyu Yin, Chak Tou Leong, Jialiang Zhang, Yicheng Gong, Shiwen Ni, Min Yang, and Xiaoyu Shen. 2025. Probing the difficulty perception mechanism of large language models. *arXiv preprint arXiv:2510.05969*.

Jie Li, Fuyong Zhao, Panfeng Chen, Jiafu Xie, Xiangrui Zhang, Hui Li, Mei Chen, Yanhao Wang, and Ming Zhu. 2025a. An astronomical question answering dataset for evaluating large language models. *Scientific Data*, 12(1):447.

Yucheng Li, Yunhao Guo, Frank Guerin, and Chenghua Lin. 2024. An open-source data contamination report for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 528–541.

Zheng Li, Qingxiu Dong, Jingyuan Ma, Di Zhang, Kai Jia, and Zhifang Sui. 2025b. Selfbudgeter: Adaptive token allocation for efficient llm reasoning. *arXiv preprint arXiv:2505.11274*.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024a. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. *Proceedings of machine learning and systems*, 6:87–100.

Junhong Lin, Xinyue Zeng, Jie Zhu, Song Wang, Julian Shun, Jun Wu, and Dawei Zhou. 2025. Plan and budget: Effective and efficient test-time scaling on large language model reasoning. *arXiv preprint arXiv:2505.16122*.

Yangting Lin, Wei Yang, Hui Zhang, Hejiu Hui, Sen Hu, Long Xiao, Jianzhong Liu, Zhiyong Xiao, Zongyu Yue, Jinhai Zhang, and others. 2024b. Return to the moon: New perspectives on lunar exploration. *Science Bulletin*, 69(13):2136–2148. Publisher: Elsevier.

Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. 2025a. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*.

Jiayi Liu, Qianyu Zhang, Xue Wan, Shengyang Zhang, Yaolin Tian, Haodong Han, Yutao Zhao, Baichuan Liu, Zeyuan Zhao, and Xubo Luo. 2024. LuSNAR: A lunar segmentation, navigation and reconstruction dataset based on muti-sensor for autonomous exploration. *arXiv preprint arXiv:2407.06512*.

Wentao Liu, Hanglei Hu, Jie Zhou, Yuyang Ding, Junsong Li, Jiayi Zeng, Mengliang He, Qin Chen, Bo Jiang, Aimin Zhou, and others. 2023. Mathematical language models: A survey. *ACM Computing Surveys*. Publisher: ACM New York, NY.

Xiang Liu, Xuming Hu, Xiaowen Chu, and Eunsol Choi. 2025b. Diffadapt: Difficulty-adaptive reasoning for token-efficient llm inference. *arXiv preprint arXiv:2510.19669*.

Haotian Luo, Li Shen, Haiying He, Yibo Wang, Shiwei Liu, Wei Li, Naiqiang Tan, Xiaochun Cao, and Dacheng Tao. 2025. O1-pruner: Length-harmonizing fine-tuning for o1-like reasoning pruning. *arXiv preprint arXiv:2501.12570*.

Wenjie Ma, Jingxuan He, Charlie Snell, Tyler Griggs, Sewon Min, and Matei Zaharia. 2025. Reasoning models can be effective without thinking. *arXiv preprint arXiv:2504.09858*.

Microsoft. 2024. Markitdown: A python tool for converting files to markdown. https://github.com/microsoft/markitdown.

Carmen Misa Moreira, Sofia Coloma Chacon, Andreas M Hein, and Miguel Olivares-Mendez. 2025. An edge computing architecture for a lunar dust recognition system. In *2025 IEEE Aerospace Conference*, pages 1–8. IEEE.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori B Hashimoto. 2025. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20286–20332.

Gregory Navarro, Valentin Porchet, Luc Truntzler, Stephane Lallee, Alexis Paillet, and Sebastien Barde. 2024. Ai4u: A digital companion for the crew for lunar and martian missions. In *2024 International Conference on Environmental Systems*.

OpenAI. 2025. Gpt-5.2 is here. https://openai.com/gpt-5/. Accessed: 2026-02-08.

Jay M Patel. 2020. Introduction to common crawl datasets. In *Getting structured data from the internet: running web crawlers/scrapers on a big data production scale*, pages 277–324. Springer.

Zhaoyu Pei, Jizhong Liu, Qian Wang, Yan Kang, Yongliao Zou, He Zhang, Yuhua Zhang, Huaiyu He, Qiong Wang, Ruihong Yang, et al. 2020. Overview of lunar exploration and international lunar research station. *Chin. Sci. Bull*, 65(24):2577–2586.

Michael Pekala, Gregory Canal, Samuel Barham, Milena B Graziano, Morgan Trexler, Leslie Hamilton, Elizabeth Reilly, and Christopher D Stiles. 2025. Towards large language models for lunar mission planning and in situ resource utilization. *arXiv preprint arXiv:2504.20125*.

Lucas Carrit Delgado Pinheiro, Ziru Chen, Bruno Caixeta Piazza, Ness Shroff, Yingbin Liang, Yuan-Sen Ting, and Huan Sun. 2025. Large language models achieve gold medal performance at the international olympiad on astronomy & astrophysics (ioaa). *arXiv preprint arXiv:2510.05016*.

Mirali Purohit, Bimal Gajera, Vatsal Malaviya, Irish Mehta, Kunal Kasodekar, Jacob Adler, Steven Lu, Umaa Rebbapragada, and Hannah Kerner. 2025. Mars-bench: A benchmark for evaluating foundation models for mars science tasks. *arXiv preprint arXiv:2510.24010*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Yi Shen, Jian Zhang, Jieyun Huang, Shuming Shi, Wenjing Zhang, Jiangze Yan, Ning Wang, Kai Wang, Zhaoxiang Liu, and Shiguo Lian. 2025. Dast: Difficulty-adaptive slow-thinking for large reasoning models. *arXiv preprint arXiv:2503.04472*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Jinghang Shi, Xiaoyu Tang, Yang Huang, Yuyang Li, Xiao Kong, Yanxia Zhang, and Caizhan Yue. 2025. Astrommbench: A benchmark for evaluating multimodal large language models capabilities in astronomy. *arXiv preprint arXiv:2510.00063*.

Parshin Shojaee, Iman Mirzadeh, Maxwell Horton, Samy Bengio, Mehrdad Farajtabar, et al. 2026. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *Advances in Neural Information Processing Systems*, 38:108018–108059.

Marshall Smith, Douglas Craig, Nicole Herrmann, Erin Mahoney, Jonathan Krezel, Nate McIntyre, and Kandyce Goodliff. 2020. The artemis program: An overview of nasa's activities to return humans to the moon. In *2020 IEEE aerospace conference*, pages 1–10. IEEE.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Hanjie Chen, and others. 2025. Stop overthinking: A survey on efficient reasoning for large language models, 2025. *URL https://arxiv. org/abs/2503.16419*.

Haoxiang Sun, Yingqian Min, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, Zheng Liu, Zhongyuan Wang, and Ji-Rong Wen. 2025. Challenging the boundaries of reasoning: An olympiad-level math benchmark for large language models. *arXiv preprint arXiv:2503.21380*.

Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. 2023. Large language models in medicine. *Nature medicine*, 29(8):1930–1940. Publisher: Nature Publishing Group US New York.

Guiyao Tie, Zeli Zhao, Dingjie Song, Fuyang Wei, Rong Zhou, Yurou Dai, Wen Yin, Zhejian Yang,

Jiangyue Yan, Yao Su, et al. 2025. A survey on post-training of large language models. *arXiv preprint arXiv:2503.06072*.

Y-S Ting, Tuan Dung Nguyen, Tirthankar Ghosal, Rui Pan, Hardik Arora, Zechang Sun, Tijmen de Haan, Nesar Ramachandra, Azton Wells, Sandeep Madireddy, et al. 2025. Astromlab 1: Who wins astronomy jeopardy!? *Astronomy and Computing*, 51:100893.

Hao Wang, Shuqi Xue, Hongbo Zhang, Chunhui Wang, and Yan Fu. 2025. A human–robot team knowledge-enhanced large language model for fault analysis in lunar surface exploration. *Aerospace*, 12(4):325. Publisher: MDPI.

Weiyi Wang, Xinchi Chen, Jingjing Gong, Xuanjing Huang, and Xipeng Qiu. 2026. Astroreason-bench: Evaluating unified agentic planning across heterogeneous space planning problems. *arXiv preprint arXiv:2601.11354*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 13484–13508.

Hao Wen, Xinrui Wu, Yi Sun, Feifei Zhang, Liye Chen, Jie Wang, Yunxin Liu, Yunhao Liu, Ya-Qin Zhang, and Yuanchun Li. 2025. Budgetthinker: Empowering budget-aware llm reasoning with control tokens. *arXiv preprint arXiv:2508.17196*.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. Ccnet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the twelfth language resources and evaluation conference*, pages 4003–4012.

Di Wu, Raymond Zhang, Enrico M Zucchelli, Yongchao Chen, and Richard Linares. 2025a. Apbench and benchmarking large language model performance in fundamental astrodynamics problems for space engineering. *Scientific Reports*, 15(1):7944.

Han Wu, Yuxuan Yao, Shuqi Liu, Zehua Liu, Xiaojin Fu, Xiongwei Han, Xing Li, Hui-Ling Zhen, Tao Zhong, and Mingxuan Yuan. 2025b. Unlocking efficient long-to-short llm reasoning with model merging. *arXiv preprint arXiv:2503.20641*.

Zhiqiu Xia, Jinxuan Xu, Yuqian Zhang, and Hang Liu. 2025. A survey of uncertainty estimation methods on large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 21381–21396.

Xin-Yu Xiao, Yalei Liu, Xiangyu Liu, Zengrui Li, Erwei Yin, and Qianchen Xia. 2025. Lunar twins: We choose to go to the moon with large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 1325–1339.

Xin-Yu Xiao, Ye Tian, Yalei Liu, Xiangyu Liu, Tianyang Lu, Erwei Yin, Chenshan Guang, et al. 2026. Towards evaluating task-oriented reasoning of llms in lunar exploration scenarios.

REN Xiaoqiang, WU Weiren, WANG Hongyu, ZHANG Zhe, ZHANG Longxi, ZHANG Xinghua, LI Maodeng, ZHANG Pengshuo, and TIAN Jian. 2025. Prospects of the lunar exploration development and key technologies. *Journal of Deep Space Exploration*, 12(2):99–109. Publisher: Beijing Institute of Technology Press.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Peng Zhang, Wei Dai, Ran Niu, Guang Zhang, Guanghui Liu, Xin Liu, Zheng Bo, Zhi Wang, Haibo Zheng, Chengbao Liu, and others. 2023. Overview of the lunar in situ resource utilization techniques for future lunar missions. *Space: Science & Technology*, 3:0037. Publisher: AAAS.

Qiyuan Zhang, Fuyuan Lyu, Zexu Sun, Lei Wang, Weixu Zhang, Wenyue Hua, Haolun Wu, Zhihan Guo, Yufei Wang, Niklas Muennighoff, et al. 2025a. A survey on test-time scaling in large language models: What, how, where, and how well? *arXiv preprint arXiv:2503.24235*.

Ruofan Zhang, Bin Xia, Zhen Cheng, Cairen Jian, Minglun Yang, Ngai Wong, and Yuan Cheng. 2025b. Dart: Difficulty-adaptive reasoning truncation for efficient large language models. *arXiv preprint arXiv:2511.01170*.

Xiang Zhao and You Song. 2024. Exploration and application of AI in space science. In *ICML 2024 AI for Science Workshop*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*.

Zibin Zheng, Kaiwen Ning, Yanlin Wang, Jingwen Zhang, Dewu Zheng, Mingxi Ye, and Jiachi Chen. 2023. A survey of large language models for code: Evolution, benchmarking, and future trends. *arXiv preprint arXiv:2311.10372*.

Wanrong Zhu, Jack Hessel, Anas Awadalla, Samir Yitzhak Gadre, Jesse Dodge, Alex Fang, Youngjae Yu, Ludwig Schmidt, William Yang Wang, and Yejin Choi. 2023. Multimodal c4: An open, billion-scale corpus of images interleaved with text. *Advances in Neural Information Processing Systems*, 36:8958–8974.

Appendices

A Training Implementation

A.1 Data Construction

Lunar Science Corpus. The pre-training corpus draws from telemetry metadata, mission logs, and analysis reports from the NASA Planetary Data System and the CNSA Data Release System. The initial collection comprises 320.6 GB of documents, including PDFs, spreadsheets, and image-embedded technical reports. We use MarkItDown (Microsoft, 2024) to convert these raw formats into structured Markdown; this tool corrects OCR errors and extracts text from embedded tables and figures. Following the CCNet protocol (Wenzek et al., 2020), we filter documents by perplexity and discard segments with n-gram repetition rates above 40%. This process yields 42.3 GB of cleaned, high-quality text for domain-adaptive pre-training.

Instruction Data & Benchmark. We construct the SFT dataset using a self-instruct pipeline (Wang et al., 2023). The process begins with 2,000 expert-authored seed instructions for lunar mission scenarios. We retain samples with an expert approval rating $\alpha > 0.9$, yielding 55,000 high-fidelity instruction-response pairs for post-training. In parallel, we **manually** build the independent Lunar-QA evaluation benchmark, consisting of 5,000 triplets (Q, A, D) drawn from held-out corpus segments, where Q is the question, A is the correct answer, and D denotes the expert-assigned difficulty label (easy, medium, hard). 13-gram overlap-based deduplication is applied to ensure no overlap with the training data. We further conduct contamination tests in Appendix F to validate the datasets.

Data Decontamination. An n-gram overlap detection algorithm systematically detects data leakage originating from shared text segments. Formally, the protocol flags a training document $d \in \mathcal{D}_{\text{train}}$ as contaminated if it shares any 13-gram subsequence with an evaluation sample $s \in \mathcal{S}_{\text{eval}}$:

$$C = \mathbb{I}\left(\begin{array}{c} \exists s \in \mathcal{S}_{\text{eval}} : \\ \text{grams}_{13}(s) \cap \text{grams}_{13}(d) \neq \emptyset \end{array} \right) \quad (10)$$

where $\mathbb{I}(\cdot)$ denotes the indicator function. Threshold is set to $k = 13$, adhering to the decontamination established by GPT-3 (Brown et al., 2020).

A.2 Training Framework

Problem Formulation We formulate the reasoning process as a token-level Markov Decision Process (MDP). Given an input query x, the policy π_θ generates a sequence $y = (y_t, y_a)$, where y_t denotes the latent reasoning chain encapsulated in <think> tags, and y_a represents the final answer. The optimization objective is to maximize the expected return under efficiency constraints:

$$\mathcal{J}(\theta) = \mathbb{E}_{y \sim \pi_\theta(\cdot|x)} \left[R(y_a, y^*) - \lambda \log(|y_t|) \right] \quad (11)$$

where y^* is the ground truth, $R(\cdot)$ is the reward function, and λ regulates the trade-off between reasoning depth and token consumption.

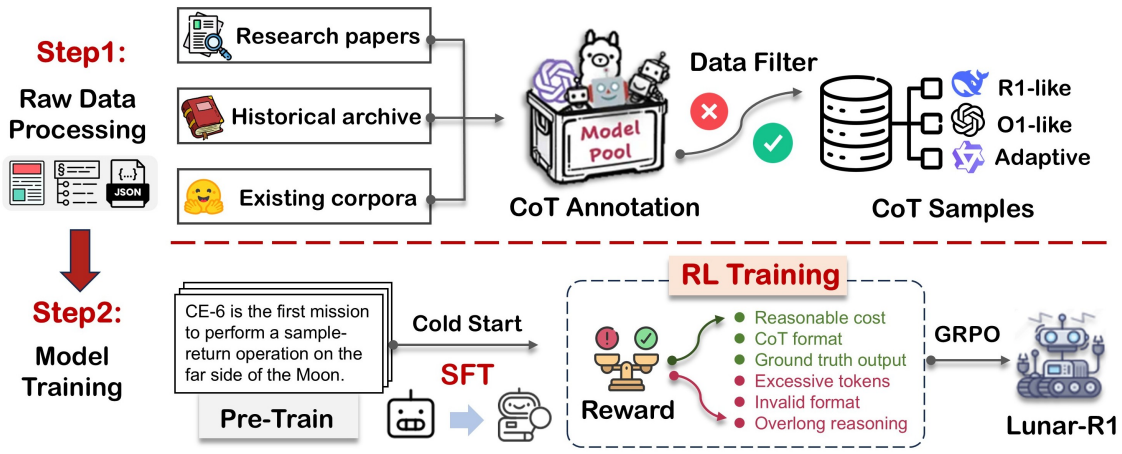

Figure 6: **Training framework for Lunar-R1.** Step 1 constructs high-quality reasoning CoT via automated annotation across a model pool. Step 2 executes a progressive training pipeline. This sequence advances through continued pre-training, SFT, and GRPO. It optimizes policy convergence via difficulty-aware reward modeling.

Stage I: Continued Pre-training. As illustrated in Figure 6, we continue pre-training on the 42.3 GB Lunar Science Corpus using the Qwen3-8B backbone. We apply MinHash-LSH deduplication with a Jaccard similarity threshold $\tau = 0.8$ to remove near-duplicate documents while retaining domain-specific terminology and mathematical expressions. Training is conducted with the LlamaFactory framework (Zheng et al., 2024) using a cosine learning rate schedule (peak $\eta_{\max} = 1 \times 10^{-5}$), a global batch size of 64, running for 3 epochs.

Stage II: Cold Start Supervised Fine-Tuning. Direct reinforcement learning from the CPT checkpoint can lead to unstable training and slow convergence, as the model lacks exposure to structured reasoning patterns (Tie et al., 2025). To address this, we perform a cold-start SFT stage using distilled CoT samples. We collect 100,000 reasoning trajectories from DeepSeek-R1, GPT-5, and Qwen3-235B, stratified by length into Short (< 512 tokens), Medium (512–2048 tokens), and Long (> 2048 tokens) categories in a 2:5:3 ratio. This distribution prevents the LLM from collapsing to overly concise responses while ensuring sufficient exposure to extended thinking. We fine-tune the model by minimizing negative log-likelihood $\mathcal{L}_{\text{SFT}} = -\mathbb{E}_{(x,y)} \log \pi_\theta(y|x)$ with learning rate 2×10^{-5}, maximum sequence length 8192, and training for 2 epochs, totaling approximately 6,250 steps on 8×NVIDIA A100 GPUs over 47 hours.

Stage III: Group Relative Policy Optimization. We compute the advantage $\hat{A}_i$ for each trajectory y_i by normalizing its reward r_i against the group

$$\hat{A}_i = \frac{r_i - \mu_G}{\sigma_G + \epsilon}, \tag{12}$$

where μ_G and σ_G denote the mean and standard deviation of rewards within the group.

Hybrid Reward Modeling The deterministic components include Correctness, which assigns a sparse positive reward exclusively for exact ground-truth matches to ensure terminal accuracy, and Format, which incentivizes schema through valid XML encapsulation verification. Crucially,the LDP linearly projects the hidden state as a continuous scalar s. The efficiency constraint utilizes a ReLU function to penalize token that exceeds the budget.

Implementation Details. Optimization is performed with the Verl framework (Sheng et al., 2024) on 8 NVIDIA A100 GPUs for approximately 35 hours. The policy is initialized from the SFT checkpoint and trained with group size $G = 16$, learning rate $\eta = 1 \times 10^{-5}$, and KL coefficient $\beta_{\text{KL}} = 0.04$ (see Figure 7). The probe's supervision target is a continuous label $y \in [0, 1]$ obtained by normalizing the empirical difficulty entropy $H(q)$ over ordered difficulty states. It maps the terminal hidden state $\mathbf{h}_T(q)$ to a sigmoid-constrained difficulty score $s \in [0, 1]$ and is optimized with mean squared error against the entropy-normalized target. After training, the probe is frozen and used only to assign adaptive reasoning budgets during GRPO. The calibrated score achieves an ECE of 0.045 with 10 equal-width bins and a ROC-AUC of 0.88 under a median-split difficulty threshold.

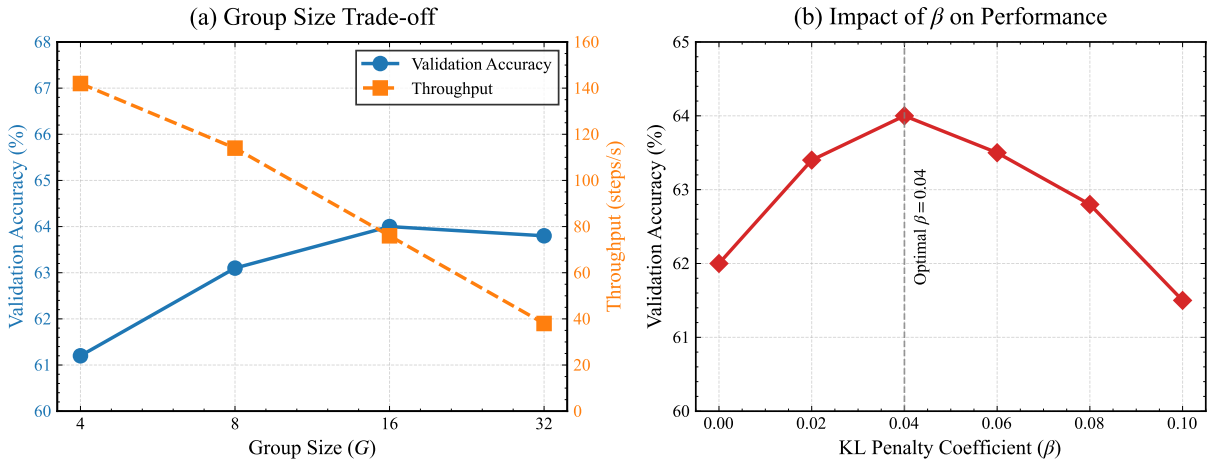

Figure 7: **GRPO hyperparameter ablation studies. (a)** The trade-off between validation accuracy and system throughput across varying group sizes G. $G = 16$ is selected as the Pareto-optimal point, balancing performance gains with computational efficiency. **(b)** The impact of the KL penalty coefficient β on validation accuracy.

B Benchmark Details

Lunar-QA (Ours). As shown in the top panel of Figure 8, it exhibits a high correlation ($\rho = 0.82$) with expert assessments, verifying its reliability in reflecting operational challenges.

Astro-QA (Li et al., 2025a). A comprehensive dataset quantifying instruction following and retrieval precision, designed to minimize hallucinations in fine-grained fact-checking tasks.

AstroBench (Ting et al., 2025). An expert-level suite assessing the breadth of knowledge and reasoning across subfields, including Stellar Physics.

AstroReason-Bench (Wang et al., 2026). A reasoning benchmark focusing on multi-step logical inference and physical unit consistency within planetary science contexts.

APBench (Wu et al., 2025a). Dedicated to astrodynamics, this benchmark mandates high-precision computation for differential equations and trajectory optimization, distinguishing mathematical proficiency from textual understanding.

IOAA (Pinheiro et al., 2025). A challenge sourced from Astronomy Olympiads. It tests the generalization of physical intuition through complex derivations in unseen scenarios.

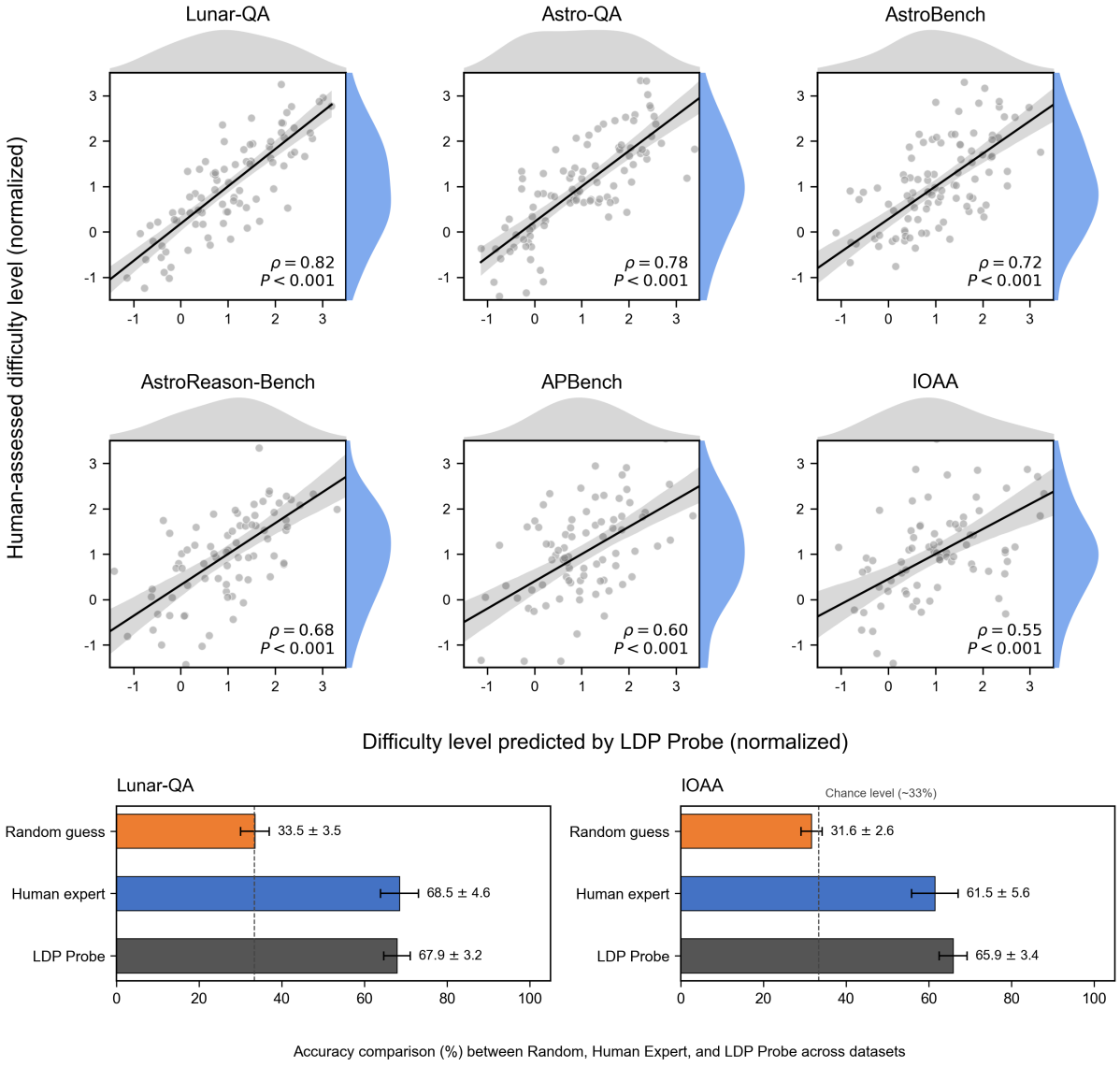

Figure 8: **Difficulty alignment and performance comparison across benchmarks.** The scatter plots (top) demonstrate the correlation (ρ) between LDP Probe predictions and human-assessed difficulty across 6 datasets. The bar charts (bottom) compare the accuracy of random guessing, human experts (n = 5), and the LDP Probe on Lunar-QA and IOAA, highlighting the distinct difficulty levels and generalization challenges of these tasks.

C Extended Experimental Analysis

Efficient reasoning has recently become an active research direction, to further evaluate the effectiveness of LDP, Table 2 reports results on GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), and AIME (Sun et al., 2025), compared with representative reasoning-control baselines. The empirical results show that LDP improves the accuracy–efficiency trade-off by reducing redundant reasoning while maintaining competitive task performance, outperforming representative methods under the reported setting.

Optimization of Computational Efficiency. Existing redundancy reduction methods often fail to balance reasoning depth with token sparsity. At the 1.5B parameter scale, OverThink reduces generation length but causes accuracy on the challenging AIME benchmark to decline from 29.4% to 28.3%. Conversely, LDP Probe compresses the average generation length to 3006 tokens, representing a 49.7% reduction compared to the original chain-of-thought baseline, while increasing average accuracy to 66.5%. These metrics indicate that LDP Probe eliminates invalid computational steps without truncating necessary reasoning logic. In the 7B parameter scale experiments, LDP Probe outperforms the dynamic routing baseline TLMRE, achieving an average accuracy of 79.2% compared to 78.2% for TLMRE while reducing token consumption from 4096 to 3426.

Reasoning Integrity on High-Complexity Tasks. The AIME dataset requires high integrity of the reasoning chain. Static pruning methods such as DAST and O1-Pruner exhibit significant performance degradation. For instance, the AIME accuracy of DAST on the 1.5B model declines to 26.9%, suggesting the erroneous removal of critical intermediate derivation processes. LDP Probe maintains robustness on AIME and achieves accuracies of 31.2% and 55.2% on 1.5B and 7B models respectively. The data confirms that LDP Probe adaptively allocates computational resources based on difficulty, compressing generation length on elementary GSM8K while preserving complete long-range reasoning capabilities for complex tasks.

Table 2: **Performance evaluation on mathematical benchmarks.** We report Top-1 accuracy (%) and average generation length (tokens) for DeepSeek-R1-Distill-Qwen backbones. **Bold** and underlined values denote the best and second-best results, respectively. LDP Probe is trained on the DeepMath-103K dataset (He et al., 2025).

Method	Accuracy↑				Length↓			
	GSM8K	MATH	AIME	AVG.	GSM8K	MATH	AIME	AVG.
DeepSeek-R1-Distill-Qwen-1.5B								
$\text{Standard}_{\text{Thinking}}$	79.0%	80.6%	29.4%	63.0%	978	4887	12073	5979
$\text{Standard}_{\text{NoThinking}}$	69.8%	67.2%	14.0%	50.3%	280	658	2190	1043
$\text{DPO}_{\text{Shortest}}$ (Rafailov et al., 2023)	78.3%	82.4%	30.7%	63.8%	804	3708	10794	5102
OverThink (Kumar et al., 2026)	77.2%	81.2%	28.3%	62.2%	709	4131	11269	5370
DAST (Shen et al., 2025)	77.2%	83.0%	26.9%	62.4%	586	2428	7745	3586
O1-Pruner (Luo et al., 2025)	74.8%	82.2%	28.9%	62.0%	458	3212	10361	4677
TLMRE (Arora and Zanette, 2025)	80.7%	85.0%	29.2%	65.0%	863	3007	8982	4284
ModelMerging (Wu et al., 2025b)	79.7%	63.0%	18.1%	53.6%	603	2723	10337	4554
$\text{RFT}_{\text{MixThinking}}$ (Ma et al., 2025)	76.0%	72.4%	25.2%	57.9%	1077	4341	11157	5525
LDP Probe (Ours)	**81.3%**	**85.4%**	**31.2%**	**66.5%**	**431**	**1815**	**6810**	**3006**
DeepSeek-R1-Distill-Qwen-7B								
$\text{Standard}_{\text{Thinking}}$	87.9%	90.2%	53.5%	77.2%	682	3674	10306	4887
$\text{Standard}_{\text{NoThinking}}$	85.1%	80.6%	24.2%	63.3%	283	697	1929	970
$\text{DPO}_{\text{Shortest}}$ (Rafailov et al., 2023)	85.7%	91.6%	52.5%	76.6%	402	2499	8699	3867
OverThink (Kumar et al., 2026)	86.3%	89.4%	53.1%	76.3%	426	2435	8744	3868
DAST (Shen et al., 2025)	86.7%	89.6%	45.6%	74.0%	459	2162	**7578**	3400
O1-Pruner (Luo et al., 2025)	87.6%	86.6%	49.2%	74.5%	428	2534	9719	4227
TLMRE (Arora and Zanette, 2025)	88.9%	91.8%	54.0%	78.2%	756	2899	8633	4096
ModelMerging (Wu et al., 2025b)	88.4%	72.6%	36.9%	66.0%	531	2280	8624	3812
$\text{RFT}_{\text{MixThinking}}$ (Ma et al., 2025)	86.2%	84.8%	49.4%	73.5%	365	2411	9969	4248
LDP Probe (Ours)	**90.3%**	**92.1%**	**55.2%**	**79.2%**	**298**	**1876**	8105	**3426**

D Theoretical Analysis of LDP

Let $X = (x_1, \ldots, x_T)$ denote the input prompt and $Y = (y_1, \ldots, y_L)$ the generated reasoning trajectory. The reasoning complexity is defined as a deterministic functional $C = \mathcal{F}(Y)$. Given a representation $\mathbf{h}$ derived from the model hidden states, our objective is to characterize $I(C; \mathbf{h})$.

Definition 1 (Causal State Update)**.** Consider a decoder-only Transformer with parameters θ. Due to the causal attention mask, the hidden state at position t is a deterministic function of the prefix:

$$\mathbf{h}_t = f(x_{\leq t}; \theta), \quad t = 1, \ldots, T. \tag{13}$$

Assumption 1 (Global Dependency)**.** The reasoning complexity C depends on the full semantic content of the prompt X. Formally, for any $t < T$,

$$I(C; x_{t+1:T} \mid x_{\leq t}) > 0. \tag{14}$$

Lemma 1 (Prefix Independence)**.** *For any $t < T$, the hidden state $\mathbf{h}_t$ is conditionally independent of the suffix $x_{t+1:T}$ given the prefix $x_{\leq t}$:*

$$P(\mathbf{h}_t \mid x_{\leq T}) = P(\mathbf{h}_t \mid x_{\leq t}). \tag{15}$$

Proof. By Definition 1, $\mathbf{h}_t$ is computed solely from $x_{\leq t}$. In the corresponding causal graph, $\mathbf{h}_t$ is d-separated from $\{x_{t+1}, \ldots, x_T\}$ given $x_{\leq t}$. □

Proposition 1 (Sufficiency of $\mathbf{h}_T$)**.** *Conditioned on model parameters θ, the terminal hidden state $\mathbf{h}_T$ is a sufficient statistic of X for the generation of Y. In particular,*

$$X \to \mathbf{h}_T \to Y \to C, \tag{16}$$

and hence $I(C; X \mid \mathbf{h}_T, \theta) = 0$.

Proof. In an autoregressive Transformer, the generation of Y is initialized from $\mathbf{h}_T$, and all subsequent decoding states are computed recursively from $\mathbf{h}_T$ and previously tokens. Therefore, conditioned on θ, the distribution $P(Y \mid X)$ depends on X only through $\mathbf{h}_T$. Since C is a deterministic function of Y, the stated sufficiency follows. □

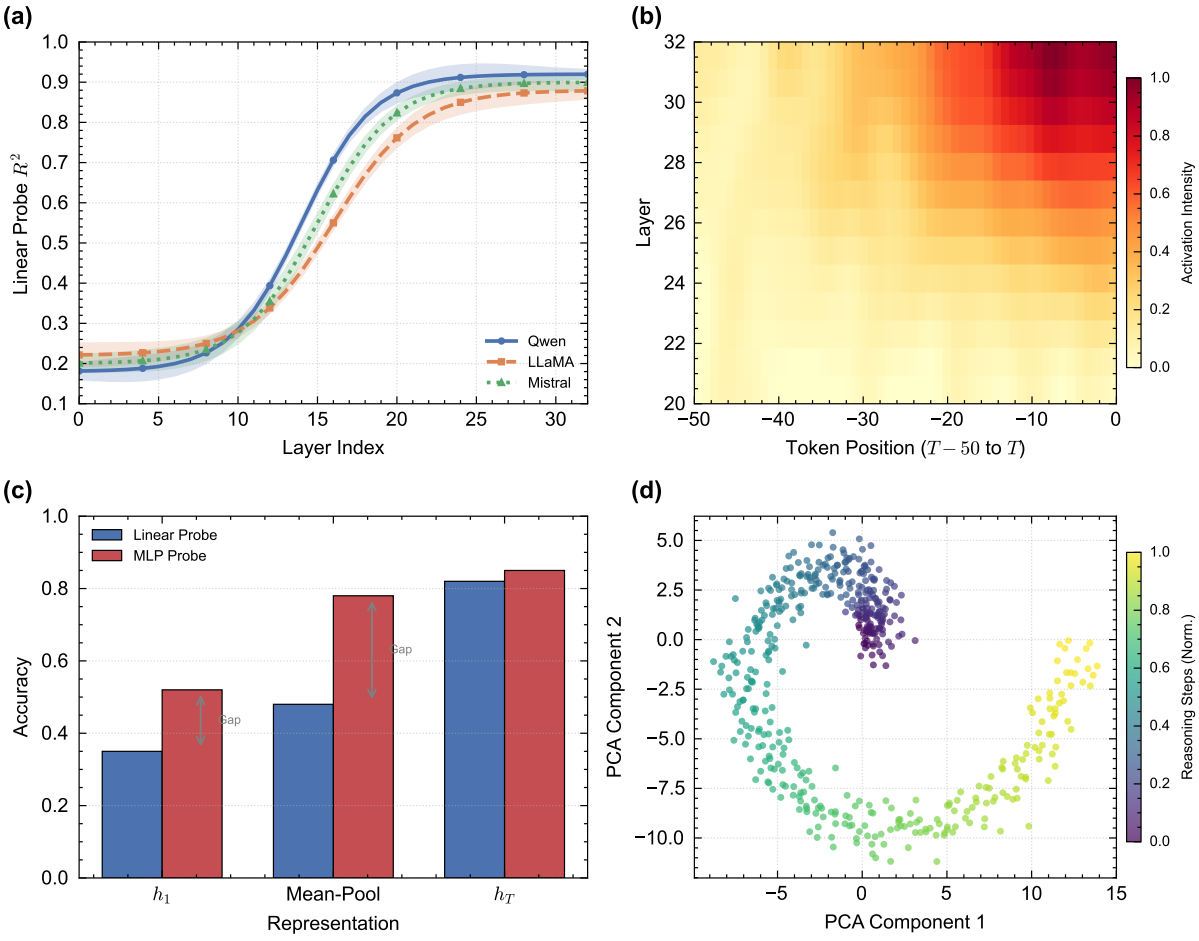

Figure 9: **Representation linearity and localization.** (a) R^2 scores across layers. (b) Activation norms by position. (c) Linear vs. MLP probe comparison. (d) PCA projection of the complexity manifold.

By Lemma 1, $\phi(X)$ cannot encode information about the suffix $x_{t+1:T}$. Under Assumption 1, this suffix carries information relevant to C.

Remark 1 (Mean Pooling)**.** Mean pooling constructs $\mathbf{h}_{\text{mean}} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{h}_t$ as a deterministic function of all hidden states. Although $\mathbf{h}_{\text{mean}}$ includes $\mathbf{h}_T$, the averaging operation may obscure the state that directly initializes generation. Unless C is invariant to positional information, it follows that

$$I(C; \mathbf{h}_{\text{mean}}) \leq I(C; \mathbf{h}_T), \tag{17}$$

making $\mathbf{h}_T$ the more principled representation.

We validate these claims using Qwen-2.5-7B, LLaMA-3-8B, and Mistral-7B-v0.3 on GSM8K and ARC-Challenge (Clark et al., 2018). Figure 9 (a) shows that R^2 scores increase progressively through deeper layers. Panel (b) demonstrates that activation norms concentrate at the terminal token position. Panel (c) compares linear probes against MLP baselines, revealing that $\mathbf{h}_T$ achieves $R^2 \approx 0.89$ compared to $R^2 \approx 0.91$ for MLPs, suggesting that complexity information is largely linearly separable. Panel (d) visualizes PCA projections, which form smooth continuous manifolds aligned with reasoning step counts.

We next test whether $\mathbf{h}_T$ plays a role in determining reasoning depth through activation steering experiments. After identifying a complexity direction $\mathbf{w}$ via linear regression, we intervene on hidden states using $\mathbf{h}'_T \leftarrow \mathbf{h}_T + \alpha\mathbf{w}$. Figure 10 (b,c) shows that reasoning length varies monotonically with the steering coefficient α: positive interventions extend the reasoning chain while negative interventions truncate it. Figure 10 (d) shows that perplexity remains stable across the intervention range $\alpha \in [-5, 5]$, indicating that complexity adjustments do not materially degrade linguistic fluency. To ensure that $\mathbf{h}_T$ captures semantic structure rather than surface token count, we evaluate on length-controlled samples. We replicate the steering experiments on LLaMA-3 and Mistral, observing consistent responses and stable perplexity.

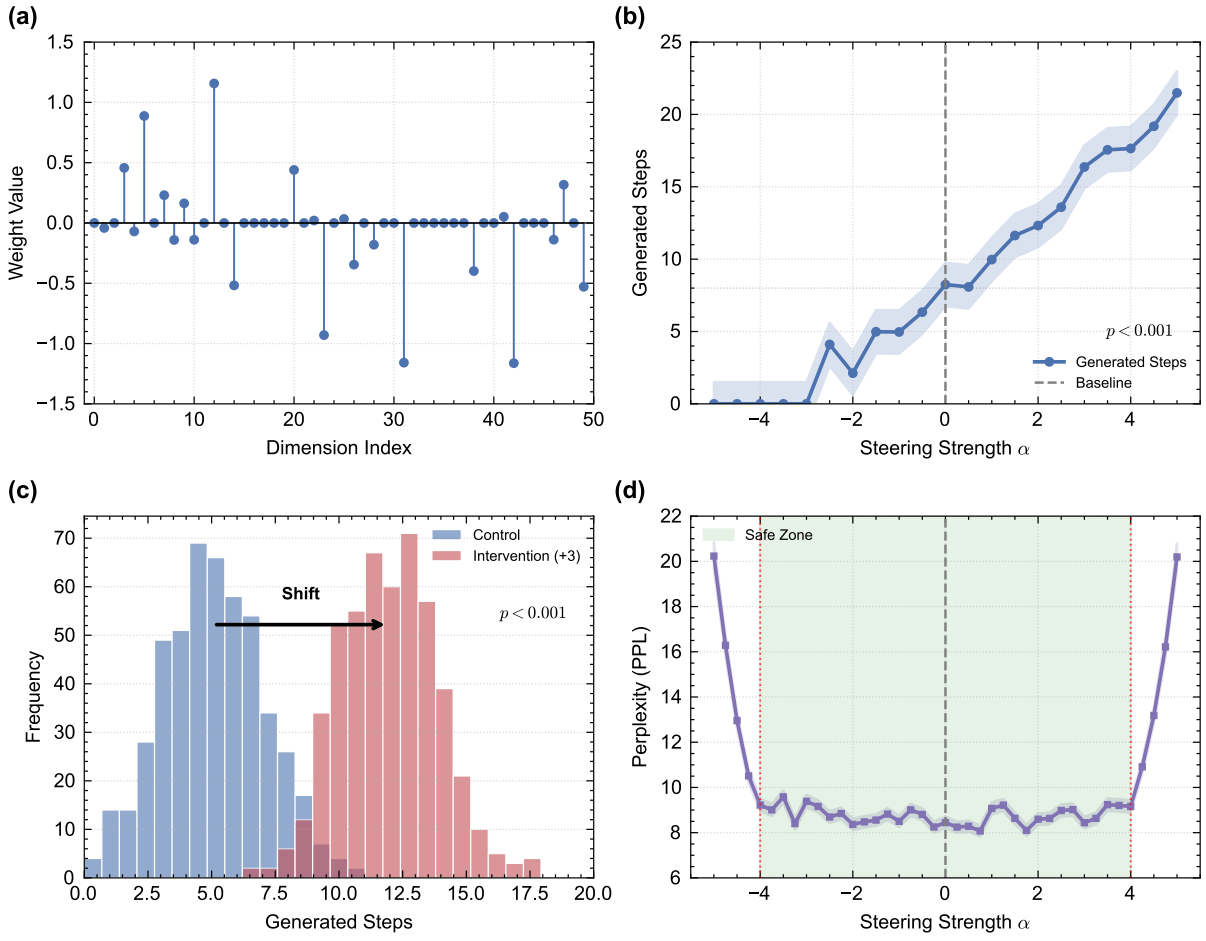

Figure 10: **LDP probe validation via activation steering.** (a) Complexity direction $\mathbf{w}$. (b) Reasoning length vs. steering coefficient α. (c) Distribution shift under intervention. (d) Perplexity stability.

E Extended Discussion of LDP

E.1 Comparison with Current Methods

Difficulty entropy frameworks provide token-level difficulty metrics (Xia et al., 2025). Task decomposition pipelines, including Plan and Budget (Lin et al., 2025) and Think_Budget (Han et al., 2025), control computational overhead through localized budgeting. DiffAdapt (Liu et al., 2025b) trains a probe to detect difficulty entropy within hidden layers but operates solely as an inference-time router, directing queries toward predefined decoding strategies (including specific prompts, sampling temperatures, and token limits) without model retraining. DIET (Chen et al., 2026) enforces difficulty-aware length compression penalties during GRPO. However, the reported implementations are either not open-sourced or rely on checkpoints, making direct reproduction on our Lunar-R1 pipeline infeasible.

LDP differs across three dimensions. First, it extracts difficulty signals linearly from terminal hidden states h_T rather than from intermediate layers, building on findings that task difficulty maintains linear separability within latent representations (Lee et al., 2025). Second, it injects this continuous difficulty signal into the GRPO reward modeling pipeline, enabling the policy model to learn adaptive reasoning allocation during training instead of relying on static inference-time routing. Third, LDP functions as an efficiency-driven budget controller; its performance ceiling remains inherently dependent on base model capabilities. In contrast, Test-Time Scaling (Zhang et al., 2025a) achieve higher performance maxima but require substantially greater token consumption.

E.2 Cross-Domain Generalization

We empirically quantify the zero-shot cross-domain generalization capability of the LDP mechanism across three training configurations. Probe1 is trained exclusively on Lunar-QA, Probe2 on MATH, and Probe3 on the comprehensive MMLU benchmark. We compute the Pearson correlation (ρ) between each probe's predicted difficulty scores and expert-assigned difficulty labels. Table 3 presents the full correlation matrix. The results confirm the predictive validity of LDP across domains. Probe1 achieves $\rho = 0.63$ on MATH, while Probe3 maintains robust performance across all evaluated domains. However, inherent distributional disparities between lunar exploration missions and abstract mathematics degrade zero-shot accuracy.

Table 3: **Cross-domain generalization of LDP probes measured via Pearson correlation (ρ).** † denotes the in-domain setting. Significance is evaluated using a two-sided Pearson correlation test, and all reported correlations satisfy $p < 0.001$.

Datasets	Probe1	Probe2	Probe3
Lunar-QA	**0.82**†	0.68	0.79
Astro-QA	**0.68**	0.57	0.64
MATH	0.63	**0.78**†	0.73
GSM8K	0.72	**0.84**†	0.79

E.3 Consistency with Human Adjudication

We benchmark LDP difficulty predictions against domain expert assessments across six datasets (please see Figure 8). The difficulty taxonomy enforces a tripartite classification into Easy, Medium, and Hard tiers, establishing a random-guess baseline of 33.3%. Empirical results show that LDP consistently achieves parity with expert human evaluations and surpasses them on high-difficulty tasks within the IOAA benchmark. All correlation metrics are statistically significant ($p < 0.001$). These findings confirm that, after minimal training, LDP can effectively substitute for human adjudication in complex difficulty estimation tasks.

E.4 Industrial Impact

As discussed in Related Work, applying LLMs to space-related tasks remains a growing but still niche trend. Although many recent studies are viewed by the broader AI community as incremental, or primarily as extensions of existing methods to new domains, their practical value should not be overlooked. Building on this perspective, the development of Lunar-R1 for upcoming lunar missions revealed several engineering challenges during training and testing. These challenges underscored the need for efficient reasoning in this context, which motivated our integration of LDP with GRPO reward modeling. Various countries are actively preparing for the next phase of long-term lunar exploration, and leading space agencies including NASA, ESA, and CNSA are actively exploring language models for edge applications. Moreover, we have deliberately avoided overly narrow domain constraints, ensuring that our approach remains broadly applicable. The proposed LDP probe is conceptually straightforward, and together with the standard CPT/SFT/RL training pipeline, it follows an industry-standard practice.

F Data Integrity Validation

F.1 Recursive Bias Analysis

Lunar-QA was developed in collaboration with five experts from the National Space Administration, with its design aligned with the operational requirements of the upcoming International Lunar Research Station (ILRS) mission. Each sample was manually annotated with both a difficulty level and a task domain. The difficulty levels (easy, medium, and hard) correspond to progressively complex reasoning demands, and each instance underwent several rounds of standardized expert cross-review (IAA > 0.92). The task domain is confined to five core scenarios: *collaboration, navigation, energy, collection, and communication.*

We restrict synthetic data to the SFT and RL stages of Lunar-R1. Continued pre-training (CPT) directly ingests lunar-domain corpora without structured question-answer formatting. Meanwhile, Lunar-QA was independently constructed from held-out materials by the same expert panel, following an MMLU-style format and excluding all synthetic generation. In fact, synthetic data are widely used in domain-specific NLP to circumvent costly manual annotation, a common practice in fields such as medicine and scientific reasoning. However, they risk introducing template-level regularity and recursive distributional drift.

To quantify this, we compare 2,000 generated instances against 2,000 expert-authored seeds across three metrics: KL divergence (lexical shift), Self-BLEU (textual diversity), and embedding similarity (semantic alignment). Results are reported in Table 4. The augmented dataset exhibits minimal KL divergence (from 0.01 to 0.04), maintains stable diversity with Self-BLEU scores ranging from 0.78 to 0.83, and achieves high semantic consistency with an embedding similarity of 0.91. These results confirm that the generation pipeline preserves the structural integrity of the expert seeds and effectively mitigates recursive bias.

Table 4: Distributional feature validation comparing expert-authored seeds and augmented samples.

Metric	Human	Augmented
KL Divergence	0.01	0.04
Self-BLEU	0.78	0.83
Similarity	0.97	0.91

F.2 Contamination Risk Analysis

To confirm the absence of leakage, we used the open-source audit toolkit described by Li et al. (2024) to examine all datasets items. In fact, lunar-specific data are scarce in open-source corpora. Common pre-training sources (COMMON CRAWL (Patel, 2020), THE PILE (Gao et al., 2020), C4 (Zhu et al., 2023), GitHub (Cosentino et al., 2016), and Wikipedia (Guo et al., 2020)) and benchmarks like MMLU contain few related queries, whereas Lunar-QA is drawn from authentic mission, training logs, and public archives. Scanning the five corpora in Table 5, we flagged 42 potential overlaps (0.84% of 5,000 items), all confirmed manually to be domain-standard terminology or public parameters. No actual contamination was found, giving a final leakage rate of **0.0%**.

Table 5: Contamination verification results.

Corpus	Flagged	Verified
Common Crawl	7	0
The Pile	4	0
C4	5	0
MMLU	11	0
Wikipedia	15	0
Total	**42**	**0**

G Future Work

Recent studies, including Lunar-Twins (Xiao et al., 2025) and Space-LlaVA (Foutter et al., 2024), have advanced the application of LLMs in space missions. However, these projects are still insufficient to support comprehensive end-user deployment and intelligent human-machine collaboration. From the current perspective, model deployment for space exploration that relies solely on VLM remains extremely challenging because applicable multimodal training data is scarce. We are actively addressing this gap by leveraging UE5 and Unity to build high-fidelity lunar environments. This ongoing project aims to collect embodied data and train VLA and VLN models for the digital twin simulation of real environments (Ding et al., 2024). Additionally, NLP remains an important part of human-machine interaction in lunar exploration missions. End-to-end autonomous navigation or real-time obstacle avoidance based on visual SLAM is one of the directions under simultaneous development.

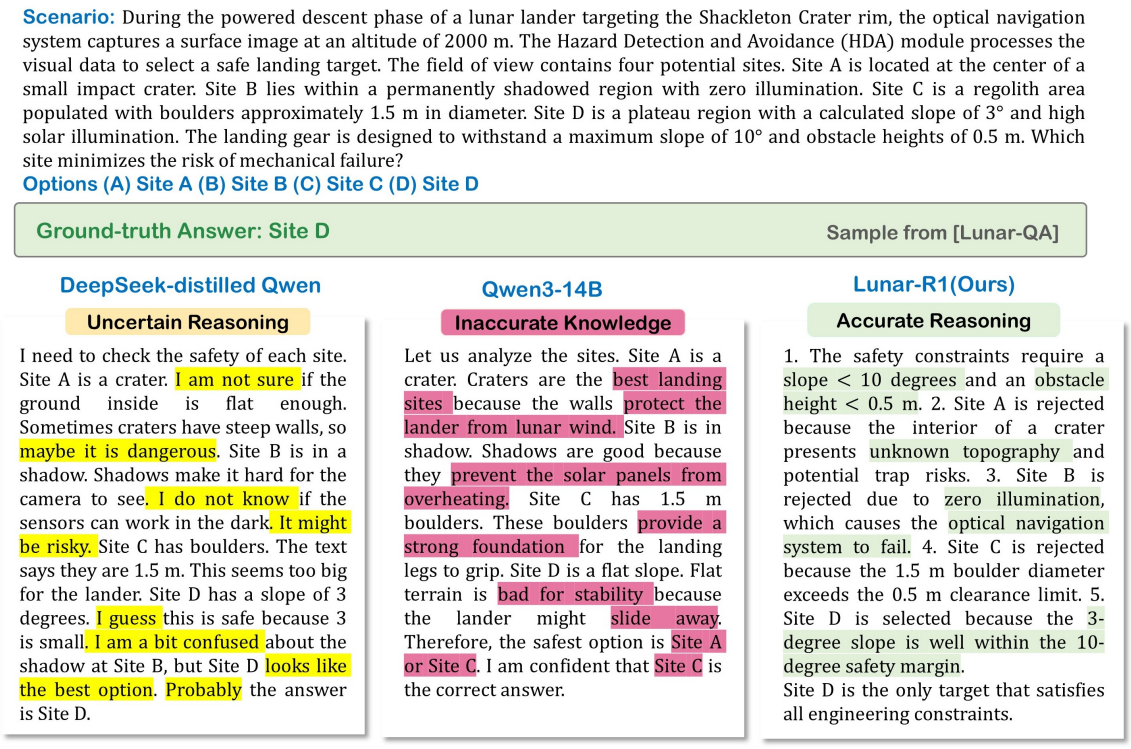

Figure 11: **Qualitative case study on the Lunar-QA benchmark.** DeepSeek-R1-distilled-Qwen-14B manifests reasoning uncertainty and fails to reject hazardous sites explicitly. Qwen3-14B exhibits knowledge hallucinations regarding lunar environmental physics such as wind protection. Lunar-R1 accurately derives the correct decision by sequentially validating slope and obstacle parameters against mechanical limits.

Table 6: LLM-as-a-Judge Prompt for Preference Evaluation

System prompt
Please act as an impartial judge and evaluate the quality of the responses provided by two AI assistants to the user question displayed below. You should choose the assistant that follows the user's instructions and answers the user's question better. Your evaluation should consider factors such as scientific accuracy, reasoning depth, and safety protocol adherence. Begin your evaluation by comparing the two responses and provide a short explanation. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Be as objective as possible. After providing your explanation, output your final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]" if assistant B is better, and "[[C]]" for a tie.
User Prompt
<|The Start of Assistant A's Conversation with User|> ### User: {QUESTION_1} ### Assistant A: {ANSWER_OF_MODEL_A} <|The End of Assistant A's Conversation with User|> <|The Start of Assistant B's Conversation with User|> ### User: {QUESTION_1} ### Assistant B: {ANSWER_OF_MODEL_B} <|The End of Assistant B's Conversation with User|>

Table 7: System Prompt for Lunar-R1 Inference and GRPO Training

System prompt
You are Lunar-R1, an advanced AI assistant specialized in lunar science and mission operations. Your knowledge is grounded in peer-reviewed planetary science literature and official mission logs. Response Format Constraints: 1. You must begin every response with a `<think>...</think>` block. 2. Inside the thinking block, explicitly assess task complexity. 3. Conclude with the final answer enclosed in `<answer>...</answer>`. 4. Do not execute or endorse kinetic commands without safety verification. Tone and Safety Guidelines: - Maintain scientific rigor and objectivity. - Use SI units for all physical quantities. - If required data is unavailable, explicitly state Data not available.
User prompt
{USER_QUERY}