

# Video Reasoning Without Training

Deepak Sridhar<sup>1,2\*</sup> Kartikeya Bhardwaj<sup>1\*</sup> Jeya Pradha Jeyaraj<sup>1</sup> Nuno Vasconcelos<sup>2</sup>  
Ankita Nayak<sup>1</sup> Harris Teague<sup>1</sup>

<sup>1</sup>Qualcomm AI Research<sup>†</sup> <sup>2</sup>University of California, San Diego

desridha@ucsd.edu, kbhardwa@qti.qualcomm.com

## Abstract

*Video reasoning using Large Multimodal Models (LMMs) relies on costly reinforcement learning (RL) and verbose chain-of-thought, resulting in substantial computational overhead during both training and inference. Moreover, the mechanisms that control the thinking process in these reasoning models are very limited. In this paper, we use the entropy of the model’s output distribution as a signal to study and guide reasoning behavior. We discover that high-quality models exhibit a characteristic pattern of micro-exploration and micro-exploitation cycles, followed by a later entropy peak (i.e., longer thinking) and a lower final entropy, indicating more deliberate exploration and confident convergence (i.e., avoid excessive randomness while the model is exploring or thinking through an answer). We then use these novel, theoretically-grounded insights to introduce *V-Reason (Video-Reason)*, an inference-time optimization method that adapts the value cache of the LMM through a lightweight, trainable controller. Our proposed controller is guided by an entropy-based objective, to tune the model’s behavior directly at inference, without using any RL or supervised fine-tuning. Our experiments show that *V-Reason* significantly outperforms the base instruction-tuned models on many video reasoning datasets, narrowing the gap with RL models to within **0.6%** accuracy on average. We achieve this without any training, while offering efficiency benefits: *V-Reason* uses **58.6%** fewer tokens than the RL model.*

## 1. Introduction

Reasoning with generative AI models, such as Large Language or Large Multimodal Models (LLMs/LMMs), has gained substantial attention recently. This capability is implemented by asking the model to “think” about a problem, before making a final recommendation, and can be accomplished by several approaches, including Chain-of-Thought

(CoT) [36], supervised fine-tuning with CoT (CoT-SFT) [10, 25], or reinforcement learning (RL) with a *thinking-before-answering* format [14, 28]. Although initial progress was shown mainly for LLMs, such ideas have now been extended to video reasoning problems [8, 10, 21, 33, 46] by exploiting LMMs. Although successful, CoT-SFT, and RL-based methods tend to be highly computationally intensive, both for training and inference, due to the long thinking traces that they tend to produce. These costs are particularly exacerbated for video, due to the high resolution and multiple frames involved in the reasoning process.

Recently, RL-based reasoning has been viewed as a sampling process [31, 48] to more effectively *search* for the reasoning traces from the pretrained knowledge of the baseline model. We thus ask whether this search can be elicited during inference in a *training-free* way. We note that we do not intend to entirely replace RL or fundamentally expand reasoning ability without training, which is an open question even for RL-based methods [43]. In this paper, we seek to address the problems stated above by considering the following **key questions**:

1. Can inference-time metrics characterize the thinking process of video reasoning models? If yes, can these metrics differentiate between higher- and lower-quality LMMs?
2. Can such metrics be used to formulate novel inference-time optimization objectives for video reasoning without requiring additional model training?

To answer these questions, we first analyze the model’s output distribution entropy at generation step  $t$  computed as  $H_t = -\sum_{i \in \mathcal{V}} p_t^i \log p_t^i$  ( $\mathcal{V}$  refers to vocabulary of the model) for instruction-tuned LMMs of various sizes, as shown in Fig. 1(a). This analysis reveals two broad trends: (i) all models exhibit a pattern of increasing and then decreasing entropy as tokens are generated, and (ii) larger, more accurate models exhibit both a *later entropy peak and a lower final entropy*, indicating more confident reasoning (see Fig. 1(a) and its inset for Qwen-VL-Instruct models).

The first trend above can suggest a formal definition of the “thinking” in terms of output distribution entropy. As the model starts generating a response, it seems to be un-

\*Equal contribution. Work done when Deepak Sridhar was an intern at Qualcomm AI Research.

<sup>†</sup>Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

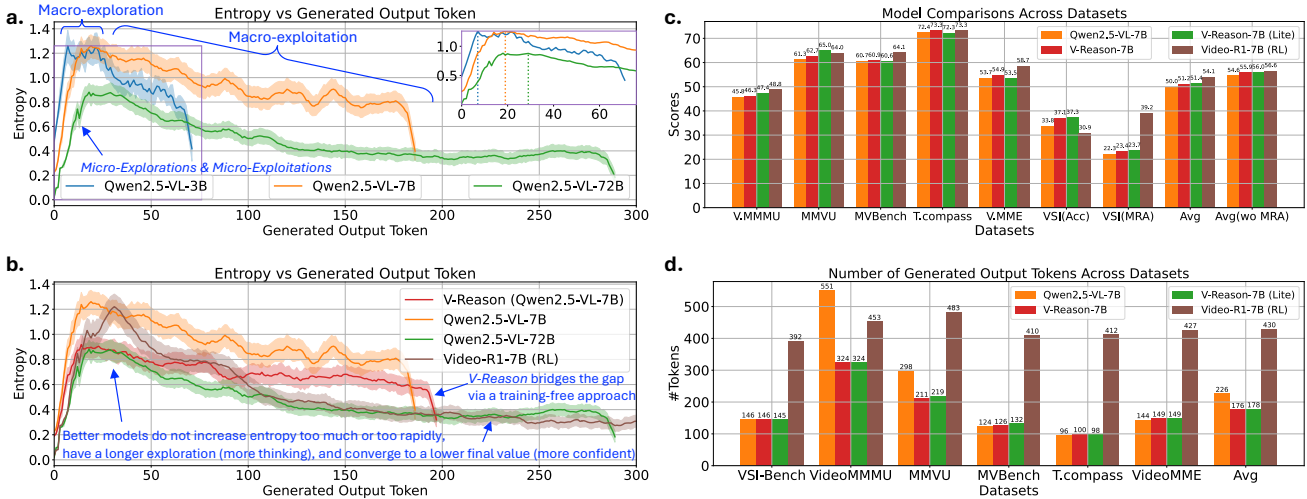


Figure 1. *V-Reason* Overview: (a) Entropy of the output distribution averaged over the MMVU [49] dataset of 625 videos. We see clear macro-exploration and macro-exploitation phases with bigger, more accurate models showing lower overall entropy (lower and later peak, followed by a lower final entropy during the macro-exploitation). We use these key insights to adapt a model’s behavior in a training-free way using an inference-time optimization technique. (b) Applying *V-Reason* on Qwen2.5-VL-7B-Instruct makes its entropy behave more similarly to the larger or the RL-trained Video-R1-7B model. (c) *V-Reason* bridges the accuracy gap with Video-R1-7B to within 0.6% for tasks well-represented in the pretrained model’s knowledge (Avg. without MRA). (d) *V-Reason* also significantly reduces the total output tokens compared to all models due to a dedicated entropy minimization phase.

certain and *searches* through multiple solution trajectories, which can explain the increase in its output entropy. We denote this as the *macro-exploration* phase. As the generation progresses, the model seems to start identifying the correct thinking thread, and becomes increasingly certain about a solution, resulting in the gradual reduction in the entropy of its output. We denote this as the *macro-exploitation* phase.

The second trend seems to suggest that entropy should *not* increase too rapidly during the macro-exploration phase. In fact, all models go through a series of *micro-exploration* and *micro-exploitation* cycles (characterized by small increases and decreases of entropy) during both macro phases of the thinking process; see Fig. 1(a) shaded regions. A delayed entropy peak can suggest that better reasoning models explore more alternative answers, leading to higher accuracy [14, 28, 36]. In this context, more and/or longer cycles of micro-exploration and micro-exploitation can lead to “longer thinking,” with lower and delayed entropy peaks and lower final entropy. Fig. 1(b, brown line) shows that the above two observations also hold for an RL-trained Video-R1-7B model [10]. This model has a slightly lower and much later entropy peak than the Qwen2.5-VL-7B-Instruct baseline model, which was used to train Video-R1-7B, and the final entropy is very close to that of the significantly larger Qwen2.5-VL-72B-Instruct model. In contrast, smaller models (e.g., 3B) peak early (i.e., think less) and converge prematurely to lower entropy leading to confident but incorrect answers (see Fig. 1(a)). It suggests that a shorter thinking phase can lead to lower reasoning accuracy.

Building on these observations, we propose *V-Reason*, a *training-free* inference-time optimization method that directly modulates the micro-exploration and micro-

exploitation behavior of the baseline instruction-tuned models to enhance their thinking capabilities. *V-Reason* introduces a small, trainable controller that adapts the LMM value cache using an entropy-based objective, requiring no supervision, dataset, or RL signal. The objective encourages more pronounced cycles of micro-exploration and micro-exploitation, by inducing the model to more strongly increase/decrease entropy during these cycles, followed by a final entropy minimization phase. This process prevents entropy from rising too fast during macro-exploration and enables the model to achieve a lower final entropy during macro-exploitation, thus making the baseline model behave more like a stronger reasoning model (see Fig. 1(a,b)). In effect, it encourages the model to think longer but more efficiently by exploring alternatives without producing unnecessary tokens. This mirrors findings from prior studies [34, 37, 38] showing that shorter yet well-targeted reasoning chains can equal or outperform longer, verbose ones. To enhance efficiency, we further introduce a “lite” variant, *V-Reason(Lite)*, which reduces memory and computational overhead by evicting 50% of the lowest-norm video tokens from the KV-cache.

Our results demonstrate that *V-Reason* and *V-Reason(Lite)* bridge the gap between baseline instruction tuned models and RL-trained models in terms of accuracy (see Fig. 1(c)). We empirically observe that this approach is most effective when the solution lies within the pretrained model’s knowledge space (e.g., classification tasks we study) but requires better search strategies to surface it. For tasks where this knowledge underrepresented in pretraining (e.g., regression tasks like VSI-MRA), training-based approaches remain more

effective. Moreover, our dedicated entropy minimization phase enables the final solution trajectory to converge faster, thus producing considerably fewer output tokens on average compared to the RL models (see Fig. 1(d)) which also helps the inference times. Thus, `V-Reason` and `V-Reason(Lite)` bridge the gap with the RL-trained model while producing significantly fewer output tokens. In summary, the paper makes the following **key contributions**:

1. To our knowledge, the problem of *video reasoning without training* has not been previously addressed in the literature. We are the first to introduce a training-free, purely inference-time optimization method for video reasoning without SFT or RL.
2. We propose `V-Reason` that enhances the micro-exploration and micro-exploitation cycles of the baseline instruction-tuned models to achieve better accuracy. We also provide simple theoretical results for our method.
3. We show that `V-Reason` with our proposed entropy based objective induces a lower and delayed entropy peak during macro-exploration and a lower final entropy during macro-exploitation, similar to the patterns observed for the reasoning models trained by RL (see Fig. 1(b)).
4. Extensive experiments on six video reasoning benchmarks show that `V-Reason` achieves an average improvement of **1.4%** over the base model, narrowing the gap to within **0.6%** of the RL-trained Video-R1-7B model (see Fig. 1(c)). We further show gains across model sizes ranging from 3B to 32B and even up to 72B LLMs. We also demonstrate that `V-Reason` is complementary to multiple SOTA decoding methods.
5. Our approach can lead to more efficient reasoning by significantly reducing the total number of output tokens generated (see Fig. 1(d)). `V-Reason` produces **21.4%** fewer tokens on average than the base Qwen2.5-7B-Instruct model, and **58.6%** fewer tokens on average than the RL-trained Video-R1-7B model. This means that its wall-clock inference time is competitive to the base model and up to **37%** lower than Video-R1-7B on average.

## 2. Related Work

**Reasoning in Large Language Models.** Reasoning in LLMs can be achieved by chain-of-thought prompting, instruction-tuning with CoTs, or reward-based fine-tuning with RL. Existing work on prompting primarily relies on eliciting better CoT reasoning paths from the model [18, 41, 50]. While these methods have achieved high accuracies, few-shot prompting techniques are task-specific, less generalizable and require manual prompt designs for each task. Better prompting techniques require extensive prompt engineering and result in inconsistent performances [51]. Overall, prompting techniques are limited by model-specific and task-specific tuning [39] making them less favorable. Recent works endeavor to improve the CoT prompting by verifi-

cation [13] that verifies and controls the intermediate steps generated by the model. Such methods still require CoT prompting and are computationally intensive due to the additional verification steps involved.

Instruction-tuning and reward-based fine-tuning are alternative ways to elicit reasoning in LLMs when additional compute is available for supervision [9, 16, 26]. However, these techniques require supervised CoT data and expensive RL stages to make the model compliant to produce the reasoning or thinking process in specified formats for easy extraction of the answers. Different from the above methods, we seek an efficient framework to elicit reasoning in LLMs at inference without any supervised data or training.

**Video Reasoning.** Video Reasoning methods have been introduced recently [6, 10] inspired by the success of LLM reasoning. Video-R1 [10] introduces a temporal GRPO loss to specifically improve temporal reasoning capabilities along with a new dataset for training. VideoChat-R1 [21] introduces a chat model with spatio-temporal reasoning abilities by training with GRPO and rule-based rewards. TinyLLaVA [45] shows that reasoning can be effective even for smaller models, using a Qwen-3B-VL model trained with standard GRPO and RL-based reward losses. All of the above methods rely on expensive training to elicit reasoning in LLMs for videos; for instance, training TinyLLaVA on 50K samples takes  $\sim 3$  days on 4 A100 GPUs, and the cost scales prohibitively for larger models (7B, 32B). To overcome this, we propose an efficient framework that leverages inference-time optimization to elicit the reasoning traces from pretrained LLMs, achieving higher accuracy with fewer output tokens compared to RL-trained models.

**Inference-time Reasoning Methods.** Inference-time optimization methods [5, 29] have gained popularity in diffusion models for improving control and consistency. Recent works have explored eliciting reasoning capabilities from LLMs at inference time [12, 35], aiming to reduce computational cost and improve interpretability. Decoding strategies such as CoT-Decoding [35] modifies token selection to surface latent reasoning traces, while ThinkLogit [47] manipulates logits with guidance from a smaller preference model to induce longer reasoning chains. In parallel, sampling-based methods such as min-p [27] and the concurrent approach top-h [2] restrict candidate tokens based on probability thresholds or rank cutoffs, improving fluency but without explicitly targeting reasoning. Our method is orthogonal to these approaches: rather than filtering outputs, we optimize the model’s intrinsic token distributions during inference and show consistent improvements even when combined with min-p and top-h sampling-based methods.

Other line of works utilize steering to modify the model’s behavior for reasoning tasks [1, 4]. ASC [1] modifies the hidden states of the model to compress CoT traces by relying on a reasoning-trained model to distinguish concise from

verbose reasoning. KV-Cache Steering [4] presents a one-shot intervention in the key-value cache to induce reasoning in small LLMs with steering vectors derived from GPT-4o [17]. In contrast to these works that have *indirect reliance on a reasoning-trained model*, we propose an inference-optimization technique that modulates the value-cache to elicit reasoning using the model’s own entropy as intrinsic feedback *without any reliance on external model or data*.

### 3. Proposed Approach: V-Reason

In this section, we describe the proposed V-Reason, its inference-time optimization objectives, and supporting theoretical results. We then address practical aspects, including redundancy reduction in video tokens to lower memory costs, and introduce V-Reason (Lite) for improved efficiency.

#### 3.1. Inference-time Optimization

Modifying the reasoning behavior of a pre-trained LMM requires two components: a set of parameters which are modified or added to the model to elicit reasoning, and an optimization objective, to optimize those parameters. As discussed in Section 1, the key goals for V-Reason are to: (a) decrease the rate of growth of the output distribution entropy during macro-exploration, by *controlling* the model behavior so as to promote more pronounced cycles of micro-exploration and micro-exploitation during the output generation, and (b) reduce the final entropy during macro-exploitation. To accomplish these objectives, we propose a value-cache controller and a novel inference-time optimization objective.

**Controller Parameters.** We propose to augment the model with the *Value-Cache Controller* shown in Fig. 2(a). This controller, denoted as  $\Delta V$ , is a small, trainable parameter added to the value cache  $\mathbb{V}_L$  of the *last* decoder layer of the model, specifically at the video token locations. All other model layers remain frozen and no modifications are applied to the input or output text tokens. In our experiments, tuning only the value cache of the last layer was sufficient and modifying earlier layers or tuning key-value cache did not yield additional improvements. The controller  $\Delta V$  is initialized to zero and updated at every  $k^{th}$  generated output token ( $k > 1$ ) via the inference-time optimization method discussed below. Note that no optimization is performed for the first token, as that is when the KV-Cache prefilling happens for all layers. To prevent the controller from destabilizing the pretrained model, we introduce the normalization

$$\mathbb{V}_L^{new} = \frac{\mathbb{V}_L + \Delta V}{\|\mathbb{V}_L + \Delta V\|} \cdot \|\mathbb{V}_L\|. \quad (1)$$

This normalization preserves the original magnitude  $\|\mathbb{V}_L\|$  of the cache vector, ensuring that the controller  $\Delta V$  introduces only a directional update. This helps maintain a stable

forward pass, ensuring consistent output token generation. This normalization is inspired by well-known methods like Weight Normalization [30, 32], which have been shown to have good optimization properties and are beneficial for recurrent and generative models.

**Optimization Objective.** In Section 1 and Fig. 1(a), we suggested that the effectiveness of a reasoning model is related to the entropy of its output token distribution. While all reasoning models exhibit a period of macro-exploration, where entropy increases, and macro-exploitation, where it decreases, better models have a macro-exploration stage characterized by lower and delayed entropy maxima. We further posited that this is largely driven by cycles of micro-exploration and micro-exploitation, which prevent the entropy from increasing or decreasing too rapidly. We interpret these cycles as periods where the model temporarily increases the output entropy (exploration) to allow alternative reasoning paths, needed to escape from a current unpromising path. The model then pursues a new path in more detail (exploitation), leading to a decrease of entropy and the potential realization that this new path is itself not promising. The cycle is then repeated. We hypothesize that stronger reasoning models are more decisive in their patterns of micro-exploration and exploitation, which leads to more and/or stronger cycles, thus reducing the rate of *macro* entropy increase. This leads to lower and delayed entropy peaks. It follows that the reasoning power of a model should increase if it is encouraged to have more *vigorous* micro-exploration/exploitation cycles. After reaching the entropy peak of macro-exploration, the model switches to macro-exploitation, where it pursues a reasoning path in detail to produce an answer, which leads to a decrease of the output entropy. We note that the *pattern of delayed peaks and lower final entropy and richer micro-cycles are signatures of better reasoning than absolute entropy values*. In this work, we propose to reinforce this behavior by optimizing the value cache controller  $\Delta V$  with the *Entropy Switching Loss*:

$$\mathcal{L}_{switch}(\Delta V) = -\alpha_k H_k = \alpha_k \sum_{i \in |\mathcal{V}|} p_k^i(\Delta V) \log(p_k^i(\Delta V)) \quad (2)$$

where  $p_k$  is the output distribution (softmax after the LM-Head) for every  $k^{th}$  token generated ( $k > 1$ ),  $H_k$  the entropy of this distribution, and  $\alpha_k \in \{-1, +1\}$  is a coefficient that switches between  $-1$  and  $+1$ . The minimization of this loss encourages an increase in the entropy (micro-exploration) when  $\alpha_k = 1$  and a decrease (micro-exploitation) when  $\alpha_k = -1$ . Hence, setting  $\alpha_k = 1$  ( $\alpha_k = -1$ ) during the micro-exploration (micro-exploitation) periods, encourages the model to be more decisive in its micro-exploration/exploitation cycles. It is also possible to explore other behaviors, e.g., using this procedure to reinforce micro-cycles during macro-exploration, followed by minimizing entropy alone ( $\alpha_k = -1$ ) during macro-exploitation.

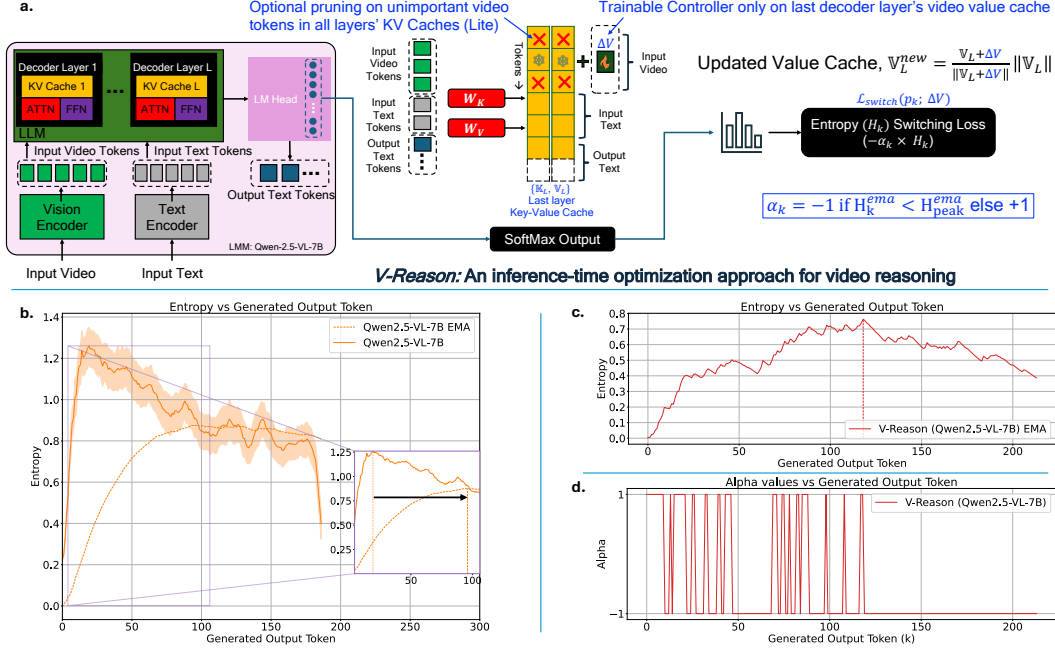


Figure 2. (a) Proposed approach for video reasoning in a training-free manner using entropy-based objective. V-Reason uses an inference optimization method to modulate the values cache of the last decoder layer with an entropy switching loss ( $\mathcal{L}_{switch}$ ) to improve the video reasoning performance. (b) The average entropy plot for Qwen-2.5-VL-7B on the MMVU dataset along with its EMA. The inset depicts the shift in the entropy maxima for the EMA curve denoted by the black arrow (c) EMA entropy plot of V-Reason for a single sample that shows the micro-exploration and micro-exploitation within the macro-exploration phase before the entropy maxima and macro-exploitation phase after. (d) Plot showing the  $\alpha_k$  switching in V-Reason for the corresponding example in (c) that ensures bounded entropy updates without a rapid increase.

To implement this, we first compute the exponential moving average (EMA) of the entropy at each generation step  $t$  (different from  $k$ , which is the optimization step for the value-cache controller)

$$H_t^{ema} = \beta H_{t-1} + (1 - \beta) H_t \quad (3)$$

where  $t > 1$ ,  $\beta$  is a smoothing coefficient (set to 0.98), and  $H_0$  is the entropy of the first token which is a small value<sup>1</sup>. The EMA is a low-pass filtered version of the raw entropy, and thus much less noisy, as shown in Fig. 2 (b). It achieves a good trade-off between oscillating too much, due to noise, and switching between increasing and decreasing entropy during micro-cycles, as shown in Fig. 2 (c). Also, because it grows much slower than the raw entropy, following the EMA naturally leads to a lower and delayed entropy peak, as shown in Fig. 2 (b). The switching coefficient  $\alpha_k$  is then defined to follow the EMA,

$$\alpha_k = \begin{cases} +1 & \text{if } H_k^{ema} \geq H_{peak}^{ema} \\ -1 & \text{if } H_k^{ema} < H_{peak}^{ema} \end{cases} \quad (4)$$

where,  $H_k^{ema}$  is the EMA at the current step, and  $H_{peak}^{ema}$  the maximum value of EMA observed before step  $k$ . This is illustrated in Figure 2 (d). It encourages the entropy to

<sup>1</sup>The baseline instruction-tuned models are certain about the very first predicted token; it is usually just the `<think>` token, even without RL or CoT-SFT, because of the instruction we give to the model.

(i) increase when the EMA is larger than the last peak, i.e., the EMA is increasing, and to (ii) decrease otherwise, i.e., the EMA is decreasing, therefore reinforcing the natural micro-cycles of the model. Once the EMA reaches a global maximum,  $\alpha_k$  becomes  $-1$  and macro-exploitation begins. This global maximum of entropy can also be seen as a more formal definition of the end of the “thinking” process. A detailed description of our method is given in Algorithm 1.

Fig. 2 (c) shows the EMA entropy plot of V-Reason for a single sample. It is clear that there are more and stronger local minima and maxima depicting the micro-exploration/exploitation cycles before the entropy maxima. This slows the entropy growth during macro-exploration, leading to a delayed peak and substantially more exploration than by the original model. Once the global maximum of the EMA is reached,  $\alpha_k$  becomes  $-1$  and the model enters the macro-exploitation stage, where it is encouraged to decrease entropy until it arrives at a solution. Overall, the optimization promotes 1) more and/or longer cycles of micro-exploration and micro-exploitation during the macro-exploration stage, which lead to “longer thinking,” with lower and delayed entropy peaks, and 2) a stronger emphasis on entropy minimization during the macro-exploitation stage, which leads to faster convergence to a lower final entropy.

We observe that the optimization of V-Reason encourages the model to arrive at the final solution significantly faster than CoT-SFT and RL models, which often produce verbose outputs. This can be seen in Fig. 1 (d). Since com-

putation is tied to the length of the output sequence, this also results in significantly more efficient inference than those models. Hence, despite the extra computation needed for the optimization, V-Reason has more efficient inference overall (section 3.2). Finally, since V-Reason exploits the natural variation in entropy, it adaptively determines how much exploration and exploitation is required by each sample. This makes it robust and adaptable to various datasets and types of video reasoning problems (see Section 4).

**Theoretical Guarantees.** We provide theoretical guarantees that the entropy updates induced by our Entropy Switching Loss remain stable and that our EMA-based objective bounds the oscillations in entropy. The formal statements are below, with assumptions and proofs discussed in Supplementary A.

**Proposition 1** (Bounded entropy updates). *Under mild smoothness and boundedness assumptions, one gradient step of size  $\eta$  on the Entropy Switching Loss changes entropy by at most*

$$|H_{t+1} - H_t| \leq \eta C + o(\eta),$$

and the process  $\{H_t\}$  remains within the compact interval  $[0, \log n]$ .

**Proposition 2** (EMA smoothing bounds oscillations). *For  $\beta \in (0, 1)$  close to 1, the EMA acts as a low-pass filter: (i) it attenuates high-frequency fluctuations of  $H_t$ , (ii) delays the attainment of entropy maxima, and (iii) enforces bounded oscillations by switching  $\alpha_k$  to  $-1$  once a new global EMA maximum is reached.*

### 3.2. Efficiency Considerations: V-Reason (Lite)

Video reasoning with LMMs can have high GPU memory costs due to a large number of input video tokens. Adding inference-time optimization to these models at first sight can seem inefficient, as it can further increase inference costs. However, V-Reason has several properties that counteract this hypothesis. First, the controller is only added to the decoder cache of the last model layer. This significantly reduces the memory overhead of storing activations for back-propagation, which reduces to the trainable controller  $\Delta V$  and a few feature maps (last decoder layer’s value cache, attention output, feedforward layers, and LM-Head). Second, and most important, because V-Reason usually arrives at the final solution with significantly less tokens as shown in Fig. 1(d), both its inference time and computation are much lower than models trained to think.

Nevertheless, we explore an additional avenue for efficiency. Before performing the V-Reason optimization, we *optionally* prune 50% of the video tokens from the KV-Cache of all decoder layers, a variant we refer to as V-Reason (Lite). This significantly reduces the KV-Cache overhead and also halves the size of the trainable controller. Interestingly, we found that for some datasets this also slightly improves V-Reason reasoning performance

(perhaps by reducing noise due to unimportant video tokens). To prune out unimportant video tokens, we measure the mean value of the  $l_2$  norm of video tokens across all value caches and eliminate the lowest 50% video tokens from both Key and Value Caches of all decoder layers. The trainable controller is then only added to the remaining video tokens in the last decoder layer. The new value update is  $\mathbb{V}_L^{new} = \frac{\mathbb{V}_L^{pruned} + \Delta V}{\|\mathbb{V}_L^{pruned} + \Delta V\|} \cdot \|\mathbb{V}_L\|$ , which still maintains the magnitude of the unpruned video value cache from equation 1. We empirically find that this reduces the error due to pruning and enables the V-Reason (Lite) models to achieve much higher accuracies than when the value cache norm is altered. Algorithm 2 in Supplementary provides the pseudo-code for the lite variant.

## 4. Experiments

**Implementation Details.** All experiments use pytorch version 2.5.1+cu121, transformers version 4.52.4, and a single NVIDIA-A100 GPU. Following [10], we use multinomial sampling with (temperature=0.1, top-p=0.001 i.e., deterministic) for our experiments unless otherwise noted. See Supplementary B for more details.

**Video Reasoning.** We evaluate V-Reason on the Qwen2.5-VL-Instruct [3] model series under 16/32 frames settings (from [10]) and maximum video pixels  $px \times 28 \times 28$  with  $px=256/128$ , respectively. Similar to [10], V-Reason is evaluated across 6 video reasoning benchmarks, covering two tasks, Multiple-Choice QA and Regression, evaluated by classification accuracy and Mean Relative Accuracy (MRA) respectively. We report the average accuracy with and without MRA to illustrate the model’s performance across different task formulations.

### 4.1. Video Reasoning Benchmark Results

Table 1 presents a comparison of V-Reason with Qwen2.5-VL-Instruct baselines and the RL-trained Video-R1-7B across multiple video reasoning benchmarks. Green brackets show the gain of the V-Reason model over the baseline, with negative gains in red. Both (at least one) versions of V-Reason improve the baseline performance for 15/18 (18/18) model/dataset combinations. Furthermore, the gain is of at least **0.9 points** for 19/36 combinations and can be as high as **3.9 points**. In many cases, these gains are a substantial part of the gap between the baseline and the RL-trained model. For example, for MMVU and 7B-256px models the 63.4 point accuracy of V-Reason (Lite) brings the relatively lower 59.5 point baseline close to the 64.2 point accuracy of the Video-R1. For the 128 px model, V-Reason even surpasses Video-R1 (**65.0 vs. 64.0**). This model also matches Video-R1 on TempCompass (73.3 each), and nearly closes the gap on VideoMMU (47.4 vs. 48.8). These very significant gains show that the baseline model

Table 1. Comparison of performance of different models on video reasoning benchmarks. #F denotes the number of frames and px denotes the maximum video pixels used, px×28 × 28.

| Model                     | #F/px  | VSI-Bench<br>(Acc/MRA)<br>[40]      | VideoMMMU<br>[15]  | MMVU<br>(mc)<br>[49] | MVBench<br>[20]    | TempCompass<br>[24] | VideoMME<br>(wo sub)<br>[11] | Avg                | Avg<br>(wo mra)    |
|---------------------------|--------|-------------------------------------|--------------------|----------------------|--------------------|---------------------|------------------------------|--------------------|--------------------|
| GPT-4o [17]               | –      | 34.0                                | 61.2               | 75.4                 | –                  | –                   | 71.9                         | –                  | –                  |
| LLaMA-VID [22]            | –      | –                                   | –                  | –                    | 41.9               | 45.6                | –                            | –                  | –                  |
| VideoLLaMA2 [7]           | –      | –                                   | –                  | 44.8                 | 54.6               | –                   | 47.9                         | –                  | –                  |
| LongVA-7B [44]            | –      | 29.2                                | 23.9               | –                    | –                  | 56.9                | 52.6                         | –                  | –                  |
| VILA-1.5-8B [23]          | –      | 28.9                                | 20.8               | –                    | –                  | 58.8                | –                            | –                  | –                  |
| Video-UTR-7B [42]         | –      | –                                   | –                  | –                    | 58.8               | 59.7                | 52.6                         | –                  | –                  |
| LLaVA-OneV-7B [19]        | –      | 32.4                                | 33.8               | 49.2                 | 56.7               | –                   | 58.2                         | –                  | –                  |
| Qwen2.5-VL-3B [3]         | 32/128 | 24.3 (31.6/17.0)                    | 32.3               | 49.3                 | 52.5               | 28.1                | 48.1                         | 37.0               | 40.3               |
| <b>V-Reason-3B (Lite)</b> | 32/128 | <b>26.3 (32.2/20.4) [+0.6/+3.4]</b> | <b>33.9 [+1.6]</b> | <b>50.9 [+1.6]</b>   | <b>53.2 [+0.7]</b> | <b>29.1 [+1.0]</b>  | <b>49.0 [+0.9]</b>           | <b>38.3 [+1.3]</b> | <b>41.3 [+1.0]</b> |
| <b>V-Reason-3B</b>        | 32/128 | <b>24.7 (31.9/17.5) [+0.3/+0.5]</b> | <b>33.2 [+0.9]</b> | <b>50.2 [+0.9]</b>   | <b>52.9 [+0.4]</b> | <b>30.4 [+2.3]</b>  | <b>48.8 [+0.7]</b>           | <b>37.9 [+0.9]</b> | <b>41.2 [+0.9]</b> |
| Qwen2.5-VL-7B [3]         | 16/256 | 26.4 (31.4/21.4)                    | 47.6               | 59.5                 | 60.4               | 72.2                | 50.5                         | 49.0               | 53.6               |
| <b>V-Reason-7B (Lite)</b> | 16/256 | <b>27.9 (34.1/21.6) [+2.7/+0.2]</b> | <b>47.6 [+0.0]</b> | <b>63.4 [+3.9]</b>   | <b>60.8 [+0.4]</b> | <b>71.6 [-0.6]</b>  | <b>51.1 [+0.6]</b>           | <b>49.9 [+0.9]</b> | <b>54.6 [+1.0]</b> |
| <b>V-Reason-7B</b>        | 16/256 | <b>28.5 (34.5/22.6) [+3.1/+1.2]</b> | <b>47.8 [+0.2]</b> | <b>62.2 [+2.7]</b>   | <b>61.0 [+0.6]</b> | <b>72.3 [+0.1]</b>  | <b>51.1 [+0.6]</b>           | <b>50.2 [+1.2]</b> | <b>54.8 [+1.2]</b> |
| Video-R1-7B [10]          | 16/256 | <b>33.8 (30.5/37.0)</b>             | <b>47.8</b>        | <b>64.2</b>          | <b>63.9</b>        | 72.2                | <b>57.2</b>                  | <b>53.3</b>        | <b>56.0</b>        |
| Qwen2.5-VL-7B [3]         | 32/128 | 28.1 (33.8/22.3)                    | 45.8               | 61.3                 | 60.7               | 72.4                | 53.7                         | 50.0               | 54.6               |
| <b>V-Reason-7B (Lite)</b> | 32/128 | <b>30.5 (37.3/23.7) [+3.5/+1.4]</b> | <b>47.4 [+1.6]</b> | <b>65.0 [+3.7]</b>   | <b>60.6 [-0.1]</b> | <b>72.4 [+0.0]</b>  | <b>53.5 [-0.2]</b>           | <b>51.4 [+1.4]</b> | <b>56.0 [+1.4]</b> |
| <b>V-Reason-7B</b>        | 32/128 | <b>30.3 (37.1/23.4) [+3.3/+1.1]</b> | <b>46.3 [+0.5]</b> | <b>62.7 [+1.4]</b>   | <b>60.9 [+0.2]</b> | <b>73.3 [+0.9]</b>  | <b>54.9 [+1.2]</b>           | <b>51.2 [+1.2]</b> | <b>55.9 [+1.3]</b> |
| Video-R1-7B [10]          | 32/128 | <b>35.6 (30.9/39.2)</b>             | <b>48.8</b>        | <b>64.0</b>          | <b>64.1</b>        | <b>73.3</b>         | <b>58.7</b>                  | <b>54.1</b>        | <b>56.6</b>        |

already has a significant ability to reason, which RL brings to the surface, but can also be mostly unlocked by much less expensive inference-time optimization of V-Reason. We empirically observe that our training-free, search-based method is most effective when the base model already possesses the relevant underlying knowledge. For certain tasks like VSI-MRA which are underrepresented in the model’s learned knowledge space, V-Reason still obtains a **+1.4%** improvement over the baseline.

Overall, across model scales and input resolutions, V-Reason and V-Reason (Lite) have average gains in **[+0.9, 1.3]%** over Qwen2.5-VL, at the 3B scale. At the 7B scale, V-Reason/V-Reason (Lite) reaches 54.8%/54.6% (256 px) and 55.9%/56.0% (128 px) average accuracy without MRA, narrowing the gap to Video-R1-7B (56.6% at 128 px) to within **0.6%**.

**Impact of Frames and Resolution.** V-Reason is robust to different frame counts and resolutions, making it adaptable to resource-constrained settings. For a comparable compute budget (256 px/16 frames vs. 128 px/32 frames), accuracy improves with more frames. Consistent with this trend, V-Reason shows larger average gains at 32 frames (**+1.4%**) compared to 16 frames (**+1.0%**).

**Full vs. Lite.** Both Full and Lite V-Reason variants surpass the base model, with Lite performing comparably or slightly better in several cases (e.g., **+1.0%** at 3B and **+1.4%** at 7B for 128 px). The Full variant offers an additional **+0.2%** improvement at 256 px with 16 frames, suggesting that structural pruning is complementary to reasoning.

**Output Sequence Length.** As shown in Fig. 1(d), V-Reason substantially reduces output token length (**58.6% reduction** over Video-R1). Table 5 (Supplemen-

Table 2. Large model results on MMVU.

| Model           | MMVU               |
|-----------------|--------------------|
| Qwen-2.5-VL-32B | 69.0               |
| V-Reason-32B    | <b>72.0 [+3.0]</b> |
| Qwen-2.5-VL-72B | 72.6               |
| V-Reason-72B    | <b>73.0 [+0.4]</b> |
| Qwen-3-VL-8B    | 64.8               |
| V-Reason-8B     | <b>69.9 [+5.1]</b> |

Table 3. Optimization objective ablations.

| Method             | MMVU               |
|--------------------|--------------------|
| Qwen-2.5-VL-7B     | 61.3               |
| Min-Entropy (Lite) | 62.1 [+0.8]        |
| Max-Entropy (Lite) | 63.8 [+2.5]        |
| KV Cache (Lite)    | 61.8 [+0.5]        |
| Last two (Lite)    | 62.2 [+0.9]        |
| V-Reason (Lite)    | <b>65.0 [+3.7]</b> |

tary C) shows that this translates into significant latency savings of up to **67%** (**37%** on average). The fact that this holds for both Full and Lite variants indicates that the gains stem from the proposed optimization rather than pruning.

**Different backbone and Larger Models.** To test the scalability of V-Reason, we further evaluate it on Qwen3-VL and larger Qwen2.5-VL backbones (32B and 72B) using MMVU. Table 2 shows that gains hold across model families: e.g., **+5.1%** over MMVU for Qwen3-VL, showing that the method is not architecture-specific. V-Reason continues to provide significant gains on larger model sizes, e.g., **+3.0%** (**72.0 vs. 69.0**) for the 32B model, demonstrating that reasoning benefits do not diminish with scale. For the 72B model, which is already strong, V-Reason still yields a **+0.4% gain (73.0 vs. 72.6)**. These results show that V-Reason generalizes to different backbones and larger models and provides benefits even at the frontier of large-scale video reasoning models.

**Comparison with Decoding Methods.** As shown in Table 4, our method is robust and complementary to different decoding strategies with significant improvements over SOTA approaches such as *min-p* [27] and *top-H* [2]. For the Qwen-2.5-VL-7B model, using the best *min-p* decoding with V-Reason (Lite) yields a gain of **+2.0** points on

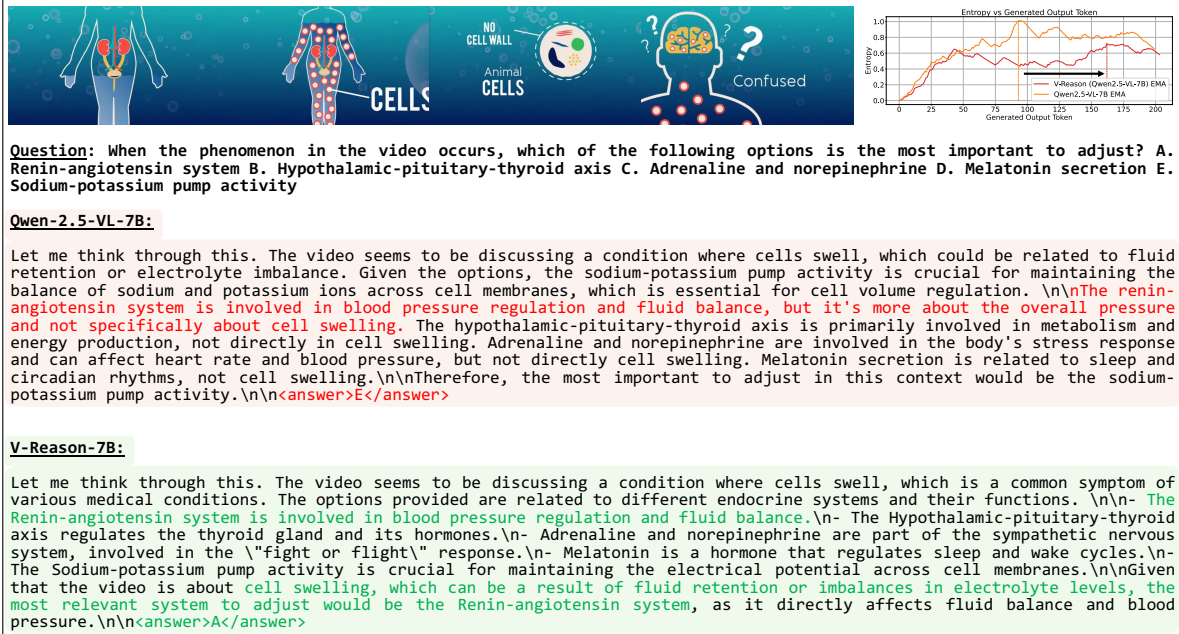


Figure 3. Qualitative result: An example output and comparison with the baseline Qwen-2.5-VL-7B together with its entropy plot shown on the top right. The black arrow in the entropy plot denotes the shift in the EMA peak demonstrating longer exploration for V-Reason compared to the baseline.

Table 4. Comparison with alternative decoding methods.

| Qwen-2.5-VL-7B        | temp | top-p | MMVU               |
|-----------------------|------|-------|--------------------|
| min-p                 | 0.3  | 0.9   | 61.8               |
| min-p+V-Reason (Lite) | 0.3  | 0.9   | <b>63.8 [+2.0]</b> |
| top-H                 | 0.3  | 0.9   | 60.2               |
| top-H+V-Reason (Lite) | 0.3  | 0.9   | <b>61.1 [+0.9]</b> |
| min-p                 | 1.0  | 0.9   | 55.0               |
| min-p+V-Reason (Lite) | 1.0  | 0.9   | <b>61.3 [+6.3]</b> |
| top-H                 | 1.0  | 0.9   | 62.2               |
| top-H+V-Reason (Lite) | 1.0  | 0.9   | <b>62.6 [+0.4]</b> |
| V-Reason-7B (Lite)    | 0.1  | 0.001 | <b>65.0 [+2.8]</b> |

MMVU, while combining with best *top-H* decoding provides a smaller improvement of **+0.4**. On higher temperatures, *min-p* loses significant accuracy but V-Reason (Lite) is able to restore it back (**+6.3%**). Most notably, V-Reason-7B (Lite) achieves the highest score of **65.0**, corresponding to a further **+2.8** gain over the best decoding baseline.

**Qualitative Results.** Figure 3 exemplifies the reasoning differences between V-Reason and the baseline, also showing their entropy profiles. The entropy plots reveal that V-Reason has a delayed EMA peak and a lower overall entropy, encouraging extended exploration that ultimately enables the model to reach the correct solution. As highlighted in red, the baseline initially follows a promising trajectory but subsequently diverges onto an incorrect reasoning path, which leads to the wrong answer. In contrast, V-Reason identifies an alternative path precisely at the point where the baseline falters, and this revised trajectory, shown in green, successfully leads to the correct answer. Please see Supplementary I for other examples.

**Alternative Losses.** The switching loss in equation 2 supports various behaviors beyond that encouraged by V-Reason. Two extreme alternatives are enforcing strictly

increasing entropy (max-entropy,  $\alpha_k = 1, \forall k$ ) and strictly decreasing entropy (min-entropy,  $\alpha_k = -1, \forall k$ ). Table 3 shows the two alternative losses, additional ablations on updating a KV cache controller and the last two layers of the model. It shows that these approaches are clearly inferior to V-Reason. However, it is interesting to note that even the basic losses (encourage macro-exploration or macro-exploitation only) improve on the performance of the baseline model. This confirms that structured entropy control helps with the reasoning ability of LMMs.

## 5. Conclusion

In this paper, we introduced V-Reason, a training-free framework that elicits video reasoning through a value-cache controller at inference. Our method leverages a theoretically-grounded entropy-based objective to reinforce the micro-exploration and micro-exploitation behaviors observed across models. This design effectively mitigates unbounded entropy growth during early generation steps, resulting in delayed entropy peak and lower final entropy, a characteristic of stronger models. We further proposed V-Reason (Lite), a "Lite" variant which improves the memory by pruning low  $l_2$ -norm entries in the value cache. Extensive experiments across multiple benchmarks demonstrate that V-Reason narrows the gap to RL-trained models (e.g., Video-R1) to within **0.6%**, while reducing output token length (**↓58.6%**); this also results in lower (**↓37%**) inference time than Video-R1. Moreover, V-Reason consistently improves performance across model scales ranging from 3B to 72B parameters and remains robust to variations in frame sampling, pixel resolution, decoding techniques, and other hyperparameter configurations.

## References

- [1] Seyedarmin Azizi, Erfan Baghaei Potraghloo, and Massoud Pedram. Activation steering for chain-of-thought compression. *arXiv preprint arXiv:2507.04742*, 2025. 3
- [2] Erfan Baghaei Potraghloo, Seyedarmin Azizi, Souvik Kundu, and Massoud Pedram. Top-h decoding: Adapting the creativity and coherence with bounded entropy in text generation. *NeurIPS*, 2025. submitted / available on arXiv, code: <https://github.com/ErfanBaghaei/Top-H-Decoding>. 3, 7
- [3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 6, 7
- [4] Max Belitsky, Dawid J. Kopiczko, Michael Dorkenwald, M. Jehanzeb Mirza, Cees G. M. Snoek, and Yuki M. Asano. Kv cache steering for inducing reasoning in small language models. *arXiv preprint arXiv:2507.08799*, 2025. 3, 4
- [5] Hila Chefer, Sagie Benaïm, Roni Paiss, and Lior Wolf. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4196–4206, 2023. 3
- [6] Yukang Chen, Wei Huang, Baifeng Shi, Qinghao Hu, Hanrong Ye, Ligeng Zhu, Zhijian Liu, Pavlo Molchanov, Jan Kautz, Xiaojuan Qi, Sifei Liu, Hongxu Yin, Yao Lu, and Song Han. Scaling rl to long videos. *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 3
- [7] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, and Lidong Bing. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024. 7
- [8] Zixu Cheng, Jian Hu, Ziquan Liu, Chenyang Si, Wei Li, and Shaogang Gong. V-star: Benchmarking video-llms on video spatio-temporal reasoning. *arXiv preprint arXiv:2503.11495*, 2025. 1
- [9] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuoyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 3
- [10] Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou Wang, and Xiangyu Yue. Video-rl: Reinforcing video reasoning in mllms. *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 1, 2, 3, 6, 7
- [11] Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024. 7
- [12] Yichao Fu, Xuewei Wang, Yuandong Tian, and Jiawei Zhao. Deep think with confidence. *arXiv preprint arXiv:2508.15260*, 2025. 3
- [13] Anna Golovneva et al. Pathfinder: Learning reasoning paths for complex question answering. In *Proceedings of ACL*, 2023. 3
- [14] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 1, 2
- [15] Kairui Hu, Penghao Wu, Fanyi Pu, Wang Xiao, Yuanhan Zhang, Xiang Yue, Bo Li, and Ziwei Liu. Video-mmmu: Evaluating knowledge acquisition from multi-discipline professional videos. *arXiv preprint arXiv:2501.13826*, 2025. 7
- [16] J. Huang, S. Gu, L. Hou, Y. Wu, X. Wang, H. Yu, and J. Han. Large language models can self-improve. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP 2023)*, pages 1051–1068, Singapore, 2023. Association for Computational Linguistics. 3
- [17] Aaron Hurst, Adam Lerer, Aditya Ramesh, Aidan Clark, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 4, 7
- [18] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 3
- [19] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 7
- [20] Kunchang Li, Yali Wang, Yanan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22195–22206, 2024. 7
- [21] Xinhao Li, Ziang Yan, Desen Meng, Lu Dong, Xiangyu Zeng, Yanan He, Yali Wang, Yu Qiao, Yi Wang, and Limin Wang. Videochat-r1: Enhancing spatio-temporal perception via reinforcement fine-tuning. *arXiv preprint arXiv:2504.06958*, 2025. 1, 3
- [22] Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language models. *arXiv preprint arXiv:2311.17043*, 2023. 7
- [23] Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. *arXiv preprint arXiv:2312.07533*, 2023. 7

- [24] Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. Tempcompass: Do video llms really understand videos? *arXiv preprint arXiv:2403.00476*, 2024. 7
- [25] Zihan Liu, Zhuolin Yang, Yang Chen, Chankyu Lee, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Acereason-nemotron 1.1: Advancing math and code reasoning through sft and rl synergy, 2025. 1
- [26] L. C. Magister, J. Mallinson, J. Adamek, E. Malmi, and A. Severyn. Teaching small language models to reason, 2023. Preprint / not yet published in a major conference (as of available info). 3
- [27] Minh Nguyen, Andrew Baker, Clement Neo, Allen Roush, Andreas Kirsch, and Ravid Shwartz-Ziv. Turning up the heat: Min- $p$  sampling for creative and coherent llm outputs. In *International Conference on Learning Representations (ICLR) 2025*, 2024. arXiv:2407.01082. 3, 7
- [28] OpenAI, :, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, and et al. Openai o1 system card, 2024. 1, 2
- [29] Litu Rout, Yujia Chen, Nataniel Ruiz, Abhishek Kumar, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Rb-modulation: Training-free stylization using reference-based modulation. In *International Conference on Learning Representations (ICLR)*, 2025. 3
- [30] Tim Salimans and Durk P Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. *Advances in neural information processing systems*, 29, 2016. 4
- [31] Yuda Song, Julia Kempe, and Remi Munos. Outcome-based exploration for llm reasoning. *arXiv preprint arXiv:2509.06941*, 2025. 1
- [32] Nathan Srebro and Adi Shraibman. Rank, trace-norm and max-norm. In *International conference on computational learning theory*, pages 545–560. Springer, 2005. 4
- [33] Andong Wang, Bo Wu, Sunli Chen, Zhenfang Chen, Haotian Guan, Wei-Ning Lee, Li Erran Li, and Chuang Gan. Sok-bench: A situated video reasoning benchmark with aligned open-world knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1
- [34] Chenlong Wang, Yuanning Feng, Dongping Chen, Zhaoyang Chu, Ranjay Krishna, and Tianyi Zhou. Wait, we don’t need to “wait”! removing thinking tokens improves reasoning efficiency. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, 2025. arXiv preprint arXiv:2506.08343. 2
- [35] Xuezhi Wang and Denny Zhou. Chain-of-thought reasoning without prompting. *arXiv preprint arXiv:2402.10200*, 2024. 3
- [36] 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. *arXiv preprint arXiv:2201.11903*, 2022. 1, 2
- [37] Yuyang Wu, Yifei Wang, Tianqi Du, Stefanie Jegelka, and Yisen Wang. When more is less: Understanding chain-of-thought length in LLMs. *arXiv preprint arXiv:2502.07266*, 2025. 2
- [38] Silei Xu, Wenhao Xie, Lingxiao Zhao, and Pengcheng He. Chain of draft: Thinking faster by writing less. *arXiv preprint arXiv:2502.18600*, 2025. 2
- [39] Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *International Conference on Learning Representations (ICLR) 2024*, 2024. 3
- [40] Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *CVPR*, 2025. 7
- [41] Matthew Yasunaga, Xinyun Chen, Yichi Li, Pan Pasupat, Jure Leskovec, Percy Liang, Ed H. Chi, and Denny Zhou. Large language models as analogical reasoners. *arXiv preprint, arXiv:2310.01714*, 2023. 3
- [42] En Yu, Kangheng Lin, Liang Zhao, Yana Wei, Zining Zhu, Haoran Wei, Jianjian Sun, Zheng Ge, Xiangyu Zhang, Jingyu Wang, and Wenbing Tao. Video-utr: Unhackable temporal rewarding for scalable video mllms. *arXiv preprint arXiv:2502.12081*, 2025. 7
- [43] Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 1
- [44] Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. Long context transfer from language to vision. *arXiv preprint arXiv:2406.16852*, 2024. 7
- [45] Xingjian Zhang, Siwei Wen, Wenjun Wu, and Lei Huang. Tinyllava-video-r1: Towards smaller llms for video reasoning. *arXiv preprint arXiv:2504.09641*, 2025. 3
- [46] Yuanhan Zhang, Yunice Chew, Yuhao Dong, Aria Leo, Bo Hu, and Ziwei Liu. Towards video thinking test: A holistic benchmark for advanced video reasoning and understanding. *arXiv preprint arXiv:2507.15028*, 2025. 1
- [47] Yunxiang Zhang, Muhammad Khalifa, Lechen Zhang, Xin Liu, Ayoun Lee, Xinliang Frederick Zhang, Farima Fatahi Bayat, and Lu Wang. Logit arithmetic elicits long reasoning capabilities without training. *arXiv preprint arXiv:2507.12759*, 2025. 3
- [48] Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. Learning to reason without external rewards. *arXiv preprint arXiv:2505.19590*, 2025. 1
- [49] Yilun Zhao, Lujing Xie, Haowei Zhang, Guo Gan, Yitao Long, Zhiyuan Hu, Tongyan Hu, et al. Mmvu: Measuring expert-level multi-discipline video understanding. *arXiv preprint arXiv:2501.12380*, 2025. 2, 7
- [50] D. Zhou, N. Schärli, L. Hou, J. Wei, N. Scales, X. Wang, D. Schuurmans, C. Cui, O. Bousquet, Q. V. Le, and E. H. Chi. Least-to-most prompting enables complex reasoning in large language models. In *International Conference on Learning Representations (ICLR) 2023*, 2023. 3

- [51] Yongchao Zhou, Andrei Ioan Muresanu, Ziwon Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers. In *International Conference on Learning Representations (ICLR) 2023*, 2023. 3