REOBSERVE: ALLOCATING A FIXED BUDGET BETWEEN VISUAL READINGS AND REASONING CHAINS

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language systems often answer a question about an image by reading the image once, sampling several reasoning chains over that reading and taking a majority vote. Chains built on the same reading share its errors. With r readings and k chains on each, the fraction of correct chains has variance $\sigma_V^2/r + \sigma_R^2/(rk)$, where σ_V^2 is the variance between readings and σ_R^2 the variance within a reading. The first term does not shrink however many chains are added, and on seven of nine benchmarks σ_V^2 is 0.48 to 1.68 times σ_R^2. We show that, on questions that most chains answer correctly, this variance bounds the accuracy of the vote from below, and that with one reading no number of chains can raise the bound above a level set by σ_V^2. We also show that the variance of any schedule equals the least variance a continuous split could reach times two factors, one for the unspent budget and one for the chains per reading, and the second changes so slowly near its optimum that rough estimates of the two terms suffice. Reobserve estimates them on a small labelled calibration set and picks the feasible integer schedule with the least predicted variance. Under the same cost cap, with the calibration cost charged to every rule, it improves on the single-reading default by 1.84 points on average over nine benchmarks and two readers and stays within 0.15 points of the best feasible schedule at our main budget. On two benchmarks with σ_V^2 close to zero, it keeps the default schedule, and no rule gains more than 0.2 points.

1 INTRODUCTION

Vision-language models (VLMs) answer a question about an image in two steps: they perceive the image and then reason about what they perceived (Liu et al., 2023; Bai et al., 2025b; Zhang et al., 2024b; Xu et al., 2025). When more computation is allowed at test time, the common way to spend it is to sample several reasoning chains and return the answer most of them reach (Wang et al., 2023; Brown et al., 2024; Snell et al., 2025). Many systems make the two steps explicit: a reader model turns the image into a text description, and a reasoner samples chains over the question and that description (Zeng et al., 2023; Qiao et al., 2024). We call one such description a *reading* and one sampled trace of the reasoner a *chain*. In the usual schedule all chains share one reading, so the whole budget goes to thinking again and none of it to reading again.

Whether this is a good use of the budget depends on what a vote can correct. A vote helps because errors that are independent tend to cancel. Chains built on one reading are not independent: if the reader misreads a tick label on a chart, every chain reasons from the same wrong number, and more chains only make the vote more confident. Fig. 1 shows the effect at a fixed budget. On MathVista, sixteen chains on one reading answer 63.72% of the questions correctly, while eight readings with one chain each answer 65.68% within the same budget. At one reading, going from one chain to sixteen adds only 1.7 points, which suggests that part of the error comes from the reading itself.

To measure this part, let p_Z be the probability that a chain on reading Z answers correctly. With r readings and k chains on each, the fraction of correct chains has variance $\sigma_V^2/r + \sigma_R^2/(rk)$, where σ_V^2 is the variance of p_Z across readings and σ_R^2 the mean variance of a chain given its reading (Section 3). We call it the vote variance. This is the classical decomposition of two-stage sampling (Cochran, 1977; Kish, 1965) with the reading as the cluster, and its first term does not depend on k. How much the term

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

matters depends on the size of σ_V^2, which we estimate on nine benchmarks under three ways of sampling readings: rewritten prompts, decoding temperatures and image crops. With rewritten prompts, σ_V^2 lies between 0.48 and 1.68 times σ_R^2 on seven benchmarks, and on SEED and RealWorldQA it is below 10^{-5}. On the three benchmarks with full grids of schedules, accuracy falls almost linearly as the logarithm of the vote variance rises ($R^2 > 0.97$). Two questions follow: why the vote variance matters for accuracy, and which split between readings and chains makes it smallest at a given cost.

Two results answer these questions (Section 4). First, when most chains are correct, the vote variance gives a lower bound on the accuracy of the vote. At one reading, no number of chains can raise this bound above $\delta^2/(\delta^2 + \sigma_V^2)$, where δ is the margin by which the mean chain accuracy exceeds one half, and only more readings raise it further. Second, the variance of any schedule equals the least variance that a continuous split could reach times two factors. One is for the budget the schedule leaves unspent, and the other for the distance between its chains per reading k and the optimum $k^\star = \sqrt{c_V \sigma_R^2/(c_R \sigma_V^2)}$, where c_V and c_R are the costs of a reading and of a chain. On MathVista with the Qwen2.5-VL-7B reader, the single-reading default reaches the vote variance that the best split reaches with 31% of the budget. The second factor changes slowly near $k^\star$: if the ratio σ_R^2/σ_V^2 is off by a factor of two, the variance rises by about 3%. Reobserve therefore estimates σ_V^2 and σ_R^2 on a small calibration set in which each image is read several times and each reading gets several chains, and picks the feasible integer schedule with the least predicted variance, with no training and no change to the reader, the reasoner or the aggregator.

We compare Reobserve with eighteen other rules under the same cost cap on nine benchmarks and two reader sizes, and we subtract its calibration cost from the cap of every rule (Section 5). Reobserve improves on the default by 1.84 points on average, and by 2.37 points on the seven benchmarks where σ_V^2 is not negligible (fourteen benchmark and reader pairs). At our main budget it is never more than 0.15 points below an oracle schedule found by searching all feasible pairs. The gain holds for nine readers from five model families and adds to the gains of four base methods. On SEED and RealWorldQA, where σ_V^2 is close to zero, Reobserve keeps the default schedule, and no rule gains more than 0.2 points. Our contributions are the measurement of the between-reading variance, the accuracy bound with its ceiling at one reading together with the efficiency identity, and a simple rule whose parts we test one at a time (Section 5.4).

2 Related Work

Test-time scaling by sampling. Self-consistency samples several chains and returns the majority answer (Wang et al., 2023; Wei et al., 2022). Later work draws many more samples (Brown et al., 2024), stops once the vote has settled (Aggarwal et al., 2023; Li et al., 2024b), spreads a budget over questions by difficulty (Snell et al., 2025; Wu et al., 2025), weights or ranks chains by confidence, verifiers or reward models (Taubenfeld et al., 2025; Cobbe et al., 2021; Lightman et al., 2024; Wang et al., 2024; Chen et al., 2023), or has the model check its own answer (Weng et al., 2023). Chen et al. (2024a) show that a vote can get worse as calls are added when many queries are hard. These methods decide how many chains to draw and how to combine them, but none models the variance between readings of the input, which Reobserve uses to decide how often to read it; Section 5 compares it with them under the same cap and combines it with three of them.

Perception in multimodal reasoning. Many errors of VLMs (Liu et al., 2023; Bai et al., 2025b; Zhu et al., 2025) and of their reasoning variants (Zhang et al., 2024b; Xu et al., 2025; Huang et al., 2026) come from misreading the image (Tong et al., 2024; Rahmanzadehgervi et al., 2024; Zhang et al., 2024a). Some designs describe the image with one model and reason over the description with a language model (Zeng et al., 2023; Qiao et al., 2024), as in our experiments. Other work improves a single reading, with dynamic high-resolution tiling (Liu et al., 2024a; Guo et al., 2024; Chen et al., 2024b), test-time augmentation over crops (Shanmugam et al., 2021), or region re-reading guided by the reasoning (Shao et al., 2024; Wu & Xie, 2024; Yang et al., 2023; Wang et al., 2025). These methods make one reading better or look again at a part of the image; Reobserve repeats the whole reading and decides how often, so the two can be combined.

Correlated votes and two-stage sampling. Correlation limits what a majority vote can gain: the Condorcet jury theorem assumes independent voters, and positive correlation weakens it (Boland, 1989; Ladha, 1992); ensembles gain from diverse members for the same reason (Krogh & Vedelsby, 1995; Kuncheva & Whitaker, 2003). Two-stage sampling studies the same structure with clusters

and their members, and gives the variance decomposition, the design effect and the cost-optimal cluster size (Cochran, 1977; Kish, 1965; Searle et al., 1992). Our additions concern multimodal voting: a lower bound on vote accuracy and its limit at one reading, the exact efficiency of every schedule relative to the classical optimum with its sensitivity to estimation error, and measurements of the between-reading variance in current VLM systems. Appendix I discusses more related work.

3 Two Levels of Variance in a Multimodal Vote

3.1 Problem Setup

A question q comes with an image and a correct answer. A reader, here a VLM, turns the image into a reading Z, a text that states what the image shows. Readings of one image differ because each is sampled with a random change to the reader's input or decoding: a rewritten extraction prompt in our main results, and a temperature or an image crop in Section 3.2. We call the resulting distribution of readings the *reading distribution*. A reasoner then samples a chain on the question and the reading, and $Y \in \{0, 1\}$ records whether its answer is correct. A *schedule* (r, k) draws r readings, samples k chains on each, and returns the answer with the most votes. A reading costs c_V and a chain costs c_R, so the schedule costs $r(c_V + kc_R)$, and it is feasible under a cap C per question when this cost is at most C. The default schedule $(1, K)$ spends the cap on K chains over one reading.

For a fixed question, let $p_Z = \mathrm{P}(Y = 1 \mid Z)$ be the accuracy of a chain on reading Z and let $\mu_q = \mathbb{E}[p_Z]$ be its mean over readings. We call $\sigma_V^2(q) = \mathrm{Var}(p_Z)$ the *between-reading variance* and $\sigma_R^2(q) = \mathbb{E}[p_Z(1 - p_Z)]$ the *within-reading variance*, the mean variance of one chain given its reading. For a benchmark we report their means over questions, written σ_V^2 and σ_R^2 without the argument q. Finally, $\bar{Y}_{r,k}$ denotes the fraction of the rk chains of a schedule that answer correctly.

Proposition 1 (Two-level variance)**.** *Let the r readings be independent draws from the reading distribution and let the k chains on a reading be independent given that reading. Then*

$$\boxed{\mathrm{Var}(\bar{Y}_{r,k} \mid q) = v_q(r, k) := \frac{\sigma_V^2(q)}{r} + \frac{\sigma_R^2(q)}{rk},} \qquad \textit{and} \qquad \sigma_V^2(q) + \sigma_R^2(q) = \mu_q(1 - \mu_q). \quad (1)$$

The proposition is the law of total variance applied twice (Casella & Berger, 2002; Appendix B.1). Its first term does not contain k, so however many chains are drawn, the variance of a vote over one reading stays above $\sigma_V^2(q)$. The same variance can be written as $\mu_q(1 - \mu_q)\,(1 + (k - 1)\rho_q)/(rk)$, where $\rho_q = \sigma_V^2(q)/(\sigma_V^2(q) + \sigma_R^2(q))$ is the intraclass correlation of the chains on a reading and $1 + (k - 1)\rho_q$ is the design effect of a cluster sample (Kish, 1965). Thus k chains on one reading are worth $k/(1 + (k - 1)\rho_q)$ independent ones. The identity on the right fixes the sum of the two terms by the mean accuracy, so a change of the reading distribution at fixed μ_q moves variance from one term to the other, while every one-chain schedule $(r, 1)$ keeps the variance $\mu_q(1 - \mu_q)/r$. We call the mean of $v_q(r, k)$ over the questions of a benchmark, $v(r, k) = \sigma_V^2/r + \sigma_R^2/(rk)$, the vote variance.

3.2 Measuring the Two Components

Fig. 1 and Table 1 report these measurements with the Qwen2.5-VL-7B reader and the Qwen2.5-7B reasoner, with rewritten prompts unless noted. We call MathVista with this reader and the third of our five budget levels, 37.49 s per question, the *reference setting*. On the MathVista grid, in every column the rows with more readings lie above the single-reading row (Fig. 1a). Reobserve (Section 4) gains 1.96 to 2.88 points over the default on seven benchmarks, with every 95% interval above 1.4 points, and stays within 0.02 points of the default on SEED and RealWorldQA (Fig. 1b). At one reading, sixteen chains remove 36% to 62% of the vote variance on the three benchmarks with full grids (MathVista, MMMU and ChartQA) but add only 0.6 to 1.7 points (Fig. 1c). Extra chains average away the within-reading part of the variance but leave the between-reading part in place.

Table 1 lists the two components. Under the three distributions of the table, $\hat{\sigma}_V^2$ lies between 0.053 and 0.151 on seven benchmarks, and with rewritten prompts it is 0.48 to 1.68 times $\hat{\sigma}_R^2$. Changing the distribution moves $\hat{\sigma}_V^2$ by 5% to 13% relative to prompt rewriting and keeps the order of the seven benchmarks by $k^\star$. On SEED and RealWorldQA, $\hat{\sigma}_V^2$ is below 10^{-5} under all three, so we use them as negative controls and call the others the seven main benchmarks. The estimate reflects how chains

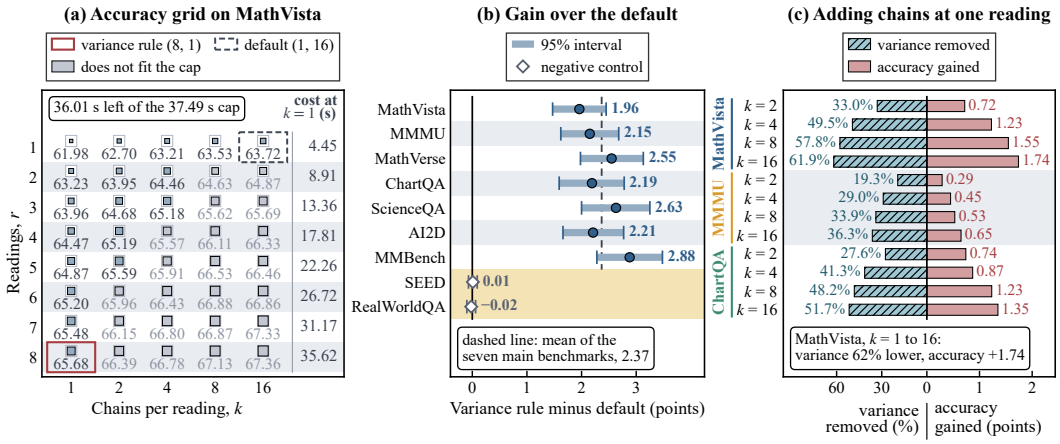

Figure 1: Readings and chains at a fixed budget (7B reader). (a) Accuracy (%) on MathVista of the measured schedules (r, k), with r readings and k chains per reading; the square in each cell grows with the accuracy, values in grey do not fit the 36.01 s left of the cap after the calibration charge, and the right column gives the cost of each row at $k = 1$. (b) Gain of Reobserve over the default on nine benchmarks, with 95% intervals; SEED and RealWorldQA, where $\sigma_V^2 < 10^{-5}$, are negative controls. (c) At one reading, the share of the vote variance that k chains remove and the accuracy they add, relative to one chain.

Table 1: Reference values of the two variance components on nine benchmarks under three reading distributions (7B reader); the temperature and crop columns use temperature 0.7 and disjoint crops. Appendix E compares the calibration estimates of Section 4.2 with these values. $k^\star$ is the continuous optimum of Theorem 2 at the reference costs $c_V = 2.349$ s and $c_R = 2.104$ s, computed with the components of each distribution, so that the column compares benchmarks by the ratio σ_R^2/σ_V^2 alone. SEED and RealWorldQA serve as negative controls.

	Prompt rewriting			Temperature		Crop	
Benchmark	$\hat\sigma_V^2$	$\hat\sigma_R^2$	$k^\star$	$\hat\sigma_V^2$	$k^\star$	$\hat\sigma_V^2$	$k^\star$
MathVista	0.0734	0.1425	1.47	0.0671	1.57	0.0801	1.38
MMMU	0.1143	0.0721	0.84	0.0998	0.98	0.1198	0.79
MathVerse	0.0936	0.1081	1.14	0.0879	1.20	0.0984	1.08
ChartQA	0.0600	0.0737	1.17	0.0526	1.31	0.0641	1.10
ScienceQA	0.0754	0.1564	1.52	0.0708	1.59	0.0792	1.47
AI2D	0.1424	0.0848	0.82	0.1342	0.88	0.1510	0.75
MMBench	0.0744	0.1319	1.41	0.0702	1.47	0.0780	1.36
SEED	$< 10^{-5}$	0.2379	> 300	$< 10^{-5}$	> 300	$< 10^{-5}$	> 300
RealWorldQA	$< 10^{-5}$	0.1794	> 300	$< 10^{-5}$	> 300	$< 10^{-5}$	> 300

group into readings: shuffling which chain belongs to which reading lowers it by a factor of 367 to 1,017 on the seven main benchmarks and leaves it below 3×10^{-4} on all nine (Appendix Fig. 6c).

Accuracy is close to linear in $\ln v(r, k)$ over the forty cells of each measured grid: a least-squares line explains 99.6% of the variance of the cells on MathVista, 97.9% on MMMU and 98.6% on ChartQA, against 83% to 86% for a line in $v(r, k)$ itself (Appendix Fig. 3a). The MathVista slope, $s = 1.78$ points per unit of $\ln v$, predicts the gains under eight further reading distributions, three settings of each source, within 0.06 points (Appendix Fig. 3b). On these benchmarks a smaller vote variance thus goes with a higher accuracy, and the next section asks which schedule has the least variance at a given cost.

4 Reobserve: Allocating Readings and Chains

4.1 Theoretical Analysis

A lower bound on vote accuracy. We first link the variance of Proposition 1 to the accuracy of the vote. Let $\delta_q = \mu_q - \frac{1}{2}$ be the margin by which the mean chain accuracy of q exceeds one half.

Theorem 1 (Lower bound on vote accuracy). *Under the assumptions of Proposition 1, if $\delta_q > 0$, then for any aggregator that returns an answer supported by more than half of the chains,*

$$\mathrm{P}(\textit{correct} \mid q) \;\geq\; \frac{\delta_q^2}{\delta_q^2 + \sigma_V^2(q)/r + \sigma_R^2(q)/(rk)}. \tag{2}$$

As $k \to \infty$ the bound rises to $\delta_q^2/(\delta_q^2 + \sigma_V^2(q)/r)$, which is below one for $r = 1$ whenever $\sigma_V^2(q) > 0$ and tends to one as $r \to \infty$. For $r = 1$ this ceiling is tight: if $\sigma_V^2(q) < \delta_q(1 - \mu_q)$, then for every $\epsilon > 0$ there is a reading distribution with the same μ_q and $\sigma_V^2(q)$ on which the majority vote between two answers over k chains on one reading has accuracy below $\delta_q^2/(\delta_q^2 + \sigma_V^2(q)) + \epsilon$ for all large k. If $\delta_q < 0$ and there are two candidate answers, a majority vote over any schedule is correct with probability at most $v_q(r,k)/(v_q(r,k) + \delta_q^2)$.

The bound follows from Cantelli's inequality applied to $\bar{Y}_{r,k}$ (Cantelli, 1932; Ghosh, 2002), and the ceiling from a two-point reading distribution (Appendix B.2). As a number the bound is loose; what matters is its ceiling at one reading, which only further readings raise. The theorem also marks when less variance does harm: for a question with $\mu_q < \frac{1}{2}$, a smaller variance lowers the upper bound on a correct vote, and in a two-answer example with $\mu_q = 0.43$ it lowers the accuracy of the vote from 33.2% to 16.1% (Appendix Table 27). We therefore expect reallocation to help when, among questions with μ_q near one half, those with $\mu_q > \frac{1}{2}$ carry more weight than those with $\mu_q < \frac{1}{2}$.

Efficiency of a schedule. We now ask which split minimises the vote variance at a given cost. Let $a = \sqrt{c_V \sigma_V^2}$ and $b = \sqrt{c_R \sigma_R^2}$ be the cost-weighted standard deviations of the two terms.

Theorem 2 (Efficiency of a schedule). *Assume $\sigma_V^2, \sigma_R^2 > 0$ and let $k^\star = \sqrt{c_V \sigma_R^2/(c_R \sigma_V^2)}$ and $\kappa = ab/(a+b)^2 \in (0, \frac{1}{4}]$. For every $r, k > 0$ with cost $B = r(c_V + k c_R)$,*

$$v(r,k) = \frac{(a+b)^2}{B}\Big[1 + \kappa\Big(\sqrt{k/k^\star} - \sqrt{k^\star/k}\Big)^2\Big]. \tag{3}$$

Hence the least vote variance over real r, k under a cap C is $v_C^\star = (a+b)^2/C$, reached only at $k = k^\star$ with $B = C$, and the relative efficiency of a schedule is $\eta(r,k) = v_C^\star / v(r,k)$, which is B/C divided by the bracket in (3). If $k^\star/\gamma \leq k \leq \gamma k^\star$ for some $\gamma \geq 1$, then $\eta(r,k) \geq (B/C)\, 4\gamma/(1+\gamma)^2$.

Appendix B.3 derives (3) by completing a square. The optimum $k^\star$ is the classical cost-optimal subsample size of two-stage sampling (Cochran, 1977). The identity adds the exact loss of every other schedule and separates two ways to lose efficiency: leaving part of the cap unspent, which costs the factor C/B, and splitting the budget badly, which costs the bracket. The squared term in the bracket depends on k only through $\ln(k/k^\star)$ and is multiplied by $\kappa \leq \frac{1}{4}$, so the bracket is flat near the optimum: any k within a factor of two of $k^\star$ keeps at least $8/9$ of the efficiency of a fully spent budget. At the reference setting, with the components of Table 1 and the 36.01 s of the cap left after the calibration charge of Section 4.2, we have $k^\star = 1.47$ and $\kappa = 0.245$. The default $(1, 16)$ spends the whole cap but has $\eta = 0.313$, while $(8, 1)$ has $\eta = 0.954$. With the slope s of Section 3.2, this efficiency ratio corresponds to a gain of $s \ln(0.954/0.313) = 1.99$ points, close to the measured 1.96 points, although both schedules lie on the grid used to fit s.

Sensitivity to estimation error. Any rule built on Theorem 2 needs σ_V^2 and σ_R^2, which must be estimated on a labelled calibration set. The next result bounds what an error in their ratio costs.

Proposition 2 (Sensitivity to estimation error). *If the estimated ratio $\hat{\sigma}_R^2/\hat{\sigma}_V^2$ differs from σ_R^2/σ_V^2 by a factor $\lambda > 0$, the continuous optimum computed from the estimates is $\tilde{k} = k^\star\sqrt{\lambda}$. At full budget its efficiency is*

$$\eta = \frac{1}{1 + \kappa\,(\lambda^{1/4} - \lambda^{-1/4})^2} \;\geq\; \frac{4\sqrt{\lambda}}{(1+\sqrt{\lambda})^2}, \qquad 1 - \eta = \frac{\kappa}{4}(\ln\lambda)^2 + O\big((\ln\lambda)^4\big). \tag{4}$$

The same holds for an error λ in the cost ratio c_V/c_R when the chosen schedule still spends the budget. If the relative errors of the two estimates are of order $n_c^{-1/2}$ on n_c calibration questions, the expected loss of efficiency is of order $1/n_c$.

A ratio off by a factor of two lowers the efficiency by at most 3%, and a factor of four by at most one ninth. On our benchmarks, 32 calibration questions give ratios within 20% of the reference values of Table 1, a loss below 0.21% (Appendix Fig. 3c), so a small calibration set is enough (Section 5.4).

4.2 Estimation and Selection

Estimation. Reobserve has two steps, listed in Appendix Algorithm 1. On n_c calibration questions with known answers, it first draws r_0 readings of each image and k_0 chains on each reading, and records the correctness Y_{ij} of chain j on reading i. With $\bar{Y}_{i\cdot}$ the mean on reading i and $\bar{Y}_{\cdot\cdot}$ the mean over the question, it estimates the two components of each question by a one-way analysis of variance (Searle et al., 1992),

$$\hat{\sigma}_R^2(q) = \frac{1}{r_0(k_0-1)} \sum_{i,j} \left(Y_{ij} - \bar{Y}_{i\cdot}\right)^2, \qquad \hat{\sigma}_V^2(q) = \frac{1}{r_0 - 1} \sum_{i} \left(\bar{Y}_{i\cdot} - \bar{Y}_{\cdot\cdot}\right)^2 - \frac{\hat{\sigma}_R^2(q)}{k_0}, \tag{5}$$

and averages both over the calibration questions. The averages $\hat{\sigma}_R^2$ and $\hat{\sigma}_V^2$ are unbiased estimates (Appendix B.5); a negative $\hat{\sigma}_V^2$, which occurs only on the negative controls, is set to zero.

Selection. For a cap C, the rule then selects

$$\boxed{(\hat{r}, \hat{k}) = \arg\min_{(r,k)\in\mathcal{F}} \frac{\hat{\sigma}_V^2}{r} + \frac{\hat{\sigma}_R^2}{rk},} \tag{6}$$

where $\mathcal{F}$ holds the integer pairs with $r \le 8$ and $k \le 16$ (the size of our evaluation pool) that cost at most $C - c_{\text{cal}}$. Here c_{cal} is the cost of the calibration divided by the number of deployment questions, and we call (6) the variance rule. For a new question it draws $\hat{r}$ readings with $\hat{k}$ chains on each and returns the majority answer. Enumeration also accounts for the factor B/C of Theorem 2, which choosing k by the bracket alone ignores: at the reference setting $(5, 2)$ has the smaller bracket, but $(8, 1)$ spends more of the cap and has the higher efficiency, 0.954 against 0.890 (Appendix Table 8).

Defaults. We use $r_0 = 8$ and $k_0 = 16$ on $n_c = 512$ questions. This costs 147,502 s, which is spread over 10^5 deployment questions as 1.475 s per question and subtracted from the cap of every rule. The *accuracy pick*, a variant without the variance model, keeps the calibration pool and chooses the feasible pair with the highest resampled accuracy, so it tests whether the variance model is needed.

5 Experiments

5.1 Setup

Benchmarks and models. We use nine benchmarks, MathVista, MMMU, MathVerse, ChartQA, ScienceQA, AI2D, MMBench, SEED-Bench and RealWorldQA (Lu et al., 2024; Yue et al., 2024; Zhang et al., 2024a; Masry et al., 2022; Lu et al., 2022; Kembhavi et al., 2016; Liu et al., 2024b; Li et al., 2024a; xAI, 2024). The readers are Qwen2.5-VL-7B and -72B (Bai et al., 2025b), with seven more from the Qwen3-VL, InternVL3, Gemma 3 and Llama 4 families in the reader study of Section 5.2 (Bai et al., 2025a; Zhu et al., 2025; Gemma Team et al., 2025; Meta AI, 2025). The reasoner is Qwen2.5-7B (Qwen Team et al., 2024), and a majority vote combines the chains.

Budget. The costs c_V and c_R are mean wall-clock times, and a rule is charged $r(c_V + kc_R)$ for its schedule plus the calibration charge. At the reference setting the cap of 37.49 s pays for the default $(1, 16)$ and this charge. Each benchmark has its own cap at this level (Appendix Table 17 lists six), and the other four levels are 0.55, 0.78, 1.35 and 1.8 times it, or 20.62 s to 67.48 s on MathVista.

Baselines. Eighteen rules share the cap and one evaluation pool of readings and chains per question, and every rule is scored by the accuracy of the schedule it selects. Seven are budget splits: the default $(1, K)$ with $K = 16$, a fixed re-reading schedule $(7, 1)$, a random feasible (r, k), an equal split, a σ-proportional split with $r : k = \sigma_V : \sigma_R$, one global k for all benchmarks, and the accuracy pick of Section 4.2. The other eleven come from prior work and enter through the schedules that their own criteria select on the pool. Three adapt the number of chains: Adaptive-Consistency (Aggarwal et al., 2023), early-stopping self-consistency (Li et al., 2024b) and compute-optimal scaling (Snell et al., 2025). Four weight or rank the chains: a confidence-weighted vote (Taubenfeld et al., 2025), best-of-n with an outcome reward model (Cobbe et al., 2021), a process-reward vote (Lightman et al., 2024; Wang et al., 2024) and universal self-consistency (Chen et al., 2023). Four re-read or crop the image: dynamic high-resolution tiling (Liu et al., 2024a), a multi-crop ensemble (Shanmugam et al., 2021), a self-verification re-read (Weng et al., 2023) and a Visual CoT re-read (Shao et al., 2024). The oracle schedule (grey) is the feasible pair with the highest accuracy over all feasible pairs. Intervals are 95% normal-approximation intervals of paired differences over questions; Appendix D gives the details.

Table 2: Accuracy (%) of nineteen rules under the same cost cap on nine benchmarks and two readers (Qwen2.5-VL-7B and -72B, reference budget level). The calibration cost of Reobserve is subtracted from the cap of every rule. SEED and RealWorldQA (RWQA) are negative controls with $\sigma_V^2 < 10^{-5}$, where no rule gains more than 0.2 points over the default, and are left unmarked (grey headers). Best in bold and second best underlined; the oracle schedule (grey) is given for the seven main benchmarks. SC: self-consistency.

	MathVista		MMMU		MathVerse		ChartQA		ScienceQA		AI2D		MMBench		SEED		RWQA		
Rule	7B	72B	7B	72B	7B	72B	7B	72B	7B	72B	7B	72B	7B	72B	7B	72B	7B	72B	Avg↑
Default $(1, K)$	63.7	72.0	59.9	66.7	49.8	55.7	85.5	92.7	81.8	88.7	78.9	86.3	78.2	87.1	74.2	80.5	59.5	64.8	73.7
Fixed re-reading $(7, 1)$	65.5	73.6	61.7	68.5	52.2	58.0	87.3	94.5	84.1	91.1	80.8	88.2	80.8	89.7	74.3	80.1	59.4	64.4	75.2
Random (r, k)	63.7	72.8	60.7	66.7	50.6	56.9	86.6	92.5	84.3	88.4	78.9	87.4	80.7	86.2	74.0	80.2	59.1	64.3	74.1
Equal split	65.0	73.3	61.2	68.1	51.9	57.7	87.3	94.6	83.6	90.5	80.8	88.1	80.0	88.9	74.4	80.5	58.9	64.5	75.0
σ-proportional split	62.0	70.3	59.2	66.0	48.2	54.0	84.1	91.4	79.2	86.1	78.1	85.5	75.9	84.8	73.9	80.4	59.3	64.4	72.4
One global k	65.2	73.5	61.3	68.1	52.0	57.9	86.6	93.8	83.9	90.8	80.1	87.4	80.3	89.2	73.7	80.0	59.6	64.7	74.9
Accuracy pick	65.6	**73.9**	61.8	68.7	52.2	58.1	87.5	**94.8**	84.3	91.2	80.8	88.1	80.9	89.8	74.2	80.5	59.3	64.6	75.3
Adaptive-Consistency	63.5	73.8	60.7	66.6	49.7	57.9	86.2	92.4	81.8	90.0	78.6	86.8	80.0	84.8	74.0	80.0	59.4	64.2	73.9
Early-stopping SC	63.0	70.3	59.6	68.0	51.1	57.9	87.1	92.8	82.8	89.0	80.1	87.6	77.5	89.2	74.0	80.6	59.4	64.3	74.1
Compute-optimal scaling	63.7	72.8	59.9	66.5	50.8	57.4	86.2	94.6	83.2	88.4	78.5	88.1	80.7	84.8	73.8	80.2	59.4	64.3	74.1
Confidence-weighted vote	63.2	71.8	59.8	67.5	51.5	56.4	86.7	93.4	81.0	89.5	78.7	86.1	80.7	86.7	73.7	80.4	59.3	64.4	73.9
Outcome-reward best-of-n	64.3	71.8	59.8	68.4	50.8	56.9	87.1	94.3	81.8	88.7	80.8	86.2	79.5	88.2	74.2	80.5	59.1	64.7	74.3
Process-reward vote	64.7	71.5	61.2	66.0	49.8	55.7	84.7	92.0	83.1	89.5	78.7	86.3	77.6	89.6	73.8	80.2	59.4	64.3	73.8
Universal SC	65.2	71.0	61.2	66.4	52.0	57.7	87.3	92.8	82.1	90.5	78.8	88.0	78.6	86.5	74.2	79.9	59.4	64.2	74.2
Dynamic high-res. tiling	64.5	73.5	61.5	66.7	51.5	56.9	84.7	92.6	82.1	90.8	79.4	86.2	80.7	85.8	74.3	80.0	59.5	64.7	74.2
Multi-crop ensemble	64.7	71.7	61.3	67.0	51.1	55.2	86.2	93.7	82.6	90.0	79.8	86.3	80.0	87.5	74.1	80.1	59.1	64.5	74.2
Self-verification re-read	63.4	72.2	61.5	68.1	48.9	56.4	85.1	92.2	82.1	90.8	80.2	86.2	78.0	87.9	74.0	80.1	58.9	64.9	73.9
Visual CoT re-read	63.4	73.8	60.7	68.4	51.9	55.4	85.5	93.7	83.9	90.8	80.1	87.4	79.5	89.6	74.3	80.3	59.5	64.2	74.6
Reobserve	**65.7**	**73.9**	**62.0**	**68.9**	**52.4**	**58.2**	**87.6**	**94.8**	**84.4**	**91.4**	**81.1**	**88.5**	**81.1**	**90.0**	74.2	80.5	59.5	64.8	**75.5**
Oracle schedule	65.7	74.0	62.0	69.0	52.5	58.3	87.7	94.9	84.5	91.5	81.2	88.6	81.1	90.1					

5.2 Main Results

Comparison with all rules. On every main benchmark and reader, Reobserve matches or exceeds every deployable rule and is never more than 0.15 points below the oracle schedule (Table 2). It averages 75.5 against 73.7 for the default, a gain of 1.84 points over all eighteen cells and of 2.37 points over the fourteen cells of the seven main benchmarks, where every 95% interval lies above 1.3 points (Appendix Table 14). The closest rules are three splits that also read several times: the accuracy pick, the fixed re-reading schedule and the equal split. The eleven external methods gain 0.1 to 0.9 points over the default, less than any of these splits. The σ-proportional split falls below the default because its ratio rounds to $(1, 1)$, which leaves most of the cap unspent, a small factor B/C in Theorem 2. On the two negative controls Reobserve keeps the default schedule, and an independent run of it stays within 0.06 points of the default; no rule gains more than 0.2 points there.

Budgets and costs. The gain grows with the budget, from 0.81 points at the smallest level to 2.65 at the largest (Appendix Table 16). Over the 45 cells of nine benchmarks and five budgets, Reobserve is ahead of or level with the best baseline in 29, including each of the seven main benchmarks at the reference budget. When readings cost more, the rule reads less often: about seven readings at a quarter of the reference cost and one or two at four times it, with more chains per reading on five of the seven main benchmarks, as $k^\star \propto \sqrt{c_V}$ predicts (Appendix Fig. 9).

Readers and other methods. Across nine readers from five families, the mean gain on the seven main benchmarks is 2.30 to 2.47 points, and over the 63 reader and benchmark cells the gain is almost uncorrelated with the accuracy of the default (correlation 0.06; Appendix Fig. 10). Combined with four base methods, Reobserve adds 2.26 to 3.13 points to each, and the joint gain stays within 0.25 points of the sum of the separate gains (Appendix Table 20). Appendix F also varies the reasoner and the aggregator.

5.3 Analysis of the Gain

The split matters more than the aggregator or the temperature. On six benchmarks, any split that reads several times recovers most of the 2.29 points between the default and the rule, and choosing k for each benchmark closes nearly all the rest: the variance rule ends 0.03 points below the oracle (Table 3, left). Changes that keep the default split move less (Table 3, right): a better aggregator adds 0.35 points and retuning the chain temperature 0.50. Of the rule's gain, 74% to 88% comes from adding readings and 14% to 28% from fewer chains per reading (Appendix Table 15).

Table 3: Splits between the default and the rule, and changes that keep the default split (7B reader, reference budget). Left: accuracy (%) on six benchmarks from the default to the variance rule; the grey row is the oracle schedule. Right: mean accuracy over the six benchmarks when only the aggregator or the chain temperature changes, against three splits, and the gain over the default (points). Best in bold, second best underlined.

Rule	MathV.	MMMU	MathVe.	ChartQA	SciQA	AI2D	Avg ↑
Default $(1, K)$	63.72	59.86	49.84	85.45	81.81	78.92	69.93
σ-proportional	61.98	59.21	48.16	84.10	79.22	78.09	68.46
One global k	65.18	61.29	52.03	86.58	83.89	80.06	71.51
Equal split	65.00	61.22	51.86	87.31	83.61	<u>80.76</u>	71.63
Accuracy pick	<u>65.59</u>	<u>61.84</u>	<u>52.23</u>	<u>87.53</u>	<u>84.26</u>	<u>80.76</u>	<u>72.04</u>
Variance rule	**65.68**	**62.01**	**52.39**	**87.64**	**84.44**	**81.13**	**72.22**
Oracle schedule	65.72	62.01	52.46	87.67	84.46	81.19	72.25

Change	Avg ↑	Gain ↑
Aggregator only	70.28	+0.35
Temperature only	70.43	+0.50
One global k	71.51	+1.58
Accuracy pick	<u>72.04</u>	<u>+2.11</u>
Variance rule	**72.22**	+2.29

Table 4: Ablations (7B reader, reference budget). Change in accuracy (points) against the full rule, as the mean and the worst case over the seven main benchmarks. n/a: the calibration charge exceeds the cap. When a wrong input selects a schedule above the cap, the rule keeps its k and uses the largest r that fits. Per-benchmark values are in Appendix G. Against the default, the full rule gains 2.37 points on average on these benchmarks.

Change	Setting	Mean ↑	Worst ↑
Fixed r	$r = 1$	−2.37	−2.88
	$r = 2$	−1.19	−1.55
	$r = 4$	−0.28	−0.58
Fixed k	$k = 1$	−0.07	−0.27
	$k = 2$	−0.19	−0.59
	$k = 4$	−0.77	−1.13
	$k = 8$	−2.56	−3.12
Inputs	grouping ignored	−2.37	−2.88
	c_R set to zero	−2.37	−2.88
	c_V set to zero	−0.13	−0.23
Reversed inputs	components swapped	−0.12	−0.43
	inverse of $k^\star$	−0.11	−0.59

Change	Setting	Mean ↑	Worst ↑
Selection	by accuracy	−0.18	−0.37
Calibration questions	$n_c = 64$	0.00	0.00
	$n_c = 128$	0.00	0.00
	$n_c = 512$	0.00	0.00
Design ($n_c = 128$)	$(r_0, k_0) = (4, 4)$	0.00	0.00
	$(r_0, k_0) = (2, 4)$	0.00	0.00
Deployment questions	10^3	n/a	n/a
	10^4	−1.04	−1.32
	10^5	0.00	0.00
Prompts	changed after calib.	−0.72	−1.80

More between-reading variance calls for fewer chains per reading. Adding noise to the readings at fixed mean accuracy moves variance from σ_R^2 to σ_V^2 (Proposition 1). On MathVista, as $\hat{\sigma}_V^2$ grows eightfold, $k^\star$ falls from 3.85 to 0.88 and the rule moves from three chains per reading to one (Appendix Table 21). The gain grows with the noise, from zero at 5% of the reference noise to 2.37 points on average at the full noise (Appendix Table 26), and the log-linear relation of Section 3.2 predicts it within 0.05 points on MathVista and MMMU and 0.17 on ChartQA (Appendix Fig. 3b).

5.4 Ablation Studies

Most of the gain comes from reading more than once. Fixing a single reading and choosing the best k loses 2.37 points (Table 4), the whole gain, and two readings still lose 1.19. Fixing k at one or two chains per reading loses only 0.07 and 0.19 points, but $k = 8$ forces a single reading and loses 2.56.

The rule needs the grouping of chains by reading and both costs. If the estimator ignores which reading each chain came from, $\hat{\sigma}_V^2$ becomes zero and the rule returns the default, losing 2.37 points. A chain cost of zero has the same effect: the rule then asks for sixteen chains per reading, and only one such reading fits the cap. A reading cost of zero leads to $(5, 2)$ or $(4, 2)$ and loses 0.13 points.

The variance model selects better schedules than calibration accuracy does. On the same calibration pool, picking the pair with the best resampled accuracy loses 0.18 points on average and up to 0.37, since it compares noisy accuracies of many pairs where the variance model sums up the whole pool in two numbers. Swapping the two components or inverting $k^\star$ loses 0.12 and 0.11 points, because the integer grid and the flat bracket of (3) absorb most of these errors.

A small calibration is enough when its cost is spread over many questions. Calibrations of 64, 128 and 512 questions select the same schedules, in line with Proposition 2, and the nested designs $(4, 4)$ and $(2, 4)$, at 5,511 s and 2,756 s, change one schedule each but not the mean accuracy. Spread over 10^4 instead of 10^5 deployment questions, the 512-question calibration loses 1.04 points, and over

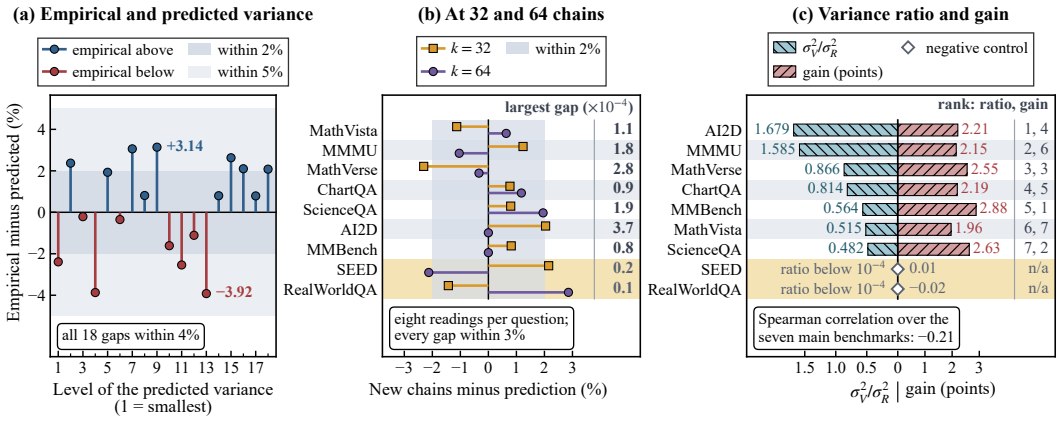

Figure 2: The vote variance after reallocation (7B reader, reference budget). (a) Empirical minus predicted vote variance, in percent of the prediction of (1), at eighteen levels of the predicted variance. (b) Variance of the vote over newly sampled chains minus the prediction, in percent of the prediction, at eight readings with 32 and 64 chains per reading; the side column gives the largest absolute gap, below 4×10^{-4} on every benchmark. (c) Ratio σ_V^2/σ_R^2 and gain of the rule on nine benchmarks, with their ranks among the seven main benchmarks. SEED and RealWorldQA (gold) are the negative controls, where the ratio is below 10^{-4}.

10^3 questions it exceeds the cap, so a smaller deployment calls for a smaller calibration. Changing the prompts after calibration keeps every schedule, so the 0.72 points it loses come from the new prompts.

5.5 Vote Variance after Reallocation

We now measure the vote variance after the rule is applied. Over eighteen levels of the predicted variance, the empirical variance of the vote differs from the prediction of (1) by at most 3.92%, with errors of both signs (Fig. 2a). At eight readings with 32 and 64 chains per reading, beyond the 16 of the calibration design, the variance of newly sampled chains differs from the prediction by less than 3% and by less than 4×10^{-4} on every benchmark (Fig. 2b), so the between-reading term stays as calibrated when many more chains share a reading. The ratio σ_V^2/σ_R^2 alone does not order the gains across benchmarks (Spearman -0.21, Fig. 2c); the gain also depends on which integer schedules each cap admits and on how many points a unit of $\ln v$ is worth, 1.45 to 2.07 on the three benchmarks with full grids.

6 Conclusion

Chains sampled over one reading share that reading's errors, so the variance of the vote has a between-reading term that more chains cannot reduce, and on seven of nine benchmarks this term is of the same order as the within-reading term. We showed that the vote variance bounds the accuracy of the vote from below when most chains are correct, with a ceiling at one reading that only more readings can raise, and that the variance of any schedule is the least variance a continuous split could reach times a factor for the unspent budget and a factor for the split. Reobserve uses these results to choose the split from a small calibration set and improves on the single-reading default by 1.84 points under the same cost cap. A system that already samples several chains per question can use it by spreading those chains over several readings, with no change to the reader, the reasoner or the aggregator.

Limitations. Reobserve chooses one schedule per benchmark from a labelled calibration set, whose cost pays off only when spread over many deployment questions. We test it with a separate reader and reasoner, not with one model that reads and reasons in a single pass, and we expect it to help most when uncertain questions have a mean chain accuracy above one half (Theorem 1). Choosing the schedule for each question, rather than one schedule for each benchmark, is a natural next step.

Reproducibility Statement

Reobserve is defined by (5), (6) and Algorithm 1. Appendix B proves every result of Section 4.1, Appendix D describes the benchmarks, models, reading distributions, cost measurements and baselines, and Appendices C and E to H list or print the numbers behind the tables and figures. The accuracy grids, calibration pools, cost measurements and code will be released with the final version.

AI Use Statement

Generative AI tools were used only to edit the wording of text written by the authors, to search for related work and check bibliographic details, and to write code for plots, tables and typesetting. They were not used for any task that the ICLR 2027 AI policy requires authors to disclose: developing theory, formulating or proving mathematical claims, proposing or refining hypotheses, designing, implementing or running the method or experiments, generating, cleaning or analyzing data, translating text, or interpreting results. The models in our experiments are objects of study, not writing aids. The authors reviewed all AI-assisted work and take full responsibility for the content of this paper.

References

Pranjal Aggarwal, Aman Madaan, Yiming Yang, and Mausam. Let's sample step by step: Adaptive-Consistency for efficient reasoning and coding with LLMs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 12375–12396, Singapore, 2023.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Philip J. Boland. Majority systems and the Condorcet jury theorem. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 38(3):181–189, 1989. doi: 10.2307/2348873.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024.

F. P. Cantelli. Sui confini della probabilità. In *Atti del Congresso Internazionale dei Matematici, Bologna 3–10 Settembre 1928*, volume 6, pp. 47–59, 1932.

George Casella and Roger L. Berger. *Statistical Inference*. Thomson Learning, Pacific Grove, CA, 2nd edition, 2002. ISBN 978-0-534-24312-8.

Lingjiao Chen, Jared Davis, Boris Hanin, Peter Bailis, Ion Stoica, Matei Zaharia, and James Zou. Are more LLM calls all you need? Towards the scaling properties of compound AI systems. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 45767–45790, 2024a.

Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, et al. Universal self-consistency for large language model generation. *arXiv preprint arXiv:2311.17311*, 2023.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, et al. How far are we to GPT-4V? Closing the gap to commercial multimodal models with open-source suites. *Science China Information Sciences*, 67(12):220101, 2024b. doi: 10.1007/s11432-024-4231-5.

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

William G. Cochran. *Sampling Techniques*. John Wiley & Sons, New York, 3rd edition, 1977. ISBN 978-0-471-16240-7.

Yifan Du, Zikang Liu, Yifan Li, Wayne Xin Zhao, Yuqi Huo, Bingning Wang, Weipeng Chen, Zheng Liu, et al. Virgo: A preliminary exploration on reproducing o1-like MLLM. *arXiv preprint arXiv:2501.01904*, 2025.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

B. K. Ghosh. Probability inequalities related to Markov's theorem. *The American Statistician*, 56(3): 186–190, 2002. doi: 10.1198/000313002119.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025. doi: 10.1038/s41586-025-09422-z.

Zonghao Guo, Ruyi Xu, Yuan Yao, Junbo Cui, Zanlin Ni, Chunjiang Ge, Tat-Seng Chua, Zhiyuan Liu, et al. LLaVA-UHD: An LMM perceiving any aspect ratio and high-resolution images. In *European Conference on Computer Vision (ECCV)*, pp. 390–406, 2024. doi: 10.1007/978-3-031-73010-8_23.

Tanmay Gupta and Aniruddha Kembhavi. Visual programming: Compositional visual reasoning without training. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14953–14962, 2023.

Wenxuan Huang, Bohan Jia, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-R1: Incentivizing reasoning capability in multimodal large language models. In *International Conference on Learning Representations (ICLR)*, 2026.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European Conference on Computer Vision (ECCV)*, pp. 235–251, 2016. doi: 10.1007/978-3-319-46493-0_15.

Leslie Kish. *Survey Sampling*. John Wiley & Sons, New York, 1965.

Anders Krogh and Jesper Vedelsby. Neural network ensembles, cross validation, and active learning. In *Advances in Neural Information Processing Systems (NIPS)*, volume 7, pp. 231–238. MIT Press, 1995.

Ludmila I. Kuncheva and Christopher J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2):181–207, 2003. doi: 10.1023/A:1022859003006.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, et al. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles (SOSP)*, pp. 611–626, 2023. doi: 10.1145/3600006.3613165.

Krishna K. Ladha. The Condorcet jury theorem, free speech, and correlated votes. *American Journal of Political Science*, 36(3):617–634, 1992. doi: 10.2307/2111584.

Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. SEED-Bench: Benchmarking multimodal large language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13299–13308, 2024a.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 292–305, Singapore, 2023.

Yiwei Li, Peiwen Yuan, Shaoxiong Feng, Boyuan Pan, Xinglin Wang, Bin Sun, Heda Wang, and Kan Li. Escape sky-high cost: Early-stopping self-consistency for multi-step reasoning. In *International Conference on Learning Representations (ICLR)*, 2024b.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, et al. Let's verify step by step. In *International Conference on Learning Representations (ICLR)*, 2024.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 34892–34916, 2023.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge. https://llava-vl.github.io/blog/2024-01-30-llava-next/, January 2024a.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, et al. MMBench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision (ECCV)*, pp. 216–233, 2024b. doi: 10.1007/978-3-031-72658-3_13.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, et al. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 2507–2521, 2022.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, et al. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*, 2024.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 2263–2279, Dublin, Ireland, 2022.

Meta AI. The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation. https://ai.meta.com/blog/llama-4-multimodal-intelligence/, April 2025.

Yuxuan Qiao, Haodong Duan, Xinyu Fang, Junming Yang, Lin Chen, Songyang Zhang, Jiaqi Wang, Dahua Lin, et al. Prism: A framework for decoupling and assessing the capabilities of VLMs. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 111863–111898, 2024.

Qwen Team, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. Vision language models are blind. In *Asian Conference on Computer Vision (ACCV)*, pp. 293–309, 2024. doi: 10.1007/978-981-96-0917-8_17.

Shayle R. Searle, George Casella, and Charles E. McCulloch. *Variance Components*. John Wiley & Sons, New York, 1992. ISBN 978-0-471-62162-1.

Divya Shanmugam, Davis Blalock, Guha Balakrishnan, and John Guttag. Better aggregation in test-time augmentation. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1214–1223, 2021.

Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. Visual CoT: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, volume 37, pp. 8612–8642, 2024.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *International Conference on Learning Representations (ICLR)*, 2025.

Amir Taubenfeld, Tom Sheffer, Eran Ofek, Amir Feder, Ariel Goldstein, Zorik Gekhman, and Gal Yona. Confidence improves self-consistency in LLMs. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20090–20111, Vienna, Austria, 2025.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? Exploring the visual shortcomings of multimodal LLMs. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9568–9578, 2024.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, et al. Math-Shepherd: Verify and reinforce LLMs step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (ACL)*, pp. 9426–9439, Bangkok, Thailand, 2024.

Weiyun Wang, Zhangwei Gao, Lianjie Chen, Zhe Chen, Jinguo Zhu, Xiangyu Zhao, Yangzhou Liu, Yue Cao, et al. VisualPRM400K: An effective dataset for training multimodal process reward models. In *International Conference on Learning Representations (ICLR)*, 2026.

Xiyao Wang, Zhengyuan Yang, Linjie Li, Hongjin Lu, Yuancheng Xu, Chung-Ching Lin, Kevin Lin, Furong Huang, et al. Scaling inference-time search with vision value model for improved visual comprehension. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1173–1184, 2025. doi: 10.1109/ICCV51701.2025.00117.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

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

Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Shengping Liu, Bin Sun, Kang Liu, et al. Large language models are better reasoners with self-verification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 2550–2575, Singapore, 2023.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal LLMs. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13084–13094, 2024.

Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. Inference scaling laws: An empirical analysis of compute-optimal inference for LLM problem-solving. In *International Conference on Learning Representations (ICLR)*, 2025.

xAI. Grok-1.5 vision preview. https://x.ai/news/grok-1.5v, April 2024.

Guowei Xu, Peng Jin, Ziang Wu, Hao Li, Yibing Song, Lichao Sun, and Li Yuan. LLaVA-CoT: Let vision language models reason step-by-step. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2087–2098, 2025. doi: 10.1109/ICCV51701.2025.00202.

Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in GPT-4V. *arXiv preprint arXiv:2310.11441*, 2023.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, et al. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9556–9567, 2024. doi: 10.1109/CVPR52733.2024.00913.

Andy Zeng, Maria Attarian, Brian Ichter, Krzysztof Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aveek Purohit, et al. Socratic models: Composing zero-shot multimodal reasoning with language. In *International Conference on Learning Representations (ICLR)*, 2023.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, et al. MathVerse: Does your multi-modal LLM truly see the diagrams in visual math problems? In *European Conference on Computer Vision (ECCV)*, pp. 169–186, 2024a. doi: 10.1007/978-3-031-73242-3_10.

Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. Multimodal chain-of-thought reasoning in language models. *Transactions on Machine Learning Research (TMLR)*, 2024b.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

APPENDIX

A Notation **15**

B Proofs and Derivations **15**

B.1 Proof of Proposition 1 . . . 15

B.2 Proof of Theorem 1 . . . 17

B.3 Proof of Theorem 2 . . . 18

B.4 Proof of Proposition 2 . . . 19

B.5 The Estimator of the Two Components . . . 19

B.6 Averaging over Questions and Heterogeneous Costs . . . 20

B.7 A Model for the Log-Linear Relation . . . 20

C Accuracy Grids and the Log-Linear Fit **21**

D Experimental Details **23**

E Estimator Checks and Calibration **24**

F Full Results **28**

G Additional Ablation Studies **33**

G.1 Fixed Number of Readings or Chains . . . 33

G.2 Calibration Settings and Cost . . . 33

G.3 Errors in the Inputs of the Rule . . . 34

G.4 Selection by Accuracy . . . 34

G.5 Reading Distributions and Reading Noise . . . 34

G.6 Questions with Mean Chain Accuracy below One Half . . . 35

H Vote Variance after Reallocation **35**

I Further Related Work **36**

A NOTATION

Table 5 lists the symbols of the paper and the place where each one is defined. A hat marks an estimate, and a symbol written without the argument q is the mean over the questions of a benchmark. The subscripts V and R stand for the visual reading and the reasoning chain. All accuracies are in percent and all gains are in points of accuracy, unless a table says otherwise.

B PROOFS AND DERIVATIONS

This appendix proves the results of Sections 3.1 and 4.1. A question q is fixed unless an average over questions is written explicitly. The readings $Z_1, \dots, Z_r$ are independent draws from the reading distribution of q, and $Y_{ij} \in \{0, 1\}$ is the correctness of chain j on reading i. Given Z_i, the chains $Y_{i1}, \dots, Y_{ik}$ are independent with $\mathrm{P}(Y_{ij} = 1 \mid Z_i) = p_{Z_i}$. We drop the argument q from $\sigma_V^2(q)$, $\sigma_R^2(q)$, μ_q and δ_q when no confusion with the benchmark means can arise.

B.1 PROOF OF PROPOSITION 1

Variance of the vote. Let $\bar{Y}_{i\cdot} = \frac{1}{k}\sum_{j=1}^{k} Y_{ij}$ be the mean of the chains on reading i, so that $\bar{Y}_{r,k} = \frac{1}{r}\sum_{i=1}^{r} \bar{Y}_{i\cdot}$. Given Z_i the k chains are independent Bernoulli variables with mean p_{Z_i},

Table 5: Symbols and terms.

Symbol or term	Meaning	Defined
q, Z	a question; a reading of its image	§3.1
Y	correctness of one chain (1 if its answer is correct)	§3.1
p_Z, μ_q	accuracy of a chain on reading Z; its mean over readings	§3.1
δ_q	margin $\mu_q - \frac{1}{2}$ of the mean chain accuracy over one half	Thm. 1
$\sigma_V^2(q)$, $\sigma_R^2(q)$	between-reading variance $\mathrm{Var}(p_Z)$; within-reading variance $\mathbb{E}[p_Z(1-p_Z)]$	§3.1
σ_V^2, σ_R^2	means of $\sigma_V^2(q)$ and $\sigma_R^2(q)$ over a benchmark	§3.1
σ_V, σ_R	standard deviations $\sqrt{\sigma_V^2}$ and $\sqrt{\sigma_R^2}$	§5.1
ρ_q	intraclass correlation $\sigma_V^2(q)/(\sigma_V^2(q)+\sigma_R^2(q))$ of the chains on a reading	§3.1
(r,k), K	schedule with r readings and k chains per reading; chains of the default $(1,K)$	§3.1
$\bar{Y}_{r,k}$	fraction of the rk chains of a schedule that answer correctly	§3.1
$v_q(r,k)$, $v(r,k)$	vote variance of question q; its mean over a benchmark	(1)
c_V, c_R	cost of one reading and of one chain, in seconds	§3.1
C, B	cap per question; cost $r(c_V + kc_R)$ of a schedule	§3.1, Thm. 2
a, b	cost-weighted standard deviations $\sqrt{c_V\sigma_V^2}$ and $\sqrt{c_R\sigma_R^2}$	§4.1
$k^\star$, κ	optimal chains per reading $\sqrt{c_V\sigma_R^2/(c_R\sigma_V^2)}$; factor $ab/(a+b)^2$	Thm. 2
$v_C^\star$, η	least vote variance under the cap C; relative efficiency $v_C^\star/v(r,k)$	Thm. 2
γ	factor within which k lies around $k^\star$	Thm. 2
λ, $\tilde{k}$	factor by which an estimated ratio differs from the true one; continuous optimum computed from the estimates	Prop. 2
n_c, (r_0,k_0)	number of calibration questions; readings and chains per reading in the nested design	§4.2
Y_{ij}, $\bar{Y}_{i\cdot}$, $\bar{Y}_{\cdot\cdot}$	correctness of chain j on reading i; mean on reading i; mean of the question	(5)
c_{cal}	calibration cost per deployment question	(6)
$(\hat{r},\hat{k})$, $\mathcal{F}$	schedule chosen by Reobserve; set of feasible integer pairs	(6)
s	slope of accuracy against $\ln v(r,k)$, in points	§3.2
m	cost of a reading relative to its reference cost	App. F
N	number of deployment questions	Alg. 1
$A(r,k)$, α	accuracy of the schedule (r,k); intercept of the log-linear fit	App. C
v_{before}, v_{after}	vote variance before and after a change of schedule	App. C
Reading distribution	the distribution of readings of one image under random changes to the reader's input or decoding: a rewritten extraction prompt (main results), a decoding temperature, or an image crop	§3.1
Variance rule	the selection (6) of Reobserve	§4.2
Accuracy pick	the feasible pair with the highest resampled accuracy on the calibration pool	§4.2
Oracle schedule	the feasible pair with the highest accuracy in a separate search over all feasible pairs	§5.1
Main benchmarks	the seven benchmarks other than SEED and RealWorldQA	§3.2
Negative controls	SEED and RealWorldQA, where $\sigma_V^2 < 10^{-5}$	§3.2
Reference setting	MathVista, Qwen2.5-VL-7B reader, prompt rewriting, third budget level	§3.2

so $\mathbb{E}[\bar{Y}_{i\cdot} \mid Z_i] = p_{Z_i}$ and $\mathrm{Var}(\bar{Y}_{i\cdot} \mid Z_i) = p_{Z_i}(1-p_{Z_i})/k$. The reading means $\bar{Y}_{1\cdot}, \ldots, \bar{Y}_{r\cdot}$ are

independent and identically distributed, and the law of total variance applies to each of them. Hence

$$\begin{aligned}\operatorname{Var}(\bar{Y}_{r,k}) &= \operatorname{Var}\Big(\frac{1}{r}\sum_{i=1}^{r}\bar{Y}_{i\cdot}\Big) = \frac{1}{r^2}\sum_{i=1}^{r}\operatorname{Var}(\bar{Y}_{i\cdot}) = \frac{1}{r}\operatorname{Var}(\bar{Y}_{1\cdot}) \\ &= \frac{1}{r}\Big(\operatorname{Var}\big(\mathbb{E}[\bar{Y}_{1\cdot} \mid Z_1]\big) + \mathbb{E}\big[\operatorname{Var}(\bar{Y}_{1\cdot} \mid Z_1)\big]\Big) \\ &= \frac{1}{r}\Big(\operatorname{Var}(p_{Z_1}) + \frac{1}{k}\,\mathbb{E}\big[p_{Z_1}(1-p_{Z_1})\big]\Big) \\ &= \frac{\sigma_V^2}{r} + \frac{\sigma_R^2}{rk}. \end{aligned} \tag{7}$$

The first line uses the independence of the readings, the second the law of total variance, the third the conditional moments above, and the last the definitions of σ_V^2 and σ_R^2 in Section 3.1.

The identity. A single chain Y is a Bernoulli variable with mean $\mathbb{E}[p_Z] = \mu$, so its variance is $\mu(1-\mu)$. The law of total variance applied to Y itself, with the reading as the conditioning variable, gives

$$\begin{aligned}\mu(1-\mu) = \operatorname{Var}(Y) &= \operatorname{Var}\big(\mathbb{E}[Y \mid Z]\big) + \mathbb{E}\big[\operatorname{Var}(Y \mid Z)\big] \\ &= \operatorname{Var}(p_Z) + \mathbb{E}\big[p_Z(1-p_Z)\big] = \sigma_V^2 + \sigma_R^2, \end{aligned} \tag{8}$$

the identity of Proposition 1. Since $\mu(1-\mu) \le \frac{1}{4}$, the two components sum to at most one quarter.

Intraclass correlation and design effect. For two chains $j \neq j'$ on the same reading, conditional independence gives $\mathbb{E}[Y_{ij}Y_{ij'}] = \mathbb{E}[p_Z^2]$, so their covariance is $\mathbb{E}[p_Z^2] - \mu^2 = \sigma_V^2$ and their correlation is $\rho = \sigma_V^2/(\sigma_V^2 + \sigma_R^2)$ by (8). Factoring $\mu(1-\mu)/(rk)$ out of (7) gives

$$\begin{aligned}\frac{\sigma_V^2}{r} + \frac{\sigma_R^2}{rk} &= \frac{\sigma_V^2 + \sigma_R^2}{rk} \cdot \frac{k\sigma_V^2 + \sigma_R^2}{\sigma_V^2 + \sigma_R^2} = \frac{\mu(1-\mu)}{rk}\Big(1 + (k-1)\frac{\sigma_V^2}{\sigma_V^2 + \sigma_R^2}\Big) \\ &= \frac{\mu(1-\mu)}{rk}\big(1 + (k-1)\rho\big), \end{aligned} \tag{9}$$

the variance of the mean of rk independent chains multiplied by the design effect $1 + (k-1)\rho$ (Kish, 1965). The rk chains are therefore worth $rk/(1 + (k-1)\rho)$ independent ones. For $k = 1$ the design effect is one, and every schedule $(r, 1)$ has variance $\mu(1-\mu)/r$, whatever the division between σ_V^2 and σ_R^2. With the components of Table 1, ρ lies between 0.33 and 0.63 on the seven main benchmarks, so sixteen chains on one reading are worth between 1.5 and 2.7 independent chains. □

B.2 Proof of Theorem 1

Lemma 1 (Cantelli's inequality, Cantelli, 1932; Ghosh, 2002)**.** *Let X have mean μ and variance $\sigma^2 > 0$, and let $t > 0$. Then $\mathrm{P}(X \le \mu - t) \le \sigma^2/(\sigma^2 + t^2)$ and $\mathrm{P}(X \ge \mu + t) \le \sigma^2/(\sigma^2 + t^2)$.*

Proof. For every $u \ge 0$, the event $X \le \mu - t$ is the event $\mu - X + u \ge t + u$, and $t + u > 0$. Markov's inequality applied to the non-negative variable $(\mu - X + u)^2$ gives

$$\mathrm{P}(X \le \mu - t) = \mathrm{P}\big(\mu - X + u \ge t + u\big) \le \frac{\mathbb{E}\big[(\mu - X + u)^2\big]}{(t+u)^2} = \frac{\sigma^2 + u^2}{(t+u)^2}. \tag{10}$$

The right side is smallest at $u = \sigma^2/t$, where

$$\frac{\sigma^2 + \sigma^4/t^2}{(t + \sigma^2/t)^2} = \frac{\sigma^2(t^2 + \sigma^2)/t^2}{(t^2 + \sigma^2)^2/t^2} = \frac{\sigma^2}{\sigma^2 + t^2}. \tag{11}$$

The upper tail follows by applying the same argument to $-X$. □

The bound. If $\bar{Y}_{r,k} > \frac{1}{2}$, more than half of the chains give the correct answer, and the aggregator returns it. With $\delta = \mu - \frac{1}{2} > 0$ and $v = v_q(r,k)$, Lemma 1 applied to $X = \bar{Y}_{r,k}$, whose mean is μ and whose variance is v by Proposition 1, gives

$$\begin{aligned}\mathrm{P}(\text{correct} \mid q) &\ge \mathrm{P}\big(\bar{Y}_{r,k} > \tfrac{1}{2}\big) = 1 - \mathrm{P}\big(\bar{Y}_{r,k} \le \mu - \delta\big) \\ &\ge 1 - \frac{v}{v + \delta^2} = \frac{\delta^2}{\delta^2 + \sigma_V^2/r + \sigma_R^2/(rk)}, \end{aligned} \tag{12}$$

which is (2). As $k \to \infty$ the term $\sigma_R^2/(rk)$ goes to zero and the bound increases to its limit $\delta^2/(\delta^2 + \sigma_V^2/r)$. The limit is below one for $r = 1$ when $\sigma_V^2 > 0$, and it tends to one only as $r \to \infty$.

The ceiling is tight. Take $r = 1$ and two candidate answers. Fix $\epsilon' > 0$ small, set $t = \delta + \epsilon'$ and $\pi = \sigma_V^2/(\sigma_V^2 + t^2)$, and let the reading distribution put mass π on readings with $p_Z = \frac{1}{2} - \epsilon'$ and mass $1 - \pi$ on readings with $p_Z = \mu + \sigma_V^2/t$. This distribution has the required mean and variance:

$$\mathbb{E}[p_Z] = \pi(\mu - t) + (1 - \pi)\Big(\mu + \frac{\sigma_V^2}{t}\Big) = \mu - \frac{\sigma_V^2\, t}{\sigma_V^2 + t^2} + \frac{t^2}{\sigma_V^2 + t^2} \cdot \frac{\sigma_V^2}{t} = \mu,$$

$$\mathrm{Var}(p_Z) = \pi\, t^2 + (1 - \pi)\frac{\sigma_V^4}{t^2} = \frac{\sigma_V^2\, t^2}{\sigma_V^2 + t^2} + \frac{\sigma_V^4}{\sigma_V^2 + t^2} = \sigma_V^2. \tag{13}$$

Both support points lie in $[0, 1]$ when $\sigma_V^2 \le t(1 - \mu)$, which holds for every $\epsilon' > 0$ under the condition $\sigma_V^2 < \delta(1 - \mu)$ of the theorem. On a reading with $p_Z < \frac{1}{2}$ the majority of k chains is wrong with probability tending to one as k grows, and on a reading with $p_Z > \frac{1}{2}$ it is right with probability tending to one, by the law of large numbers. The accuracy of the vote therefore tends to $1 - \pi = t^2/(\sigma_V^2 + t^2)$, which is below $\delta^2/(\delta^2 + \sigma_V^2) + \epsilon/2$ once ϵ' is small enough. Choosing k large then keeps the accuracy below $\delta^2/(\delta^2 + \sigma_V^2) + \epsilon$ for all large k, as the theorem states.

Questions below one half. If $\delta < 0$ and there are two candidate answers, the vote is correct only if $\bar{Y}_{r,k} \ge \frac{1}{2}$, where a tie may be broken either way. The upper tail of Lemma 1 with $t = |\delta|$ gives $\mathrm{P}(\text{correct} \mid q) \le \mathrm{P}(\bar{Y}_{r,k} \ge \mu + |\delta|) \le v/(v + \delta^2)$, which decreases as v decreases. □

B.3 Proof of Theorem 2

The identity. For a schedule with cost $B = r(c_V + kc_R)$ we have $r = B/(c_V + kc_R)$. Substituting this into $v(r, k) = (\sigma_V^2 + \sigma_R^2/k)/r$ and completing a square gives

$$\begin{aligned} v(r,k) &= \frac{(c_V + kc_R)(\sigma_V^2 + \sigma_R^2/k)}{B} = \frac{1}{B}\Big(c_V\sigma_V^2 + c_R\sigma_R^2 + k\, c_R\sigma_V^2 + \frac{c_V\sigma_R^2}{k}\Big) \\ &= \frac{1}{B}\Big(a^2 + b^2 + 2ab + \big(\sqrt{kc_R}\,\sigma_V - \sqrt{c_V/k}\,\sigma_R\big)^2\Big) \\ &= \frac{1}{B}\Big((a+b)^2 + c_R\sigma_V^2\, k^\star\Big(\sqrt{k/k^\star} - \sqrt{k^\star/k}\Big)^2\Big) \\ &= \frac{(a+b)^2}{B}\Big[1 + \kappa\Big(\sqrt{k/k^\star} - \sqrt{k^\star/k}\Big)^2\Big]. \end{aligned} \tag{14}$$

The second line adds and subtracts $2ab = 2\sqrt{c_R\sigma_V^2\, c_V\sigma_R^2}$. The third uses $c_V\sigma_R^2 = (k^\star)^2 c_R\sigma_V^2$, which is the definition of $k^\star$, so that $\sqrt{kc_R}\,\sigma_V - \sqrt{c_V/k}\,\sigma_R = \sqrt{c_R\sigma_V^2 k^\star}\,(\sqrt{k/k^\star} - \sqrt{k^\star/k})$. The last uses $c_R\sigma_V^2 k^\star = \sqrt{c_R\sigma_V^2}\sqrt{c_V\sigma_R^2} = ab$ and $\kappa = ab/(a+b)^2$. By the inequality of arithmetic and geometric means, $(a+b)^2 \ge 4ab$, so $\kappa \le \frac{1}{4}$, with equality when the two cost-weighted terms are equal.

The optimum. Under the cap, $B \le C$ gives $(a+b)^2/B \ge (a+b)^2/C$, and the bracket is at least one with equality exactly when $k = k^\star$. Hence $v(r,k) \ge v_C^\star = (a+b)^2/C$, with equality exactly when $k = k^\star$ and $B = C$, that is, at $r^\star = C/(c_V + k^\star c_R)$. This continuous optimum is the classical cost-optimal subsample size of two-stage sampling (Cochran, 1977); the identity (14) also gives the variance of every other schedule. Dividing $v_C^\star$ by (14) gives the relative efficiency $\eta(r,k)$ of Theorem 2.

Schedules near the optimum. Write $x = k/k^\star$ and $g(x) = (\sqrt{x} - 1/\sqrt{x})^2 = x + 1/x - 2$. The function g depends on x only through $|\ln x|$ and increases with it. If $k^\star/\gamma \le k \le \gamma k^\star$, then $|\ln x| \le \ln\gamma$ and

$$1 + \kappa\, g(x) \le 1 + \frac{g(\gamma)}{4} = 1 + \frac{\gamma + 1/\gamma - 2}{4} = \frac{(\gamma+1)^2}{4\gamma},$$

$$\text{so} \qquad \eta(r,k) \ge \frac{B}{C} \cdot \frac{4\gamma}{(1+\gamma)^2}. \tag{15}$$

For $\gamma = 2$ the second factor is $8/9$. The default $(1, K)$ has $B = C$ and $g(K/k^\star) = (K - k^\star)^2/(Kk^\star)$, so its efficiency is $\eta(1,K) = 1/[1 + \kappa(K - k^\star)^2/(Kk^\star)]$, which falls like $k^\star/(\kappa K)$ as K grows.

Numbers at the reference setting. With $c_V = 2.349\,\text{s}$, $c_R = 2.104\,\text{s}$, $\sigma_V^2 = 0.0734$, $\sigma_R^2 = 0.1425$, and the cap left after the calibration charge, $C - c_{\text{cal}} = 36.01\,\text{s}$, in place of C, we get $a = 0.415$, $b = 0.548$, $\kappa = 0.245$, $k^\star = 1.47$ and $v_C^\star = 0.02574$. The default has $v(1, 16) = 0.0823$ and $\eta = 0.313$; the pair $(8, 1)$ has $v = 0.0270$ and $\eta = 0.954$. Table 8 lists η for all 35 feasible pairs. □

B.4 Proof of Proposition 2

The efficiency after an error. Let the estimated ratio be $\hat{R} = \lambda R$ with $R = \sigma_R^2/\sigma_V^2$. The continuous optimum computed from the estimates is $\tilde{k} = \sqrt{c_V \hat{R}/c_R} = k^\star\sqrt{\lambda}$, so $x = \tilde{k}/k^\star = \sqrt{\lambda}$ and $g(x) = (\lambda^{1/4} - \lambda^{-1/4})^2$. At full budget, $B = C$ and $\eta = 1/(1 + \kappa g)$. With $\kappa \le \frac{1}{4}$,

$$\eta = \frac{1}{1 + \kappa\,(\lambda^{1/4} - \lambda^{-1/4})^2} \ge \frac{1}{1 + \frac{1}{4}(\lambda^{1/4} - \lambda^{-1/4})^2} = \frac{4}{\lambda^{1/2} + 2 + \lambda^{-1/2}}$$
$$= \frac{4}{(\lambda^{1/4} + \lambda^{-1/4})^2} = \frac{4\sqrt{\lambda}}{(1 + \sqrt{\lambda})^2}. \tag{16}$$

An error λ in the cost ratio c_V/c_R enters $\tilde{k}$ through the same square root and gives the same result.

Second order in the error. Let $u = \ln\lambda$. Then $\lambda^{1/4} - \lambda^{-1/4} = 2\sinh(u/4)$ and

$$g = 4\sinh^2\Big(\frac{u}{4}\Big) = \frac{u^2}{4} + \frac{u^4}{192} + O(u^6), \qquad 1 - \eta = \frac{\kappa g}{1 + \kappa g} = \frac{\kappa}{4}u^2 + O(u^4). \tag{17}$$

An error in the ratio thus costs efficiency only at second order, as the bracket of (3) is flat at $k^\star$.

Calibration size. Write $\hat{\sigma}_V^2 = \sigma_V^2(1 + \epsilon_V)$ and $\hat{\sigma}_R^2 = \sigma_R^2(1 + \epsilon_R)$. Both estimates are means over n_c independent calibration questions of unbiased per-question statistics with finite variance (Appendix B.5). For $\sigma_V^2 > 0$ the chance that the mean is clipped at zero falls exponentially in n_c by Hoeffding's inequality, since every per-question estimate is bounded, so up to this term ϵ_V and ϵ_R have mean zero and standard deviations of order $n_c^{-1/2}$. Expanding the logarithm,

$$u = \ln\lambda = \ln(1 + \epsilon_R) - \ln(1 + \epsilon_V) = \epsilon_R - \epsilon_V + O(\epsilon^2),$$
$$\mathbb{E}[u^2] = \mathrm{Var}(\epsilon_R) + \mathrm{Var}(\epsilon_V) - 2\mathrm{Cov}(\epsilon_R, \epsilon_V) + O(n_c^{-3/2}) = O(n_c^{-1}),$$
$$\mathbb{E}[1 - \eta] = \frac{\kappa}{4}\,\mathbb{E}[u^2] + O(n_c^{-2}) = O(n_c^{-1}). \tag{18}$$

On our benchmarks, 32 calibration questions give λ between 0.865 and 1.199, so $1 - \eta < 0.21\%$; at 512 questions λ lies between 0.981 and 1.018 and $1 - \eta < 0.003\%$ (Fig. 3c). □

B.5 The Estimator of the Two Components

Fix a calibration question with r_0 readings and k_0 chains on each. The within-reading estimate $\hat{\sigma}_R^2(q)$ of (5) is an average of unbiased sample variances. Given Z_i, the sum of squares $\sum_j (Y_{ij} - \bar{Y}_{i\cdot})^2$ has mean $(k_0 - 1)p_{Z_i}(1 - p_{Z_i})$, so

$$\mathbb{E}\big[\hat{\sigma}_R^2(q)\big] = \frac{1}{r_0(k_0 - 1)} \sum_{i=1}^{r_0} \mathbb{E}\Big[\mathbb{E}\Big[\sum_{j=1}^{k_0} \big(Y_{ij} - \bar{Y}_{i\cdot}\big)^2 \,\Big|\, Z_i\Big]\Big]$$
$$= \frac{1}{r_0(k_0 - 1)} \sum_{i=1}^{r_0} (k_0 - 1)\,\mathbb{E}\big[p_{Z_i}(1 - p_{Z_i})\big] = \sigma_R^2(q). \tag{19}$$

The reading means $\bar{Y}_{1\cdot}, \ldots, \bar{Y}_{r_0\cdot}$ are independent with common variance $\sigma_V^2(q) + \sigma_R^2(q)/k_0$ by (7), so their sample variance is unbiased for it, and

$$\mathbb{E}\big[\hat{\sigma}_V^2(q)\big] = \mathbb{E}\Big[\frac{1}{r_0 - 1}\sum_{i=1}^{r_0} \big(\bar{Y}_{i\cdot} - \bar{Y}_{\cdot\cdot}\big)^2\Big] - \mathbb{E}\Big[\frac{\hat{\sigma}_R^2(q)}{k_0}\Big]$$
$$= \Big(\sigma_V^2(q) + \frac{\sigma_R^2(q)}{k_0}\Big) - \frac{\sigma_R^2(q)}{k_0} = \sigma_V^2(q). \tag{20}$$

The benchmark estimates $\hat{\sigma}_R^2$ and $\hat{\sigma}_V^2$ are the means of these per-question estimates over the n_c calibration questions, so they are unbiased for σ_R^2 and σ_V^2 when these questions are drawn at random from the benchmark. A single question often gives a negative $\hat{\sigma}_V^2(q)$ when its $\sigma_V^2(q)$ is small, and averaging before clipping keeps these values from biasing the mean. The mean is set to zero only if it is negative, which in our pools happened only on the two negative controls (Tables 11 and 12). The design uses $r_0 + r_0 k_0$ calls per question, r_0 readings and $r_0 k_0$ chains, 136 for $(8, 16)$, and $512 \times 136 = 69{,}632$ calls in total at the reference setting, or 147,502 s at the measured costs.

B.6 Averaging over Questions and Heterogeneous Costs

The benchmark objective. The rule minimises the mean of the per-question vote variances. Since $v_q(r, k)$ is linear in the two components,

$$\mathbb{E}_q\big[v_q(r,k)\big] = \frac{\mathbb{E}_q[\sigma_V^2(q)]}{r} + \frac{\mathbb{E}_q[\sigma_R^2(q)]}{rk} = \frac{\sigma_V^2}{r} + \frac{\sigma_R^2}{rk} = v(r,k). \tag{21}$$

For a question drawn at random from the benchmark, the variance of $\bar{Y}_{r,k}$ is $\mathbb{E}_q[v_q(r,k)] + \mathrm{Var}_q(\mu_q)$, and the second term does not depend on (r, k). The schedule that minimises $v(r, k)$ therefore also minimises this total, whatever the spread of question difficulty, which the rule never needs to model.

Costs that vary across questions. If the costs vary across questions and every question gets the cap C with one global k, question q receives $r_q = C/(c_V(q) + k c_R(q))$ readings. The same algebra as in (14) gives

$$\begin{aligned}\mathbb{E}_q\big[v_q(r_q,k)\big] &= \frac{1}{C}\,\mathbb{E}_q\Big[\big(c_V(q) + k c_R(q)\big)\Big(\sigma_V^2(q) + \frac{\sigma_R^2(q)}{k}\Big)\Big] \\ &= \frac{1}{C}\Big(\mathbb{E}_q[c_V\sigma_V^2] + \mathbb{E}_q[c_R\sigma_R^2] + k\,\mathbb{E}_q[c_R\sigma_V^2] + \frac{\mathbb{E}_q[c_V\sigma_R^2]}{k}\Big),\end{aligned} \tag{22}$$

which is minimised at $k = \sqrt{\mathbb{E}_q[c_V\sigma_R^2]/\mathbb{E}_q[c_R\sigma_V^2]}$. With constant costs this is $k^\star$ of Theorem 2.

B.7 A Model for the Log-Linear Relation

Section 3.2 reports that accuracy is close to linear in $\ln v(r, k)$ on the three measured grids. This subsection gives a simple model under which the relation holds approximately. Under a normal approximation to $\bar{Y}_{r,k}$, the vote of question q is correct with probability close to $\Phi(\delta_q/\sqrt{v_q})$, where Φ is the standard normal distribution function and ϕ its density. Suppose that the schedule scales the variance of every question by a common factor h, so that $v_q = \tau_q^2 h$ with a per-question constant τ_q^2, as happens when the questions share one intraclass correlation. With the standardised margin $t_q = \delta_q/\tau_q$, the accuracy is $A(h) = \mathbb{E}_q[\Phi(t_q/\sqrt{h})]$ and

$$\frac{\mathrm{d}A}{\mathrm{d}\ln h} = -\frac{1}{2}\,\mathbb{E}_q\Big[\phi\Big(\frac{t_q}{\sqrt{h}}\Big)\frac{t_q}{\sqrt{h}}\Big]. \tag{23}$$

Let a share w of the questions have margins spread evenly on a logarithmic scale between t_1 and t_2, with density $w/(t\ln(t_2/t_1))$. Their contribution to (23) is, with $x = t/\sqrt{h}$,

$$\begin{aligned}-\frac{1}{2}\int_{t_1}^{t_2}\phi\Big(\frac{t}{\sqrt{h}}\Big)\frac{t}{\sqrt{h}}\cdot\frac{w}{t\ln(t_2/t_1)}\,\mathrm{d}t &= -\frac{w}{2\ln(t_2/t_1)}\int_{t_1/\sqrt{h}}^{t_2/\sqrt{h}}\phi(x)\,\mathrm{d}x \\ &= -\frac{w}{2\ln(t_2/t_1)}\Big[\Phi\Big(\frac{t_2}{\sqrt{h}}\Big) - \Phi\Big(\frac{t_1}{\sqrt{h}}\Big)\Big] \\ &\approx -\frac{w}{4\ln(t_2/t_1)},\end{aligned} \tag{24}$$

where the last step holds when $\sqrt{h}$ lies well inside $[t_1, t_2]$. The slope is then close to constant, so the accuracy is close to affine in $\ln h$ and hence in $\ln v$. Questions whose margins are negative contribute with the opposite sign, as the last part of Theorem 1 states for a single question. A spread of margins that is even on a logarithmic scale is therefore enough for the log-linear relation, and the slope measures how much of the benchmark sits near the decision threshold.

C ACCURACY GRIDS AND THE LOG-LINEAR FIT

The grids. Table 6 lists every cell of the $r \times k$ grid on the three benchmarks with full grids (MathVista, MMMU and ChartQA), including the cells that do not fit the cap. Its first row uses a single reading, and its first column shows what readings add at one chain each: on MathVista, eight readings with one chain each beat sixteen chains on one reading by 1.96 points at a slightly lower cost.

Table 6: The full $r \times k$ grid of accuracies (%) on the three grid benchmarks with the 7B reader, including the cells above the reference cap. At $k = 1$ row r costs $r(c_V + c_R)$, and every further chain per reading adds rc_R.

	MathVista					MMMU					ChartQA				
r	k=1	k=2	k=4	k=8	k=16	k=1	k=2	k=4	k=8	k=16	k=1	k=2	k=4	k=8	k=16
1	61.98	62.70	63.21	63.53	63.72	59.21	59.50	59.66	59.74	59.86	84.10	84.84	84.97	85.33	85.45
2	63.23	63.95	64.46	64.63	64.87	60.35	60.57	60.96	60.66	60.74	85.60	85.98	86.55	86.96	87.11
3	63.96	64.68	65.18	65.62	65.69	60.65	60.98	61.16	61.26	61.29	86.30	86.96	87.22	87.57	87.69
4	64.47	65.19	65.57	66.11	66.33	61.29	61.59	61.55	62.05	61.77	87.53	87.64	88.24	88.51	88.44
5	64.87	65.59	65.91	66.53	66.46	61.42	61.78	61.83	62.28	62.14	87.37	88.30	88.45	88.59	88.64
6	65.20	65.96	66.43	66.88	66.86	61.99	62.05	62.31	62.27	62.31	87.83	88.29	88.81	89.05	89.15
7	65.48	66.15	66.80	66.87	67.33	62.01	61.84	62.41	62.79	62.79	88.30	88.71	89.37	89.46	89.49
8	65.68	66.39	66.78	67.13	67.36	62.21	62.72	62.74	62.83	62.98	88.62	88.98	89.67	89.52	90.10

The log-linear fit. For each benchmark we fit $A(r, k) = \alpha - s \ln v(r, k)$, with intercept α and slope s, to the forty cells of its grid, with $v(r, k)$ computed from the components of Table 1. The slope is $s = 1.78$ points per unit of $\ln v$ on MathVista, 1.45 on MMMU and 2.07 on ChartQA, and the line explains 99.6%, 97.9% and 98.6% of the variance of the cells (Fig. 3a). A line in $v(r, k)$ explains 83% to 86%. The relation turns any change of schedule into a predicted change of accuracy, $s \ln(v_{\text{before}}/v_{\text{after}})$. With the grid slope, this prediction matches the gains measured under the nine reading distributions of Table 25, eight of which lie off the grid used for the fit, within 0.06 points, and the effect of reading noise within 0.05 points on MathVista and MMMU and within 0.17 points on ChartQA (Fig. 3b and Table 7). Appendix B.7 gives a model for it.

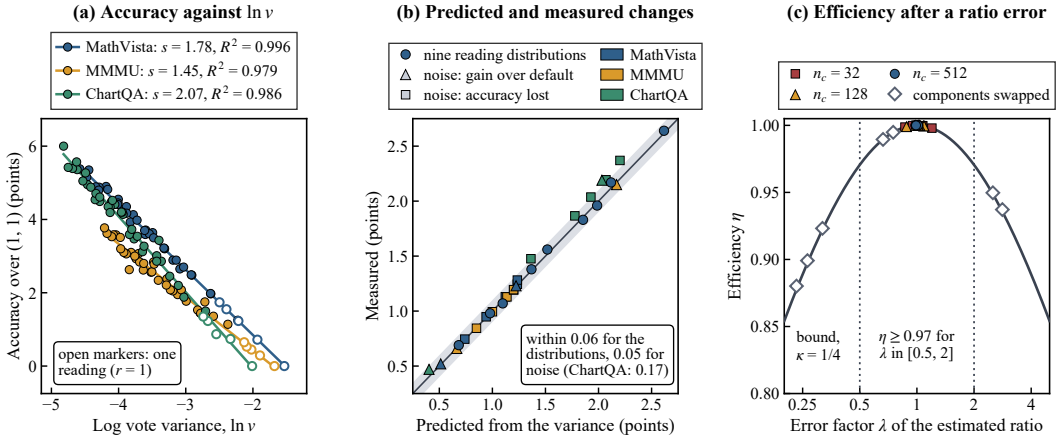

Figure 3: Checks of the analysis. (a) Accuracy of the forty grid cells against $\ln v(r, k)$, relative to the $(1, 1)$ cell, with least-squares lines; open markers are single-reading cells. (b) Measured against predicted changes of accuracy, $s \ln(v_{\text{before}}/v_{\text{after}})$ with the slope s of (a), for the nine reading distributions on MathVista and for added reading noise on three benchmarks; the line marks equality and the band ± 0.1 points. (c) Efficiency η of the continuous optimum when the estimated ratio $\hat{\sigma}_R^2/\hat{\sigma}_V^2$ is off by a factor λ: the bound of Proposition 2 and the values reached by the calibration estimates at $n_c = 32$, 128 and 512 and by swapped components.

Every feasible pair. Table 8 lists the 35 integer pairs that fit the cap at the reference setting, sorted by their vote variance, with the relative efficiency η of Theorem 2. The order by variance is nearly the order by accuracy: over the 19 pairs whose accuracy is listed, the Spearman correlation is -0.995. The two best pairs show why the bracket alone is not enough. The pair $(5, 2)$ has the smaller bracket in (3), but it spends 91% of the cap and reaches $\eta = 0.890$, while $(8, 1)$ spends 99% and reaches $\eta = 0.954$. The oracle schedule $(4, 3)$ ranks third by variance and is 0.04 points above $(8, 1)$ in accuracy. All pairs with $r = 1$ rank 19th or lower, below every pair with $r \geq 2$ except $(2, 1)$, because their chains divide only the second term of (1), so their vote variance stays above $\hat{\sigma}_V^2 = 0.0734$.

Table 7: Predicted and measured changes of accuracy (points, 7B reader) behind Fig. 3(b). Predictions are $s \ln(v_{\text{before}}/v_{\text{after}})$ with the slope s of each benchmark (Fig. 3a). Left: gain over the default for nine reading distributions on MathVista. Right: accuracy lost to reading noise at five levels and gain over the default at two levels.

Setting	Predicted	Measured
Prompt, rewrite 1	1.99	1.96
Prompt, rewrite 2	1.37	1.38
Prompt, rewrite 3	0.98	0.98
Temperature 0.3	0.68	0.69
Temperature 0.7	1.85	1.83
Temperature 1.0	2.62	2.64
Crop, centre	1.10	1.07
Crop, overlapping	1.52	1.56
Crop, disjoint	2.12	2.17

	MathVista		MMMU		ChartQA	
Noise	Pred.	Meas.	Pred.	Meas.	Pred.	Meas.
Accuracy lost to the noise						
0.2	0.74	0.75	0.85	0.84	1.36	1.48
0.4	0.94	0.95	1.00	0.99	1.78	1.87
0.6	1.12	1.13	1.14	1.13	1.93	2.04
0.8	1.24	1.24	1.20	1.19	2.07	2.19
1	1.24	1.28	1.20	1.20	2.20	2.37
Gain over the default						
0.2	0.51	0.52	0.67	0.66	0.40	0.47
0.5	1.22	1.23				
1			2.17	2.15	2.03	2.19

Table 8: Every feasible integer pair at the reference setting, sorted by their vote variance $v(r,k)$ from lowest to highest. Cost is $r(c_V + kc_R)$ with $c_V = 2.349$ s and $c_R = 2.104$ s, against 36.01 s, the cap minus the calibration charge. The efficiency η of Theorem 2 is computed from $v(r,k)$ and $v_C^\star = 0.02574$. Accuracy (%) is listed for the pairs on the grid of Fig. 1(a) and for the oracle schedule $(4,3)$, and left blank for the others.

Rank	r	k	Cost (s)	$v(r,k)$	η	Acc.
1	8	1	35.62	0.02699	0.954	65.68
2	5	2	32.78	0.02892	0.890	65.59
3	4	3	34.64	0.03022	0.852	65.72
4	7	1	31.17	0.03084	0.835	65.48
5	6	1	26.72	0.03598	0.715	65.20
6	4	2	26.23	0.03616	0.712	65.19
7	3	4	32.29	0.03633	0.708	65.18
8	3	3	25.98	0.04029	0.639	
9	5	1	22.26	0.04318	0.596	64.87
10	2	7	34.15	0.04686	0.549	
11	3	2	19.67	0.04821	0.534	64.68
12	2	6	29.94	0.04856	0.530	
13	2	5	25.74	0.05094	0.505	
14	4	1	17.81	0.05397	0.477	64.47
15	2	4	21.53	0.05450	0.472	64.46
16	2	3	17.32	0.06044	0.426	
17	3	1	13.36	0.07196	0.358	63.96
18	2	2	13.11	0.07231	0.356	63.95
19	1	16	36.01	0.08228	0.313	63.72
20	1	15	33.91	0.08287	0.311	
21	1	14	31.80	0.08355	0.308	
22	1	13	29.70	0.08433	0.305	
23	1	12	27.60	0.08525	0.302	
24	1	11	25.49	0.08633	0.298	
25	1	10	23.39	0.08762	0.294	
26	1	9	21.28	0.08920	0.289	
27	1	8	19.18	0.09118	0.282	63.53
28	1	7	17.08	0.09373	0.275	
29	1	6	14.97	0.09712	0.265	
30	1	5	12.87	0.10187	0.253	
31	2	1	8.91	0.10794	0.238	63.23
32	1	4	10.76	0.10900	0.236	63.21
33	1	3	8.66	0.12087	0.213	
34	1	2	6.56	0.14463	0.178	62.70
35	1	1	4.45	0.21588	0.119	61.98

Schedules per benchmark. Table 9 lists the pair chosen by the variance rule on each benchmark and the k chosen by the accuracy pick. The two rules differ by one chain per reading on six of the seven main benchmarks and by the number of readings on AI2D, and the variance rule is ahead by 0.09 to 0.37 points on each of them. On the two negative controls the continuous $k^\star$ exceeds 300, and the variance rule keeps the default $(1,16)$, since only the total number of chains then matters.

Table 9: Schedules of the variance rule and of the accuracy pick (7B reader, reference budget). The pair chosen by the variance rule, the continuous $k^\star$ at the reference costs, the k chosen by the accuracy pick, the accuracy (%) of the two rules, and their difference in points. The accuracy pick uses $(5,2)$ on MathVista.

Benchmark	Variance rule	$k^\star$	Pick k	Acc. rule	Acc. pick	Difference
MathVista	(8,1)	1.47	2	65.68	65.59	0.09
MMMU	(7,1)	0.84	2	62.01	61.84	0.17
MathVerse	(6,1)	1.14	2	52.39	52.23	0.16
ChartQA	(4,2)	1.17	1	87.64	87.53	0.11
ScienceQA	(5,2)	1.52	1	84.44	84.26	0.18
AI2D	(5,1)	0.82	1	81.13	80.76	0.37
MMBench	(8,1)	1.41	2	81.08	80.87	0.21
SEED	(1,16)	> 300	4	74.18	74.17	0.01
RealWorldQA	(1,16)	> 300	3	59.45	59.34	0.11

D EXPERIMENTAL DETAILS

Benchmarks and models. The nine benchmarks cover mathematical reasoning over figures and plots (MathVista, MathVerse), expert questions across disciplines (MMMU), charts (ChartQA), science diagrams (ScienceQA, AI2D), and general perception (MMBench, SEED-Bench, RealWorldQA) (Lu et al., 2024; Zhang et al., 2024a; Yue et al., 2024; Masry et al., 2022; Lu et al., 2022; Kembhavi et al., 2016; Liu et al., 2024b; Li et al., 2024a; xAI, 2024). The two main readers are Qwen2.5-VL-7B and Qwen2.5-VL-72B (Bai et al., 2025b). The seven other readers are Qwen3-VL-8B, Qwen3-VL-32B and Qwen3-VL-30B-A3B (Bai et al., 2025a), InternVL3-8B and InternVL3-78B (Zhu et al., 2025), Gemma-3-27B (Gemma Team et al., 2025) and Llama-4-Scout (Meta AI, 2025). The main reasoner is the text-only Qwen2.5-7B, and the three other reasoners are Qwen2.5-72B (Qwen Team et al., 2024), Llama-3.3-70B (Grattafiori et al., 2024) and R1-Distill-Qwen-32B (Guo et al., 2025).

Readings and chains. A reading is a text produced by the reader from the image and an extraction prompt, and a chain is a reasoning trace of the reasoner over the question and one reading. Table 25 lists nine reading distributions: three prompt rewrites, three decoding temperatures (0.3, 0.7 and 1.0) and three kinds of crop (centre, overlapping and disjoint). All nine are measured on the same reader and calibration questions, and prompt rewrite 1 is the distribution of the main results. The default aggregator is a majority vote over the final answers of all chains of a schedule.

Costs and budgets. Costs are wall-clock seconds per call, measured on three serving stacks at batch sizes 1, 4 and 16 with prefix caching (Kwon et al., 2023); Table 10 lists them for MathVista. The reference setting uses the standard stack at batch size 1, where $c_V = 2.349\,\text{s}$ and $c_R = 2.104\,\text{s}$. The five budget levels are 0.55, 0.78, 1, 1.35 and 1.8 times the reference cap of each benchmark, and Table 17 lists the caps of six benchmarks. A rule is charged $r(c_V + kc_R)$ for the schedule it selects plus the calibration charge of 1.475 s per question; the extra models of four external rules are not charged.

Calibration. Algorithm 1 lists the rule. The calibration pool holds $n_c = 512$ questions with $r_0 = 8$ readings and $k_0 = 16$ chains on each, and Table 29 checks the variance model beyond this design.

Algorithm 1 Reobserve

1: **Input:** reader, reasoner and aggregator; reading distribution; calibration questions $q = 1, \ldots, n_c$ with answers; cap C per question; number N of deployment questions.
2: For each calibration question, draw r_0 readings and k_0 chains on each; record Y_{ij}; measure c_V and c_R.
3: Compute $\hat{\sigma}_R^2(q)$ and $\hat{\sigma}_V^2(q)$ by (5) for each question, average them over the n_c questions, and set a negative average $\hat{\sigma}_V^2$ to zero.
4: $c_{\text{cal}} \leftarrow n_c\,(r_0 c_V + r_0 k_0 c_R)/N$.
5: **for** every integer pair (r, k) with $r \le 8$, $k \le 16$ and $r(c_V + kc_R) \le C - c_{\text{cal}}$ **do**
6: $\quad \hat{v}(r, k) \leftarrow \hat{\sigma}_V^2/r + \hat{\sigma}_R^2/(rk)$
7: **end for**
8: $(\hat{r}, \hat{k}) \leftarrow \arg\min \hat{v}(r, k)$, with ties broken toward the larger r.
9: **Deployment:** for each new question, draw $\hat{r}$ readings and $\hat{k}$ chains on each, and return the aggregator's answer.

The rule at the reference setting. On MathVista with the 7B reader, the 512-question calibration gives $\hat{\sigma}_V^2 = 0.07396$ and $\hat{\sigma}_R^2 = 0.14150$ (Table 11) at a cost of 147,502 s, or 1.475 s per question over 10^5 deployment questions. This leaves 36.01 s of the 37.49 s cap for the schedule. With $c_V = 2.349\,\text{s}$ and $c_R = 2.104\,\text{s}$ the continuous optimum is $k^\star = 1.46$ (1.47 with the reference components of Table 1), and among the 35 feasible pairs of Table 8 the predicted variance is smallest at $(8, 1)$, which costs 35.62 s and spends 99% of what is left of the cap. The default $(1, 16)$ spends all of it but has three times the predicted variance. At deployment $(8, 1)$ scores 65.68 against 63.72 for the default, a gain of 1.96 points with the 95% interval $[1.47, 2.45]$.

Baselines. All rules draw their readings and chains from the same evaluation pool under the same cap, and each is scored by the majority-vote accuracy of the schedule it selects. The default spends the cap on chains over one reading, with at most sixteen chains, the size of the pool, so at the two largest budgets it leaves part of the cap unspent. The fixed re-reading schedule uses seven readings with one chain each, the random rule draws a feasible pair at random, the equal split sets $r = k$, the σ-proportional split sets $r : k = \sigma_V : \sigma_R$, a ratio that rounds to $(1, 1)$ at the reference

setting, and one global k uses the same number of chains per reading on all benchmarks. The eleven external methods enter only through the schedules that their own selection produces on the pool; Adaptive-Consistency and early-stopping self-consistency, for example, stop drawing chains once the vote is settled. The two reward-based rules and the two re-reads also use a model beyond the reader and the reasoner. Table 13 lists the schedule and the cost of every rule at the reference setting.

Oracle and intervals. The oracle schedule is the feasible pair with the highest accuracy in a separate search over all feasible pairs. Intervals are 95% normal-approximation intervals of the paired difference between two rules over the same questions, $\bar{d} \pm 1.96\, s_d/\sqrt{n}$, where $\bar{d}$ and s_d are the mean and the standard deviation of the per-question differences in correctness over the n questions.

Table 10: Costs on MathVista. Cost of one reading and of one chain, and cost of the schedule $(8, 1)$ including the calibration charge of 1.475 s, in seconds, for three serving stacks and three batch sizes with prefix caching on. The reference setting uses the standard stack at batch size 1; the ratio c_V/c_R stays between 0.88 and 1.16.

Stack	Batch	c_V (s)	c_R (s)	$(8, 1)$ (s)
Standard	1	2.349	2.104	37.10
Standard	4	1.342	1.337	22.91
Standard	16	0.839	0.885	15.27
Fused	1	1.932	1.670	30.29
Fused	4	1.104	1.062	18.80
Fused	16	0.690	0.702	12.61
Portable	1	2.263	2.183	37.04
Portable	4	1.293	1.388	22.92
Portable	16	0.808	0.918	15.28

E Estimator Checks and Calibration

Checks of the estimator. On a control whose components are known by construction, the estimator recovers σ_V^2 within 1.6×10^{-4}. On the full calibration pool, the autocorrelation between chains on one reading, which would break the conditional independence behind (1), is -2.5×10^{-4}.

Calibration size. Table 11 and Fig. 4 show how the estimates depend on the number of calibration questions. With 512 questions, $\hat{\sigma}_V^2$ is within 1.1% of its reference value on all seven main benchmarks, and $\hat{\sigma}_R^2$ is within 1% on eight of nine. With 32 questions the error of $\hat{\sigma}_V^2$ reaches 12.4% on ChartQA. The predicted mean variance of the vote at the calibration design matches the observed value within 4.6% on average at 512 questions and within 28.7% at 32. These errors enter the schedule only through the ratio of the two components, where Proposition 2 makes them second order. The schedules chosen from 64, 128 and 512 questions are the same on every benchmark (Table 23).

Nested designs. Table 12 and Fig. 5 compare the designs $(4, 4)$, $(8, 16)$ and $(8, 32)$ on 128 calibration questions. The three designs recover σ_V^2 within 4.7%, 3.3% and 2.8% on average. On the seven main benchmarks, going from $(4, 4)$ to $(8, 16)$ lowers the predicted variance of the vote at the design by 59% on average, and doubling the chains again lowers it by a further 5%. The design $(4, 4)$ costs 5,511 s against 36,876 s for $(8, 16)$ on the same questions, and it changes the schedule of one benchmark without changing the mean accuracy (Table 23), so the cheaper design serves about as well.

Further checks at one reading. Fig. 6 gives three further views of what chains can do at one reading. The first extra chain buys 41% to 55% of the whole gain from one to sixteen chains, and half of that gain is reached by $k = 3$ on the three benchmarks with full grids. At three levels of reading variance, with the reference level in the middle, a higher level leaves fewer points for extra chains to gain. Finally, shuffling which chain belongs to which reading before estimation lowers $\hat{\sigma}_V^2$ by a factor of 367 to 1,017 on the seven main benchmarks and leaves it below 3×10^{-4} on all nine. The between-reading component is therefore a property of how chains group into readings.

Table 11: Calibration size. Estimated components at $n_c = 32$, 128 and 512 calibration questions with the design $(r_0, k_0) = (8, 16)$, whether the raw $\hat{\sigma}_V^2$ was negative and clipped to zero, and the predicted against the observed mean variance of the vote at the calibration design. Clipping occurs only on the two negative controls.

Benchmark	n_c	$\hat{\sigma}_V^2$	$\hat{\sigma}_R^2$	Clipped	Predicted	Observed
MathVista	32	0.07442	0.14282	no	0.01042	0.00846
	128	0.06822	0.14369	no	0.00965	0.00845
	512	0.07396	0.14150	no	0.01035	0.01023
MMMU	32	0.11373	0.06209	no	0.01470	0.01876
	128	0.11447	0.07406	no	0.01489	0.01595
	512	0.11357	0.07282	no	0.01477	0.01511
MathVerse	32	0.09067	0.11250	no	0.01221	0.01229
	128	0.08906	0.11222	no	0.01201	0.01144
	512	0.09468	0.10724	no	0.01267	0.01374
ChartQA	32	0.05254	0.07736	no	0.00717	0.01221
	128	0.06452	0.06981	no	0.00861	0.00934
	512	0.05946	0.07436	no	0.00801	0.00823
ScienceQA	32	0.07710	0.15887	no	0.01088	0.01258
	128	0.07340	0.15753	no	0.01041	0.01112
	512	0.07581	0.15546	no	0.01069	0.01030
AI2D	32	0.14151	0.08929	no	0.01839	0.02418
	128	0.14294	0.08366	no	0.01852	0.01670
	512	0.14139	0.08539	no	0.01834	0.01674
MMBench	32	0.07618	0.12799	no	0.01052	0.00811
	128	0.07471	0.13476	no	0.01039	0.01026
	512	0.07483	0.13124	no	0.01038	0.00953
SEED	32	0.00000	0.23626	yes	0.00185	0.00258
	128	0.00023	0.23838	no	0.00189	0.00209
	512	0.000003	0.24052	no	0.00188	0.00177
RealWorldQA	32	0.00000	0.18211	yes	0.00142	0.00186
	128	0.00000	0.17874	yes	0.00140	0.00140
	512	0.000001	0.17779	no	0.00139	0.00140

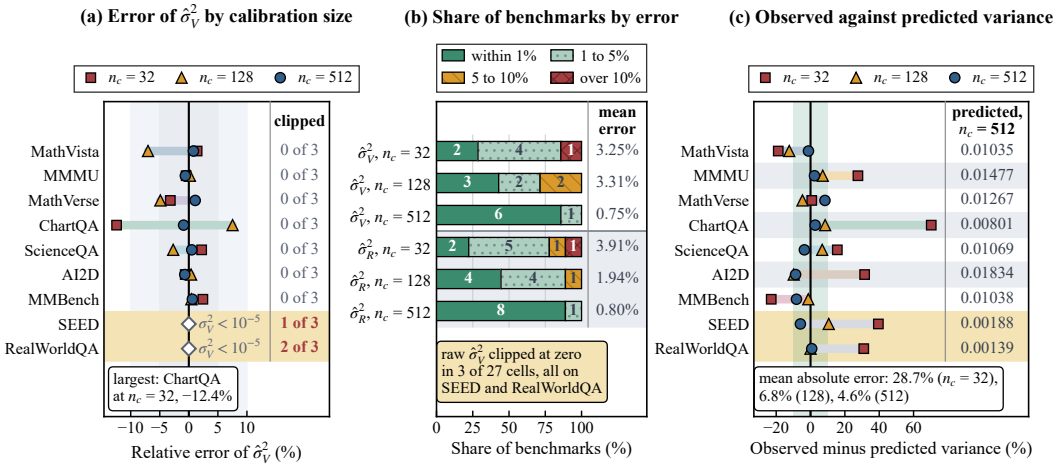

Figure 4: Calibration size. (a) Relative error of $\hat{\sigma}_V^2$ at $n_c = 32$, 128 and 512, with the number of sizes at which it was clipped to zero; the bands mark 5% and 10%. (b) Share of the seven main benchmarks ($\hat{\sigma}_V^2$) and of all nine ($\hat{\sigma}_R^2$) in each error range at each size, with the mean error. (c) Observed minus predicted mean vote variance at the calibration design, relative to the prediction, with the predicted value at $n_c = 512$; the band marks 10%.

Table 12: Nested designs on 128 calibration questions. Estimated components under $(r_0, k_0) = (4, 4), (8, 16)$ and $(8, 32)$, whether $\hat{\sigma}_V^2$ was clipped to zero, and the predicted and observed mean vote variance at each design.

Benchmark	(r_0, k_0)	$\hat{\sigma}_V^2$	$\hat{\sigma}_R^2$	Clipped	Predicted	Observed
MathVista	$(4, 4)$	0.07119	0.14404	no	0.02680	0.02488
	$(8, 16)$	0.06822	0.14369	no	0.00965	0.00845
	$(8, 32)$	0.07230	0.14335	no	0.00960	0.01276
MMMU	$(4, 4)$	0.10538	0.07764	no	0.03120	0.03076
	$(8, 16)$	0.11447	0.07406	no	0.01489	0.01595
	$(8, 32)$	0.11432	0.07367	no	0.01458	0.01252
MathVerse	$(4, 4)$	0.09661	0.10726	no	0.03086	0.03536
	$(8, 16)$	0.08906	0.11222	no	0.01201	0.01144
	$(8, 32)$	0.08886	0.10765	no	0.01153	0.01038
ChartQA	$(4, 4)$	0.05857	0.07618	no	0.01940	0.01920
	$(8, 16)$	0.06452	0.06981	no	0.00861	0.00934
	$(8, 32)$	0.05749	0.07481	no	0.00748	0.00773
ScienceQA	$(4, 4)$	0.08129	0.14762	no	0.02955	0.02782
	$(8, 16)$	0.07340	0.15753	no	0.01041	0.01112
	$(8, 32)$	0.07569	0.15952	no	0.01008	0.01124
AI2D	$(4, 4)$	0.13317	0.09180	no	0.03903	0.04035
	$(8, 16)$	0.14294	0.08366	no	0.01852	0.01670
	$(8, 32)$	0.14580	0.07892	no	0.01853	0.01913
MMBench	$(4, 4)$	0.07620	0.12972	no	0.02716	0.03020
	$(8, 16)$	0.07471	0.13476	no	0.01039	0.01026
	$(8, 32)$	0.07001	0.13595	no	0.00928	0.00836
SEED	$(4, 4)$	0.00012	0.24121	no	0.01511	0.01409
	$(8, 16)$	0.00023	0.23838	no	0.00189	0.00209
	$(8, 32)$	0.00000	0.23839	yes	0.00093	0.00107
RealWorldQA	$(4, 4)$	0.00219	0.18294	no	0.01198	0.01142
	$(8, 16)$	0.00000	0.17874	yes	0.00140	0.00140
	$(8, 32)$	0.00000	0.18095	yes	0.00071	0.00079

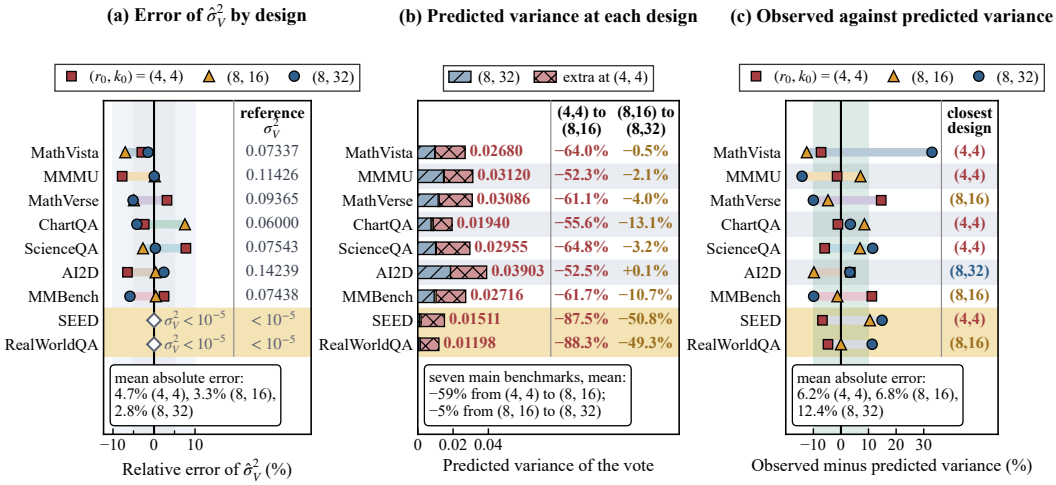

Figure 5: Nested designs on 128 calibration questions. (a) Relative error of $\hat{\sigma}_V^2$ against its reference value under $(r_0, k_0) = (4, 4), (8, 16)$ and $(8, 32)$, with the reference σ_V^2 and bands at 5% and 10%. (b) Predicted variance of the vote at $(8, 32)$ and its excess at $(4, 4)$; the columns give the change from $(4, 4)$ to $(8, 16)$ and from $(8, 16)$ to $(8, 32)$. (c) Observed minus predicted variance at each design, relative to the prediction, with a band at 10% and the design closest to its prediction; the mean absolute gaps of the designs are 6.2%, 6.8% and 12.4%.

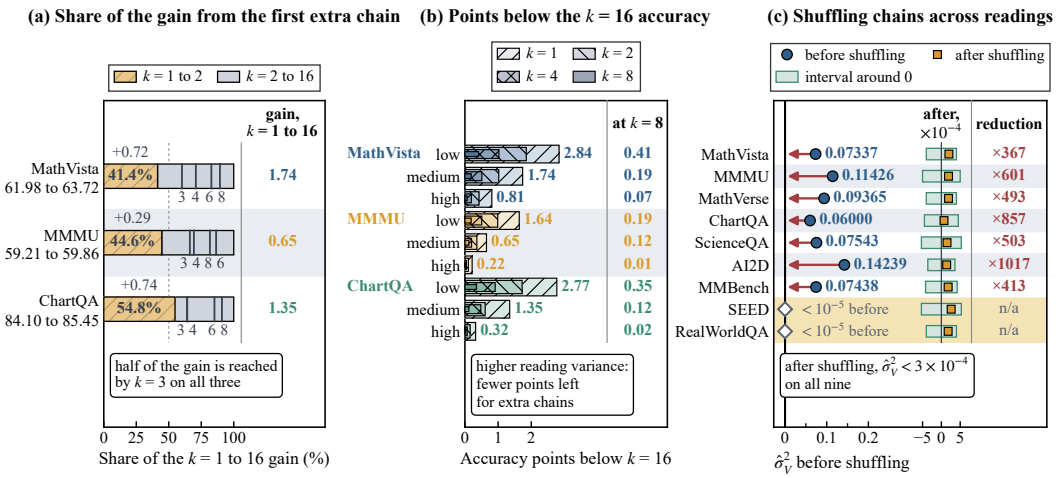

Figure 6: Further checks at one reading (7B reader). (a) Share of the $k = 1$ to 16 gain bought by the first extra chain, whose gain in points is printed, with ticks at the cumulative share for $k = 3$, 4, 6 and 8; under each name, the accuracy at $k = 1$ and $k = 16$. (b) Accuracy points below the $k = 16$ accuracy at $k = 1$, 2, 4 and 8, with the value at $k = 1$ at each bar end and at $k = 8$ on the right, at a low, a medium (the reference) and a high level of reading variance. (c) $\hat{\sigma}^2_V$ before and after shuffling which chain belongs to which reading; the inset shows the shuffled values inside their intervals around zero, and the right column gives the reduction factor.

F FULL RESULTS

Schedules, gains and their decomposition. Table 13 lists the schedule and the cost of every rule at the reference setting. Nine rules use a single reading at this setting, and the others use between two and eight. Table 14 gives the gain of the variance rule over the default in the 27 cells of nine benchmarks and three readers (the two Qwen2.5-VL sizes and Qwen3-VL-8B). All 21 intervals of the seven main benchmarks lie above 1.3 points, and the intervals of the negative controls contain zero. On the seven main benchmarks, Table 15 splits each gain into the part due to more readings, the part due to fewer chains per reading, and the rounding to integers.

Table 13: Schedule and cost of every rule at the reference setting (MathVista, 7B reader, cap 37.49 s). Readings r, chains per reading k, total chains rk, cost $r(c_V + kc_R)$ plus the calibration charge, and whether the rule needs a model beyond the reader and the reasoner. The external rules use the schedules chosen by their own criteria on the evaluation pool; the grey row is the oracle schedule, which is not deployable.

Rule	r	k	rk	Cost (s)	Extra model
Default $(1, K)$	1	16	16	37.49	no
Fixed re-reading $(7, 1)$	7	1	7	32.65	no
Random (r, k)	1	16	16	37.49	no
Equal split	3	3	9	27.46	no
σ-proportional split	1	1	1	5.93	no
One global k	3	4	12	33.77	no
Accuracy pick	5	2	10	34.26	no
Oracle schedule	4	3	12	36.12	no
Adaptive-Consistency	1	8	8	20.66	no
Early-stopping SC	1	3	3	10.14	no
Compute-optimal scaling	1	16	16	37.49	no
Confidence-weighted vote	1	4	4	12.24	no
Outcome-reward best-of-n	2	3	6	18.80	yes
Process-reward vote	2	6	12	31.42	yes
Universal SC	3	4	12	33.77	no
Dynamic high-res. tiling	4	1	4	19.29	no
Multi-crop ensemble	3	2	6	21.15	no
Self-verification re-read	1	6	6	16.45	yes
Visual CoT re-read	1	6	6	16.45	yes
Reobserve	8	1	8	37.10	no

Table 14: Gain of the variance rule over the default (points) with 95% intervals, on nine benchmarks and three readers at the reference budget. The mean over the eighteen cells of Table 2 is 1.84, and every interval on the seven main benchmarks lies above 1.3 points. Lower and Upper are the ends of the interval.

	Qwen2.5-VL-7B			Qwen2.5-VL-72B			Qwen3-VL-8B		
Benchmark	Gain	Lower	Upper	Gain	Lower	Upper	Gain	Lower	Upper
MathVista	+1.96	+1.47	+2.45	+1.91	+1.38	+2.44	+2.06	+1.54	+2.58
MMMU	+2.15	+1.62	+2.68	+2.24	+1.63	+2.85	+2.41	+1.82	+3.00
MathVerse	+2.55	+1.98	+3.13	+2.49	+1.88	+3.10	+2.50	+1.94	+3.05
ChartQA	+2.19	+1.59	+2.78	+2.14	+1.54	+2.74	+2.09	+1.57	+2.60
ScienceQA	+2.63	+2.00	+3.25	+2.72	+2.07	+3.37	+2.28	+1.68	+2.89
AI2D	+2.21	+1.66	+2.77	+2.18	+1.59	+2.77	+2.02	+1.43	+2.60
MMBench	+2.88	+2.28	+3.48	+2.95	+2.29	+3.61	+2.74	+2.13	+3.36
SEED	+0.01	−0.07	+0.09	−0.06	−0.14	+0.02	+0.04	−0.04	+0.12
RealWorldQA	−0.02	−0.10	+0.06	−0.02	−0.10	+0.06	−0.03	−0.11	+0.05

Five budget levels. Table 16 and Figs. 7 and 8 repeat the comparison at five budgets. The gain of the rule grows from 0.81 to 2.65 points with the budget. The default gains 0.19 points up to the reference level and none beyond it, where it already uses the sixteen chains of the pool. At the reference budget the rule is ahead of the best baseline on all seven main benchmarks and its schedules spend 98% of the cap; at the other budgets it is ahead or level on four to seven of the nine benchmarks. Table 17 gives every split on six benchmarks at every budget. The σ-proportional split falls below the default only at the reference budget, where it rounds to $(1, 1)$, and it is close to the

Table 15: Decomposition of the gain of the variance rule over the default (points) on the seven main benchmarks into the move from one reading to $r^\star$ readings, the move from K to $k^\star$ chains per reading, and the rounding to integers, with the share of the total due to the readings. The rounding part is a small loss on every benchmark.

Benchmark	Readings	Chains	Rounding	Total	Readings share (%)
MathVista	$+1.46$	$+0.55$	-0.05	$+1.96$	74
MMMU	$+1.86$	$+0.31$	-0.02	$+2.15$	87
MathVerse	$+2.24$	$+0.43$	-0.12	$+2.55$	88
ChartQA	$+1.82$	$+0.41$	-0.04	$+2.19$	83
ScienceQA	$+2.18$	$+0.51$	-0.06	$+2.63$	83
AI2D	$+1.93$	$+0.38$	-0.10	$+2.21$	87
MMBench	$+2.25$	$+0.66$	-0.03	$+2.88$	78

oracle schedule at several other budgets. Over the 30 benchmark and budget cells, the variance rule is within 0.25 points of the oracle schedule in 29 cells and within 0.40 points in all of them (Fig. 8c).

Table 16: Five budget levels (7B reader). Mean accuracy (%) over nine benchmarks of the default and of the variance rule, their difference in points, and the number of benchmarks on which the rule is ahead of, level with or behind the best baseline at that budget. Table 17 lists the caps and the splits of six benchmarks.

Budget	Default	Variance rule	Gain	Ahead	Level	Behind
$\times 0.55$	69.97	70.78	$+0.81$	3	1	5
$\times 0.78$	70.03	71.45	$+1.42$	3	2	4
$\times 1$	70.16	72.00	$+1.84$	7	0	2
$\times 1.35$	70.16	72.42	$+2.26$	4	2	3
$\times 1.8$	70.16	72.81	$+2.65$	5	2	2

Table 17: Budget splits at all five budget levels on six benchmarks (7B reader). Accuracy (%) of each split at each cap (seconds per question, calibration charge included); $\times 1$ is the reference budget. Best deployable value per row in bold; grey: the oracle schedule. Acc. pick: the accuracy pick; Var. rule: the variance rule.

Benchmark	Budget	Cap (s)	Default	Equal	σ-prop.	Global k	Acc. pick	Var. rule	Oracle
MathVista	$\times 0.55$	20.62	63.53	63.95	**64.87**	64.46	64.68	64.47	64.87
	$\times 0.78$	29.24	63.53	65.00	**65.20**	64.46	65.19	**65.20**	65.20
	$\times 1$	37.49	63.72	65.00	61.98	65.18	65.59	**65.68**	65.72
	$\times 1.35$	50.61	63.72	65.70	65.72	65.70	65.92	**66.20**	66.24
	$\times 1.8$	67.48	63.72	65.70	65.72	66.10	66.24	**66.44**	66.50
MMMU	$\times 0.55$	19.65	59.80	60.52	**61.21**	60.70	61.10	61.20	61.21
	$\times 0.78$	27.87	59.80	61.22	**61.79**	60.70	61.53	61.76	61.79
	$\times 1$	35.73	59.86	61.22	59.21	61.29	61.84	**62.01**	62.01
	$\times 1.35$	48.24	59.86	61.22	**62.21**	61.29	62.10	62.16	62.21
	$\times 1.8$	64.31	59.86	61.70	62.21	61.70	62.21	**62.30**	62.32
MathVerse	$\times 0.55$	28.02	49.68	50.57	51.49	51.06	51.49	**51.54**	51.54
	$\times 0.78$	39.74	49.68	51.86	**52.02**	51.06	51.86	51.98	52.02
	$\times 1$	50.95	49.84	51.86	48.16	52.03	52.23	**52.39**	52.46
	$\times 1.35$	68.78	49.84	51.86	**52.83**	52.03	52.77	52.80	52.83
	$\times 1.8$	91.71	49.84	52.72	53.15	52.72	53.21	**53.50**	53.58
ChartQA	$\times 0.55$	18.46	85.33	86.19	**86.44**	85.10	**86.44**	86.42	86.44
	$\times 0.78$	26.18	85.33	86.19	87.05	86.58	87.05	**87.06**	87.06
	$\times 1$	33.56	85.45	87.31	84.10	86.58	87.53	**87.64**	87.67
	$\times 1.35$	45.31	85.45	87.31	87.92	87.45	87.92	**88.09**	88.14
	$\times 1.8$	60.41	85.45	88.06	**88.53**	88.06	**88.53**	**88.53**	88.53
ScienceQA	$\times 0.55$	21.72	81.53	82.08	82.81	82.84	82.84	**83.13**	83.13
	$\times 0.78$	30.80	81.53	83.61	83.39	82.84	83.61	**83.83**	83.88
	$\times 1$	39.49	81.81	83.61	79.22	83.89	84.26	**84.44**	84.46
	$\times 1.35$	53.31	81.81	83.61	84.61	83.89	84.61	**84.89**	84.93
	$\times 1.8$	71.08	81.81	84.64	84.61	84.64	84.94	**85.26**	85.33
AI2D	$\times 0.55$	20.85	78.85	79.82	**80.21**	78.72	79.98	80.18	80.21
	$\times 0.78$	29.57	78.85	79.82	**80.76**	80.06	80.61	80.75	80.76
	$\times 1$	37.91	78.92	80.76	78.09	80.06	80.76	**81.13**	81.19
	$\times 1.35$	51.18	78.92	80.76	**81.55**	80.84	81.19	**81.55**	81.55
	$\times 1.8$	68.24	78.92	81.40	**81.84**	81.40	81.59	81.80	81.84

Cost of a reading. Table 18 and Fig. 9 vary the cost of a reading from a quarter to four times its reference value on each benchmark, with the cost of a chain fixed. The continuous $k^\star$ grows with the square root of this cost, as Theorem 2 states, and the rule reads less often: from seven or eight

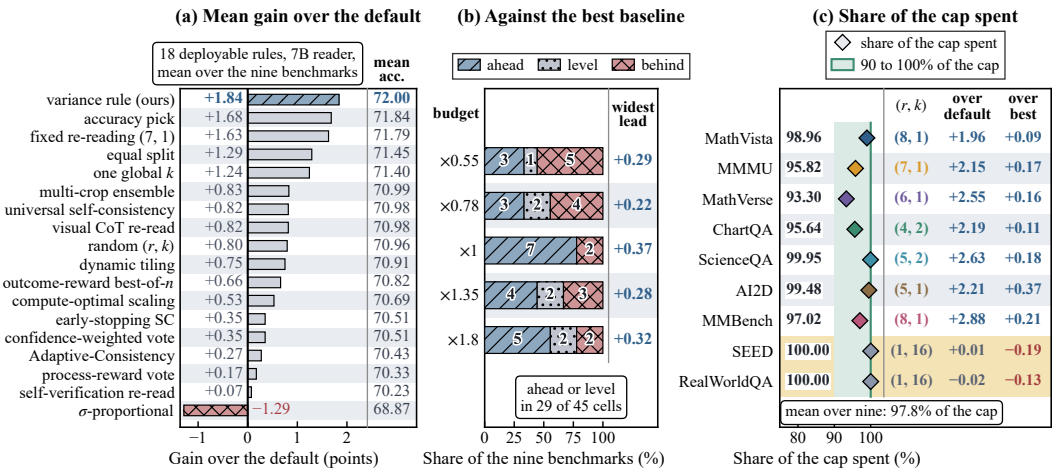

Figure 7: Comparison with all rules (7B reader). (a) Mean gain over the default on nine benchmarks at the reference budget for every deployable rule of Table 2, computed from the accuracies in Tables 2, 9 and 17. (b) Share of the nine benchmarks on which the variance rule is ahead of, level with or behind the best baseline at each of the five budgets, with its widest lead. (c) Share of the cap spent by the chosen schedule at the reference budget, calibration charge included, with the gain over the default and over the best baseline on each benchmark.

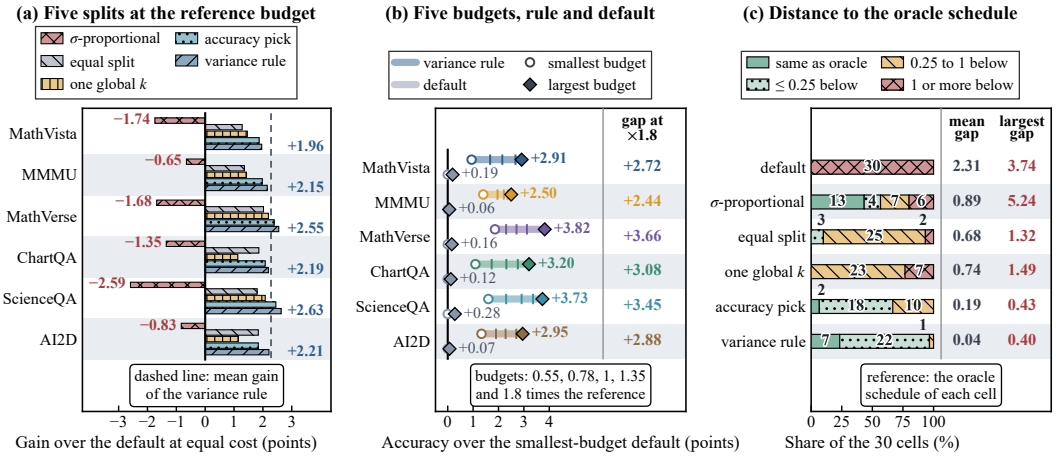

Figure 8: Budget splits at five budgets on six benchmarks (7B reader). (a) Gain over the default of five splits at the reference budget. (b) Accuracy over the smallest-budget default at the five budget levels (ticks mark the middle three), for the variance rule and the default; the right column gives their gap at the largest budget. (c) Distance of each split to the oracle schedule over the 30 benchmark and budget cells, with the mean and largest gaps.

readings at the lowest cost to one or two at the highest. Going from a quarter to four times the reference cost of a reading lowers accuracy by 1.6 to 3.8 points on the seven main benchmarks.

Table 18: Cost of a reading. When one reading costs $m = 0.25$ to 4 times its reference cost, the continuous $k^\star$, the pair chosen by the rule and its accuracy (%) with the 7B reader at the reference budget.

	$m = 0.25$			$m = 0.5$			$m = 1$			$m = 2$			$m = 4$		
Benchmark	$k^\star$	(r, k)	Acc.	$k^\star$	(r, k)	Acc.	$k^\star$	(r, k)	Acc.	$k^\star$	(r, k)	Acc.	$k^\star$	(r, k)	Acc.
MathVista	0.74	(7,2)	66.15	1.04	(6,2)	65.96	1.47	(8,1)	65.68	2.08	(4,2)	65.19	2.94	(2,4)	64.46
MMMU	0.42	(7,2)	62.23	0.59	(8,1)	62.21	0.84	(7,1)	62.01	1.19	(3,3)	61.22	1.68	(2,3)	60.64
MathVerse	0.57	(7,2)	53.58	0.80	(6,2)	53.21	1.14	(6,1)	52.39	1.61	(3,2)	51.54	2.27	(2,1)	49.83
ChartQA	0.59	(7,2)	88.71	0.83	(8,1)	88.62	1.17	(4,2)	87.64	1.66	(2,4)	86.55	2.34	(1,8)	85.33
ScienceQA	0.76	(7,2)	85.33	1.08	(6,2)	84.93	1.52	(5,2)	84.44	2.15	(3,2)	83.13	3.04	(2,2)	82.08
AI2D	0.41	(8,1)	82.10	0.58	(7,1)	81.84	0.82	(5,1)	81.13	1.15	(2,3)	79.98	1.63	(1,6)	78.81
MMBench	0.70	(7,2)	81.70	0.99	(6,2)	81.32	1.41	(8,1)	81.08	1.99	(4,2)	80.31	2.81	(2,4)	79.26
SEED	210	(4,4)	74.17	297	(2,8)	74.17	421	(1,16)	74.18	595	(2,4)	74.17	841	(1,8)	74.17
RealWorldQA	165	(4,4)	59.45	234	(2,8)	59.45	330	(1,16)	59.45	467	(2,6)	59.45	661	(2,4)	59.45

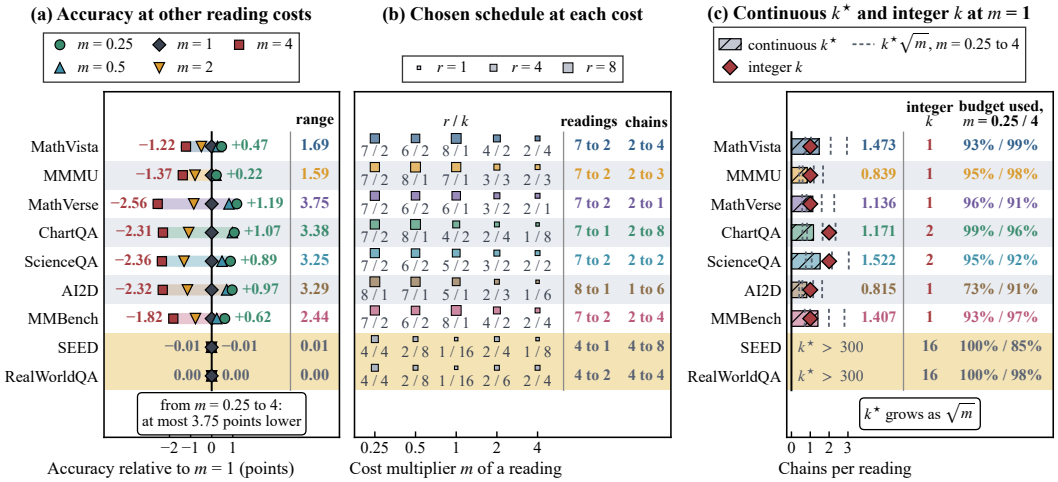

Figure 9: Cost of a reading (7B reader, reference budget). (a) Accuracy relative to the reference cost $m = 1$ when a reading costs m times its reference cost, with the range of the accuracy over the five costs. (b) Schedule (r, k) chosen at each cost; the square area grows with r, and the side columns give the readings and chains from $m = 0.25$ to 4. (c) Continuous $k^\star$ and integer k at $m = 1$, with dashes at $k^\star\sqrt{m}$ for the other costs; the side columns give the integer k and the share of the cap spent at $m = 0.25$ and 4.

Other readers, reasoners and aggregators. Table 19 and Fig. 10 change one component at a time. With nine readers from five families, the mean gain over the seven main benchmarks is 2.30 to 2.47 points, the gain in each cell is 1.81 to 3.13 points, and a reader's accuracy and its gain correlate at 0.06 over the 63 cells. On the two negative controls every reader stays within 0.06 points of zero. With four reasoners and five aggregators, the rule's mean accuracy on four benchmarks is 72.1% to 74.1%.

Table 19: Other readers, reasoners and aggregators (reference budget). Accuracy (%) of the variance rule on four benchmarks and their mean. The first row of each block is the reference configuration. SC: self-consistency.

Configuration	MathV.	MMMU	ChartQA	AI2D	Avg
Reader					
Qwen2.5-VL-7B	65.68	62.01	87.64	81.13	74.11
Qwen2.5-VL-72B	73.91	68.94	94.84	88.48	81.54
Qwen3-VL-8B	67.63	61.65	86.12	82.61	74.50
Qwen3-VL-32B	65.59	60.52	87.04	79.35	73.12
Qwen3-VL-30B-A3B	63.07	60.16	88.28	82.64	73.54
InternVL3-8B	65.90	64.45	84.87	82.29	74.38
InternVL3-78B	73.43	68.76	96.40	88.59	81.79
Gemma-3-27B	67.90	63.71	85.15	78.60	73.84
Llama-4-Scout	64.23	60.20	89.29	78.29	73.00

Configuration	MathV.	MMMU	ChartQA	AI2D	Avg
Reasoner					
Qwen2.5-7B	65.68	62.01	87.64	81.13	74.11
Qwen2.5-72B	64.31	61.56	88.87	78.74	73.37
Llama-3.3-70B	64.23	61.44	86.14	81.27	73.27
R1-Distill-Qwen-32B	64.44	60.36	84.94	78.50	72.06
Aggregator					
Majority vote	65.68	62.01	87.64	81.13	74.11
Confidence-weighted	63.34	61.65	86.30	82.90	73.55
Process reward	64.70	61.55	86.26	79.96	73.12
Outcome reward	64.11	59.25	89.54	82.45	73.84
Universal SC	64.38	63.80	86.61	81.10	73.97

Combination with other methods. Table 20 adds Reobserve to four base methods: self-consistency, a confidence-weighted vote, a process-reward vote and iterative perception, which re-reads regions of the image while reasoning. For each base method, Reobserve adds 2.26 to 3.13 points, more than the base method's own improvement of 0.91 to 1.19 points, and the two together add 3.42 to 4.50 points. The joint gain is within 0.25 points of the sum of the separate gains, so the two are close to additive.

Table 20: Reobserve combined with four base methods at the reference budget. Accuracy (%) of each base method alone, with its own improvement, with Reobserve and with both; the gains of the three variants over the base method; the sum of the two separate gains, $\Delta_{\text{own}} + \Delta_{\text{ours}}$; and the difference between the joint gain and that sum (points). Each base method is run in a separate experiment, so its accuracies are not cells of Table 2.

Base method	Alone	+Own	+Ours	Both	Δ_{own}	Δ_{ours}	Δ_{both}	Sum	Difference
Self-consistency	64.12	65.03	66.38	67.54	+0.91	+2.26	+3.42	+3.17	+0.25
Confidence-weighted vote	74.38	75.57	77.21	78.22	+1.19	+2.83	+3.84	+4.02	−0.18
Process-reward vote	73.06	73.97	75.48	76.55	+0.91	+2.42	+3.49	+3.33	+0.16
Iterative perception	69.59	70.78	72.72	74.09	+1.19	+3.13	+4.50	+4.32	+0.18

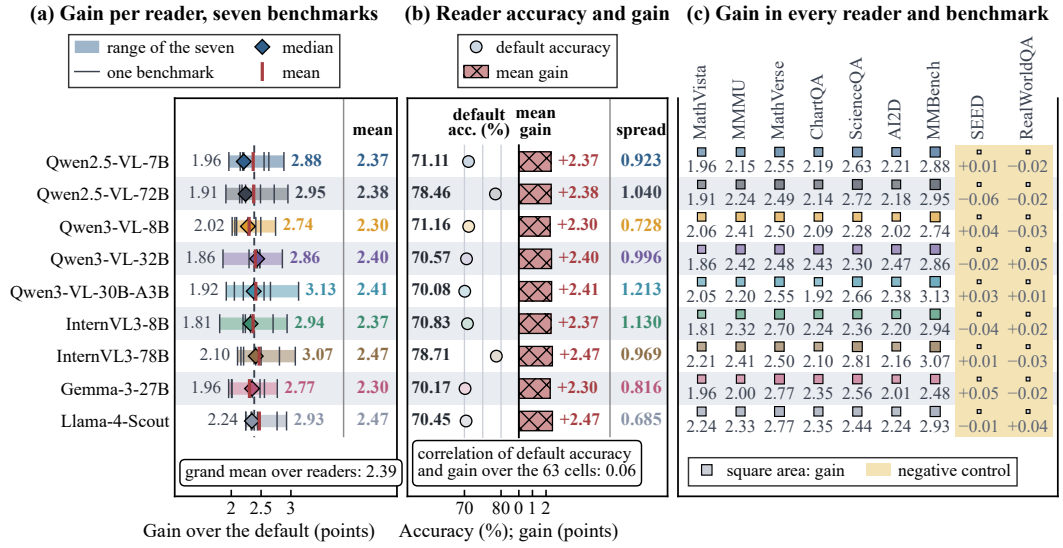

Figure 10: Nine readers (reference budget). (a) Range, median and mean of the gain of the variance rule over the default on the seven main benchmarks; the dashed line is the mean over readers, and the column on the right gives the mean of each reader. (b) Mean accuracy of the default with each reader over the seven main benchmarks and the mean gain of the rule, with the spread of the gain (largest minus smallest). (c) Gain in every reader and benchmark cell; SEED and RealWorldQA are the negative controls, drawn with a fixed square.

Reading noise. Table 21 and Fig. 11 change the between-reading variance directly, with the questions and their mean accuracy held fixed. Adding noise to the readings moves variance from $\hat{\sigma}_R^2$ to $\hat{\sigma}_V^2$ at a constant sum; over six noise steps on MathVista the continuous $k^\star$ falls from 3.85 to 0.88. When the noise is scaled from zero to its reference level, the schedule moves from $(1, 16)$ without noise to one or two chains per reading on all seven main benchmarks, and accuracy falls by up to 2.42 points.

Table 21: Reading noise at fixed mean accuracy on MathVista (7B reader). Noise added to the readings in six steps, from below to above the reference $\hat{\sigma}_V^2$, moves variance between the two components at a constant sum; the table gives the two components, the continuous $k^\star$ and the integer k chosen. Step 4 is closest to the reference.

Step	$\hat{\sigma}_V^2$	$\hat{\sigma}_R^2$	$k^\star$	k
1	0.0151	0.2008	3.85	3
2	0.0259	0.1900	2.86	3
3	0.0453	0.1706	2.05	3
4	0.0691	0.1468	1.54	1
5	0.0972	0.1187	1.17	1
6	0.1274	0.0885	0.88	1

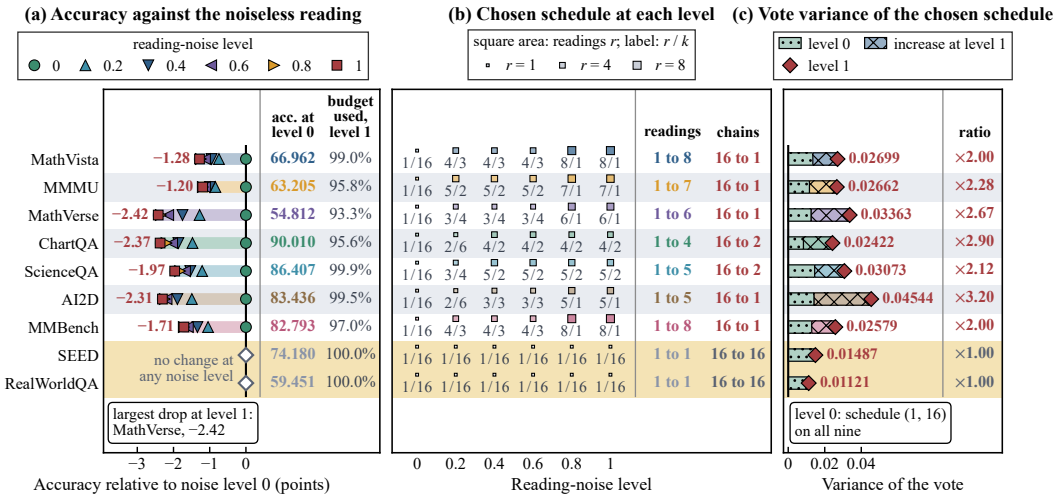

Figure 11: Reading noise at fixed mean accuracy (7B reader, reference budget). (a) Accuracy relative to the noiseless reading as the noise level grows from 0 to 1 (1 is the reference level), with the accuracy at level 0 and the share of the cap spent at level 1. (b) Schedule (r, k) chosen at each noise level, with the readings and chains from level 0 to 1. (c) Vote variance of the chosen schedule without noise and at the reference level, and their ratio.

G Additional Ablation Studies

Unless noted otherwise, the tables below use the 7B reader and the reference budget. Tables 22 and 24 give the change in accuracy of each variant against the full rule on each of the seven main benchmarks.

G.1 Fixed Number of Readings or Chains

Table 22 fixes r or k and lets the rule choose the other. Fixing one reading returns the default on every benchmark and loses the whole gain. Two readings recover half of it and four readings most of it. With k fixed, one or two chains per reading lose at most 0.59 points, while $k = 8$ forces a single reading and loses more than $k = 16$. At this budget the rule itself uses one or two chains per reading, so adding readings matters more than the exact k; k matters more when readings are expensive (Table 18).

Table 22: Fixed number of readings or chains (7B reader, reference budget). Change in accuracy (points) against the full rule when r or k is fixed and the rule chooses the other, on each of the seven main benchmarks and on average. n/a: no k fits the cap for that r; the mean is over the benchmarks where the setting fits.

Fixed	MathVista	MMMU	MathVerse	ChartQA	ScienceQA	AI2D	MMBench	Mean
$r = 1$	−1.96	−2.15	−2.55	−2.19	−2.63	−2.21	−2.88	−2.37
$r = 2$	−1.05	−1.24	−1.22	−0.94	−1.30	−1.05	−1.55	−1.19
$r = 4$	+0.04	−0.42	−0.22	0.00	−0.58	−0.43	−0.34	−0.28
$r = 8$	0.00	n/a	n/a	n/a	n/a	n/a	0.00	0.00
$k = 1$	0.00	0.00	0.00	−0.27	−0.19	0.00	0.00	−0.07
$k = 2$	−0.09	−0.23	−0.22	0.00	0.00	−0.59	−0.21	−0.19
$k = 4$	−0.50	−0.85	−0.43	−1.09	−0.56	−1.13	−0.81	−0.77
$k = 8$	−2.15	−2.27	−2.78	−2.31	−2.93	−2.34	−3.12	−2.56
$k = 16$	−1.96	−2.15	−2.55	−2.19	−2.63	−2.21	−2.88	−2.37

G.2 Calibration Settings and Cost

Table 23 varies the calibration and the number of deployment questions. From 64 to 512 questions the schedule does not change on any benchmark. The designs $(4, 4)$ and $(2, 4)$ cost 15% and 7.5% of $(8, 16)$ on the same questions and change one schedule each without changing the mean accuracy. Spread over 10^4 deployment questions, the 512-question calibration loses between 0.81 and 1.32 points per benchmark, and over 10^3 questions its charge exceeds the cap.

Table 23: Calibration size, nested design and amortisation (7B reader). Top: calibration cost in seconds and in calls, cost per deployment question over 10^5 questions, number of benchmarks whose schedule changed, and mean accuracy (%) of the rule over nine benchmarks. Bottom: change in accuracy (points) against the full rule when the 512-question calibration is spread over fewer deployment questions; n/a: the charge exceeds the cap.

Change	Setting	Seconds	Calls	Per question (s)	Changed	Mean acc.
Questions, $(8, 16)$	64	18,438	8,704	0.184	0	72.00
	128	36,876	17,408	0.369	0	72.00
	512	147,502	69,632	1.475	0	72.00
Design, $n_c = 128$	$(8, 16)$	36,876	17,408	0.369	0	72.00
	$(4, 4)$	5,511	2,560	0.055	1	72.00
	$(2, 4)$	2,756	1,280	0.028	1	72.00

Deployment questions	MathVista	MMMU	MathVerse	ChartQA	ScienceQA	AI2D	MMBench
1,000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
10,000	-0.84	-0.81	-0.97	-1.23	-1.32	-0.98	-1.16
100,000	0.00	0.00	0.00	0.00	0.00	0.00	0.00

G.3 Errors in the Inputs of the Rule

Table 24 feeds the rule wrong inputs and runs the schedule it then selects; when that schedule does not fit the cap, the rule keeps its k and uses the largest r that fits. Ignoring the grouping of chains by reading sets $\hat{\sigma}_V^2$ to zero and returns the default. A chain cost of zero makes the rule ask for sixteen chains per reading, and only one such reading fits, so it also returns the default. A reading cost of zero makes readings look free, and after fitting the cap the rule runs $(5, 2)$ or $(4, 2)$, which loses 0.13 points on average. Swapping the two components or inverting $k^\star$ changes the schedule on three and two benchmarks and loses 0.12 and 0.11 points on average, because the bracket of (3) is flat and the integer grid absorbs part of the error. Calibrating with one set of prompts and deploying with another keeps every schedule, so its loss of 0.72 points on average comes from the new prompts.

Table 24: Errors in the inputs of the rule, and selection by accuracy (7B reader, reference budget). Top: change in accuracy (points) against the full rule; the last row picks the pair with the best resampled accuracy on the calibration pool. Bottom: the schedule executed under each change of the inputs. When a changed input selects a schedule above the cap, the rule keeps its k and uses the largest number of readings r that still fits.

Change	MathVista	MMMU	MathVerse	ChartQA	ScienceQA	AI2D	MMBench	Mean
Components swapped	0.00	-0.23	0.00	0.00	-0.19	-0.43	0.00	-0.12
Inverse of $k^\star$	0.00	0.00	0.00	0.00	-0.19	-0.59	0.00	-0.11
Grouping ignored	-1.96	-2.15	-2.55	-2.19	-2.63	-2.21	-2.88	-2.37
c_V set to zero	-0.09	-0.23	-0.22	0.00	0.00	-0.14	-0.21	-0.13
c_R set to zero	-1.96	-2.15	-2.55	-2.19	-2.63	-2.21	-2.88	-2.37
Prompts changed after calibration	$+0.18$	-0.07	-1.37	-1.80	-0.04	-1.39	-0.55	-0.72
Selection by accuracy	-0.09	-0.17	-0.16	-0.11	-0.18	-0.37	-0.21	-0.18
Full rule	(8,1)	(7,1)	(6,1)	(4,2)	(5,2)	(5,1)	(8,1)	
Components swapped	(8,1)	(5,2)	(6,1)	(4,2)	(7,1)	(3,3)	(8,1)	
Inverse of $k^\star$	(8,1)	(7,1)	(6,1)	(4,2)	(7,1)	(3,2)	(8,1)	
Grouping ignored	(1,16)	(1,16)	(1,16)	(1,16)	(1,16)	(1,16)	(1,16)	
c_V set to zero	(5,2)	(5,2)	(4,2)	(4,2)	(5,2)	(4,2)	(5,2)	
c_R set to zero	(1,16)	(1,16)	(1,16)	(1,16)	(1,16)	(1,16)	(1,16)	
Prompts changed after calibration	(8,1)	(7,1)	(6,1)	(4,2)	(5,2)	(5,1)	(8,1)	

G.4 Selection by Accuracy

The last row of the top part of Table 24 compares the accuracy pick with the variance rule on the same calibration pool. Picking by resampled accuracy loses between 0.09 and 0.37 points on each of the seven main benchmarks. The variance model sums up the whole pool in two numbers, while the pick compares the resampled accuracies of all feasible pairs, each with its own sampling noise.

G.5 Reading Distributions and Reading Noise

Table 25 moves the reading distribution over nine settings of the three sources. On MathVista the gain of the rule increases with σ_V^2 across all nine settings, from 0.69 points at temperature 0.3, where $\sigma_V^2 = 0.019$, to 2.64 points at temperature 1.0, where $\sigma_V^2 = 0.111$. The chosen schedule moves from $(4, 3)$ when σ_V^2 is small to $(8, 1)$ when it is large. Table 26 scales the reading noise from zero

to its reference level at fixed mean accuracy. Up to 5% of the reference level the rule keeps the default and gains nothing; at 20%, 50% and 100% it gains 0.62, 1.47 and 2.37 points on average.

Table 25: Nine reading distributions (7B reader, reference budget). On MathVista: the between-reading variance, the schedule chosen and the gain over the default (points). Over the seven main benchmarks: the mean between-reading variance, the mean gain and its range. Prompt rewrite 1 is the reference reading distribution.

	MathVista			Seven benchmarks			
Setting	$\hat{\sigma}_V^2$	(r,k)	Gain	Mean $\hat{\sigma}_V^2$	Mean gain	Min	Max
Prompt, rewrite 1	0.0734	(8,1)	+1.96	0.0905	+2.37	1.96	2.88
Prompt, rewrite 2	0.0437	(4,3)	+1.38	0.0500	+1.54	1.21	1.94
Prompt, rewrite 3	0.0272	(4,3)	+0.98	0.0292	+0.99	0.67	1.38
Temperature 0.3	0.0186	(4,3)	+0.69	0.0305	+0.95	0.69	1.16
Temperature 0.7	0.0671	(8,1)	+1.83	0.0832	+2.18	1.74	2.71
Temperature 1.0	0.1106	(8,1)	+2.64	0.1327	+2.97	2.44	3.73
Crop, centre	0.0315	(4,3)	+1.07	0.0348	+1.12	0.73	1.49
Crop, overlapping	0.0528	(5,2)	+1.56	0.0587	+1.68	1.29	2.16
Crop, disjoint	0.0801	(8,1)	+2.17	0.0958	+2.51	2.06	3.09

Table 26: Reading noise at fixed mean accuracy. Gain of the rule over the default (points) on the seven main benchmarks when reading noise is scaled from 0 to its reference level, with the schedule chosen on MathVista.

Noise	(r,k), MathVista	MathVista	MMMU	MathVerse	ChartQA	ScienceQA	AI2D	MMBench	Mean
0	(1,16)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.05	(1,16)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.2	(4,3)	+0.52	+0.66	+0.81	+0.47	+0.56	+0.55	+0.77	+0.62
0.5	(4,3)	+1.23	+1.42	+1.64	+1.36	+1.59	+1.31	+1.77	+1.47
1	(8,1)	+1.96	+2.15	+2.55	+2.19	+2.63	+2.21	+2.88	+2.37

G.6 Questions with Mean Chain Accuracy below One Half

Table 27 computes the two-answer case of Theorem 1 exactly. Each chain is correct with probability μ_q independently, and a majority of k chains decides. Raising k from 9 to 49 lowers the variance of the vote from 0.027 to 0.005 in both cases. With $\mu_q = 0.57$ the vote is right 83.9% of the time instead of 66.8%, and with $\mu_q = 0.43$ it is right 16.1% of the time instead of 33.2%. The same reduction of variance moves accuracy by 17.1 points in opposite directions, as the two parts of Theorem 1 state.

Table 27: Mean chain accuracy below and above one half: exact binomial values for two answers and k independent chains of accuracy μ_q on one reading. Vote variance, accuracy (%) and its change from $k = 9$ to 49.

μ_q	k	Variance	Accuracy	Change
0.43	9	0.02723	33.22	
0.43	49	0.00500	16.11	−17.11
0.57	9	0.02723	66.78	
0.57	49	0.00500	83.89	+17.11

H Vote Variance after Reallocation

Table 28 lists the numbers behind Fig. 2a. At eighteen levels of the predicted variance, the empirical variance differs from the prediction by -3.92% to $+3.14\%$, with both signs along the range.

Table 29 gives the numbers behind Fig. 2b. At eight readings with 32 and 64 chains per reading, more than the $k_0 = 16$ of the calibration design, the variance predicted by (1) from the reference components of Table 1 matches the variance of newly sampled chains within 4×10^{-4} on every benchmark.

Table 28: Empirical and predicted vote variance after reallocation at eighteen levels of the predicted variance, with their relative difference (%), as in Fig. 2(a). All eighteen differences lie within 4% in size.

Level	Predicted	Empirical	Difference	Level	Predicted	Empirical	Difference
1	0.00209	0.00204	−2.39	10	0.01370	0.01348	−1.61
2	0.00337	0.00345	+2.37	11	0.01497	0.01459	−2.54
3	0.00468	0.00467	−0.21	12	0.01625	0.01607	−1.11
4	0.00595	0.00572	−3.87	13	0.01758	0.01689	−3.92
5	0.00725	0.00739	+1.93	14	0.01882	0.01897	+0.80
6	0.00854	0.00851	−0.35	15	0.02015	0.02068	+2.63
7	0.00981	0.01011	+3.06	16	0.02139	0.02184	+2.10
8	0.01109	0.01118	+0.81	17	0.02274	0.02292	+0.79
9	0.01241	0.01280	+3.14	18	0.02400	0.02450	+2.08

Table 29: Beyond the calibration design. Per benchmark, at eight readings with $k = 32$ and 64 chains per reading: the vote variance predicted by (1), the variance measured on newly sampled chains, and their difference, as in Fig. 2(b). All differences are below 4×10^{-4} in size and below 3% of the prediction.

Benchmark	k	Predicted	New chains	Difference
MathVista	32	0.00973	0.00962	−0.00011
	64	0.00945	0.00951	+0.00006
MMMU	32	0.01456	0.01474	+0.00018
	64	0.01442	0.01427	−0.00015
MathVerse	32	0.01213	0.01185	−0.00028
	64	0.01192	0.01188	−0.00004
ChartQA	32	0.00779	0.00785	+0.00006
	64	0.00764	0.00773	+0.00009
ScienceQA	32	0.01004	0.01012	+0.00008
	64	0.00973	0.00992	+0.00019
AI2D	32	0.01813	0.01850	+0.00037
	64	0.01796	0.01796	0.00000
MMBench	32	0.00981	0.00989	+0.00008
	64	0.00956	0.00956	0.00000
SEED	32	0.00093	0.00095	+0.00002
	64	0.00047	0.00046	−0.00001
RealWorldQA	32	0.00070	0.00069	−0.00001
	64	0.00035	0.00036	+0.00001

I Further Related Work

Sampling and stopping. Self-consistency (Wang et al., 2023) samples chain-of-thought traces (Wei et al., 2022) and returns the most frequent answer. Repeated sampling keeps raising the chance that at least one sample is correct over several orders of magnitude of samples (Brown et al., 2024), and the best use of a fixed test-time budget depends on the question, the model and the inference strategy (Snell et al., 2025; Wu et al., 2025). Adaptive-Consistency (Aggarwal et al., 2023) and early-stopping self-consistency (Li et al., 2024b) stop sampling once further samples are unlikely to change the vote. Chen et al. (2024a) show that the accuracy of a vote need not rise with the number of calls and can fall when a benchmark mixes easy and hard queries, which Theorem 1 describes for a single question through the sign of δ_q. In the two-step setting of this paper, all of these methods spend their budget on chains over one reading, that is, on the second term of (1).

Verifiers, reward models and confidence. Outcome verifiers score whole solutions and pick the best of n (Cobbe et al., 2021), process reward models score each step (Lightman et al., 2024), Math-Shepherd builds step labels without human annotation (Wang et al., 2024), and VisualPRM400K extends process supervision to multimodal reasoning (Wang et al., 2026). Confidence-weighted votes (Taubenfeld et al., 2025), universal self-consistency (Chen et al., 2023) and self-verification (Weng et al., 2023) change how sampled chains are weighted or selected. All of them act on chains already drawn, so when every chain inherits the same misreading they choose among answers that share it. The gain of Reobserve adds to that of the confidence-weighted and process-reward votes (Table 20).

Perception as a bottleneck. Several studies trace failures of multimodal models to perception: encoders confuse images that clearly differ (Tong et al., 2024), models miss simple geometric facts (Rahmanzadehgervi et al., 2024), answers often follow the question text and ignore the diagram (Zhang et al., 2024a), and models describe objects absent from the image (Li et al., 2023). Decoupled pipelines make the perception step explicit. Socratic Models compose vision and language models through language (Zeng et al., 2023), Prism separates perception from reasoning to assess each (Qiao et al., 2024), and visual programming turns a question into a program over vision modules (Gupta & Kembhavi, 2023). Our readings are this perception step, and Reobserve measures their variance.

Better single readings. Another line improves what one reading captures. Dynamic high-resolution tiling feeds image crops at native resolution (Liu et al., 2024a; Guo et al., 2024; Chen et al., 2024b), test-time augmentation aggregates predictions over transformed inputs (Shanmugam et al., 2021), and region re-reading lets the model look again at parts of the image during reasoning, through visual chain-of-thought (Shao et al., 2024), guided visual search (Wu & Xie, 2024), marks on image regions (Yang et al., 2023) or a value model that guides search (Wang et al., 2025). Multimodal reasoning models trained to write longer chains (Zhang et al., 2024b; Xu et al., 2025; Huang et al., 2026; Du et al., 2025) strengthen the second step instead. These methods lower the error of one reading or one chain, while Reobserve decides how many readings to draw, and its gain adds to that of iterative perception, which re-reads regions of the image while reasoning (Table 20).

Correlated votes and two-stage sampling. The Condorcet jury theorem gives the accuracy of a majority of independent voters who are each right with probability above one half, and its extensions show that positive correlation between voters lowers this accuracy (Boland, 1989; Ladha, 1992). In ensembles, the error of an averaged prediction falls with the disagreement of its members (Krogh & Vedelsby, 1995), and measures of that diversity relate to ensemble accuracy only in part (Kuncheva & Whitaker, 2003). Two-stage sampling gives the variance of a mean over clusters and their members, the design effect $1 + (k - 1)\rho$ and the cost-optimal number of members per cluster (Cochran, 1977; Kish, 1965), and the analysis of variance estimates the two components (Searle et al., 1992). A reading plays the role of a cluster and its chains the role of members; the accuracy bound, its ceiling at one reading and the efficiency of every schedule are what we add for votes.