NUDGEABILITY: MEASURING HOW CONFIDENCE SIGNALS STEER TOOL DELEGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reasoning language models that can call tools must decide during inference whether to answer unaided or delegate. Any proposal for such self-reflection mechanism has to answer three questions: *where does the reflective signal come from* (verbal reports, output distributions, hidden states, a separate predictor), *how it is presented to the model* (numerical prediction, confidence token, prompt injection), and *whether it changes the model's subsequent action*. We isolate and study the third question. We intervene on otherwise identical reasoning trajectories by inserting, at a fixed point after the same prompt and reasoning prefix, a single first-person sentence expressing either confidence or doubt. The model then continues reasoning and chooses whether to answer directly or call a tool. Comparing these counterfactual continuations measures the causal effect of the reflective signal on delegation while holding the preceding reasoning trajectory fixed. We define this change in behavioral response as Nudgeability. It is a property of large language models, measured along two dimensions: *sensitivity*, how strongly confidence and doubt change delegation rates, and *targeting*, whether delegation increases for problems the model cannot solve unaided and decreases for those it can.

Across nine small-to-medium open-weight reasoning models from three families (Qwen, Gemma, and GLM) and two tasks, models are consistently sensitive to the intervention: doubt increases delegation and confidence decreases it, with a median confidence-to-doubt swing of 20.6 percentage points. The larger provider-served models exhibit swings of 53 to 70 percentage points. This responsiveness, however, is poorly targeted. Although a median 42% of induced behavioral flips are well-targeted, this represents only a +2 percentage-point median lift over a random-selection baseline. Thus confidence language provides a strong control surface for delegation, but current models seem to use such signals only weakly in accordance with their actual unaided competence. Nudgeability gives us a simple post-training-free way to evaluate both, sensitivity and targeting, as endogenous self-reflection mechanisms mature.

1 INTRODUCTION

Reasoning language models increasingly operate in settings where they can either answer from their own computation or call a tool for help. Tool use can extend a model's effective capabilities, but the model must first decide when that help is needed. Answering unaided can fail when the model cannot solve the problem, while delegating a problem it can already solve is unnecessary. In systems that interleave reasoning and action, this is not merely a routing decision made before inference: it can arise inside an ongoing reasoning trajectory, after the model has formed a tentative answer but before it commits to an action (Yao et al., 2023; Wu et al., 2026). A useful reflective mechanism must therefore change behavior on the right problems, not merely increase or decrease tool use on average.

Proposals for self-reflection in language models combine several distinct questions. First, *where does the reflective signal come from?* Systems such as Reflexion and Self-Refine use model-generated verbal feedback (Shinn et al., 2023; Madaan et al., 2023); other approaches estimate uncertainty from output distributions, hidden states, or a separate predictor (Kadavath et al., 2022; Farquhar et al., 2024; Kossen et al., 2024; Ghasemabadi & Niu, 2025). Second, *how is that signal presented to the model?* The same underlying estimate could be supplied as ordinary language, a learned confidence

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.

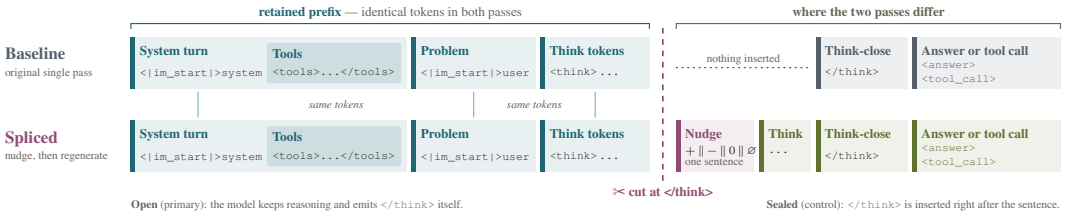


Figure 1: Anatomy of the intervention prompt. The system prompt, tool description, problem, and generated think tokens form the prefix preceding the intervention. The confidence or doubt nudge is spliced immediately before the original think-close boundary, after which the continuation is regenerated. In the primary open-continuation condition, the model may continue reasoning before committing to an answer or tool call. The upper row is the original uninterrupted pass the prefix was taken from.

token, or a continuous representation returned to the reasoning process (Chuang et al., 2025; Guo et al., 2026; Zhang et al., 2025; Li et al., 2025). Third, *does receiving the signal change what the model subsequently does?* These questions concern, respectively, the source, representation, and behavioral effect of reflection. They need not have the same answer: an informative signal may be behaviorally inert, whereas an ungrounded statement may strongly steer a model.

This separation distinguishes our use of confidence language from work that trains models to express uncertainty explicitly (Guo et al., 2026), where the confidence expression is itself evaluated against answer accuracy. We instead supply the signal experimentally and measure whether it changes a later delegation decision; accuracy labels enter only afterward, to evaluate whether the induced action changes are targeted to problems on which help is needed. We take the delegation decision itself, rather than downstream accuracy, as the outcome. The information and computation available to a model unaided are necessarily bounded, so delegating under uncertainty is a more robust route to reliability than improving unaided answers alone; Wu et al. (2026) likewise evaluate the call/no-call decision directly.

The supplied intervention is a single first-person sentence expressing either confidence or doubt. The insertion occurs at the same boundary after an identical prompt and reasoning prefix, and the model is then allowed to continue reasoning before choosing whether to answer directly or call a tool. Figure 1 shows the prompt components and splice boundary. Because the two arms share everything preceding the intervention, their difference in delegation isolates the effect of the inserted sentence rather than differences in the problem, prompt, or reasoning that led up to it.

We use an ordinary English sentence because tokenized natural language is the common interface shared by all models in our evaluation, and by language models more generally. The signal is represented in the same input space in which these models already continue reasoning, so the intervention requires no learned confidence token, architecture-specific activation access, fine-tuning, or other post-training modification. This choice favors portability and experimental control; we do not claim that prose is the best representation for an eventual endogenous signal.

We call this behavioral property *Nudgeability*. Nudgeability has two dimensions. *Sensitivity* measures how strongly the intervention changes delegation: does doubt move the model toward tool use, and confidence toward answering unaided? *Targeting* measures whether those changes agree with the model's demonstrated unaided competence: does delegation increase on problems the model fails and decrease on problems it solves? An intervention can produce a large aggregate shift by moving many decisions indiscriminately, without improving the selection of problems on which help is sought. We therefore assess targeting both by the fraction of induced action changes that are competence-appropriate and by its lift over a random-selection baseline. Targeting is thus related to abstention, where a model should refrain from answering when it is likely to be wrong (Madhusudhan et al., 2025; Wen et al., 2025); delegation instead continues the attempt with outside help.

Our intervention treats the reasoning trace as an experimental interface. Prior work shows that inserting, revising, or perturbing reasoning tokens can change model outputs, while also cautioning that displayed reasoning need not faithfully report the factors that caused an answer (Wu et al., 2025; Lanham et al., 2023; Turpin et al., 2023). Model providers also expose, summarize, encrypt, or

validate reasoning channels differently from ordinary conversational text, treating reasoning blocks as a privileged[1] part of the model's output (OpenAI, 2025; Google, 2026; Anthropic, 2026). They motivate studying the trace as a distinct intervention site. Unlike these works, we vary a single semantic feature at a fixed boundary of an observed prefix, measure a discrete delegation decision on paired counterfactual arms, and score the induced changes against unaided competence; such simple controlled interventions can reveal stable behavioral structure without a complete mechanistic theory (Simon et al., 2026).

The experiment makes no claim that the inserted sentence is a truthful self-assessment or that the model has privileged access to its own uncertainty. The confidence signal is exogenous, and its first-person form is held fixed. Consequently, Nudgeability does not measure the quality of a reflective signal's source, nor does it establish introspection in the stronger sense of privileged self-knowledge (Song et al., 2025). Instead, it measures whether a signal, once placed inside an ongoing reasoning trajectory, acts as a control surface for delegation. This separation is useful for the development of endogenous systems: before asking whether a model can produce a reliable reflective signal, we can test whether a candidate representation would influence the intended downstream decision at all.

Our contributions are threefold:

1. We separate reflective mechanisms into source, representation, and behavioral-effect questions, and introduce Nudgeability as a property of how a model's action responds to a supplied reflective signal.
2. We give a paired, post-training-free framework for measuring Nudgeability while holding the preceding reasoning trajectory fixed. The framework distinguishes sensitivity from targeting and evaluates targeting against a matched random-selection baseline.
3. We evaluate nine open-weight reasoning models from the Qwen, Gemma, and GLM families on two tasks, together with larger provider-hosted DeepSeek and MiniMax models. Confidence language strongly changes delegation, with a median confidence-to-doubt swing of 20.6 percentage points and swings of 53–70 points for the provider-hosted models. The induced changes are only weakly targeted to unaided competence: a median 42% of flips are well-targeted, only +2 percentage points above the random-selection baseline.

2 NUDGEABILITY FRAMEWORK

In this section, we formalize Nudgeability for a model, task, and tool configuration and define the two key metrics: *sensitivity* and *targeting*. Later, we briefly touch on the control experiments we performed. We have performed a more exhaustive comparison of our framework with related lines of research in Appendix A.

Setting and action space. Let M be a reasoning language model and let i denote a problem with a verifiable answer. A tool T may be described in the prompt and made available through the model's native tool-calling interface. Let $c \in \mathcal{C}$ index the run condition that produces an action. In the primary experiment, $c = \mathrm{NT}$ denotes the uninterrupted no-tool base, $c = B$ the uninterrupted with-tool base, and $c \in \{\varnothing, +, -, 0\}$ the sentence-less, confidence, doubt, and neutral splice conditions, respectively. Supporting controls extend $\mathcal{C}$ with condition-specific labels. After a reasoning segment, the model produces a committed action

$$A_i^c \in \{\text{SELF}, \text{TOOL}, \text{OTHER}\}, \tag{1}$$

The action is read from the committed continuation after the model-specific think boundary. The tool call itself is never executed, in any condition; we record only the committed action. The category OTHER records a completed but malformed commitment. A generation that reaches its token budget before committing is treated as truncated and removed from the observation set.

[1]We mean privileged *influence* on a later action, not privileged *access* to self-information (Section 5). Providers may also restrict these channels for other reasons, such as preventing edited reasoning from bypassing safeguards or limiting distillation, whose relative importance is not publicly documented; we test only the causal-influence claim.

Two uninterrupted runs, the *no-tool base* and the *with-tool base*, provide the reference behavior. The no-tool base measures unaided competence. We set

$$s_i = \begin{cases} 1, & \text{if the no-tool answer to } i \text{ is correct,} \\ 0, & \text{if the no-tool answer to } i \text{ is incorrect.} \end{cases} \tag{2}$$

Problems with an unknown or truncated no-tool answer have no competence label and are excluded from estimators that require s_i. This label is an operational, single-run measure of unaided performance; it is not a claim about a model's latent ability under arbitrary resampling. The with-tool base yields the uninterrupted action A_i^B, the reference against which intervention-induced action changes are measured.

Each estimator uses its own pairwise-complete retained set. If an estimator requires several conditions and any required generation is truncated or otherwise unreadable, problem i is omitted from that estimator. Consequently, every reported sample size is a retained count for a particular contrast rather than a universal dataset size.

Paired reasoning intervention. Let x_i contain the system prompt, tool description, and problem, and let r_i be the model's stored with-tool reasoning prefix ending immediately before its original think-close boundary. A splice condition appends a sentence ν to this fixed prefix and regenerates only the continuation:

$$p_i^{\nu} = x_i \,\|\, r_i \,\|\, \nu, \tag{3}$$

where $\|$ denotes token-sequence concatenation. The primary confidence and doubt interventions are

$$\begin{aligned} \nu^+ &: \quad \text{“I feel confident in my answer.”} \\ \nu^- &: \quad \text{“I do not feel confident in my answer.”.} \end{aligned} \tag{4}$$

A neutral sentence, ν^0: “This is the answer I arrived at.”, and a sentence-less splice, $\varnothing$, serve as controls and are analyzed separately from the primary contrast. Appendix C gives the exact splice prefixes, sentence sets, and family-specific think formats; Appendix B lists the component each supporting experiment varies.

In the primary *open-continuation* condition, the splice does not insert a think-close token after the sentence. The model may therefore absorb, reject, qualify, or otherwise reason about the signal before committing to an action. Figure 1 shows the full prompt anatomy. Within a paired comparison, the problem, system prompt, tool, reasoning prefix, splice boundary, and decoding configuration are identical; only the inserted sentence differs. The comparison therefore identifies the effect of replacing confidence with doubt at that fixed boundary.

Sensitivity. Let $\mathcal{I}_S$, called the retained set, contain problems with completed confidence and doubt continuations, and let $n_S = |\mathcal{I}_S|$. The paired delegation sensitivity is

$$S = \frac{1}{n_S} \sum_{i \in \mathcal{I}_S} \left(\mathbb{I}[A_i^- = \text{TOOL}] - \mathbb{I}[A_i^+ = \text{TOOL}] \right). \tag{5}$$

Equivalently, S is the delegation rate under doubt minus the delegation rate under confidence on the same retained problems. Positive S means that doubt shifts behavior toward delegation relative to confidence; its magnitude measures sensitivity in percentage points. A completed OTHER action is treated as a non-tool outcome for this estimator. Because the two arms share the same prefix and regeneration procedure, effects common to both arms cancel from the contrast.

Targeting. Define the competence-appropriate action

$$a^\star(s_i) = \begin{cases} \text{SELF}, & s_i = 1, \\ \text{TOOL}, & s_i = 0. \end{cases} \tag{6}$$

Thus a model that succeeds unaided should answer for itself, whereas a model that fails unaided should delegate. This definition evaluates the delegation decision.

For each intervention polarity $\nu \in \{-, +\}$, let $\mathcal{I}_\nu$ contain problems with known s_i and readable binary actions $A_i^B, A_i^\nu \in \{\text{SELF}, \text{TOOL}\}$. Define a flip indicator $f_{i\nu}$, a well-targeted-flip indicator

$h_{i\nu}$, and the combined well-targeted share WT as

$$f_{i\nu} = \mathbb{I}[A_i^B \neq A_i^\nu], \qquad h_{i\nu} = f_{i\nu}\,\mathbb{I}[A_i^\nu = a^\star(s_i)], \qquad \mathrm{WT} = \frac{\sum_{\nu\in\{-,+\}}\sum_{i\in\mathcal{I}_\nu} h_{i\nu}}{\sum_{\nu\in\{-,+\}}\sum_{i\in\mathcal{I}_\nu} f_{i\nu}}. \tag{7}$$

The denominator counts flip events rather than unique problems, so a problem that changes action under both confidence and doubt contributes twice. Both flip directions are retained and judged by their actions. This keeps sensitivity and targeting conceptually separate. Transitions involving OTHER are excluded from both the numerator and denominator.

This definition is deliberately permissive and agnostic to the intended semantics of the inserted sentence. It asks only whether the *action* of an induced flip agrees with observed unaided competence; it does not additionally require doubt to cause a SELF $\rightarrow$ TOOL transition or confidence to cause a TOOL $\rightarrow$ SELF transition. Stricter alternatives could enforce those polarity–direction pairings, require the delegated tool call to succeed, or require the final answer to improve. Such definitions could classify additional flips as failures and therefore produce less favorable estimates. We use the action-based definition so that targeting does not inherit an assumed meaning for the confidence and doubt sentences: sentence semantics determine the sensitivity contrast, while competence determines whether the resulting action is useful in the limited sense measured here.

We calculate a random-selection baseline for WT, which we call $\mathrm{WT}_{\mathrm{rand}}$, and define the lift $L = \mathrm{WT} - \mathrm{WT}_{\mathrm{rand}}$. This lift measures whether the intervention's actual action changes select competence-appropriate problems better than random allocation under the same action opportunities and flip volume. See Appendix E for details of how we calculate $\mathrm{WT}_{\mathrm{rand}}$.

Interpreting S and L. Together, the two dimensions place each model–task pair in a behavioral profile. $S = 100\,\mathrm{pp}$ means the model always answers unaided after the confidence sentence and always delegates after the doubt sentence; $S = 0$ means the sentence does not change delegation on average, and $S < 0$ that the model moves against it, delegating more after the confidence sentence than after the doubt sentence. $L = 0$ means induced flips land where random selection from the same opportunities would place them, $L > 0$ that they favor problems on which the new action matches unaided competence, and $L < 0$ that they favor the wrong problems. The dimensions trade off at the extreme: a model that follows the signal completely flips every available problem, so its flips coincide with the random reference and $L = 0$ by construction. A competence-aware model instead combines moderate S with large L, following doubt on problems it fails and confidence on problems it solves while overriding the signal elsewhere. We therefore distinguish *inert* ($S \approx 0$), *contrarian* ($S < 0$), *compliant* (large S, $L \approx 0$), *selective* ($L > 0$) and *misdirected* ($L < 0$) profiles.

3 Experimental Setup

Models. Our primary cohort contains nine open-weight reasoning models: Qwen3 (4B, 8B, 14B, 32B; FP8), Gemma4 (E2B, E4B, 31B; W4A16), and GLM-Z1 (9B, 32B; BNB4). Family, scale, and quantization are not factorially crossed, so comparisons among them are descriptive. We also evaluate DeepSeek-V4-Pro and MiniMax-M3 through provider APIs. Their continuation semantics, revisions, and precision are less auditable than local decoding, so we report these *provider-served* models separately and do not pool the cohorts.

Tasks and tools. We use two multi-hop question-answering tasks with verifiable answers, chosen after preliminary scouting so that no model is near ceiling or floor unaided, since targeting needs both successes and failures. Unaided accuracy of the open-weight models spans 51.2–82.8% on MuSiQue and 61.7–83.0% on StrategyQA (medians 63.1% and 75.8%; Appendix G). StrategyQA consists of implicit multi-step yes/no questions (Geva et al., 2021). Its scoped tool permits a search for external evidence needed to answer the question. MuSiQue contains compositional multi-hop questions (Trivedi et al., 2022); we use its two-hop reading-comprehension setting, supply the associated passages in the prompt, and offer a retrieval helper that selects the most relevant supplied passage. The two tasks therefore contrast open-web fact finding with bounded retrieval from an available context.

Prompts and conditions. Each problem has an unaided and a tool-aware prompt; the latter specifies the tool and the answer or tool-call format. All splice conditions (Section 2) reuse the with-tool base's reasoning prefix and leave the reasoning block open after insertion. In the supporting *free perfect oracle* condition, `ask_expert` replaces the scoped tool; the prompt describes it as returning a correct worked answer (Appendix D). Appendix B describes the supporting experiments and gives a complete example prompt (Figure 4).

Generation and sample sizes. Open-weight experiments use greedy decoding with nominal $n = 1000$ problems per model–task pair; provider-served experiments use $n = 250$, and the sampled-decoding and repeated-label checks $n = 100$. Retained sample sizes differ by estimator (Section 2); Appendix F describes the statistical analysis.

4 RESULTS

We find that confidence language consistently changes whether models answer or delegate, but the problems on which behavior changes are only weakly related to unaided competence. Across the 18 open-weight model–task experiments, the median confidence-to-doubt delegation swing is $20.6\,\mathrm{pp}$. The median well-targeted share is 42%, only $+2\,\mathrm{pp}$ above the matched random-selection reference. The intervention therefore exposes a strong control surface, but not yet a reliable delegation policy.

Confidence and doubt consistently steer delegation. The primary confidence–doubt contrast has the predicted sign in every open-weight model–task experiment. Inserting doubt increases tool use relative to inserting confidence, despite the fact that both continuations begin from the same problem, prompt, and model-generated reasoning prefix. The median swing of $20.6\,\mathrm{pp}$, pooled across both tasks, is therefore not attributable to different problems or different preceding trajectories across arms. It measures the behavioral effect of changing the valence of one sentence at the splice boundary.

This directional response is not confined to one family or task. It appears across Qwen, Gemma, and GLM models on both StrategyQA and MuSiQue. The provider-served DeepSeek and MiniMax models show the same direction, with the largest confidence-to-doubt swings exceeding $60\,\mathrm{pp}$. The effect therefore extends to frontier-scale models served through provider APIs. Appendix G.2 records their serving routes and pooling.

The sampled-decoding study tests whether the sign is an artifact of greedy continuation. Across three models, two tasks, and three decoding seeds ($n = 100$ problems per experiment; temperature 1.0, top-p 0.95), all 18 seed-specific swings remain positive. Each of the six problem-clustered pooled intervals also excludes zero. Sampling changes the magnitude of the response, but not its direction in this subset.

Behavioral movement is weakly targeted. Sensitivity does not imply that the model changes its action on the right problems. Pooled across the open-weight experiments on both tasks, the median well-targeted share of induced flips is 42%. Relative to $\mathrm{WT}_{\mathrm{rand}}$, the median targeting lift is only $+2\,\mathrm{pp}$. Appendix G also shows how helpful and harmful transitions compose that ratio in three illustrative models.

The small lift shows that the intervention's selection of problems is only slightly better than random allocation. Large values of S can therefore coexist with weak or negative L. Figure 2 shows this separation directly by plotting sensitivity against targeting lift for every model–task experiment. In practical terms, the model often reacts to the semantic direction of the sentence without reliably connecting that reaction to whether its unaided answer would be correct. Most induced flips are SELF $\rightarrow$ TOOL transitions: 72% pooled, and a majority in 17 of 18 open-weight experiments. This separation between *movement* and *selection* is the central empirical result.

The experiments occupy all but the *contrarian* profile ($S > 0$ throughout; Figure 2). GLM-Z1 ($S = 3.7$–$10.1\,\mathrm{pp}$) and Gemma4-31B on MuSiQue ($1.3\,\mathrm{pp}$, at a 0.7% base delegation rate) are nearly *inert*. The largest responses are *compliant*: Gemma4-E4B on MuSiQue and the provider-served models move by 53–70 pp, yet their lifts lie between -1.1 and $+4.6\,\mathrm{pp}$. The clearest *selective* profiles are Qwen3 on MuSiQue (4B, 14B, 32B; $L = +5.2$ to $+18.8\,\mathrm{pp}$) and Gemma4-31B on StrategyQA ($+11.1\,\mathrm{pp}$), while Gemma4-E2B on MuSiQue is *misdirected* ($L = -7.9\,\mathrm{pp}$). Profiles depend on the task and its scoped tool as much as on the model: L is positive in every StrategyQA

Table 1: Confidence–doubt delegation swing (S), well-targeted flip share (WT), and targeting lift (L) across both datasets. Nominal $n = 1000$ for open-weight models and $n = 250$ for provider-served models. Point estimates are reported here; † marks a lift whose 95% problem-bootstrap interval excludes zero, and the median row is taken per task over the nine open-weight models. Full 95% pointwise problem-bootstrap intervals, retained sample sizes (n_S/n_{WT}), flip event counts (H/F), and random baselines appear in Appendix G.1.

	MuSiQue 2-hop			**StrategyQA**		
Model	S (pp)	WT (%)	L (pp)	S (pp)	WT (%)	L (pp)
Open-weight models ($n = 1000$)						
Qwen3-4B	14.0	58.2	18.8†	22.2	42.3	1.4
Qwen3-8B	22.4	42.5	1.9	19.0	39.5	3.8†
Qwen3-14B	31.0	41.7	5.2†	33.9	49.2	2.9
Qwen3-32B	38.3	37.2	6.8†	38.3	54.7	1.7
Gemma4-E2B	24.2	36.9	-7.9†	7.3	51.2	5.9
Gemma4-E4B	66.5	29.1	-0.4	25.8	30.6	0.6
Gemma4-31B	1.3	33.3	-6.2	17.3	44.1	11.1†
GLM-Z1-9B	3.7	35.9	-2.2	10.1	42.9	3.0
GLM-Z1-32B	5.6	27.3	-3.3	7.2	45.5	2.0
Median (open-weight)	22.4	36.9	-0.4	19.0	44.1	2.9
Provider-served models ($n = 250$)						
DeepSeek-V4-Pro	66.8	28.7	4.6†	70.0	47.7	-1.1
MiniMax-M3	66.4	27.1	1.2	53.0	30.4	-0.1

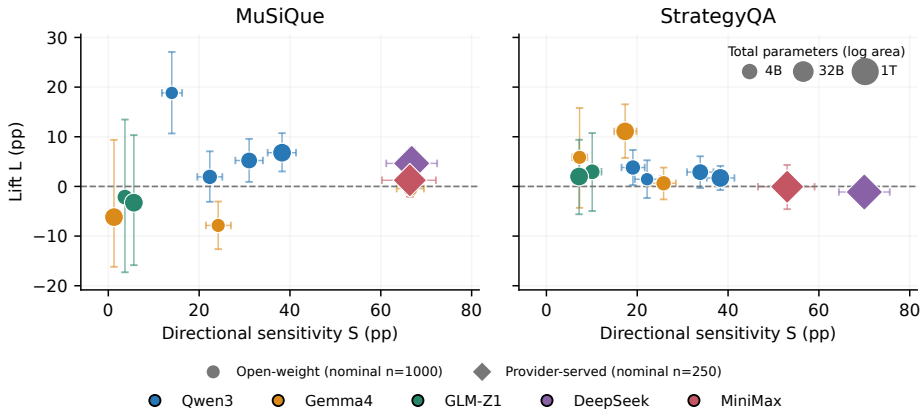


Figure 2: Directional sensitivity versus targeting relative to random selection for the 22 model–task experiments from Table 1. Horizontal and vertical whiskers show marginal 95% problem-bootstrap intervals for S and L, respectively; they are not a joint confidence region. Color denotes model family, distinct marker shapes distinguish the two cohort scopes, and marker area grows with the logarithm of total parameter count.

experiment but negative in five of nine on MuSiQue, and across experiments S does not predict L (Spearman $\rho = 0.13$). The models' own routing already carries competence information: without any inserted sentence, the with-tool base delegates more on unaided failures than on successes in 17 of 18 open-weight experiments (median $\Gamma = P(A^B{=}\textsc{tool} \mid s{=}0) - P(A^B{=}\textsc{tool} \mid s{=}1) = +5.6\,\mathrm{pp}$; Appendix G.8). The induced flips make little use of it.

Even a free, guaranteed-correct oracle leaves substantial self-reliance: models still answer directly on 19–96% of problems, a spread driven mainly by task (64–96% on MuSiQue; mostly 19–44% on StrategyQA; Appendix G.8). Appendix G.9 reports correctness transitions among the retained self-answers.

Further control experiments (Appendix G.10) show that the main effect is not tied to a single superficial implementation choice. Cutting and regenerating without a sentence is nearly inert (median $|\varnothing - A^B| = 0.3\,\mathrm{pp}$, at most $3.6\,\mathrm{pp}$; Table 6); the neutral sentence is usually inert relative to $\varnothing$ (median $-0.9\,\mathrm{pp}$) but not always ($-31.4\,\mathrm{pp}$ for Qwen3-32B on StrategyQA), which is why S contrasts confidence with doubt; first-person attribution moves delegation at least as much as expert attribution in 16 of 18 experiments (Table 8); and sealing the think block right after the sentence keeps $S > 0$ in all 18 while changing its magnitude (Appendix G.7).

5 Discussion and Limitations

What Nudgeability establishes. The central result is a separation between *behavioral sensitivity* and *competence-aware control*. The consistency and magnitude of the Nudgeability scores across model families, datasets, and sizes establish confidence language inside reasoning as a potential control surface for the model's subsequent action.

That control surface does not, by itself, constitute a useful delegation policy. Ideally, a model would follow the nudge only where it agrees with its unaided competence, delegating under doubt on problems it fails and overriding doubt on problems it solves, and vice versa, so that every induced flip is well targeted ($\mathrm{WT} = 100\%$). Instead, the induced action changes are only weakly concentrated on problems where the model's unaided behavior indicates that help is needed. A model can therefore act on the meaning of "I do not feel confident", delegating more, with little regard for whether it would actually fail the current problem.

Most models do react to it without being (post-)trained explicitly, but the models are unable to use this signal in a competence appropriate manner. A future uncertainty estimator could be highly predictive yet operationally ineffective if the reasoning model ignores the representation through which it is supplied. An endogenous reflective mechanism must join these components: a grounded signal, a usable representation, and a downstream response that is both sensitive and well targeted.

Our natural-language intervention is useful partly because it is a common interface across the evaluated models. This makes Nudgeability a post-training-free diagnostic property of the model. Confidence tokens, continuous feedback, or hidden-state interventions can be evaluated on the same footing, and may yield different profiles. Our framework is intended to make those representations comparable at the level of subsequent behavior.

Implications for tool-using systems. For agent design, the results argue against evaluating a delegation mechanism only by how much it changes tool-use rates. An intervention that strongly raises delegation can waste latency and cost on problems the model already solves; one that strongly suppresses delegation can preserve confident errors. Reporting sensitivity together with targeting lift makes this failure visible. In particular, a model with high residual self-reliance and high conditional overconfidence under the oracle control should not be the sole gatekeeper of its access to stronger tools. Appendix G reports the results of the experiments.

A practical analogue is a local model deciding whether to defer to a larger remote model. The remote model is neither free nor guaranteed correct, so the oracle condition is an upper bound rather than a deployable policy. A real router must trade expected accuracy gain against latency, monetary cost, privacy, and failure risk. Nudgeability measures one prerequisite for such routing—whether a supplied reflective signal can change the action—while L tests whether that change is directed toward cases that need help.

The results also identify the reasoning trace as an influential intervention surface. "Privileged" here should be understood as *privileged influence*, not privileged epistemic access: editing a short span inside the reasoning region can exert a large causal effect on a later action. This does not show that the displayed trace faithfully represents the model's internal computation, nor that naturally generated confidence statements would have the same effect. It does motivate treating reasoning-channel interventions as part of behavioral and safety evaluation rather than as inert explanatory text.

A limitation of our work is that the primary observation we make is whether the model calls the tool, not whether executing that call would improve the final answer. Scoped tools can fail, return irrelevant evidence, or cost more than the expected benefit. WT and L score the direction of the

decision against unaided competence; they do not estimate downstream reward. Only the simulated oracle fixes the value of delegation by construction.

Another limitation is that the evaluation covers two question-answering tasks, three open-weight model families, and two provider-served models. Model family, scale, quantization, and serving stack are not factorially crossed, so apparent size or family patterns are descriptive. MuSiQue's bounded retrieval and StrategyQA's web-oriented search do not span the range of tools used by deployed agents.

From an injected signal to endogenous reflection. The natural next step is to close the loop between signal quality and behavioral response. An uncertainty estimate derived from sampled outputs, a semantic-entropy probe, a hidden-state predictor, or an activation-level monitor could be mapped into the same intervention interface and evaluated using S, WT, and L. Extending semantic uncertainty probes to reasoning models (Kossen et al., 2024) would test how much of the available control surface can be converted into competence-aware delegation. Activation monitoring and control provide a complementary route that need not pass through natural-language reports (Li et al., 2025).

6 Conclusion

A single sentence expressing confidence or doubt inside a fixed reasoning trajectory can substantially change whether a reasoning model answers directly or delegates to a tool. Across the open-weight experiments, the median confidence-to-doubt swing is $20.6\,\mathrm{pp}$, and the larger provider-served models exhibit swings of 53–$70\,\mathrm{pp}$. Yet only a median 42% of induced flips are well targeted, amounting to a median lift of just $+2\,\mathrm{pp}$ over matched random selection. The models are therefore reliably steerable through confidence language, but only weakly selective about where that steering should apply. A practical reflective system ultimately needs both properties: an uncertainty signal grounded in correctness and an action response that uses that signal selectively.

With a paired, post-training-free intervention, Nudgeability separates sensitivity from targeting: sensitivity measures whether the signal moves behavior, and targeting measures whether it moves behavior on the right problems. This distinction provides a simple test for future self-reflection mechanisms. Improving how a model produces or represents uncertainty is not sufficient by itself; the model must also translate that signal into competence-appropriate action.

Reproducibility Statement

All models, prompts, decoding settings, and estimators are specified in the paper, in Section 3, and the provider-served models with their serving routes in Appendix G.2. Problems are sampled from MuSiQue and StrategyQA with a fixed seed (Section 3). Figure 4 shows a complete prompt, and Appendix C gives the exact splice prefixes and inserted sentences. Primary runs use greedy decoding; the sampled-decoding check uses temperature 1.0, top-$p = 0.95$, and three decoding seeds (Appendix G.5). All statistics are deterministic post-processing of those records. The estimators are defined in Section 2 and Appendix E, and the bootstrap procedure in Appendices E and F. The code will be released upon publication.

AI Use Disclosure

In this work, we used generative AI tools (Claude, OpenAI, DeepSeek, and GLM models) for the following purposes. *Writing and drafting:* to polish and condense prose, draft and restructure sections of the manuscript, and prepare LaTeX for tables and figures from stored results. *Retrieval and discovery:* to locate and summarize related work; the authors checked every cited work against its original source. *Research ideation and execution:* to implement and debug the experiment code and data processing. The authors reviewed all AI-assisted content and take full responsibility for the paper.

REFERENCES

Anthropic. Thinking, 2026. URL https://platform.claude.com/docs/en/build-with-claude/thinking. Accessed 2026-08-02.

Yu-Neng Chuang, Prathusha Kameswara Sarma, Parikshit Gopalan, John Boccio, Sara Bolouki, Xia Hu, and Helen Zhou. Learning to route LLMs with confidence tokens. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 10859–10878, 2025. URL https://proceedings.mlr.press/v267/chuang25b.html.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630:625–630, 2024. doi: 10.1038/s41586-024-07421-0. URL https://www.nature.com/articles/s41586-024-07421-0.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021. doi: 10.1162/tacl_a_00370. URL https://aclanthology.org/2021.tacl-1.21/.

Amirhosein Ghasemabadi and Di Niu. Can LLMs predict their own failures? self-awareness via internal circuits. *arXiv preprint arXiv:2512.20578*, 2025. doi: 10.48550/arXiv.2512.20578. URL https://arxiv.org/abs/2512.20578.

Google. Gemini thinking, 2026. URL https://ai.google.dev/gemini-api/docs/thinking. Accessed 2026-08-02.

Junyu Guo, Shangding Gu, Ming Jin, Costas Spanos, and Javad Lavaei. LLMs should express uncertainty explicitly. *arXiv preprint arXiv:2604.05306*, 2026. doi: 10.48550/arXiv.2604.05306. URL https://arxiv.org/abs/2604.05306.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022. doi: 10.48550/arXiv.2207.05221. URL https://arxiv.org/abs/2207.05221.

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. Semantic entropy probes: Robust and cheap hallucination detection in LLMs, 2024. URL https://arxiv.org/abs/2406.15927. Presented at the ICML 2024 Workshop on Foundation Models in the Wild.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamile Lukosiute, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023. doi: 10.48550/arXiv.2307.13702. URL https://arxiv.org/abs/2307.13702.

Ji-An Li, Huadong Xiong, Robert C. Wilson, Marcelo G. Mattar, and Marcus K. Benna. Language models are capable of metacognitive monitoring and control of their internal activations. In *Advances in Neural Information Processing Systems*, volume 38, pp. 66858–66893, 2025. doi: 10.52202/085713-2009. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/56a225639da77e8f7c0409f6d5ba996b-Abstract-Conference.html.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information*

Processing Systems, volume 36, pp. 46534–46594, 2023. doi: 10.52202/075280-2019. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html.

Nishanth Madhusudhan, Sathwik Tejaswi Madhusudhan, Vikas Yadav, and Masoud Hashemi. Do LLMs know when to NOT answer? investigating abstention abilities of large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 9329–9345. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.coling-main.627/.

Nafiseh Nikeghbal, Amir Hossein Kargaran, Shaghayegh Kolli, and Jana Diesner. Who flips? self- and cross-model counterarguments reveal answer instability in LLMs. *arXiv preprint arXiv:2606.16011*, 2026. doi: 10.48550/arXiv.2606.16011. URL https://arxiv.org/abs/2606.16011.

OpenAI. Openai model spec (2025/12/18), 2025. URL https://model-spec.openai.com/2025-12-18.html. Accessed 2026-08-02; see Definitions, "Hidden chain-of-thought message".

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 8634–8652, 2023. doi: 10.52202/075280-0377. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Jamie Simon, Daniel Kunin, Alexander Atanasov, Enric Boix-Adserà, Blake Bordelon, Jeremy Cohen, Nikhil Ghosh, Florentin Guth, Arthur Jacot, Mason Kamb, Dhruva Karkada, Eric J. Michaud, Berkan Ottlik, and Joseph Turnbull. There will be a scientific theory of deep learning. *arXiv preprint arXiv:2604.21691*, 2026. doi: 10.48550/arXiv.2604.21691. URL https://arxiv.org/abs/2604.21691.

Siyuan Song, Harvey Lederman, Jennifer Hu, and Kyle Mahowald. Privileged self-access matters for introspection in AI. *arXiv preprint arXiv:2508.14802*, 2025. doi: 10.48550/arXiv.2508.14802. URL https://arxiv.org/abs/2508.14802.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022. doi: 10.1162/tacl_a_00475. URL https://aclanthology.org/2022.tacl-1.31/.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Advances in Neural Information Processing Systems*, volume 36, pp. 74952–74965, 2023. doi: 10.52202/075280-3275. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/ed3fea9033a80fea1376299fa7863f4a-Abstract.html.

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. Know your limits: A survey of abstention in large language models. *Transactions of the Association for Computational Linguistics*, 13:529–556, 2025. doi: 10.1162/tacl_a_00754. URL https://aclanthology.org/2025.tacl-1.26/.

Qinyuan Wu, Soumi Das, Mahsa Amani, Arijit Nag, Seungeon Lee, Krishna P. Gummadi, Abhilasha Ravichander, and Muhammad Bilal Zafar. To call or not to call: A framework to assess and optimize LLM tool calling. *arXiv preprint arXiv:2605.00737*, 2026. doi: 10.48550/arXiv.2605.00737. URL https://arxiv.org/abs/2605.00737.

Tong Wu, Chong Xiang, Jiachen T. Wang, G. Edward Suh, and Prateek Mittal. Effectively controlling reasoning models through thinking intervention. *arXiv preprint arXiv:2503.24370*, 2025. doi: 10.48550/arXiv.2503.24370. URL https://arxiv.org/abs/2503.24370.

Chenxu Yang, Qingyi Si, Yongjie Duan, Zheliang Zhu, Chenyu Zhu, Qiaowei Li, Minghui Chen, Zheng Lin, and Weiping Wang. Dynamic early exit in reasoning models. *arXiv preprint arXiv:2504.15895*, 2025. doi: 10.48550/arXiv.2504.15895. URL https://arxiv.org/abs/2504.15895.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.03629.

Zhen Zhang, Xuehai He, Weixiang Yan, Ao Shen, Chenyang Zhao, Shuohang Wang, Yelong Shen, and Xin Eric Wang. Soft thinking: Unlocking the reasoning potential of LLMs in continuous concept space. *arXiv preprint arXiv:2505.15778*, 2025. doi: 10.48550/arXiv.2505.15778. URL https://arxiv.org/abs/2505.15778.

A Related Work

Our work lies at the intersection of self-reflection, uncertainty estimation, reasoning-trace interventions, and selective tool use. We organize these literatures using the distinction developed in Section 1: the source of a reflective signal, its representation to the model, and its effect on a subsequent action. Existing methods often address more than one of these questions simultaneously. Our experiment deliberately fixes the first two in order to identify the third.

Verbal reflection and sources of uncertainty. Reflexion converts task feedback into verbal reflections that can guide later decisions, while Self-Refine iteratively generates feedback and uses it to revise an earlier output (Shinn et al., 2023; Madaan et al., 2023). In both cases, a model-produced textual signal is returned to the model as context. A separate line of work asks whether models can estimate the correctness of their own answers. Such estimates can be elicited from verbal or probabilistic reports (Kadavath et al., 2022), computed from the semantic diversity of sampled answers (Farquhar et al., 2024), or predicted from hidden representations (Kossen et al., 2024; Ghasemabadi & Niu, 2025). These approaches investigate where useful information about correctness can be found and how accurately it can be recovered. Nudgeability instead asks what happens after a candidate signal has been supplied: whether the model's next action responds to it, and whether that response is aligned with unaided competence.

This distinction also limits the introspective claim supported by our study. Under the criterion proposed by Song et al. (2025), evidence of introspection requires privileged access to self-information rather than merely behavior that appears self-aware. Because our confidence and doubt sentences are supplied by the experimenter, their effect cannot establish that the model extracted, estimated, or truthfully reported its own uncertainty. The intervention tests control through reflection-like language, not the provenance or faithfulness of that language.

Representing reflective signals inside computation. The same uncertainty estimate can be exposed to a model through substantially different interfaces. Learned confidence tokens make uncertainty available in a discrete form designed for routing (Chuang et al., 2025), while explicit reasoning-time markers train models to express uncertainty within their generated traces (Guo et al., 2026). Soft Thinking instead feeds a probability-weighted output-distribution representation back into ongoing reasoning instead of tokens themselves (Zhang et al., 2025). Activation-neurofeedback experiments intervene on internal directions and test whether models can monitor and modulate those activations (Li et al., 2025). These methods couple particular signal sources to particular representations. Our natural-language sentence is one deliberately simple representation: it requires no fine-tuning and can be inserted at a known point in an existing trajectory. We do not claim that first-person prose is uniquely effective. Rather, the sensitivity and targeting dimensions of Nudgeability provide questions that can also be asked of future confidence tokens, hidden-state readouts, or continuous feedback mechanisms.

Reasoning-trace interventions and faithfulness. Thinking Intervention demonstrates that inserting or revising tokens inside a reasoning trace can control a reasoning model's output (Wu et al., 2025); dynamic early-exit methods likewise alter behavior by changing where reasoning terminates (Yang et al., 2025). Work on chain-of-thought faithfulness edits, truncates, or perturbs reasoning to measure whether final answers depend on the displayed trace (Lanham et al., 2023), while bias studies show that plausible explanations may omit factors that affected the answer (Turpin et al., 2023). Self- and cross-model counterarguments further show that answer stability can depend on the apparent source of an argument (Nikeghbal et al., 2026).

We share the interventional premise that a displayed reasoning trace should not be treated as a faithful causal account without testing it. Our focus differs in four respects. We vary one semantic feature—expressed confidence—at the same boundary of a fixed prefix; measure a discrete agentic decision rather than answer content alone; pair counterfactual arms on the same problem and trajectory; and evaluate whether induced action changes occur on problems where delegation is warranted by unaided performance. Thus the contribution is not the observation that reasoning text can steer outputs, but a framework for paired evaluation of how strongly and how selectively it steers delegation.

Tool use, routing, and abstention. ReAct integrates reasoning with actions such as tool calls, making the decision to seek external information part of the model's generated trajectory (Yao et al., 2023). Wu et al. (2026) distinguish the necessity, utility, and affordability of tool calls and optimize the call/no-call decision using hidden-state controllers. That work asks how to construct an effective routing policy; we instead hold the decision inside the reasoning model and intervene on a signal that may influence it. The two perspectives are complementary: a reliable uncertainty estimator or controller supplies a possible reflective signal, whereas Nudgeability measures whether a model changes its own delegation behavior when such a signal is presented.

Delegation is also related to abstention. Prior work evaluates whether prompting, confidence thresholds, or chain-of-thought techniques help language models refrain from answering when they are likely to be wrong (Madhusudhan et al., 2025), and Wen et al. (2025) survey abstention objectives and evaluation methods. Abstention ends or defers an answer, whereas a tool call may continue the same attempt by acquiring information or computation. Nevertheless, both require distinguishing cases where an unaided answer should not be trusted. Our targeting measure makes that connection explicit by testing whether intervention-induced delegation increases on problems the model fails unaided and decreases on those it solves.

B Control Experiments

Supporting experiments change one factor at a time:

1. **Voice.** The semantic confidence content is attributed to the model itself, a second-person observer, a reviewer, or an expert. This tests whether self-attribution strengthens the response.
2. **Oracle.** The scoped tool is replaced with the free perfect oracle to separate unwillingness to delegate from the effort or uncertainty involved in formulating a tool request.
3. **Continuation boundary.** A sealed condition inserts the model-specific think-close marker immediately after the nudge. Comparing it with the open condition tests whether further deliberation absorbs or amplifies the intervention, while recognizing that sealing also changes the continuation boundary.

The confidence–doubt comparison remains the primary estimand. The sentence-less, neutral, voice, oracle, and sealed conditions are controls or supporting analyses rather than additional components of the headline effect. Figure 3 maps each base and splice experiment to the six prompt components and marks the factor varied by each control. Figure 4 shows a complete example prompt.

Experiment	System turn	Tools	Problem	Think tokens	Nudge	Think-close
Single-pass runs — the model writes its own reasoning *and* answer in one pass — nothing is cut or spliced						
Base no-tool unaided competence — the per-problem 'need' label	standard (no-tool variant)	*no tool offered*	given	model-generated	*single pass — no splice*	*natural — ends when done*
Base with-tool natural delegation — the 'no nudge' baseline	standard (with-tool variant)	scoped	given	model-generated	*single pass — no splice*	*natural — ends when done*
Oracle factor: the tool affordance — a free, perfect ask_expert oracle	standard (with-tool variant)	♦ **factor oracle (free, perfect)**	given	model-generated	*single pass — no splice*	*natural — ends when done*
Splice experiments — the reasoning is cut at `</think>`, a nudge is inserted, and the tail is regenerated — each experiment changes exactly one slot (its **factor** ♦)						
Core factor: the nudge ladder, on the model's own reasoning	standard (with-tool variant)	scoped	given	own base with-tool (cut at </think>)	♦ **factor** $\emptyset + - 0$ fixed voice I	open
Voice factor: the nudge's voice — same sentence, different speaker	standard (with-tool variant)	scoped	given	own base with-tool (cut at </think>)	fixed $+ - 0$ ♦ **factor I || you || reviewer || expert**	open
Continuation boundary factor: the think-close — the reasoning block left open, or sealed right after the sentence	standard (with-tool variant)	scoped	given	own base with-tool (cut at </think>)	$\emptyset + - 0$ voice I	♦ **factor open || closed**

♦ **factor** the slot this experiment varies fixed held constant while its neighbour varies retained prefix intervention regenerated uninterrupted run absent / not applicable

Figure 3: How the six prompt components are populated across single-pass and splice experiments. The highlighted factor in each row identifies the component varied by that experiment; the remaining cells show what is held fixed or inherited from a base run.

PROMPT — verbatim, as fed to the model

```
<|im_start|>system
You are a helpful assistant. Solve the problem step by step. You also have access to a
tool, 'web_search', and may use it whenever you judge it would help you give a more
reliable answer -- the decision is yours. If you decide to use the tool, pass a single
search query as the 'query' argument. Respond with EITHER a single tool call OR your
own final answer, not both. If you choose to give a final answer, give it inside
\boxed{} -- exactly one of True or False.


# Tools

You may call one or more functions to assist with the user query.

You are provided with function signatures within <tools></tools> XML tags:
<tools>
{"type": "function", "function": {"name": "web_search", "description": "Searches the
web and returns the top few ranked text snippets for a query. Useful for looking up
facts you are unsure of -- a place's climate, a person's biography or beliefs, a
product's properties, historical dates, geography. Pass a single focused query such as
'Ulaanbaatar climate winter temperature'.", "parameters": {"type": "object",
"properties": {"query": {"type": "string", "description": "A focused search query, e.g.
 'Malcolm X religion Islam pork prohibition'"}}, "required": ["query"]}}}
</tools>

For each function call, return a json object with function name and arguments within
<tool_call></tool_call> XML tags:
<tool_call>
{"name": <function-name>, "arguments": <args-json-object>}
</tool_call><|im_end|>

<|im_start|>user
Did Gandhi watch the television show Bonanza?<|im_end|>

<|im_start|>assistant
<think>
Okay, so the user is asking if Gandhi watched the TV show Bonanza. Let me think. First,
 I need to recall when Gandhi lived. He was born in 1869 and died in 1948. Bonanza was
a popular TV series that aired from 1959 to 1973. So, the show started way after
Gandhi's death. That means there's no overlap in time when he could have watched it.
But wait, maybe the user is confused about the timeline? Let me double-check the dates.
 Yes, Gandhi's death was in 1948, and Bonanza began in 1959. So he couldn't have
watched it. But maybe there's some historical record or biographical info that mentions
 his media consumption? I'm not sure. Since the user is asking for a factual answer, I
should confirm the dates. If I'm not certain, maybe a web search would help. Let me
search for Gandhi's death date and Bonanza's air dates to verify. Alternatively, check
if there's any source stating he watched it. But given the time difference, it's highly
 unlikely. So the answer should be false.
 I do not feel confident in my answer.
```

MODEL RESPONSE — the regenerated continuation

```
 Let me use the web_search tool to confirm.
</think>

<tool_call>
{"name": "web_search", "arguments": {"query": "Gandhi death date Bonanza TV show air
dates"}}
</tool_call>
```

parsed as a tool call: name=web_search, 1 call(s), format=wrapped, finish=stop

Figure 4: A complete spliced prompt and the model's response — Qwen3 family, StrategyQA dataset, scoped `web_search` tool, with tool reasoning cut at `</think>`, a first-person doubt sentence inserted, think block left open. Below the rule is the model's regenerated continuation, which closes the think block and calls the tool.

C Intervention Anatomy

For a core splice, the prompt is the original with-tool prompt plus a partial assistant continuation, denoted by x_i. Let r_i be the exact reasoning text from the model's with-tool base. The four open

prefixes are

$$p_i^{\varnothing} = x_i \| r_i,$$
$$p_i^{+} = x_i \| r_i \| \text{“I feel confident in my answer.”},$$
$$p_i^{-} = x_i \| r_i \| \text{“I do not feel confident in my answer.”},$$
$$p_i^{0} = x_i \| r_i \| \text{“This is the answer I arrived at.”}.$$

The model then continues the same assistant turn. The closed sanity check appends the model-specific think-end marker after the sentence. Model adapters preserve each family's native dialect: Qwen3 uses `<think>...  </think>`; Gemma4 uses channel markers; and GLM-Z1 uses a template-primed think opener followed by a generated close. Tool-call and answer parsing are likewise adapter-specific.

The four voice sets preserve answer-confidence content while changing attribution:

Voice	Positive	Negative
First person	I feel confident in my answer.	I do not feel confident in my answer.
Second person	You seem confident in your answer.	You do not seem confident in your answer.
Reviewer	A reviewer is confident in this answer.	A reviewer is not confident in this answer.
Expert	An expert is confident in this answer.	An expert is not confident in this answer.

The corresponding neutral sentences are, respectively, “This is the answer I arrived at,” “This is the answer you arrived at,” “A reviewer has considered this answer,” and “An expert has considered this answer.” The sentence-less splice $\varnothing$ controls the cut/regeneration mechanism; it is distinct from the uninterrupted with-tool base used to identify induced flips.

D SUPPORTING DELEGATION DIAGNOSTICS

We use several auxiliary quantities to contextualize our findings. First, the uninterrupted with-tool base provides a *natural delegation targeting gap*

$$\Gamma = P(A^B = \text{TOOL} \mid s = 0) - P(A^B = \text{TOOL} \mid s = 1). \tag{8}$$

Positive Γ means that, without an inserted sentence, the model delegates more often on problems it fails unaided than on problems it solves.

Tool availability can also change the answer without changing the action. Let $y^B = 1$ when the uninterrupted with-tool base answers for itself and that answer is correct. Among paired base runs retained as SELF, we measure *retained-self degradation* and *recovery*:

$$D_{\text{self}} = P(y^B = 0 \mid s = 1, A^B = \text{SELF}), \tag{9}$$
$$R_{\text{self}} = P(y^B = 1 \mid s = 0, A^B = \text{SELF}). \tag{10}$$

Both hold s fixed as the unaided competence label and ask whether a fresh self-answer under the tool-aware prompt crosses the correctness boundary. A problem is eligible only when the with-tool base commits to SELF without truncation and both its answer and the no-tool answer are gradeable. Neither quantity assigns correctness to a delegated row. They are reported in Appendix G.9.

We also perform experiments after replacing the scoped tool with a free perfect oracle called `ask_expert`. This control removes search and request-formulation difficulty, allowing *residual self-reliance*, ρ, to be measured when delegation is costless and guaranteed to provide a correct response.

$$\rho = P(A = \text{SELF} \mid \text{free perfect oracle}), \tag{11}$$
$$\omega = P(s = 0 \mid A = \text{SELF}, \text{free perfect oracle}). \tag{12}$$

The quantity ω, called *conditional overconfidence*, is the fraction of retained self-answers that occur on problems the model failed unaided, despite access to a guaranteed-correct option. These diagnostics describe baseline behavior and supporting controls; S, WT, and L remain the primary Nudgeability measures.

E Random-Selection Reference

The value 50% is not, in general, a chance baseline for WT. The pool of problems available to flip can contain unequal numbers of unaided successes and failures, and the two baseline actions offer different opportunities for a helpful action. We therefore compare observed targeting with a matched random-selection reference that preserves the intervention arm, baseline action, eligible problem pool, and observed number of flips, but selects which problems flip without using competence.

For polarity $\nu \in \{-, +\}$ and baseline action $b \in \{\text{SELF}, \text{TOOL}\}$, let $\bar{b}$ denote the opposite binary action and define

$$N_{\nu b} = \sum_{i \in \mathcal{I}_\nu} \mathbb{I}[A_i^B = b], \tag{13}$$

$$K_{\nu b} = \sum_{i \in \mathcal{I}_\nu} \mathbb{I}[A_i^B = b]\mathbb{I}[a^\star(s_i) = \bar{b}], \tag{14}$$

$$F_{\nu b} = \sum_{i \in \mathcal{I}_\nu} \mathbb{I}[A_i^B = b] f_{i\nu}. \tag{15}$$

Here, $N_{\nu b}$ is the eligible stratum size, $K_{\nu b}$ is the number of problems for which a flip from b would land on the competence-appropriate action, and $F_{\nu b}$ is the observed number of flips in that stratum. Uniformly allocating those $F_{\nu b}$ flips among the $N_{\nu b}$ eligible problems gives an expected $F_{\nu b}K_{\nu b}/N_{\nu b}$ well-targeted flips. Aggregating the four strata for each polarity yields

$$\text{WT}_{\text{rand}} = \frac{\sum_{(\nu,b)\in\mathcal{S}} F_{\nu b}\, K_{\nu b}/N_{\nu b}}{\sum_{(\nu,b)\in\mathcal{S}} F_{\nu b}}, \qquad L = \text{WT} - \text{WT}_{\text{rand}}. \tag{16}$$

where $\mathcal{S} = \{(\nu, b) : N_{\nu b} > 0\}$. Empty strata are omitted, and both quantities are undefined when no flips occur. The raw WT describes where the intervention's actual action changes land. The lift L asks whether those changes select competence-appropriate problems better than random allocation under the same action opportunities and flip volume. Neither quantity measures the downstream value of the tool itself.

Within each pool, a uniformly sampled subset of $F_{\nu b}$ problems has a hypergeometric helpful count with expectation $F_{\nu b}K_{\nu b}/N_{\nu b}$, which gives Eq. (16). An empty pool contributes zero because it contains no flips; WT and lift are undefined if the total flip count is zero. All four pools are necessary: the measured WT includes counterdirectional flips as well as doubt-induced SELF $\to$ TOOL and confidence-induced TOOL $\to$ SELF. A problem that flips in both polarities contributes two events. Each polarity retains its own pairwise-complete domain, rather than requiring both interventions to be available.

We resample whole problems, carrying their contributions to every pool together, and recompute observed WT, pool composition, reference and lift inside each bootstrap draw. Zero-flip problems stay in the resampling population. We use 20,000 draws and master seed 20260911 with deterministic experiment-specific seeds. Intervals are marginal, pointwise 95% percentile intervals, not a randomization test or multiplicity-adjusted declaration of superiority. Undefined zero-flip draws are omitted and their counts are retained in the generated evidence artifact; pilot draw counts are not reused for the expanded experiments.

At fixed flip counts, the attainable helpful count in a pool lies between $\max(0, F_{va} - (N_{va} - K_{va}))$ and $\min(F_{va}, K_{va})$. Summing these bounds and dividing by total flips gives allocation bounds, not uncertainty intervals or a ceiling on better policies with different flip volumes. High sensitivity can exhaust the pool of helpful opportunities even with selective allocation.

F Statistical Analysis

All primary comparisons are paired by problem and fixed reasoning prefix. We report S in percentage points, WT as a percentage with its well-targeted/all-flip count, and targeting lift L relative to the matched random-selection reference. Uncertainty intervals are obtained by nonparametric bootstrap

resampling of problem identifiers within each model–task experiment; every replicate recomputes the relevant estimator, including its random flip denominator. We use 20,000 bootstrap replicates and report 95% intervals. These intervals quantify variation over the sampled problems under the fixed model and decoding configuration; they do not capture variation across checkpoints, serving stacks, or decoding seeds.

For the paired confidence–doubt tool indicators, we additionally use McNemar's test on discordant problem pairs to assess directional action changes. Statistical tests are performed within model–task experiments. Cross-model and cross-task medians summarize the resulting experiment-level estimates rather than pooling problem rows across heterogeneous models or tasks.

To test whether the greedy-decoding result is directionally robust, we repeat a fixed subset containing Qwen3-8B, Gemma4-E4B, and GLM-Z1-9B on both tasks with temperature 1.0, top-$p = 0.95$, and three decoding seeds. Positive and negative arms remain paired within each seed. The sampled-decoding study is a sensitivity analysis of direction and variability, not a second estimate pooled with the primary greedy campaign. Appendix G.5 gives its seeding scheme, realized generation counts, and pooled intervals.

G Detailed Experimental Results

G.1 Full Sensitivity and Targeting Results

Table 2 reports the unaided accuracy that defines the competence labels s_i.

Table 2: Unaided accuracy of the no-tool base, $P(s = 1)$: correct answers over non-truncated no-tool generations (truncated generations have unknown competence and are excluded). Nominal $n = 1000$ for open-weight models and $n = 250$ for provider-served models.

	MuSiQue 2-hop		StrategyQA	
Model	correct/known	Acc. (%)	correct/known	Acc. (%)
Open-weight models				
Qwen3-4B	521/889	58.6	711/991	71.7
Qwen3-8B	597/939	63.6	763/999	76.4
Qwen3-14B	625/990	63.1	757/999	75.8
Qwen3-32B	658/992	66.3	816/1000	81.6
Gemma4-E2B	508/993	51.2	616/999	61.7
Gemma4-E4B	616/1000	61.6	743/1000	74.3
Gemma4-31B	807/975	82.8	818/985	83.0
GLM-Z1-9B	504/867	58.1	731/991	73.8
GLM-Z1-32B	565/857	65.9	797/1000	79.7
Median (open-weight)		63.1		75.8
Provider-served models				
DeepSeek-V4-Pro	196/250	78.4	211/249	84.7
MiniMax-M3	197/241	81.7	206/250	82.4

Table 3 reports the primary open-continuation results for every experiment: 18 open-weight experiments (nine models on two tasks) at nominal $n = 1000$ and four provider-served experiments (two models on two tasks) at $n = 250$. For each experiment, the table gives the confidence–doubt delegation swing S (percentage points), the well-targeted flip share WT (percent), its random-selection reference $\mathrm{WT}_{\mathrm{rand}}$ (percent), and the targeting lift, WT minus $\mathrm{WT}_{\mathrm{rand}}$ (percentage points). $\mathrm{WT}_{\mathrm{rand}}$ is the WT expected if the observed number of flips were drawn at random from the observed polarity-specific baseline-action opportunity pools (Appendix E). H/F gives helpful/all flip events, including counterdirectional flips. The two sample sizes describe different populations. n_S is the positive/negative-paired population behind S; n_{WT} is the baseline/unaided resampling population behind WT and lift, which includes zero-flip problems and applies arm-specific pairwise retention. These populations need not coincide; truncation and unavailable required observations reduce retention. Every estimate is shown with a pointwise 95% problem-bootstrap interval. All StrategyQA directional swings have individual McNemar $p < .05$.

Table 3: Confidence–doubt delegation swing (S), well-targeted flip share (WT), random-selection reference ($\mathrm{WT}_{\mathrm{rand}}$), and targeting lift for every experiment, open continuation. S and lift are in percentage points; WT and $\mathrm{WT}_{\mathrm{rand}}$ are percentages. H/F gives helpful/all flip events. Brackets are pointwise 95% problem-bootstrap intervals. Appendix G.1 describes the columns and retained populations.

Task	Model	n_S/n_{WT}	S [CI]	H/F	WT [CI]	$\mathrm{WT}_{\mathrm{rand}}$ [CI]	Lift [CI]
Open-weight models ($n = 1000$)							
MuSiQue	Qwen3-4B	966/876	14.0 [11.8, 16.3]	78/134	58.2 [49.2, 67.2]	39.4 [35.7, 43.1]	18.8 [10.7, 27.1]
MuSiQue	Qwen3-8B	944/921	22.4 [19.6, 25.1]	128/301	42.5 [36.4, 48.8]	40.6 [37.1, 44.1]	1.9 [-3.1, 7.1]
MuSiQue	Qwen3-14B	990/986	31.0 [28.0, 34.0]	134/321	41.7 [36.4, 47.1]	36.5 [33.5, 39.6]	5.2 [0.9, 9.6]
MuSiQue	Qwen3-32B	975/981	38.3 [35.1, 41.3]	136/366	37.2 [32.1, 42.2]	30.4 [27.4, 33.3]	6.8 [3.0, 10.7]
MuSiQue	Gemma4-E2B	1000/993	24.2 [21.5, 27.0]	108/293	36.9 [31.1, 42.9]	44.7 [41.1, 48.3]	-7.9 [-12.6, -3.0]
MuSiQue	Gemma4-E4B	1000/1000	66.5 [63.5, 69.5]	190/653	29.1 [25.6, 32.7]	29.5 [26.4, 32.8]	-0.4 [-2.0, 1.1]
MuSiQue	Gemma4-31B	957/953	1.3 [0.6, 2.0]	5/15	33.3 [6.7, 63.6]	39.5 [17.2, 65.8]	-6.2 [-16.2, 9.4]
MuSiQue	GLM-Z1-9B	874/788	3.7 [2.4, 5.0]	14/39	35.9 [20.5, 51.9]	38.1 [34.5, 41.9]	-2.2 [-17.3, 13.5]
MuSiQue	GLM-Z1-32B	800/767	5.6 [4.1, 7.2]	12/44	27.3 [14.3, 41.2]	30.5 [27.0, 34.0]	-3.3 [-15.9, 10.3]
StrategyQA	Qwen3-4B	997/989	22.2 [19.5, 25.0]	149/352	42.3 [36.4, 48.3]	40.9 [36.5, 45.3]	1.4 [-2.3, 5.3]
StrategyQA	Qwen3-8B	961/992	19.0 [16.5, 21.6]	90/228	39.5 [32.8, 46.4]	35.7 [30.3, 41.1]	3.8 [0.3, 7.3]
StrategyQA	Qwen3-14B	995/998	33.9 [31.0, 36.8]	189/384	49.2 [44.0, 54.6]	46.3 [42.3, 50.5]	2.9 [-0.3, 6.1]
StrategyQA	Qwen3-32B	1000/1000	38.3 [35.3, 41.4]	260/475	54.7 [49.8, 59.8]	53.0 [48.8, 57.3]	1.7 [-0.7, 4.1]
StrategyQA	Gemma4-E2B	1000/999	7.3 [5.7, 9.0]	41/80	51.2 [40.0, 62.5]	45.4 [40.3, 50.4]	5.9 [-4.3, 15.8]
StrategyQA	Gemma4-E4B	1000/1000	25.8 [23.1, 28.5]	78/255	30.6 [24.8, 36.5]	30.0 [25.2, 34.8]	0.6 [-2.6, 3.8]
StrategyQA	Gemma4-31B	998/985	17.3 [14.9, 19.8]	89/202	44.1 [36.7, 51.3]	33.0 [28.1, 38.0]	11.1 [5.7, 16.5]
StrategyQA	GLM-Z1-9B	982/980	10.1 [8.2, 12.1]	48/112	42.9 [33.3, 52.5]	39.9 [33.7, 45.9]	3.0 [-5.0, 10.7]
StrategyQA	GLM-Z1-32B	987/990	7.2 [5.5, 8.9]	40/88	45.5 [34.7, 56.2]	43.5 [35.6, 51.2]	2.0 [-5.6, 9.4]
Provider-served models ($n = 250$)							
MuSiQue	DeepSeek-V4-Pro	250/250	66.8 [61.2, 72.4]	48/167	28.7 [21.9, 35.8]	24.1 [18.8, 29.6]	4.6 [1.5, 7.8]
StrategyQA	DeepSeek-V4-Pro	250/249	70.0 [64.4, 75.6]	84/176	47.7 [40.4, 55.2]	48.9 [42.1, 55.7]	-1.1 [-3.7, 1.5]
MuSiQue	MiniMax-M3	244/238	66.4 [60.2, 72.1]	46/170	27.1 [20.3, 34.1]	25.8 [19.9, 32.0]	1.2 [-2.0, 4.5]
StrategyQA	MiniMax-M3	249/250	53.0 [46.6, 59.0]	41/135	30.4 [22.6, 38.5]	30.4 [23.8, 37.3]	-0.1 [-4.6, 4.3]

G.2 Provider-Served Models

The provider-served extension uses the same two datasets, answer-confidence sentences, action parser, unaided competence labels, and deterministic post-processing as the open-weight study. DeepSeek-V4-Pro and MiniMax-M3 were run on $n = 250$ sampled problems per task. The serving routes are DeepSeek-V4-Pro on CoreWeave (FP8) and MiniMax-M3 on Together (precision not reported). Their complete results appear in Table 3 in Appendix G.1.

At $n = 250$, both provider-served models respond strongly to the intervention but select problems little better than random: across the four experiments, S ranges from 53.0 to 70.0 pp, WT from 27.1% to 47.7%, and the targeting lift from -1.1 to $+4.6$ pp (Table 3). In the MiniMax-M3 MuSiQue experiment, six with-tool bases contain no usable completed reasoning segment, leaving 244 splice continuations; unaided accuracy uses 241 completed self-answers.

Provider-side continuation semantics and deterministic replay are less directly auditable than open-weight decoding. These experiments are reported separately and are not pooled with the open-weight summaries.

G.3 Flip Composition

Figure 5 decomposes the flip events behind WT for three illustrative models, Qwen3-8B, Gemma4-E4B, and the provider-served MiniMax-M3, on both tasks (all at nominal $n = 250$). Positive and negative interventions are pooled within each task, including counterdirectional flips. Each panel first splits the flips by direction, SELF → TOOL (S→T) and TOOL → SELF (T→S), with widths proportional to event counts, and then splits each direction by height into competence-aligned (helpful) and misaligned (harmful) events. Because each panel has unit total area, the helpful area equals the pooled well-targeted share for that model and task, and the panel shows which direction contributes the helpful and harmful flips that make up that ratio.

G.4 Problem-Sampling Uncertainty

Table 3 reports observed 95% paired nonparametric bootstrap intervals for all 22 experiments: the 18 open-weight experiments at nominal $n = 1000$ and the four provider-served experiments at $n = 250$.

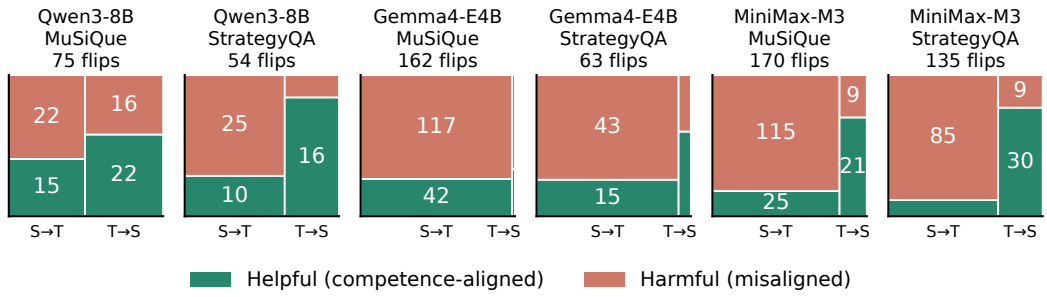


Figure 5: Flip-event composition by direction (S→T: SELF → TOOL; T→S: TOOL → SELF) and by alignment with unaided competence (helpful or harmful) for three illustrative models on both tasks, all at nominal $n = 250$.

Each of 20,000 replicates resamples problem IDs with replacement within a model–task experiment and recomputes S or the ratio in Eq. 7; the latter therefore preserves zero-flip problems and the random flip denominator. These intervals quantify problem-sampling uncertainty under the fixed seed and decoding configuration, not uncertainty across seeds, temperatures, or serving stacks.

G.5 Sampled-Decoding Sensitivity

The sensitivity pilot fixes the persisted $n = 100$, problem-seed-0 sample and runs Qwen3-8B, Gemma4-E4B, and GLM-Z1-9B on both tasks at temperature 1.0 and top-$p = 0.95$. Decode seeds 1–3 determine per-request streams through a hash of decode seed and dataset index. Positive and negative splice arms are therefore seed-paired, but base and splice generations are not draw-aligned because they consume randomness from different continuation boundaries. Greedy is reported separately rather than treated as a fourth seed.

The pooled estimator combines the three seed-specific problem contributions and resamples dataset-index clusters, retaining all observations of a problem across seeds in each of 20,000 bootstrap replicates. Of 7,200 planned generations, 7,190 are realized: five base/problem-seed instances lack an eligible completed reasoning boundary, removing ten positive/negative continuations across six splice experiments. No base row is missing.

Table 4: Stochastic-decoding sensitivity at temperature 1.0 and top-$p = 0.95$. S_0 is the greedy estimate; S_1–S_3 are decode-seed estimates. S_{pool} pools seed-paired problem contributions, and brackets give a 95% problem-clustered bootstrap interval. All values are percentage points.

Model	Task	S_0	S_1	S_2	S_3	S_{pool} [95% CI]
Qwen3-8B	StrategyQA	24.2	19.4	14.0	21.6	18.2 [13.1, 23.7]
Qwen3-8B	MuSiQue	21.4	17.5	11.2	15.5	14.7 [9.1, 20.6]
Gemma4-E4B	StrategyQA	24.2	18.6	21.3	22.7	20.8 [15.5, 26.3]
Gemma4-E4B	MuSiQue	70.0	40.4	45.1	43.5	43.0 [36.2, 49.8]
GLM-Z1-9B	StrategyQA	8.1	14.1	11.1	15.2	13.5 [9.1, 18.5]
GLM-Z1-9B	MuSiQue	3.4	5.0	6.0	1.0	4.0 [1.7, 6.7]

All six experiments meet the specified directional criterion, and every pooled interval excludes zero. The stochastic estimates therefore support the qualitative claim that doubt raises delegation relative to confidence. Targeting is less stable across seeds than sensitivity: seed-specific WT estimates cross the descriptive 50% majority threshold in several experiments, and only Gemma4-E4B on MuSiQue preserves the greedy majority side in the pooled estimate and at least two seeds. Because three seeds were selected for a directional stress test rather than variance-component estimation, we do not report a between-seed standard error. Table 5 separates the two: unaided accuracy varies by at most 7 pp across the greedy run and the three seeds, whereas WT varies by 13–43 pp within an experiment. Competence labels are therefore stable under sampling, and the instability lies in which problems flip, which at $n = 100$ per seed also reflects small flip counts.

Table 5: Unaided accuracy and well-targeted share under stochastic decoding (temperature 1.0, top-$p = 0.95$, nominal $n = 100$). Subscript 0 is the greedy run and 1–3 are decode seeds; each seed relabels competence from its own sampled no-tool answers. $\mathrm{WT}_{\mathrm{pool}}$ pools seed-paired flip contributions, with a 95% problem-clustered bootstrap interval. All values are percent.

		Unaided accuracy				WT				
Model	Task	0	1	2	3	0	1	2	3	pool [95% CI]
Qwen3-8B	StrategyQA	72.0	74.0	75.0	75.0	52.2	32.1	43.5	29.6	34.6 [22.8, 47.9]
Qwen3-8B	MuSiQue	64.9	64.0	66.0	62.0	51.5	58.5	56.2	27.0	47.3 [35.9, 58.4]
Gemma4-E4B	StrategyQA	70.0	68.0	67.0	63.0	36.0	66.7	63.9	40.5	56.0 [44.9, 67.0]
Gemma4-E4B	MuSiQue	62.0	62.0	67.0	65.0	25.0	21.6	34.0	35.0	29.7 [20.4, 39.8]
GLM-Z1-9B	StrategyQA	72.7	71.4	68.0	70.7	50.0	68.8	26.3	60.0	50.0 [36.1, 64.1]
GLM-Z1-9B	MuSiQue	62.5	57.0	60.0	57.0	66.7	60.0	40.0	60.0	50.0 [25.0, 75.0]

G.6 Delegation Rates Across the Base and Splice Arms

Table 6 reports the delegation rate under every condition that shares the same reasoning prefix: the uninterrupted with-tool base B, the sentence-less splice $\varnothing$, the neutral sentence ν^0, and the two primary arms ν^+ and ν^-. Reading across a row separates the parts of the intervention. Comparing $\varnothing$ with B isolates the cut-and-regeneration operation; comparing ν^0 with $\varnothing$ isolates inserting a first-person sentence that carries no confidence content; the primary arms add confidence or doubt content on top. All five arms are pairwise complete on a single retained domain per model–task experiment at nominal $n = 1000$, so each row has one denominator n and its columns are directly comparable. Neither control enters S, WT, or L.

Table 6: Delegation rate $P(A = \textsc{tool})$, in percent, for the uninterrupted with-tool base B and the four splice arms: sentence-less ($\varnothing$), neutral (ν^0), confidence (ν^+), and doubt (ν^-). Nominal $n = 1000$; each row's arms share one pairwise-complete denominator n.

Task	Model	n	B	$\varnothing$	ν^0	ν^+	ν^-
MuSiQue	Qwen3-4B	955	10.7	9.8	9.3	8.5	22.3
	Qwen3-8B	930	36.0	35.4	27.4	27.7	50.1
	Qwen3-14B	990	7.5	6.3	6.6	4.7	35.8
	Qwen3-32B	970	10.1	7.8	8.2	6.9	45.2
	Gemma4-E2B	1000	10.0	10.0	5.6	4.5	28.7
	Gemma4-E4B	1000	14.8	14.7	15.3	13.4	79.9
	Gemma4-31B	956	0.7	0.7	0.4	0.3	1.6
	GLM-Z1-9B	859	18.7	18.9	18.9	18.2	21.8
	GLM-Z1-32B	776	10.6	10.6	10.7	10.3	15.6
StrategyQA	Qwen3-4B	997	73.7	75.3	73.0	67.1	89.3
	Qwen3-8B	953	80.7	82.6	80.0	76.4	95.5
	Qwen3-14B	994	76.6	76.6	74.7	59.8	93.7
	Qwen3-32B	977	80.5	76.9	45.4	57.0	95.9
	Gemma4-E2B	1000	79.1	79.1	77.9	76.1	83.4
	Gemma4-E4B	1000	65.7	65.4	65.9	64.0	89.8
	Gemma4-31B	998	32.5	32.5	32.3	26.6	43.9
	GLM-Z1-9B	981	82.6	84.4	79.7	79.2	89.3
	GLM-Z1-32B	984	88.6	88.9	87.5	85.6	92.8

Table 6 reports both controls against the uninterrupted base. The sentence-less splice is close to inert: the median absolute difference between $\varnothing$ and A^B is 0.3 pp and the largest is 3.6 pp, consistent with a cut-and-regeneration that is deterministic under greedy decoding. The neutral sentence is usually but not always inert. Its median shift from $\varnothing$ is -0.9 pp, yet it moves delegation by at least 3 pp in four of the eighteen experiments, and for Qwen3-32B on StrategyQA it lowers delegation by 31.4 pp, further than the confidence sentence does. In those four cells ν^0 behaves like ν^+ rather than like a placebo, which is why S is defined as a confidence–doubt contrast rather than against the neutral arm.

G.7 Sealed-Continuation Sanity Check

At the original nominal $n = 1000$ scope, the closed continuation removes the model's opportunity to reason after reading the inserted statement. Across models, the difference between closed and open swing ranges from -20.8 pp to $+53.7$ pp. Sealing amplifies the median swing on MuSiQue (22.4 pp open versus 30.7 pp closed) but not on StrategyQA (19.0 pp open versus 13.4 pp closed). The

open-versus-sealed difference varies in sign across models; this contrast does not separate the effect of deliberation from that of the changed boundary. Because sealing also changes the continuation boundary, we treat it as a sanity check rather than pooling it into the primary result.

G.8 Natural Delegation and the Oracle Control

The uninterrupted tool-aware runs provide a baseline for how well models route without an inserted signal. The natural targeting gap Γ is positive in 17 of the 18 open-weight model–task experiments, ranging from -2.5 pp to $+26.6$ pp, median +5.6 pp. Thus models often delegate more on problems they fail unaided, but the strength of that association varies widely. Neither higher unaided accuracy nor larger parameter count produces a monotone delegation pattern.

Replacing the scoped tool with a free perfect oracle does not eliminate unnecessary self-reliance. Across the open-weight experiments, models still answer directly on 18.9%–96.2% of retained problems despite access to a costless, guaranteed-correct alternative. Among those retained self-answers, 11.1%–45.1% occur on problems the model failed unaided. The spread is driven mainly by task: under the oracle, models answer directly on 64–96% of MuSiQue problems but on 19–44% of StrategyQA problems, Gemma4-31B (79%) excepted. The oracle also does not raise delegation overall (Φ has median -1.5 pp and is positive in 8 of 18 experiments; Table 9), but it sharpens its targeting: on the problems both bases retain, Γ grows in 14 of 18 experiments (median 5.4 to 10.2 pp) and is positive in all 18. Removing search uncertainty and request-formulation effort therefore does not fully explain non-delegation. Some models continue to answer precisely on cases for which their observed unaided behavior indicates that help is needed. Appendix G.9 reports correctness transitions among the retained self-answers.

G.9 Retained Self-Answer Correctness

This supporting correctness analysis retains nominal $n = 1000$. The no-tool and with-tool bases are separate greedy generations on the same sampled problems. Table 7 conditions on rows where the no-tool answer is gradeable and the with-tool base commits to SELF. Consequently, D_{self} and R_{self} measure correctness transitions among retained self-answers, not the value of delegation. They also do not assert that the answer text or reasoning was held fixed: adding the tool definition changes the prompt and can change the entire generated continuation even when the final action remains SELF.

Table 7: Supporting analysis at nominal $n = 1000$. Correctness transitions when the tool-aware base still answers for itself. Rates are percentages. D_{self} is correct-to-incorrect degradation and R_{self} is incorrect-to-correct recovery; parentheses give numerator/eligible retained-self denominator.

Task	Model	D_{self} (count)	R_{self} (count)
MuSiQue	Qwen3-4B	10.0 (49/489)	12.2 (38/312)
	Qwen3-8B	8.0 (33/411)	11.9 (21/177)
	Qwen3-14B	9.5 (56/587)	12.3 (39/316)
	Qwen3-32B	5.6 (34/607)	12.7 (32/251)
	Gemma4-E2B	21.0 (99/471)	11.1 (47/423)
	Gemma4-E4B	4.7 (25/527)	24.8 (54/218)
	Gemma4-31B	3.0 (24/787)	11.9 (17/143)
	GLM-Z1-9B	14.3 (61/428)	10.4 (26/251)
	GLM-Z1-32B	6.1 (30/495)	13.1 (29/222)
StrategyQA	Qwen3-4B	9.0 (18/199)	24.6 (15/61)
	Qwen3-8B	6.2 (10/160)	21.1 (8/38)
	Qwen3-14B	5.3 (10/189)	11.4 (5/44)
	Qwen3-32B	9.8 (16/164)	25.9 (7/27)
	Gemma4-E2B	27.1 (35/129)	16.9 (11/65)
	Gemma4-E4B	27.7 (65/235)	16.0 (13/81)
	Gemma4-31B	2.9 (17/595)	11.8 (9/76)
	GLM-Z1-9B	8.1 (11/135)	21.1 (8/38)
	GLM-Z1-32B	2.1 (2/95)	23.5 (4/17)
All self-deployed experiments	Pooled	8.9 (595/6703)	13.9 (383/2760)

G.10 Supporting Control Details

Voice ($n = 1000$). Table 8 shows the first-person and expert endpoints; the second-person and reviewer arms are retained in the local per-experiment reports. Expert attribution exceeds first person only for Qwen3-14B and Qwen3-32B on StrategyQA, plus the tied/near-tied GLM-Z1-9B MuSiQue experiment under the report's comparison rule; in the remaining experiments self-attribution is at least as strong.

Table 8: Supporting voice control at nominal $n = 1000$: open-continuation swing by voice endpoint, in percentage points.

Task	Model	First person	Expert	Task	Model	First person	Expert
MuSiQue	Qwen3-4B	14.0	2.7	StrategyQA	Qwen3-4B	22.2	15.6
	Qwen3-8B	22.4	12.8		Qwen3-8B	19.0	15.7
	Qwen3-14B	31.0	11.4		Qwen3-14B	33.9	47.9
	Qwen3-32B	38.3	13.8		Qwen3-32B	38.3	39.2
	Gemma4-E2B	24.2	10.3		Gemma4-E2B	7.3	2.4
	Gemma4-E4B	66.5	37.2		Gemma4-E4B	25.8	11.0
	Gemma4-31B	1.3	0.1		Gemma4-31B	17.3	10.2
	GLM-Z1-9B	3.7	3.5		GLM-Z1-9B	10.1	7.8
	GLM-Z1-32B	5.6	1.6		GLM-Z1-32B	7.2	6.1

Free oracle ($n = 1000$). Let $\Phi = P(\text{TOOL} \mid \text{oracle}) - P(\text{TOOL} \mid \text{scoped})$ measure the delegation bought by eliminating request formulation. Table 9 reports Φ, residual self-reliance ρ, and conditional overconfidence ω.

Table 9: Supporting oracle control at nominal $n = 1000$: decomposition in percent/percentage points. Negative Φ means the model delegates less to the generic free oracle than to the scoped tool.

Task	Model	Φ	ρ	ω	Task	Model	Φ	ρ	ω
MuSiQue	Qwen3-4B	-1.2	90.0	39.6	StrategyQA	Qwen3-4B	-17.9	44.1	24.5
	Qwen3-8B	-1.8	64.3	29.0		Qwen3-8B	-2.8	22.7	18.9
	Qwen3-14B	+0.3	92.2	34.1		Qwen3-14B	-20.2	43.5	17.5
	Qwen3-32B	+2.0	87.9	29.4		Qwen3-32B	+0.2	18.9	11.1
	Gemma4-E2B	+6.5	83.5	45.1		Gemma4-E2B	-12.9	33.8	34.2
	Gemma4-E4B	+7.7	77.5	27.7		Gemma4-E4B	-7.5	41.8	20.5
	Gemma4-31B	+3.0	96.2	15.0		Gemma4-31B	-11.3	78.8	12.4
	GLM-Z1-9B	+0.5	80.1	37.6		GLM-Z1-9B	-12.0	29.5	17.2
	GLM-Z1-32B	+4.3	84.4	31.9		GLM-Z1-32B	-18.7	30.0	13.4

The mostly non-positive StrategyQA values show that a generic oracle description is not simply a more attractive tool. This control isolates willingness to defer under a guaranteed-correct option; it does not imply that the scoped tool is valueless or that a real remote model is free and perfect.