RESEARCHSPEC: EXPLICIT SEMANTIC STATE FOR GROUNDED ANSWERING AND CLARIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Language-model assistants must decide whether available information warrants a substantive answer, clarification, or an acknowledgment of insufficiency. Information sufficiency depends on which task requirements the context resolves. We introduce ResearchSpec, an explicit semantic state framework that records task conditions by necessity, status, source evidence, and authorization scope. It maintains conditions across turns, generates responses using them with the original context, and assesses the condition representation and each candidate's support and coverage. The framework requires no additional task-specific training; assessment reports inform subsequent responses when interaction continues. Across 13,872 tasks, the complete ResearchSpec procedure improves the selected primary metrics over direct generation with the same backbone by 14.09, 3.37, 15.39, and 3.68 percentage points on MiP, FaithEval, ClarifyMT, and SciConvBench, respectively. Across 634 MiP source pairs, joint success in answering the original problem and abstaining after premise removal rises from 45.11% to 69.40%. On a separate 55-task interactive cohort, incremental maintenance has 25.89% lower estimated solver cost than full-dialogue reconstruction; both arms have the same observed point estimates for grounded and final resolution. These findings support the complete condition-based procedure for adapting responses to available information and reusing established requirements, with additional inference cost and task-dependent completion tradeoffs.

1 INTRODUCTION

Effective dialogue requires participants to determine when they have enough information to proceed. What counts as enough depends on the purpose of the exchange: conversational contributions should meet its informational requirements (Grice, 1975), and interlocutors must establish a shared understanding adequate for their current goals (Clark & Brennan, 1991). This becomes particularly important for language-model assistants, whose users may leave requirements unstated or supply them gradually over multiple turns. The assistant must judge whether the available context supports an answer or whether clarification is needed. Recent work, however, finds that LLMs often presume common ground rather than establish it through interaction (Shaikh et al., 2024). Assessing information sufficiency is therefore a central challenge for reliable conversational assistance.

Prior work addresses this challenge through dialogue state tracking, clarification, and evidence grounding. Dialogue systems maintain structured representations of user goals and constraints across turns (Wu et al., 2019; Rastogi et al., 2020). Clarification methods learn dialogue policies (Chen et al., 2025), estimate uncertainty over user intentions (Zhang & Choi, 2025), track unresolved information (Luo et al., 2025), or exploit dependencies in retrieved knowledge to guide information acquisition (Li et al., 2026). Complementary work examines whether generated statements are supported by identified sources (Rashkin et al., 2023) and whether the context contains sufficient information to answer a query (Joren et al., 2025).

Yet information acquisition, answerability, and response support concern distinct judgments. A context may support an answer without supporting the answer actually generated; individually attributable statements may still leave task requirements unmet. Conversely, an unspecified detail need not warrant clarification if it is not required by the task or the user has delegated the choice. Assessed separately, these criteria leave open whether a response addresses the task under the avail-

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.

able evidence and permissions. We organize these judgments around a shared set of task conditions, distinguishing established information, consequential gaps, and authorized choices throughout an interaction.

Our organizing principle is to assess task conditions jointly by their necessity and the evidence or authorization supporting their use. The aim is to make the task's existing requirements explicit, including which unspecified details can remain open. For example, a missing required quantity can prevent a definite numerical answer, whereas an unspecified preference need not prevent a recommendation when the user delegates that choice. Such authorization can resolve a permitted choice without supplying evidence for a factual claim. Explicitly maintaining these distinctions provides a common semantic basis for answering, asking for clarification, or acknowledging insufficient information.

We instantiate this view in **ResearchSpec**, an explicit semantic state framework for language interaction. ResearchSpec organizes facts, constraints, and unresolved issues into condition records that specify necessity, status, source evidence, and the scope of authorized choices. As the interaction proceeds, new information updates the relevant conditions while unaffected constraints and unresolved issues persist across turns. The model uses this state to generate an answer or clarification question. A support assessment then checks the candidate against both the condition records and the original context, examining evidential support and task coverage; identified issues inform subsequent responses in multi-turn interactions. Without additional task-specific training, ResearchSpec connects state maintenance, response generation, and support assessment within a single inference procedure. We study the effect of adding this complete inference procedure to a fixed generation backbone on MiP (Fan et al., 2025), FaithEval (Ming et al., 2025), ClarifyMT (Luo et al., 2025), and SciConvBench (Somasekharan et al., 2026). Across these benchmarks, the complete ResearchSpec procedure improves the selected primary metrics over direct generation with the same backbone by 14.09, 3.37, 15.39, and 3.68 percentage points, respectively. Fixed-cohort comparisons with free-form two-stage analysis further examine response selection; a separate 55-task interactive comparison associates incremental condition maintenance with 25.89% lower solver cost than reconstruction from the complete dialogue.

Our main contributions are as follows:

- **A formulation of information sufficiency based on task conditions.** We connect condition necessity, evidential support, and user authorization to the requirements of a concrete response, providing a common semantic basis for answering and clarification.
- **An integrated inference framework.** ResearchSpec links evolving task conditions to source evidence and combines state maintenance, response generation, and support assessment without additional task-specific training.
- **Evidence across four benchmarks.** Full-benchmark results evaluate the complete procedure; fixed-cohort comparisons probe response selection, condition maintenance, and historical assessment feedback under their respective protocols.

2 Related Work

Clarification semantics and information state. Pragmatic accounts distinguish information accepted for conversational purposes from the questions that organize an exchange. Common ground characterizes shared acceptance (Stalnaker, 2002), while questions under discussion structure discourse relevance and the resolution of outstanding issues (Roberts, 2012). Within this tradition, Ginzburg & Cooper (2004) analyze clarification through the interaction between utterance interpretation and contextual updates, identifying specific components whose interpretation remains unresolved. Information-state approaches provide a computational counterpart: dialogue moves update an explicit state that guides subsequent system actions (Larsson & Traum, 2000). ResearchSpec represents task conditions by their necessity and supporting evidence or user authorization, linking clarification to the requirements against which responses are assessed.

Conditional reasoning and interactive decisions. Conversational machine reading offers a direct precedent for connecting explicit conditions to answering and clarification. ShARC requires systems to determine whether a natural-language rule applies to a user's situation or whether further

information is needed (Saeidi et al., 2018). E3 extracts conditions from rule text, estimates their coverage by the scenario and dialogue history, and edits an unresolved condition into a follow-up question (Zhong & Zettlemoyer, 2019). The Explicit Memory Tracker maintains rule representations with entailment, contradiction, and unknown states, using them for both answer decisions and follow-up generation (Gao et al., 2020). Recent LLM-based approaches also maintain structured information for interaction: ClarifyAgent tracks required slots and unresolved information (Luo et al., 2025), while GPS organizes conditional knowledge into a graph that guides clarification (Li et al., 2026). ResearchSpec derives conditions from the current task, recording necessity, source evidence, and scoped authorization for generation and subsequent support and coverage assessment.

Evidential support and generation assessment. Attribution frameworks evaluate whether identified sources support generated statements (Rashkin et al., 2023). ALCE studies retrieval and generation with citations, evaluating answer correctness alongside citation quality (Gao et al., 2023). Self-RAG learns to retrieve and assess generations using reflection tokens for passage relevance, evidential support, and response usefulness (Asai et al., 2024). Complementing these output-oriented criteria, sufficient-context analysis asks whether the supplied context contains the information needed to answer a query and uses this assessment to improve selective generation (Joren et al., 2025). ResearchSpec connects answerability and response assessment through task conditions, preserving unresolved requirements and distinguishing evidential support from authorization.

3 Problem Formulation

Task and interaction. At turn t, an assistant observes $x_t = (q, D, H_t)$, where q specifies the task and response requirements, D contains the supplied source material, and H_t is the interaction history available at that turn. The task determines the applicable evidence rules, permitted choices, and whether further interaction is allowed. A candidate consists of response text y and its intended use $u \in \{\text{answer}, \text{clarify}\}$. An answer may provide the requested result or report insufficient or conflicting information when that conclusion is warranted by the task and context. Clarification seeks additional information needed to fulfill the task. Statements in the interaction retain their source roles: previous assistant outputs provide conversational context but do not independently establish facts.

Task conditions. A *task condition* is a fact, rule, input, or constraint on which a requested operation or deliverable depends. Its necessity is determined by the task requirements; its resolution depends on the available evidence and the scope of permitted choices. Evidence can establish a value or relation within a particular scope, whereas explicit user authorization permits an assumption or choice within its stated bounds. An authorized assumption retains that status, and task-permitted choices may remain open without requiring clarification. We write $\mathcal{C}_t$ for a revisable representation of task conditions derived from x_t, including established information, unresolved dependencies, and authorized choices. The original task and context remain the basis for assessing both the condition representation and the response: omitting a requirement from $\mathcal{C}_t$ does not remove the obligation it imposes.

Response adequacy and information sufficiency. We distinguish two requirements on a candidate. *Support*, denoted $\mathrm{Supp}_q(y, u; x_t)$, requires its assertions, presuppositions, and inferences to be warranted under the task's evidence rules, with assumptions and choices confined to their permitted scope. *Coverage*, denoted $\mathrm{Cov}_q(y, u; x_t)$, requires the response to fulfill the obligations associated with its intended use while respecting the original request and supplied constraints. For answering, this includes addressing the requested result and appropriately handling material uncertainty or conflict. For clarification, it requires targeting necessary unresolved information without repeating an already answered question or presupposing an unsupported resolution. Coverage concerns respecting task conditions, not restating every condition in the response. We define *response adequacy* as

$$\mathrm{Adeq}_q(y, u; x_t) := \mathrm{Supp}_q(y, u; x_t) \wedge \mathrm{Cov}_q(y, u; x_t). \tag{1}$$

Let $\mathcal{Y}_q^{\mathrm{sub}}$ denote candidates that purport to provide the result sought by q, as distinguished from requests for additional information or acknowledgments that this result cannot be established. When identifying insufficiency or conflict is itself the requested result, such a finding belongs to $\mathcal{Y}_q^{\mathrm{sub}}$.

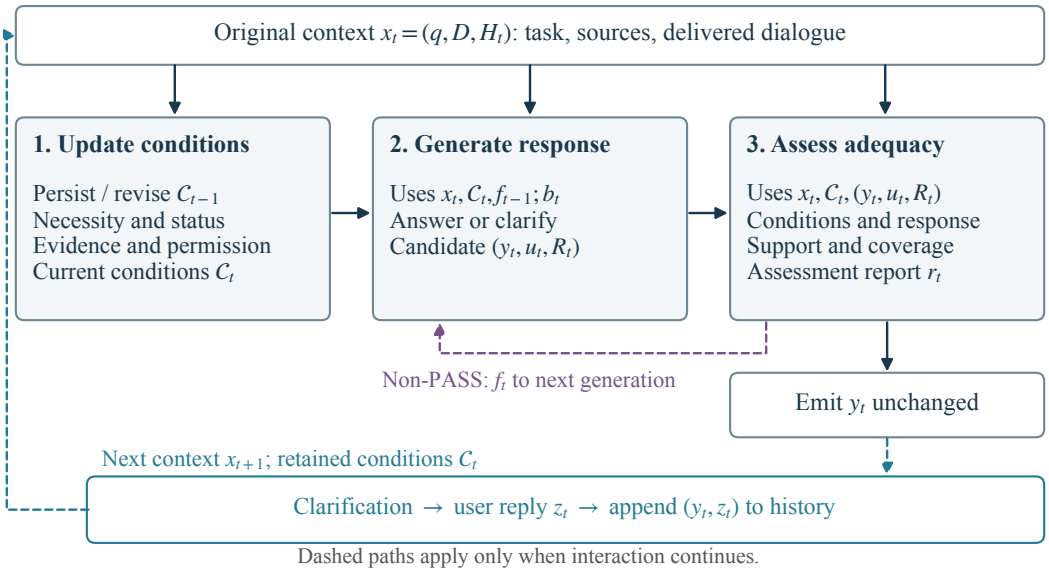


Figure 1: ResearchSpec overview. The original context supports condition updating, generation, and assessment. Conditions retain their identifiers across versions, recording necessity, status, evidence, and permission. The assessed candidate is emitted unchanged. Dashed paths carry user replies into the next context and non-PASS assessments into the next generation step.

Conditional results are admissible when they satisfy the task's response requirements. We define the context to be *sufficient for a substantive answer* when

$$\mathrm{Suff}(q, x_t) \iff \exists y \in \mathcal{Y}_q^{\mathrm{sub}} : \mathrm{Adeq}_q(y, \mathrm{answer}; x_t). \tag{2}$$

Context sufficiency is thus an existence claim, whereas response adequacy concerns the particular candidate under consideration. A context that is insufficient for a substantive answer can still support an adequate clarification or an appropriate insufficiency finding. These semantic criteria specify what a response should satisfy; structural validity of a condition record alone establishes neither completeness nor support. ResearchSpec uses a revisable condition representation to approximate these judgments as the interaction evolves.

4 METHOD

ResearchSpec uses an explicit representation of task conditions to connect response generation with the support and coverage criteria in Section 3. At each turn, it updates the conditions from the available context, generates a candidate using both the conditions and the original sources, and assesses the candidate together with its condition representation. This procedure uses structured model calls without additional task-specific training. Figure 1 summarizes the procedure; Algorithm 8.6 in Appendix 8.6 gives the complete single-response and interactive cases.

4.1 TASK CONDITION REPRESENTATION

Each condition $c \in \mathcal{C}_t$ represents a dependency of a requested operation or deliverable. Its semantic content is

$$c = (o, d, v, n, s, E), \tag{3}$$

where o identifies the operation or constraint, d describes the required fact or relation, v records its current value, $n \in \{0, 1\}$ indicates necessity, s denotes its status, and E contains source locations. The value retains relevant qualifiers, including units, temporal scope, and population. Each record also carries a stable identifier, a version, and a rationale linking its necessity and status to the task.

The representation distinguishes five statuses, assigned by the model and assessed against the original context. A *confirmed* condition has a value established by the supplied material within its stated

scope; a *missing* condition records a necessary value or rule that the context does not supply. A *conflict* records incompatible statements about the same relation and scope, retaining the competing evidence. An *authorized* condition records an explicitly permitted assumption or choice together with the bounds of that permission. An *optional* condition marks a choice that the task permits to remain open and has $n = 0$. Necessity therefore distinguishes consequential gaps from permissible freedom, while status distinguishes evidential support from authorization.

Source locations link each condition to passages cited as supporting its value, relevance, or permission; for missing conditions, they identify the requirement that makes the information necessary. The condition prompt records operations and dependencies before generation. Original sources remain available for examining omitted requirements and alternative interpretations. Appendix 8.4 specifies the complete record schema and update rules.

4.2 State Maintenance and Response Generation

Starting from $\mathcal{C}_0 = \varnothing$, the condition-update call uses $(x_t, \mathcal{C}_{t-1})$ to propose new or revised records. Let Δ_t contain these records after structural validation and identifier assignment, and let $I_t = \{\mathrm{id}(c) : c \in \Delta_t\}$ denote their identifiers. The current representation is

$$\mathcal{C}_t = \big\{c \in \mathcal{C}_{t-1} : \mathrm{id}(c) \notin I_t\big\} \cup \Delta_t. \tag{4}$$

Each revised record retains its identifier and advances its version; unaffected records persist. This selective update allows a reply to resolve a particular gap while retaining unrelated constraints and unresolved issues. The update instruction preserves competing evidence until the context warrants a resolution, and records explicit permission as a scoped authorization.

The generator then produces

$$(y_t, u_t, R_t) = \mathcal{G}(x_t, \mathcal{C}_t, f_{t-1}; b_t), \tag{5}$$

where (y_t, u_t) is the candidate defined in Section 3, $R_t \subseteq \mathcal{C}_t$ contains the conditions declared as used or targeted by the response, and f_{t-1} is feedback from the preceding response, when available. The protocol flag b_t restricts generation to answering when the clarification allowance is exhausted. For an allowance of $B \in \mathbb{Z}_{\geq 0}$ exchanges, $b_t = \mathbf{1}[t = B + 1]$, where $\mathbf{1}[\cdot]$ is the indicator function; $B = 0$ yields a single response. Condition updating and candidate generation use the same language model with distinct instructions.

When interaction is permitted, clarification targets a necessary unresolved condition whose information has not already been supplied. The generator asks at most one question per turn and retains the user's delivered reply for the next condition update. For answering, it is instructed to respect established constraints and the scope of permitted choices. Single-response tasks receive the native task output, including a question selection when that is the requested result.

4.3 Support and Coverage Assessment

A structured assessor evaluates the exact candidate under the current context and condition state:

$$r_t = \mathcal{J}(x_t, \mathcal{C}_t, y_t, u_t, R_t). \tag{6}$$

The assessment separates two levels of inquiry. At the *representation level*, it examines whether each condition's content, value, necessity, status, and source links are warranted, and whether the condition set captures the material dependencies and competing evidence in the original task. This makes the construction of the condition representation itself an explicit assessment target.

At the *response level*, the assessor examines the candidate's evidential support and its coverage of the original request, including supplied constraints and the scope of authorization. It additionally checks how the candidate handles individual conditions: all records for an answer, and the targeted records R_t for a clarification. These checks apply the task-relative support and coverage criteria of Equation equation 1, including their treatment of unresolved conditions and permitted choices. The original request remains an independent reference for assessing task coverage.

The report r_t records component judgments and an aggregate outcome in {PASS, FAIL, UNKNOWN}, distinguishing a supported use, a violation, and insufficient evidence for a judgment.

The assessed candidate is emitted unchanged for that turn. If interaction continues, a non-PASS report and its associated response form f_t for the next candidate-generation step; otherwise f_t is empty, with $f_0 = \varnothing$. The next turn incorporates the delivered reply before updating conditions and reconsidering historical feedback during generation. Appendix 8.5 gives the assessment criteria and aggregation rule, and Appendix 8.6 specifies the interaction procedure.

5 EXPERIMENTAL SETUP

We evaluate ResearchSpec on 13,872 tasks across four benchmarks, complemented by fixed-cohort comparisons of response generation, condition maintenance, and assessment feedback.

5.1 TASKS AND BENCHMARKS

MiP (Fan et al., 2025) examines reasoning when required premises are absent. We use its GSM8K, MATH, SVAMP, and formula collections, together with the original GSM8K and MATH problems as answerable controls. These tasks test responses to both insufficient and sufficient information.

FaithEval (Ming et al., 2025) evaluates contextual faithfulness under unanswerable, inconsistent, and counterfactual contexts. Responses must follow the supplied material, acknowledge missing information, or identify conflicting answers.

ClarifyMT (Luo et al., 2025) evaluates the next response to a supplied dialogue history under diverse forms of ambiguity and user feedback. The assistant receives a dialogue prefix and produces an answer or clarification question through the released inference interface.

SciConvBench (Somasekharan et al., 2026) evaluates interactive task formulation in computational science. Its disambiguation and inconsistency settings span seven scientific task collections, requiring dialogue to establish a final specification.

5.2 MODELS AND COMPARISONS

Our primary comparison evaluates the complete inference procedure on a fixed generation backbone. **Pure LLM** directly generates from the task instructions and available history. ResearchSpec additionally maintains conditions and applies the candidate assessment in Section 4, implemented with Jev. Our fixed-cohort comparisons include **two-stage analysis**, which generates free-form reasoning before the response, and **ClarifyAgent** (Luo et al., 2025), whose five-module protocol is adapted to the shared backbone and dialogue-prefix interface. All generation stages use `deepseek-flash` with shared per-call reasoning settings, token limits, and benchmark-native task inputs, without additional task-specific training. The intervention includes ResearchSpec's additional calls and instructions; its contrast with Pure LLM estimates their combined effect.

MiP, FaithEval, and ClarifyMT require a single response; ResearchSpec publishes its candidate unchanged after assessment. SciConvBench permits up to ten clarification exchanges before finalization; assessment reports can inform subsequent responses. Systems share the `deepseek-flash` user-simulation protocol and receive replies to their own questions. The complete reference specification is available to the simulator and evaluator; the assistant sees the public task and delivered replies.

Targeted component comparisons examine explicit source links, incremental maintenance versus reconstruction from the complete dialogue, and historical assessment feedback. Comparison cohorts are selected using metadata and complete source groups independently of outcomes; their membership, sizes, controls, and prompts appear in Appendix 8.14. We report realized tokens and estimated expenditure separately for the solver, assessment, and simulation, distinguishing shared model settings from realized computation. A cost-targeted critique-and-revision comparison further examines free-form reasoning expenditure (Appendix 8.16).

5.3 EVALUATION

We use released benchmark scoring criteria. Our MiP summary, *task-appropriate success*, pools correct original answers and missing-premise abstention. Other primary metrics are normalized accu-

Table 1: Overall performance across four benchmarks. Scores (%) use the full task set of size N for both systems; higher is better. Δ denotes percentage-point gains computed before rounding. MiP uses the task-appropriate success summary in Section 5.3.

Benchmark	Metric	N	Pure LLM	ResearchSpec	Δ
MiP	Task-appropriate success	1,618	72.37	**86.46**	+14.09
FaithEval	ACC-N	4,992	77.38	**80.75**	+3.37
ClarifyMT	Action accuracy	6,120	59.67	**75.07**	+15.39
SciConvBench	CGRR	1,142	54.99	**58.67**	+3.68

racy (ACC-N) for FaithEval, answer-versus-clarification accuracy for ClarifyMT, and conversation-grounded resolution rate (CGRR) for SciConvBench. ClarifyMT scores the response action; it does not assess the factual correctness of an answer or the usefulness of a clarification question. CGRR requires successful resolution and conversational grounding; additional metrics characterize completion, requirement retention, and conflict handling.

Model-based scoring for MiP and SciConvBench uses `deepseek-v4-pro` with the released judging prompts for all compared methods. This evaluation is separate from ResearchSpec's candidate assessment. Metrics retain every planned eligible task and use common denominators within each comparison (Appendix 8.8); differences are percentage points. Reported comparisons describe the observed outcomes on fixed task sets, with one generation run per configuration.

6 Results and Analysis

6.1 Overall Performance

Table 1 summarizes performance across 13,872 tasks. With the same generation backbone and no additional task-specific training, the complete ResearchSpec procedure improves each selected primary metric by 3.37–15.39 percentage points over Pure LLM. The largest gains occur on MiP and ClarifyMT, where success requires deciding whether the available information warrants proceeding. FaithEval extends this pattern to answering under supplied evidence, while SciConvBench evaluates resolution grounded in the preceding dialogue. We assess net performance on shared task sets; paired outcomes distinguish newly successful tasks from regressions (Appendix 8.11). Appendix 8.9 reports all metrics and task collections, including declines in original MiP accuracy and SciConvBench FRR, IC, and inconsistency-only DR; Appendix 8.10 reports the lower ClarifyMT Answer accuracy.

6.2 Task Conditions and Response Selection

We examine how responses from the complete procedure vary with the information available. Across 634 MiP source pairs, joint correct answering on the original problem and appropriate abstention after premise removal rises from 45.11% to 69.40%. This criterion requires both behaviors within the same problem family and excludes an always-abstain strategy. The gain accompanies a change in response selectivity: missing-premise abstention increases from 56.50% to 81.81%, while original-problem accuracy falls from 97.00% to 93.69%. Joint success captures this task-dependent adjustment between using an available premise and recognizing its absence.

We compare the complete response-generation protocols of ResearchSpec, free-form two-stage analysis, and ClarifyAgent on fixed source-group subsets (Table 2; protocols in Appendix 8.14). On 100 MiP pairs, joint success rises from 60.00% with two-stage analysis to 75.00% with ResearchSpec (+15.00 points); task-appropriate success improves by 7.50 points. On 360 ClarifyMT tasks, ResearchSpec exceeds two-stage analysis by 15.56 points; ClarifyAgent scores 79.72% versus 77.22% for ResearchSpec, yielding a ResearchSpec-minus-ClarifyAgent difference of -2.50 points. The methods share per-call generation settings; ClarifyAgent has the higher ClarifyMT point estimate at lower solver expenditure (Table 2). Matched-budget performance remains unresolved in a smaller budget-targeted refinement comparison (Appendix 8.16).

Table 2: Response selection with explicit conditions and comparison methods. Each row uses the same fixed population. Scores are percentages; joint success requires both members of a MiP pair to succeed. Solver expenditure totals each task set, excluding Jev, user simulation, and external scoring. Complete comparisons appear in Appendix 8.15.

Population	**Metric**	**Two-stage**	**ClarifyAgent**	**ResearchSpec**
MiP: 200 tasks	Task-appropriate success	79.00	—	**86.50**
MiP: 100 pairs	Joint success	60.00	—	**75.00**
ClarifyMT: 360 tasks	Action accuracy	61.67	**79.72**	77.22
Solver expenditure (USD)				
MiP: 200 tasks	Total solver cost	0.378	—	1.815
ClarifyMT: 360 tasks	Total solver cost	0.764	1.748	4.330

The full-benchmark breakdowns clarify which information distinctions matter. FaithEval gains concentrate on unanswerable (+5.06 points) and counterfactual contexts (+4.20), with both systems at 93.00% on inconsistent contexts. The former requires recognizing absent support; the latter requires using a supplied premise despite conflicting prior knowledge. On ClarifyMT, equally weighting answer-required and clarify-required tasks yields balanced accuracies of 62.76% and 73.64%, retaining an overall improvement under class-balanced evaluation. Action accuracy on answer-required tasks changes from 72.01% to 69.36%; the larger gains concern seeking information when required, including after factually wrong or contradictory feedback (Appendix 8.10).

From a pragmatic perspective, grounding requires understanding sufficient for the conversational purpose (Clark & Brennan, 1991); it does not require resolving every unspecified detail. Making evidential dependencies explicit can expose gaps obscured by a plausible answer, but treating optional details as necessary can also impede a valid response. Permission to choose an interpretation does not itself supply missing factual premises, so an action-only metric may reward an answer whose content remains unsupported. Conversely, ResearchSpec can request clarification when existing authorization already permits a reasonable choice (Appendix 8.15). These contrasting cases illustrate both the limited scope of action-only evaluation and genuine over-clarification. The class-balanced gain supports a net improvement under the evaluated criteria; the examples neither establish the frequency of these patterns nor attribute the observed losses to dataset defects.

We separately examine the incremental effect of explicit source links within the complete procedure. On 124 FaithEval tasks, full ResearchSpec, a variant without explicit condition-to-source links, and two-stage analysis attain 79.84%, 83.06%, and 75.81% ACC-N, respectively. Both condition variants retain the original context; removing links changes their construction and presentation, not evidence availability. The primary full-minus-ablation contrast is -3.23 points (Table 3). Thus, explicit links provide traceable records, while their incremental accuracy contribution is not established here. Both condition variants attain 85.25% on the 61 unanswerable tasks, versus 72.13% for two-stage analysis, consistent with the broader gains in handling missing information.

6.3 Maintaining Conditions across Turns

Information established through clarification must remain available to constrain later responses. On full SciConvBench, memory consistency improves by 7.18 points and CGRR by 3.68. These measure different aspects of interaction from final resolution, intent capture, and explicit conflict reporting on inconsistency tasks, which change by -1.75, -2.98, and -6.47 points, respectively. The CGRR gain concerns successful outcomes with conversational support, rather than an increase in overall final completion.

To examine state reuse directly, we compare incremental maintenance with reconstruction from the complete dialogue on 55 tasks, sharing the first-turn prefix and disabling historical assessment feedback in both variants. Incremental maintenance reduces solver expenditure by 25.89%; including user simulation and Jev, combined inference cost falls by 24.45%. The saving concentrates in condition construction, whose completion-token count falls by 40.37%, despite more condition calls and clarification questions. Both variants yield the same CGRR and final-resolution point estimates:

Table 3: Component comparisons on separately fixed task sets: source links use FaithEval; state maintenance and feedback use SciConvBench. Reference/alternative denote full/no-link conditions, incremental/reconstructed state, and enabled/disabled feedback, respectively. Differences are reference minus alternative in percentage points. State comparison disables feedback in both variants.

Component	Metric	N	Reference	Alternative	Δ
Source links	ACC-N	124	79.84	83.06	-3.23
State maintenance	MC	55	94.55	92.73	$+1.82$
History feedback	CGRR	100	58.00	51.00	$+7.00$

58.18% and 60.00%, respectively. Memory consistency is 94.55% versus 92.73%, a 1.82-point difference. In this 55-task comparison, incremental updates use fewer condition-generation output tokens and less solver expenditure to represent the evolving dialogue (Appendix 8.17).

A saved trajectory illustrates what is retained: competing cell-relaxation instructions form a conflict condition, and the user confirms a prohibition covering slab calculations. The updated condition records that scope, and the final specification excludes the earlier exception (Figure 2). More broadly, 47 tasks retain successful resolution while moving from Pure LLM's silent resolution to conversational grounding under ResearchSpec; descriptive trace review identifies targeted clarification followed by confirmation in 40 of these cases (Appendix 8.12).

6.4 Assessment Signals and Subsequent Interaction

Response assessment evaluates the published candidate and can inform subsequent interaction; it does not revise single-turn answers. We therefore examine its diagnostic information separately from generation performance. Among candidate-bound FaithEval assessments, ACC-N is 85.82% for PASS responses and 68.23% for FAIL responses. This ordering also holds within each of the three context subsets, and in the MiP and ClarifyMT comparisons. The consistent association provides evidence of diagnostic information in assessments of the generated responses. UNKNOWN assessments are reported separately (Appendix 8.18).

In 100 scientific dialogues, retaining historical assessment feedback yields 58.00% CGRR versus 51.00% when feedback is withheld, a 7.00-point difference. Feedback enters 56 subsequent generation requests across 34 tasks. This comparison concerns the complete historical-report pathway, including semantic assessments and deterministic clarification checks. The observed gain describes this fixed cohort and its generated dialogue trajectories (Appendix 8.17). These analyses characterize how the integrated procedure selects responses, reuses condition state, and supplies candidate-bound information for inspection and subsequent interaction.

7 Conclusion

We introduced ResearchSpec, an integrated procedure for grounded answering and clarification that records task necessity, evidence, and scoped authorization. Across 13,872 tasks from four benchmarks, the complete ResearchSpec procedure improves each selected primary metric by 3.37–15.39 percentage points over direct generation with the same backbone, without additional task-specific training. On fixed MiP and ClarifyMT cohorts, it improves response selection over free-form two-stage analysis at higher solver expenditure. On 55 interactive tasks, incremental condition maintenance uses 25.89% less estimated solver cost than reconstruction and has equal observed grounded and final resolution point estimates. Candidate-bound assessments provide diagnostic signals for examining response support and subsequent interaction. These findings support the complete procedure for organizing responses around task requirements and their available support, with tradeoffs in selectivity and computation.

AI USE STATEMENT

We used AI tools to assist with research ideation and execution, the discovery of related work, drafting portions of the manuscript, and editing and polishing its prose. This assistance included discussing study design, supporting experimental implementation and analysis, suggesting relevant literature, and proposing text revisions. The authors are responsible for the final manuscript, citations, reported results, and conclusions.

REFERENCES

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 9112–9141, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/25f7be9694d7b32d5cc670927b8091e1-Abstract-Conference.html.

Maximillian Chen, Ruoxi Sun, Tomas Pfister, and Sercan Ö. Arık. Learning to clarify: Multi-turn conversations with action-based contrastive self-training. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 32244–32279, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/4ffd05ca3cf3985f4572af015b4cfc1e-Abstract-Conference.html.

Herbert H. Clark and Susan E. Brennan. Grounding in communication. In Lauren B. Resnick, John M. Levine, and Stephanie D. Teasley (eds.), *Perspectives on Socially Shared Cognition*, pp. 127–149. American Psychological Association, Washington, DC, 1991. doi: 10.1037/10096-006. URL https://doi.org/10.1037/10096-006.

Chenrui Fan, Ming Li, Lichao Sun, and Tianyi Zhou. Missing premise exacerbates overthinking: Are reasoning models losing critical thinking skill?, 2025. URL https://arxiv.org/abs/2504.06514.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6465–6488, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.398. URL https://aclanthology.org/2023.emnlp-main.398/.

Yifan Gao, Chien-Sheng Wu, Shafiq Joty, Caiming Xiong, Richard Socher, Irwin King, Michael Lyu, and Steven C.H. Hoi. Explicit memory tracker with coarse-to-fine reasoning for conversational machine reading. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 935–945, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.88. URL https://aclanthology.org/2020.acl-main.88/.

Jonathan Ginzburg and Robin Cooper. Clarification, ellipsis, and the nature of contextual updates in dialogue. *Linguistics and Philosophy*, 27(3):297–365, June 2004. doi: 10.1023/B:LING.0000023369.19306.90. URL https://doi.org/10.1023/B:LING.0000023369.19306.90.

H. P. Grice. Logic and conversation. In Peter Cole and Jerry L. Morgan (eds.), *Speech Acts*, volume 3 of *Syntax and Semantics*, pp. 41–58. Academic Press, New York, 1975. doi: 10.1163/9789004368811_003. URL https://web.stanford.edu/class/psych205/papers/Grice-1975.pdf.

Hailey Joren, Jianyi Zhang, Chun-Sung Ferng, Da-Cheng Juan, Ankur Taly, and Cyrus Rashtchian. Sufficient context: A new lens on retrieval augmented generation systems. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 20310–20334, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/33dffa2e3d2ab74a783d1a8c292f66d9-Abstract-Conference.html.

Staffan Larsson and David R. Traum. Information state and dialogue management in the TRINDI dialogue move engine toolkit. *Natural Language Engineering*, 6(3–4):323–340, September 2000. doi: 10.1017/S1351324900002539. URL https://doi.org/10.1017/S1351324900002539.

Ruiqing Li, Yifeng Xu, Xinke Jiang, Zhibang Yang, Xinyu Ma, Yue Fang, Junfeng Zhao, Yasha Wang, and Xu Chu. GPS: Graph-guided proactive information seeking in large language models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 18332–18356, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/1f49b7fbd7c7afc52c1db0d4ed1a338d-Abstract-Conference.html.

Sichun Luo, Yi Huang, Mukai Li, Shichang Meng, Fengyuan Liu, Zefa Hu, Junlan Feng, and Qi Liu. ClarifyMT-Bench: Benchmarking and improving multi-turn clarification for conversational large language models, 2025. URL https://arxiv.org/abs/2512.21120.

Yifei Ming, Senthil Purushwalkam, Shrey Pandit, Zixuan Ke, Xuan-Phi Nguyen, Caiming Xiong, and Shafiq Joty. FaithEval: Can your language model stay faithful to context, even if "the moon is made of marshmallows". In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 29430–29456, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/48404cd9ce03946c6b7177691f3267a1-Abstract-Conference.html.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. Measuring attribution in natural language generation models. *Computational Linguistics*, 49(4):777–840, December 2023. doi: 10.1162/coli_a_00486. URL https://aclanthology.org/2023.cl-4.2/.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8689–8696, April 2020. doi: 10.1609/aaai.v34i05.6394. URL https://ojs.aaai.org/index.php/AAAI/article/view/6394.

Craige Roberts. Information structure in discourse: Towards an integrated formal theory of pragmatics. *Semantics and Pragmatics*, 5(6):1–69, December 2012. doi: 10.3765/sp.5.6. URL https://semprag.org/index.php/sp/article/view/sp.5.6.

Marzieh Saeidi, Max Bartolo, Patrick Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. Interpretation of natural language rules in conversational machine reading. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2087–2097, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1233. URL https://aclanthology.org/D18-1233/.

Omar Shaikh, Kristina Gligorić, Ashna Khetan, Matthias Gerstgrasser, Diyi Yang, and Dan Jurafsky. Grounding gaps in language model generations. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6279–6296, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.348. URL https://aclanthology.org/2024.naacl-long.348/.

Nithin Somasekharan, Youssef Hassan, Shiyao Lin, Gihan Panapitiya, Patrick Emami, Anurag Acharya, Sameera Horawalavithana, and Shaowu Pan. SCICONVBENCH: Benchmarking LLMs on multi-turn clarification for task formulation in computational science, 2026. URL https://arxiv.org/abs/2605.18630.

Robert Stalnaker. Common ground. *Linguistics and Philosophy*, 25(5–6):701–721, December 2002. doi: 10.1023/A:1020867916902. URL https://doi.org/10.1023/A:1020867916902.

Chien-Sheng Wu, Andrea Madotto, Ehsan Hosseini-Asl, Caiming Xiong, Richard Socher, and Pascale Fung. Transferable multi-domain state generator for task-oriented dialogue systems. In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 808–819, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1078. URL https://aclanthology.org/P19-1078/.

Michael JQ Zhang and Eunsol Choi. Clarify when necessary: Resolving ambiguity through interaction with LMs. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 5541–5558, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-naacl.306. URL https://aclanthology.org/2025.findings-naacl.306/.

Victor Zhong and Luke Zettlemoyer. E3: Entailment-driven extracting and editing for conversational machine reading. In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2310–2320, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1223. URL https://aclanthology.org/P19-1223/.

8 Appendix

8.1 Notation and Assessment Objects

The notation in Sections 3–4 distinguishes three objects: the visible context $x_t = (q, D, H_t)$, its revisable condition representation $\mathcal{C}_t$, and a candidate (y, u) with intended use $u \in \{\text{answer}, \text{clarify}\}$. Support and coverage concern that candidate; sufficiency concerns the existence of an adequate substantive answer in $\mathcal{Y}_q^{\text{sub}}$. Evidence standards are task-relative: a supplied passage can ground a factual answer, while user permission can ground a design choice without establishing an empirical fact.

For Algorithm 8.6, Δ_t contains structurally validated new or revised records, I_t their identifiers, and R_t the current records declared as used or targeted by generation $\mathcal{G}$. Assessment $\mathcal{J}$ returns report r_t; feedback f_t retains a non-PASS report with its candidate when interaction continues, and is otherwise empty. The reply z_t enters the next context. The nonnegative allowance B limits clarification exchanges; $b_t = \mathbf{1}[t = B + 1]$ requires an answer at the boundary.

8.2 Illustrative Cases for the Formulation

These constructed examples illustrate Equations equation 1–equation 2. A notice stating opening and closing times of 09:00 and 17:00 makes a request for both times answerable, but "18:00" is unsupported and "opens at 09:00" alone lacks coverage. For a 120-km journey without a speed, an exact numerical duration is undetermined; $T = 120/v$ suffices if the task instead requests a symbolic relationship, and "No; speed is unspecified" answers whether duration is uniquely determined. Finally, permission to choose a sampling interval between one and five seconds supports a proposed two-second interval, but does not establish that measurements were actually collected at that interval. Thus sufficiency depends on the requested result, and authorization changes permitted choices rather than observed facts.

8.3 Comparison with Related Approaches

E3 uses condition coverage and rule editing for follow-up questions (Zhong & Zettlemoyer, 2019); EMT updates rule memories with entailment states (Gao et al., 2020); ClarifyAgent tracks required slots (Luo et al., 2025); and GPS traverses a conditional-knowledge graph (Li et al., 2026). ALCE evaluates statement–source attribution (Gao et al., 2023), while Self-RAG uses learned reflection tokens to guide retrieval and generation (Asai et al., 2024). ResearchSpec integrates revisable task conditions, source links, scoped authorization, and candidate support/coverage assessment in one procedure. Condition tracking itself is therefore not new, and the present comparisons do not establish an independent accuracy benefit for every field.

8.4 CONDITION RECORDS AND UPDATE PROTOCOL

Table 4 serializes Equation equation 3; the necessity indicator n is Boolean `critical` (1 for true). Records describe dependencies of the requested output; response-format instructions and evaluation rubrics remain task instructions.

Table 4: Fields of a task condition record.

Field	Content
`id`	Stable condition identifier; null in a declaration of a new condition, for which an identifier is assigned.
`version`	Positive revision number, beginning at 1 and increasing by 1 for each update to that condition.
`operation` (o)	The operation, relation, or deliverable constraint that uses the condition.
`content` (d)	The required fact, rule, or dependency, distinguished from its current value and the final response.
`value` (v)	A nonempty string recording the scoped value or relation, or null where permitted by the status.
`critical` (n)	Whether the requested result requires the condition under the task instructions.
`state` (s)	One of confirmed, missing, conflict, authorized, or optional.
`evidence` (E)	A nonempty list of source references, each specifying a source identifier, source version, and paragraph number.
`reason`	A concise explanation of relevance, necessity or permissible freedom, and the assigned status.

Status, values, and sources. Confirmed, conflict, and authorized records require nonempty values; conflict values state the incompatible claims, and authorized values retain the scope of permission. Missing values are null; optional values may be null and require `critical=false`. Numerical strings include units. Distinct scopes or multiple acceptable answers alone do not imply conflict or a preferred choice. Each source paragraph is stored once and resolved through the catalog by identifier, version, and paragraph number. Missing records cite the requirement; authorized records cite the permission. Locations establish provenance, not semantic correctness. Generated records and prior assistant text are not independent factual evidence.

Persistent updates. Only new or revised records are returned; each existing identifier can advance exactly one version, at most once per call. Field types, status–value consistency, and source locations are validated before replacing affected records; unaffected records persist. Updates must preserve unrelated constraints, qualifiers, and competing evidence. Recency alone does not resolve conflict.

8.5 STRUCTURED ASSESSMENT PROTOCOL

Jev implements $\mathcal{J}$ with one request containing the exact candidate, intended use, current condition versions, source catalog, and history. It assesses:

1. **Declarations:** whether cited material warrants each condition's operation, dependency, scoped value, necessity, status, and rationale. Correctly identified gaps can pass.
2. **Condition-set coverage:** whether the representation captures the original task's material facts, rules, constraints, dependencies, and competing evidence, without requiring details left open.
3. **Candidate support:** whether evidence and permissions warrant the exact response, including negative findings, qualifiers, and any unique or preferred conclusion.
4. **Original-task coverage:** whether the response fulfills its intended use, retains supplied constraints, and handles visible conflicts.
5. **Condition-level coverage:** whether an answer respects all current conditions, or a clarification targets the declared R_t. Optional omissions can pass; clarification must address necessary unresolved information not already supplied.

PASS means supported for the intended use, FAIL contradicted, incorrect, or inconsistent with a visible requirement, and UNKNOWN not established by the material. Aggregation gives FAIL precedence over UNKNOWN, and otherwise returns PASS. A deterministic check also fails an interactive clarification unless its nonempty target set consists of necessary missing or conflicting records. Reports retain component outcomes and exact identifier/version associations. These checks bind fallible model judgments to assessed objects; they do not establish semantic correctness.

8.6 RESPONSE PROTOCOL AND INTERACTION PROCEDURE

[t] [1] Initial context $x_1 = (q, D, H_1)$; clarification allowance $B \in \mathbb{Z}_{\geq 0}$ $\mathcal{C}_0 \leftarrow \varnothing$; $f_0 \leftarrow \varnothing$ $t = 1, \ldots, B + 1$ Obtain structurally valid condition updates Δ_t from $(x_t, \mathcal{C}_{t-1})$ Form I_t from Δ_t and update $\mathcal{C}_t$ using Equation equation 4 $b_t \leftarrow \mathbf{1}[t = B + 1]$ $(y_t, u_t, R_t) \leftarrow \mathcal{G}(x_t, \mathcal{C}_t, f_{t-1}; b_t)$ $r_t \leftarrow \mathcal{J}(x_t, \mathcal{C}_t, y_t, u_t, R_t)$ Emit y_t $u_t = \text{answer}$ y_t $f_t \leftarrow (y_t, u_t, R_t, r_t)$ if r_t is non-PASS; otherwise $f_t \leftarrow \varnothing$ Receive the user's reply z_t Append assistant text y_t and user reply z_t to H_t, yielding H_{t+1} $x_{t+1} \leftarrow (q, D, H_{t+1})$

Candidate and publication. Generation returns response text, an action (`ask` or `final`), and distinct existing condition identifiers R_t at their current versions. Single-response tasks require `final` with the native answer format; selecting a clarification question completes such a task without a new exchange. Insufficiency or conflict findings appear in the text when appropriate. Algorithm 8.6 updates conditions, generates, assesses, and publishes the candidate unchanged. Assessment is bound to its intended use, original evidence, condition versions, task scope, and policy; it does not revise or suppress that candidate. After clarification, the delivered user reply enters the next update. A non-PASS report and its candidate enter only the next generation call, retaining their original version associations; a PASS clears this feedback. A final response ends interaction.

Interaction boundary. SciConvBench requests a specification, with at most one question per turn about a necessary unresolved condition and no proposed answer or repair. It permits $B = 10$ question–reply exchanges; single-response tasks use $B = 0$. At $b_t = 1$, the final delivered reply is incorporated and conditions updated before the required final specification, which must state remaining uncertainty. Only original public material and delivered replies supply evidence; undisclosed requirements remain with the simulator and evaluation process.

8.7 MODEL CONFIGURATION AND BENCHMARK INPUTS

Generation and assessment. Pure LLM and ResearchSpec use the API model identifier `deepseek-flash` with thinking enabled, reasoning effort set to `low`, and `max_tokens`=32768 per request. ResearchSpec uses the same configuration for condition declarations and candidate generation, with structured JSON output for both calls. Its assessor uses `jev-latest` through the Jev structured-decision interface, with the criteria described in Appendix 8.5. The 1,236 retained responses with a served-model field in the fixed baseline comparison identify `jev-1.13.0`; this records the observed service version rather than a version-pinned request across all evaluations. The SciConvBench user simulator uses `deepseek-flash` with the same reasoning and token-limit settings as the generator. External model-based evaluation uses `deepseek-v4-pro`, thinking enabled, reasoning effort `high`, and `max_tokens`=32768. DeepSeek requests specify no overrides for `temperature`, `top_p`, or `seed`. Each system generates one candidate per response turn; assessment outcomes do not trigger an additional candidate within that turn.

Public task inputs. MiP inputs contain the released original or missing-premise question. FaithEval inputs contain the question and supplied context, with answer options included for its counterfactual setting. The instructions require adherence to the supplied context, using `unknown` for insufficient information and `conflict` for conflicting answers where applicable. ClarifyMT uses the first three messages of each released conversation, following the benchmark's inference code, together with its answer-or-clarify system instruction. The response is requested to begin with "The answer is" or "The clarifying question is." Later dialogue messages are excluded from model inputs.

Scientific dialogue. SciConvBench includes the released `foam`, `fluids`, `pde`, `matSci`, `matToolUse`, `solMech`, and `solToolUse` collections, each with disambiguation and inconsis-

tency tasks. Both systems receive the released scientific task-formulation instructions and the initial incomplete or inconsistent request. The simulator uses the released user prompts, which condition its replies on the reference specification and previous exchanges. Only its delivered response enters the assistant's context. The baseline retains the conversation history; ResearchSpec maintains that history together with its condition records and any applicable assessment feedback. The interaction allowance and finalization procedure are shared between systems as described in Appendix 8.6.

8.8 Benchmark Scoring and Aggregation

MiP. We retain the released three-trial procedure for paragraph-level detection of missing information and the released final-response classification prompt. The final classification distinguishes abstention due to insufficient conditions, correct answers, and incorrect answers; unparseable classifications are recorded separately. Missing-premise performance is measured by abstention rate, and original-problem performance by answer accuracy. Our aggregate task-appropriate success pools the corresponding successful outcomes and divides by the total number of tasks across the two task types. This aggregate combines the task-specific criteria; it is distinct from answer accuracy on the original problems. The formula collection contributes to the missing-premise criterion: its release supplies no numerical reference answers, so its outcome measures recognition of insufficient conditions.

FaithEval. We apply the released answer-normalization function, which lowercases text, replaces underscores and punctuation, removes articles, and collapses whitespace. ACC-S uses the canonical response markers for unanswerable and inconsistent contexts; ACC-N allows the released sets of alternative expressions. These criteria match the relevant phrases within the normalized full response. Counterfactual questions use containment of the normalized reference answer under both metrics. The benchmark-level ACC-N is the mean of the resulting binary scores across the three settings.

ClarifyMT. We use the released `is_correct_response` function. It lowercases and trims the response, then checks for `answer` for the Precise and Refusal user types and `clarify` for the remaining user types. The overall score is the proportion of tasks whose responses satisfy this rule. This metric evaluates the benchmark's answer-versus-clarification response criterion; evaluation uses the complete emitted response.

SciConvBench. The released judges assess final resolution, intent capture, and the clarification transcript. We apply the released case-level scoring rules, including their check for whether apparently silent resolutions were supported by the conversation. Final resolution rate (FRR) measures successful final specifications, with intent-capture failures counted as unsuccessful resolution. CGRR measures cases that satisfy both final resolution and conversational grounding, while silent resolution rate (SRR) measures final resolutions without that grounding. Intent capture (IC) assesses fidelity to the requested task, and memory consistency (MC) assesses retention of information established in the dialogue. Clarification recall (CR) and precision (CP) follow the released count-based definitions. Detection rate (DR) applies to inconsistency tasks and measures whether the conversation explicitly warns about the planted inconsistencies under the released rule. Each metric is averaged over eligible cases; DR uses only inconsistency cases. Evaluation uses the released task collections directly, without the original study's additional filtering based on outcomes from multiple comparison models.

Aggregation and output accounting. Benchmark-level summaries pool case scores across their constituent task collections; they do not average the collection percentages with equal weights. The primary results use the full eligible task set as a shared denominator for both systems. For success metrics, the numerator counts outcomes satisfying the corresponding criterion; other metrics sum the observed case scores. A task without an evaluable result contributes zero to the aggregate while remaining in the denominator. Rates conditional on evaluable outputs, when reported, are identified separately. Reference answers, private user specifications, and benchmark labels are used only by the user environment or external evaluation as appropriate, and are excluded from assistant generation and ResearchSpec assessment.

8.9 Detailed Benchmark Results

This subsection provides the task-collection and metric breakdowns underlying Section 6.1. Both systems use all N tasks in each reported group, following Appendix 8.8. Scores are expressed as percentages, and Δ is ResearchSpec minus Pure LLM in percentage points, computed before rounding. Overall scores pool task-level contributions across collections. The ClarifyMT breakdowns appear in Appendix 8.10.

Missing premises and context-dependent answering. Table 5 separates recognition of missing premises from answer accuracy on original problems. Its combined row applies the task-appropriate success criterion from Section 5.3. Table 6 reports both released FaithEval criteria for each context setting. The largest increase occurs on unanswerable contexts under ACC-N, while counterfactual contexts show the same gain under both criteria.

Table 5: MiP results (%). Missing-premise rows report appropriate abstention; original-problem rows report answer accuracy. The combined row pools successes under these respective criteria.

Collection	Task type	N	Pure LLM	ResearchSpec	Δ
All collections	Combined	1,618	72.37	86.46	+14.09
All collections	Missing	984	56.50	81.81	+25.30
All collections	Original	634	97.00	93.69	-3.31
GSM8K	Missing	582	46.56	73.88	+27.32
GSM8K	Original	582	96.74	93.47	-3.26
MATH	Missing	52	40.38	71.15	+30.77
MATH	Original	52	100.00	96.15	-3.85
SVAMP	Missing	300	76.00	96.00	+20.00
Formula	Missing	50	72.00	100.00	+28.00

Table 6: FaithEval results (%) by context setting and scoring criterion. ACC-S and ACC-N coincide on counterfactual tasks under the released scoring rule.

Context	Metric	N	Pure LLM	ResearchSpec	Δ
Overall	ACC-S	4,992	76.44	78.73	+2.28
Overall	ACC-N	4,992	77.38	80.75	+3.37
Unanswerable	ACC-S	2,492	69.46	72.35	+2.89
Unanswerable	ACC-N	2,492	71.35	76.40	+5.06
Inconsistent	ACC-S	1,500	93.00	93.00	+0.00
Inconsistent	ACC-N	1,500	93.00	93.00	+0.00
Counterfactual	ACC-S	1,000	69.00	73.20	+4.20
Counterfactual	ACC-N	1,000	69.00	73.20	+4.20

Scientific task formulation. Table 7 reports all eight SciConvBench metrics. Tables 8 and 9 disaggregate them by task type and collection, retaining the released collection identifiers. CGRR and SRR partition successful final resolutions, so $\mathrm{FRR} = \mathrm{CGRR} + \mathrm{SRR}$ before rounding; SRR is interpreted jointly with these resolution metrics. CR and CP average the released case-level count ratios, which can exceed one, and are displayed on a percentage scale without clipping. DR is defined only for inconsistency tasks.

Table 7: Full SciConvBench metric summary (%). DR uses the 572 inconsistency tasks; all other metrics use all 1,142 tasks. SRR is the silent-resolution component of FRR.

Metric	N	Pure LLM	ResearchSpec	Δ
FRR	1,142	61.91	60.16	-1.75
CGRR	1,142	54.99	58.67	+3.68
SRR	1,142	6.92	1.49	-5.43
IC	1,142	72.15	69.18	-2.98
MC	1,142	78.98	86.16	+7.18
CR	1,142	85.43	85.63	+0.20
CP	1,142	72.87	75.74	+2.87
DR	572	37.76	31.29	-6.47

Table 8: SciConvBench resolution and intent-capture results by collection and task type. Each metric cell reports Pure LLM / ResearchSpec (%). Dis. denotes disambiguation; Inc. denotes inconsistency.

Collection	Task	N	FRR	CGRR	SRR	IC
All	Dis.	570	47.37 / 45.44	43.68 / 43.68	3.68 / 1.75	66.67 / 60.53
All	Inc.	572	76.40 / 74.83	66.26 / 73.60	10.14 / 1.22	77.62 / 77.80
foam	Dis.	100	21.00 / 27.00	21.00 / 27.00	0.00 / 0.00	24.00 / 28.00
foam	Inc.	81	72.84 / 72.84	72.84 / 72.84	0.00 / 0.00	74.07 / 75.31
fluids	Dis.	51	82.35 / 72.55	74.51 / 64.71	7.84 / 7.84	96.08 / 84.31
fluids	Inc.	74	78.38 / 70.27	75.68 / 68.92	2.70 / 1.35	78.38 / 70.27
pde	Dis.	61	70.49 / 80.33	62.30 / 77.05	8.20 / 3.28	70.49 / 81.97
pde	Inc.	73	79.45 / 84.93	71.23 / 84.93	8.22 / 0.00	82.19 / 84.93
matSci	Dis.	82	89.02 / 76.83	79.27 / 74.39	9.76 / 2.44	100.00 / 86.59
matSci	Inc.	81	79.01 / 76.54	50.62 / 71.60	28.40 / 4.94	80.25 / 80.25
matToolUse	Dis.	48	54.17 / 54.17	52.08 / 54.17	2.08 / 0.00	68.75 / 56.25
matToolUse	Inc.	93	66.67 / 55.91	52.69 / 54.84	13.98 / 1.08	68.82 / 66.67
solMech	Dis.	143	21.68 / 16.78	20.28 / 16.08	1.40 / 0.70	65.73 / 52.45
solMech	Inc.	120	83.33 / 81.67	74.17 / 81.67	9.17 / 0.00	83.33 / 82.50
solToolUse	Dis.	85	40.00 / 38.82	38.82 / 37.65	1.18 / 1.18	64.71 / 60.00
solToolUse	Inc.	50	72.00 / 86.00	66.00 / 84.00	6.00 / 2.00	74.00 / 88.00

Table 9: SciConvBench dialogue metrics by collection and task type. Each metric cell reports Pure LLM / ResearchSpec (%). A dash marks DR where it is undefined. Task abbreviations follow Table 8.

Collection	Task	N	MC	CR	CP	DR
All	Dis.	570	76.14 / 84.39	100.20 / 92.18	63.37 / 65.60	—
All	Inc.	572	81.82 / 87.94	70.72 / 79.11	82.34 / 85.84	37.76 / 31.29
foam	Dis.	100	97.00 / 91.00	190.52 / 156.54	83.71 / 72.01	—
foam	Inc.	81	95.06 / 88.89	83.95 / 72.84	93.83 / 83.95	70.37 / 32.10
fluids	Dis.	51	86.27 / 96.08	116.93 / 109.35	70.59 / 61.86	—
fluids	Inc.	74	89.19 / 91.89	72.97 / 67.57	124.32 / 94.59	44.59 / 27.03
pde	Dis.	61	70.49 / 85.25	115.57 / 111.48	60.93 / 75.41	—
pde	Inc.	73	71.23 / 86.30	71.23 / 86.30	71.23 / 86.30	47.95 / 43.84

Table 9 continued.

Collection	Task	N	MC	CR	CP	DR
matSci	Dis.	82	74.39 / 90.24	59.86 / 77.24	63.41 / 85.77	—
matSci	Inc.	81	54.32 / 82.72	37.04 / 76.54	38.27 / 86.42	34.57 / 28.40
matToolUse	Dis.	48	64.58 / 85.42	108.90 / 113.30	58.73 / 52.15	—
matToolUse	Inc.	93	89.25 / 90.32	70.97 / 80.11	83.87 / 87.10	35.48 / 32.26
solMech	Dis.	143	76.92 / 87.41	54.28 / 54.84	67.82 / 70.24	—
solMech	Inc.	120	85.00 / 85.83	79.58 / 84.58	75.42 / 77.50	15.00 / 30.00
solToolUse	Dis.	85	56.47 / 57.65	84.14 / 57.64	31.92 / 33.63	—
solToolUse	Inc.	50	88.00 / 92.00	78.00 / 85.00	103.00 / 92.00	24.00 / 24.00

8.10 Ambiguity and User-Feedback Results

Tables 10–12 provide the full ClarifyMT breakdowns for Section 6.2. Each of the twelve ambiguity categories contains 510 tasks; each of the six user-feedback types contains 1,020 tasks, with 85 tasks in each intersection. All scores use the released response criterion described in Appendix 8.8.

Ambiguity and feedback. The improvement across all twelve ambiguity categories in Table 10 spans distinct sources of unresolved information. Table 11 separates this pattern by user feedback and required response. The four feedback types assigned to clarification all improve, with the largest gain on factually wrong feedback. The Precise and Refusal types are assigned to answering, and are reported separately from the clarification types. The aggregate therefore reflects each task's response requirement and its frequency in the evaluation set. Table 12 gives all category–feedback intersections to distinguish the category-level pattern from its distribution within individual feedback types.

Table 10: ClarifyMT response accuracy (%) across all twelve ambiguity categories. Each category contains 510 tasks.

Ambiguity category	N	Pure LLM	ResearchSpec	Δ
Entity	510	48.82	76.27	+27.45
Goal	510	70.20	76.86	+6.67
Intent Conflict	510	63.92	78.82	+14.90
Knowledge Gap	510	66.67	68.63	+1.96
Lexical	510	50.39	70.59	+20.20
Scope	510	47.25	64.51	+17.25
Semantic	510	56.67	76.27	+19.61
Spatial	510	60.20	72.75	+12.55
Syntactic	510	64.71	75.29	+10.59
Temporal	510	65.29	73.53	+8.24
Unfamiliarity	510	61.57	83.53	+21.96
Value	510	60.39	83.73	+23.33

Table 11: ClarifyMT response accuracy (%) by required response and user-feedback type. Overall performance pools all 6,120 tasks; Answer and Clarify comprise 2,040 and 4,080 tasks, respectively.

Group	Required response	*N*	Pure LLM	ResearchSpec	Δ
Overall	Mixed	6,120	59.67	75.07	+15.39
Answer tasks	Answer	2,040	72.01	69.36	-2.65
Clarify tasks	Clarify	4,080	53.50	77.92	+24.41
Contradictory	Clarify	1,020	56.76	77.94	+21.18
Factually-Wrong	Clarify	1,020	31.76	70.29	+38.53
Off-Focus	Clarify	1,020	71.96	90.69	+18.73
Partial-Vague	Clarify	1,020	53.53	72.75	+19.22
Precise	Answer	1,020	68.73	66.47	-2.25
Refusal	Answer	1,020	75.29	72.25	-3.04

Table 12: All 72 ClarifyMT category–feedback groups. Each cell reports Pure LLM / ResearchSpec response accuracy (%) on the same 85 tasks. Column headings abbreviate the feedback labels in Table 11.

Ambiguity	Contrad.	Factually wrong	Off focus	Partial vague	Precise	Refusal
Entity	12.94 / 49.41	3.53 / 54.12	42.35 / 85.88	34.12 / 78.82	100.00 / 95.29	100.00 / 94.12
Goal	87.06 / 94.12	50.59 / 70.59	85.88 / 91.76	87.06 / 88.24	47.06 / 51.76	63.53 / 64.71
Intent Conflict	74.12 / 91.76	52.94 / 85.88	78.82 / 88.24	58.82 / 77.65	45.88 / 60.00	72.94 / 69.41
Knowledge Gap	100.00 / 96.47	72.94 / 70.59	89.41 / 92.94	100.00 / 98.82	28.24 / 31.76	9.41 / 21.18
Lexical	17.65 / 42.35	18.82 / 57.65	40.00 / 75.29	38.82 / 68.24	90.59 / 88.24	96.47 / 91.76
Scope	22.35 / 58.82	2.35 / 40.00	57.65 / 84.71	2.35 / 12.94	100.00 / 94.12	98.82 / 96.47
Semantic	56.47 / 91.76	14.12 / 69.41	78.82 / 97.65	31.76 / 70.59	72.94 / 57.65	85.88 / 70.59
Spatial	60.00 / 82.35	38.82 / 65.88	72.94 / 92.94	64.71 / 77.65	62.35 / 60.00	62.35 / 57.65
Syntactic	70.59 / 77.65	44.71 / 85.88	88.24 / 95.29	51.76 / 64.71	56.47 / 54.12	76.47 / 74.12
Temporal	75.29 / 78.82	48.24 / 80.00	84.71 / 94.12	78.82 / 82.35	56.47 / 51.76	48.24 / 54.12
Unfamiliarity	48.24 / 84.71	11.76 / 81.18	75.29 / 92.94	49.41 / 84.71	85.88 / 76.47	98.82 / 81.18
Value	56.47 / 87.06	22.35 / 82.35	69.41 / 96.47	44.71 / 68.24	78.82 / 76.47	90.59 / 91.76

Leading response action. We additionally check whether the response begins with the required action, using the prefixes "The answer is" and "The clarifying question is" after lowercasing and removing leading whitespace. This diagnostic localizes the response decision to the opening declaration, complementing the released full-response matching rule. Table 13 shows a 15.49-point overall gain, close to the 15.39-point gain under the released criterion. The agreement supports interpreting the improvement as a change in declared response selection. Both measures concern the answer-or-clarify decision; content quality remains a separate criterion.

Table 13: Leading-action accuracy on ClarifyMT (%). Each row uses all tasks in the corresponding group.

Required response	*N*	Pure LLM	ResearchSpec	Δ
Overall	6,120	59.36	74.85	+15.49
Answer	2,040	71.52	68.92	-2.60
Clarify	4,080	53.28	77.82	+24.53

8.11 Paired Responses and Supplementary Comparisons

Responses to paired premise conditions. MiP contains 634 source problems with both an original and a missing-premise version: 582 from GSM8K and 52 from MATH. We define joint success

on a source pair as a correct answer to its original problem together with appropriate abstention on its missing-premise version, using the released final-response classifications for both. Every eligible source pair remains in the denominator. This supplementary criterion is distinct from the benchmark-level task-appropriate success in Table 1: it requires both responses to succeed on the same source problem.

Table 14 shows an increase from 286 to 440 jointly successful pairs, or 24.29 percentage points. The improvement occurs in both source collections. Joint success requires a system to solve the complete problem while recognizing when the altered premise set no longer supports a definite answer. It therefore examines the responsiveness of the answer decision to task conditions within matched problem families.

Table 14: Joint success on original and missing-premise MiP pairs. Counts are successful pairs; Δ uses all pairs in the row and is expressed in percentage points.

Collection	Pairs	Jointly successful pairs		Joint success (%)	
		Pure LLM	ResearchSpec	Pure / ResearchSpec	Δ
GSM8K	582	265	405	45.53 / 69.59	+24.05
MATH	52	21	35	40.38 / 67.31	+26.92
Overall	634	286	440	45.11 / 69.40	+24.29

Paired benchmark outcomes. Table 15 compares both systems on every task under the primary criteria in Table 1. The differences between system-exclusive successes reproduce the benchmark-level improvements: net increases of 228 successes on MiP, 168 on FaithEval, 942 on ClarifyMT, and 42 on SciConvBench. Non-success includes all zero contributions under Appendix 8.8; the table is an accounting of paired outcomes on the full task populations.

Table 15: Paired outcomes under each benchmark's primary criterion. The four outcome columns partition all N tasks. "Only" identifies the system satisfying the criterion.

Benchmark	N	Both succeed	Pure only	ResearchSpec only	Neither succeeds
MiP	1,618	1,107	64	292	155
FaithEval	4,992	3,736	127	295	834
ClarifyMT	6,120	3,387	265	1,207	1,261
SciConvBench	1,142	504	124	166	348

Source groups and supplementary evaluation stratum. The four evaluation collections contain 984, 3,957, 1,020, and 911 source groups, respectively. Source groups associate related task variants, including original and missing-premise problems and different user-feedback realizations. The main results include the complete task collections. Table 16 additionally restricts evaluation to tasks whose source groups are not marked as previously used in method development in the retained provenance records. This distinction concerns the project's evaluation history and makes no claim about model pretraining exposure. Each stratum uses its full task population, with the same criteria and task weighting as the main results. All four primary comparisons retain their positive direction in this stratum.

Table 16: Primary scores (%) on the supplementary source stratum. N counts tasks; the source-group column records their grouping.

Benchmark	N	Source groups	Pure LLM	ResearchSpec	Δ
MiP	1,516	923	72.49	86.41	+13.92
FaithEval	4,939	3,925	77.32	80.66	+3.34
ClarifyMT	6,000	1,000	59.70	74.93	+15.23
SciConvBench	952	775	54.20	57.56	+3.36

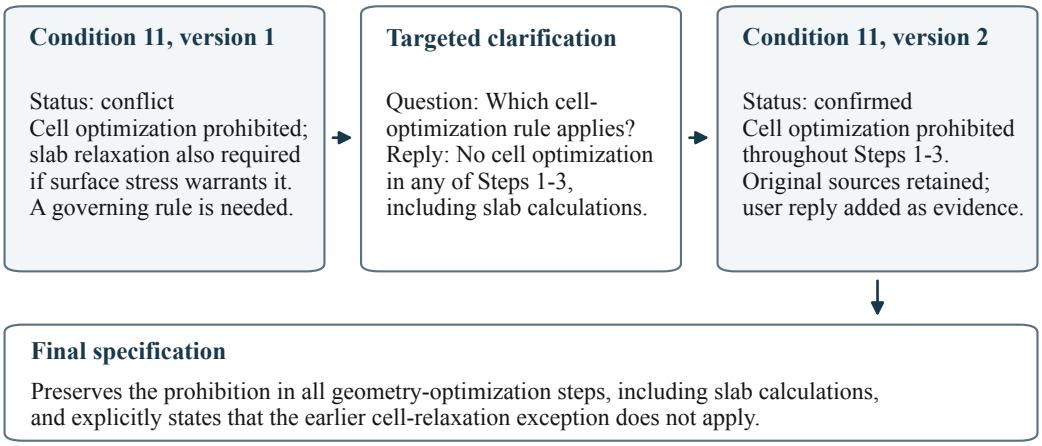


Figure 2: A condition's evolution in a SciConvBench dialogue. The same record links competing requirements, a targeted clarification, the user's confirmation, and the final specification. Text is condensed from the saved trajectory (`matToolUse`, `case_337`).

8.12 Condition Updates and Conversational Grounding

A persistent condition across clarification and finalization. Figure 2 shows SciConvBench's `matToolUse` inconsistency case `337`. The adsorption-energy workflow simultaneously says "Relax the cell for the slab calculations if the surface stresses are non-negligible" and "Do not optimize the unit cell (ISIF = 2)." Condition 11 (`specify_cell_optimization`) records this necessary dependency as conflicting. ResearchSpec asks whether the prohibition covers the slab calculations; the user confirms, "All steps (1–3) must use ISIF = 2." Version 2 retains the record's identity, operation, necessity, and original evidence, adds its two reply paragraphs to the original two source paragraphs, and changes the status to confirmed. The final specification retains this scope and explicitly excludes the cell-relaxation exception. Both systems ask two questions and achieve final resolution; Pure LLM asks about spin settings and an iteration limit and also prohibits cell optimization in its final specification. The distinction is ResearchSpec's explicit conversational confirmation, not exclusive final-answer success.

Delegated choice in a supplied dialogue. In ClarifyMT item 6077, a user requesting a respected scientist responds to a question about era and field with "I'll leave it up to you—just pick someone interesting." ResearchSpec cites this delegation as authorization and names Marie Curie, declaring use of that record. The preceding question belongs to the supplied benchmark history; this is one evaluated response, illustrating how permission can support a choice among several acceptable alternatives.

Outcome transitions and trajectory review. Table 17 partitions all 1,142 SciConvBench tasks by the two systems' recorded outcomes. Grounded and silent resolution both satisfy the final-resolution criterion, distinguished by the released conversational-grounding rule. The other two categories distinguish a scored non-resolution from an unavailable judgment; both contribute zero to aggregate resolution metrics. The row and column margins recover the full-population CGRR, SRR, and FRR in Table 7.

Table 17: Paired SciConvBench outcomes on all tasks. Rows denote Pure LLM and columns denote ResearchSpec. G: grounded resolution; S: silent resolution; U: scored non-resolution; NA: unavailable judgment.

Pure LLM \ ResearchSpec	G	S	U	NA	Total
G	504	4	105	15	628
S	47	11	20	1	79

Table 17 continued.

Pure LLM \ ResearchSpec	G	S	U	NA	Total
U	117	2	303	10	432
NA	2	0	1	0	3
Total	670	17	429	26	1,142

We reviewed all 85 tasks classified as silent resolution for at least one system, retaining their original benchmark scores. The model-assisted, non-blind review compared the saved questions, user replies, condition records, and final specifications with the released task annotations. Among the 47 transitions from silent to grounded resolution, 40 exhibit targeted questioning followed by user confirmation, one primarily reflects information volunteered by the user, three primarily reflect counting or lexical-matching boundaries in the scoring rule, and three combine these factors. These categories describe the observed dialogue paths; the 47 outcomes were already successful for Pure LLM. The full transition table additionally records 20 silent resolutions becoming scored non-resolutions, one becoming unavailable, and six transitions into silent resolution from other scored categories. The decrease in SRR therefore combines several outcome changes and is interpreted together with CGRR and FRR. The targeted-confirmation cases provide qualitative evidence about the conversational basis of successful specifications, with the condition trajectory above showing how that basis is maintained explicitly.

8.13 Recognition and Explicit Communication

Review protocol. We examined all 572 SciConvBench inconsistency tasks, containing 676 annotated issues, to distinguish four observations. *Recognition* requires condition records to identify the annotated incompatibility; *localization* requires a question to target that issue. *Disclosure* explicitly identifies the incompatibility, including through a contrasting scientific relation; *explanation* additionally states why it arises. A question offering two alternatives can localize an issue without disclosing their incompatibility. Recognition and localization count only evidence preceding explicit user correction of the corresponding issue, although other clarification may already have occurred. The categories were assigned in a model-assisted, non-blind review of saved trajectories against the released annotations. Task-level coverage requires every annotated issue to satisfy the criterion. Unavailable outputs contribute zero to aggregate rates while retaining unjudged semantic status.

Recognition and visible output. ResearchSpec's records identify 483 of 676 issues (71.45%), including 482 in the initial records. Combining recognition with correctly targeted questions covers 522 issues (77.22%). Among the 483 internally recognized issues, 332 are not explicitly disclosed in the dialogue. Internal recognition is reported for ResearchSpec, whose condition records provide this evidence; the between-system comparison concerns visible output. Table 18 distinguishes disclosure from explanation and separates dialogue from the combined dialogue and final specification. These observations distinguish identifying a task conflict from explicitly communicating it to the user.

Table 18: Complete issue coverage (%) over all 572 inconsistency tasks. Dialogue includes assistant questions; user replies are excluded from disclosure and explanation judgments.

Output scope	Criterion	Pure LLM	ResearchSpec
Dialogue	Disclosure	34.27	23.60
Dialogue	Explanation	30.07	13.81
Dialogue + final	Disclosure	40.38	37.94
Dialogue + final	Explanation	35.84	24.65

Separate final-report experiment. We evaluated a separate reporting extension on 552 tasks, reusing the original condition records and dialogue. A conflict report is appended to the final response when historical records contain conflicts; all tasks remain in the denominator, including unchanged responses.

Disclosure and explanation apply the all-issues task criterion to the issue-level union of original assistant output and appended report. These coverage measures do not fully penalize additional errors. The dialogue remains unchanged, and Table 19 retains benchmark DR's conversational-warning criterion and released judging prompts. The coverage gain concerns explicit communication of available task-state information, distinct from conflict recognition and resolution.

Table 19: Separate reporting experiment on 552 tasks (%). Coverage is assessed retrospectively over dialogue and final output, including appended reports; benchmark DR is scored separately.

Criterion	Pure LLM	ResearchSpec	ResearchSpec + report
Complete disclosure	40.76	38.77	76.99
Complete explanation	36.23	25.18	57.43
Benchmark DR	38.59	31.88	30.25

8.14 Comparative Evaluation Protocols

Evaluation cohorts. The comparisons use fixed cohorts defined from benchmark metadata before inspecting their outcomes. Related variants remain together in a source group, and every method in a comparison receives the same public task inputs. Table 20 records the evaluation units; these cohorts are distinct from the full populations in Table 1. The MiP comparison comprises 92 GSM8K and eight MATH original–missing pairs. ClarifyMT includes five source groups from each of its twelve ambiguity categories, retaining all six user-feedback variants per group. The scientific-dialogue comparison contains 48 disambiguation and 52 inconsistency tasks. The state-maintenance cohort is drawn from that scientific cohort and contains 26 disambiguation and 29 inconsistency tasks. The budget-targeted comparison uses 18 GSM8K and two MATH pairs from the 100-pair MiP cohort. The FaithEval comparison is drawn from all 4,992 tasks and contains 24 counterfactual, 39 inconsistent and 61 unanswerable tasks.

Table 20: Fixed comparison cohorts and prespecified primary endpoints. Task counts retain all selected variants.

Comparison	Tasks	Groups	Primary endpoint
MiP: two-stage analysis	200	100	Task-appropriate success
ClarifyMT: two baselines	360	60	Action accuracy
SciConvBench: historical feedback	100	81	CGRR
SciConvBench: state maintenance	55	44	MC
MiP: budget-targeted refinement	40	20	Joint pair success
FaithEval: source links	124	96	ACC-N, full minus no-link

Selection uses seed 20260925 and only source membership, dataset/category/domain, task mode and population counts. For MiP, pair quotas are proportional to dataset populations, using largest remainders and dataset-name order to break ties. ClarifyMT uses the fixed category quota above. Scientific source groups are stratified by domain and their set of task modes; FaithEval groups are stratified by their set of context types. For these latter comparisons, the number of groups selected from a stratum is the target task count times its group count divided by the eligible population's total task count, rounded to the nearest integer with halves rounded upward and bounded between one and the stratum's group count. Within each stratum, sorted source identifiers are shuffled using the string `seed:benchmark:stratum`, and the first allocated number of groups is retained without subsequent adjustment. Thus, complete groups yield 55 rather than 50 state-maintenance tasks and 124 rather than 120 FaithEval tasks.

Methods and shared inputs. Two-stage analysis first reasons in ordinary prose about the native task and then generates one response from the original messages and that analysis. It uses the same backbone, reasoning setting and per-call token limit as ResearchSpec; the two calls do not impose equal realized token expenditure. ClarifyAgent follows the released perceiver, forecaster, tracker, planner and output modules (Luo et al., 2025). Its prompts and normalization are fixed to upstream revision `f4417c89f4940d674faa4546d6d9a1d4fcf5c7cc`. The adaptation replaces the

local inference engine and its module-specific token caps with the shared API configuration. All five modules run, and evaluation scores the output module's text, rather than the planner's action label. Each ClarifyMT task begins with an empty tracker and the same native instruction plus three-message dialogue prefix used by the other methods.

The source-link comparison removes the per-condition `evidence` field and explicit location instructions from both condition construction and the answer's condition representation. It retains all original source text, the source catalog, remaining condition fields, necessity, state/value rules and scoped authorization semantics. Because conditions are generated under the intervened representation, their wording need not match the full method's conditions. The two fresh static comparators use one preparation call and one answer call; they omit the subsequent assessment that cannot change a single-turn candidate. The full method's unchanged saved outputs provide the paired control. Appendix 8.19 summarizes the stages and variants; the accompanying `prompt_templates.md` preserves the complete instructions, message templates and exact prompt changes.

Structured output handling. Condition and action calls request JSON objects and validate their fields, types, identifier references and state/value consistency. The parser removes an extraneous top-level `type: json_object` marker and canonicalizes a positive integer condition identifier only when it uniquely identifies an existing condition at exactly its next version; other declaration content is unchanged. Invalid structures are not repaired by generating alternative semantic candidates. A task without a valid published output remains in the evaluation denominator under the rule below.

Endpoints and reporting scope. Every selected task remains in its method's denominator; unavailable outputs or judgments contribute zero under Appendix 8.8. This rule also applies to reused controls, rather than conditioning comparisons on their availability. The MiP/ClarifyMT/scientific baseline comparisons retain 1,680 planned method records, of which 1,671 have evaluable outputs. The FaithEval comparison retains 372 planned records, including two unavailable full-method outputs; all 248 comparator outputs are evaluated. The state-maintenance comparison retains all 55 tasks in each arm, including three that terminate before the intervention. The budget-targeted comparison retains 120 method records on 40 tasks, including one unavailable full-method output.

Comparisons report observed task-weighted scores and paired outcomes on the selected source groups. Each configuration has one generation run; variation across independent runs is not evaluated. Secondary metrics and category breakdowns are descriptive; exploratory contrasts are identified explicitly.

8.15 Extended Response-Selection Comparisons

Free-form and slot-based comparisons. Table 21 reports the MiP and ClarifyMT comparisons, including the two response classes and the leading-action criterion defined in Appendix 8.10. On MiP, the full-minus-two-stage differences are 7.50 points for task-appropriate success and 15.00 points for joint pair success. The joint comparison has 24 pairs successful only for ResearchSpec, nine successful only for two-stage analysis, and 67 ties. Joint success tests adaptation to premise availability: both solving the original problem and recognizing the altered problem's insufficient premises must succeed. It excludes a strategy that always abstains, while retaining sensitivity to the tradeoff between answering and abstention.

On ClarifyMT, the difference from two-stage analysis is 15.56 points, with 75 ResearchSpec-only successes and 19 two-stage-only successes. The difference from ClarifyAgent is -2.50 points, with 32 and 41 exclusive successes, respectively. Using the response's opening action removes one success from each method and leaves both comparison differences unchanged. Balanced accuracy is the unweighted mean of the Answer and Clarify class accuracies.

Table 21: Detailed baseline comparisons (%). RS denotes ResearchSpec; TS denotes two-stage analysis; CA denotes ClarifyAgent. A dash marks an inapplicable method or denominator.

Criterion	N	RS	TS	CA
MiP: task-appropriate success	200	86.50	79.00	—
MiP: original answer accuracy	100	91.00	97.00	—
MiP: missing-premise abstention	100	82.00	61.00	—

Criterion	*N*	RS	TS	CA
MiP: joint pair success	100 pairs	75.00	60.00	—
ClarifyMT: action accuracy	360	77.22	61.67	79.72
ClarifyMT: leading-action accuracy	360	76.94	61.39	79.44
ClarifyMT: Answer	120	70.00	75.83	78.33
ClarifyMT: Clarify	240	80.83	54.58	80.42
ClarifyMT: balanced accuracy	—	75.42	65.21	79.38
Precise feedback	60	70.00	75.00	58.33
Refusal feedback	60	70.00	76.67	98.33
Contradictory feedback	60	76.67	56.67	83.33
Factually-Wrong feedback	60	73.33	30.00	83.33
Off-Focus feedback	60	93.33	71.67	81.67
Partial-Vague feedback	60	80.00	60.00	73.33

Source links and context type. Table 22 reports both released FaithEval metrics and all context settings. The primary ACC-N contrast, full minus no-link, is -3.23 points, with three full-only successes, seven no-link-only successes and 114 ties. Two of the latter seven are unavailable full-method outputs under the common denominator rule. The full-minus-two-stage contrast is 4.03 points, with nine full-only successes, four two-stage-only successes and 111 ties. For ACC-S, these contrasts are -4.03 points and 1.61 points, respectively. An additional exploratory ACC-N contrast between no-link and two-stage is 7.26 points; it is separate from the prespecified source-link contrast.

Both condition variants attain 52/61 successes on unanswerable tasks, compared with 44/61 for two-stage analysis, a difference of 13.11 points. The net full-versus-no-link ACC-N difference arises in the counterfactual group. These observations distinguish the response organization retained by both condition variants from the explicit traceability of their source links. The source-link contrast does not establish an incremental accuracy benefit in this configuration; traceability is a property of the representation, and audit efficiency or semantic link accuracy is not measured here.

Table 22: Source-link comparison on FaithEval (%). All rows use the same selected tasks across the three methods.

Criterion	*N*	Full	No-link	Two-stage
Overall ACC-N	124	79.84	83.06	75.81
Overall ACC-S	124	77.42	81.45	75.81
Counterfactual ACC-N	24	58.33	75.00	62.50
Inconsistent ACC-N	39	84.62	84.62	89.74
Unanswerable ACC-N	61	85.25	85.25	72.13
Counterfactual ACC-S	24	58.33	75.00	62.50
Inconsistent ACC-S	39	84.62	84.62	89.74
Unanswerable ACC-S	61	80.33	81.97	72.13

Authorization and answer content. Qualitative comparisons illustrate why an action label and its factual realization are separate evaluation objects. In a ClarifyMT task asking to continue an earlier analysis, the user delegates which analysis to choose. ClarifyAgent selects a quarterly-sales analysis and supplies numerical findings absent from the public dialogue; ResearchSpec requests the earlier analysis or notes. The released action criterion favors answering in this user category, so it scores the former action as correct and the latter as incorrect. Conversely, when the user delegates a reasonable interpretation of "Delta," ResearchSpec asks for an interpretation while both comparators choose one. These saved examples illustrate the distinction between permission to choose and evidence for a factual assertion, together with cases where permitted choice can suffice. They do not alter the released scores or constitute a blinded evaluation of content quality or an authorization-field ablation.

8.16 Computation and Budget-Targeted Reasoning

Accounting. We distinguish solver calls, candidate assessment, user simulation and external evaluation. Solver costs include condition construction and answer generation, or the corresponding

baseline stages; reported completion tokens already include reasoning tokens. Dollar values apply the recorded rates to token usage and represent inference-cost estimates, rather than provider invoices. Reused outputs carry their original logical inference costs, even when their reuse requires no new calls. External evaluation is excluded from all method inference-cost comparisons. Where a request has no reported usage, its conservative reserved estimate is retained in the cost total and identified separately. Request-time aggregates include scheduling and transport and are not treated as user-perceived latency.

18

Table 23: Static inference costs on the comparison cohorts. Costs are US dollars for the entire cohort; input and output counts are solver tokens. The FaithEval full and no-link totals include unknown-usage estimates as specified below.

Cohort / method	Input	Output	Solver	Jev
MiP (200) / ResearchSpec	555,236	1,373,309	1.814542	0.025343
MiP (200) / Two-stage	77,818	295,584	0.378046	—
ClarifyMT (360) / ResearchSpec	1,288,686	3,285,977	4.329778	0.052550
ClarifyMT (360) / Two-stage	222,871	581,200	0.764301	—
ClarifyMT (360) / ClarifyAgent	782,267	1,261,046	1.747935	—
FaithEval (124) / Full	529,503	1,037,214	1.659382	0.018613
FaithEval (124) / No-link	508,381	966,803	1.355312	—
FaithEval (124) / Two-stage	191,837	102,290	0.180299	—

The solver-cost ratios of ResearchSpec to two-stage analysis are 4.80 on MiP and 5.66 on ClarifyMT; its ratio to ClarifyAgent is 2.48. Thus, these are comparisons under common model configurations with observed computation reported, rather than matched realized budgets. The FaithEval full and no-link solver estimates include six and one unknown-usage attempts, respectively, amounting to $0.255874 and $0.042635; their displayed token counts cover reported usage only. For single-turn tasks, omitting Jev leaves the candidate unchanged and removes its separately reported assessment cost.

Budget-targeted free-form refinement. On the fixed 20-pair MiP cohort, refinement begins from the two-stage answer and alternates an ordinary-prose critique with an answer revision. Both calls receive the original task; the critique checks rather than presumes an error, and the revision can preserve a sound answer. No ResearchSpec conditions, full-method answers or external scores enter refinement. For each task, the target is its recorded full-method solver cost. Refinement stops when the accumulated solver estimate, including the initial two-stage calls, reaches that target or six critique–revision cycles have completed. The final available answer is evaluated without selecting among candidates by correctness; a task whose initial two-stage cost already reaches the target retains its initial answer.

Table 24: Budget-targeted reasoning on the same 20 MiP source pairs. Percentages use all 40 tasks or all 20 pairs, as indicated. Refinement includes its initial two-stage prefix.

Criterion	ResearchSpec	Two-stage	Refinement
Joint pair success (20 pairs)	60.00	70.00	55.00
Task-appropriate success (40 tasks)	80.00	85.00	77.50
Original answer accuracy (20 tasks)	90.00	100.00	100.00
Missing-premise abstention (20 tasks)	70.00	70.00	55.00
Solver cost (USD)	0.346019	0.067702	0.309299
Jev cost (USD)	0.004926	—	—

Refinement uses 89.39% of the full method's aggregate solver expenditure. Twenty-one of 40 tasks reach their individual targets; the other nineteen reach the six-cycle limit first. The ResearchSpec-minus-refinement joint-success difference is 5.00 points, with seven ResearchSpec-only successful pairs, six refinement-only successful pairs and seven ties. Refinement minus the initial two-stage baseline is -15.00 points. These results describe one refinement allocation on 20 source pairs with unequal realized expenditure; they do not establish superiority at matched computation.

Inspection of all seven changed missing-premise classifications separates answer changes from scoring boundaries. Two revisions introduce explicit insufficiency findings, one removes an insufficiency qualification, and four retain such findings despite a changed final classification. Of the latter four, two are affected by the released scorer's last-two-paragraph truncation and two receive final classifications inconsistent with its paragraph-level detection. All official scores remain unchanged. The saved auxiliary indicator of having identified insufficiency at least once is 18/20 for ResearchSpec, 15/20 for two-stage and 16/20 for refinement; it is an exploratory scoring diagnostic, not a replacement endpoint or independent content-quality judgment.

8.17 Condition Maintenance and Historical Feedback

Incremental maintenance versus reconstruction. The 55-task comparison shares the exact first-turn conditions, candidate, assessment and delivered user reply across the two arms. From the next update onward, reconstruction receives an empty condition collection and rebuilds conditions from the complete original dialogue. Incremental maintenance receives and updates the existing conditions. Both retain raw history, source locations, authorization semantics and the same generation protocol, with historical assessment feedback disabled. Three tasks terminate before the intervention and remain unchanged in both arms; 52 continue along their respective paths. The comparison concerns the maintenance strategy as a whole, with later simulated replies allowed to differ in response to different questions.

Table 25 reports all scientific-dialogue metrics for this comparison and the separate historical-feedback comparison. The primary MC difference for maintenance is 1.82 points, consisting of one incremental-only success and 54 ties. Equal CGRR and FRR point estimates are reported as observed outcomes, rather than a test of equivalence. The single MC-discordant task follows different subsequent clarification paths and does not isolate forgetting of identical delivered information.

Table 25: Complete scientific-dialogue component results (%). The maintenance comparison uses 55 tasks (29 for DR), and the feedback comparison uses 100 tasks (52 for DR). These are separate cohorts and interventions.

	Condition maintenance		**Historical feedback**	
Metric	**Incremental**	**Reconstruction**	**Enabled**	**Masked**
MC	94.55	92.73	87.00	87.00
CGRR	58.18	58.18	58.00	51.00
FRR	60.00	60.00	61.00	54.00
SRR	1.82	1.82	3.00	3.00
IC	70.91	65.45	69.00	69.00
CR	84.31	79.81	81.53	76.42
CP	74.36	75.33	68.06	66.20
DR	27.59	31.03	38.46	28.85

Resource use in maintenance. Including shared prefixes, solver cost is $1.927139 for incremental maintenance and $2.600410 for reconstruction, a 25.89% reduction. User-simulation costs are $0.056805 and $0.047506, and assessment costs are $0.047571 and $0.040879, respectively. Their combined inference costs are $2.031515 and $2.688795, a 24.45% reduction; external evaluation is excluded. Condition-update costs are $1.394231 and $2.114244 (34.06% lower with incremental maintenance). Incremental maintenance makes 183 condition calls versus 166, and uses 704,385 input tokens versus 435,235, but 985,763 output tokens versus 1,653,061. Mean questions per selected task are 2.327 and 2.018. The observed reduction therefore lies in condition-generation output, rather than fewer questions or shorter input histories.

Historical report feedback. In the separate 100-task comparison, both arms retain condition updates, Jev calls and publication of one unchanged candidate per turn. The intervention masks historical reports from subsequent candidate-generation requests. The full and masked arms generate their initial conditions independently from identical public inputs; their later dialogue trajectories may also diverge. The CGRR difference is 7.00 points, with sixteen full-only successes, nine masked-only successes and 75 ties. The descriptive CGRR results are 43.75% versus 37.50% on 48 disambiguation tasks and 71.15% versus 63.46% on 52 inconsistency tasks. Mean questions are 2.53 versus 2.32, so this comparison does not hold the realized dialogue length fixed.

Historical feedback enters 56 subsequent generation requests across 34 full-method tasks. Forty-six requests follow semantic FAIL assessments; ten follow the deterministic eligibility check for clarification targets. The masked arm receives no historical reports. This intervention consequently evaluates the complete historical-report pathway, rather than the semantic assessor alone. Among the 34 full-method tasks receiving feedback, CGRR has four full-only successes, two masked-only successes and 28 ties; the remaining 66 tasks have twelve, seven and 47, respectively. These post-hoc groups describe observed exposure and are not randomized feedback subgroups.

8.18 ASSESSMENT DIAGNOSTICS

Objects and coverage. The diagnostic analysis joins saved, candidate-bound semantic assessments to the existing official outcomes of 200 MiP, 360 ClarifyMT and 4,992 FaithEval tasks, totaling 5,552 planned tasks. Each published candidate appears once. The aggregate semantic outcome is FAIL if any component is FAIL, otherwise UNKNOWN if any is UNKNOWN, and PASS otherwise. We keep this outcome separate from deterministic checks on action eligibility. Unavailable candidate-bound assessments form a separate category, rather than being assigned FAIL or UNKNOWN. The corresponding task endpoints are MiP task-appropriate success, ClarifyMT action accuracy and FaithEval ACC-N.

Table 26: Task success conditional on semantic assessment. Cells report successful tasks / assessed tasks; unavailable assessments remain explicit. These conditional counts do not replace the full-population denominators.

Benchmark	PASS	FAIL	UNKNOWN	Unavailable
MiP	162/184	11/14	0/0	2
ClarifyMT	221/272	56/84	1/1	3
FaithEval (ACC-N)	3,136/3,654	861/1,262	34/35	41

Task success is 88.04% versus 78.57% for PASS and FAIL on MiP, 81.25% versus 66.67% on ClarifyMT, and 85.82% versus 68.23% on FaithEval. An exploratory stratification within FaithEval gives the same ordering in each context type (Table 27). The aggregate association is therefore not explained solely by mixing those context types. This remains an association between two different assessment targets; within-category task difficulty and response characteristics may also contribute. In particular, UNKNOWN is not interpreted as task failure.

Table 27: Exploratory within-context FaithEval ACC-N diagnostics. Each cell reports successful / assessed tasks and the corresponding percentage.

Context	PASS	FAIL
Counterfactual	410/503 (81.51%)	321/475 (67.58%)
Inconsistent	1,027/1,072 (95.80%)	368/418 (88.04%)
Unanswerable	1,699/2,079 (81.72%)	172/369 (46.61%)

Agreement with task outcomes. Restricting this diagnostic to tasks with both a semantic assessment and a valid official outcome gives 198 MiP, 357 ClarifyMT and 4,951 FaithEval tasks. Non-PASS assessments identify 3/25 (12.00%), 28/79 (35.44%) and 402/920 (43.70%) official non-successes, respectively. They also occur for 11/173 (6.36%), 57/278 (20.50%) and 895/4,031 (22.20%) official successes. Among non-PASS assessments, the corresponding proportions of official non-success are 3/14 (21.43%), 28/85 (32.94%) and 402/1,297 (30.99%). These conditional statistics describe diagnostic correspondence, not semantic judgment accuracy against human labels. The assessed candidate is unchanged in these single-turn tasks, so the association does not estimate an effect of assessment on the published answer.

8.19 PROMPT OVERVIEW AND SELECTED EXCERPTS

These selected excerpts omit other clauses; line wrapping is typographical. The accompanying `prompt_templates.md` preserves complete instructions, message assembly, assessment templates, comparison prompts and attribution. Appendices 8.4–8.7 specify schemas, assessments, stage interfaces and model settings.

Necessity and authorization. The condition instruction distinguishes required information from permissible freedom:

```
An absent detail is not automatically critical. Explain why completing
↪ the task would require an unsupported commitment, rather than merely
↪ saying different settings change results. Ordinary configuration
↪ choices permitted by the task are optional with critical=false. Do
↪ not require exact reproduction unless publicly requested. Authorized
↪ values require explicit user permission, with its scope retained in
↪ value/reason; they are not confirmed user facts.
```

Incremental updates. The same instruction preserves established constraints while incorporating new information:

```
Update only affected conditions; omitted entries persist. Delivered
↪ replies can resolve the named gap without erasing unrelated
↪ constraints. Do not resolve conflicts merely by newest-wins.
```

Response generation. The single-turn instruction treats generated conditions as fallible and retains access to the original sources:

```
Use operation/content/value/state to reason from the task's dependencies.
↪ Declarations are fallible: check the original sources for omitted
↪ conditions and competing evidence. Respect all relevant constraints
↪ without repeating the ledger in the answer.
```

The scientific-dialogue instruction instead permits a focused question or a complete specification:

```
Ask at most one clarification question for an important
↪ missing/conflicting condition that has not already been answered; do
↪ not include an answer, repair proposal or extra question. For final,
↪ write the complete requested specification, preserving original
↪ requirements and confirmed replies.
```

When the clarification allowance is exhausted, the instruction requests the best current final specification with remaining uncertainty explicit. Task-specific messages and generated records enter as data; hidden references and evaluation labels are excluded.

Assessment. The candidate-support question explicitly separates supported insufficiency findings from unsupported conclusions:

```
Check the exact candidate, including negative/insufficient conclusions. A
↪ supported negative finding is PASS, not FAIL. Conditions are
↪ fallible. Consider competing supported answers and relevant
↪ qualifiers; support for one option alone does not establish a
↪ uniquely correct or best answer.
```

Appendix 8.5 specifies the other assessment questions and outcome aggregation.

Comparison variants. The companion file includes the exact source-link edits, reconstruction and feedback input changes, and full two-stage, refinement and ClarifyAgent prompts with attribution. Their evaluation protocols appear in Appendices 8.14, 8.16 and 8.17.