LLM-Guided Planning for Multi-hop Reasoning over Multimodal Nuclear Regulatory Documents

Mingyu Jeon [*1] **Bokyeong Kim** [*2 3] **Suwan Cho** [1] **Jae Young Suh** [1] **Yonggyun Yu** [2 3]

Abstract

Reviewing nuclear regulatory documents requires multi-hop reasoning across tens of thousands of pages, where judgments depend on evidence assembled across multiple chapters. We frame this task as planning: an LLM-based agent observes the evidence collected so far, picks the next document fragment to inspect, and stops when the evidence is sufficient. The agent operates over a vectorless document tree using browse, read, and search tools, and maintains a dynamic knowledge graph (KG) as state. On a 200-question benchmark over NuScale Final Safety Analysis Report (FSAR) documents, the system reaches **81.5%** accuracy with a RAGAS Faithfulness of **0.93**. The dominant performance factor is planning: against PageIndex, which uses the same document tree without state-conditioned action selection, the gap is **+38.0pp** (43.5% to 81.5%, $p < 0.001$). The system also outperforms LightRAG (73.0%, $p < 0.05$), HippoRAG (70.5%, $p < 0.01$), and GraphRAG (49.5%, $p < 0.001$), and matches RAPTOR (75.5%, $p = 0.11$) without offline indexing. Edge inference adds 2.8× cost without raising accuracy; we retain it as a traceability module. Of 7,391 inferred edges, 3 VIOLATES edges (0.04%) flag scope boundaries (Q058) and partial conformance (Q176) as typed annotations that a human reviewer can audit.

1. Introduction

LLM-based agents have advanced rapidly in general-purpose domains through paradigms such as ReAct (Yao et al., 2023b), Toolformer (Schick et al., 2023), and Self-RAG (Asai et al., 2024). Safety-critical regulatory domains impose constraints these paradigms do not address. Osprey (Hellert et al., 2026) identifies the absence of action pre-visibility in existing frameworks. Lee (Lee, 2025) analyzes the conflict between LLM opacity and the quality assurance traceability requirements of 10 CFR (Code of Federal Regulations) 50 Appendix B. Nuclear regulatory document review, which examines Final Safety Analysis Reports (FSARs) and renders conformance judgments to enable plant licensing, is where these constraints bind most tightly.

FSAR review has three properties that distinguish it from single-shot question answering. First, judgments depend on evidence accumulated from multiple chapters: answering "Does the Emergency Core Cooling System (ECCS) design satisfy 10 CFR 50.46(b)?" requires specifications from Chapter 5 cross-referenced against requirements in Chapter 1, with the verdict synthesized from both. Second, evidence is multimodal: specification tables and engineering drawings co-determine answers alongside the prose. Third, the review process includes a sufficiency judgment in which the reviewer decides when collected evidence is enough to render a verdict. Existing RAG (Retrieval-Augmented Generation) methods address none of these. Chunking severs cross-references (Gao et al., 2023). Graph-based methods (GraphRAG (Edge et al., 2024), LightRAG (Guo et al., 2024)) build static global graphs and lack an iterative judgment loop. All require offline indexing that is incompatible with continuously revised FSARs.

Our key observation is that regulatory review is a **planning problem**: given a goal (regulatory judgment), state (evidence collected), and actions (document navigation), the reviewer must decide what to examine next. This is a sequential decision-making process irreducible to single-shot retrieval. We construct the document as a **text-based environment** with browse/read/search actions, enabling a closed planning loop, where 33% of queries terminate early and 67% use the full 4-hop budget.

We make four contributions. (1) **Document as environment for planning**: a state-conditioned planning loop over a vectorless document tree, isolated against PageIndex (same environment, no planning) by a +38.0pp accuracy gap

[*]Equal contribution [1]MODULABS, Seoul, Republic of Korea [2]Korea Atomic Energy Research Institute, Daejeon, Republic of Korea [3]University of Science & Technology, Daejeon, Republic of Korea. Correspondence to: Yonggyun Yu <ygyu@kaeri.re.kr>.

Proceedings of the 43rd International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

($p < 0.001$). (2) **Multimodal evidence handling integrated into the planning loop**: vision processing applied at the answer step yields +18pp over RAPTOR on table-only questions while keeping all intermediate operations text-only. (3) A **200-question nuclear regulatory benchmark** along three orthogonal axes (reasoning type, evidence complexity, modality) that jointly determine the regulatory review process. (4) **Traceability analysis via VIOLATES case study**: post-retrieval edge inference adds 2.8× cost with no accuracy gain, but produces auditable reasoning paths. Among 7,391 inferred edges, only 3 (0.04%) are VIOLATES, identifying scope exclusions (Q058) and partial conformance (Q176) as typed annotations that natural-language answers cannot represent structurally. This satisfies 10 CFR 50 Appendix B traceability requirements.

2. Related Work

RAG and graph-based retrieval. Standard RAG (Lewis et al., 2020) fragments context through chunking. GraphRAG (Edge et al., 2024) and LightRAG (Guo et al., 2024) construct global knowledge graphs but require expensive pre-indexing and generic edge schemas. RAPTOR (Sarthi et al., 2024) builds recursive abstractive trees; HippoRAG (Gutiérrez et al., 2024) uses hippocampal memory models. All perform single-shot retrieval without iterative evidence accumulation.

Agentic information retrieval. Iterative retrieval methods such as IRCoT, FLARE, and Iter-RetGen interleave retrieval with chain-of-thought generation but lack persistent state and dynamic termination. Self-RAG (Asai et al., 2024) inserts reflection tokens for adaptive retrieval within a single pass. PRISM (Nahid & Rafiei, 2025) separates precision and recall through an iterative Selector–Adder agent loop. APEX-Searcher (Chen et al., 2026) combines reinforcement learning (RL) with supervised fine-tuning (SFT) for planning but requires training. For document navigation, ReadAgent (Lee et al., 2024) uses gist memories, DocAgent (Sun et al., 2025) extracts XML outlines, and BookRAG (Wang et al., 2025) routes queries through hierarchical indices. PageIndex (Zhang & Tang, 2025) builds vectorless trees but operates as a single-pass tool. Our work departs by constructing the document as a **scalable text-based environment** with persistent KG state, a planning loop, and dynamic termination, in a **training-free** setting.

Planning, world models, and KG-RAG. ReAct (Yao et al., 2023b), Tree of Thoughts (Yao et al., 2023a), and LATS (Zhou et al., 2024) establish LLM-based planning in PDDL/robotic/web environments through tree search with reflection; we extend this paradigm to **information environments** over structured documents. GWM (Feng et al., 2025) uses graph-structured state with message-passing; we similarly employ a Dynamic Sub-KG but generate explicit relational edges through LLM inference rather than embeddings. Our edge ontology draws on SysML traceability (Friedenthal et al., 2014), argumentation mining (Peldszus & Stede, 2013; Cabrio & Villata, 2012), causal KG (Hassanzadeh et al., 2019), and prerequisite learning (Pan et al., 2017).

Nuclear NLP and evaluation. NuclearQA (Acharya et al., 2023) and NukeBERT (Jain et al., 2020) address single-hop factual extraction. Our benchmark is the first to target multi-hop, multimodal, cross-chapter regulatory judgment. We adopt dual evaluation: RAGAS (Es et al., 2024) for grounding quality and LLM-as-Judge (Zheng et al., 2023) with three-evaluator majority vote.

3. Method

3.1. Problem Formulation: Regulatory Review as Planning

We formulate regulatory document exploration as a planning problem $\langle \mathcal{S}, \mathcal{A}, f_{\text{tr}}, \phi \rangle$ with a single agent acting over a structured information environment:

- **State** $s_t \in \mathcal{S}$: the Dynamic Sub-Knowledge Graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$ collected through hop t, representing the agent's current evidence and inferred relationships (§3.3).
- **Action** $a_t \in \mathcal{A}$: a tool invocation from $\{\texttt{browse}(d, v), \texttt{read}(d, v), \texttt{search}(\kappa)\}$ over documents d, tree nodes v, and keywords κ (§3.4).
- **Transition** $f_{\text{tr}}(s_t, a_t) \rightarrow s_{t+1}$: tool execution followed by node integration and (optionally) edge inference, producing the updated KG (§3.5).
- **Goal test** $\phi(s_t, q) \in \{0, 1\}$: an LLM-judged sufficiency check on whether $\mathcal{G}_t$ contains enough evidence to answer query q; termination occurs when $\phi = 1$ or $t = T_{\max}$.

Unlike plan-then-execute frameworks, the agent observes s_t, selects a_t conditioned on accumulated state, and immediately incorporates environment feedback before selecting a_{t+1}. This state-conditioned planning structure is the central mechanism we isolate empirically against PageIndex (§5.2).

3.2. Environment: Vector-Free Multimodal Document Tree

The overall architecture is shown in Figure 1. The planning loop described in this section (state estimation, action

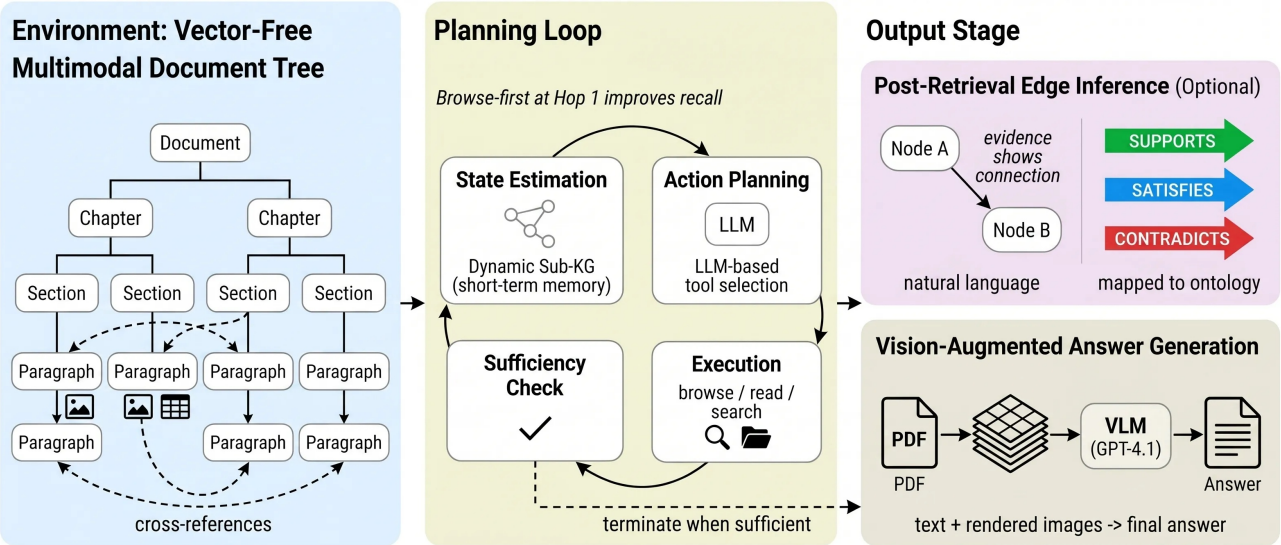

Figure 1. Overall architecture. The vectorless document tree (left) serves as the environment. The planning loop (center) iterates through state estimation, action planning, execution, and sufficiency checking. Post-retrieval edge inference and vision-augmented answer generation (right) are applied at the output stage.

selection, dynamic termination) is architecturally domain-agnostic and applies to any hierarchically structured document corpus. Regulatory documents such as FSARs present the conditions under which this approach is most advantageous over conventional RAG: deep hierarchical structure that chunking destroys, dense cross-references between sections and figures, multimodal evidence (specification tables, engineering drawings) that co-determines answers, and a review process that inherently requires multi-hop evidence gathering with sufficiency judgment. The domain-specific component is the edge ontology (Section 3.5), which encodes regulatory reasoning relations (SATISFIES, VIOLATES); the rest of the architecture transfers directly to other structured document domains.

The environment is represented as a JSON hierarchical tree organized into chapter → section → paragraph nodes, preserving the native structure of regulatory documents without any chunking or embedding. To support multimodal reasoning, the system parses the LIST OF FIGURES/TABLES and detects in-text references such as "Figure 5.1-1," attaching figure and table metadata to the corresponding nodes via a `references` field. This directly addresses the "figure on different page" problem, wherein the referencing text and the actual diagram reside on different pages of the PDF.

Rather than relying on dense vector retrieval, the system adopts a vector-free design using BM25Okapi keyword search over the full document tree. Section titles receive a $3\times$ weight boost, and document-length normalization naturally promotes short, focused leaf nodes to higher rankings. At the scale evaluated in this work, the tree spans Ch.01 with 866 nodes (34 figures, 19 tables) and Ch.05 with 26 nodes (29 figures, 30 tables).

3.3. State (Short-Term Memory): Dynamic Sub-KG and Two-Tier Edge Ontology

The agent state at timestep t is defined as a dynamic knowledge graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$. The node set $\mathcal{V}_t$ comprises document sections (evidence nodes) collected through exploration along with their associated multimodal references. The edge set $\mathcal{E}_t$ is governed by a domain-specific two-tier ontology, with edges retained only when confidence is ≥ 0.4 (empirically set).

The ontology is summarized in Table 1. Tier 1 consists of *structural* edges that organize the exploration trajectory:

An example KG with edge distribution is shown in Figure 3. Empirically, structural edges (REFERENCES, SPECIFIES) dominate in single-hop factual queries by forming the exploration path, while semantic edges (SATISFIES, SUPPORTS) emerge in composite multi-hop judgment queries to support regulatory compliance synthesis. In correct answers relative to incorrect ones, SUPPORTS appears $+6.8$ percentage points more frequently and SATISFIES $+3.2$ percentage points more frequently.

3.4. Action Planning: LLM-Based Tool Selection

Rather than precomputing a full retrieval plan, the system performs closed-loop planning. At each hop, the agent observes the current KG state $\mathcal{G}_t$ and decides action a_t, with environment feedback (retrieved results) immediately incorporated into the subsequent plan. This constitutes a state-based iterative decision-making structure, distinct from the

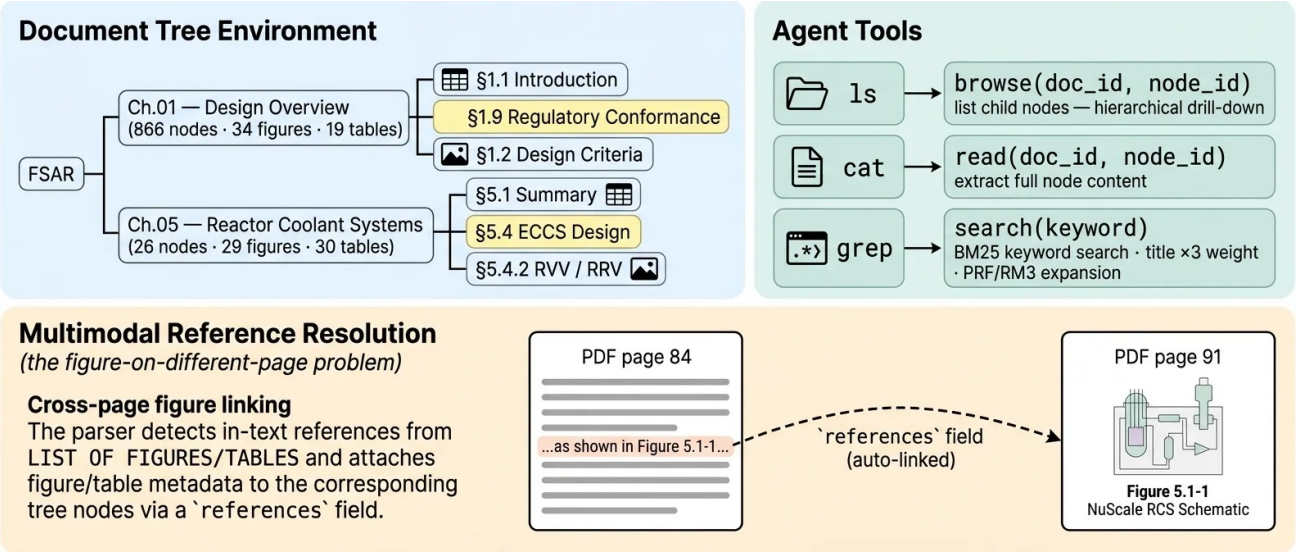

Figure 2. Document tree environment (left), three agent tools (right), and multimodal reference resolution linking in-text references to actual PDF pages (bottom).

Table 1. Two-tier regulatory edge ontology.

Edge	Semantics & Grounding
Tier 1: Structural	
REFERENCES	Cross-reference (A cites B)
SPECIFIES	Higher-level elaborated by lower-level (Friedenthal et al., 2014)
Tier 2: Semantic	
SATISFIES / VIOLATES	Design meets/fails requirement (Friedenthal et al., 2014; Zhong et al., 2012)
SUPPORTS / CONTRADICTS	Evidence corroborates/refutes claim (Peldszus & Stede, 2013; Cabrio & Villata, 2012)
LEADS_TO	Causal chain (Hassanzadeh et al., 2019)
IS_PREREQUISITE_OF	Concept dependency (Pan et al., 2017)

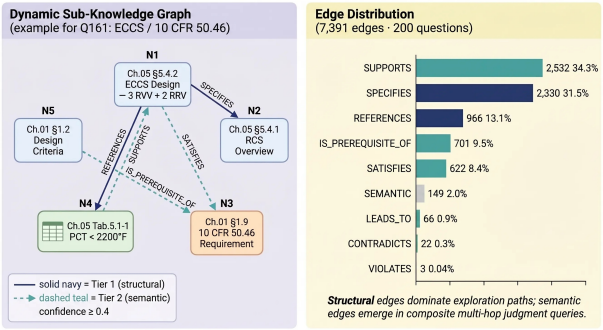

Figure 3. An example Dynamic Sub-Knowledge Graph showing five evidence nodes connected by structural (Tier 1) and semantic (Tier 2) edges. The edge distribution summary (right) shows the prevalence of each edge type across 7,391 edges from 200 questions.

plan-then-execute separation of APEX-Searcher (Chen et al., 2026) and the token-level reactive retrieval of Self-RAG (Asai et al., 2024). Unlike passive embedding-similarity retrieval in conventional RAG, the LLM actively evaluates the current state and plans the next action.

The agent has access to three tools (Figure 2) that mirror filesystem operations: **browse** lists the child nodes of a tree node (analogous to `ls`), **read** extracts the full content of a specific node (analogous to `cat`), and **search** performs BM25-ranked keyword search across all documents (analogous to `grep`).

A *browse-first* pattern is enforced at Hop 1, where the document structure (table of contents, ToC) is automatically injected so that the agent obtains a global map before searching. This intervention improved single-evidence Context Recall from 0.45 to 0.89. To address vocabulary mismatch, Pseudo-Relevance Feedback (PRF, RM3) automatically expands queries using the top-3 retrieved results at zero additional LLM cost. The agent also maintains a search history to prevent duplicate keyword queries across hops.

Dynamic termination is implemented as a plan sufficiency check beginning at Hop 2: before each hop, the LLM judges whether the current KG already contains sufficient evidence to answer the query. If so, the agent terminates early. This functions as a goal test within the planning loop, automatically calibrating exploration depth to query complexity. Across 200 questions, 33% of queries terminate early at 1–3 hops (mean 3.4, maximum 4).

3.5. Post-Retrieval Edge Inference (Optional Component)

Edge inference makes explicit the relationships among collected evidence nodes and is performed concurrently with the state transition $f_{\text{tr}}(\mathcal{G}_t, a_t) \rightarrow \mathcal{G}_{t+1}$. The accuracy impact of this component is evaluated in Section 6.2; we offer it as an optional module for use cases requiring traceability.

Inference proceeds in two stages. In Stage 1 (Description), the LLM produces a single natural-language sentence describing the relationship between two nodes, without imposing any classification pressure. This follows the free-form relation extraction approach of LightRAG (Guo et al., 2024); for example: "The ECCS design of 3 RVV + 2 RRV is configured to meet the acceptance criteria of 10 CFR 50.46." In Stage 2 (Ontology Mapping), the free-form description is mapped onto the regulatory domain ontology (SATISFIES, VIOLATES, etc.). When no mapping is applicable, the relationship is preserved as SEMANTIC, ensuring no relational information is discarded.

Verification is integrated into the evidence-gathering loop at every hop rather than applied as a post-processing step: immediately after new evidence is retrieved (planning), relationship inference is performed (verification), and the resulting enriched KG state informs the plan sufficiency judgment for the subsequent hop. This design differs from embedding-based implicit relations used in GraphRAG and GWM (Feng et al., 2025); by grounding relationships in explicit LLM-generated natural-language descriptions, the inference results remain human-inspectable.

3.6. Vision-Augmented Final Answer Generation

Multimodal processing is applied exclusively at the final answer generation step, with all intermediate operations (search, plan, infer) remaining text-only. This cost-efficient design avoids the expense of vision API calls during iterative exploration while still enabling visually grounded final answers.

The implementation proceeds in three steps: (1) all Figure/Table references across KG nodes are collected; (2) the corresponding PDF pages are rendered to JPEG using PyMuPDF; and (3) the full text KG context together with the rendered images is passed to the GPT-4.1 vision-language model (VLM) API. For tables specifically, PyMuPDF's `find_tables()` function extracts row and column structure directly as structured text, making VLM image processing unnecessary. This approach achieves 86.0% accuracy on table-only questions, compared to 68.0% for RAPTOR (+18 percentage points).

The complete pipeline is summarized in Algorithm 1.

Algorithm 1 SUBKG-AGENT Planning Pipeline

Input: Query q, documents $\mathcal{D}$, max hops T
Initialize $\mathcal{G}_0 \leftarrow \emptyset$; inject ToC at hop 0
for $t = 0$ **to** $T - 1$ **do**
  $a_t \leftarrow \text{LLM}(q, \mathcal{G}_t, \text{ToC})$ {plan actions}
  Execute a_t via `browse`/`read`/`search`
  $\mathcal{G}_{t+1} \leftarrow f_{\text{tr}}(\mathcal{G}_t, a_t)$ {integrate + infer edges (optional)}
  if $t \geq 1$ **and** SufficientEvidence$(q, \mathcal{G}_{t+1})$ **then**
    break
  end if
end for
return VisionAugmentedAnswer$(q, \mathcal{G}_{\text{final}})$

Table 2. Existing document QA benchmarks. NuScale-MQA is the first to combine all five properties.

Benchmark	M-hop	Tab.	Fig.	X-doc	Judg.
NuclearQA (Acharya et al., 2023)	–	–	–	–	–
FDARxBench (Xiong et al., 2026)	✓	✓	Part.	–	✓
MMLongBench (Ma et al., 2024)	✓	✓	✓	–	–
M3DocRAG (Cho et al., 2024)	✓	✓	✓	✓	–
DesignQA (Doris et al., 2024)	–	✓	✓	✓	Part.
SEC-QA (Lai et al., 2024)	✓	✓	–	✓	–
TAT-QA (Zhu et al., 2021)	✓	✓	–	–	–
Ours	✓	✓	✓	✓	✓

4. Benchmark: Nuclear Regulatory Multi-hop QA

No existing benchmark combines nuclear regulatory domain, multi-hop reasoning, multimodal evidence (tables and engineering drawings), and regulatory judgment. We survey the most relevant benchmarks along these dimensions in Table 2. FDARxBench (Xiong et al., 2026) is the most similar (regulatory documents with judgment) but covers only single documents without engineering drawings. MMLongBench-Doc (Ma et al., 2024) and M3DocRAG (Cho et al., 2024) support multimodal understanding but lack regulatory judgment. DesignQA (Doris et al., 2024) addresses engineering compliance but not multi-hop reasoning. NuclearQA (Acharya et al., 2023) targets single-hop factual extraction only.

4.1. Design and Composition

We construct 200 questions over NuScale FSAR Chapter 01 (352 pages) and Chapter 05 (160 pages), organized along three orthogonal axes. **Reasoning type**: factual (70), comparative (65), judgment (65). **Evidence complexity**: single-evidence (50), multi-evidence (75), cross-document (75). **Modality**: text-only (80), table-only (50), image-only (30), composite (40). The benchmark contains 357 ground-truth evidence items (152 text, 125 table, 80 figure). This distribution is summarized in Table 3; the judgment $\times$ cross-

Table 3. Question distribution in NuScale-MQA (200 total).

	Single	Multi	Cross-doc	Total
Factual (70)	30	25	15	70
Comparative (65)	15	25	25	65
Judgment (65)	5	25	**35**	65
Total	50	75	75	200

document cell (35 questions) is the largest, reflecting the core regulatory review task.

4.2. Dual Evaluation Framework

We adopt dual evaluation: **RAGAS** (Es et al., 2024) for grounding quality (Faithfulness, Answer Relevancy, Context Recall, Factual Correctness) and **LLM-as-Judge** (Zheng et al., 2023) with three independent evaluators, Eval-A (GPT-4-turbo), Eval-B (GPT-4o), and Eval-C (Claude Sonnet 4.5), combined via majority vote. The RAGAS–Judge agreement rate is 66.2%, with the 34% disagreement providing complementary information: RAGAS measures grounding fidelity while the Judge captures practical correctness.

5. Experiments

5.1. Setup and Baselines

All methods use GPT-4.1 for generation with temperature 0 and max_tokens 300. Our agent is configured with max_hops=4 and top_k=2. We compare against RAPTOR (Sarthi et al., 2024) (recursive summarization tree), HippoRAG (Gutiérrez et al., 2024) (hippocampal associative KG), LightRAG (Guo et al., 2024) (dual-level graph + vector DB), GraphRAG (Edge et al., 2024) (community-based local search), and PageIndex (Zhang & Tang, 2025) as an ablation baseline. Full baseline configurations are in Appendix A.

PageIndex is the critical comparison: it operates over the **identical** document tree with the same tools, browse-first ToC injection, BM25 configuration, PRF/RM3 query expansion, search history deduplication, and dynamic termination. The only difference is action selection: our system conditions on accumulated KG state; PageIndex selects actions from the query and immediate results alone. The gap therefore isolates state-conditioned planning.

5.2. Main Results

We report LLM-as-Judge accuracy in Table 4. Our system (planning only) achieves **81.5%** overall. The most important comparison is PageIndex at 43.5%, isolating a +38.0pp planning contribution (McNemar $p < 0.001$). Among external baselines, we significantly outperform HippoRAG ($p < 0.01$), LightRAG ($p < 0.05$), and GraphRAG ($p < 0.001$). The +6.0pp gap over RAPTOR is consistent across all categories but does not reach significance (McNemar $p = 0.11$, $n = 200$); however, our system requires zero pre-indexing cost versus RAPTOR's 44 minutes and $1.4 of offline indexing.

Judgment caveat. 98% of judgment questions have "Yes" as the correct answer (FSARs document compliant designs by definition), so the judgment column has limited discriminative power. Comparative and factual categories provide more meaningful comparisons.

5.3. RAGAS and Efficiency

We report RAGAS metrics in Table 5. Our system achieves Faithfulness of **0.93** and Context Recall of **0.93**, ranking first across all metrics. GraphRAG scores 0.28/0.18, indicating factual loss during community summarization. PageIndex achieves only 0.58 Faithfulness, illustrating the limits of unguided retrieval. The per-reasoning-type breakdown (Table 6) confirms this dominance across all three categories: our system leads both Faithfulness and Context Recall on factual, comparative, and judgment questions. GraphRAG's Context Recall of 0.11 on factual questions confirms that specific numerical values are systematically lost during community summarization.

Efficiency. We report indexing and per-query costs across methods in Table 7. Our system requires 8–20 min and $4.1 for pre-indexing, but higher per-query cost ($0.215, 93s) due to multi-hop exploration. Total cost for 200 questions is ∼$46 vs. ∼$2.3 for RAPTOR, yielding a per-accuracy-point cost of $0.56/%p. Dynamic termination partially mitigates this: 33% of queries terminate early, with costs ranging from $0.03 (1-hop factual) to $0.29 (4-hop cross-document judgment). For regulatory applications, where human reviewers spend hours per question, the $46 total for 200 questions is negligible compared to manual review cost.

6. Analysis

6.1. Component Ablation (10Q)

To isolate each component's contribution, we remove one at a time from the full system on a 10-question subset spanning diverse reasoning types (Table 8). Only the full system achieves 10/10, and each removal fails on a distinct question type: Vision RAG removal drops Q101 (table/comparative) and Q131 (composite/comparative), with Faithfulness falling from 0.96 to 0.83; Edge inference removal drops Q058 (seismic scope boundary) due to missing VIOLATES edge but saves 66% of cost; Browse-first removal drops Q191 (image/judgment/cross) due to lost structural orientation. At $n = 10$, differences are indicative rather than statistically conclusive, motivating the 200Q scale-up

Table 4. LLM-as-Judge accuracy on NuScale-MQA (200 questions). PageIndex table-only and composite results were not collected in the same evaluation run.

Method	Overall	Judg.	Comp.	Fact.	Cross-doc	Table	Compos.
Ours (plan. only)	**81.5**	**92.3**	**80.0**	**72.9**	**84.0**	**86.0**	77.5
Ours (plan.+edges)	81.0	90.8	78.5	74.3	81.3	86.0	**85.0**
RAPTOR	75.5	92.3	72.3	62.9	73.3	68.0	72.5
LightRAG	73.0	92.3	66.1	61.4	76.0	60.0	67.5
HippoRAG	70.5	86.2	63.1	62.9	69.3	64.0	60.0
GraphRAG	49.5	61.5	49.2	38.6	37.3	42.0	47.5
PageIndex	43.5	61.5	44.6	25.7	45.3	—	—

Table 5. RAGAS comparison across all methods.

Method	Faith.	AR	CR	FC
GraphRAG	0.28	0.59	0.18	0.32
PageIndex	0.58	0.77	0.66	0.30
RAPTOR	0.74	0.83	0.77	0.40
HippoRAG	0.76	0.83	0.76	0.37
LightRAG	0.89	0.83	0.88	0.36
Ours	**0.93**	**0.84**	**0.93**	**0.42**

for the edge inference question.

6.2. Ablation: Edge Inference at Scale (200Q)

Since edge inference accounted for 65% of per-query cost in the 10Q ablation, we scale this comparison to all 200 questions (Table 9).

Key finding: Planning is the primary driver. Removing edge inference does not reduce accuracy; it provides traceability (human-readable paths like "Section A SATISFIES Regulation B") but no accuracy gain. Planning mechanisms alone, namely browse-first (CR 0.45→0.89), dynamic termination (33% early), and state-conditioned selection (PageIndex +38.0pp), suffice to outperform all baselines. This parallels APEX-Searcher (Chen et al., 2026) and PRISM (Nahid & Rafiei, 2025) in scope but diverges in method: both require RL/SFT training, while our system achieves equivalent planning in a **training-free** setting.

6.3. Edge Distribution and Case Studies

Across 200 questions, 7,391 edges are generated: SUPPORTS (34.3%), SPECIFIES (31.5%), REFERENCES (13.1%), IS_PREREQUISITE_OF (9.5%), SATISFIES (8.4%), with VIOLATES at only 0.04% (3 instances). Semantic edges correlate with correctness: SUPPORTS +6.8pp and SATISFIES +3.2pp more frequent in correct answers.

Why VIOLATES appears in a certified document. The FSAR has already been certified by the U.S. Nuclear Regulatory Commission (NRC), so the emergence of VIOLATES edges warrants explanation. All three instances capture **scope boundary exclusions** rather than design deficiencies. This reveals a regulatory reasoning capability absent from standard RAG:

- **Q058 (Seismic scope, ×2):** Both VIOLATES edges (confidence 0.85, 0.90) identify that non-safety-related systems (Chilled Water, Condensate Storage) are intentionally placed outside Seismic Category I classification. The FSAR explicitly justifies this: "failure of non-safety systems, structures, and components (SSCs) does not affect safety-related SSCs." The agent marks these scope boundaries in the KG, so that a human reviewer can immediately see which requirements do not apply. Without this marking, such information would require manual cross-referencing across chapters. The distinction matters for regulatory review: "requirement does not apply" and "requirement is not satisfied" carry different licensing consequences, and a representation that does not separate them structurally shifts the burden of distinguishing onto the human reviewer.
- **Q176 (Partial conformance, ×1):** The VIOLATES edge (confidence 0.85) captures that NuScale's integrated Steam Generator (SG) design eliminates the traditional containment bypass problem but introduces a leakage detection limitation: the system cannot distinguish identified from unidentified leakage, resulting in partial conformance with Design-Specific Review Standard (DSRS) 15.6.5. This is the space between full SATISFIES and full VIOLATES, the kind of state where regulatory review typically requires additional engineering analysis or compensatory measures. The agent represents the state as a typed edge with confidence 0.85; the baselines we evaluate in §5.2 produce no equivalent typed annotation.

These cases show that while edge inference does not improve accuracy (Table 9), it provides concrete regulatory value: (1) explicit scope boundary identification, marking where requirements apply and do not apply; (2) partial con-

Table 6. RAGAS by reasoning type across all methods. Our system leads Faithfulness and Context Recall on all three reasoning types.

Type	Metric	Ours	LightRAG	RAPTOR	HippoRAG	PageIndex	GraphRAG
Factual	Faith.	**0.92**	0.91	0.81	0.84	0.58	0.26
Factual	CR	**0.92**	0.86	0.75	0.79	0.57	0.11
Comparative	Faith.	**0.92**	0.88	0.68	0.70	0.54	0.32
Comparative	CR	**0.91**	0.89	0.74	0.71	0.67	0.18
Judgment	Faith.	**0.97**	0.89	0.74	0.74	0.63	0.27
Judgment	CR	**0.96**	0.91	0.82	0.76	0.76	0.25

Table 7. Efficiency comparison across methods. Per-query cost extrapolated from 5-question sample with ±10–15% variance.

	Indexing		Per-query		Total (200Q)		Summary	
Method	Time	Cost	Time	Tokens	Time	Cost	Acc.	\$/%p
Ours	8–20 min	\$4.1	93 s	86K	∼320 min	∼\$46	**81.5%**	\$0.56
RAPTOR	∼44 min	∼\$1.4	1.8 s	2K	∼50 min	∼\$2.3	75.5%	\$0.03
LightRAG	52 min	∼\$7.0	∼5 s	24K	∼69 min	∼\$17	73.0%	\$0.23
HippoRAG	29 min	∼\$4.3	∼3 s	2.6K	∼39 min	∼\$5.4	70.5%	\$0.08
GraphRAG	40 min	∼\$3.3	5.2 s	693	∼57 min	∼\$3.7	49.5%	\$0.07

Table 8. 10-question component ablation. Each removal fails on a distinct question type.

Variant	Judge	Faith.	AR	CR	Time	Cost
full	**10/10**	**0.96**	0.84	0.95	104s	\$0.216
no_vision	8/10	0.83	0.82	0.92	98s	\$0.196
no_edges	9/10	0.93	0.84	**1.00**	**34s**	**\$0.073**
no_browse_first	9/10	0.97	0.79	0.94	91s	\$0.180

Table 9. Edge inference ablation (200 questions).

Metric	Plan.+Edges	Plan. Only
3-Judge Accuracy	81.0%	**81.5%**
Faithfulness	**0.930**	0.897
Context Recall	**0.930**	0.919
Cost / question	\$0.215	**\$0.076**
Time / question	115.3s	**47.5s**

formance that mirrors actual regulatory judgment; and (3) auditable reasoning paths satisfying the traceability requirements of 10 CFR 50 Appendix B. The rarity of VIOLATES (3 of 7,391 edges, 0.04%) itself serves as a quality signal consistent with a certified document. For practical use, these three typed edges identify the locations a human reviewer can focus on without re-reading every natural-language answer.

The 66.2% RAGAS–Judge agreement leaves 67 disagreement cases that decompose into two qualitatively distinct categories. **29 cases (RAGAS Good + Judge X)**: answers grounded in the KG (Faith $\geq$ 0.8, CR $\geq$ 0.8) but penalized for wording differences from the reference. Q019 is representative: reference "the pressurizer volume is 578 ft^3," agent "the pressurizer region volume is 578 ft^3 and the cylindrical section is 487 ft^3," RAGAS Faith 1.00 / CR 1.00, Judge X. These reflect benchmark-level evaluation strictness, not regulatory error. **38 cases (RAGAS Bad + Judge O)**: answers Judge marks correct despite low Context Recall, meaning the agent answered from knowledge not fully captured by the retrieved KG. This is the deployment-relevant category: $38/200 = 19\%$ of correct answers cannot be fully verified by inspecting the KG alone, and a reviewer needing audit-grade reasoning should flag this fraction for separate verification. The two categories motivate dual evaluation: RAGAS measures grounding, Judge measures correctness, and either alone misclassifies cases that matter for safety review.

6.4. Failure Mode Taxonomy

We classify the 38 incorrect answers (Judge X) by RAGAS scores, using the same Faith ≥ 0.8 and CR ≥ 0.8 thresholds as §5.3.

Expression / reasoning (29/38, 76.3%): well-grounded by both metrics, Judge X. The dominant pattern is wording mismatch with the reference, as in Q019 above. The penalty is for added detail, not regulatory error.

Hallucination (2/38, 5.3%): Faith < 0.5. The answer asserts content the KG does not support.

Retrieval miss (1/38, 2.6%): CR < 0.5. Retrieval missed evidence the reference relies on.

Mixed (6/38, 15.8%): intermediate Faith and CR.

Table 10. Accuracy by hop count.

Bucket	Hops	n	Accuracy
Early	1	11	100.0%
Early	2	13	84.6%
Early	3	45	82.2%
Full	4	129	78.3%
Early total	1–3	69	**85.5%**
Full total	4	129	78.3%

Excluding the wording-mismatch category, the deployment-relevant failure rate is $9/200 = 4.5\%$. Hallucination at scale is $2/200 = 1.0\%$. These are the rates a reviewer should expect to verify against external evidence.

6.5. Self-Assessment Calibration

Dynamic termination (§3.4) lets the agent decide when its evidence is sufficient. Whether the early-stopping signal is reliable determines whether early termination introduces a deployment risk. We stratify accuracy by hop count: early-terminated queries (1–3 hops) are the high-confidence subset; full-budget queries (4 hops) are the low-confidence subset.

Early-terminated queries reach 85.5% accuracy; full-budget queries reach 78.3%. Higher accuracy on the high-confidence subset is the calibration signature: the early-stopping signal is reliable. Question difficulty confounds the comparison (simple factual queries terminate at hop 1, cross-document judgment queries use 4 hops), but the ordering rules out the failure mode in which the agent quits too early on hard cases. That mode would invert the ordering.

7. Limitations

System. Our system underperforms RAPTOR on text-only questions (76.2% vs. 80.0%, -3.8pp); adding summary nodes to the tree is a potential improvement. Per-query cost ($0.215, 93 s) is 42× RAPTOR's cost but justified for regulatory use where a human reviewer requires hours per question; the system's 94.3% accuracy on the 35 judgment×cross-document questions, the core regulatory review task, justifies the cost premium where it matters most. The RAPTOR gap ($+6.0$pp) cannot be asserted as a reliable improvement at the current sample size (McNemar $p = 0.11$). A follow-reference tool for directly navigating "see Table 5.1-1"-style pointers remains unimplemented. The max_hops=4 ceiling is used by 67% of queries; whether increasing the budget would improve accuracy on the most complex questions remains an open question.

Benchmark. The benchmark exhibits five structural limitations identified during evaluation: (1) *Factual Correctness ceiling* (∼0.42 across all methods) reflects single-perspective expected answers rather than retrieval failure; (2) *judgment polarity bias*: 98% "Yes" answers, as FSARs document by definition compliant designs; (3) *limited evidence depth*: 66% of questions are effectively 2-hop; (4) *document coverage imbalance*: Ch.01 uses only 19% of its pages; (5) *no external validation*: the benchmark is self-designed, with mitigating factors including the three-axis orthogonal design, 3-evaluator majority voting, and uniform comparison across 5 methods under identical conditions.

Edge ontology coverage. Our eight-relation ontology is chosen for FSAR conformance review and does not cover relations relevant to other regulatory frameworks, such as cross-jurisdictional applicability, temporal validity, conditional applicability, and engineering exception precedents. The three VIOLATES instances we report come from a single FSAR corpus; whether similar typed-edge patterns appear in other regulatory domains is unknown. Deploying the system to a different framework requires domain-expert ontology curation and human oversight.

8. Conclusion

This paper frames multi-hop regulatory document review as a planning problem and instantiates it as an LLM-based agent that operates over a vectorless document tree with a dynamic knowledge graph as state. On a 200-question NuScale FSAR benchmark, the system reaches 81.5% accuracy with RAGAS Faithfulness 0.93, outperforming GraphRAG, HippoRAG, and LightRAG, and matching RAPTOR while eliminating offline indexing cost. The $+38.0$pp gap over PageIndex isolates state-conditioned planning as the primary accuracy driver. A 200-question ablation finds edge inference contributes no accuracy gain at 2.8× cost, but produces typed regulatory relations: 3 VIOLATES edges among 7,391 (0.04%) make scope-bounded inapplicability and partial conformance explicit, the audit form that 10 CFR 50 Appendix B requires. Failure mode analysis shows 29 of 38 errors are wording mismatch rather than regulatory error, leaving a deployment-relevant failure rate of 4.5%; hop-stratified accuracy (85.5% early vs. 78.3% full) confirms that the early-stopping mechanism is well-calibrated. The architecture (document-as-environment, action interface, KG state, dynamic termination) transfers to any domain whose review process requires multi-hop evidence gathering and explicit conformance judgment, with the edge ontology as the domain-specific component.

Impact Statement

The primary goal of this work is to improve the efficiency and accuracy of agent-based machine learning systems. We anticipate no immediate negative societal impacts arising

uniquely from our contributions beyond those generally associated with the deployment of large language model based agents. On the contrary, by reducing the computational cost of multi-hop reasoning over large document collections, our approach can help make sophisticated agentic workflows more accessible and environmentally efficient in settings where compute is a limiting factor. In safety-critical domains such as nuclear regulatory review, we emphasize that the system is intended to assist, not replace, expert human judgment, and that its outputs should remain subject to domain-expert oversight.

Acknowledgements

This work was supported by the Substantiation Support Program, through the Korea Innovation Foundation funded by the Ministry of Science and ICT (No. 76170-26).

References

Acharya, A., Munikoti, S., Hellinger, A., Smith, S., Wagle, S., and Horawalavithana, S. NuclearQA: A human-made benchmark for language models for the nuclear domain. *arXiv preprint arXiv:2310.10920*, 2023.

Asai, A., Wu, Z., Wang, Y., Sil, A., and Hajishirzi, H. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *Proceedings of the 12th International Conference on Learning Representations (ICLR)*, 2024. Oral.

Cabrio, E. and Villata, S. Combining textual entailment and argumentation theory for supporting online debates interactions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 208–212, 2012.

Chen, K., Kong, Q., Zhao, F., and Mao, W. APEX-Searcher: Augmenting LLMs' search capabilities through agentic planning and execution. *arXiv preprint arXiv:2603.13853*, 2026.

Cho, J. et al. M3DocRAG: Multi-modal retrieval is what you need for multi-page multi-document understanding. *arXiv preprint arXiv:2411.04952*, 2024.

Doris, A. C. et al. DesignQA: A multimodal benchmark for evaluating large language models' understanding of engineering documentation. *arXiv preprint arXiv:2404.07917*, 2024.

Edge, D., Trinh, H., Cheng, N., Bradley, J., Chao, A., Mody, A., Truitt, S., and Larson, J. From local to global: A Graph RAG approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*, 2024.

Es, S., James, J., Anke, L. E., and Schockaert, S. RAGAs: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pp. 150–158, 2024.

Feng, T., Wu, Y., Lin, G., and You, J. Graph world model. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

Friedenthal, S., Moore, A., and Steiner, R. *A Practical Guide to SysML: The Systems Modeling Language*. Morgan Kaufmann, 3rd edition, 2014.

Gao, Y., Xiong, Y., Gao, X., et al. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2023.

Guo, Z., Xia, L., Yu, Y., Ao, T., and Huang, C. LightRAG: Simple and fast retrieval-augmented generation. *arXiv preprint arXiv:2410.05779*, 2024.

Gutiérrez, B. J., Shu, Y., Gu, Y., Yasunaga, M., and Su, Y. HippoRAG: Neurobiologically inspired long-term memory for large language models. *arXiv preprint arXiv:2405.14831*, 2024.

Hassanzadeh, O., Bhattacharjya, D., Feblowitz, M., Srinivas, K., Perrone, M., Sohrabi, S., and Katz, M. Answering binary causal questions through large-scale text mining. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 5003–5009, 2019.

Hellert, T., Montenegro, J., and Sulc, A. Osprey: Production-ready agentic AI for safety-critical control systems. *APL Machine Learning*, 4(1):016103, 2026.

Jain, A., Meenachi, N. M., and Venkatraman, B. NukeBERT: A pre-trained language model for low resource nuclear domain. *arXiv preprint arXiv:2003.13821*, 2020.

Lai, V. D. et al. SEC-QA: A systematic evaluation corpus for financial QA. *arXiv preprint arXiv:2406.14394*, 2024.

Lee, K.-H., Chen, X., Furuta, H., Canny, J., and Fischer, I. ReadAgent: A human-inspired reading agent with gist memory of very long contexts. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

Lee, Y. P. Mechanistic interpretability of LoRA-adapted language models for nuclear reactor safety applications. *arXiv preprint arXiv:2507.09931*, 2025.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., and Kiela, D. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in*

Neural Information Processing Systems (NeurIPS), volume 33, pp. 9459–9474, 2020.

Ma, Y. et al. MMLongBench-Doc: Benchmarking long-context document understanding with visualizations. *arXiv preprint arXiv:2407.01523*, 2024.

Nahid, M. M. H. and Rafiei, D. PRISM: Agentic retrieval with LLMs for multi-hop question answering. *arXiv preprint arXiv:2510.14278*, 2025.

Pan, L., Li, C., Li, J., and Tang, J. Prerequisite relation learning for concepts in MOOCs. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1447–1456, 2017.

Peldszus, A. and Stede, M. From argument diagrams to argumentation mining in texts: A survey. *International Journal of Cognitive Informatics and Natural and Artificial Intelligence*, 7:1–31, 2013.

Sarthi, P., Abdullah, S., Tuli, A., Khanna, S., Goldie, A., and Manning, C. D. RAPTOR: Recursive abstractive processing for tree-organized retrieval. In *Proceedings of the 12th International Conference on Learning Representations (ICLR)*, 2024.

Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., and Scialom, T. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

Sun, L., He, L., Jia, S., He, Y., and You, C. DocAgent: An agentic framework for multi-modal long-context document understanding. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2025.

Wang, S., Zhou, Y., and Fang, Y. BookRAG: A hierarchical structure-aware index-based approach for RAG on complex documents. *arXiv preprint arXiv:2512.03413*, 2025.

Xiong, B. et al. FDARxBench: Benchmarking regulatory and clinical reasoning on FDA generic drug assessment. *arXiv preprint arXiv:2603.19539*, 2026.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023a.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. ReAct: Synergizing reasoning and acting in language models. In *Proceedings of the 11th International Conference on Learning Representations (ICLR)*, 2023b.

Zhang, M. and Tang, Y. PageIndex: Next-generation vectorless, reasoning-based RAG. https://pageindex.ai/, 2025.

Zheng, L., Chiang, W.-L., Sheng, Y., et al. Judging LLM-as-a-judge with MT-Bench and chatbot arena. *arXiv preprint arXiv:2306.05685*, 2023.

Zhong, B., Ding, L., Luo, H., Zhou, Y., Hu, Y., and Hu, H. Ontology-based semantic modeling of regulation constraint for automated construction quality compliance checking. *Automation in Construction*, 28:58–70, 2012.

Zhou, A., Yan, K., Shlapentokh-Rothman, M., Wang, H., and Wang, Y.-X. Language agent tree search unifies reasoning, acting, and planning in language models. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

Zhu, F. et al. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. *arXiv preprint arXiv:2105.07624*, 2021.

Appendix

A. Baseline Configurations

The configuration used for each baseline is detailed in Table 11. All methods share the same generation LLM (GPT-4.1, temperature 0, max_tokens 300) and operate on the same 200-question benchmark over NuScale FSAR Chapters 01 and 05. The configurations below reflect each method's published defaults, tuned only where the original settings were incompatible with our document corpus (e.g., chunk size adjusted to accommodate FSAR section lengths).

RAPTOR constructs a recursive summarization tree using 100-token leaves, producing 3,422 leaves from our corpus, with retrieval via the `collapse_tree` strategy under a 2,000-token budget. **HippoRAG** extracts a hippocampal associative knowledge graph from 1,000-character passages (1,080 in total) and retrieves via Personalized PageRank combined with dense similarity (top-10). **LightRAG** performs three-pass entity-relation extraction over 1,200-token chunks (266 chunks), using hybrid retrieval of top-40 entities plus top-20 chunks. **GraphRAG** builds a community graph over 1,200-token chunks (267 chunks) and retrieves via local search at community level 2. **PageIndex** shares our system's tree environment and tools (browse, read, search) exactly, differing only in action-selection logic (no KG state accumulation), and thus serves as the single-variable ablation for state-conditioned planning.

Table 11. Baseline configurations. All methods use GPT-4.1 for generation.

Method	Paper	Core Mechanism	Chunking	Retrieval Config
RAPTOR	(Sarthi et al., 2024)	Recursive summ. tree	100 tok, 3,422 leaves	collapse_tree, 2K token budget
HippoRAG	(Gutiérrez et al., 2024)	PPR + hippocampal KG	1,000 chars, 1,080 pass.	PPR + dense, top-10
LightRAG	(Guo et al., 2024)	Dual-level graph + vector DB	1,200 tok, 266 chunks	hybrid, top-40 ent. + 20 chunks
GraphRAG	(Edge et al., 2024)	Community-based local search	1,200 tok, 267 chunks	local search, community_level=2
PageIndex	(Zhang & Tang, 2025)	Vectorless tree (no planning)	None (tree nodes)	BM25 + browse/read/search, max 4 hops

B. RAGAS: Our System by Reasoning Type

The cross-method per-type breakdown is shown in the main body (Table 6). Here we provide our system's decomposition across all four RAGAS metrics (Faithfulness, Answer Relevancy, Context Recall, Factual Correctness) by reasoning type, giving a fuller picture of where the planning loop is strongest and where it struggles.

Table 12. RAGAS metrics for our system by reasoning type.

Metric	Overall	Fact.	Comp.	Judg.
Faithfulness	**0.93**	0.92	0.92	**0.97**
Answer Rel.	**0.84**	0.85	0.78	**0.89**
Context Rec.	**0.93**	0.92	0.91	**0.96**
Factual Corr.	0.42	0.35	**0.49**	0.41

Two patterns emerge. First, **judgment questions dominate across three of four metrics** (Faith. 0.97, AR 0.89, CR 0.96), indicating that the planning loop is particularly effective at assembling complete evidence chains for regulatory conformance reasoning, the scenario most aligned with multi-hop state-conditioned exploration. Second, **Factual Correctness is the only metric where judgment does not lead**: the 0.42 overall FC reflects the wording-sensitive nature of this metric (exact string match against the expected answer) rather than a retrieval failure. A closer look at flagged cases shows that FC penalizes correct answers phrased differently than the reference (e.g., "vertical helical once-through SG with 1,380 tubes" vs. the reference's "helical coil SG integrated within RPV"). Both responses are factually correct; the metric penalizes the paraphrase. This is a benchmark-level limitation (single-perspective reference answers) rather than a model failure.

C. Per-Question Cost Breakdown

To illustrate the dynamic termination mechanism concretely, we profiled five representative questions drawn from different points in the three-axis taxonomy. Token counts were measured with `tiktoken` (`o200k_base` encoding) over the full LLM inputs (retrieved context, system prompts, and agent reasoning) and outputs, and costs were computed using GPT-4.1 pricing ($2/M input tokens, $8/M output tokens). Because the LLM is non-deterministic even at temperature 0 (due to backend batching), node and edge counts can vary slightly between runs; the numbers below are from a single dedicated profiling run.

Table 13. Per-question cost breakdown (5-question profiling sample). Illustrates dynamic termination: simple queries (Q001) cost $0.03, complex queries (Q191) cost $0.29.

QID	Type	Hops	Nodes	Edges	Tokens	Cost
Q001	factual / single	1	4	0	9,395	$0.03
Q071	comparative / single	4	19	63	124,264	$0.30
Q131	composite / cross	4	11	34	75,213	$0.18
Q161	judgment / multi	4	17	65	104,463	$0.25
Q191	image / cross	4	19	81	117,023	$0.29

The breakdown demonstrates that cost scales approximately with question complexity: the single-hop factual query Q001 consumes less than 10K tokens and costs $0.03, while cross-document queries (Q071, Q131, Q161, Q191) use the full 4-hop budget and cost between $0.18 and $0.30. Node counts also correlate with complexity, ranging from 4 nodes (single factual) to 19 nodes (multi-hop cross-document). Edge counts scale super-linearly with node counts, reflecting the pairwise nature of the two-stage edge inference. Total 200-question costs reported in the main body (Table 7) are extrapolated from this sample with an estimated $\pm$10–15% variance; a larger profiling run would tighten these estimates but would not change the qualitative conclusion that per-query cost is dominated by a small number of 4-hop complex queries.

D. 10Q Ablation: Per-Question Judge Detail

The 10-question ablation summary table is shown in the main text (Table 8). Here we provide the per-question O/X detail across the four variants (`full`, `no_vision`, `no_edges`, `no_browse_first`). The 10 questions were selected to span the three-axis taxonomy, including at least one question from each reasoning type and each modality.

Table 14. Per-question 3-Judge detail across ablation variants.

QID	Type	full	no_vis	no_edg	no_brw
Q001	fact/single	O	O	O	O
Q010	fact/multi	O	O	O	O
Q031	fact/single/table	O	O	O	O
Q058	fact/cross	O	O	**X**	O
Q071	comp/single	O	O	O	O
Q101	comp/cross/table	O	**X**	O	O
Q131	comp/cross/comp	O	**X**	O	O
Q161	judg/multi	O	O	O	O
Q176	judg/cross	O	O	O	O
Q191	judg/cross/image	O	O	O	**X**

Robustness vs. diagnostic questions. Six of the ten questions (Q001, Q010, Q031, Q071, Q161, Q176) are answered correctly by all four variants, indicating that the baseline system, even with any single component removed, handles single-hop factual queries, multi-hop factual queries, and judgment queries over well-structured evidence paths. These six *robustness* questions confirm that the planning loop as a whole is resilient to individual component ablation for straightforward cases. The remaining four questions are *diagnostic*: each is failed by exactly one variant, revealing which component is the critical dependency for that question type.

Failure modes by component. Vision RAG removal drops Q101 (table/comparative) and Q131 (composite/comparative). These are the only two questions requiring numerical content from figure-rendered tables; without vision processing the system cannot recover the relevant cells. Faithfulness also drops from 0.96 to 0.83, reflecting ungrounded answers when tabular evidence is withheld. **Edge inference removal** drops Q058 (seismic scope boundary), where the VIOLATES edge was required to mark non-safety-related systems as outside Category I scope. This failure does not replicate at the 200Q scale: both variants answer Q058 correctly in the main 200Q evaluation, indicating the 10Q Q058 result reflects execution-level variance rather than a systematic dependency. **Browse-first removal** drops Q191 (image/judgment/cross), where the agent, lacking the table-of-contents injection at Hop 1, selects poorly-targeted sections and never recovers the cross-document evidence chain.

Why scale up only edge inference to 200Q. Each component is responsible for a distinct failure mode at the 10Q level, supporting the complementarity claim. However, Q058 was the only failure that we suspected might be a sample-size artifact (because the VIOLATES edge is a rare edge type, it is plausible that the dependency does not generalize). Scaling edge inference to 200Q confirms this suspicion: the effect is reliably null (Table 9), while the vision and browse-first effects continue to manifest (Q101, Q131, Q191 all remain diagnostic at larger scales). This selective scale-up also reflects practical constraints: the 200Q full ablation consumes roughly 200× the compute of the 10Q pilot, so we prioritized the ablation with the strongest a priori cost-accuracy hypothesis (edge inference accounts for 65% of per-query cost).