RISK-CONTROLLED EVIDENCE SELECTION FOR RETRIEVAL-AUGMENTED QUESTION ANSWERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Evidence compression must account for both shared context and the reader's dependence on the removed text. A low-cost solution to a predicted support graph need not preserve an answer. We separate these questions through shared-resource selection and calibration of the complete compression-and-fallback policy. The selector retains the classical additive covering guarantee while a validated flow relaxation supplies a separate, instance-specific bound on graph cost. A common budget-aware search addresses a further mismatch between penalized selection and hard admission budgets. This search preserves already-admitted contexts but must itself be included in calibration. Whole-policy calibration controls paired answer degradation among admitted questions, not absolute factual correctness. On 120 controlled graphs, safeguarded shared-cost selection reduces mean cost by 18.5% relative to additive greedy. Its graph-cost certificates have median ratio 1.132 and maximum 2.076, with no zero lower bounds in this family. A budget-repair study covers 2,400 paired selector configurations and retains a case where finite search misses a feasible budget. Simulations over 1,500 calibration samples quantify the cost of fallback. These studies establish optimization and statistical properties.

1 INTRODUCTION

A retrieval-augmented question-answering system must decide which of its candidate passages to give the reader. Ranking is only part of this decision. Two highly ranked sentences may repeat the same fact, whereas a lower-ranked sentence may supply the bridge needed to combine them. Selecting both sentences from one document also need not cost twice as much as selecting one, because their title and surrounding context can be shared. The useful optimization problem is therefore not simply to retain the largest relevance scores.

The difficulty is that the selector cannot observe evidence sufficiency directly. It uses predictions about which text supports which part of the question. Even a minimum-cost solution to that predicted problem can omit a genuinely necessary fact. Moreover, a reader can answer incorrectly despite receiving sufficient evidence. This leaves two separate questions. How efficiently did the selector solve the problem it was given, and how much harm does its decision cause to the deployed reader? Neither answer determines the other.

We represent documents, sentences, shared resources, and predicted support relationships in a directed graph. The selected subgraph records provenance, not a logical proof. Its computational certificate concerns the declared graph and costs. A separate statistical certificate concerns the observed loss of the complete selection-and-reading policy on new questions.

1.1 RELATED WORK AND SCOPE

Retrieval-augmented generation combines external evidence with a pretrained reader (Lewis et al., 2020). IRCoT interleaves retrieval and reasoning, Self-RAG learns retrieval and critique decisions, and ALCE evaluates both answers and citations (Trivedi et al., 2023, Asai et al., 2024, Gao et al., 2023). We instead hold the candidate context and reader fixed. LLMLingua-2 and xRAG learn token selection or compressed representations (Pan et al., 2024, Cheng et al., 2024). Our selector retains identifiable source fragments, although whole-policy calibration can also wrap a fixed learned compressor.

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.

Related method	Established target	Question studied here
G-Retriever	PCST retrieval and a graph-aware reader	Shared resources with a fixed reader
Coherent factuality	Filtering deducibility subgraphs	Paired error under fallback
C-RAG	Certified generation risk	Cost of a selected support graph
Context filtering	Relevant-snippet retention	Reader degradation among admitted questions
Learn then Test	Simultaneous risk control	A fixed compression-policy family

Table 1: Conceptual positioning against prior work. Graph cost, evidence retention, and reader error are distinct targets. The table does not compare empirical performance.

Connected retrieval has substantial precedent. BANKS, minimum-cost connected-tree search, and STAR exploit database or relationship graphs (Bhalotia et al., 2002, Ding et al., 2007, Kasneci et al., 2009). G-Retriever uses prize-collecting Steiner retrieval for textual graph question answering (He et al., 2024). Our contribution is not Steiner retrieval or the classical weighted-cover greedy guarantee (Chvátal, 1979). It is an interface between shared-resource selection, a computable lower bound for the declared graph problem, and selective reader calibration.

Coherent factuality applies conformal methods to deducibility subgraphs (Rubin-Toles et al., 2025). C-RAG certifies generation risks and tests RAG configurations (Kang et al., 2024). Conformal context filtering controls retention of relevant snippets before generation (Chakraborty et al., 2026). These methods control different objects and should not be described as uncalibrated heuristics. Here the endpoint is reader degradation among questions admitted for compression, with full-context fallback. This endpoint fits general bounded-loss risk control and is not a new statistical principle. We use simultaneous testing over a fixed policy family (Angelopoulos et al., 2025). Appendix E compares the selected objects, shared costs, dependencies, losses, and sampling units.

1.2 Contributions

C1. Shared-resource selection. We distinguish the additive and graph objectives, preserve the weighted-cover guarantee under a shared-cost safeguard, and validate an instance-specific graph lower bound without a global fragment-completeness assumption.

C2. Budget and risk separation. We expose the gap between penalized selection and budget feasibility. A common bounded search repairs some rejected contexts without changing accepted ones. Its complete admission-and-reading policy, rather than isolated support scores, is the unit of risk calibration.

C3. Diagnostics with independent targets. Experiments report graph-bound tails, sharing and connectivity examples, budget-feasible misses, and true support separately. Calibration simulations retain non-certifying repetitions and fallback cost. They are not evidence of public-task answer gains.

2 Learned support and shared resources

A running example. Consider a question about the city containing a writer's archive. One sentence identifies the institution holding the archive, and another identifies that institution's location. Suppose two fragments share a document header costing 100 units and each has a 20-unit sentence. Selecting both costs 140 units, although charging each fragment independently gives 240. A competing 150-unit paragraph can appear cheaper under additive charging. The difference comes from charging, not the covering approximation. If the location sentence is incorrectly associated with a different institution, selecting all declared requirements can still produce a wrong answer. That is a support-prediction error, not a cost-optimization error.

Observable inputs. A question q and a fixed candidate collection determine a rooted directed graph $G_q = (V_q, E_q, r_q)$. Vertices represent the query, documents, or sentences. Directed edges record retrieval and containment dependencies. A resource is an object charged once when retained, such as

a document header or sentence. Node charges can be represented by splitting a node into an entry and exit joined by a charged edge. All costs are finite and nonnegative.

The query representation supplies m requirements. Requirement i has a predicted terminal group $T_i \subseteq V_q$. A selected graph covers it when some vertex in T_i is reachable from r_q. Groups represent alternatives, not independent assertions of truth. For example, several sentences may mention the institution in the running example. In the implemented QA interface, the groups are query facets derived without answer labels, and a learned sentence scorer filters their candidate matches. These lexical facets are a surrogate for evidence needs, not gold reasoning steps.

Let $M \subseteq [m]$ contain mandatory requirements. Each optional requirement has an omission cost $\pi_i \geq 0$. For a rooted selected graph H, write $y_i(H)$ for declared coverage and define

$$F(H) = C(H) + \sum_{i \notin M} \pi_i (1 - y_i(H)), \qquad C(H) = \sum_{e \in E(H)} c_e, \qquad y_i(H) = 1 \quad (i \in M). \tag{1}$$

Omission pays for leaving a requirement unresolved rather than supplying evidence. Mandatory requirements remain constraints.

Learned scores and labels. The support scorer $s_\phi(q, v) \in [0, 1]$ is trained on a training partition, separately from calibration and evaluation. A threshold and a query-to-sentence match define membership in T_i. Predicted groups, costs, and policy parameters are frozen before their calibration sample is used. Gold supporting facts are used to train the scorer and to evaluate its downstream consequences, but never to construct test-time coverage sets. The cost model uses measured resource units rather than gold labels. Mixed resource costs require declared weights and separate reporting of each resource.

Fragments and feasibility. A fragment A is a rooted, dependency-closed subgraph H_A with declared coverage $\Gamma(A) \subseteq [m]$, resource set R_A, and charge $\kappa_A \geq C(H_A)$. Dependency closure means that the fragment includes its required document and containment edges. We restrict attention to read-only fragments whose unions remain structurally admissible. The model does not authorize side-effecting tool calls or imply that conflicting actions can be combined safely.

The family $\mathcal{A}$ contains real fragments. For optional requirement i, add an omission fragment $\perp_i$ with coverage $\{i\}$, charge π_i, and no resources. Denote the expanded family by $\mathcal{A}'$. For a covering selection $\mathcal{S} \subseteq \mathcal{A}'$, its additive charge is

$$J(\mathcal{S}) = \sum_{A \in \mathcal{S}} \kappa_A, \qquad H(\mathcal{S}) = \bigcup_{A \in \mathcal{S} \cap \mathcal{A}} H_A, \qquad F(H(\mathcal{S})) \leq J(\mathcal{S}). \tag{2}$$

The two optimization comparators are

$$\mathrm{OPT}_{\mathcal{A}} = \min_{\mathcal{S} \text{ covers } [m]} J(\mathcal{S}), \qquad \mathrm{OPT}_G = \min_{H \in \mathcal{F}_q} F(H), \qquad \mathrm{OPT}_G \leq \mathrm{OPT}_{\mathcal{A}}, \tag{3}$$

where $\mathcal{F}_q$ is the set of feasible declared graphs. The inequality can be strict because additive charges duplicate shared resources. Neither optimum is defined using answer correctness.

Redundant omission fragments are removed if a later real fragment covers the same requirement. If a mandatory group has no available support, the selector reports infeasibility rather than treating an omission as successful coverage.

3 Selection and Computational Certificates

3.1 A Safeguarded Shared-Cost Selector

CRESS applies weighted-cover greedy to $\mathcal{A}'$. At each step, it selects the fragment with the smallest charge per newly resolved requirement. Let U be the unresolved requirements and R the resources already retained. A shared-cost candidate instead charges a real fragment only for resources in $R_A \setminus R$. This second rule accounts for reuse but is not covered by the usual additive-cost proof. We therefore retain the additive solution as a safeguard and return the less expensive feasible graph under the declared union-cost objective.

Algorithm 1 CRESS-S with additive safeguard

Require: Expanded family $\mathcal{A}'$, requirements $[m]$, resource costs c

for each rule $b \in \{\text{additive}, \text{shared}\}$ **do**
  $U \leftarrow [m]$, $\mathcal{S}_b \leftarrow \emptyset$, $R \leftarrow \emptyset$
  while $U \neq \emptyset$ **do**
    For positive-gain fragments, let $g_A = |\Gamma(A) \cap U|$
    Use $d_A = \kappa_A$, or $d_A = \sum_{e \in R_A \setminus R} c_e$ for shared real fragments
    If no positive-gain fragment exists, terminate this rule as infeasible
    Select a minimum d_A/g_A, breaking ties deterministically
    $U \leftarrow U \setminus \Gamma(A)$, $R \leftarrow R \cup R_A$, $\mathcal{S}_b \leftarrow \mathcal{S}_b \cup \{A\}$
  end while
  Remove omission fragments already covered by real fragments
end for
Return the feasible $H(\mathcal{S}_b)$ minimizing $F(H(\mathcal{S}_b))$

Theorem 1 (Inherited covering guarantee)**.** *Suppose $\mathcal{A}'$ covers all requirements, fragment unions are admissible, and $\kappa_A \geq C(H_A)$. Use the additive comparator $\mathrm{OPT}_{\mathcal{A}}$ from Eq.* (3) *and let $H_m = \sum_{j=1}^{m} 1/j$. The additive output $\mathcal{S}_G$ and safeguarded graph $\widehat{H}$ satisfy*

$$F(\widehat{H}) \leq F(H(\mathcal{S}_G)) \leq J(\mathcal{S}_G) \leq H_m \,\mathrm{OPT}_{\mathcal{A}}\,. \tag{4}$$

Zero-cost positive-gain fragments are permitted. If all requirements are mandatory, the covering problem already contains weighted set cover.

The final inequality is the classical weighted-cover result (Chvátal, 1979). The safeguard preserves it for the returned graph, even when the shared-cost rule produces a selection with a larger additive charge. It does not establish a new approximation factor for unrestricted directed Steiner problems. The logarithmic covering barrier follows from the set-cover reduction and the hardness result of Dinur and Steurer (2014). Full proofs and the treatment of omission fragments appear in Appendix A.

3.2 A GRAPH-LEVEL CERTIFICATE WITHOUT A COMPLETENESS PARAMETER

Equation (4) compares against the fragment optimum. A ρH_m graph bound additionally requires a uniform fragment-decomposition assumption. An unmeasured ρ, or a ratio for one reference graph, does not give a useful certified guarantee. We use an instance-specific graph lower bound instead.

We instead compute a lower bound directly on the declared graph. For group i, add a sink t_i and zero-cost edges from T_i to t_i. Let $x_e \in [0,1]$ select a physical edge and $z_i \in [0,1]$ represent omission, with $z_i = 0$ for $i \in M$. Each group has its own unit flow from the root to its sink, scaled by $1 - z_i$. If B_i is the directed incidence matrix, a valid relaxation is

$$\begin{aligned} L_G = \min \quad & \sum_{e \in E_q} c_e x_e + \sum_{i \notin M} \pi_i z_i \\ \text{subject to} \quad & B_i f_i = (1 - z_i)(\mathbf{e}_{r_q} - \mathbf{e}_{t_i}), \\ & 0 \leq f_{ie} \leq x_e \quad (e \in E_q), \qquad 0 \leq f_{ie} \leq 1 \quad (e \text{ added}), \\ & 0 \leq x_e, z_i \leq 1, \qquad z_i = 0 \quad (i \in M). \end{aligned} \tag{5}$$

Capacities are imposed separately for each group. Summing the flows before imposing capacities would incorrectly charge a shared edge once for every group. Additional type constraints may be relaxed when constructing L_G, since this only weakens the lower bound.

Theorem 2 (Instance-specific graph certificate)**.** *Every feasible declared support graph induces a feasible solution to Eq.* (5) *of no greater cost. Thus any verified lower bound $\underline{L}_G \leq L_G$ gives*

$$\underline{L}_G \leq \mathrm{OPT}_G \leq F(\widehat{H}), \qquad \frac{F(\widehat{H})}{\mathrm{OPT}_G} \leq \frac{F(\widehat{H})}{\underline{L}_G} \quad \textit{when } \underline{L}_G > 0. \tag{6}$$

The relaxation requires additional computation and can give a weak bound. When $\underline{L}_G = 0 < F(\widehat{H})$, no finite ratio follows. The implementation validates a dual lower bound using rational arithmetic.

This bounds optimization quality on the predicted graph, not whether its groups express the question's evidence needs.

A numerical example. Both greedy rules choose the running example's 150-unit paragraph, but the graph optimum and validated lower bound are 140. The certificate ratio is $15/14$. A reader can nevertheless follow the paragraph to an incorrect city. Appendix D gives this constructed graph and answer check, not a benchmark result.

3.3 Budgets are admission constraints

The penalized objective in Eq. (1) does not solve a hard-budget problem. In the running example, a 145-unit budget rejects the greedy 150-unit paragraph even though the two-sentence graph costs 140. Cheap omission penalties can create a different rejection by leaving a predicted requirement uncovered. Neither event establishes that no admissible context exists.

We compare the original gate with a common repair applied to every selector. Let $\tau(H)$ be the cost of the serialized context and let $a_\theta(q)$ require a nonempty context, at least one predicted requirement, complete predicted coverage, the common confidence threshold, and $\tau(H) \le B$. The repair preserves every context that already passes. Otherwise it searches unions of real fragments, excluding omission choices. Each expansion must cover a new requirement. The search retains 32 partial states and stops after at most m expansions along a path. Costs are recomputed on the serialized union. The ordering, width, and tie rules are fixed before calibration and are identical across selectors.

The repaired policy uses a complete candidate only if it passes the original gate. Otherwise it falls back. A finite search can miss the only feasible path, so failure means no witness was found, not infeasibility. Appendix B gives the details.

Proposition 1 (Admission under common repair)**.** *For every question, the original and repaired gates satisfy $a_\theta(q) \le a_\theta^+(q)$, and their contexts agree when $a_\theta(q) = 1$. This does not imply smaller selective risk, greater admission after calibration, or a smaller penalized objective.*

A repair can recover a useful context that costs more than the penalty for omission. Theorem 1 therefore applies to the safeguarded core, not automatically to the repaired output. A repaired feasible graph may still be evaluated by Theorem 2. Its reader-risk guarantee instead requires calibration of the repaired policy. For tokenized text, τ need not equal the additive resource charge or be monotone under concatenation. The implementation does not discard a partial context merely because its current serialized cost exceeds B.

4 Calibrating the complete reading policy

4.1 Why local calibration is insufficient

Suppose a verifier assigns confidence 0.911 to every edge in a bin. Ten percent of these edges are cheap and fail with probability 0.8. The rest are expensive and fail with probability 0.01. The bin-average failure probability is 0.089. A large calibration sample can estimate this average accurately, yet a cost-minimizing selector that retains cheap edges has failure probability 0.8. Simultaneous confidence bounds across bins do not fix the problem, because selection occurred within a bin.

A union bound is valid for local risks conditional on the selected graph under sound composition. Bin averages do not supply those conditional risks, and support edges do not account for reader errors after selection. We therefore calibrate the complete reader policy against the observed task loss.

4.2 Selective risk and fallback

Fix a trained scorer, reader, prompt, decoding rule, and finite family of K policies before observing a calibration sample. Policy θ includes the selector, its resource budget, and an admission rule $a_\theta(q) \in \{0, 1\}$. Admission may depend on scores, selected size, predicted feasibility, or a computational certificate, but not on the question's answer label. A rejected compression uses the baseline reader on the full candidate context. Let $\ell_\theta(q, y) \in \{0, 1\}$ be a prespecified loss for the compressed prediction

and

$$R_\theta = \Pr\{\ell_\theta(Q, Y) = 1 \mid a_\theta(Q) = 1\}. \tag{7}$$

The target may be answer mismatch, a missing reference support, or a prespecified joint event. The choice must precede calibration. A confidence bound for one endpoint is not automatically a certificate for another.

On n independent calibration questions, let N_θ count admitted questions and K_θ count losses among them. Use the one-sided Clopper–Pearson upper bound (Clopper and Pearson, 1934)

$$U_\theta = \begin{cases} 1, & N_\theta = 0 \text{ or } K_\theta = N_\theta, \\ \mathrm{Beta}^{-1}(1 - \delta/K,\ K_\theta + 1,\ N_\theta - K_\theta), & \text{otherwise.} \end{cases} \tag{8}$$

Among policies with $U_\theta \le \alpha$, select the one minimizing calibration-set mean deployment cost, including full-context fallback. If none qualifies, retain the baseline for every question.

Theorem 3 (Finite-family selective control)**.** *Assume the policy family is independent of the calibration data and calibration questions are independent draws from the deployment distribution. With probability at least $1 - \delta$ over calibration, every policy certified by Eq.* (8) *has $R_\theta \le \alpha$. Any data-dependent choice among those policies retains this property. No independence between different policies' predictions on the same question is required.*

This is a simultaneous testing construction (Angelopoulos et al., 2025). The independent sampling unit is the question, not an edge, sentence, or alternative context. The guarantee permits choosing among certified candidates on the calibration data. Minimizing their empirical cost does not certify a population-cost optimum. It does not allow retraining the scorer, changing the prompt, or adding new thresholds after examining the calibration labels.

4.3 Controlling degradation rather than absolute error

A weak baseline reader may make absolute error control impossible at a useful admission rate. An operational alternative is to control degradation relative to that reader. Let E_θ and E_0 be the compressed and full-context answer-error indicators on the same question, and calibrate the paired loss

$$D_\theta = \mathbf{1}\{E_\theta = 1,\ E_0 = 0\}. \tag{9}$$

For the deployed policy that falls back whenever $a_\theta = 0$, pointwise comparison of the two error indicators gives the following consequence.

Corollary 1 (Excess error with fallback)**.** *On the event of Theorem 3, calibration of Eq.* (9) *yields*

$$\Pr(E_{\mathrm{deploy}} = 1) - \Pr(E_0 = 1) \le \alpha \Pr(a_{\widehat{\theta}} = 1). \tag{10}$$

The controlled probability is $\Pr(E_\theta = 1, E_0 = 0 \mid a_\theta = 1)$. Its denominator is all admitted questions, not just those answered correctly by the baseline. In particular, it does not control $\Pr(E_\theta = 1 \mid a_\theta = 1, E_0 = 0)$ or absolute answer error at the same level. Appendix C.3 gives the decomposition. Baseline errors, admission, and total fallback-inclusive cost must accompany the bound.

Calibration feasibility. With zero paired failures, certification requires

$$N_\theta \ge N_{\min} = \left\lceil \frac{\log(\delta/K)}{\log(1 - \alpha)} \right\rceil. \tag{11}$$

The fixed public-data family contains seven selectors at three budgets, each with and without the common repair. Thus $K = 42$, $\alpha = 0.02$, and $\delta = 0.05$ require $N_{\min} = 334$ admitted questions with no paired failures. This is 11.13% of the planned 3,000 calibration questions. Both variants enter the same simultaneous correction. The former 21-policy family required 299 admissions, but that smaller correction cannot cover the augmented family. No reader calibration has yet occurred. Failed certification remains an outcome, not a reason to remove policies or loosen the target.

Selector	F/OPT_G	$F/\underline{L}_G$	True recall
Maximum coverage	1.409 ± 0.345	1.469	0.894
Score-weighted coverage	1.507 ± 0.426	1.575	0.910
CRESS additive	1.344 ± 0.330	1.403	0.912
Union-cost greedy	1.193 ± 0.206	1.243	**0.933**
CRESS-S	$\mathbf{1.160 \pm 0.189}$	**1.209**	0.927

Table 2: Controlled graph comparison over 120 instances. Ratios and recall are averaged per instance. The first ratio includes its standard deviation across instances. All methods cover every declared group. Bold indicates the smallest cost ratio or highest independently evaluated recall, not a significance claim.

5 Controlled evaluation

We evaluate graph optimization and policy calibration separately. All results in this section come from controlled instances with saved inputs and per-run outputs. The first study uses latent true-support labels that differ from the labels given to the selector. The second has an analytically known population risk, allowing violations of the calibration guarantee to be measured directly. Neither experiment is a public-data question-answering result.

5.1 Shared costs and imperfect support predictions

Each graph has 12 requirements, six shared-resource hubs, 48 candidate evidence nodes, and 12 costly singleton candidates. The singleton candidates ensure feasibility in this diagnostic, not realism of a retrieval system. Root-to-hub costs are scaled by $b \in \{0.25, 1, 4, 16\}$. Evidence costs are log-normal. Every ordinary evidence node has one to three latent support labels. Each true label is removed independently with probability $\eta \in \{0, 0.1, 0.25\}$ and each other label is added with probability $\eta/6$. Selectors observe only the corrupted memberships. Ten seeds across the twelve settings give 120 instances.

All methods receive the same candidates, corrupted memberships, costs, and exogenous scores. We compare additive CRESS, score-weighted marginal coverage per charge, maximum marginal coverage, union-cost greedy, and safeguarded CRESS-S. Mixed-integer linear programs solve both the graph and additive-fragment objectives to optimal solver status on every instance. Rationally validated flow bounds provide a separate lower bound. These comparisons test sharing and approximate optimization, rather than asking a ranker without coverage information to match a coverage oracle.

CRESS-S reduces mean union cost by 18.5% relative to additive CRESS, with a seed-cluster bootstrap interval of [13.3%,23.3%]. Its mean ratio to the graph optimum is 1.160, while the mean certified ratio is 1.209. The improvement grows when shared resources become expensive. The reference ratio $\operatorname{OPT}_{\mathcal{A}} / \operatorname{OPT}_G$ increases from 1.11 to 2.31 across the smallest and largest sharing multipliers. This directly measures an instance-specific representation gap, without treating it as an estimate of a uniform ρ.

The certificate distribution is also reported rather than only its mean. For CRESS-S, the median, 95th percentile, and maximum of $F/\underline{L}_G$ are 1.132, 1.616, and 2.076. One of 120 ratios exceeds two. No selector encounters a zero or near-zero bound in this family. Separate adversarial examples give valid but loose ratios up to 2.5 and a zero witness with no finite ratio. Appendix F.2 gives every selector's tails and distinguishes relaxation weakness from support error.

Independent support evaluation gives a different ordering. At $\eta = 0.25$, CRESS-S has mean true-support recall 0.867 despite covering every declared group. Union-cost greedy reaches 0.875 and has the highest mean true recall overall. The safeguard chooses between candidates by cost, not by hidden labels, so it cannot guarantee a semantic improvement. These observations explain why a low graph-cost ratio is not a substitute for reader-level calibration. Appendix F reports the full grid and separate cost, coverage, and certificate plots.

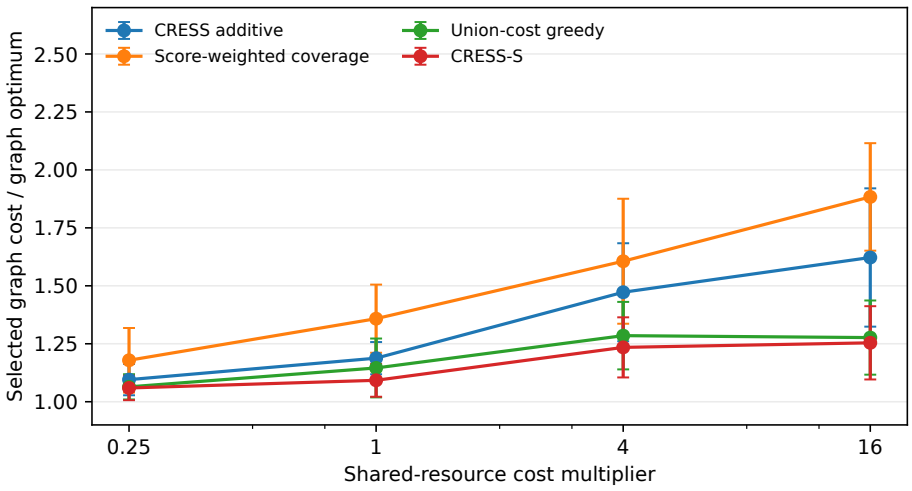


Figure 1: Resource sharing and mean approximation ratio. Error bars are standard deviations over ten seeds after averaging noise settings within each seed. All methods use the same graph-MILP comparator.

Budget	Feasible	Base	Repaired	Cost ratio	True recall
12	80	67	**79**	0.446	0.931
24	93	77	**93**	0.334	0.927
48	120	77	**120**	0.073	0.922
96	120	77	**120**	0.075	0.922

Table 3: Common repair for CRESS-S, with 120 graphs per budget. Feasibility uses the graph optimum. Cost includes fallback relative to the full graph. True recall is among repaired admissions. Bold highlights admission, not reader quality.

5.2 Penalties, budget rejection, and common repair

We reuse the 120 candidate graphs and apply an omission penalty of four units, fixed before this diagnostic. Four budgets give 480 distinct searches and 2,400 paired configurations across the five selectors. The search never receives the integer-program solution or latent support labels. The integer optimum determines only whether a budget-feasible declared cover exists.

At budget 12, repair recovers 12 of 13 feasible CRESS-S rejections and still misses one feasible instance. At budget 48 it recovers all 43 rejected contexts. The same repair helps every comparator, often by more, and the remaining cost differences narrow. Increased admission does not ensure better evidence. At budget 48, true recall among CRESS-S admissions changes from 0.923 to 0.922. These are graph-resource units and latent support labels, not tokenizer costs or reader accuracy. Appendix F.3 reports all methods, search failures, and fallback costs. The common search is part of the proposed reader policy and must be evaluated and calibrated with it.

5.3 Policy calibration with known population risk

For each calibration question, draw U, V independently and uniformly on $[0, 1]$. Its error indicator is $\mathbf{1}\{V < 0.02 + 0.5U^2\}$. A threshold policy admits $U \le t$, so its conditional population risk is $0.02 + t^2/6$. We fix 16 thresholds from 0.10 to 1.00. Compressed cost is $200 + 100U$ and fallback cost is 1,000, both in arbitrary units. The population deployment cost is $1,000 - 800t + 50t^2$. We target $\alpha = 0.10$ and $\delta = 0.05$, using 500 independent calibration samples at each of three sizes.

Small calibration samples often support no useful compression policy. At $n = 256$, 360 of 500 repetitions fall back everywhere. With 4,096 questions, every repetition certifies a nonempty policy and mean expected deployment cost falls to 54.3% of baseline. The corresponding mean conditional

n	Certified samples	Admission	Cost / baseline	Risk violations
256	140/500	0.115	0.911	3/500
1,024	497/500	0.476	0.631	1/500
4,096	**500/500**	**0.593**	**0.543**	0/500

Table 4: Selective calibration simulation. A certified sample is a calibration repetition in which at least one policy qualifies. Admission and cost include repetitions that fall back on every question. A violation means that the chosen nonempty policy has population risk above 0.10. The zero count in the last row does not establish zero violation probability.

population risk is 0.079. These are simulation results, not measured language-model error rates. Separately, a 50,000-edge realization of the example in Section 4 gives a pooled one-sided upper bound of 0.092 against selected-subgroup risk 0.8, illustrating the failure of local calibration after selection.

5.4 What these experiments do not establish

The public-data protocol fixes HotpotQA distractor contexts (Yang et al., 2018), a trained scorer, the Qwen2.5 reader (Qwen Team, 2024), and all 42 policies before calibration. It has not been executed. The present results establish neither a 98% answer-correctness claim nor QA token or latency savings. Paired degradation can be small when both readers are wrong. A useful result must report absolute accuracy, the four paired outcomes, admission, fallback, and measured scoring, search, caching, and reader costs. Appendix G describes this unrun experiment.

6 Limitations

Missing candidates, incorrect predicted memberships, and incomplete annotations change what the graph optimum means. A better cost ratio need not imply a better answer. Neither the graph certificate nor common repair corrects these errors. Provenance records permit inspection, but improved human diagnosis would require a user study.

The calibration guarantee requires questions independently drawn from the deployment population, with training and policy construction completed beforehand. Distribution shift, changed prompts, model updates, and correlated questions treated as independent can invalidate it. Paired degradation limits added mistakes, not errors already made by the baseline. The unrun public-reader experiment remains the main empirical gap, including whether any repaired policy certifies at a useful admission rate.

7 Conclusion and implications

Evidence selection has distinct computational and statistical objectives. Shared-resource optimization can reduce the cost of a declared support graph, and a validated lower bound can quantify the remaining optimization gap. Neither result establishes that the declared requirements capture the question or that the reader will preserve its answer. Whole-policy calibration addresses a different question by controlling a prespecified loss after selection, admission, and generation.

The budget diagnostic exposes a practical consequence of this separation. A good penalized solution can fail a hard admission gate despite a feasible alternative. Common repair recovers many such cases, but it can miss a feasible context, increase semantic error, or fail calibration. Its benefit must therefore be judged on the joint risk, admission, and deployment-cost frontier, not admission alone. The next empirical step is the frozen public-reader study, including non-certifying policies and baseline errors. More expressive support prediction, adaptive evidence acquisition, and distribution shift remain important extensions. Each changes the deployed policy and must be reflected in its evaluation rather than hidden behind a graph-cost certificate.

REFERENCES

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control. *The Annals of Applied Statistics*, 19(2):1641–1662, 2025. doi: 10.1214/24-AOAS1998. URL https://gsbpreserve.stanford.edu/view/32366/learn-then-test-calibrating-predictive-algorithms-to-achieve-risk-control.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/25f7be9694d7b32d5cc670927b8091e1-Abstract-Conference.html.

Gaurav Bhalotia, Arvind Hulgeri, Charuta Nakhe, Soumen Chakrabarti, and S. Sudarshan. Keyword searching and browsing in databases using BANKS. In *Proceedings of the 18th International Conference on Data Engineering*, pages 431–440, 2002. doi: 10.1109/ICDE.2002.994756. URL https://repository.ias.ac.in/100106/.

Debashish Chakraborty, Eugene Yang, Daniel Khashabi, Dawn Lawrie, and Kevin Duh. Principled context engineering for RAG: Statistical guarantees via conformal prediction, 2026. URL https://arxiv.org/abs/2511.17908v2. Version 2, January 19, 2026. Accepted at ECIR 2026.

Xin Cheng, Xun Wang, Xingxing Zhang, Tao Ge, Si-Qing Chen, Furu Wei, Huishuai Zhang, and Dongyan Zhao. xRAG: Extreme context compression for retrieval-augmented generation with one token. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3476. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/c5cf13bfd3762821ef7607e63ee90075-Abstract.html.

Vašek Chvátal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979. doi: 10.1287/moor.4.3.233. URL https://pubsonline.informs.org/doi/10.1287/moor.4.3.233.

C. J. Clopper and E. S. Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413, 1934. doi: 10.1093/biomet/26.4.404. URL https://doi.org/10.1093/biomet/26.4.404.

Bolin Ding, Jeffrey Xu Yu, Shan Wang, Lu Qin, Xiao Zhang, and Xuemin Lin. Finding top-k min-cost connected trees in databases. In *Proceedings of the 23rd International Conference on Data Engineering*, pages 836–845, 2007. URL https://dbgroup.se.cuhk.edu.hk/node/92.

Irit Dinur and David Steurer. Analytical approach to parallel repetition. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pages 624–633, 2014. doi: 10.1145/2591796.2591884. URL https://www.dsteurer.org/paper/productgames/.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, 2023. doi: 10.18653/v1/2023.emnlp-main.398. URL https://aclanthology.org/2023.emnlp-main.398/.

Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V. Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. G-Retriever: Retrieval-augmented generation for textual graph understanding and question answering. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/eaf1c9726648c8ba363a5c927440529-Abstract.html.

Mintong Kang, Nezihe Merve Gürel, Ning Yu, Dawn Song, and Bo Li. C-RAG: Certified generation risks for retrieval-augmented language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 22963–23000, 2024. URL https://proceedings.mlr.press/v235/kang24a.html.

Gjergji Kasneci, Maya Ramanath, Mauro Sozio, Fabian M. Suchanek, and Gerhard Weikum. STAR: Steiner-tree approximation in relationship graphs. In *Proceedings of the 25th International Conference on Data Engineering*, pages 868–879, 2009. doi: 10.1109/ICDE.2009.64. URL https://portal.fis.tum.de/en/publications/star-steiner-tree-approximation-in-relationship-graphs/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474, 2020. URL https://proceedings.nips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Rühle, Yuqing Yang, Chin-Yew Lin, H. Vicky Zhao, Lili Qiu, and Dongmei Zhang. LLMLingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 963–981, 2024. doi: 10.18653/v1/2024.findings-acl.57. URL https://aclanthology.org/2024.findings-acl.57/.

Qwen Team. Qwen2.5-3B-Instruct model card. Hugging Face model repository, 2024. URL https://huggingface.co/Qwen/Qwen2.5-3B-Instruct. Revision 82fb6b13373667c7301a7d785746d66db51e3fa9.

Maxon Rubin-Toles, Maya Gambhir, Keshav Ramji, Aaron Roth, and Surbhi Goel. Conformal language model reasoning with coherent factuality. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/679fcceef65c3d855aa885bd024542c1-Abstract-Conference.html.

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

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pages 10014–10037, 2023. doi: 10.18653/v1/2023.acl-long.557. URL https://aclanthology.org/2023.acl-long.557/.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, 2018. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

A PROOFS

A.1 THE OPTIMIZATION PROBLEM BEING CERTIFIED

Fix one question and its predicted support graph. All statements in this appendix condition on that graph, its costs, its groups, and its fragment family. They are deterministic unless a probability distribution is explicitly introduced. A feasible graph covers every mandatory group. Optional groups need not be covered, but their omission costs are included in F. Feasibility concerns these declared groups, not an unknown collection of genuinely necessary facts.

The expanded covering problem requires every index in $[m]$ to be resolved by either a real fragment or its allowed omission fragment. Mandatory groups have no omission fragments. If a selected real fragment later covers a group whose omission fragment was selected earlier, the latter is removed. Consequently the remaining omission charge is precisely the charge for groups not covered by the selected real fragments. The unexpanded graph may sometimes cover additional groups through composition of selected fragments. That can only reduce F relative to the additive objective and does not invalidate the inequalities below.

All resource charges are nonnegative. Every real fragment is supplied with a dependency-closed resource set, and the union of accepted fragments is required to be admissible. This closure condition is appropriate for reading evidence from documents. It is not sufficient for tool plans with incompatible side effects. Such plans need additional feasibility constraints and are outside the guarantees studied here.

Lemma 1 (Omission expansion)**.** *The minimum additive charge of a real-fragment selection plus penalties for unresolved optional groups equals the minimum charge of a cover by $\mathcal{A}'$.*

Proof. Start with a real-fragment selection. Add $\perp_i$ for each optional group that no selected real fragment covers. The resulting family covers every group and has the original real-fragment charge plus exactly the required penalties. Conversely, consider a cover in $\mathcal{A}'$. Remove each omission fragment whose group is covered by a real fragment. This operation preserves coverage and cannot increase charge. The remaining omission fragments correspond to optional groups that are not covered by real fragments. Reading their charges as penalties produces a feasible solution to the original additive problem of the same value. The two transformations show both inequalities between the optima. □

A.2 WEIGHTED COVERING AND THE SAFEGUARD

Proof of Theorem 1. If $m = 0$, the empty selection has cost zero and the conclusion is immediate. Suppose $m > 0$. Because $\mathcal{A}'$ covers all groups, each iteration with nonempty unresolved set U has a positive-gain candidate. Each selected candidate resolves at least one group, so the algorithm terminates in at most m iterations.

Let $\mathcal{S}^\star$ be a minimum-charge expanded cover, with charge $J^\star$. At a greedy iteration with $u = |U|$ unresolved groups,

$$\sum_{A \in \mathcal{S}^\star} |\Gamma(A) \cap U| \geq u. \tag{12}$$

Discard the zero-gain terms. The weighted average of the ratios $\kappa_A/|\Gamma(A) \cap U|$, using their gains as weights, is at most $J^\star/u$. Hence some positive-gain atom has ratio at most $J^\star/u$, including when $J^\star = 0$. The greedy choice has no larger ratio.

Charge a selected fragment's cost equally to its g newly resolved groups. At the iteration starting with u unresolved groups, this creates total charge at most $gJ^\star/u$. Since

$$\frac{g}{u} \leq \sum_{j=u-g+1}^{u} \frac{1}{j}, \tag{13}$$

summing over the disjoint blocks of resolved group indices gives $J(\mathcal{S}_G) \leq H_m J^\star$. Removing redundant omission fragments only decreases this charge.

Nonnegative resource costs and $\kappa_A \geq C(H_A)$ imply that the cost of the resource union is at most the sum of its fragment charges. The remaining penalties are included among the selected

charges. Therefore $F(H(\mathcal{S}_G)) \leq J(\mathcal{S}_G)$. Both portfolio candidates are checked for the same feasibility conditions, and the safeguard chooses the smaller value of F, giving $F(\widehat{H}) \leq F(H(\mathcal{S}_G))$. Combining these inequalities proves Eq. (4). No approximation property of the shared-cost candidate was used. □

The guarantee applies to union cost, not necessarily to the additive charge of the winning shared-cost selection. That distinction prevents a common but invalid inference. A portfolio can have a better resource union while charging a shared header repeatedly when its fragments are summed. Nothing in Theorem 1 claims that the additive charge of that portfolio winner is H_m-optimal.

Proposition 2 (The classical lower bound applies)**.** *The declared mandatory selection problem contains weighted set cover, including instances whose rooted graph has depth two. Thus a general improvement below logarithmic covering hardness would improve the corresponding set-cover approximation threshold.*

Proof. For a set-cover instance with universe $[m]$ and sets S_j of costs w_j, construct a root, a vertex v_j for each set, and one terminal t_i for each element. The root-to-v_j edge costs w_j, and the edge from v_j to t_i has cost zero whenever $i \in S_j$. Group i is the singleton $\{t_i\}$. Selecting a set creates the corresponding rooted fragment containing its outgoing terminal edges. A set cover gives a graph of the same cost covering all terminal groups. Conversely, the selected positive-cost root edges of any feasible graph identify sets covering the universe. The transformations preserve the objective and approximation ratio. The hardness statement therefore follows from set-cover hardness (Dinur and Steurer, 2014). It is not a new lower bound specific to language models. □

A.3 Why a fragment bound need not be a graph bound

Proposition 3 (Shared-cost gap)**.** *For every positive integer m, a rooted graph and a path-fragment family exist for which the ratio between the additive-fragment optimum and the union-graph optimum approaches m.*

Proof. Connect the root to a hub by an edge of cost $B > 0$, and connect the hub to each of m distinct mandatory terminals by an edge of cost $\varepsilon > 0$. Supply only one root-to-terminal path fragment for each terminal, each charged $B + \varepsilon$. Every additive cover selects all m fragments, so $\mathrm{OPT}_{\mathcal{A}} = m(B + \varepsilon)$. The selected graph pays the hub edge once, giving $\mathrm{OPT}_G = B + m\varepsilon$. Their ratio is

$$\frac{\mathrm{OPT}_{\mathcal{A}}}{\mathrm{OPT}_G} = \frac{m(B+\varepsilon)}{B+m\varepsilon} \longrightarrow m \quad \text{as } B/\varepsilon \longrightarrow \infty. \tag{14}$$

Thus an atom-relative approximation factor cannot simply be relabeled a graph-relative factor. This example concerns charging, even though every terminal is proposed correctly and the fragment family can represent the optimal graph union. □

The usual ρH_m transfer remains valid when every feasible graph has a covering decomposition of additive charge at most $\rho F(H)$. Applying that condition to a graph optimum gives $\mathrm{OPT}_{\mathcal{A}} \leq \rho\,\mathrm{OPT}_G$, and Theorem 1 supplies the result. The condition must include both coverage and permitted omission decisions. Merely having a path to each terminal does not imply a small charging factor. Measuring a ratio for a nonoptimal reference graph also does not establish a worst-case upper bound for other graphs or questions.

A.4 The flow relaxation and its certificate

Proof of Theorem 2. Take any feasible selected graph H. Set $x_e = 1$ on its physical edges and zero elsewhere. Set $z_i = 0$ for covered groups and $z_i = 1$ for omitted optional groups. Mandatory groups are covered, so their prescribed value $z_i = 0$ is respected.

For each covered group, choose one directed path in H from the root to a member of T_i, append its zero-cost edge to t_i, and send one unit of group-i flow on that path. Cycles can be removed before choosing the path. Set the flow of each omitted group to zero. Each flow satisfies its conservation equations and its separate capacity constraints. Several groups may use the same physical edge because the relaxation imposes $f_{ie} \leq x_e$ for each i, not $\sum_i f_{ie} \leq x_e$.

This assignment has objective $F(H)$. Relaxing integrality and any additional type restrictions can only decrease the minimum. Hence $L_G \leq \mathrm{OPT}_G$. Any validated dual lower bound satisfies $\underline{L}_G \leq L_G$. Feasibility of the returned graph implies $\mathrm{OPT}_G \leq F(\widehat{H})$. When $\underline{L}_G > 0$, division of the positive quantities gives Eq. (6). □

An optimization solver's primal objective is not itself a certified lower bound unless its optimality and numerical tolerances are accounted for. We use a box-constrained dual evaluation that remains valid even when the supplied multipliers have residuals. Write the relaxation in the form

$$\min c^\top v \quad \text{subject to} \quad Av \leq b,\ Bv = d,\ 0 \leq v \leq u. \tag{15}$$

Here the finite upper bounds u include zeros for mandatory omission variables. For any rational $\lambda \leq 0$ and any rational ν, define

$$r = c - A^\top \lambda - B^\top \nu, \qquad L(\lambda, \nu) = b^\top \lambda + d^\top \nu + \sum_j \min\{0, u_j r_j\}. \tag{16}$$

Lemma 2 (Residual-safe lower bound)**.** *For every feasible v in Eq. (15), $L(\lambda, \nu) \leq c^\top v$. No additional dual feasibility conditions are required.*

Proof. Because $Av \leq b$ and $\lambda \leq 0$, $\lambda^\top Av \geq \lambda^\top b$. Equality feasibility gives $\nu^\top Bv = \nu^\top d$. Therefore

$$c^\top v = r^\top v + \lambda^\top Av + \nu^\top Bv \geq r^\top v + \lambda^\top b + \nu^\top d. \tag{17}$$

The minimum of $r_j v_j$ on $[0, u_j]$ is $\min\{0, u_j r_j\}$. Sum this inequality over coordinates to obtain the claimed bound. □

The implementation clips the solver's inequality multipliers at zero, converts costs and multipliers to their rational binary representations, and evaluates Eq. (16) with rational arithmetic. It records the numerator and denominator of the bound. The selected cost is also summed rationally. Lower-bound decimals are rounded downward and reported ratio bounds upward. The certificate thus concerns the declared numerical instance, whose costs are the represented finite values. This procedure prevents a small infeasible dual residual from becoming an overoptimistic optimality claim.

A.5 Sensitivity to uncertain omission costs

Uncertainty about whether a requirement is needed can be analyzed separately from support-graph correctness. If p_i is a requirement probability for optional group i, define the additive objective

$$J_p(\mathcal{S}) = \sum_{A \in \mathcal{S} \cap \mathcal{A}} \kappa_A + \sum_{i \text{ omitted by } \mathcal{S}} p_i \pi_i. \tag{18}$$

Mandatory requirements remain separate hard constraints. All comparisons in the next statement concern this same additive family.

Proposition 4 (Uniform perturbation)**.** *If $\sum_{i \notin M} \pi_i |p_i - \widehat{p}_i| \leq \Delta$ and a selection $\widehat{\mathcal{S}}$ is an α-approximation for $J_{\widehat{p}}$, then for an optimizer $\mathcal{S}^\star$ of J_p,*

$$J_p(\widehat{\mathcal{S}}) \leq \alpha J_p(\mathcal{S}^\star) + (\alpha + 1)\Delta. \tag{19}$$

Proof. For any fixed feasible selection, only omission terms differ between the objectives. The triangle inequality gives $|J_p(\mathcal{S}) - J_{\widehat{p}}(\mathcal{S})| \leq \Delta$ uniformly over selections. Hence

$$\begin{aligned} J_p(\widehat{\mathcal{S}}) &\leq J_{\widehat{p}}(\widehat{\mathcal{S}}) + \Delta \\ &\leq \alpha \min_{\mathcal{S}} J_{\widehat{p}}(\mathcal{S}) + \Delta \\ &\leq \alpha J_{\widehat{p}}(\mathcal{S}^\star) + \Delta \\ &\leq \alpha J_p(\mathcal{S}^\star) + (\alpha + 1)\Delta. \end{aligned} \tag{20}$$

□

An additive-objective approximation must not be silently changed into an approximation to the union-cost optimum. The two objectives differ whenever resources are shared. A separate graph-transfer assumption or a graph lower bound is needed for that comparison. The public QA implementation uses a fixed omission price rather than claiming to estimate a population requirement probability.

B COMMON REPAIR AND THE BUDGET GATE

The budget is a constraint on the serialized context, not the objective minimized by the core selector. We use $\tau(R)$ for the serialization cost of a union of real resources. In the graph diagnostic this is the sum of the retained resource charges. In the reader implementation it is the tokenizer count of the final text. Costs for different representations are not substituted for one another.

Fixed search. A state consists of the covered predicted groups, the selected real resources, and the fragment indices. Start from the empty state. Extend a state only by a fragment that covers at least one currently uncovered group. Each path has at most m extensions. Complete states satisfying $\tau(R) \leq B$ are candidate witnesses. Equal group/resource states are deduplicated. At each depth, partial states are ordered by normalized budget overflow, decreasing covered-group count, increasing cost, and deterministic fragment indices. At most 32 states survive. The cheapest complete witness found is checked against the original confidence and nonempty-facet gate before being used.

An already-admitted original context bypasses the search and is returned unchanged. Rejected original contexts share the same search, candidates, width, ordering, cost function, and final gate across all selectors. In particular, the search does not inspect answer labels, reference supports, reader outputs, or the integer optimum. A cached search result may be reused by selectors with the same question, candidates, and budget.

For the diagnostic's additive nonnegative resource costs, an over-budget state cannot acquire lower cost and may be pruned. For text serialization, tokenization at a new boundary can change the count nonadditively. The reader search therefore retains over-budget partial states according to the fixed beam ordering rather than using monotonicity as an unproved assumption. Finite-width pruning remains heuristic in either case. A report of no witness is not a report of infeasibility.

Proof of Proposition 1. Fix a question. If the original gate accepts, the repair returns its context unchanged, so both gates equal one. If the original gate rejects, the repaired gate is either zero or one. Hence $a_\theta \leq a_\theta^+$ pointwise. This is a statement about the raw gate before policy certification.

To see why it does not control selective risk, consider two equally likely types of question. The original policy admits only the first type and has zero paired degradation on it. Suppose repair admits the second type, on which the baseline is correct and the repaired reader is wrong. The repaired conditional degradation risk is then $1/2$. Raw admission increased, but the repaired policy may fail calibration while the original certifies.

Nor is the penalized objective preserved. In the running graph, two omissions priced at one unit each have penalized cost two and fail the gate. A complete repaired graph costs 140. The repair can be useful at a budget of 145 while increasing the original penalized objective. Consequently the additive approximation proof cannot be transferred merely by noting that repair found a complete context. Theorem 3 instead applies to both policies when they are fixed together before calibration and tested with the joint family correction. □

The fixed search does not make this a budget-optimal algorithm. It can miss a feasible context, and allowing a larger budget can alter the beam ordering and return a more expensive witness. The retained search traces expose both limitations. The safeguard's approximation certificate is recorded before repair. A graph lower bound may be computed for any repaired feasible graph under its declared objective, but is not automatically an optimality certificate for the separate serialized-token constraint.

C STATISTICAL GUARANTEES AND THEIR LIMITS

C.1 A COUNTEREXAMPLE TO BINWISE POST-SELECTION VALIDITY

Proposition 5 (Selection within a calibrated bin)**.** *A valid bin-average upper confidence bound need not bound the error probability of an edge selected using additional attributes, even when calibration edges are independent and identically distributed.*

Proof. Let Z identify a cheap type with probability 0.1. All edges have the same score. Conditional on $Z = 1$, error is Bernoulli with parameter 0.8. Conditional on $Z = 0$, its parameter is 0.01. Independent calibration draws have marginal error probability $0.1(0.8) + 0.9(0.01) = 0.089$. A consistent binomial upper confidence bound converges to 0.089 as sample size increases. A selector that accepts only $Z = 1$ has conditional error 0.8. Therefore the valid bound on the bin average fails for the selected subgroup. There is only one bin, so a correction for multiple bins does not repair the argument. □

The same construction can give calibrated confidence scores by assigning confidence 0.911 to every edge. The problem is not merely a miscalibrated confidence value. It is the difference between conditioning on a score and conditioning on everything used by the selection policy.

C.2 WHOLE-POLICY CALIBRATION

Condition on the training data and every random choice used to construct the finite policy family. For each policy, let (A_j, L_j) denote admission and binary loss on calibration question j. The policy is fixed, so these pairs are independent draws from a common distribution. Different policies on the same question can be arbitrarily dependent.

Lemma 3 (Conditional binomial count)**.** *For a fixed policy with positive population admission probability and selective risk R, conditional on $N = \sum_j A_j = k$, the number of admitted errors has distribution* $\mathrm{Binomial}(k, R)$.

Proof. Condition first on the entire admission vector $(A_1, \ldots, A_n)$. Since the original question-level observations are independent, conditioning on their separate admission events preserves independence across questions. Every admitted loss has the conditional Bernoulli distribution with mean $\Pr(L = 1 \mid A = 1) = R$. Thus the sum of the k admitted losses is binomial with parameters (k, R). This distribution does not depend on which k positions were admitted. Averaging over admission vectors with the same count proves the claim. □

Proof of Theorem 3. For each fixed policy θ and each positive admitted count, the one-sided Clopper–Pearson construction applied to Lemma 3 satisfies

$$\Pr\{R_\theta > U_\theta \mid N_\theta = k\} \le \delta/K. \tag{21}$$

When $N_\theta = 0$, the procedure sets $U_\theta = 1$ and never certifies it for $\alpha < 1$. Averaging Eq. (21) over the admitted count gives the same upper bound on the unconditional failure event. A union bound over the fixed K policies yields

$$\Pr\{\exists\theta \text{ with } R_\theta > U_\theta\} \le \delta. \tag{22}$$

On the complementary event, every policy with $U_\theta \le \alpha$ has selective risk at most α. Any choice among the certified policies, including one minimizing cost estimated on the same calibration questions, lies inside this simultaneous event. Independence across policies is unnecessary. Finally average over the training data and family-construction randomness, which are independent of calibration, to obtain the stated probability. □

The result is a high-probability statement about population selective risk. It is not a per-question correctness guarantee and does not guarantee that the observed error rate on every future finite test sample is below α. Independence of questions is stronger than exchangeability. General exchangeable but dependent labels do not automatically justify the binomial inversion used here.

C.3 FALLBACK AND PAIRED LOSS

Proof of Corollary 1. For one question, let $A = a_\theta(Q)$. The deployed error is $AE_\theta + (1 - A)E_0$. Therefore

$$E_{\mathrm{deploy}} - E_0 = A(E_\theta - E_0) \le A\mathbf{1}\{E_\theta = 1, E_0 = 0\} = AD_\theta. \tag{23}$$

Taking expectations gives an excess error at most $\Pr(A = 1)\Pr(D_\theta = 1 \mid A = 1)$. On the simultaneous calibration event, the second factor is at most α for the selected policy. This proves Eq. (10). If no policy is certified, deployment always uses the baseline and the excess error is zero by identity. □

For a fixed policy with positive admission probability, let

$$b_\theta = \Pr(E_0 = 0 \mid a_\theta = 1), \qquad h_\theta = \Pr(E_\theta = 1 \mid a_\theta = 1, E_0 = 0). \tag{24}$$

When $b_\theta > 0$, the calibrated paired risk is $R_\theta = b_\theta h_\theta$. Consequently $R_\theta \le \alpha$ gives only $h_\theta \le \min\{1, \alpha/b_\theta\}$, not $h_\theta \le \alpha$. When $b_\theta = 0$, h_θ is undefined and $R_\theta = 0$ supplies no reassurance about baseline or compressed accuracy. Likewise,

$$\Pr(E_\theta = 1 \mid a_\theta = 1) = R_\theta + \Pr(E_\theta = 1, E_0 = 1 \mid a_\theta = 1). \tag{25}$$

The second term is uncontrolled by a paired-risk certificate. These identities follow by partitioning admitted questions according to the baseline's error indicator.

A complementary paired event is $I_\theta = \mathbf{1}\{E_\theta = 0, E_0 = 1\}$. Pointwise, $E_\theta - E_0 = D_\theta - I_\theta$. Thus the deployed excess error is admission probability times the difference between conditional harm and improvement probabilities. We report both events and the full paired outcome table. A small observed degradation rate is not interpreted as high absolute accuracy when both readers are frequently wrong.

The full-context reader must be the same baseline in calibration, evaluation, and fallback. Comparing against a different prompt or stronger model changes the paired endpoint. Independent model revisions are also different policies and require a corresponding family correction or fresh calibration. The policy family in the supplied implementation is fixed before either calibration losses or held-out losses are observed.

Sample size and empty admission. With no admitted errors, Eq. (8) simplifies to

$$U_\theta = 1 - (\delta/K)^{1/N_\theta}. \tag{26}$$

The smallest admitted count is the ceiling in Eq. (11). The original family had 21 policies and required 299 admissions with no failures. The augmented family has 42 policies because each selector–budget pair appears with and without repair. It requires 334 admissions. Both variants enter the same confidence allocation, including candidates that subsequently fail certification. No reader outcomes have been observed under either protocol.

Fixed policies	Zero failures	One failure	Two failures	Five failures
1	149	236	313	523
7	245	349	439	677
21	299	410	506	757
42	**334**	448	547	805

Table A.1: Minimum admitted calibration counts at risk 0.02 and confidence error 0.05. Binomial inversion determines the counts. The bold entry applies to the augmented family, not the earlier 21-policy protocol.

The 334-admission boundary corresponds to 33.4%, 16.7%, and 11.13% of calibration samples of sizes 1,000, 2,000, and 3,000. These are necessary best-case fractions, not predicted admission rates. The revised protocol fixes 3,000 calibration questions. It does not increase the sample until certification succeeds or discard unsuccessful policies. At $N = 0$, the descriptive selective risk is undefined and the operational upper bound is one. A fallback-only policy inherits the baseline's absolute error, not zero absolute error.

D A WORKED GRAPH AND A SEPARATE ANSWER CHECK

Consider the following constructed query. Ada's archive is held by North College, whose city is Paris. The two requirements are to identify the archive holder and its city. The graph has vertices r, d, s_1, s_2, p and physical edges

$$(r, d, 100), \quad (d, s_1, 20), \quad (d, s_2, 20), \quad (r, p, 150). \tag{27}$$

Each triple gives the source, destination, and resource charge. The predicted terminal groups are $T_1 = \{s_1, p\}$ and $T_2 = \{s_2, p\}$. The sentence fragments include their shared header d. The paragraph fragment p is a standalone source and is predicted to support both requirements.

The source at s_1 states the archive-holder relation, and s_2 states that North College is in Paris. The paragraph p instead gives the same archive-holder relation but names Lyon as the city. This intentionally incorrect association is invisible to the declared group memberships. A deterministic two-relation reader follows the archive-holder relation and then the city relation in the selected source. It returns Lyon from p and Paris from s_1, s_2. This is a constructed semantic check, not inference by a pretrained model.

Selection	Additive charge	Union cost	Declared recall	Answer match
Shared sentences s_1, s_2	240	140	1	1
Paragraph p	150	150	1	0

Table A.2: The graph objective and constructed answer label are different evaluation targets. The additive and shared greedy rules both select the paragraph in this example.

The additive optimum is 150. The graph optimum is 140, as direct enumeration of the two inclusion-minimal covers shows. Each greedy rule starts with no retained resource. The paragraph's cost per newly covered group is 75, compared with 120 for either sentence, so both select p and terminate. The safeguard therefore returns cost 150. It cannot exploit every possible shared-cost improvement.

The flow relaxation has value 140. A primal witness sets both sentence paths to one. A lower-bound witness can be checked directly. Write $t = x_{rp}$ and $x_{rd} = h$. The two groups require $x_{ds_1} + t \geq 1$, $x_{ds_2} + t \geq 1$, and $h + t \geq 1$. Hence every feasible relaxation has cost at least

$$100(1-t) + 20(1-t) + 20(1-t) + 150t = 140 + 10t \geq 140. \tag{28}$$

The selected graph therefore has certified ratio $15/14$, while its constructed answer is wrong. The supplement includes the solver's dual witness and an independent check of this example. This demonstrates both the meaning and the limit of the graph certificate.

E Comparison with risk-controlled retrieval

C-RAG analyzes generation loss and selects retrieval-augmented generation configurations satisfying a risk target (Kang et al., 2024). Its bounded-loss formulation is broad, so changing the endpoint to paired degradation is not by itself a new risk-control framework. Our use of independent calibration and a simultaneous family correction is also standard. The distinct optimization question is how shared context should be charged and what can be certified about the resulting graph problem.

Conformal context filtering targets the retention of a labeled-relevant query–snippet pair (Chakraborty et al., 2026). Its positive-example calibration determines a snippet-score cutoff. That target differs from question-level paired reader degradation. Losing a relevant snippet need not change an answer, and retaining every annotated snippet need not make the reader correct. Neither guarantee implies the other without an additional relation between annotations and reader behavior.

A numerical comparison of these approaches requires more than matching a method name. A snippet filter must be fitted using calibration data separate from any second-stage reader-risk calibration, and its scorer, threshold, labels, and reader must be fixed before that second stage. A C-RAG comparison must preserve the intended loss, configuration family, and generation protocol. Neither full system has been rerun here. The shared-candidate baseline experiment in this paper is therefore not presented as such a comparison.

F Experimental specification and additional results

F.1 Graph inputs, baselines, and independent labels

There are 67 vertices and 66 physical edges in each controlled graph. The root connects to six hubs. The remaining 60 vertices each attach to one hub, with attachment determined cyclically rather than from the target label. Each of the first 48 evidence vertices receives one to three distinct true requirement indices drawn uniformly. For each true membership, deletion occurs independently with probability η. Each false membership is added independently with probability $\eta/6$. The final

Dimension	C-RAG	Context filtering	This formulation
Controlled quantity	Bounded generation loss	Labeled-relevant snippet retention	Paired reader degradation among admitted questions
Selected object	A tested RAG configuration	A score-filtered snippet set	A shared-resource graph and a reading policy
Calibration unit	Input–output examples	Relevant query–snippet pairs	Independent questions with paired reader outcomes
Resource model	Not a shared-edge optimization certificate	Retained snippet context	Edge-union cost distinct from additive charges
Dependencies	Retrieval and generation configuration	Snippet-score threshold	Rooted, dependency-closed read-only fragments
Deployment rule	Deploy a tested configuration	Retain snippets above the cutoff	Full context whenever compression is rejected

Table A.3: Scope of the compared formulations. Entries identify the mechanisms being compared rather than ruling out extensions of any prior framework. No performance ranking or reproduction of either prior system is asserted.

12 evidence vertices are costly singleton candidates, one for each requirement, with unchanged memberships. These ensure that optimization failure is not confused with a deliberately infeasible declared instance.

Root costs are bW for W uniform on $[0.75, 1.25]$. Ordinary evidence-edge cost is log-normal with log mean zero and log standard deviation 0.5. Singleton evidence-edge cost is 9. Exogenous ordinary scores are beta distributed with parameters 5 and 2, and singleton scores are 0.3. These scores are not calibrated probabilities. They are supplied equally to methods that use scores. Coverage and group memberships are identical across methods within each instance.

The five selectors differ only in the decision rule described in Section 3. Maximum coverage uses new-group count, then charge and identifier to break ties. Score-weighted coverage divides charge by score times new-group count. Union-cost greedy uses the incremental cost of newly selected resources. CRESS-S chooses the better union-cost output of additive and union-cost greedy, without access to true support labels. All methods resolve every predicted group, so that statistic is not used as efficacy evidence. Independent true recall counts the fraction of latent requirements supported by at least one selected evidence node.

MILP runs use binary physical-edge and omission variables with continuous per-group flows. The requested relative MIP gap is 10^{-9} and the time limit is 30 seconds per problem. Every graph and fragment problem in the reported grid returned optimal solver status. The LP lower bounds are separately validated by Lemma 2. The supplement contains the underlying instances and results.

Uncertainty summaries. The tables average each instance's ratios, not a ratio of aggregate objectives. The percentage cost reduction uses the ratio of mean union costs. For its interval, the bootstrap resamples the ten seed identifiers, keeping all sharing and noise conditions of a seed together. There are 5,000 bootstrap replicates with bootstrap seed 2718. This accounts for the use of common random numbers across the grid. It is an uncertainty summary for these controlled instances, not a population confidence statement about public QA.

Comparator	CRESS-S mean cost reduction	Seed-cluster interval
CRESS additive	18.5%	[13.3%,23.3%]
Score-weighted coverage	28.4%	[23.9%,32.2%]
Maximum coverage	21.2%	[17.6%,24.8%]
Union-cost greedy	2.3%	[1.1%,3.8%]

Table A.4: Paired cost comparisons on the new 120-instance grid. The small gain against union-cost greedy isolates the safeguard's contribution. Intervals resample seed identifiers and retain their complete parameter grids.

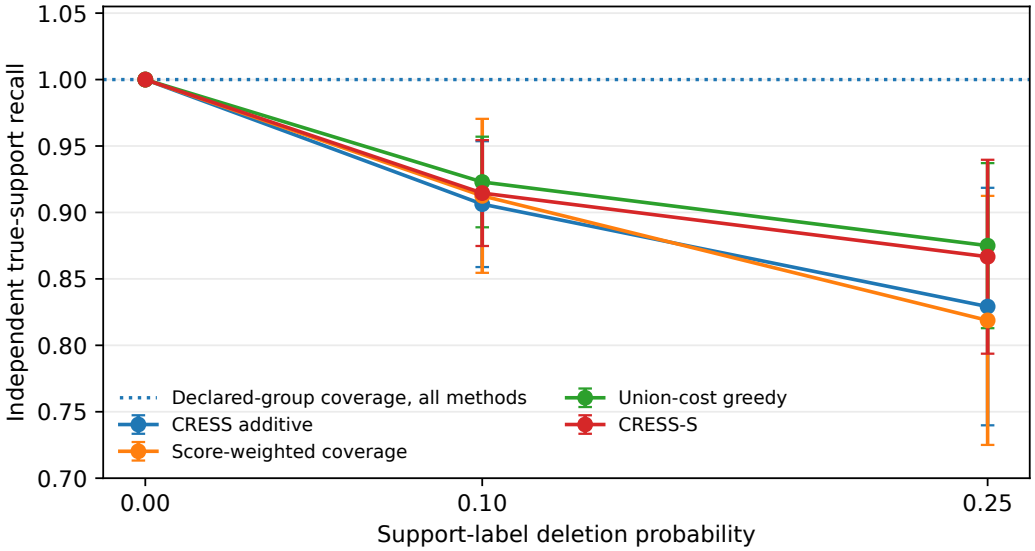


Figure A.1: Predicted coverage is not semantic sufficiency. Every method covers all declared groups, but latent support recall falls when membership predictions are corrupted. Error bars summarize ten seed means, each averaging the four sharing multipliers. The safeguard improves declared cost, not guaranteed true recall.

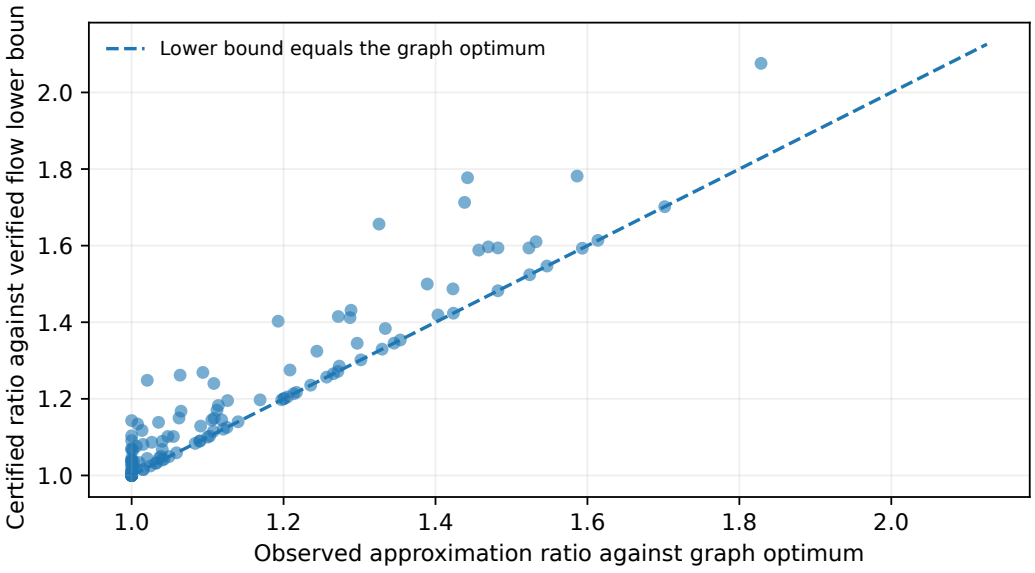


Figure A.2: A posteriori graph certificates for CRESS-S. Each point is one controlled graph. The vertical coordinate uses the rationally validated LP lower bound, whereas the horizontal coordinate uses the MILP reference optimum. Points lie on or above the diagonal as required.

F.2 CERTIFICATE TAILS AND WEAK LOWER BOUNDS

Each selector has 120 certificate ratios. We classify a positive bound as near zero when $\underline{L}_G \leq 10^{-8} \max\{1, F(H)\}$. No bound is zero or near zero in this controlled family. The smallest normalized bound $\underline{L}_G / F(H)$ for CRESS-S is 0.482. This is a property of these instances, not a uniform lower-bound quality guarantee.

Selector	Median	90th	95th	Maximum	Above 2
Maximum coverage	1.340	2.057	2.243	2.564	14
Score-weighted	1.463	2.216	2.484	3.271	24
CRESS additive	1.281	1.947	2.220	2.564	11
Union-cost	1.156	1.596	1.686	2.076	1
CRESS-S	**1.132**	**1.589**	**1.616**	**2.076**	**1**

Table A.5: Distribution of $F(H)/\underline{L}_G$ over 120 instances per selector. Quantiles are empirical, not confidence limits. All methods use the same predicted graph lower bound. Bold highlights the safeguarded selector.

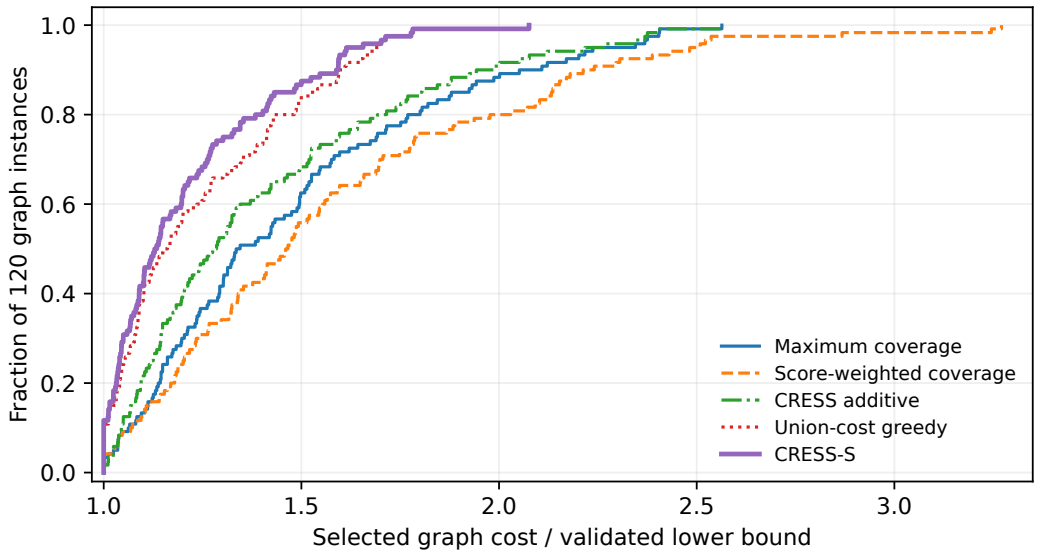


Figure A.3: Empirical distributions of graph-cost certificates. Every one of the 120 observations per method appears in the distribution, including the largest ratios. A smaller value is a tighter optimization certificate and is not evidence of a more reliable answer.

We also construct examples where the lower bound is weak. Start with n set vertices reached from the root by edges of cost one. For each k-element subset of the set vertices, add a required terminal reached by zero-cost edges from precisely those k vertices. Selecting set vertices must hit every k-subset. The integral optimum is $n - k + 1$, while the flow-relaxation optimum is n/k.

Values of the subset example. An integral selection fails if its unselected vertices contain a k-subset. Thus at least $n - k + 1$ set vertices must be purchased, and any such selection succeeds. For the relaxation, purchasing each root edge to extent $1/k$ lets every terminal route $1/k$ through each of its k predecessors. This costs n/k. Conversely, each terminal requires the sum of its predecessor purchase variables to be at least one. Summing these constraints over all k-subsets counts each variable $\binom{n-1}{k-1}$ times and gives $\sum_j x_j \geq \binom{n}{k} / \binom{n-1}{k-1} = n/k$. The displayed fractional solution is therefore optimal. □

The shared-bridge graph has a 10-unit root edge, a one-unit connector, and two one-unit edges from the connector to the required terminals. Sharing gives cost 13. Deleting the connector disconnects

Constructed instance	Selected	OPT_G	$\underline{L}_G$	Ratio
Shared bridge	13	13	13	1.0
Subset $(4, 2)$	3	3	2	1.5
Subset $(6, 3)$	4	4	2	2.0
Subset $(8, 4)$	5	5	2	2.5
Zero-cost graph	0	0	0	Undefined
Bridge with zero witness	13	13	0	Undefined

Table A.6: Constructed checks of connectivity and bound strength. Subset rows use an optimal integral selection, so their ratios expose relaxation weakness rather than selector error. The last row uses a deliberately weak zero dual witness, not the optimized relaxation.

both requirements even if their terminal vertices are retained. This checks rooted connectivity rather than terminal presence alone. It does not show that a language-model reader needs a particular semantic bridge. That claim requires public-task cases with actual reader outputs.

F.3 Common repair across all selectors

This diagnostic changes the omission penalty to four units and applies budgets of 12, 24, 48, and 96 resource units to each of the 120 existing graphs. The choice is fixed for all selectors. It deliberately exercises the mismatch between the penalized objective and a gate requiring every predicted group. It is not a model of actual token budgets. The search uses the same 60 real candidates, corrupted memberships, and resource costs for every method. No latent support label or optimization oracle enters the search.

There are 480 distinct graph–budget searches. Of these, 412 return a feasible complete witness and 68 do not. The independent graph oracle identifies 413 feasible combinations. Thus the finite search misses one feasible combination, shared by the method comparisons at that graph and budget. Admission after repair also preserves any already-accepted context, so the relevant comparison is paired by graph, budget, and selector.

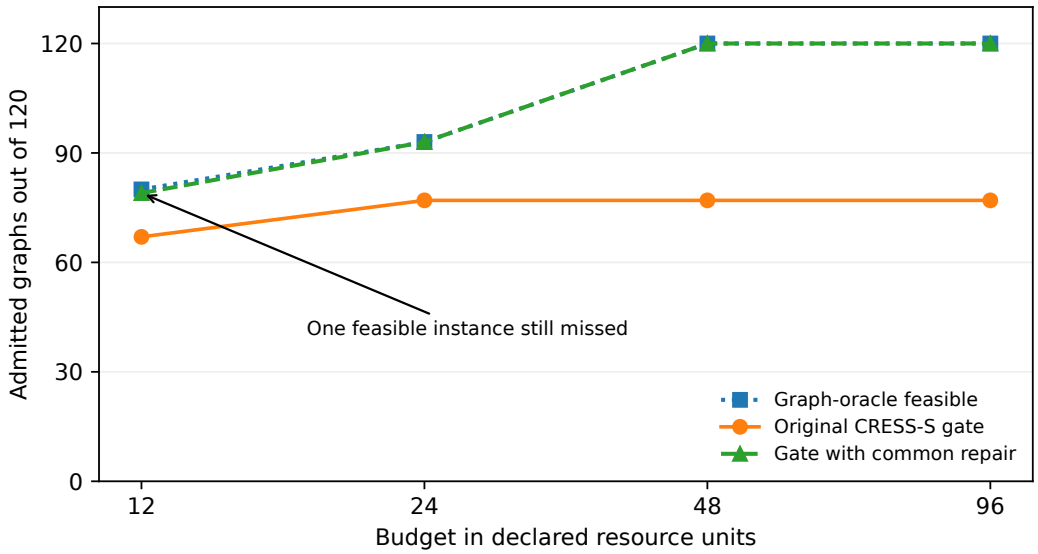


Figure A.4: Declared budget feasibility and common repair for CRESS-S. The integer-program count is an upper limit on admission, not an input to the search. At the smallest budget the repaired method still misses one feasible instance. The full table reports the same repair for every selector.

The search changes the comparison as well as improving admission. At budget 48, original admission ranges from 62 to 78 graphs across methods, while every repaired variant admits all 120. The cost

Selector	Budget	Admission	Feasible	Deployed cost	True recall
Max. coverage	12.0	59/79	80	0.597/0.448	0.895/0.905
Max. coverage	24.0	67/93	93	0.540/0.337	0.892/0.901
Max. coverage	48.0	68/120	120	0.530/0.076	0.891/0.901
Max. coverage	96.0	72/120	120	0.496/0.082	0.884/0.897
Score-weighted	12.0	60/79	80	0.591/0.449	0.908/0.914
Score-weighted	24.0	62/93	93	0.576/0.336	0.910/0.913
Score-weighted	48.0	62/120	120	0.576/0.075	0.910/0.911
Score-weighted	96.0	62/120	120	0.576/0.077	0.910/0.911
CRESS additive	12.0	61/79	80	0.582/0.447	0.923/0.926
CRESS additive	24.0	66/93	93	0.545/0.335	0.917/0.919
CRESS additive	48.0	66/120	120	0.545/0.074	0.917/0.916
CRESS additive	96.0	66/120	120	0.545/0.076	0.917/0.916
Union-cost	12.0	67/79	80	0.536/0.447	0.933/0.936
Union-cost	24.0	78/93	93	0.455/0.335	0.932/0.934
Union-cost	48.0	78/120	120	0.455/0.074	0.932/0.927
Union-cost	96.0	78/120	120	0.455/0.076	0.932/0.927
CRESS-S	12.0	67/79	80	0.536/0.446	0.928/0.931
CRESS-S	24.0	77/93	93	0.462/0.334	0.923/0.927
CRESS-S	48.0	77/120	120	0.462/0.073	0.923/0.922
CRESS-S	96.0	77/120	120	0.462/0.075	0.923/0.922

Table A.7: All paired common-repair configurations. A slash separates original and repaired values. Each row has 120 graphs. Admission and feasible columns are counts. Deployed cost is the per-graph cost divided by full-context resource cost, then averaged, including fallback. True recall is conditional on the corresponding admission gate. These observations share ten seeds and are not 2,400 independent draws.

ratios then lie between 0.073 and 0.075. Thus it would be misleading to attribute the recovered admission to CRESS-S alone. At the same budget, its latent true-support recall falls slightly from 0.923 to 0.922, and union-cost greedy falls from 0.932 to 0.927. Complete predicted coverage is not complete semantic support.

The full-context cost in this diagnostic is the sum of all resources in the common candidate graph. Selected evidence and fallback are both evaluated in those same declared units. No real reader is called, so the table does not estimate paired answer risk or a calibrated deployment frontier. In particular, the repaired variant is not declared safe merely because it has a feasible graph witness. The public-reader comparison must recalibrate both variants together.

F.4 Calibration repetitions and known population quantities

The threshold grid has 16 points, $0.10 + 0.06j$ for $j = 0, \ldots, 15$. Each calibration repetition samples fresh independent U, V values. The empirical mean cost used to choose a certified policy includes 1,000 units for each rejected compression. If the accepted threshold is t, its population admission probability is t and integration gives

$$R(t) = \frac{1}{t}\int_0^t (0.02+0.5u^2)\,du = 0.02+\frac{t^2}{6}, \qquad C(t) = \int_0^t (200+100u)\,du+1000(1-t). \quad (29)$$

Thus risk violations are checked against known population risk, not against a noisy empirical test rate. Repetitions with no certified policy are counted in admission and cost summaries but have undefined selective risk. The reported mean risk among nonempty certified policies is 0.050, 0.059, and 0.079 for sample sizes 256, 1,024, and 4,096, respectively. The fraction of calibration repetitions violating the target is 0.6%, 0.2%, and 0%. These fractions estimate procedure behavior in this constructed distribution and do not replace the finite-sample theorem.

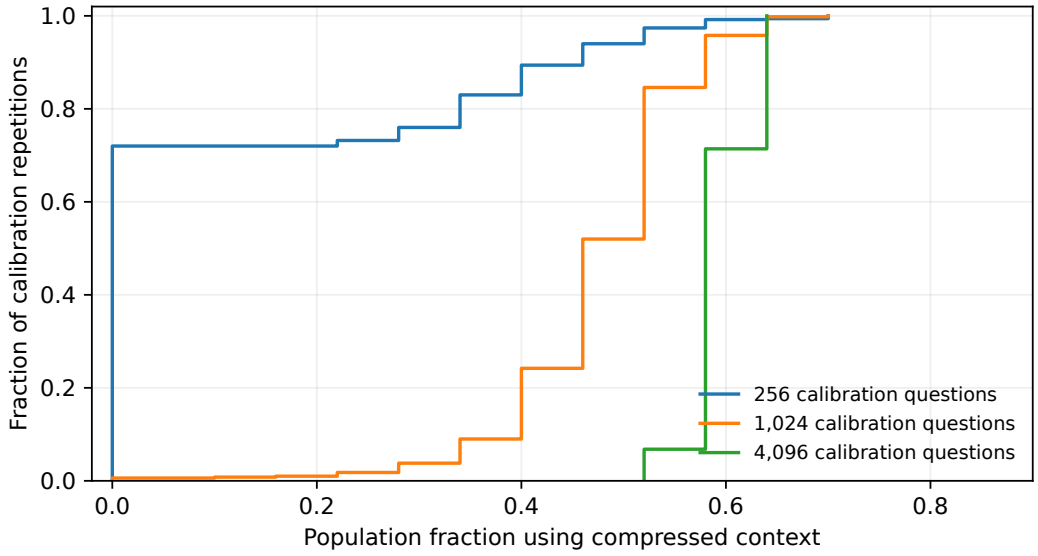


Figure A.5: Distribution of admission across 500 calibration repetitions at each sample size. The jump at zero records fallback-only policies rather than zero-error predictions. Larger calibration samples permit more compression in the specified simulation.

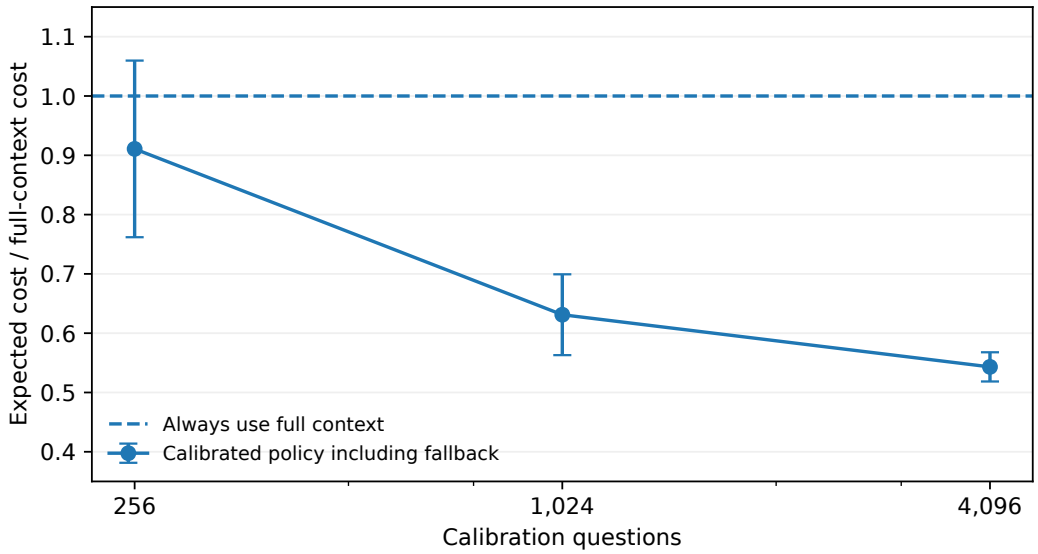


Figure A.6: Expected deployment cost, including fallback. Error bars show standard deviations across calibration repetitions. Units are normalized by the full-context cost of 1,000. These are synthetic population costs, not measured language-model token savings.

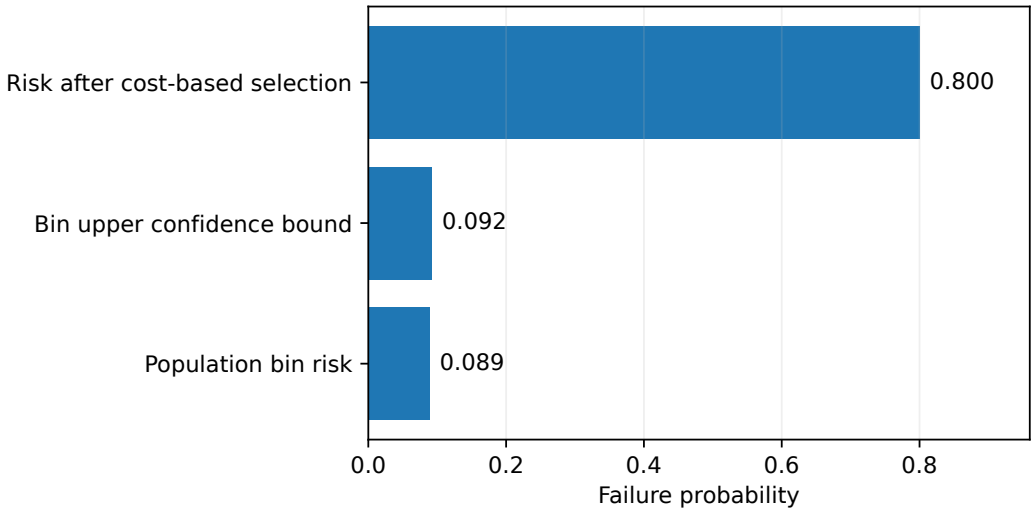


Figure A.7: Failure of bin-average calibration after cost-based selection. The 50,000-edge calibration sample produces a pooled upper bound of 0.092. That valid population-bin bound does not cover the cheap selected subgroup's 0.8 error probability.

G PUBLIC QA STUDY

Data and partitions. The proposed study uses the official HotpotQA training and distractor-development sets, retaining every candidate paragraph and sentence identifier. The candidate contexts are not restricted to annotated supporting facts, and no synthetic distractors are added. A seeded assignment based on question identifiers defines disjoint partitions. Seed 17 selects 5,000 fitting questions from training, 3,000 calibration questions from development, and a separate 1,000-question evaluation subset from development.

Input-label separation. The selector receives the question, candidate sentences, document titles, sentence indices, a frozen scorer, policy parameters, and a tokenizer. Reference answers and supporting facts are held separately. Supporting-fact labels are used to fit the scorer on training questions. On calibration and evaluation questions, labels enter only after selection and reading.

Learned support and query representation. The default support model is logistic regression trained on sentence-support labels. Features include query-sentence TF–IDF interactions, lexical overlap with sentence and title, sentence position, and length. The vocabulary is fitted only on training evidence. The maximum feature count is 40,000, with unigram and bigram features. The regularization parameter is 1.0 and the iteration cap is 400. It is a modest supervised support model, not a claimed new neural retriever.

Up to four query facets are selected from question terms using training-set inverse document frequency. A sentence belongs to a facet group when it matches the term in its title or text and its support score is at least 0.15. This representation deliberately does not use answer entities, gold decomposition steps, or gold supports at evaluation time. It can miss relational requirements, coreference, or paraphrases. Such misses are measured through reference-support recall and downstream answer loss, not treated as infeasible ground truth. Richer learned query decompositions would require their own training and evaluation.

Graph and cost construction. The root connects to each document, and document nodes connect to their sentences. Header and sentence resources are charged separately. Each sentence fragment includes its header, and each paragraph fragment includes all of its sentence resources. Omission cost is 200 declared units. Standalone tokenization of individual resources defines the additive resource model. Because tokenization across concatenation boundaries need not be additive, the actual serialized context is tokenized again for admission and reported cost. The approximation

theorem concerns the declared resource model, not an unsupported identity between separate and concatenated token counts.

Policies and comparisons. The fixed family contains seven selectors at budgets of 256, 512, and 1,024 serialized evidence tokens, each with and without the common 32-state repair. All 42 candidates enter a single calibration family. The selectors are CRESS, CRESS-S, score-weighted coverage, descending support score, support score per incremental token, maximal-marginal-relevance with mixing weight 0.7, and a rooted prize-collecting Steiner operator on the common document–sentence graph.

Every selector uses the same admission rule. The context must be nonempty and within budget, the predicted facet set must be nonempty, every predicted facet must be covered, and the common confidence threshold must be met. The repaired variant preserves an admitted original context and otherwise applies the search in Appendix B. The scorer, candidate collection, and calibration procedure are identical across methods. The PCST comparator uses learned score prizes, without a graph neural network or a trained graph-to-language prompt. It is a selection operator, not the full G-Retriever system (He et al., 2024).

Reader and output contract. The reader is a fixed revision of Qwen2.5-3B-Instruct (Qwen Team, 2024). Generation is greedy with at most 96 new tokens. Every policy uses the same prompt, which requests a short answer with sentence-identifier citations and `noanswer` when the evidence is insufficient. The input limit is 16,000 tokens. Contexts exceeding that limit are rejected rather than silently truncated. The reader, prompt, decoding rule, numerical settings, and software versions remain fixed throughout calibration and evaluation. Predictions may be reused only under identical inference conditions.

Loss, calibration, and final evaluation. The calibration target is paired answer degradation at $\alpha = 0.02$ and $\delta = 0.05$, with absolute answer mismatch available as a separate loss. Theorem 3 governs selection from the prespecified family. A global policy, one policy per selector, and one policy per selector–repair pair are chosen by calibration cost alone. Held-out outcomes cannot alter the family, admission thresholds, budgets, scorer, reader, or these choices. Changing any component requires new calibration.

The comparison retains all 42 candidates, including those that fail certification. It distinguishes compressed and baseline accuracy among admitted questions from deployed and baseline accuracy over all questions. The four paired correctness outcomes separate degradation from improvement. Admission, fallback, supporting-fact recall, citation precision, selected evidence tokens, and total deployment tokens are reported separately. With no certified policy, the method retains full context. Its selective risk is undefined, its excess error is zero by identity, and its deployment-token ratio is one. Baseline errors remain visible.

Computation is accounted for at the question and policy levels. Scoring is charged once per question. Selection time includes repair, whose time is also reported as a component rather than added twice. Full-context predictions are reused across policies, and identical compressed prompts share one reader prediction. Actual inference calls and elapsed time count each computation once. Gate rejections, admitted cache hits, baseline calls, and new reader calls remain distinct, as do the original costs of reused predictions. The reader policy makes no graph-LP calls. Its LP time is therefore zero, while relaxation costs belong to the separate optimization study.

Before non-admission and prediction reuse, the 42-policy calibration family permits at most $42 \cdot 3000 + 3000 = 129{,}000$ reader contexts. The corresponding upper count for 21 policies is 66,000. These are planning bounds, not measured costs. Savings in selected tokens need not imply savings in total computation.

Answer match and token F1 are evaluated independently of predicted coverage. Supporting-fact recall and precision use the reference annotations. Proposal recall measures reference support present in the common candidate set, and conditional selector recall measures how much available support is retained. A valid citation identifier establishes that the cited sentence was present, not that it entails the answer. The comparison retains the selected sources, predicted facets, omissions, and reader outputs so that these quantities can be distinguished.

A paired bootstrap resamples questions while preserving all policy outcomes for each question. Its intervals describe held-out uncertainty and do not replace the calibration guarantee. Conditional quantities with zero admission remain undefined. The supplement contains the implementation and full study specification.

Status and remaining comparisons. The public-data reader study has not been run. The completed software tests and controlled experiments do not establish pretrained-reader performance or public-task accuracy. No HotpotQA answer-quality table or reader success case is reported here. A completed study should include both successful multi-hop cases and failures in which lexical coverage omits a necessary relation, showing the original and selected contexts, predicted facets, reference supports, and both reader outputs. Constructed bridge examples do not substitute for those observations.

A second reader and a second benchmark such as MuSiQue (Trivedi et al., 2022) would test robustness beyond this setting. A full G-Retriever comparison, a SteinLib optimizer comparison, and a human failure-localization study also remain open. Pretrained-model loading, public-data reader inference, and the optional PCST integration have not been tested here.

H Archived implementation measurements

The archived implementation study contains CPU runs on compressed sparse-row fragment pools and accelerator runs on a single NVIDIA A100-SXM4-40GB. These earlier measurements are separate from the graph and calibration experiments reported in the main text.

Measurement	Groups	Candidates	Seconds	Completion
CPU CSR	100,000	600,000	13.666 ± 0.399	Full expanded cover
CPU CSR	150,000	900,000	21.092 ± 0.804	Full expanded cover
CPU CSR	200,000	1,200,000	27.932	Full expanded cover
A100 tensor sweep	16,384	131,072	0.515 ± 0.210	1,024 steps
A100 tensor sweep	32,768	262,144	0.803 ± 0.009	2,048 steps
A100 tensor sweep	65,536	524,288	1.714 ± 0.015	4,096 steps

Table A.8: Archived implementation timings. CPU rows have three, three, and one seeds. GPU rows each have three seeds. The two backends use different workloads and stopping rules, so these values are not a CPU-to-GPU speedup comparison.

The largest GPU record reports 78.563 MiB of allocated tensor memory, not total device memory. Its 4,096-step cap covers about half the 65,536 groups, so 1.714 seconds is not a full-cover time. The historical generator samples memberships with replacement and counts repeated identifiers in its gain. Its timings remain measurements of that program, but its gains do not implement the weighted-cover rule. The corrected kernel deduplicates gains and passes CPU parity checks. It has not been rerun on CUDA.

Hardware timings do not establish answer quality or negligible optimization overhead. A deployment comparison must include graph construction, scoring, selection, calibration, prompt processing, generation, and fallback. The archived CPU and GPU workloads also differ and cannot support a speedup ratio.

I Supplement

The supplement contains the code and data for the reported experiments.

J AI use statement

Generative AI assisted in literature search, grammatical and structure revision, and figure construction. The authors assume responsibility for the content.