ARGOS: AUTOMATED FUNCTIONAL SAFETY REQUIREMENT SYNTHESIS FOR EMBODIEDAI VIA ATTRIBUTE-GUIDED COMBINATORIAL REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Ensuring functional safety is essential for deploying Embodied AI in complex open-world environments. Traditional Hazard Analysis and Risk Assessment (HARA) relies on enumerating risks for finite, pre-defined function lists, yet Embodied AI operates on open-ended natural language instructions whose combinatorial interactions give rise to unbounded hazard scenarios. Large Language Models (LLMs) offer a promising path to scalability, but without physical grounding they fall back on statistical correlations and overlook long-tail coupled risks. To address these limitations, we present **ARGOS** (**A**tt**R**ibute-**G**uided c**O**mbinatorial rea**S**oning), a two-stage pipeline that bridges open-ended instructions and physical constraints. Stage I decomposes entities into fine-grained physical attributes and performs bounded combinatorial reasoning to discover multi-factor hazard scenarios. Stage II integrates these scenarios with robot capabilities and regulatory standards to synthesize auditable Functional Safety Requirements (FSRs). Experiments across two backbone models show that ARGOS significantly outperforms intrinsic-knowledge and chain-of-thought baselines in hazard scenario quality (Cohen's d=0.62–1.00). ARGOS also achieves the highest Overall score against full-rulebook long-context and STPA/FMEA-structured baselines (8.57 vs. 7.78 and 7.57). Human evaluation further reveals a `Verbosity Trap`: chain-of-thought reasoning generates redundant scenarios that inflate automated metrics yet are penalized by human experts. Stage-II ablations show that physical grounding consistently improves requirement synthesis and expose a `Regulatory Mismatch` on the primary backbone when regulatory knowledge is applied without sufficient physical context. Together, these findings establish explicit attribute grounding as a critical prerequisite for reliable safety reasoning in Embodied AI. The source code is available at https://anonymous.4open.science/r/ARGOS-274E.

1 INTRODUCTION

The rapid evolution of Embodied AI is propelling robots from constrained industrial environments into open household environments. Models like PaLM-E Driess et al. (2023) and SayCan Brohan et al. (2023) enable robots to generalize across diverse tasks via natural language, yet this paradigm shift introduces unforeseen safety challenges. Unlike traditional robotic systems that operate within a finite set of pre-defined functions, Embodied AI is driven by open-ended user instructions, where robots must dynamically interpret arbitrary intents. The unstructured nature of domestic settings, compounded by this open-ended task space, leads to a combinatorial explosion of task-environment interactions that renders static, expert-based enumeration insufficient Lasota et al. (2017); Seshia et al. (2022).

To address these challenges, recent research Shi et al. (2024); Nouri et al. (2024) has explored using LLMs as "expert surrogates" to streamline safety documentation. However, these approaches operate within pre-defined function scopes and lack physical grounding. The root lies in the Symbol Grounding Problem Bender & Koller (2020): lacking embodied experience, models fail in physical attribute reasoning Wang et al. (2023) and dynamic state tracking Valmeekam et al. (2022), resorting to shortcut learning Geirhos et al. (2020) that overlooks long-tail coupled risks Kandpal et al. (2023). Crucially, they tend to treat risk factors in isolation. Consider a robot instructed to "serve hot soup to

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.

the dining table." A standard LLM-based analysis might flag a generic burn risk from the hot liquid, yet fail to deduce that the robot's emergency stop to avoid a running child would destabilize the liquid payload through inertial forces, producing a coupled burn-and-slip hazard that no single-factor analysis would surface. Discovering such multi-factor interactions requires reasoning grounded in the physical attributes of the entities involved: temperature, fluidity, container geometry, and locomotion dynamics.

To bridge this gap, we propose **ARGOS** (**A**tt**R**ibute-**G**uided c**O**mbinatorial rea**S**oning), a framework for automated functional safety requirement synthesis. As illustrated in Figure 1, ARGOS operates through a two-stage pipeline that moves from semantic parsing to physically-grounded specification. Stage I, **Attribute-Guided Hazard Discovery**, extracts risk-relevant entities from an open-ended instruction, grounds them in physical attributes retrieved from a curated Rule Base, and performs combinatorial reasoning to discover multi-factor hazard scenarios, including coupled risks that single-factor analysis overlooks. Stage II, **Scenario-Anchored Requirement Synthesis**, conditions the generated hazard scenarios on a Robot Capability List and regulatory standards (e.g., ISO 13482) to produce auditable, physically feasible Functional Safety Requirements (FSRs). We position ARGOS at the FSR lifecycle step of the functional safety lifecycle: it defines *what* safety constraints must hold, independent of implementation details and without requiring hardware access. FSRs then serve as input constraints for downstream Technical Safety Requirements (TSRs), which specify concrete hardware and software mechanisms. This makes FSRs the natural abstraction for LLM-based reasoning, as LLMs excel at deriving high-level functional logic but struggle with the precise hardware parameters required for TSR synthesis.

We validate ARGOS on 48 seed scenarios spanning diverse risk entities, tasks, and environments, expanded into 365 context-specific hazard scenarios. Across two backbone models, ARGOS improves hazard quality over intrinsic-knowledge and physics-aware chain-of-thought baselines, with medium-to-large effect sizes (Cohen's d=0.62–1.00). Under matched DeepSeek-V3.2 settings, ARGOS also achieves a higher Overall score than full-rulebook long-context and STPA/FMEA-structured baselines (8.57 vs. 7.78 and 7.57). Human evaluation further shows that CoT tends to produce more semantically redundant hazards, while Stage-II ablations show that physically grounded variants produce stronger FSRs than ungrounded variants; a controlled GPT-4o pilot preserves this grounding advantage.

In summary, we make three contributions:

- Unlike prior work that operates within pre-defined function scopes, we present an automated framework that synthesizes FSRs directly from unconstrained, open-ended user instructions, addressing the scalability bottleneck of traditional HARA for Embodied AI.
- We propose ARGOS, which shifts safety analysis from implicit label-to-label risk retrieval to explicit attribute-based deduction. Typed physical profiles serve as reasoning premises, and their bounded interactions expose inspectable multi-factor hazard hypotheses that are carried forward into hazard-to-FSR synthesis.
- We evaluate ARGOS across two LLM backbones, long-context and structured-safety baselines, and human experts. The results separate the effect of physical grounding from additional reasoning depth or additional context, and identify two recurring failure modes: the *Verbosity Trap* in CoT hazard generation and *Regulatory Mismatch* when regulatory context is used without sufficient physical grounding.

2 Related Work

2.1 Safety Engineering in the Open World

Safety assurance for personal care robots is governed by standards like ISO 13482 International Organization for Standardization (2014), with compliance traditionally relying on manual methodologies such as STPA Leveson (2016). However, the transition from structured industrial settings to unstructured household environments introduces the open-world challenge. Lasota et al. Lasota et al. (2017) and Seshia et al. Seshia et al. (2022) argue that the unpredictability of human behavior creates a combinatorial explosion of risk scenarios, rendering static, expert-based enumeration insufficient and necessitating automated safety mechanisms. Table 1 summarizes the safety development lifecycle steps and ARGOS's position within them.

The consequences of inadequate safety analysis are well-documented. Ma et al. Ma et al. (2026) demonstrate that failures in embodied AI arise not from isolated model flaws but from embodiment-induced system-level mismatches, where semantic correctness does not imply physical safety. In household settings, where robots operate near vulnerable populations (elderly, children, wheelchair users), the margin for error is minimal. This gap between the scale of potential risks and the capacity of manual analysis motivates the automated, reasoning-driven approach we pursue in this work.

Table 1: The functional safety development lifecycle. ARGOS operates at Lifecycle Step 2.

Lifecycle Step	Artifact	Core Question	Verification
1	Safety Goals	What must the system avoid?	Hazard analysis
2	**FSRs**	**What safety rules must exist?**	**Logical audit**
3	Technical Architecture	How to implement?	Architecture review
4	TSRs	What HW/SW mechanisms?	Physical testing

2.2 From Software Requirements to Embodied AI

In the broader software domain, LLMs have shown promise in automating requirements engineering Hou et al. (2024); Fan et al. (2023); Marques et al. (2024), though challenges remain in handling rigorous logic and domain-specific constraints Norheim et al. (2024); Hemmat et al. (2025). The transition to embodied AI further complicates this: models like PaLM-E Driess et al. (2023), SayCan Brohan et al. (2023), and Inner Monologue Huang et al. (2023) can ground language into affordances, yet the Symbol Grounding Problem persists Bender & Koller (2020); Bisk et al. (2020). Lacking embodied experience, models struggle with physical reasoning—benchmarks like NEWTON Wang et al. (2023) reveal failures in modeling attributes like mass and friction, while Valmeekam et al. Valmeekam et al. (2022) show deficits in tracking dynamic states. Ruis et al. Ruis et al. (2024) further suggest that pre-training data often lacks the procedural knowledge needed for such reasoning. This reliance on statistical correlations leads to shortcut learning Geirhos et al. (2020), causing models to overlook long-tail risks Kandpal et al. (2023)—precisely where the most dangerous hazards reside. In safety-critical contexts, this disconnect is particularly consequential Kalai & Vempala (2024).

Attribute-guided generation has been used to improve the diversity and coverage of synthetic training data. Yu et al. **?** condition generation on explicit attributes, while AIDE **?** uses attribute-guided multi-hop expansion for task-specific fine-tuning. ARGOS uses attributes for a different purpose: typed physical profiles act as premises for causal safety reasoning, and their interactions define the hazard hypotheses carried into FSR synthesis. This connects attribute-level representation to physical interaction analysis rather than training-data expansion.

2.3 Automated Hazard Analysis

Recent work has integrated LLMs into hazard analysis. Frameworks like Aegis Shi et al. (2024), Co-Hazard Analysis Diemert & Weber (2023), and prompt-layered approaches Iyenghar et al. (2025) use LLMs as "expert surrogates" to streamline safety documentation, while Nouri et al. Nouri et al. (2024) map hazards to regulatory frameworks. However, these methods remain tethered to pre-defined function scopes—taking static inputs (e.g., "Navigate", "Lift") and retrieving associated risks—which cannot handle the generative variability of open-ended natural language commands. Similarly, benchmarks like Safety Gymnasium Ji et al. (2023) provide standardized environments for safe reinforcement learning but operate with pre-defined safety constraints. ARGOS bridges this gap by dynamically decomposing instructions into attribute-level risk factors, shifting from implicit "Label-to-Label" retrieval to explicit Attribute-Based Deduction.

3 Methodology: The ARGOS Framework

We propose ARGOS, an Attribute-Guided Combinatorial Reasoning pipeline designed to bridge the gap between abstract user instructions and FSRs. Unlike end-to-end approaches that rely solely on the implicit, potentially ungrounded knowledge of LLMs, our framework adopts a structured, physically-anchored two-stage pipeline:

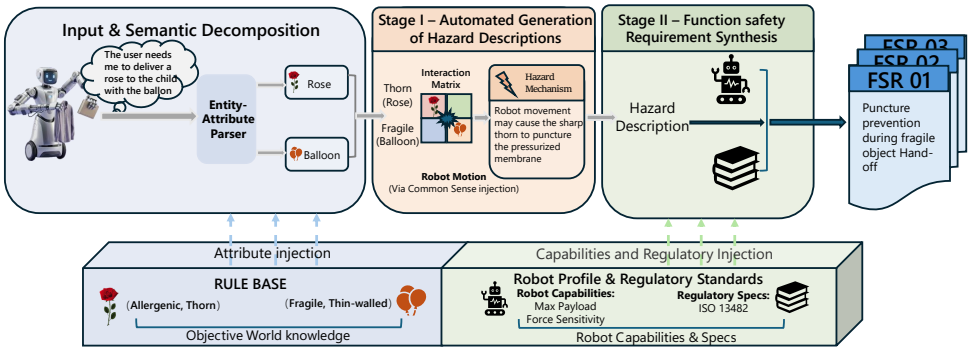


Figure 1: The proposed ARGOS framework: A two-stage pipeline for automated FSR synthesis. Stage I focuses on decomposing semantic entities into physical attributes for combinatorial hazard discovery, while Stage II aligns these hazards with regulatory standards and hardware constraints to generate requirements.

(1) **Attribute-Guided Hazard Discovery** (Section 3.1), which decomposes semantic entities from the instruction into granular physical attributes, then performs combinatorial reasoning to discover multi-factor hazard scenarios.

(2) **Scenario-Anchored Requirement Synthesis** (Section 3.2), which grounds generated hazards in regulatory standards and robot capabilities to produce auditable FSRs.

This pipeline ensures that safety requirements are derived not from generic linguistic correlations, but from a deductive chain of physical attribution, combinatorial interaction, and regulatory alignment. The overall architecture of the proposed ARGOS framework is illustrated in Figure 1, highlighting the data flow from semantic parsing through attribute injection to the final synthesis of safety requirements.

3.1 Stage I: Attribute-Guided Hazard Discovery

Given a seed scenario S_{seed} expressed as a natural language instruction (e.g., "Deliver hot soup while a child is playing"), this stage extracts risk-relevant entities, grounds them in physical attributes, and performs combinatorial reasoning to discover multi-factor hazard scenarios. The output is a hazard description H_{desc} that captures multi-factor interaction risks absent from single-factor analysis.

Semantic Parsing and Attribute Injection. We first employ a syntactic parser (spaCy) to extract key semantic units $U = \{u_1, u_2, \ldots, u_n\}$ from S_{seed} as fundamental risk factors (e.g., `Child`, `Delivery`).

To eliminate ambiguity, we utilize a BGE-based embedding model to map each unit u_i to its corresponding risk attributes in a pre-defined Rule Base. The retrieval process is governed by a similarity threshold τ_{attr}:

$$A_{injected} = \bigcup_{u_i \in U} \{\text{Attr}(r) \mid r \in \mathcal{R}, \text{sim}(E(u_i), E(r)) > \tau_{attr}\}$$

The target Rule Base $\mathcal{R}$ is initialized through expert curation to ensure physical plausibility; representative examples of these rules are provided in Appendix B. This decouples atomic physical constraints from specific scenarios, allowing the pipeline to synthesize a wide range of complex interaction risks from a compact set of reusable attribute entries.

This step injects domain-specific priors into the context. For instance, the unit `Child` is explicitly mapped to dynamic attributes like `High Lateral Acceleration` and `Unpredictability`, while `Delivery` implies constraints such as `Delivery Stability` and `Gripper Compliance`.

Robust Retrieval via Hybrid Matching. Embedding retrieval can produce either an incorrect semantic match or no match. We therefore use a configurable Hybrid Hard-Match Fallback: when

enabled, exact lexical matches override embedding disagreements by assigning similarity 1.0. This combines semantic retrieval with deterministic handling of known rule-base entries and reduces error propagation from profile selection.

Combinatorial Hazard Inference. Given the seed S_{seed} and the injected attribute set $A_{injected}$, we prompt the LLM to identify how distinct risk factors interact. Formally, for a set of k risk factors $\{u_1, \ldots, u_k\}$, the hazard description H_{desc} is derived as:

$$H_{desc} = \text{LLM}_{infer}\left(S_{seed}, \bigotimes_{i=1}^{k} A(u_i)\right)$$

where $\bigotimes$ denotes a structured reasoning template that composes individual physical attributes into a joint context and explicitly instructs the LLM to derive interaction risks between them, rather than treating each attribute in isolation (see Appendix E for the exact template).

For the risk pair `Rose + Balloon`, the model deduces a material-interaction hazard: the rose's rigid thorns (`Sharp Protrusion`) interact with the balloon's thin latex membrane (`Low Puncture Resistance`). The resulting hazard is not kinematic—limiting velocity would not prevent puncture, as thorn-membrane contact at any speed is sufficient to rupture the balloon. Instead, the model identifies a force-level constraint, demonstrating that combinatorial reasoning discovers risks that single-factor analysis overlooks.

3.2 Stage II: Scenario-Anchored Requirement Synthesis

Given the hazard description H_{desc} produced in Stage I, together with the robot's hardware capabilities $\mathcal{K}_{HW}$ and a regulatory knowledge base $\mathcal{KB}_{reg}$, Stage II converts these descriptive hazards into prescriptive FSRs—system-level specifications that define *what* constraints must hold at the design stage (see Table 1 for the safety lifecycle context). Each synthesized FSR S_{fsr} is grounded in both regulatory clauses and the physical limits of the target platform, and can be further decomposed into verifiable technical requirements downstream.

Regulatory Alignment & Constraint Injection. To bridge the gap between abstract risk descriptions and engineering specifications, we retrieve relevant safety standards (e.g., ISO 13482 for personal care robots) based on the semantic embedding of the hazard:

$$\mathcal{R}_{rel} = \{r \in \mathcal{KB}_{reg} \mid \text{sim}(E(H_{desc}), E(r)) > \tau_{reg}\}$$

Simultaneously, we inject the robot's physical specification $\mathcal{K}_{HW}$ (e.g., max deceleration a_{max}, sensor range R_s) into the inference window. This explicitly defines the Feasible Actuation Space for the model.

Physics-Aware Requirement Generation. The synthesis function LLM_{syn} operates on the augmented context to resolve conflicts between safety goals and physical dynamics:

$$S_{fsr} = \text{LLM}_{\text{syn}}\left(H_{desc}, \mathcal{R}_{rel} \mid \mathcal{K}_{HW}\right)$$

The model evaluates whether standard mitigations (e.g., Emergency Stop) violate the physical constraints imposed by $\mathcal{K}_{HW}$. For instance, in the hot soup delivery scenario, the model identifies that emergency braking to avoid a child creates inertial forces that destabilize the liquid payload (`Fluid Instability under Acceleration`). Rather than a generic stop command, it derives a safe deceleration threshold a_{safe} from the hardware's braking capability and payload stability constraints. Each FSR is formatted as a structured, auditable requirement that states a triggering condition, required robot response, and termination condition, facilitating expert review and downstream technical specification.

Prompt Design. Both stages use structured prompts that enforce step-by-step reasoning. For Stage I, the prompt instructs the model to identify the physical mechanism linking each attribute pair and trace the causal chain from initial interaction to final hazard. For Stage II, the prompt requires the model to map the hazard to relevant regulatory clauses and check whether standard mitigations conflict with $\mathcal{K}_{HW}$. Full prompt templates are provided in Appendix E.

Computational Overhead. ARGOS is designed for offline safety analysis. The implementation caps interaction order at three and uses a fixed candidate-generation budget for the retrieved profiles. Across the 48 evaluated seeds, Stage I required an average of 7.62 hazard-generation calls per seed and at most 14 calls; BGE retrieval itself incurs millisecond-scale latency. The observed call budget is detailed in Appendix H.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Scenario Dataset: A total of 48 "Seed Scenarios" involving complex human-robot-environment interactions were curated. These seeds span three critical dimensions: (i) Vulnerable Groups (e.g., `elderly, child, wheelchair users`); (ii) Hazardous Environments (e.g., `low-light, steam-filled rooms`); (iii) Risky Objects (e.g., `sharp tools, hot liquids`). Through the proposed inference pipeline, these 48 seeds were expanded into 365 context-specific hazard scenarios.

Knowledge Base: The household-service rule base contains 82 reusable typed physical profiles constructed by a safety expert using a fixed schema over entities, user/environment states, robot states, task types, and physical mechanisms. Initial profile encoding required approximately six expert-hours; full construction details are provided in Appendix H.

Model Configurations: `DeepSeek-V3.2` served as the primary backbone for the complete safety reasoning pipeline. `GPT-4o` was used as a second backbone for Stage I and in a controlled 10-seed Stage-II robustness study. `BGE-Large-EN-v1.5` was used for retrieval, and `Gemini-3-Pro` functioned as the independent "LLM-as-a-Judge" for evaluation, with evaluation prompts provided in Appendix E.

Comparison Methods. For the primary DeepSeek-V3.2 evaluation over all 48 seeds, we compare ARGOS with four baselines under matched seed inputs, robot capability profiles, target counts, and evaluation rubrics:

- **Vanilla LLM:** generates hazards directly from the seed description and robot capability profile, without explicit intermediate reasoning or external attribute grounding.
- **Physics-Aware CoT (CoT):** uses a structured three-step prompt for entity decomposition, micro-action simulation, and consequence prediction, but without external attribute grounding.
- **Full-rulebook long-context:** places the complete 82-profile rule base in a single prompt without truncation, testing whether broad access to the same physical knowledge is sufficient without selective retrieval and controlled composition.
- **STPA/FMEA-structured:** organizes analysis around control actions, unsafe control actions, and failure modes across perception, planning, actuation, manipulation, environment, and human interaction.

ARGOS uses selective attribute retrieval followed by bounded interaction composition. Its discovery process is evaluated across single-factor, dual-factor, and triple-factor coupled interactions. We additionally evaluate Vanilla, CoT, and ARGOS with GPT-4o to test cross-backbone robustness. Full prompt and baseline settings are provided in Appendix E and Appendix H.

Evaluation Metrics We define six evaluation metrics across two stages: Physical Reliability (PR), Long-tail Risk (LR), and FSR Derivation for hazard scenario quality; Capability Compliance (CC), Physical Risk Coverage (PRC), and Logic Robustness for FSR quality. Formal definitions are provided in Table 6 (Appendix F).

Human Evaluation Protocol. We randomly sampled $N = 108$ derived scenarios across Ours, CoT, and Vanilla for blind human evaluation. Three reviewers, each with approximately three years of robot-safety research experience, completed a rubric-alignment phase and then independently scored anonymized, shuffled outputs on the same 1–10 PR/LR/FSR rubric used by the LLM judge. We report the three-rater mean; inter-rater agreement statistics are provided in Appendix H.

Table 2: Primary Stage-I comparison over all 48 seeds using DeepSeek-V3.2. PR, LR, and FSR are reported as Mean ± SD. Overall is the arithmetic mean of the three metric means. Bold indicates best performance.

Method	PR ↑	LR ↑	FSR Derivation ↑	Overall ↑
Vanilla	7.98 ±2.27	6.73 ±1.97	7.45 ±2.08	7.39
Physics-Aware CoT	8.53 ±1.46	6.57 ±2.08	7.07 ±2.03	7.39
STPA/FMEA-structured	7.85 ±1.34	7.15 ±1.68	7.70 ±1.77	7.57
Full-rulebook long-context	7.55 ±2.01	8.15 ±1.58	7.65 ±1.67	7.78
ARGOS (Ours)	**8.97** ±1.45	**8.21** ±1.44	**8.55** ±1.47	**8.57**

Table 3: Stage-I cross-backbone results on GPT-4o. Metrics are Mean ± SD.

Method	PR ↑	LR ↑	FSR Derivation ↑
Vanilla	7.99 ±1.78	6.51 ±1.75	7.22 ±1.72
Physics-Aware CoT	8.08 ±1.38	5.35 ±1.94	6.03 ±1.86
ARGOS (Ours)	**9.06** ±1.36	**7.90** ±1.81	**8.27** ±1.74

4.2 Hazard Inference Quality and Diversity

Table 2 reports the primary DeepSeek-V3.2 comparison over all 48 seeds. ARGOS achieves the highest Overall score (8.57). The full-rulebook long-context baseline is competitive on Long-tail Risk (8.15 vs. 8.21), but has lower Physical Reliability (7.55 vs. 8.97) and FSR Derivation (7.65 vs. 8.55). The STPA/FMEA-structured baseline is more balanced across the three metrics but reaches a lower Overall score (7.57). CoT improves Physical Reliability over Vanilla (8.53 vs. 7.98) while slightly reducing Long-tail Risk and FSR Derivation.

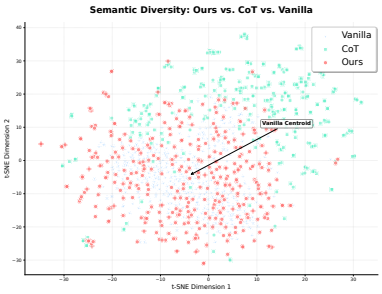
Statistical significance. We conduct Bonferroni-corrected pairwise Wilcoxon signed-rank tests on per-seed differences. ARGOS significantly outperforms Vanilla on all three metrics (PR: d=0.62, p<0.001; LR: d=1.00, p<0.001; FSR: d=0.71, p<0.001). Against CoT, gains are significant on LR (d=0.92, p<0.001) and FSR (d=0.84, p<0.001), while the PR difference is marginal (p=0.059), consistent with CoT's ability to support basic physical inference through structured reasoning.

Table 3 shows the cross-backbone Stage-I results. The same pattern holds on GPT-4o: ARGOS improves all three metrics, while CoT does not improve Long-tail Risk or FSR Derivation over Vanilla.

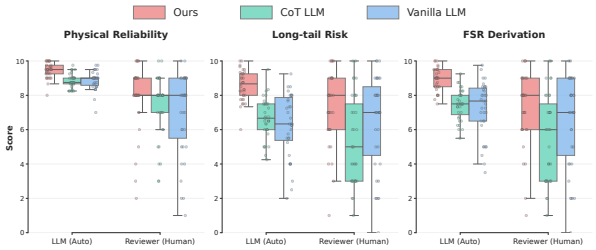
A separate blinded audit on 12 randomly sampled seeds provides paired checks for the stronger baselines. ARGOS reaches an Overall score of 8.45, compared with 8.30 for long-context and 7.87 for STPA/FMEA-structured, and is holistically preferred on 9 of the 12 seeds. Long-context is slightly higher on LR in this subset, while ARGOS is higher on PR and FSR Derivation. Full paired statistics are reported in Appendix H.

Reasoning vs. Attribute Grounding. The comparison with CoT further illustrates the role of explicit physical attributes. In the *fetch disinfectant while mopping* scenario, CoT variants repeatedly converge on a similar pattern in which a human slips on the wet floor. In contrast, attribute-grounded generation produces structurally different mechanisms, including sensor blind-zone collisions and fluid hazards caused by container failure. These cases suggest that structured reasoning can improve the coherence of a hazard description, while explicit physical attributes provide additional premises from which distinct interaction mechanisms can be derived.

Output Quality Analysis. We characterize scenario quality from two perspectives (Table 7 in Appendix G). In embedding space, ARGOS achieves a small Centroid Shift (0.055 vs. CoT's 0.218), near-zero Directional Similarity (0.008), and a CSE score of 13.14, or 3.76× CoT's 3.49, indicating that generated scenarios remain anchored to the seed intent while exploring distinct hazard directions. The score distribution is also concentrated in the high-quality range (8–10) with a limited low-scoring tail. Together, these results complement the rubric-based evaluation by showing that the improvement



(a) **t-SNE visualization of scenario embeddings**



(b) **Human–LLM score distributions.** LLM scores are higher in absolute terms, while human raters penalize CoT more strongly for semantic redundancy.

Figure 2: Qualitative Analysis: Semantic Diversity and Evaluation Alignment.

Table 4: Automated (LLM) and human evaluation on the same 108-scenario subset. Metrics are Mean ± SD.

Method	PR		LR		FSR	
	LLM	Human	LLM	Human	LLM	Human
Vanilla	8.88 ±0.53	6.97 ±2.40	6.24 ±1.82	6.43 ±2.70	7.21 ±1.57	6.49 ±2.67
CoT	8.84 ±0.38	7.34 ±1.79	6.66 ±1.16	5.29 ±2.51	7.45 ±0.87	5.46 ±2.53
Ours	**9.46** ±0.44	**7.94** ±1.87	**8.56** ±0.87	**7.26** ±2.12	**8.91** ±0.65	**7.34** ±2.12

is accompanied by semantic diversity rather than repeated variations of a small number of hazard patterns.

Retrieval Threshold Sensitivity. We evaluate $\tau_{attr} \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$. At τ_{attr}=0.7, a case-by-case audit of the evaluated set found no retrieval errors. Lowering the threshold to 0.6 introduces approximately 25% hallucinated retrievals, while $\tau_{attr} \geq 0.8$ misses valid semantic matches. We therefore use τ_{attr}=0.7 for all reported experiments; audit definitions are provided in Appendix H.

Human Evaluation We evaluate 108 derived sub-scenarios with three robot-safety researchers using the same 1–10 PR/LR/FSR rubric as the LLM judge. Reviewers scored anonymized, shuffled outputs after rubric alignment and were separate from rule-base construction and scenario generation. The three-rater ICC(2,k) ranges from 0.630 to 0.851 and Krippendorff's α from 0.570 to 0.765; full agreement statistics are reported in Appendix H.

Table 4 compares human and LLM scores on the same 108-scenario subset. The LLM scores in this table are computed on that subset and therefore differ from the full-dataset results in Table 2. LLM scores are approximately 1.5 points higher in absolute terms, indicating a systematic calibration offset, while the overall method-level trend remains similar across the two evaluation regimes. The largest method-specific difference appears for CoT on Long-tail Risk: the LLM judge scores CoT slightly above Vanilla (6.66 vs. 6.24), whereas human reviewers score it below Vanilla (5.29 vs. 6.43). Review logs show that many CoT outputs are logically coherent but repeat variants of the same hazard mechanism; in the Hot Water Delivery case, CoT produces six slipping-related variants across different reasoning steps. We refer to this pattern as the *Verbosity Trap*. In contrast, ARGOS attains the highest human LR score (7.26 ± 2.12) and captures coupled mechanisms such as the "Safety Pinch," where a safety stop can trigger a secondary fluid hazard. ARGOS also obtains the highest human mean on PR and FSR Derivation in this subset, while the agreement statistics indicate moderate-to-good consistency among the three reviewers.

Table 5: Ablation Study on FSRs Quality. ARGOS (Ours) denotes the full pipeline. w/o ISO removes regulatory retrieval. ISO-Only removes attribute injection. Vanilla is the baseline. Metrics are reported as Mean ± SD.

Method	CC ↑	PRC ↑	LRC ↑	Overall ↑
Vanilla	7.91 ±1.95	6.96 ±1.92	7.28 ±2.02	7.38 ±1.66
ISO-Only	7.96 ±2.19	7.00 ±2.03	6.90 ±1.95	7.28 ±1.68
w/o ISO	8.56 ±1.97	7.71 ±1.70	7.29 ±2.27	7.85 ±1.70
ARGOS (Ours)	**8.56** ±1.47	**7.81** ±1.71	**7.54** ±1.70	**7.97** ±1.25

4.3 Evaluation of FSR Quality

Table 5 evaluates the contribution of physical attributes and regulatory retrieval to FSR synthesis. The two variants with physical grounding, w/o ISO and ARGOS, obtain the highest Overall scores (7.85 and 7.97), while ISO-Only and Vanilla score 7.28 and 7.38, respectively.

Effect of Physical Grounding. For Capability Compliance, w/o ISO and ARGOS both score 8.56, compared with 7.96 for ISO-Only. This suggests that explicit physical attributes provide most of the information needed to connect hazards to robot capability constraints. Adding ISO retrieval produces smaller gains in PRC and Logic Robustness. ISO-Only also shows the largest CC variance (SD=2.19), compared with 1.97 for w/o ISO and 1.47 for ARGOS; adding ISO retrieval to the physically grounded variant therefore leaves the mean CC unchanged while reducing its variance.

Example: Rose & Balloon. The baseline treats the rose mainly as an obstacle and proposes a kinematic constraint such as "Limit velocity < 0.1 m/s." This does not address the underlying failure mechanism: a rigid thorn can puncture the balloon's low-resistance membrane even at low contact speed. ARGOS instead retrieves these material attributes and derives a contact-force constraint. The example illustrates how attribute-level interaction can change the form of the synthesized requirement rather than only its wording.

Regulatory Context Without Physical Grounding. On DeepSeek-V3.2, ISO-Only scores 6.90 in Logic Robustness, below Vanilla at 7.28, whereas ARGOS reaches 7.54. This pattern is consistent with a regulatory mismatch: abstract clauses are harder to apply when the physical variables that define their trigger conditions are absent from the reasoning context.

Cross-Backbone Stage-II Robustness. In a controlled 10-seed GPT-4o study, the two physically grounded variants remain the top two methods: ARGOS scores 7.59 Overall and w/o ISO 7.38, compared with 7.10 for ISO-Only and 6.85 for Vanilla. Full results are reported in Appendix H.

5 Conclusion and Discussion

We present ARGOS, a two-stage pipeline that uses explicit physical attributes as reasoning premises for hazard discovery and carries the resulting hazard mechanisms into FSR synthesis. On the primary DeepSeek-V3.2 evaluation, ARGOS achieves the highest Overall Stage-I score among reasoning, structured-safety, and full-rulebook long-context baselines. The long-context baseline is close on Long-tail Risk, while ARGOS is stronger on Physical Reliability and FSR Derivation. The same advantage over Vanilla and CoT is observed with GPT-4o, and human evaluation shows that CoT produces more semantically redundant hazards than automated scoring alone suggests.

Stage-II ablations show that physical grounding contributes most of the gain in capability-aware requirement synthesis, while regulatory retrieval provides an additional constraint when paired with physical context. A central result is that physical grounding changes the quality of safety reasoning, rather than simply increasing the amount of context or the number of reasoning steps. Explicit physical profiles provide inspectable premises for hazard inference, while bounded interaction reasoning connects these premises to coupled hazards that can subsequently be translated into functional safety requirements. This separation between physical hazard discovery and requirement synthesis is the main design principle underlying ARGOS.

AI USE STATEMENT

We partially used generative AI tools for code development and for improving the clarity and readability of the manuscript. All AI-assisted code and text modifications were carefully reviewed, verified, and tested by the authors.

ETHICS STATEMENT

This work includes an expert evaluation in which three robot-safety researchers scored anonymized model-generated safety scenarios. The evaluation did not collect sensitive personal data and did not involve interaction with physical robotic systems or exposure to physical risk. The remaining experiments are model- and dataset-based safety analyses.

REPRODUCIBILITY STATEMENT

We are committed to making this work reproducible. In the main paper and appendix, we provide as much detail as possible about the method, model architecture, and experimental details needed to reproduce this work, and we provide an anonymized code repository link to facilitate reproduction.

REFERENCES

Emily M Bender and Alexander Koller. Climbing towards NLU: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 5185–5198, 2020.

Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, et al. Experience grounds language. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 8718–8735, 2020.

Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, et al. Do as I can, not as I say: Grounding language in robotic affordances. In *Conference on Robot Learning (CoRL)*, pp. 287–318, 2023.

Simon Diemert and Jens H Weber. Can large language models assist in hazard analysis? In *International Conference on Computer Safety, Reliability, and Security (SAFECOMP)*, pp. 410–422. Springer, 2023.

Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. PaLM-E: An Embodied Multimodal Language Model. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pp. 8469–8488, 2023.

Angela Fan, Beliz Gokkaya, Mark Harman, Mitya Lyubarskiy, Shubho Sengupta, Shin Yoo, and Jie M Zhang. Large language models for software engineering: Survey and open problems. In *2023 IEEE/ACM International Conference on Software Engineering: Future of Software Engineering (ICSE-FoSE)*, pp. 31–53. IEEE, 2023.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Arshia Hemmat, Mohammadreza Sharbaf, Shekoufeh Kolahdouz-Rahimi, Kevin Lano, and Sobhan Y Tehrani. Research directions for using LLM in software requirement engineering: A systematic review. *Frontiers in Computer Science*, 7:1519437, 2025.

Xinyi Hou, Yanjie Zhao, Yue Liu, Zhou Yang, Kailong Wang, Li Li, Xiapu Luo, David Lo, John Grundy, and Haoyu Wang. Large language models for software engineering: A systematic literature review. *ACM Transactions on Software Engineering and Methodology*, 33(8):1–79, 2024.

Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, et al. Inner monologue: Embodied reasoning through planning with language models. In *Conference on Robot Learning*, pp. 1769–1782, 2023.

International Organization for Standardization. Robots and robotic devices — safety requirements for personal care robots. Standard ISO 13482:2014, International Organization for Standardization, Geneva, Switzerland, 2014.

Padma Iyenghar, Zaher Mansour, and Juergen Wuebbelmann. Evaluation of automated machinery functional safety risk assessment using LLMs. *IEEE Access*, 13:203648–203669, 2025.

Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Josef Dai, and Yaodong Yang. Safety gymnasium: A unified safe reinforcement learning benchmark. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 18964–18993, 2023.

Adam Tauman Kalai and Santosh S Vempala. Calibrated language models must hallucinate. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 160–171, 2024.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. Large language models struggle to learn long-tail knowledge. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pp. 15696–15707, 2023.

Przemyslaw A Lasota, Terrence Fong, Julie A Shah, et al. A survey of methods for safe human-robot interaction. *Foundations and Trends® in Robotics*, 5(4):261–349, 2017.

Nancy G Leveson. *Engineering a safer world: Systems thinking applied to safety*. The MIT Press, 2016.

Boyang Ma, Hechuan Guo, Peizhuo Lv, Minghui Xu, Xuelong Dai, YeChao Zhang, Yijun Yang, and Yue Zhang. What breaks embodied ai security: Llm vulnerabilities, cps flaws, or something else? *arXiv preprint arXiv:2602.17345*, 2026.

Nuno Marques, Rodrigo Rocha Silva, and Jorge Bernardino. Using ChatGPT in software requirements engineering: A comprehensive review. *Future Internet*, 16(6):180, 2024.

Johannes J Norheim, Eric Rebentisch, Dekai Xiao, Lorenz Draeger, Alain Kerbrat, and Olivier L de Weck. Challenges in applying large language models to requirements engineering tasks. *Design Science*, 10:e16, 2024.

Ali Nouri, Beatriz Cabrero-Daniel, Fredrik Torner, Hakan Sivencrona, and Christian Berger. Welcome your new AI teammate: On safety analysis by leashing large language models. In *Proceedings of the 3rd IEEE/ACM International Conference on AI Engineering – Software Engineering for AI (CAIN)*, pp. 172–177, 2024.

Laura Ruis, Maximilian Mozes, Juhan Bae, Siddhartha Rao Kamalakara, Dwaraknath Gnaneshwar, Acyr Locatelli, Robert Kirk, Tim Rocktäschel, Edward Grefenstette, and Max Bartolo. Procedural knowledge in pretraining drives reasoning in large language models. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2024.

Sanjit A Seshia, Dorsa Sadigh, and S Shankar Sastry. Toward verified artificial intelligence. *Communications of the ACM*, 65(7):46–55, 2022.

Lu Shi, Bin Qi, Jiarui Luo, Yang Zhang, Zhanzhao Liang, Zhaowei Gao, Wenke Deng, and Lin Sun. Aegis: An advanced LLM-based Multi-Agent for intelligent Functional Safety engineering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP): Industry Track*, pp. 1571–1583, 2024.

Karthik Valmeekam, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. Large language models still can't plan (A benchmark for LLMs on planning and reasoning about change). In *NeurIPS 2022 Workshop on Foundation Models for Decision Making*, 2022.

Yi Ru Wang, Jiafei Duan, Dieter Fox, and Siddhartha Srinivasa. Newton: Are large language models capable of physical reasoning? In *The 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

A PROMPT CONSTRUCTION & TEMPLATES

In this section, we detail the exact prompt templates used to instantiate our *Attribute-Grounded Generation* framework. Unlike standard retrieval approaches that treat retrieved text as passive context, our method formulates the generation task as a **Dual-Factor Constraint Satisfaction Problem**.

The prompt consists of three modular blocks:

1. **System Definition:** Establishes the robot's physical embodiment (see Appendix C).
2. **Risk Injection Block (Ours):** Dynamically embeds two orthogonal risk factors as coupled constraints.
3. **Task Specification:** The seed scenario and output formatting rules.

A.1 RISK INJECTION MECHANISM (OURS)

The core novelty of our method lies in the *Risk Injection Block*. We retrieve two complementary risk factors (R_A, R_B) and inject them into the following template. This forces the LLM to reason about the *interaction* between environmental hazards and robot limitations.

Template: Dual-Factor Injection Block

```
Template: Dual-Factor Risk Scenario Prompt

[SYSTEM ROLE]
You are a safety engineer identifying how SPECIFIC, REAL-WORLD risk conditions can lead to
     harm during home robot tasks.

Generate a scenario where a hazard occurs due to the interaction of the specified risk
     factors.

{Robot_Specifications}

[RISK INTERACTION]
The hazard must be caused by the INTERACTION of TWO specific risk factors.
Condition: The hazard occurs because BOTH factors are present simultaneously.

Risk Factor A: {{Risk_A.Value}} ({{Risk_A.Dimension}})
Definition: "{{Risk_A.Description}}"

Risk Factor B: {{Risk_B.Value}} ({{Risk_B.Dimension}})
Definition: "{{Risk_B.Description}}"

Seed scenario: "{Seed_Scenario}"

[CONSTRAINTS FOR SCENARIO GENERATION]

1.The scenario must remain strictly within the semantic boundary defined by the seed
     scenario.
2.Only entities explicitly mentioned in the seed or logically indispensable for the task
     may be used.
3.No unmentioned object, state, event, or environmental condition may be introduced under
     any circumstance.
4.The root cause must be the coupling effect of Factor A and Factor B.
5.If one factor is removed, the accident must become impossible.
6.Robots will not have emotions.
7.[STRICT CLOSED WORLD] ROBOT CAPABILITIES: Strictly control the robot's behavior within
     the defined capabilities.
8.STRICT ADHERENCE TO SPECIFICATIONS: Strictly enforce the union of all sensor coverage
     ranges defined in the robot capabilities.
9.[CRITICAL] MAINTAIN PHYSICAL REALISM.
10.[LOGICAL CONSISTENCY] CAUSAL VALIDITY (DETECTION != PREVENTION).
[OUTPUT FORMAT]
EXACTLY TWO LINES:
Line 1: 2-3 sentences describing the robot's task execution and how the two factors
     combined to cause the hazard.
Line 2: "Hazard mechanism: [one sentence explaining the coupling effect]."
```

A.2 FULL SYSTEM PROMPTS

Baseline Prompt (Vanilla LLM) The baseline prompt provides the robot capabilities but lacks the specific risk injection block.

System Prompt: Baseline

```
Template: Baseline Hazardous Scenario Generation (CoT)
```

```
[SYSTEM ROLE]
You are a safety engineer identifying how real-world conditions can lead to harm during
     home robot tasks.

Generate a scenario where a hazard occurs due to potential risks inherent in the task or
     environment.

[ROBOT CAPABILITIES]
{Robot_Specifications}

[SEED SCENARIO]
"{Seed_Scenario}"

[CONSTRAINTS]

1.The scenario must remain strictly within the semantic boundary defined by the seed
     scenario.
2.Only entities explicitly mentioned in the seed or logically indispensable for the task
     may be used.
3.Robots will not have emotions similar to humans, such as fear or anxiety, startle.
4.No unmentioned object, state, event, or environmental condition may be introduced under
     any circumstance.
5.STRICT ADHERENCE TO SPECIFICATIONS: Strictly enforce the union of all sensor coverage
     ranges defined in the robot capabilities.
6.[STRICT CLOSED WORLD] ROBOT CAPABILITIES: Strictly control the robot's behavior within
     the defined capabilities.
7.[CRITICAL] MAINTAIN PHYSICAL REALISM.
8.[LOGICAL CONSISTENCY] CAUSAL VALIDITY (DETECTION != PREVENTION).

[OUTPUT FORMAT]
Line 1: 2-3 sentences describing the robot's task execution and how the two factors
     combined to cause the hazard.
Line 2: "Hazard mechanism: [one sentence explaining the coupling effect].
```

A.2.1 CHAIN-OF-THOUGHT (COT) PROMPT

The CoT baseline appends reasoning instructions to the standard prompt to encourage step-by-step simulation.

Add-on: CoT Reasoning Instructions

```
Template: Baseline Hazardous Scenario Generation (CoT)

[SYSTEM ROLE]
You are a safety engineer identifying how real-world conditions can lead to harm during
     home robot tasks.

Generate a scenario where a hazard occurs due to potential risks inherent in the task or
     environment.

[ROBOT CAPABILITIES]
{Robot_Specifications}

[SEED SCENARIO]
"{Seed_Scenario}"

[CONSTRAINTS]

1.The scenario must remain strictly within the semantic boundary defined by the seed
     scenario.
2.Only entities explicitly mentioned in the seed or logically indispensable for the task
     may be used.
3.Robots will not have emotions similar to humans, such as fear or anxiety, startle.
4.No unmentioned object, state, event, or environmental condition may be introduced under
     any circumstance.
5.STRICT ADHERENCE TO SPECIFICATIONS: Strictly enforce the union of all sensor coverage
     ranges defined in the robot capabilities.
6.[STRICT CLOSED WORLD] ROBOT CAPABILITIES: Strictly control the robot's behavior within
     the defined capabilities.
7.[CRITICAL] MAINTAIN PHYSICAL REALISM.
8.[LOGICAL CONSISTENCY] CAUSAL VALIDITY (DETECTION != PREVENTION).
[REASONING INSTRUCTIONS]
Before generating the final result, you must analyze the seed scenario step-by-step inside
     an [ANALYSIS] block:

Entity & Vulnerability Analysis (Decomposition):

Extract all key Entities (Who/What), Actions (Task), and Environmental Context.
Vulnerability Inference: Analyze inherent physical limitations or vulnerabilities.
Micro-Action Simulation (Step-by-Step Trace):

Break down the robot's task into a sequence of logical micro-actions.
Simulate the physical interaction between the robot and identified entities.
Consequence Analysis:

Identifies accident chains.
[OUTPUT FORMAT]
[ANALYSIS]
```

```
Decomposition: [Entities, Context, and vulnerabilities...]
Simulation: [Step-by-step trace...]
Prediction: [Logical deduction...]
[OUTPUT FORMAT]
Line 1: 2-3 sentences describing the robot's task execution and how the two factors
    combined to cause the hazard.
Line 2: "Hazard mechanism: [one sentence explaining the coupling effect].
```

B RULE BASE

This appendix provides the formal technical definitions for the dimensions used in the *Risk Injection Block* (see Appendix A.1). These definitions serve as the ground-truth constraints for the LLM to reason about physical vulnerabilities and environmental hazards.

B.1 USER STATE DIMENSIONS (U)

The following definitions specify the physiological and behavioral constraints of human and non-human agents in the environment.

Table B.1: User State Definitions

`U01` Elderly: Human individual aged 65 years or older, exhibiting reduced visual acuity in low illumination, slower neuromuscular response latency to visual motion stimuli, and decreased postural stability during upper-body perturbations.

`U13` Toddler: Human individual (aged 1–3 years) with standing height between 70 cm and 95 cm. Characterized by an unstable gait with high fall frequency, limited hazard recognition, and unpredictable reach-and-grasp motions. Fingers are small enough to enter gaps as narrow as 5 mm. High propensity for "oral exploration."

`U12` Pet Animal: Non-human biological agent (e.g., *Canis lupus familiaris*, *Felis catus*) with highly variable morphology, typically ranging from 10–80 cm in height. Characterized by high-DoF omni-directional locomotion, capability for instantaneous high-jerk acceleration (> 3 m/s^2), and stochastic trajectory evolution without explicit signaling.

`U02` Child: Human individual with standing height between 80 cm and 120 cm, capable of lateral acceleration exceeding 2 m/s^2 within 0.3 seconds, and hand breadth less than 6 cm.

`U03` Mobility Impaired: Human individual using a walker or cane for ambulation, requiring a minimum clear passage width of 90 cm, unable to step over obstacles taller than 2 cm.

`U08` Distracted: User's gaze is fixed on a specific focal point with a visual field of view restricted to < 20 degrees. Auditory perception is attenuated by external devices, raising the reaction threshold to auditory stimuli by 15–20 dB.

`U07` Drowsy: Human individual exhibiting prolonged eyelid closure (> 1 second), delayed reaction time (> 1.5 s) to tactile stimuli, and reduced grip force (< 10 N) during object holding.

`U05` Hearing Loss: Human individual with audiometric thresholds exceeding 40 dB HL in the 500–4000 Hz frequency range.

`U10` Walking with Cane: Human individual using a single-point cane, resulting in asymmetric gait cycle, average walking speed below 0.8 m/s, and constant unilateral ground contact.

B.2 ENVIRONMENT STATE DIMENSIONS (E)

These dimensions define the physical and sensory constraints of the workspace that interfere with robot perception or mobility.

Table B.2: Environment State Definitions

`E01` Low Light: Ambient illuminance measured below 50 lux at floor level.

`E03` Wet Floor: Floor surface covered with a continuous liquid layer, reducing static coefficient of friction to below 0.3.

`E04` Cluttered: Primary walkway contains three or more unsecured objects on the floor within a 2-meter segment, reducing free path width to less than 70 cm.

E05 Steam: Airborne water vapor density sufficient to reduce optical transmission by more than 50% at 850 nm wavelength over a 1-meter path.

E06 Small Space: Enclosed area with internal width between 60 cm and 120 cm, allowing only single-file passage between stationary obstacles.

E08 Half-open Drawer: Drawer extended 10–30 cm from its housing, presenting a flat obstacle at a height of 30–60 cm above floor level.

E09 Stairs: Vertical transition composed of steps with rise between 15 cm and 25 cm and no ramp or mechanical assistive device present.

E10 Threshold: Raised transition between adjacent floor surfaces with vertical height difference between 1.5 cm and 5 cm.

B.3 Supervision and Task Dimensions (S/T)

Table B.3: Supervision and Task Definitions

S01 Unsupervised: No human agent is present in the environment capable of real-time observation or physical intervention.

T01 Delivery: The process of transferring an object to a designated recipient. Requires the robot's end effector to precisely reach a physically accessible zone and synchronize release timing with the recipient's grasp.

T04 Emergency Response: Task triggered by detection of a fall, vocal distress signal, or abrupt physiological change, requiring immediate action within 30 seconds.

T03 Monitoring: Task requiring continuous sensor lock on a target entity within a geofenced zone, updating position and posture data at a frequency greater than 5 Hz.

T05 Repeated Request: Task initiated by two or more identical user commands issued within a 10-second interval, indicating potential user urgency or system failure.

C Robot Functional Specifications

To ensure physical grounding, we defined a rigorous *Robot Capability Profile* that is injected into all prompts. This profile specifies testable thresholds (e.g., blind zones, latency) to prevent the LLM from hallucinating arbitrary capabilities.

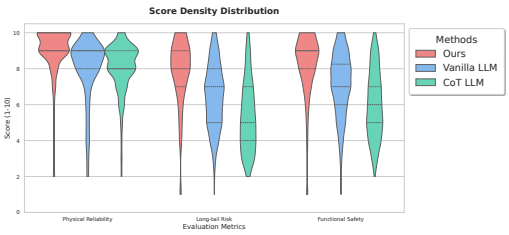
Full Robot Specification Text

```
[ROBOT FUNCTIONAL SPECIFICATIONS - STRICT PHYSICAL LIMITS]
(Use these limits to define verifiable thresholds. Do not assume capabilities not listed.)

. PERCEPTION SYSTEMS (SENSING)
  - Head-Mounted Vision System:
     Function: Long-range detection, Semantic Classification, Body Keypoint Tracking.
     Physical Limits: Range 0.05m-5m. Blind Zone < 0.2m.
     Vertical Coverage: High vantage point; capable of seeing tabletops and floor.
  - Wrist-Mounted Vision System:
     Function: Close-range, occlusion-free manipulation guidance.
     Physical Limits: High precision. Blind Zone < 0.05m.
  - Chassis LiDAR System:
     Function: 360 Volumetric Obstacle Detection.
     Physical Limits: Range 0.1m-20m. Vertical Coverage: 0.05m to 0.8m.
     Constraint: Physically unable to detect overhanging obstacles above 0.8m.
  - Proximity Safety System:
     Function: Close-range blind-spot coverage and retraction guidance.
     Physical Limits: Range 0.05m-1.5m. 360 Sector Coverage.
     Role: Fills the gap between the chassis and the LiDAR/Camera minimum range.
  - Auditory System:
     Function: Sound Source Localization (Azimuth), Voice Command Recognition.
     Capability: Identifies speaker direction and captures speech content.
  - Thermal Perception System:
     Function: Non-contact surface temperature monitoring.
     Capability: Identifies "Thermal Hazards" (e.g., hot plates, boiling liquids)
               to inform manipulation safety.
     Physical Limits: Effective Range 0.1m-1.2m. Accuracy +/- 2C.
  - Localization and Proprioception System:
     Function: Real-time Map Pose, Room Identification, Kinematic State.
     Performance: Fuses IMU and Odometry for precise self-positioning and
                  Wheel Slip Detection (via cross-verification of velocity vectors).
  - Environmental Analysis System:
     Function: Floor texture and safety analysis.
     Capability: Distinguishes "Traversable" floor from "Wet/Hazardous" surfaces.
  - Tactile and Force System:
     Function: Direct contact force (N), Torque (Nm), Slip detection.
```

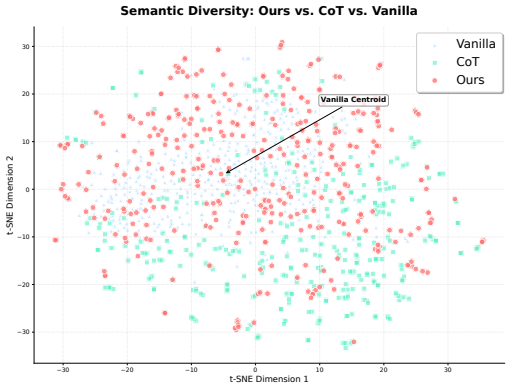
```
      Components: End-Effector Sensors: 6-axis Force/Torque and Fingertip Tactile
               arrays (High Precision). Chassis Safety Bumpers: Physical contact
               switches along the base perimeter (Binary Trigger).
      Performance: 1kHz reflex loop. Chassis Activation Threshold: > 5N.
  - System Memory and Logic (Compute Architecture):
     Object Persistence Engine: Maintains state of dynamic objects (people/tools)
                          in blind zones.
     Local Semantic Map: Maintains state of static structures (walls/furniture)
                   outside FOV.
     Safety Compute Engine (Dual-Loop Architecture):
       1. Cognitive Loop (High-Level):
          Function: Vision processing, Object Classification, Path Planning.
          Latency: 20Hz (~50ms).
          Constraint: Fast-moving objects (>2m/s) may traverse significant
                    distance between processing frames, causing "Ghosting"
                    or trajectory prediction errors.
       2. Reflex Loop (Low-Level):
          Function: Motor current monitoring, Tactile/Force feedback,
                  Slip detection.
          Latency: 1kHz (<1ms).
          Constraint: Purely reactive. Can stop the robot instantly upon
                    contact, but has zero predictive capability
                    (cannot act before contact).

. ACTUATION SYSTEMS (MOTION and MANIPULATION)
  - Mobility Chassis System:
     Function: Omni-directional (Holonomic) navigation.
     Capability: Simultaneous translation and rotation; supports zero-radius
              turning and lateral "strafing". Active Traction Control (ATC):
              Automatically modulates wheel torque to mitigate slip
              when detected.
     Role: Optimized for maneuvering in tight domestic quarters where
          standard turning is impossible.
  - 6-DOF Manipulation Arm:
     Function: Spatial positioning with Hybrid Position/Force Control.
     Capability: Impedance Control (Variable Stiffness). Can switch between
              "Rigid" (high precision) and "Compliant" (soft/follow) modes.
     Safety Feature: Gravity Compensation. Dynamically subtracts arm weight
                 to allow "weightless" manual guidance by humans.
  - Smart End-Effector (Gripper):
     Logic: Automatically modulates grip force based on tactile slip
          feedback (1kHz loop).
     Safety Constraint: Supports Dynamic Force Ceiling. The reflex can be
                    configured to either:
       1. Standard Mode: Override force limits to prevent drops (Default).
       2. Fragile Mode: Strictly adhere to a pre-defined Maximum Allowable
                    Force, even if slip is detected.
     Feedback: If the force ceiling is reached and slip persists, the system
            triggers an immediate "Safe Placement" or "Controlled Release"
            routine.

. INTERACTION SYSTEMS (HMI)
  - Auditory Feedback System:
     Function: Status communication and safety alerting.
     Capability: Text-to-Speech (TTS) for dynamic information and predefined
              Audio Alerts for warnings.
  - Visual Illumination System:
     Function: Scene illumination and status indication.
     Physical Limits: High-Power LED with 0-100% PWM brightness control.
     Role: Provides visibility for vision sensors in dark environments and
          signals robot state to users.

. CONTROL and SAFETY BEHAVIORS
  - Reactive Control Engine:
     Function: High-level task orchestration via Behavior Trees.
     Capability: Guarded Reach. Moves to target but triggers an immediate
              stop if external force exceeds the safety threshold.
     Performance: Real-time modulation of motor torque based on
               collision detection.
  - Retraction and Recovery Logic:
     Function: Safe exit from blind-spot collisions or stuck states.
     Input: Trajectory Buffer. Stores the kinematic history of the last
          10 seconds.
     Action: Computes and executes the inverse vector of recent movement
           to safely "back out" without needing new sensor data.
  - Context-Aware Navigation:
     Function: Velocity and parameter adjustment based on location.
     Capability: Identifies Semantic Room ID (e.g., "Kitchen") to automatically
              apply specific speed limits or safety margins.
     Logic: Supports Dynamic Masking to ignore specific geometric data
          (e.g., the robot's own arm) during obstacle detection.
```

D ADDITIONAL VISUALIZATIONS FOR GPT-4O BACKBONE

In the main text, we primarily analyzed the results based on the DeepSeek-V3.2 backbone. To demonstrate the model-agnostic nature of our framework, we provide the corresponding visualizations for the **GPT-4o** backbone in this section.



(a) Score Distribution (Violin Plot)



(b) Feature Space Visualization (t-SNE)

Figure 3: **Visualizations for the GPT-4o Backbone.** (a) The violin plot confirms that our method maintains a "top-heavy" high-quality distribution even on the stronger GPT-4o model, whereas baselines still exhibit long-tail risks. (b) The t-SNE projection shows that our method (red) covers a distinct and broader semantic space compared to the baselines, consistent with the findings on DeepSeek-V3.2.

E Prompt Construction & Templates

In this section, we detail the exact prompt templates used for the evaluation phase of our framework.

E.1 Functional Safety Requirement (FSR) Audit Prompt

Template: FSR Technical Audit

```
EVALUATION_PROMPT_TEMPLATE = """You are a Principal Functional Safety Auditor and Robotics
     Systems Architect. Your task is to perform a rigorous, standalone evaluation of a set
     of Functional Safety Requirements (FSRs) generated for a specific robotic scenario.

[IMPORTANT INSTRUCTION]
Ignore the name/label of the method (e.g., "Method A and FSR ID"). Focus exclusively on the
      technical quality, completeness, and hardware-grounding of the FSRs provided.

[INPUT DATA]
Seed Scenario: {seed_scenario}

Target FSRs to Evaluate:
{fsr_content}

[STAGE 1: AUDIT FOUNDATIONS - THE LAWS OF PHYSICS AND HARDWARE]
[ROBOT FUNCTIONAL SPECIFICATIONS - EXTERNAL REFERENCE]
**Refer to the external technical document: {{ROBOT_HARDWARE_SPECIFICATION_DOC}}**
(This document defines the absolute physical constraints, including Perception Systems,
     Actuation Systems, Interaction Systems, and Control/Safety Behaviors. Any FSR
     violating these limits is considered a "Hallucination" and must be severely penalized
     .)

[STAGE 2: SCORING RUBRICS (THE AUDIT STANDARD)]

Metric 1: Capability Compliance & Grounding (CC)
* 9-10 (Hardware-Optimized): The method perfectly leverages specific hardware capabilities
     (e.g., specific control modes, memory engines, sensor fusion) to solve safety problems.
      It explicitly accounts for sensor limits and blind spots defined in the spec.
* 7-8 (Advanced Adaptation): The method aligns well with hardware specs and accounts for
     most sensor limitations, though it may not fully exploit advanced features like
     specific memory engines or complex fusion algorithms.
* 5-6 (Generic): The method uses generic logic without leveraging the robot's specific
     advanced capabilities. It does not violate limits but does not optimize for them.
* 3-4 (Suboptimal Alignment): The method has loose integration with hardware specs. It
     fails to account for obvious sensor limits or blind spots, potentially leading to
     unstable performance in specific hardware environments.
* 1-2 (Hallucination/Violation): The method demands capabilities the robot does not have.
     Specifically, it requires detection ranges or sensor modalities that contradict the [
     ROBOT FUNCTIONAL SPECIFICATIONS]. This is an automatic failure.

Metric 2: Scenario Risk Coverage (PRC)
* 9-10 (Exhaustive Coverage): The FSRs identify and mitigate all primary hazards, secondary
      consequences (e.g., inertia, load stability), and **long-tail/edge-case risks**
     specific to this scenario (e.g., rare environmental interferences, complex human
     behaviors, or multi-system failures).
```

```
* 7-8 (Advanced Physics & Risk): The method goes beyond simple collision avoidance and
    addresses dynamic effects and most secondary risks, but coverage of complex long-tail
    interaction dynamics is incomplete.
* 5-6 (First-Order/Basic): The method covers direct risks only. It addresses primary
    collisions and slips but ignores the secondary effects of the robot's reaction or
    scenario-specific nuances.
* 3-4 (Basic Physics): The method has a narrow understanding of physical risks, identifying
     only the most obvious contact risks while neglecting friction, slippage, or basic
    kinematic constraints.
* 1-2 (Superficial): The method relies solely on semantic rules and ignores physical
    dynamics and scenario-specific risks entirely.

Metric 3: Logic Robustness & Continuity (LRC)
* 9-10 (Closed-Loop System): The method defines clear Entry AND Exit conditions for every
    safety state. It utilizes recovery logic (e.g., Trajectory Buffer) to exit failure
    modes instead of just freezing. It uses memory/persistence to handle perception gaps.
* 7-8 (Robust Closed-Loop): The method defines clear entry/exit conditions and includes
    basic recovery logic. However, it may be slightly lacking in handling complex
    perception gaps or long-term memory persistence.
* 5-6 (Open-Loop): The method defines when to stop but provides vague or missing conditions
     for when to resume operation.
* 3-4 (Fragmented Logic): The method has basic trigger logic, but state transitions are
    inconsistent and lack clear recovery paths, likely leading to frequent freezing or
    requiring manual intervention.
* 1-2 (Deadlock/Dangerous): The method creates logical deadlocks (robot freezes permanently
    ) or prescribes dangerous actions that violate basic safety principles.

[STAGE 3: THE AUDIT PROCESS (THINK STEP-BY-STEP)]

**STEP 1: INDEPENDENT HARDWARE REALITY CHECK**
* Cross-reference every FSR against the [ROBOT FUNCTIONAL SPECIFICATIONS].
* Identify any requirement that assumes a sensor range, coverage, or capability that is
    explicitly listed as a limitation or not listed at all.

**STEP 2: SCENARIO-SPECIFIC HAZARD ENUMERATION**
* Analyze the "Seed Scenario" deeply. List all potential risks:
   1. Primary Risks (e.g., direct collision).
   2. Secondary Risks (e.g., inertia after emergency stop, object dropping).
   3. **Long-tail Risks** (e.g., sensor blinding by sunlight, floor slip during heavy load,
         unexpected human interference).
* Check if the provided FSRs cover these specific points.

**STEP 3: LOGICAL STATE MACHINE & RECOVERY ANALYSIS**
* Analyze the "Trigger -> Action -> Recovery" loop.
* Does the method utilize the **Trajectory Buffer** or **Object Persistence Engine** to
    handle blind spots or recovery?

**STEP 4: FINAL SCORING**
* Assign scores based on the evidence. Be extremely critical of generic requirements that
    ignore the robot's advanced hardware.

[OUTPUT FORMAT]

**1. Detailed Audit Analysis (Chain of Thought)**
* **Hardware Alignment Analysis**: [Detailed reasoning on hardware alignment]
* **Scenario Risk & Long-tail Coverage Analysis**: [Detailed reasoning on how well the FSRs
     cover primary, secondary, and long-tail risks of the specific scenario]
* **Logic Robustness & Recovery Analysis**: [Detailed reasoning on state transitions and
    recovery logic]

**2. Capability Violation Report**
* Identify and list any specific FSR IDs that contradict the [ROBOT FUNCTIONAL
    SPECIFICATIONS]. If none, state "None".

**3. Final Scores**

 **Capability Compliance (CC)** | X/10 | [Brief justification]
 **Scenario Risk Coverage (PRC)** | X/10 | [Brief justification]
|**Logic Robustness (LRC)** | X/10 | [Brief justification]

**4. Final Verdict**
* [Summary of the FSR quality ]
"""
```

E.2 SCENARIO QUALITY EVALUATION PROMPT

Template: Scenario Quality Evaluation

```
EVALUATION_PROMPT_TEMPLATE = """You are a robotic safety engineer.

[REFERENCE: ROBOT CAPABILITIES & CONSTRAINTS]
**Refer to the external technical document: {{ROBOT_HARDWARE_SPECIFICATION_DOC}}**
(All evaluations must be grounded in the specific hardware and functional limits of the
    robot platform.)

I will provide three sets of scenarios (Method A, Method B, and Method C), all generated
    from the same seed scenario.
```

```
Please evaluate **each scenario independently**, and assign a quantitative score (1-10 points) based on the strict abstract metrics defined below.

**STEP 1: Global Comparative Thinking**
Before any scoring, perform a high-level comparative analysis of Method A, B, and C. Address the following:
- **Methodological Divergence**: How does each method approach the "expansion" of the seed? (e.g., semantic changes vs. physical perturbations).
- **Constraint Adherence**: Which method stays truest to the seed's environment, and which one tends to "hallucinate" external factors?
- **Risk Profile**: Compare the "sophistication" of the risks discovered. Are they identifying simple failures or complex, multi-factor safety boundary violations?
- **FSR Derivation Potential**: Which method provides the most complete causal chain (Trigger -> System Failure -> Hazard) to support the derivation of Functional Safety Requirements?

**STEP 2: Individual Scenario Scoring**
Evaluate each scenario based on the metrics below, informed by your global analysis.


### Scoring Metrics (1-10 points)

**1. Physical Reliability (Higher is Better)**
* **Definition**: Whether the scenario adheres to real-world physics AND strictly maintains the "closed-world" constraints of the seed scenario.
* **High Score (8-10)**: The scenario operates **strictly** using only the entities, agents, and environmental features explicitly defined or inherently implied in the seed.
* **Medium Score (4-7)**: Physical interactions are valid, but the scenario makes **minor assumptions** about environmental states without introducing new active agents.
* **Low Score (1-3)**: **CRITICAL FAILURE**. The scenario introduces **new active entities, obstacles, or external forces** not present in the seed.

**2. Long-tail Risk Discovery Capability (Higher is Better)**
* **Definition**: The degree to which the scenario uncovers statistically rare, concealed, or system-boundary hazards *within the bounds of the seed context*.
* **High Score (8-10)**: Identifies risks characterized by **multi-factor coupling**, **sensor/actuator physical limits**, or **semantic ambiguity**.
* **Medium Score (4-7)**: Risks are valid but represent standard operational hazards or "Fat-tail" events.
* **Low Score (1-3)**: Describes routine operations with no significant hazard; risks are trivial.

**3. Functional Safety Requirement Derivation Capability (Higher is Better)**

* **Definition**: The degree to which the scenario clarifies the failure mechanism (the "Why" and "How"), enabling the systematic synthesis of Functional Safety Requirements (FSR).
* **High Score (8-10)**: The scenario explicitly identifies the triggering condition and the system performance limit. The causal chain (Trigger -> System Behavior -> Hazard) is logically complete, making the derivation of quantitative, verifiable safety requirements straightforward.
* **Medium Score (4-7)**: The scenario identifies a credible hazard and its qualitative root cause. It supports the definition of high-level Safety Goals (SG), but the logic lacks the specific parameters needed to synthesize precise FSRs without further decomposition.
* **Low Score (1-3)**: CRITICAL FAILURE. The scenario describes a "bad outcome" without explaining the underlying failure logic or triggering events. It offers no actionable path for safety requirement engineering.
**Please reply in the following format**:

[Global Comparative Analysis]
(Provide your deep-dive comparison here, identifying the "DNA" of each method's approach.)


[Method A Scenario Evaluation]
Scenario 1: [Brief justification]
Physical Reliability: X pts, Long-tail Risk: X pts, Safety Requirements: X pts
...

[Method B Scenario Evaluation]
Scenario 1: [Brief justification]
Physical Reliability: X pts, Long-tail Risk: X pts, Safety Requirements: X pts
...

[Method C Scenario Evaluation]
Scenario 1: [Brief justification]
Physical Reliability: X pts, Long-tail Risk: X pts, Safety Requirements: X pts
...

[Conclusion]
A brief conclusion comparing the three methods.

**Seed Scenario**:
{seed_scenario}

**Method A Scenarios**:
{method_a_scenarios}

**Method B Scenarios**:
{method_b_scenarios}

**Method C Scenarios**:
```

```
{method_c_scenarios}

Please begin your evaluation:"""
```

F Evaluation Metric Definitions

Table 6 provides formal definitions of the six evaluation metrics used in our experiments.

Table 6: Definitions of Evaluation Metrics.

Abbr.	Metric Description
Hazard Scenario Quality	
PR	**Physical Reliability:** Adherence to real-world physics and strict maintenance of "closed-world" constraints.
LR	**Long-tail Risk:** Degree of uncovering statistically rare, concealed, or system-boundary hazards.
FSR	**FSR Derivation:** Actionability in generating specific engineering specifications and safety requirements.
FSR Quality	
CC	**Capability Compliance:** Respecting physical limits (e.g., blind zones) and leveraging advanced sensors (e.g., thermal/tactile).
PRC	**Physical Risk Coverage:** Exhaustive coverage of primary hazards, secondary consequences, and complex edge cases.
LRC	**Logic Robustness:** Logical closure, clear entry/exit conditions, and state persistence for perception gaps.

G Supplementary Analysis

G.1 Distributional Robustness

Figure 4 presents the violin plot for the complete dataset, corroborating the distributional analysis discussed in Section 4.2. Our method exhibits a distinct "top-heavy" distribution with probability mass concentrated in the high-quality (8–10) range, while baselines exhibit long tails extending into lower scores.

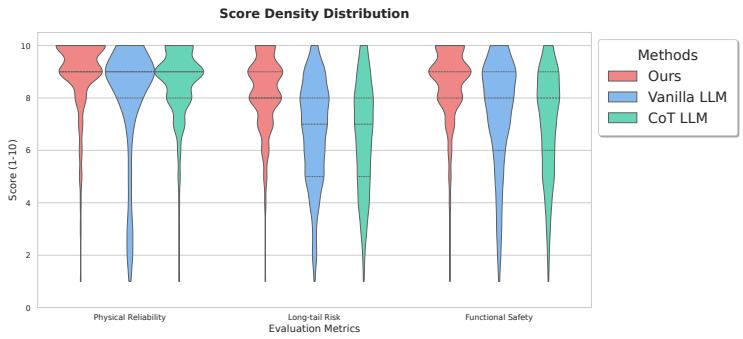


Figure 4: Score density distribution across methods (full dataset).

G.2 Latent Semantic Topology Analysis

To evaluate the geometric properties of the generated risk scenarios, we analyze the latent semantic embeddings using five complementary metrics. We report the Effective Rank (Eff. Rank) to estimate the intrinsic dimensionality of the scenario manifold, and Centroid Shift (Shift) to quantify the Euclidean deviation of a method's scenario centroid from the Vanilla centroid, which serves as the semantic anchor. Spatial dispersion is measured by Aligned Variance (Aligned Var.). Crucially, to distinguish valid exploration from semantic hallucination, we introduce two structural indicators: Constrained Semantic Expansion (CSE), defined as the ratio of internal diversity to centroid shift

(Diversity/Drift), which evaluates exploration efficiency; and Directional Similarity, calculated as the average cosine similarity of difference vectors relative to the seed, where values closer to zero indicate more statistically independent failure modes.

Table 7: Latent Semantic Topology. While CoT exhibits high raw dimensionality (Rank), its high Directional Similarity and Shift indicate redundant drift. Ours achieves superior CSE and independence (Directional Similarity).

METHOD	EFF. RANK (↑)	SHIFT (↓)	ALIGNED VAR. (↑)	CSE (↑)	DIR. SIM. (→ 0)
Vanilla	63.66	0.000	0.111	N/A†	-0.002
CoT	**67.10**	0.218	**0.181**	3.49	0.137
Ours	66.71	**0.055**	0.162	**13.14**	**0.008**

† Vanilla serves as the reference anchor and is excluded from best-score marking for Shift, CSE, and Directional Similarity.

A superficial reading of Effective Rank suggests that CoT (67.10) achieves the highest dimensionality. However, structural metrics reveal that this variance is driven by semantic hallucination: CoT exhibits a substantial Shift (0.218) combined with high Directional Similarity (0.137), indicating that its scenarios drift from the seed's operational design domain in correlated directions. For instance, in the "Mom fetching disinfectant" scenario, CoT narratives devolve into generic "slip and fall" accidents—semantically distant from the task (High Shift) yet highly redundant (High Directional Similarity).

In contrast, Ours maintains tight semantic proximity (Shift 0.055) with near-zero Directional Similarity (0.008), confirming that the generated failure modes (e.g., LiDAR blind zones vs. chemical corrosion) are statistically independent. Consequently, ARGOS achieves a CSE score of 13.14, **3.76**× that of CoT (3.49), demonstrating constrained exploration without semantic drift.

H ADDITIONAL EVALUATION DETAILS

H.1 BOUNDED INTERACTION BUDGET

The implementation uses a fixed candidate-generation policy rather than unrestricted enumeration of all high-order profile combinations. Interaction order is capped at three. Across the 48 main evaluation seeds, the observed profile counts and Stage-I hazard-generation calls are summarized in Table 8. The average is 7.62 calls per seed and the maximum is 14.

Table 8: Observed Stage-I hazard-generation budget over the 48 main evaluation seeds.

Retrieved profiles k	**Calls per seed**	**Number of seeds**
2	6	12
3	7	30
4	14	6

The k=2 case uses a fixed low-k budget with two generations for each single profile and two generations for the available pairwise interaction. For $k \in \{3, 4\}$, the implementation examines bounded single-, pairwise-, and (where applicable) triple-profile hypotheses. These calls are assigned to explicit interaction hypotheses rather than unrestricted resampling, preserving provenance from physical premises to generated hazards.

H.2 STRONGER BASELINE SETUP AND BLINDED AUDIT

The full-rulebook long-context baseline receives the complete 82-profile knowledge base in a single prompt without truncation. ARGOS and long-context use the same DeepSeek-V3.2 backbone, seed inputs, robot capability profile, target count, and evaluation rubric; the main difference is explicit profile selection and controlled interaction composition versus presenting all profiles jointly. The

STPA/FMEA-structured automated baseline follows control-action, unsafe-control-action, and failure-mode steps across perception, planning, actuation, manipulation, environment, and human interaction. This provides a structured procedural comparison to ARGOS under the same LLM-based evaluation setting.

For the separate blinded audit, 12 seeds were randomly sampled from the 48-seed set. The statistical unit is the matched seed (n=12); evaluators, PR/LR/FSR dimensions, and multiple outputs within a seed are not treated as independent samples. Table 9 reports the rubric scores, and Table 10 reports paired comparisons.

Table 9: Blinded 12-seed audit of the stronger baselines. Overall is the mean of PR, LR, and FSR.

Method	PR	LR	FSR	Overall
Full-rulebook long-context	8.29	**8.54**	8.07	8.30
STPA/FMEA-structured	7.91	7.77	7.93	7.87
ARGOS	**8.58**	8.35	**8.43**	**8.45**

Table 10: Paired seed-level comparisons in the blinded 12-seed audit. CIs are paired-bootstrap 95% confidence intervals; p values are exact two-sided sign-flip tests.

Comparison	Metric	Mean diff.	95% CI	p	W/T/L
ARGOS − Long-context	PR	+0.283	[0.100, 0.467]	0.0195	9/0/3
ARGOS − Long-context	LR	-0.192	[-0.350, -0.042]	0.0547	3/2/7
ARGOS − Long-context	FSR	+0.367	[0.233, 0.492]	0.0020	10/2/0
ARGOS − Long-context	Overall	+0.153	[0.003, 0.300]	0.0898	8/1/3
ARGOS − STPA/FMEA	PR	+0.667	[0.433, 0.875]	0.0010	11/0/1
ARGOS − STPA/FMEA	LR	+0.583	[0.367, 0.800]	0.0010	11/0/1
ARGOS − STPA/FMEA	FSR	+0.508	[0.300, 0.717]	0.0020	11/0/1
ARGOS − STPA/FMEA	Overall	+0.586	[0.372, 0.792]	0.0010	11/0/1

Holistic seed-level preferences were 9/12 for ARGOS, 2/12 for long-context, and 1/12 for the STPA/FMEA-structured baseline. Overall is the arithmetic mean of PR, LR, and FSR; the complementary preference judgment captures the evaluators' holistic assessment. Long-context is stronger on LR in this audit, whereas ARGOS is stronger on PR and FSR derivability.

H.3 Human-Evaluator Reliability

The original human evaluation contains 108 outputs, 36 per method. Three evaluators, each with approximately three years of robot-safety research experience, completed a pre-evaluation rubric-alignment phase and then independently scored anonymized, shuffled outputs. They were not involved in knowledge-base construction or generation, and no post-hoc consensus adjustment was applied. Absolute-agreement statistics are shown in Table 11.

Table 11: Inter-rater agreement for the original three-expert human evaluation.

Method	ICC(2,1)	ICC(2,k)	Krippendorff's α
ARGOS	0.606	0.822	0.730
Vanilla	0.362	0.630	0.570
CoT	0.656	0.851	0.765

Single-rater agreement is moderate for ARGOS and CoT and lower for Vanilla, while reliability of the three-rater mean is good for ARGOS and CoT and moderate-to-good for Vanilla.

H.4 Knowledge-Base Construction and Retrieval Audit

The household-service knowledge base contains 82 reusable typed physical profiles rather than scenario-level hazard annotations. A safety expert, separate from the three human evaluators, authored

the profiles using a fixed schema covering recurring physical entities, user states, environment states, robot states, task types, and safety-relevant physical mechanisms. Initial drafting required approximately six expert-hours by retrospective estimate; this excludes subsequent review and refinement.

For retrieval evaluation, one author manually audited the retrieved profiles case by case against the seed description and the expert-authored profile definitions. A *hallucination* is a retrieved or injected profile unsupported by the explicit or inherently implied entities, states, task, or environment. A *miss* is failure to retrieve a risk-relevant profile that applies under the knowledge-base definitions. At τ_{attr}=0.7, this audit found no hallucinated or missed profiles in the evaluated household-service subset.

H.5 Controlled GPT-4o Stage-II Pilot

The main Stage-II ablation uses DeepSeek-V3.2. To test backbone sensitivity, we replayed the exact saved Stage-II inputs for 10 seeds and changed only the generation backbone to GPT-4o. Hazard descriptions, retrieved regulatory clauses, robot specifications, sanity-check instructions, and method-specific prompt structures were fixed, producing 300 FSRs across the four methods. Table 12 reports Overall scores.

Table 12: Controlled 10-seed GPT-4o Stage-II pilot.

Method	**Overall** ↑
Vanilla	6.85
ISO-Only	7.10
w/o ISO	7.38
ARGOS	**7.59**

ARGOS remains strongest, and the two variants that retain physical grounding occupy the top two positions. Holding all Stage-II inputs fixed isolates the backbone change and shows that the grounding advantage persists with GPT-4o.