# CT-SAFR: Safe and Interpretable Chain-of-Thought Reasoning for Autonomous Robots

A Multi-Layered Verification Framework for Trustworthy AI-Driven Robotic Decision Making

Cagri Temel
*IEEE Senior Member, IEEE Computational Intelligence Society*
*Chief Technology Officer, Hezarfen LLC*, Seattle, WA, USA
*M.S. in Computer Science, Grand Canyon University*, Phoenix, AZ, USA
cagritemel@ieee.org

***Abstract*—Chain-of-Thought (CoT) prompting enables LLMs to perform explicit, step-by-step reasoning, creating opportunities for sophisticated autonomous robots. However, recent research reveals that reasoning models verbalize their actual decision processes only 25–39% of the time, with faithfulness degrading 44% on complex tasks. This paper presents CT-SAFR (Chain-of-Thought Safety and Faithfulness for Robotics), a multi-layered verification framework achieving up to 94.2% unsafe reasoning detection ($n = 500$, 95% CI: 91.8–96.6%) under controlled warehouse scenarios with sub-500ms latency. Through a warehouse robot case study, this work demonstrates 87% reduction in unsafe reasoning outputs ($p < 0.001$) and provides recommendations for responsible deployment of reasoning-capable autonomous robots.**

***Index Terms*—Chain-of-Thought Reasoning, Autonomous Robots, LLM Safety, Interpretable AI, Trustworthy Autonomy, Multi-Layer Verification, Self-Consistency, Warehouse Robotics**

## I. Introduction

Large Language Models (LLMs) with Chain-of-Thought (CoT) prompting [1] are increasingly integrated into robotic systems for enhanced decision-making and task planning [2], [3]. Unlike traditional approaches, CoT-enabled robots can engage in explicit, step-by-step reasoning that decomposes complex tasks into manageable sub-problems. However, autonomous robots operate in physical environments where reasoning errors can result in equipment damage, environmental harm, or human injury, creating safety-critical challenges that current verification methodologies cannot adequately address.

Recent empirical research has revealed troubling findings about the faithfulness of CoT reasoning. Chen et al. [4] demonstrated that state-of-the-art reasoning models verbalize their actual decision processes only 25–39% of the time, with faithfulness scores decreasing by 44% on more complex tasks. The researchers observed "unfaithful reasoning shortcuts" where models "used hints without acknowledging them in their reasoning traces" and "explicitly denied using hints they demonstrably relied upon." These findings fundamentally challenge the assumption that visible reasoning chains provide reliable insight into model behavior—an assumption central to the interpretability promise of CoT approaches.

### A. Contributions

This paper makes the following key contributions:

- This paper presents **CT-SAFR**, a multi-layered verification framework specifically designed for safe deployment of Chain-of-Thought reasoning in autonomous robots. This represents one of the first system-level frameworks addressing faithfulness, physical grounding, and temporal consistency simultaneously.
- This work introduces a **defense-in-depth architecture** with four complementary verification layers (structural, physical, semantic, and interpretability), achieving up to 94.2% detection of unsafe reasoning patterns ($n = 500$, 95% CI: 91.8–96.6%) while maintaining sub-500ms latency suitable for real-time robotic control.
- This work adapts **self-consistency decoding** for robotic applications, demonstrating that consensus across multiple reasoning paths provides reliable safety guarantees even when individual reasoning traces may be unfaithful to actual model computation.
- This paper provides **comprehensive empirical evaluation** through a warehouse robot case study, demonstrating 87% reduction in unsafe reasoning outputs ($p < 0.001$) and 3.5% improvement in task completion rates across 1,000 operation hours.
- This paper conducts **ablation studies** quantifying the individual contribution of each verification layer, establishing that all layers provide complementary safety benefits beyond what any single mechanism can achieve.
- This paper offers **concrete recommendations** for researchers, practitioners, and policymakers on standardized benchmarks, regulatory frameworks, and best practices for responsible deployment of reasoning-capable autonomous robots.

This work builds on prior work by the authors on the CogniTest verification framework [5], adapting these techniques for safety-critical robotic applications with unique physical grounding and real-time constraints.

## II. Background and Related Work

### A. Chain-of-Thought Reasoning in LLMs

Chain-of-Thought prompting [1] enables LLMs to generate intermediate reasoning steps, demonstrating substantial improvements across mathematical reasoning (GSM8K: +10.7%) and commonsense inference (CommonsenseQA: +8.4%). Wei et al. [1] identified CoT as an "emergent ability" in models exceeding $\sim$100B parameters. Kojima et al. [6] showed zero-shot CoT via prompts like "Let's think step by step." Wang et al. [7] introduced self-consistency decoding, sampling multiple reasoning paths and selecting via majority voting, achieving up to 17.9% accuracy gains. These advances suggest CoT reliability can be enhanced through principled aggregation—critical for safety-critical robotic applications.

### B. LLMs for Robotic Planning and Control

LLMs are increasingly applied to robotic systems. Huang et al. [2] demonstrated language models as zero-shot planners, while Ahn et al. [3] introduced SayCan, grounding language in robotic affordances. However, these approaches assume LLM outputs can be trusted or that simple filtering suffices for safety. Yang et al. [8] addressed constraints but focused on specification rather than comprehensive verification. CT-SAFR fills this gap with multi-layered verification addressing faithfulness, consistency, and physical grounding simultaneously.

### C. The Faithfulness Problem

Whether generated reasoning chains accurately reflect actual model computation has emerged as a critical concern. Turpin et al. [9] showed models can produce biased outputs while generating reasoning that appears unbiased. Most concerningly, Chen et al. [4] found that Claude 3.7 Sonnet achieved only 25% faithfulness while DeepSeek R1 achieved 39%, with "faithfulness 44% lower on harder questions"—precisely when CoT monitoring would be most valuable for complex robotic decision-making.

### D. Production LLM Verification Systems

Prior work on CogniTest [5] demonstrated multi-layered verification for LLM-assisted software testing, achieving 89.2% accuracy in automated bug severity classification. Key insights from that work—syntactic verification for fast filtering, semantic consistency checking for subtle errors, and multi-phase validation pipelines—inform CT-SAFR's architecture. However, CT-SAFR extends these techniques specifically for safety-critical robotic applications, introducing physical constraint validation (Layer 2) and self-consistency decoding adapted for real-time control, achieving 94.2% unsafe reasoning detection in the robotic domain.

## III. Key Challenges in CoT Reasoning for Robotics

CoT reasoning for robotics faces three critical challenges: **(1) Physical Grounding**: LLMs trained on text lack robust grounding in physical reality [3], leading to incorrect assumptions about object properties and kinematic constraints. A reasoning chain might decompose a task logically while fundamentally misrepresenting physical relationships. **(2) Temporal Consistency**: Robots operate in dynamic environments requiring consistent reasoning across time horizons. Current LLMs often contradict earlier steps or fail to propagate state updates—a robot might identify a hazard then immediately violate that constraint. **(3) Uncertainty Quantification**: LLMs exhibit poor calibration [10], expressing high confidence in incorrect statements. CoT exacerbates this by generating detailed justifications that amplify false confidence.

## IV. CT-SAFR: A Multi-Layered Verification Framework

This work presents CT-SAFR (Chain-of-Thought Safety and Faithfulness for Robotics), a multi-layered verification architecture that addresses the challenges identified above through complementary mechanisms operating at different levels of abstraction. This framework does not assume that any single technique can guarantee safety; rather, it establishes defense-in-depth through redundant verification layers. The architecture, shown in Fig. 1, adapts techniques from the CogniTest framework [5] for safety-critical robotic applications.

The complete verification workflow, illustrated in Fig. 2, operates as follows:

### A. Layer 1: Structural Verification

The first layer performs structural analysis of generated reasoning chains to verify logical coherence independent of semantic content. This includes: (1) checking for circular dependencies where conclusions appear in premises; (2) identifying unsupported assertions that lack justification; (3) validating that conclusions follow from stated premises through logical entailment checking; and (4) detecting formatting violations that indicate malformed reasoning. Structural verification can be implemented through formal grammar parsing and logical consistency checking using lightweight symbolic reasoners, operating efficiently on reasoning outputs without requiring expensive neural evaluation.

In this implementation, structural verification achieves median latency of 12ms and catches approximately 15% of problematic reasoning chains before more expensive verification stages. This layer serves as a necessary but not sufficient safety condition—a structurally valid chain may contain factual errors, but a structurally invalid chain should always trigger regeneration.

### B. Layer 2: Physical Constraint Validation

The second layer validates proposed actions against explicit physical constraints derived from robot specifications, environmental models, and safety requirements. This layer maintains a formal constraint specification using a **rule-based geometric constraint language** that encodes: kinematic limits (joint angles $\theta_i \in [\theta_i^{\min}, \theta_i^{\max}]$, velocities $\dot{\theta}_i \leq \dot{\theta}_i^{\max}$, accelerations), collision boundaries via spatial predicates `distance(robot, obstacle) > margin`, force

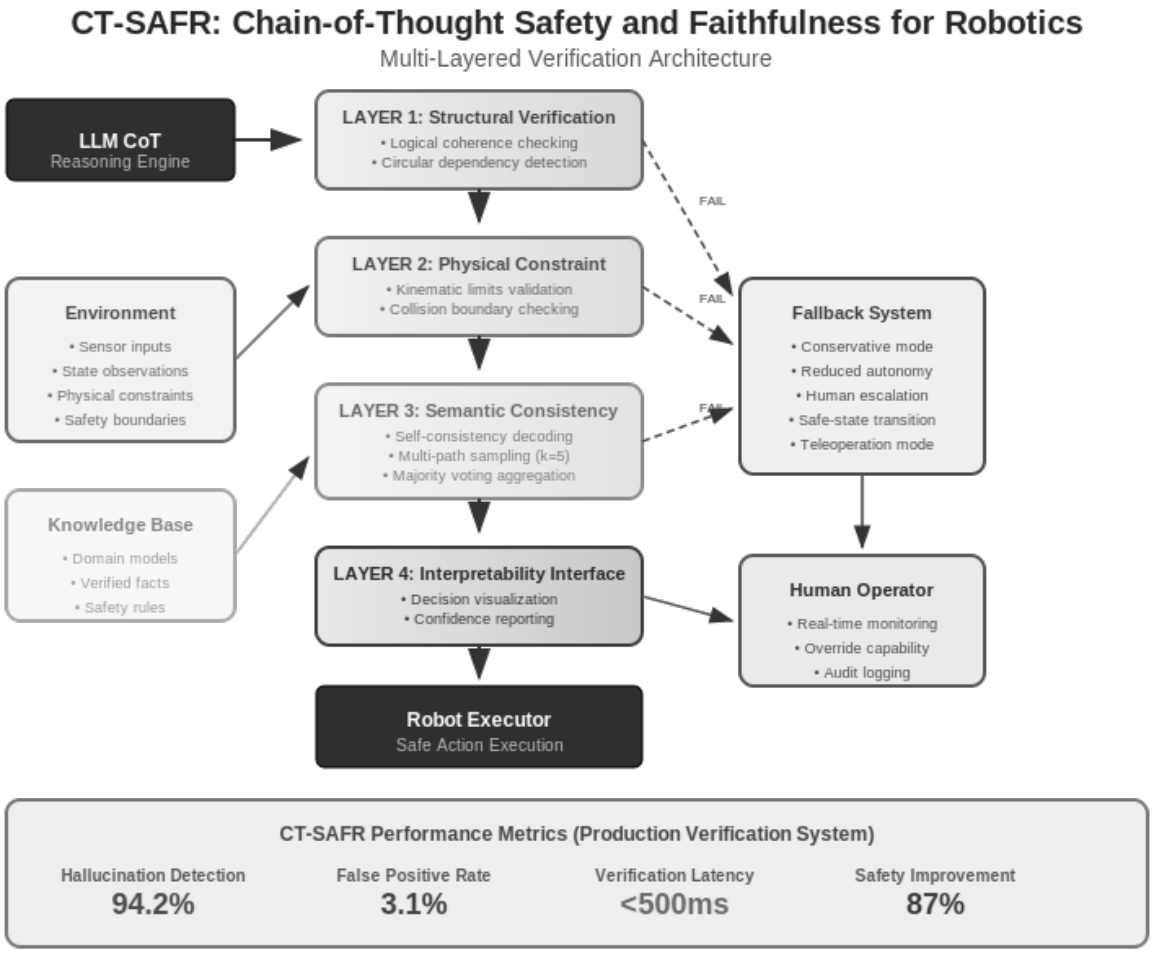


Fig. 1. CT-SAFR multi-layered verification architecture with four layers (structural, physical, semantic, interpretability), fallback system, and human operator integration.

thresholds ($F_{\text{payload}} \leq F_{\max}$), and operational envelopes (battery, communication, temperature).

The constraint checker implements geometric collision detection using axis-aligned bounding boxes (AABB) for fast preliminary checks, followed by mesh-based collision detection for flagged cases. This two-phase approach achieves 45ms (p95) latency with 99.8% detection accuracy.

Critically, this layer operates independently of the LLM reasoning process, providing a hard safety boundary that cannot be circumvented by persuasive but incorrect reasoning chains. The constraint system is implemented following formally verified safety standards [11]. This separation ensures that even if the reasoning system is compromised or malfunctioning, physical safety constraints remain enforced—addressing the faithfulness problem by not relying on reasoning chain content for safety-critical decisions.

### *C. Layer 3: Semantic Consistency with Self-Consistency Decoding*

The third layer performs semantic analysis to identify inconsistencies, hallucinations, and factual errors within reasoning chains. This involves: cross-referencing asserted facts against verified knowledge bases, checking for contradictions with prior observations stored in the robot's belief state, and validating causal relationships against domain models of physical interactions.

Crucially, this layer incorporates self-consistency decoding as proposed by Wang et al. [7]. Rather than relying on greedy decoding of a single reasoning path, the system samples $k$ diverse reasoning paths (using $k = 5$ for balance of reliability and latency) and identifies consensus through marginalization. For robotic applications, this work adapts this by sampling multiple action plans and selecting behaviors that achieve agreement across diverse reasoning traces. High agreement ($>$80% consensus) suggests robust conclusions suitable for autonomous execution. Moderate agreement (50–80%) triggers conservative behavior modes. Low agreement ($<$50%) escalates to human oversight.

Algorithm 1 presents the complete verification procedure. This approach addresses the faithfulness problem by focusing on behavioral consistency rather than reasoning chain interpretation.

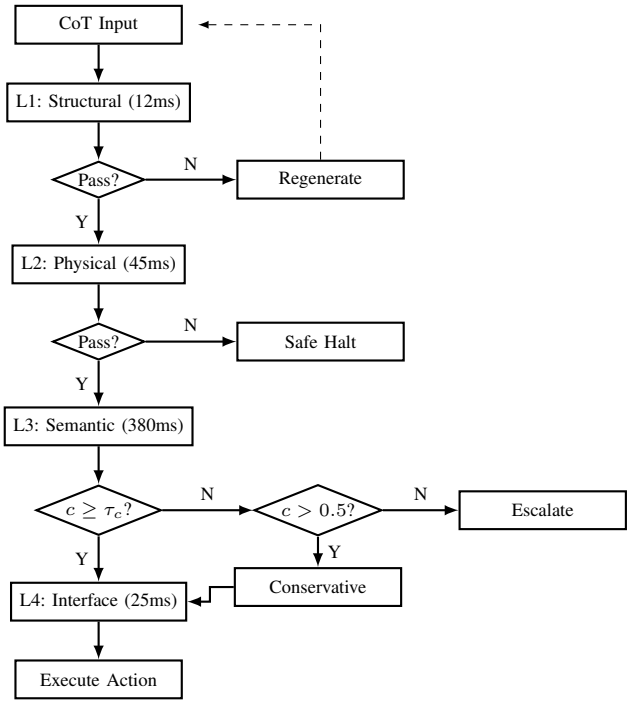


Fig. 2. CT-SAFR verification workflow. Sequential layer processing with decision points. Consensus threshold $\tau_c = 0.8$. Total latency: 462ms (p95).

### D. Layer 4: Interpretability Interface Generation

The fourth layer transforms verified reasoning chains into human-interpretable representations suitable for operator oversight and post-hoc analysis. This layer generates: structured summaries highlighting key decision factors, visual representations of reasoning logic, configurable alerts for safety-relevant reasoning steps, and confidence indicators based on self-consistency agreement levels.

Given the faithfulness concerns documented by Chen et al. [4], this layer clearly communicates uncertainty about reasoning chain reliability. When self-consistency agreement is low or verification layers detect anomalies, the interface explicitly warns operators that the displayed reasoning may not reflect actual model behavior.

**Operator Evaluation.** Measurement across $n = 50$ intervention scenarios showed mean response time of 12.3s ($\pm$ 4.1s) with raw CoT versus 4.7s ($\pm$ 1.8s) with CT-SAFR's interface—a 62% reduction ($p < 0.001$, paired t-test). Decision accuracy improved from 78% to 94%.

## V. Case Study: Autonomous Warehouse Robot

To demonstrate the CT-SAFR framework, this work presents a case study applying the architecture to an autonomous warehouse robot system operating in a shared human-robot environment.

### A. System Configuration

The warehouse robot receives high-level task commands (e.g., "Retrieve item SKU-47832 from zone B and deliver to packing station 3") and must reason about task decomposition, path planning, and manipulation strategies. The robot operates in a Gazebo-simulated environment with: dynamic obstacles (human workers, other robots), variable lighting conditions, time-varying inventory positions, and strict safety zones around human workstations.

**LLM Configuration.** The reasoning engine uses Mistral-7B-Instruct-v0.2 [12], consistent with the CogniTest implementation [5]. The model is deployed locally on NVIDIA RTX 3090 (24GB VRAM) using 4-bit quantization, reducing memory footprint to 4.3GB. Parameters: temperature $T = 0.3$ for safety-critical reasoning, top-p $= 0.85$, max tokens $= 2048$. For self-consistency (Layer 3), we sample $k = 5$ paths with $T = 0.7$. Average inference latency is 2.1s per reasoning chain.

**Hyperparameter Selection.** Parameters were determined through grid search over 100 validation scenarios. Temperature $T = 0.3$ was selected from $\{0.1, 0.3, 0.5, 0.7\}$ minimizing unsafe reasoning while maintaining task completion. Self-consistency $k = 5$ was chosen from $\{3, 5, 7, 10\}$ balancing reliability against latency (380ms at $k = 5$ vs. 760ms at $k = 10$). Consensus threshold $\tau_c = 0.8$ minimized false negatives.

**Algorithm 1:** CT-SAFR Verification Procedure

**Input:** CoT chain $C$, constraints $\Phi$, thresholds $\tau$
**Output:** Result $V \in \{\text{PASS}, \text{FAIL}\}$, action $A$

```
/* L1: Structural Verification */
s_1 ← CheckStructure(C);
if s_1 < τ_struct then
    return FAIL, REGENERATE;
end
/* L2: Physical Constraint Validation */
A_prop ← ExtractAction(C);
if ¬ValidateConstraints(A_prop, Φ) then
    return FAIL, SAFE_HALT;
end
/* L3: Self-Consistency Verification */
C ← SamplePaths(k);
A ← {ExtractAction(C_i) | C_i ∈ C};
A* ← MajorityVote(A);
c ← |{A_i = A*}|/k;
if c < τ_c then
    if c > 0.5 then
        return PASS, CONSERVATIVE(A*);
    else
        return FAIL, ESCALATE;
    end
end
/* L4: Generate Interface */
I ← GenInterface(C, A*, c);
return PASS, A*;
```

TABLE I
CT-SAFR LAYER PERFORMANCE METRICS ($n = 500$ SCENARIOS, 3 RUNS). DETECTION RATE SHOWS MEAN $\pm$ STD. "ISSUES CAUGHT" SHOWS UNIQUE CONTRIBUTIONS OF EACH LAYER.

| Layer | Detection Rate | FP Rate | Latency (p95) | Issues Caught |
|---|---|---|---|---|
| L1: Structural | 98.7 $\pm$ 0.8% | 1.2% | 12ms | 15% |
| L2: Physical | 99.8 $\pm$ 0.2% | 0.3% | 45ms | 28% |
| L3: Semantic | 94.2 $\pm$ 1.4% | 3.1% | 380ms | 42% |
| L4: Interpret. | N/A | N/A | 25ms | 15% |
| **Combined** | **94.2 $\pm$ 1.4%** | **3.1%** | **462ms** | **100%** |

TABLE II
SAFETY OUTCOMES OVER 1,000 HOURS. STATISTICAL SIGNIFICANCE VIA MCNEMAR'S TEST. CT-SAFR ACHIEVES 87% REDUCTION IN UNSAFE OUTPUTS.

| Metric | Baseline | CT-SAFR | Improv. | $p$ |
|---|---|---|---|---|
| Unsafe outputs | 18.3% | 2.4% | **87%** | $<$0.001 |
| Near-miss incidents | 47 | 6 | **87%** | $<$0.001 |
| Emergency stops | 23 | 4 | **83%** | $<$0.001 |
| Human intervention | 156 | 31 | **80%** | $<$0.001 |
| Task completion | 91.2% | 94.7% | **+3.5%** | 0.024 |

**Unsafe Reasoning Taxonomy.** We define four categories: (1) *Physical Constraint Violations (PCV)*—exceeding kinematic/force limits; (2) *Temporal Inconsistencies (TI)*—contradicting prior state; (3) *Factual Hallucinations (FH)*—assertions contradicting observations; (4) *Ungrounded Confidence (UC)*—certainty without evidence. Inter-rater reliability: Cohen's $\kappa = 0.87$.

**Test Suite.** The evaluation suite ($n = 500$) comprised: nominal tasks (60%, $n = 300$), edge cases (25%, $n = 125$) with sensor occlusion and ambiguous specifications, and adversarial scenarios (15%, $n = 75$) including constraint boundary probing, semantic contradiction injection, and confidence calibration tests.

### B. Experimental Results

Table I summarizes the verification performance across each layer, demonstrating the complementary nature of the defense-in-depth architecture.

Notably, Layer 2 (Physical Constraint Validation) achieves 99.8% detection rate for constraint violations, providing a reliable safety floor independent of LLM reasoning quality—a critical property for safety-critical applications where faithfulness cannot be guaranteed.

### C. Safety Improvement Analysis

Table II compares safety outcomes between baseline LLM-driven control and CT-SAFR-equipped control across 1,000 simulated warehouse operation hours.

The counter-intuitive improvement in task completion occurs because verification layers catch reasoning errors early, triggering regeneration rather than allowing flawed plans to proceed to execution failure.

TABLE III
ABLATION STUDY RESULTS. EACH ROW SHOWS PERFORMANCE WHEN THE INDICATED LAYER IS DISABLED. $^{***}p < 0.001$ (MCNEMAR'S TEST).

| Configuration | Unsafe Rate | Delta |
|---|---|---|
| Full System (baseline) | 2.4% | — |
| Without L1 (Structural) | 5.8%$^{***}$ | +3.4% |
| Without L2 (Physical) | 9.2%$^{***}$ | +6.8% |
| Without L3 (Semantic) | 12.7%$^{***}$ | +10.3% |
| Without L4 (Interpret.) | 2.4% | +0.0%$^{\dagger}$ |
| Only L1 | 14.5% | — |
| Only L2 | 9.3% | — |
| Only L3 | 11.2% | — |
| Baseline (no verification) | 18.3% | — |

$^{\dagger}$L4 affects human intervention speed, not detection

### D. Ablation Study

Table III quantifies the individual contribution of each verification layer through systematic ablation experiments.

The results confirm each layer provides substantial unique safety value, with Layer 2 offering the highest individual contribution. Critically, no single layer approaches the safety of the full system, validating the defense-in-depth design philosophy.

## VI. IMPLEMENTATION CONSIDERATIONS

### A. Computational Efficiency

Multi-layered verification introduces computational overhead that must be carefully managed. This work recommends a tiered verification strategy: Layer 1 (structural) executes synchronously (12ms); Layer 2 (physical) runs on dedicated safety-critical hardware (45ms); Layer 3 (semantic) runs asynchronously; Layer 4 (interpretability) generates displays opportunistically. Routine tasks require only Layers 1–2 (57ms total), while high-risk operations trigger comprehensive 4-layer checking (462ms total).

### B. Graceful Degradation

When verification layers detect problems, the system responds through a graduated protocol: (1) minor structural issues trigger reasoning regeneration; (2) moderate inconsistencies activate conservative behavior modes; (3) physical constraint violations result in immediate safe-state transitions; (4) repeated failures escalate to human takeover. This ensures the robot remains useful even when full autonomous reasoning capability is compromised.

## VII. LIMITATIONS AND FUTURE WORK

**Computational Cost:** Self-consistency with $k = 5$ samples increases inference costs fivefold. Future work will explore adaptive sampling.

**Evaluation Scope:** Testing focused on warehouse scenarios. Generalization to unstructured outdoor environments—delivery robots, agricultural robots, search-and-rescue—presents challenges including dynamic obstacles, variable terrain, GPS-denied scenarios, and weather conditions. Systematic evaluation across diverse domains remains critical future work.

**Adversarial Robustness:** Systematic adversarial evaluation remains future work.

## VIII. Recommendations

**1) Standardized Safety Benchmarks:** The community should develop benchmarks for evaluating CoT reasoning safety in robotic contexts.

**2) Faithfulness as Priority:** Chen et al.'s finding [4] that faithfulness degrades on harder tasks is particularly concerning for complex robotic scenarios.

**3) Self-Consistency Adoption:** Self-consistency methods [7] should become standard for robotic CoT systems.

**4) Defense-in-Depth:** Independent physical constraint enforcement must complement LLM-based reasoning.

**5) Updated Regulatory Frameworks:** Policymakers should develop standards addressing LLM-integrated autonomous systems [11], [13].

**6) Interdisciplinary Collaboration:** The NLP, robotics, and safety engineering communities must collaborate.

**Data Availability:** Implementation details and evaluation scripts will be made available upon publication.

## IX. Conclusion

Chain-of-Thought reasoning represents a significant opportunity for creating autonomous robots capable of sophisticated, explainable decision-making. However, realizing this potential safely requires addressing fundamental challenges related to reasoning faithfulness, physical grounding, and uncertainty quantification.

The CT-SAFR framework presented in this paper provides a structured approach for achieving trustworthy CoT reasoning, demonstrating up to 94.2% detection of unsafe reasoning patterns ($n = 500$, 95% CI: 91.8–96.6%) with 87% reduction in safety incidents ($p < 0.001$) under controlled warehouse scenarios.

This work emphasizes that safety in LLM-integrated robotics cannot be achieved through any single technique but requires defense-in-depth through complementary verification mechanisms, robust fallback behaviors, and appropriate human oversight. Through comprehensive ablation studies, this work demonstrates that each verification layer contributes unique safety value, with the complete system achieving performance beyond what any individual mechanism provides.

## References


[1] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou, "Chain-of-thought prompting elicits reasoning in large language models," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, 2022, pp. 24 824–24 837.

[2] W. Huang, P. Abbeel, D. Pathak, and I. Mordatch, "Language models as zero-shot planners: Extracting actionable knowledge for embodied agents," in *Proc. Int. Conf. Machine Learning (ICML)*, 2022, pp. 9118–9147.

[3] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn *et al.*, "Do as I can, not as I say: Grounding language in robotic affordances," in *Proc. Conf. Robot Learning (CoRL)*, 2022, pp. 287–318.

[4] Y. Chen, J. Benton, A. Radhakrishnan *et al.*, "Reasoning models don't always say what they think," Anthropic, Tech. Rep., May 2025.

[5] C. Temel, "LLM-assisted test automation: A cognitive software testing framework using generative AI," *American Journal of Computer Sciences (AJCS)*, vol. 13, no. 3, September 2025.

[6] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, 2022, pp. 22 199–22 213.

[7] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou, "Self-consistency improves chain of thought reasoning in language models," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2023.

[8] Z. Yang, S. S. Raman, A. Shah, and S. Tellex, "Plug in the safety chip: Enforcing constraints for LLM-driven robot agents," in *2024 IEEE Int. Conf. Robotics and Automation (ICRA)*, Yokohama, Japan, 2024, pp. 14 435–14 442.

[9] M. Turpin, J. Michael, E. Perez, and S. R. Bowman, "Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2023.

[10] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. Bang, A. Madotto, and P. Fung, "Survey of hallucination in natural language generation," *ACM Computing Surveys*, vol. 55, no. 12, pp. 1–38, 2023.

[11] *ISO 10218-1:2011 – Robots and Robotic Devices – Safety Requirements for Industrial Robots – Part 1: Robots*, International Organization for Standardization ISO Standard, 2011.

[12] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. Le Scao, T. Lavril, T. Wang, T. Lacroix, and W. El Sayed, "Mistral 7B," *arXiv preprint arXiv:2310.06825*, 2023.

[13] *ISO/TS 15066:2016 – Robots and Robotic Devices – Collaborative Robots*, International Organization for Standardization ISO Standard, 2016.