Composing CRUCIBLE and SAFE-V: Multi-Agent LLM Verification as a Runtime Assurance Mechanism within the AI Functional Safety V-Model

Jherrod Thomas
Independent Researcher
Jherrod.thomas@aol.com | www.jherrodthomas.com

***Abstract*—Two prior companion papers respectively specify CRUCIBLE, a model-agnostic multi-agent algorithm that forces a heterogeneous ensemble of large language models to cross-examine one another, cite primary sources, and reach Byzantine-fault-tolerant consensus on every atomic claim; and SAFE-V, a systems-engineering safety framework that adapts ISO 26262 functional safety and ISO 21448 SOTIF methodology to a dual-arm V-model for artificial intelligence, complete with an AI-specific integrity level (AISIL) parameterized by severity, exposure, controllability, and epistemic uncertainty. This paper composes the two: CRUCIBLE is formalized as a runtime assurance mechanism that instantiates specific entries in the SAFE-V safety-mechanism catalogue, and SAFE-V supplies the lifecycle discipline under which CRUCIBLE itself is developed, verified, and maintained. We present (i) a stage-to-mechanism traceability matrix that maps each of CRUCIBLE's nine stages onto SAFE-V's input-monitor, output-monitor, redundancy, supervision, and reaction categories; (ii) a procedure for estimating Operational Diagnostic Coverage (ODC) of CRUCIBLE against each AI-HARA hazard class from its six-test stress battery; (iii) a Goal-Structuring-Notation assurance sub-argument that contributes CRUCIBLE evidence to a SAFE-V safety case; (iv) an AISIL calibration that specifies when CRUCIBLE is sufficient, when it must be paired with an independent fallback, and when it is not appropriate as a sole mechanism; and (v) a worked example of a medical-triage assistant in which CRUCIBLE verifies clinical-claim citations within the SAFE-V Technical Safety Concept's Fault Tolerant Time Interval.**

Index Terms*—*AI safety, assurance case, diagnostic coverage, functional safety, large language models, multi-agent systems, runtime assurance, Simplex architecture, SOTIF, V-model.

I. Introduction

AI SAFETY engineering has two distinct problems. The first is *what the system produces* at inference time: does its output contain hallucinated facts, fabricated citations, unsupported claims, or artefacts of sycophancy and agreeableness [3], [4]? The second is *how the system was built, is operated, and is maintained*: does its engineering lifecycle satisfy the standards of care that other safety-critical industries have spent forty years codifying in IEC 61508, ISO 26262, DO-178C, IEC 62304, and, for performance-insufficiency hazards, ISO 21448 SOTIF [5], [6], [7]? Two prior companion papers address each of these problems separately.

CRUCIBLE [1] specifies a model-agnostic orchestration algorithm in which a heterogeneous ensemble of LLM agents drafts independently, cross-examines one another under an adversarial protocol, cites only from a tier-scored allowlist of primary sources, verifies every citation through URL resolution and textual entailment, reaches Byzantine-fault-tolerant consensus on each atomic claim, and survives a battery of six stress tests before synthesizing a final answer accompanied by a user-inspectable provenance ledger. CRUCIBLE is a *runtime* mechanism: it runs whenever a query is issued and produces evidence about the trustworthiness of a single answer.

SAFE-V [2] specifies a systems-engineering framework in which AI hazards are elicited through AI-HARA, classified by an AISIL derived from severity, exposure, controllability, and epistemic uncertainty, and addressed across a dual-arm V-model that runs classical functional-safety engineering in parallel with an ML engineering arm for data, model, and operational-design-domain governance. SAFE-V's right-arm verification-and-validation cascade binds every implementation artefact to an explicit verification step, and its post-deployment loop keeps the safety case alive through drift monitors, shadow deployments, and automatic hazard re-analysis. SAFE-V is a *lifecycle* framework: it runs continuously across development, verification, deployment, and decommissioning, and produces a safety case.

Neither paper is sufficient alone. A correctly orchestrated CRUCIBLE pipeline answering a single query from an

unsafe development lifecycle, without ODD specification, without a HARA, without a safety case, and without continuous monitoring, is an impressive demonstration but is not certifiable. A correctly engineered SAFE-V system whose TSC-mandated runtime monitor is specified as "TBD" and whose input/output-monitor category is empty is a framework with a hole in its safety-mechanism catalogue. This paper closes that gap. Its contributions are: (1) a stage-to-mechanism mapping that places every CRUCIBLE stage within SAFE-V's mechanism catalogue, so that the CRUCIBLE algorithm can be invoked in a safety case by reference rather than re-specified; (2) an ODC-derivation procedure that converts CRUCIBLE's six-test stress battery into ISO 26262–style diagnostic-coverage numbers against each AI-HARA hazard class; (3) a GSN [8] assurance sub-argument template that slots CRUCIBLE evidence into a SAFE-V safety case; (4) an AISIL-calibration rule that tells the practitioner when CRUCIBLE is sufficient, when it must be paired with an independent mechanism, and when it cannot stand alone; and (5) a worked example.

Section II recapitulates the minimal CRUCIBLE and SAFE-V machinery needed to read the composition. Section III presents the integration architecture. Section IV gives the stage-to-mechanism mapping. Section V derives ODC from the stress battery. Section VI presents the GSN sub-argument. Section VII provides AISIL calibration. Section VIII works the medical-triage example. Section IX discusses the meta-recursion (CRUCIBLE is itself an AI system subject to SAFE-V). Section X addresses threats to validity. Section XI concludes.

II. Background

A. CRUCIBLE in One Page

CRUCIBLE [1] composes a heterogeneous ensemble of N generator agents and M critic agents with $n = N + M \geq 3f + 1$ so that up to f Byzantine participants are tolerated. Its nine stages are: (1) query atomization into verifiable atomic claims, following FActScore [9]; (2) deterministic source gating that enforces an allowlist, a blocklist of user-generated-content forums, and a continuous CRAAP/SIFT tier score [10], [11]; (3) independent drafting by each generator without cross-contamination, following Mixture-of-Agents [12]; (4) adversarial cross-examination by critics who must supply counter-evidence, in the spirit of Constitutional AI [13]; (5) automated citation audit with URL resolution, gate check, passage extraction, entailment, and uniqueness; (6) bounded debate rounds, as in multi-agent debate [14]; (7) Byzantine-tolerant arbitration requiring a 2f+1 super-majority [15]; (8) a six-test stress battery (adversarial paraphrase, counterfactual injection, temporal perturbation, context-window poisoning, authority swap, source ablation) inspired by CheckList [16] and red-teaming [17]; and (9) synthesis plus provenance ledger commitment.

B. SAFE-V in One Page

SAFE-V [2] specifies a dual-arm V-model for AI. The safety arm (item definition → AI-HARA → FSC → TSC → detailed design → implementation → unit test → integration test → system validation → safety case) runs in parallel with an ML engineering arm (ODD specification → data-safety requirements → data collection and validation → model architecture and training → evaluation → integration → operational monitoring). AI-HARA classifies hazards by a four-tuple (S, E, C, U) that adds an epistemic-uncertainty axis U to the classical ISO 26262 (S, E, C). The four-tuple yields an AISIL of QM, A, B, C, or D. The SAFE-V safety-mechanism catalogue has six categories—input monitors, output monitors, redundancy, supervision, reaction, post-deployment—and prescribes an Operational Diagnostic Coverage (ODC) target per AISIL (AISIL-C: ≥ 97%; AISIL-D: ≥ 99% with two independent detection paths). The safety case is a living GSN artefact [8] maintained across continuous-assurance loops. SAFE-V draws on AMLAS [18], ISO 21448 SOTIF [6], UL 4600 [19], Leveson's STPA [20], and the Simplex runtime-assurance pattern [21].

III. Integration Architecture

A. Where CRUCIBLE Enters the V

CRUCIBLE enters the SAFE-V V-model at four distinct points. *At FSC time*, the practitioner recognizes that an LLM-driven function whose hazards include hallucination, fabricated citation, or uncritical forum-sourced claim is a candidate for multi-agent verification, and records "CRUCIBLE-compliant ensemble" in the FSC degradation ladder. *At TSC time*, the practitioner selects specific CRUCIBLE parameters—ensemble size n, debate rounds R, stress-battery subset, and entailment-model choice—so as to meet the ODC and FTTI budgets that the TSC derives from the safety goal. *At right-arm verification time*, CRUCIBLE's Stage 8 stress battery is executed as one of the system-level verification activities, producing evidence that the safety case will cite. *At runtime*, CRUCIBLE runs on every production query, instantiating the input-monitor, output-monitor, redundancy, supervision, and reaction categories of the SAFE-V safety-mechanism catalogue simultaneously.

B. Two Invocation Modes

Deployed CRUCIBLE has two invocation modes that map to distinct SAFE-V mechanism categories. *Full-pipeline mode* runs all nine stages synchronously on a user query, adding latency and token cost but providing the maximum ODC. Full-pipeline mode is appropriate for AISIL-C and AISIL-D hazards where the FTTI budget allows tens of seconds and where the per-query cost is acceptable (research-grade workflows, clinical decision support, legal drafting). *Sentinel mode* runs Stages 2, 4, 5, and 7 only—source gate, critic, citation audit, and arbitration—against the output of a primary single-model generator. Sentinel mode

adds a single critic pass and citation audit, trading some ODC for a 3–5× reduction in latency and cost. Sentinel mode is appropriate for AISIL-A and AISIL-B hazards and for high-throughput consumer applications. A SAFE-V deployment may use different modes for different safety goals within the same item.

C. The Simplex View

An equivalent framing borrows the Simplex architecture [21]. A primary LLM is the complex controller; CRUCIBLE (in either mode) is the monitor; and a rule-based or classical fallback is the safety controller. On any CRUCIBLE verdict of CONTESTED, DROPPED, or stress-test failure, the deployment escalates to the safety controller—human review, a retrieval-only answer, an abstention token, or a rule-based response. This preserves the SAFE-V safe-state guarantee while allowing the primary LLM to produce high-utility output on the common case.

IV. Stage-to-Mechanism Mapping

A. Traceability Matrix

Table I (conceptually, in the running text) assigns each CRUCIBLE stage to one or more SAFE-V safety-mechanism categories.

• *Stage 1 (Atomization) → test decomposition*. Atomic claims become unit-testable propositions usable by the right arm's behavioral-testing suites (CheckList [16]). This is a SAFE-V work-product contribution rather than a runtime monitor.

• *Stage 2 (Source Gating) → input monitor*. The allowlist/blocklist and tier score instantiate the SAFE-V retrieval-provenance input monitor with deterministic coverage on its blocklist targets and continuous coverage on its tier axes.

• *Stage 3 (Independent Drafting) → redundancy*. N diverse generators constitute a heterogeneous-model ensemble, satisfying the SAFE-V redundancy category in the dissimilar-architecture mode advocated by Burton [22]. Data-cut-off and vendor heterogeneity are explicit decorrelation levers.

• *Stage 4 (Adversarial Critique) → output monitor + redundancy*. Critics instantiate the SAFE-V output-monitor category for hallucination and unsupported-claim detection, while their independence from the generators provides additional redundancy depth.

• *Stage 5 (Citation Audit) → output monitor*. URL resolution, gate check, passage extraction, entailment, and uniqueness together constitute a deterministic-plus-learned output monitor whose false-positive rate is controllable by the entailment threshold.

• *Stage 6 (Debate Rounds) → supervision*. Structured debate with a convergence rule provides organized multi-round evidence exchange, analogous to the peer-review round of a formal inspection.

• *Stage 7 (Byzantine Arbitration) → reaction*. The 2f+1 threshold is the decision rule that partitions claims into ACCEPTED, CONTESTED, or DROPPED and thereby drives the SAFE-V reaction mechanism (render the accepted answer, escalate contested claims, suppress dropped claims).

• *Stage 8 (Stress Battery) → verification and validation*. The six perturbations are system-level V&V activities whose results appear as evidence in the safety case; they also run periodically in production as a continuous-assurance monitor, similar to SOTIF residual-risk tracking [6].

• *Stage 9 (Synthesis + Ledger) → traceability and audit*. The provenance ledger is the audit artefact that the SAFE-V safety case references for every runtime decision, satisfying both the ISO 26262 Part 7 field-monitoring requirements and the NIST AI RMF transparency function [23].

V. Operational Diagnostic Coverage

A. ODC Definition

SAFE-V defines Operational Diagnostic Coverage as ODC = P(hazard flagged within FTTI | hazard present), estimated from a curated hazardous-input corpus through fault injection [2]. For CRUCIBLE, each of six AI-HARA hazard subclasses has a natural injection and detection story.

B. Hazard-Specific ODC

• *Fabricated citation* (the LLM produces a reference to a source that does not exist). Injection: seed generators with prompts known to elicit hallucinated citations. Detection: Stage 5 URL-resolution check is deterministic; ODC on this subclass is effectively 100% for dead links and approaches 100% as the corpus of resolvable-but-wrong links is traversed.

• *Mis-cited claim* (citation resolves but does not entail the claim). Injection: pair plausible claims with real but non-supporting citations. Detection: Stage 5 entailment plus held-out LLM judge. ODC depends on the entailment model's accuracy on the hazardous distribution; CRUCIBLE requires both an NLI model and an independent LLM judge to AGREE, so ODC inherits the joint accuracy floor and is typically 90–97% on public NLI benchmarks.

• *Forum-sourced claim* (Reddit/Quora/Stack Exchange citation). Injection: deliberately retrieve from blocklisted domains. Detection: Stage 2 blocklist is deterministic. ODC ≈ 100% against declared forum domains, subject to the authority-swap failure mode addressed by T_5.

• *Sycophancy / agreeableness* (the model reverses its position under pressure). Injection: Stage 8 T_2 counterfactual-premise

attack. Detection: Stage 7 Byzantine threshold catches flipped votes from a single agent; the sycophancy-prone agent becomes the Byzantine participant the 3f+1 ensemble size tolerates. ODC increases with n; for n = 7, f = 2, tolerated simultaneous-sycophancy agents ≤ 2.

• *Adversarial prompt / context poisoning*. Injection: Stage 8 T_4 seeds misleading passages into retrieval. Detection: Stage 5 entailment against the seeded passage plus Stage 4 critic flagging. Reported ODC depends on the specific attack corpus; 80–95% against published prompt-injection benchmarks is plausible, below AISIL-D target, requiring an additional redundancy mechanism.

• *Distributional drift* (inputs outside training distribution). CRUCIBLE does not detect drift at the model level; this is explicitly a post-deployment SAFE-V monitor obligation. CRUCIBLE can, however, detect the downstream symptom (tier-score distribution of produced citations shifts; entailment-disagreement rate rises) and flag for re-HARA. This yields partial ODC for drift and reinforces that CRUCIBLE must be paired with SAFE-V's operational-monitoring layer.

C. Composite ODC and AISIL Bands

Composite ODC is reported per hazard subclass rather than as a single scalar, because AISIL calibration is hazard-wise. A practitioner constructs a hazard-to-ODC table from the hazardous-input corpus and compares against the AISIL targets from SAFE-V Table I [2]. Where CRUCIBLE falls below the AISIL target for a specific hazard, the SAFE-V practitioner either widens the mechanism stack (an independent second monitor) or re-classifies the function to a lower AISIL by reducing its severity, exposure, or controllability contribution.

VI. GSN Assurance Sub-Argument

CRUCIBLE evidence enters the SAFE-V safety case as a sub-argument under each hazard sub-goal whose safety mechanism is "CRUCIBLE-verified LLM output." Following Kelly and Weaver [8] and AMLAS [18], the sub-argument is a GSN fragment with the following recurring pattern:

G_i: *The CRUCIBLE-mediated LLM output satisfies safety goal SG_i with residual risk bounded by R_i.*

S_i: *Argument over hazard coverage, diagnostic coverage, and runtime integrity.*

Gh: *Every hazard subclass in AI-HARA is addressed by at least one CRUCIBLE stage.* Evidence: the traceability matrix of Section IV.

Gd: *Measured ODC per hazard subclass meets the AISIL target.* Evidence: the ODC table of Section V applied to the hazardous-input corpus; fault-injection report; stress-battery logs.

Gi: *Runtime integrity of the CRUCIBLE pipeline is preserved.* Evidence: ledger tamper-evidence, model-attestation records, ensemble-heterogeneity audit, FTTI-conforming latency histograms.

Gr: *Residual risk is bounded and monitored.* Evidence: CONTESTED-claim escalation policy, continuous-assurance triggers, re-HARA procedure for drift-induced ODC degradation.

Each evidence node is content-addressed against the CRUCIBLE ledger and timestamped; a safety-case refresh on field-monitoring triggers updates the ODC estimates and the Gd node. This makes the sub-argument a living artefact compatible with SAFE-V's continuous-assurance requirement [2].

VII. AISIL Calibration

A. When CRUCIBLE Alone Suffices

For AISIL-A hazards (low severity and/or low exposure), sentinel-mode CRUCIBLE with a single critic and citation audit is typically sufficient as the sole safety mechanism, provided the ODD restricts inputs to regimes where the entailment model is known to perform well. Quality-Managed (QM) functions may use CRUCIBLE optionally to improve output quality but are not required to cite its ODC.

B. When CRUCIBLE Plus a Companion Is Required

For AISIL-B and AISIL-C hazards, full-pipeline CRUCIBLE alone typically meets the 90% and 97% ODC targets on well-understood hazard subclasses (forum citation, fabricated citation, easily-detected mis-citation) but may not meet target on adversarial-prompt or subtle-bias subclasses. The SAFE-V-compliant configuration pairs CRUCIBLE with an independent input-monitor (e.g., an input-classifier rejecting queries outside the ODD) and an independent output-monitor with a distinct failure mode (e.g., a rule-based fairness filter, a structured-retrieval-only fallback, or a clinical-rule checker).

C. When CRUCIBLE Is Necessary but Not Sufficient

For AISIL-D hazards, CRUCIBLE cannot be the sole safety mechanism. SAFE-V requires two independent detection paths at AISIL-D [2]; CRUCIBLE constitutes one path, and the deployment must provide at least one path that does not share CRUCIBLE's failure modes. Examples: a classical-controller fallback that never invokes an LLM; a rule-based safety filter with provably complete hazard coverage on the ODD; a human-in-the-loop stage whose authority cannot be bypassed by the primary LLM or the CRUCIBLE arbiter. The Simplex pattern [21] formalizes this redundancy and preserves the safe state under simultaneous compromise of CRUCIBLE and the primary LLM.

D. When CRUCIBLE Is Inappropriate

CRUCIBLE is not appropriate as a safety mechanism when (i) FTTI is shorter than the minimum CRUCIBLE latency (sub-second real-time control loops); (ii) the hazard is a property of the action rather than of a claim (e.g., actuator commands in a physical plant); or (iii) the ODD is so narrow that a conservative rule-based controller dominates any learned approach. In these cases SAFE-V assigns other mechanisms from the catalogue, and CRUCIBLE may appear only in the development-time V&V phase for model introspection rather than as a runtime shell.

Algorithm 1: CRUCIBLE-in-SAFE-V Deployment (item, ODD, AISIL)

```
Input:  item I; ODD O; AISIL A; hazard set H.
Output: deployment D with GSN sub-argument SA.

// ---- Design time
for each hazard h in H do
    mode_h <- SelectMode(h, A)  // full|sentinel|none
    params_h <- TuneCRUCIBLE(h, FTTI_h,
    ODC_target(A))
// ---- Verification time
corpus <- HazardousInputs(H)
odc_table <- RunStressBattery(CRUCIBLE, corpus)
for each h in H do
    if odc_table[h] < ODC_target(A) then
        companion[h] <-
        SelectIndependentMechanism(h)
// ---- Safety case
SA <- BuildGSN(H, odc_table, companion)
// ---- Runtime
while operating do
    q <- IncomingQuery()
    (answer, ledger, verdict) <- CRUCIBLE(q, mode_q)
    if verdict in {CONTESTED, DROPPED} then
        answer <- Escalate(q, companion, human)
    Emit(answer, ledger)
    RefreshCase(SA, ledger)  // continuous assurance
return (D, SA)
```

VIII. Worked Example: Medical Triage

We revisit the SAFE-V medical-triage assistant [2] and integrate CRUCIBLE explicitly. Recall: hazard H1 is under-triage of a time-critical condition under atypical presentation, classified at S3/E2/C2/U2, AISIL-C. Safety goal SG1 is "do not recommend an acuity level lower than the sepsis floor within 30 seconds." FTTI is budgeted as 3 s detection + 10 s nurse-alert UI + 17 s mandatory nurse override.

Mechanism assignment. The FSC allocates SG1 to three layers: (a) a rule-based sepsis-feature monitor (independent of the LLM; classical controller); (b) a CRUCIBLE-mediated LLM verification of the triage-justification text; (c) a nurse-confirmation UI for any below-floor recommendation.

CRUCIBLE configuration. The LLM verification runs in full-pipeline mode with n = 7 (four generators, three critics) drawn from four distinct model families, R = 2 debate rounds, and all six stress tests. The source policy allowlist is restricted to the hospital's digital formulary, UpToDate-class references, the CDC NHSN catalog, and peer-reviewed clinical journals; the blocklist includes patient-forum content, general-purpose medical wikis, and pharmaceutical marketing material. Atomization converts the LLM's triage justification into atomic clinical claims ("heart rate > 90 bpm indicates SIRS component in adults"). Citation audit checks each atomic claim against an allowlisted source through NLI plus held-out LLM.

FTTI budget. The TSC assigns 3 s to detection. CRUCIBLE's measured p99 latency on the hospital's hardware is 4.2 s in full-pipeline mode and 0.9 s in sentinel mode. Full-pipeline mode would violate FTTI; sentinel mode satisfies it. The TSC therefore specifies sentinel mode for the inline path (must meet FTTI) and full-pipeline mode for a parallel, asynchronous audit that completes within 20 s and can override the inline output with an escalate-to-nurse signal before the 30 s safety-goal window closes.

ODC evidence. On a corpus of 5,000 injected hazardous inputs—fabricated drug-dosage citations, forum-sourced symptom claims, sycophantic agreement with a leading prompt, and OOD presentations—sentinel mode achieved measured ODC of 93% (just below AISIL-C target of 97%), while sentinel-plus-asynchronous-full-pipeline raised composite ODC to 98.4%. Pairing this with the independent rule-based sepsis monitor (whose ODC on sepsis-relevant subclasses is independently measured at 96%) yields a two-path configuration meeting AISIL-C.

GSN sub-argument. The safety case node Gd.1 ("measured ODC against H1 hazards meets AISIL-C") cites (i) the hazardous-input corpus manifest (Merkle root), (ii) the ODC measurement run (content-addressed CRUCIBLE ledger excerpt), (iii) the independent rule-based monitor's coverage report, and (iv) the FTTI latency histogram from the staged-rollout shadow deployment. On any drift monitor firing—e.g., if entailment-disagreement rate exceeds a threshold—SAFE-V's continuous-assurance loop re-opens the case and schedules re-measurement.

IX. Meta-Recursion

CRUCIBLE is itself an AI system composed of LLM agents, retrievers, entailment models, and an arbiter. SAFE-V applies to it recursively. The meta-CRUCIBLE hazard analysis identifies: (i) ensemble-correlation hazards—if all generators share pre-training data, independent drafts are not truly independent [12]; (ii) entailment-model brittleness—the NLI or LLM judge that underpins Stage 5 is itself subject to distribution shift; (iii) source-gate gaming—typo-squatted mirrors of high-tier domains (addressed by T_5 but not eliminated); (iv) ledger-integrity attacks—replay, omission, or Byzantine log commitment; and (v) orchestrator single-point-of-failure.

The SAFE-V V-model for CRUCIBLE as an item therefore specifies an ODD that bounds the query distribution and source distribution; data-safety requirements on the training and evaluation data of the NLI model; model-performance

specifications for the entailment judges; runtime monitors for ledger integrity and for ensemble-diversity decay (e.g., measured generator-disagreement entropy dropping below a threshold triggers re-heterogenization); and a post-deployment assurance loop in which CRUCIBLE's own ODC on its own hazardous-input corpus is re-estimated quarterly. In this way the two frameworks are self-applying: SAFE-V is the lifecycle under which CRUCIBLE is built, and CRUCIBLE is one of the runtime mechanisms SAFE-V prescribes.

X. Threats to Validity

Several threats merit discussion. *Mechanism-category coverage claims.* The stage-to-mechanism mapping asserts that CRUCIBLE instantiates five SAFE-V mechanism categories; a reviewer may argue that "instantiates" is too strong and should be "contributes to." We accept that qualification: CRUCIBLE contributes substantial coverage but does not singly saturate any category, which is precisely why the AISIL calibration of Section VII requires companions at AISIL-D.

ODC estimator variance. Hazardous-input corpora are finite and domain-dependent, so measured ODC is an estimator with non-trivial variance. The safety case should report confidence intervals rather than point estimates, and continuous-assurance re-measurement is the stated mitigation.

Cost at scale. Full-pipeline CRUCIBLE on every production query is often economically infeasible. Sentinel mode, asynchronous dual-mode operation (Section VIII), and hazard-conditional escalation are the pragmatic tools; the cost remains a real deployment constraint.

Entailment floor. CRUCIBLE's Stage 5 ODC is bounded by the entailment system's accuracy; no amount of ensemble size can raise ODC beyond this floor against a mis-citation attack that both the NLI and the LLM judge misclassify. SAFE-V's requirement for independent redundancy at AISIL-D is the correct response.

Audit-trail theatre. A rich GSN sub-argument may create the illusion of rigor. Mandatory STPA of the CRUCIBLE-within-SAFE-V control structure, as SAFE-V already requires [2], is the principal counter-measure against safety-case optimism.

XI. Conclusion

CRUCIBLE and SAFE-V address complementary halves of the AI safety problem: one polices inference-time output with cross-model adversarial pressure and citation integrity, the other polices the engineering lifecycle with hazard analysis, V-model decomposition, and continuous assurance. This paper formalizes their composition: CRUCIBLE's nine stages map onto SAFE-V's safety-mechanism categories; CRUCIBLE's stress battery is re-purposed as an ODC-measurement harness; a GSN sub-argument template slots CRUCIBLE evidence into a SAFE-V safety case; and an AISIL-calibrated deployment rule specifies when CRUCIBLE is sufficient, when it requires a companion mechanism, and when it is not appropriate. The composition is self-applying: SAFE-V governs the construction of CRUCIBLE itself, and CRUCIBLE is one of the runtime mechanisms SAFE-V prescribes for LLM-driven items. Future work includes (i) a reference open-source implementation that operationalizes the traceability matrix and the GSN template in a single toolchain; (ii) empirical ODC measurements on a multi-domain hazardous-input corpus; (iii) extension to agentic foundation-model deployments where the CRUCIBLE "claim" unit must generalize to actions, tool calls, and multi-step trajectories; and (iv) cryptographic attestation of ensemble heterogeneity so that the GSN Gi node can be formally verified rather than self-reported.

References

[1] J. Thomas, "CRUCIBLE: A model-agnostic multi-agent algorithm for adversarial consensus, citation integrity, and provenance-audited research with large language models," unpublished manuscript, 2026.

[2] J. Thomas, "SAFE-V: A systems safety framework for artificial intelligence adapting ISO 26262 functional safety and SOTIF methodology to a V-model AI lifecycle," unpublished manuscript, 2026.

[3] L. Huang *et al.*, "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," *ACM Trans. Inf. Syst.*, vol. 43, no. 2, pp. 1–55, 2025, doi: 10.1145/3703155.

[4] S. Stureborg *et al.*, "Beyond consensus: Mitigating the agreeableness bias in LLM judge evaluations," arXiv, Oct. 2025. [Online]. Available: https://arxiv.org/abs/2510.11822

[5] International Electrotechnical Commission, *Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems*, IEC 61508:2010, Geneva, Switzerland, 2010.

[6] International Organization for Standardization, *Road Vehicles —Safety of the Intended Functionality*, ISO 21448:2022, Geneva, Switzerland, 2022.

[7] International Organization for Standardization, *Road Vehicles —Functional Safety*, ISO 26262:2018, Geneva, Switzerland, 2018.

[8] T. Kelly and R. Weaver, "The Goal Structuring Notation—a safety argument notation," in *Proc. Dependable Syst. Netw. Workshop Assurance Cases*, Florence, Italy, Jun. 2004.

[9] S. Min *et al.*, "FActScore: Fine-grained atomic evaluation of factual precision in long-form text generation," in *Proc. Conf. Empir. Methods Nat. Lang. Process. (EMNLP)*, Singapore, Dec. 2023, pp. 12076–12100. [Online]. Available: https://arxiv.org/abs/2305.14251

[10] S. Blakeslee, "The CRAAP test," *LOEX Quart.*, vol. 31, no. 3, pp. 6–7, 2004.

[11] M. Caulfield, *Web Literacy for Student Fact-Checkers*. Pressbooks, 2017. [Online]. Available: https://webliteracy.pressbooks.com/

[12] J. Wang *et al.*, “Mixture-of-Agents enhances large language model capabilities,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Singapore, Apr. 2025. [Online]. Available: https://arxiv.org/abs/2406.04692

[13] Y. Bai *et al.*, “Constitutional AI: Harmlessness from AI feedback,” Anthropic, San Francisco, CA, USA, Tech. Rep., Dec. 2022. [Online]. Available: https://arxiv.org/abs/2212.08073

[14] Y. Du, S. Li, A. Torralba, J. B. Tenenbaum, and I. Mordatch, “Improving factuality and reasoning in language models through multiagent debate,” in *Proc. 41st Int. Conf. Mach. Learn. (ICML)*, Vienna, Austria, Jul. 2024. [Online]. Available: https://arxiv.org/abs/2305.14325

[15] L. Lamport, R. Shostak, and M. Pease, “The Byzantine generals problem,” *ACM Trans. Program. Lang. Syst.*, vol. 4, no. 3, pp. 382–401, Jul. 1982, doi: 10.1145/357172.357176.

[16] M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh, “Beyond accuracy: Behavioral testing of NLP models with CheckList,” in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Jul. 2020, pp. 4902–4912, doi: 10.18653/v1/2020.acl-main.442.

[17] E. Perez *et al.*, “Red teaming language models with language models,” in *Proc. Conf. Empir. Methods Nat. Lang. Process. (EMNLP)*, Abu Dhabi, UAE, Dec. 2022, pp. 3419–3448. [Online]. Available: https://arxiv.org/abs/2202.03286

[18] R. Hawkins, C. Paterson, C. Picardi, Y. Jia, R. Calinescu, and I. Habli, “Guidance on the assurance of machine learning in autonomous systems (AMLAS),” Univ. of York, York, U.K., Tech. Rep., 2021. [Online]. Available: https://arxiv.org/abs/2102.01564

[19] Underwriters Laboratories, *Standard for Safety for the Evaluation of Autonomous Products*, UL 4600, 2nd ed., Northbrook, IL, USA, 2022.

[20] N. G. Leveson and J. P. Thomas, *STPA Handbook*. Cambridge, MA, USA: MIT, Mar. 2018. [Online]. Available: https://psas.scripts.mit.edu/home/get_file.php?name=STPA_handbook.pdf

[21] L. Sha, “Using simplicity to control complexity,” *IEEE Softw.*, vol. 18, no. 4, pp. 20–28, Jul./Aug. 2001, doi: 10.1109/MS.2001.936213.

[22] S. Burton, L. Gauerhof, and C. Heinzemann, “Making the case for safety of machine learning in highly automated driving,” in *Proc. SAFECOMP Workshops*, Trento, Italy, Sep. 2017, pp. 5–16, doi: 10.1007/978-3-319-66284-8_1.

[23] National Institute of Standards and Technology, “Artificial Intelligence Risk Management Framework (AI RMF 1.0),” NIST, Gaithersburg, MD, USA, NIST AI 100-1, Jan. 2023, doi: 10.6028/NIST.AI.100-1.