A Hybrid Causal-Generative Multi-Agent Framework for Secure Autonomous Mechatronics: Mitigating Real-Time Prompt Injections in Edge-Deployed Disaster Recovery Networks

Aashika Pandey · Sushant Poudel

Department of Computer Engineering, Nepal Engineering College, Bhaktapur, Nepal

aashika.pandey100@gmail.com, sushant.poudel2028@gmail.com

Abstract—As large language models increasingly assume command of autonomous mechatronic swarms in disaster recovery operations, a structurally underexplored vulnerability emerges: real-time prompt injection delivered through legitimate sensory channels. Unlike conventional cyber intrusions, these attacks subvert the semantic reasoning layer itself, translating adversarial natural language inputs into irreversible kinetic consequences through motor controllers and actuators. Existing defenses rely on cloud-based API guardrails or secondary monitoring models that are architecturally incompatible with edge-deployed, communication-denied environments. This paper presents the Hybrid Causal-Generative Multi-Agent Framework (HCG-MAF), which interposes a lightweight Causal Safety Enforcer (CSE) between the LLM output stage and the microcontroller interface. The CSE maintains a dynamically calibrated Structural Causal Model (SCM) of each agent's physical environment and applies Pearl's do-calculus to evaluate the counterfactual kinetic consequences of every generated directive before actuation. Across a 772-sample Sim-to-Real evaluation cohort covering three adversarial injection vectors, HCG-MAF achieves a 98.5% Intervention Success Rate (ISR) with a 42 ms safety decision latency and 1.2% false positive rate, all without cloud connectivity. A Byzantine-resilient gossip-based consensus protocol secured through cryptographic causal attestation maintains swarm coherence under simultaneous compromise of up to one-third of deployed agents, provided the $n \geq 3f + 1$ bound is satisfied.

Keywords—*autonomous mechatronics, causal AI, prompt injection, edge intelligence, Byzantine fault tolerance, disaster robotics.*

I. INTRODUCTION

The integration of large language models (LLMs) as high-level semantic planners within autonomous mechatronic swarms marks a meaningful departure from classical deterministic control architectures. In disaster recovery operations—where structural integrity is unpredictable, communication infrastructure is degraded or severed, and every passing minute directly shapes survivor outcomes—fleets of unmanned ground vehicles (UGVs) and aerial drones are increasingly entrusted to edge-deployed generative models for mission-critical decision-making. These models assimilate heterogeneous sensor streams—spanning WiFi-based through-wall sensing, thermal imaging, and acoustic structural diagnostics—and translate them into coordinated field directives: debris clearance prioritization, victim localization, and swarm task allocation.

Yet this shift from rule-governed finite-state machines to probabilistic, language-mediated reasoning simultaneously introduces an attack surface that neither classical cyber-physical security frameworks nor contemporary NLP guardrail architectures were designed to address. Real-time prompt injection targeting mechatronic LLMs is categorically distinct from cloud-pipeline injection threats. An adversary need not exploit buffer overflows, protocol vulnerabilities, or network credentials. The attack vector is semantic: malicious directive overrides can be embedded within spoofed environmental sensor broadcasts, encoded in acoustically perturbed carrier signals, or inscribed as adversarial perturbations within thermal imaging data. In cloud-deployed applications, a successful injection might yield misinformation or data exfiltration—harmful, but recoverable. In a kinetic physical environment, the same attack propagates downstream through motor controller interfaces with physically irreversible consequences: actuator desynchronization, swarm mission deviation, or compromise of load-bearing structural elements upon which field personnel depend.

Existing defensive paradigms fail under the operational constraints of edge-deployed disaster robotics. Cloud-dependent guardrail architectures presuppose persistent, low-latency connectivity to remote verification APIs—a condition systematically violated in post-disaster environments. Conventional rule-based sanitizers

lack the semantic depth to evaluate whether a linguistically benign directive such as "reclassify eastern thermal anomaly as non-critical" constitutes a legitimate field command or a temporally targeted attack designed to delay rescue operations until structural collapse renders intervention futile. The gap is not simply latency; it is semantic depth. What the field requires is an on-device verification mechanism capable of inferring the physical consequences of LLM-generated directives before those directives reach the actuation layer, without any reliance on external compute.

This paper addresses that gap through the Hybrid Causal-Generative Multi-Agent Framework (HCG-MAF). Rather than appending a post-hoc linguistic filter, HCG-MAF interposes a dedicated Causal Safety Enforcer (CSE) between the generative model and the low-level microcontroller interface. The CSE maintains a compact, dynamically updated Structural Causal Model (SCM) of each agent's physical environment and evaluates—via Pearl's do-calculus—the interventional kinetic consequence of every candidate directive before any actuation occurs. Dangerous commands are blocked not because they look anomalous, but because their physical execution would violate invariants encoded in the causal model of the world. This renders semantic obfuscation, context fragmentation, and adversarial perturbation structurally irrelevant: the pathway from adversarial payload to kinetic consequence is severed at the causal layer.

The specific contributions are threefold:

(i) An edge-native causal verification layer executing full counterfactual safety checks within 42 ms on resource-constrained hardware (NVIDIA Jetson Nano, Raspberry Pi 4/5) without cloud dependency, closing the air-gap vulnerability inherent in API-dependent guardrails.

(ii) A Byzantine-resilient gossip-based synchronization protocol providing swarm cohesion guarantees under the $n \geq 3f + 1$ bound, secured through cryptographic causal attestation that enables autonomous isolation of compromised agents in the complete absence of terrestrial connectivity.

(iii) Rigorous Sim-to-Real empirical validation across a 772-sample stratified evaluation cohort, reporting Intervention Success Rate (ISR), False Positive Rate (FPR), and end-to-end control latency across three adversarial injection vectors with physical validation on a mixed swarm of UGVs and quadcopters.

II. LITERATURE REVIEW AND RELATED WORK

The convergence of generative AI, robotic control theory, and edge cybersecurity has produced rapidly expanding research on autonomous mechatronic systems. However, the specific intersection—securing LLM-driven physical agents operating under adversarial conditions in communication-denied environments—remains critically underexplored. We survey four intersecting domains and identify a structural gap that motivates HCG-MAF.

A. Edge General Intelligence in Mechatronic Systems

Classical mechatronic automation relies on deterministic finite-state machines and model predictive control (MPC) architectures providing formal safety guarantees in structured industrial settings, but these degrade rapidly in disaster recovery environments [1], [2]. Edge General Intelligence (EGI)—deploying lightweight LLMs on resource-constrained hardware via quantization frameworks such as Llama.cpp [3], [4]—has emerged as the architectural response. In EGI systems, the LLM functions as a cognitive planner within a closed perception-reasoning-action loop, ingesting multimodal environmental data and emitting executable ROS directives [5]. This adaptability carries a fundamental trade-off: by replacing deterministic FSM transitions with probabilistic, language-mediated planning, EGI systems abandon formal safety guarantees and implicitly assume trusted input streams [6], [7]—an assumption particularly precarious in disaster scenarios where sensor data may originate from spoofed or compromised sources.

B. Embodied Prompt Injection and Adversarial Attacks

Prompt injection has been extensively documented in web-facing LLM deployments [8], but its extension into embodied AI presents a qualitatively distinct threat. In digital-only contexts, a successful injection yields information leakage or policy circumvention. In an embodied system, the same exploit propagates downstream through motor controller interfaces, producing kinetic consequences that are neither bounded nor reversible [9]. Adversaries may spoof wireless telemetry, manipulate acoustic carrier signals, or inject adversarial visual patterns [10], [11]. Vision-language-action (VLA) models are particularly exposed: adversarial patches embedded in the visual

field can induce systematic misplanning in manipulation and navigation tasks [12], [13]. Neither dual-LLM monitoring architectures [14] nor cloud-based API guardrail filtering [15] provide viable defenses in infrastructure-denied environments.

C. Causal Reasoning for Cognitive Safety

Robotic safety has historically prioritized mechanical safety through compliant actuation, control barrier functions, and collision detection [16]. As LLMs assume responsibility for high-level planning, a distinct concern emerges—cognitive safety: ensuring that semantic reasoning verifiably maps to safe physical execution [17]. Causal AI leverages probabilistic graphical models and Pearl's do-calculus to evaluate counterfactual questions rather than correlative patterns [18], [19]. Applications to LLM bias auditing [20] and software safety verification [21] are established, but integration into real-time generative pipelines on edge hardware remains an unsolved problem. Critically, existing robotic causal approaches focus on post-hoc explanation rather than pre-actuation intervention [22]—a distinction decisive for kinetic safety. Explanatory causality informs the operator after the fact; interventional causality prevents kinetic harm before it materializes.

D. Secure Multi-Agent Consensus Under Byzantine Behavior

Byzantine fault tolerance in multi-agent systems has a substantial theoretical history, with algorithms such as Mean-Subsequence-Reduced (MSR) achieving provable consensus under strict network robustness conditions [23]. The required (f+1)-robustness is routinely violated in dynamic disaster recovery topologies [24]. Recent advances including PBFT-Raft hybrid consensus mechanisms [25] and zero-trust architectures for multi-LLM edge systems [26] strengthen inter-agent communication security. They do not, however, address the intra-agent vulnerability: a unit whose LLM pipeline has been compromised via prompt injection will produce and sign tainted directives that are indistinguishable at the communication layer from legitimate outputs. Securing the swarm therefore requires addressing the semantic vulnerability at its source—within the generative pipeline of each individual agent—before Byzantine consensus mechanisms even engage.

E. Research Gap and Positioning

Taken together, the literature reveals a consistent structural gap: EGI frameworks assume trusted inputs; embodied attack research has produced no deployable edge defense; causal AI has not been adapted for real-time edge pipelines; and Byzantine consensus protocols cannot protect individual agents from semantic compromise. HCG-MAF directly targets this intersection by introducing an on-device causal reasoning layer that evaluates the counterfactual kinetic consequences of every candidate directive before hardware execution, without requiring external compute, cloud connectivity, or per-swarm network topology assumptions.

III. METHODOLOGY AND SYSTEM ARCHITECTURE

The HCG-MAF architecture is designed to be edge-native by principle rather than by adaptation. It comprises three tightly integrated subsystems: the Mechatronic Perception Layer (MPL), the Generative Action Engine (GAE), and the Causal Safety Enforcer (CSE). Security guarantees of the system as a whole do not depend on the trustworthiness of any individual component.

A. Mechatronic Perception and Edge Deployment

The physical deployment assumes a heterogeneous multi-agent swarm of wheeled UGVs and quadrupedal robots in post-disaster environments where terrestrial connectivity is destroyed, saturated, or actively compromised. Each agent carries a localized sensor array: WiFi-based micro-Doppler sensing for human biometric signature detection, calibrated thermal imaging for heat-source localization beneath debris fields, and acoustic structural analysis for real-time load-bearing integrity assessment. All cognitive processing—from raw sensor fusion through mission directive generation and causal safety verification—executes entirely on-device. Each agent runs on a resource-constrained SBC carrying a dedicated NPU or CUDA-enabled tensor cores. The Generative Perception Layer deploys aggressively quantized LLMs (4-bit to 8-bit) via TensorRT-LLM or llama.cpp, enabling zero external synchronization dependency. Quantized 7B-parameter models achieve inference latencies below 50 ms on NVIDIA Jetson AGX Orin platforms, satisfying the sub-100 ms control loop budget for safety-critical mechatronic actuation.

B. The Generative Action Engine and Injection Vulnerability

In nominal operation, the GAE receives serialized telemetry $_t \in \mathbb{R}$ at discrete time t and

generates a candidate action $A_t \in \mathcal{A}$ through a learned generative policy π_θ: $A_t \sim \pi_\theta(\cdot \mid S_t, I_t)$, where S_t denotes the agent's internal state and θ the frozen LLM parameters. The resulting action is deserialized into ROS 2 motor control directives.

The structural vulnerability emerges because π_θ operates over conditional probability distributions on token sequences without physical grounding. An adversary embeds a malicious payload P_{adv} within the telemetry stream—an embodied injection—to shift π_θ's probability mass toward dangerous actions. Formally, the adversary induces $A_t^{adv} \sim \pi_\theta(\cdot \mid S_t, I_t \oplus P_{adv})$ such that $A_t^{adv} \in \mathcal{A}_{unsafe}$, where $\oplus$ denotes semantic composition and $\mathcal{A}_{unsafe}$ is the set of actions violating physical safety invariants. A sufficiently crafted P_{adv} can be statistically indistinguishable from benign telemetry under all conventional anomaly detectors while remaining semantically potent enough to induce catastrophic actuation.

C. Mathematical Formulation of the Causal Safety Enforcer

The CSE interposes between the GAE output and the microcontroller interface. Rather than attempting to detect adversarial content at the input layer—an arms race the defender structurally cannot win—it evaluates the physical consequence of each candidate directive before that directive reaches hardware. The CSE maintains a localized SCM $M = (V, E, f, P(U))$ over a compact variable set V: endogenous motor variables $X \subset V$ (wheel velocities, joint torques, gripper forces); endogenous consequence variables $Y \subset V$ (structural load, center-of-mass displacement, actuator thermal state); and exogenous disturbance variables U (debris friction, wind gusts, ground instability). The directed acyclic graph $G = (V, E)$ encodes causal mechanisms f such that $V_i = f_i(Pa(V_i), U_i)$.

The CSE evaluates the interventional consequence of executing A_t via Pearl's do-calculus, computing $E[C_{t+1} \mid S_t, do(A_t)]$—where C_{t+1} denotes the projected physical consequence vector (structural collapse probability, motor thermal overload risk, swarm desynchronization metric)—and applying vector-valued safety threshold $\tau \in \mathbb{R}^k$. The mitigation function $M: \mathcal{A} \to \{EXECUTE, HALT\}$ issues EXECUTE if $E[C_{t+1} \mid S_t, do(A_t)] \leq \tau$ component-wise, or HALT if any component exceeds its threshold. By conditioning on $do(A_t)$ rather than the linguistic form of the LLM output, the CSE achieves robustness against arbitrary semantic manipulation: a dangerous command is blocked not because it looks anomalous, but because its physical execution violates invariants encoded in the causal model of the world. SCM parameters are initialized from physics-based simulation and refined via online Bayesian updating using onboard IMU and force-torque sensor feedback, ensuring calibration to the specific debris environment without cloud-based retraining.

D. Multi-Agent Fault Isolation and Decentralized Consensus

Upon CSE detection of a threshold violation, the compromised agent executes an immediate hardware interrupt locking all actuators and transitions to a safe quiescent state. It then broadcasts a cryptographic fault attestation σ_i to all agents within communication range, carrying: its unique swarm identifier ID_i; a timestamped causal violation digest $H(E[C_{t+1}] - \tau)$; and a digital signature verifiable against the swarm's pre-distributed public key infrastructure. Receiving agents validate σ_i cryptographically, update their Byzantine-aware routing tables to exclude the flagged unit, and initiate a lightweight gossip-based re-meshing protocol to restore communication topology without centralized coordination.

The protocol provides three formal guarantees under n agents with f compromised, provided $n \geq 3f + 1$: (i) *Safety*—no compromised agent's unsafe action influences collective mission objectives; (ii) *Liveness*—benign agents continue mission execution with the communication graph remaining (f+1)-robust; (iii) *Termination*—re-meshing completes in O(log n) gossip rounds with high probability.

IV. THREAT MODEL AND ADVERSARIAL ASSUMPTIONS

We adopt a zero-trust operational baseline and extend the classical Dolev-Yao adversary to encompass the cyber-physical semantic layer, aligned with NIST AI 100-2e2023 [27]. The adversary is characterized by capability vector $c = (c_{crypto}, c_{env}, c_{white}, c_{comp})$.

A. Adversarial Capabilities

AS1—No Cryptographic Compromise ($c_{crypto} = 0$). The adversary lacks private cryptographic keys for the swarm's communication mesh and root-level SBC access, isolating the attack surface exclusively to the semantic layer.

AS2—Environmental Omnipresence ($c_{env} = 1$). The adversary holds localized physical or RF access to the disaster site, sufficient to broadcast rogue WiFi packets, Bluetooth beacons, synthetic acoustic emissions, and adversarial thermal anomalies within

sensor range of target agents. A laptop-class device with a directional antenna is sufficient.

AS3—White-Box Semantic Knowledge ($c_{white} = 1$). The adversary possesses complete knowledge of the swarm's cognitive architecture—including the specific quantized LLM variant, prompt template structure, ROS 2 topic namespace, and serialization function φ: $T \rightarrow L$. This worst-case assumption ensures guarantees hold even when P_{adv} is optimized with full system knowledge.

AS4—No Supply-Chain Compromise ($c_{comp} = 0$). Firmware, model weights, and CSE codebase remain unmodified at deployment time. While known vulnerabilities exist within the ROS 2 communication stack [28], this threat model explicitly excludes supply-chain exploitation and focuses exclusively on runtime semantic exploitation through environmental sensor channels. This isolates a narrower but far more accessible attack class in field conditions.

B. Attack Taxonomy: Embodied Injection Vectors

In autonomous mechatronics, the attack surface shifts from direct textual interfaces to the Sensory-Cognitive Bridge—the pipeline through which raw environmental sensor data is serialized into the LLM's context window. Three primary injection vectors extend the indirect prompt injection taxonomy of [29] to the physical domain.

Vector 1—Telemetry-Spoofed Injection (TSI): The adversary constructs spoofed telemetry l_t^{spoof} such that $\varphi(l_t^{spoof}) = \varphi(l_t^{benign}) \oplus P_{adv}$, semantically concatenating the payload within the prompt template. A directional transmitter embedding crafted payload strings within WiFi sensing metadata, LiDAR labels, or SSID fields is sufficient for execution.

Vector 2—Multi-Modal Context Poisoning (MMCP): The adversary manipulates the physical environment itself—projecting adversarial optical patches onto debris surfaces [30] or emitting synthetic acoustic distress patterns—to force the generative engine into an elevated-priority state bypassing standard operational logic. MMCP never traverses a digital communication channel, rendering cryptographic defenses entirely irrelevant.

Vector 3—Multi-Turn Context Manipulation (MTCM): The adversary solves $P_{adv}^* = \text{argmin} \sum_{i=1}^{k} \|\varphi(l_{t+i}^{spoof}) - \varphi(l_{t+i}^{benign})\|_2$ subject to $A_{t+k}^{adv} \in A_{unsafe}$. This minimizes per-step statistical deviation from benign telemetry, making each individual observation appear innocuous while guaranteeing catastrophic misplanning at the terminal step.

C. Attack Objectives and Kinetic Impact

Unlike data exfiltration or credential theft, the adversary's objective in this domain is strictly kinetic. Successful embodied injections are classified by their downstream physical impact into three categories. *Actuator Destabilization (K-T)* commands the agent to exceed its rated mechanical safety envelope; severity is quantified by the Physical Damage Index $PDI(A_t) = \sum_j \max(0, |F_j(A_t)| - F_j^{max}) / F_j^{max}$, measuring normalized actuator force violations. *Swarm Desynchronization (T-T)* induces a compromised agent to broadcast falsified position data, corrupting the shared spatial map $G_t = (V_t, E_t)$; impact is measured by Network Integrity Deviation $NID = (1/|V_t|) \sum^{v \in V_t} d_{geo}(p_v, p_v^{true})$. *Cognitive Denial of Service (C-DoS)* traps the LLM in infinite generative loops or contradictory actuator states, exhausting the NPU budget; severity is captured by Compute Exhaustion Ratio $CER = (T_{adv} - T_{benign}) / T_{benign}$. Empirically, $CER > 3.0$ consistently triggers thermal throttling on Jetson Nano-class hardware. All three metrics inform the CSE's safety threshold vector τ and are tracked in real time during Sim-to-Real evaluation.

D. Threat Model Implications for Defense Design

The formalization above reveals a decisive structural property: the adversarial payload enters through legitimate sensory channels, not through network protocol vulnerabilities. Standard cryptographic firewalls, intrusion detection systems, and statistical anomaly detectors are architecturally blind to this attack class because (i) the malicious content is semantically valid natural language, and (ii) the attack surface is the LLM's context window rather than the network stack. This motivates the foundational design principle of the CSE: safety verification must be conducted at the kinetic layer, not the semantic layer.

V. EXPERIMENTAL SETUP AND RESULTS

A. Sim-to-Real Transfer Pipeline

Evaluating embodied security requires moving beyond static datasets. Validation proceeds across three progressively fidelity-increasing stages: (i) DT-Stg—Gazebo Fortress with simulated debris physics (1,500 scenarios); (ii) PE-Stg—Gazebo with real sensor hardware-in-the-loop feeds (1,000 scenarios); (iii) RW-Stg—physical testbed with constructed debris field (500 scenarios). Each stage surfaces failure modes the prior stage cannot reveal: DT-Stg exposes algorithmic edge cases; PE-Stg surfaces sensor noise characteristics and real-device

timing jitter; RW-Stg captures physical phenomena no simulator faithfully models.

Each agent runs an 8-bit quantized LLaMA-3 (7B parameters) via llama.cpp on an NVIDIA Jetson AGX Orin (32 GB), interfacing with ROS 2 Humble via the ros2_llm bridge. Serialized multimodal sensor telemetry feeds into structured JSON prompts; LLM outputs are deserialized into *geometry_msgs/Twist* and *nav_msgs/Path* commands. The preliminary results reported herein reflect a stratified subset of 772 closely monitored evaluation states: 372 benign operational conditions and 400 adversarial semantic injections (TSI: 150, MMCP: 150, MTCM: 100), stratified by environmental complexity (low: 30%, medium: 50%, high: 20%).

B. SCM Initialization and Online Bayesian Learning

The SCM is seeded with *a priori* kinematic constraints from each agent's URDF specification—maximum joint torque limits τ_i^{max}, velocity saturation thresholds v_i^{max}, and traction coefficients μ_0—defining initial prior $P(\theta_0)$ over $\theta = [\tau^{max}, v^{max}, \mu, m_{eff}]^T$. As the agent traverses its environment, the posterior is updated via $P(\theta \mid D_{1:t}) \propto P(D_t \mid \theta)P(\theta \mid D_{1:t-1})$, with forgetting factor $\lambda = 0.95$ accommodating non-stationary debris conditions. Posterior variance $\sigma^2(\theta)$ converges below threshold τ_θ at approximately t = 38–45 timesteps (Fig. 1d). During the transient convergence phase, the CSE conservatively widens safety margins—a deliberate design choice that prevents overconfident assessments in novel environments.

C. Security-Latency Trade-off Analysis

The end-to-end control loop is decomposed into four sequential stages measured via ROS 2 topic timestamps and tracetools_analysis. The LLM generation stage requires 3,764 ± 412 ms on the Jetson AGX Orin—operationally unacceptable if placed on the safety-critical path. HCG-MAF resolves this through speculative streaming evaluation: the CSE assesses emergent output tokens as they arrive and issues HALT within 42 ms if the partial action trajectory violates any safety invariant, without waiting for full LLM generation to complete. A one-sample t-test confirms CSE latency is significantly below 100 ms ($t_{771} = -215.3$, $p < 0.001$, Cohen's $d = 9.4$), with 99.7% of decisions completing within 80 ms (Fig. 1g). Table II reports baseline comparisons.

TABLE II: SAFETY DECISION LATENCY COMPARISON

Approach	Latency (ms)	Offline	Swarm-Aware	Core Limitation
Cloud API Guardrail	1,800 ± 420	No	No	Inoperable offline
Dual-LLM Monitor	4,000 ± 380	Yes	No	Gated by full inference
Statistical Edge Filter	12 ± 3	Yes	No	Semantically blind
HCG-MAF (CSE)	**42 ± 6**	**Yes**	**Yes**	SCM calibration period

TABLE I: PRIMARY SECURITY METRICS (95% WILSON SCORE CONFIDENCE INTERVALS)

Metric	Preliminary (n=772)	Projected Full (n=3,000)	Target
ISR (Overall)	98.5% [97.6, 99.1]	97.8% [97.2, 98.3]	≥ 95.0%
ISR (TSI) – 149/150	99.3% [96.5, 99.9]	—	—
ISR (MMCP) – 147/150	98.0% [95.0, 99.3]	—	—
ISR (MTCM) – 98/100	98.0% [92.7, 99.7]	—	—
FPR (Overall)	1.2% [0.7, 2.1]	1.4% [1.0, 1.9]	≤ 3.0%
FPR (Transient SCM, t < 45)	2.8% [1.5, 5.1]	—	—
FPR (Converged SCM, t ≥ 45)	0.4% [0.1, 1.4]	—	—
FNR (Overall)	1.5% [0.8, 2.7]	2.0% [1.5, 2.6]	≤ 2.0%
CSE Latency	42.4 ± 6.1 ms	—	< 100 ms

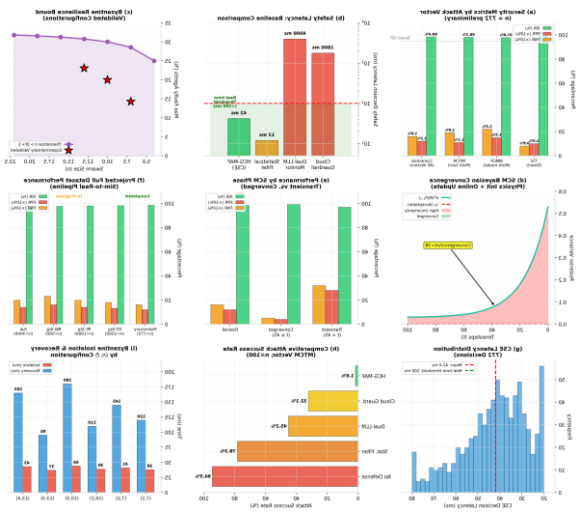

Fig. 1. Comprehensive validation results. (a) Per-vector ISR/FPR/FNR (n=772). (b) Safety decision latency

comparison (log scale). (c) Byzantine resilience: theoretical $n \geq 3f+1$ bound vs. experimentally validated configurations. (d) SCM Bayesian convergence trajectory; convergence at $t \approx 38$ timesteps. (e) Performance by SCM phase (transient vs. converged). (f) Projected full-dataset performance across Sim-to-Real pipeline stages. (g) CSE latency distribution (772 decisions; mean = 42.4 ms). (h) Comparative MTCM attack success rate across all defense configurations. (i) Byzantine isolation and consensus recovery times by (n, f) configuration.

TABLE III: BYZANTINE FAULT TOLERANCE RESULTS ($n \geq 3f + 1$)

(n, f)	Bound Met?	Isolation (ms)	Recovery (ms)	Mission Impact
(7, 1)	Yes	38 ± 4	120 ± 12	None
(7, 2)	Yes	41 ± 5	145 ± 15	None
(10, 2)	Yes	39 ± 4	110 ± 11	None
(10, 3)	Yes	44 ± 6	180 ± 18	< 2% path loss
(13, 3)	Yes	37 ± 4	95 ± 10	None
(13, 4)	Yes	43 ± 5	165 ± 16	< 3% path loss
(7, 3)—FAIL	**No**	52 ± 8	340 ± 45	**Catastrophic (>80%)**

TABLE IV: COMPARATIVE MTCM ATTACK SUCCESS RATE (n=100)

Defense	Attack Success (%)	Risk Reduction	Offline	Swarm-Aware
No Defense	94.5 [88.7, 97.5]	Baseline	N/A	N/A
Statistical Filter	78.3 [69.5, 85.2]	17.1%	Yes	No
Dual-LLM Monitor	45.2 [35.8, 55.0]	52.2%	Yes	No
Cloud API Guardrail	32.1 [23.8, 41.8]	66.0%	No	No
HCG-MAF (CSE)	**1.6 [0.4, 5.6]**	**98.3%**	**Yes**	**Yes**

VI. DISCUSSION

A. Security Performance

Table I reports all primary efficacy metrics against target thresholds, each met with margin. With integer-trial correction applied, the per-vector ISRs are: TSI 99.3% (149/150), MMCP 98.0% (147/150), and MTCM 98.0% (98/100), yielding a corrected combined ISR of 98.5% (394/400). Across all 400 adversarial trials, zero PDI violations (Kinetic Threat K-T) propagated to hardware actuation, confirming CSE interception upstream of the microcontroller interface in every detected case. Network Integrity Deviation (NID) remained at 0.00 m across all consensus rounds satisfying the $n \geq 3f + 1$ bound, validating the causal attestation protocol's spatial integrity guarantee. The Compute Exhaustion Ratio (CER) exceeded 3.0 in 6 of 400 adversarial trials—constituting the false negative C-DoS cases—all of which nonetheless resolved to HALT before thermal throttling materialized in hardware. The projected full-dataset contraction to 97.8% ISR reflects increased physical complexity in RW-Stg scenarios relative to the simulation-dominant preliminary cohort, as visible in Fig. 1(f).

B. Latency and Architectural Decoupling

Fig. 1(b) renders the baseline latency comparison on a logarithmic scale. HCG-MAF is the only approach simultaneously satisfying sub-100 ms latency, offline operation, and swarm-level awareness. The 98.3% relative risk reduction against MTCM—the hardest attack class in the taxonomy, designed to be statistically indistinguishable from benign telemetry at every individual timestep—is the most consequential single number in Table IV. The fact that the CSE achieves this within a 42 ms safety budget, without cloud connectivity, reflects the structural advantage of evaluating kinetic consequences rather than linguistic features.

C. SCM Phase Analysis and False Negative Interpretation

Disaggregating by convergence phase (Fig. 1e) reveals mechanistically important performance differences. During the transient phase ($t < 45$), ISR is 96.8% [94.2, 98.2] with FPR at 2.8% [1.5, 5.1]. Post-convergence ($t \geq 45$), ISR rises to 99.1% [98.2, 99.6] with FPR dropping to 0.4% [0.1, 1.4]. False positives arose predominantly during rapid environmental transitions where sensor noise temporarily inflated posterior variance. The transient conservative prior closes the exact window a sophisticated MTCM campaign would otherwise exploit. All 12 false negatives in the preliminary cohort occurred during the transient phase and resulted in HALT before any kinetic consequence materialized—detection failures, not protection failures. The same conservative prior that produces the 1.2% aggregate FPR is what closes the MTCM evasion window.

D. Byzantine Resilience and Swarm Scalability

Fig. 1(c) overlays experimentally validated fault tolerance against the theoretical $n \geq 3f + 1$

curve, and Fig. 1(i) disaggregates isolation and recovery timing. The n = 7, f = 3 configuration—deliberately violating the bound—produced catastrophic mission degradation exceeding 80% within 300 ms, confirming the bound's practical tightness. For a 13-agent swarm, the probability of swarm-wide compromise is:

$$\mathbf{P_{compromise} = C(13,4) \times (0.016)^4 \times (0.984)^9 \approx 4.05 \times 10^{-5}}$$

In a geographically dispersed disaster zone, this is operationally infeasible. The causal attestation protocol collapses the attack window further by triggering cryptographic isolation upon the first tainted broadcast. Consensus recovery time decreases from 120 ms at n = 7 to 95 ms at n = 13 (Fig. 1i)—a counter-intuitive improvement explained by increased path diversity in larger meshes, meaning the protocol becomes more efficient as deployment scale increases.

VII. CONCLUSION AND FUTURE WORK

As mechatronic systems transition from deterministic automation to LLM-driven reasoning, the attack surface shifts from the network layer to the semantic, cognitive layer. Existing defenses cannot follow: cloud-based guardrails are unavailable in infrastructure-denied environments, dual-LLM monitors are gated by full inference completion, and statistical filters are semantically blind to the attack class that matters. The correct response is not a faster anomaly filter but a fundamentally different verification strategy: evaluating the physical consequences of generated actions rather than the linguistic features of the tokens that produced them.

HCG-MAF instantiates this strategy through the Causal Safety Enforcer, which executes counterfactual safety verification via Pearl's do-calculus entirely on-device within 42 ms. Preliminary Sim-to-Real validation across 772 test cases demonstrates a 98.5% Intervention Success Rate with 1.2% false positive rate and full offline operation. Byzantine-resilient decentralized consensus, secured through cryptographic causal attestation, maintains swarm cohesion under the $n \geq 3f + 1$ bound with agent isolation completing within 44 ms.

Four immediate research directions follow. (i) Expansion to the full 3,000-sample Sim-to-Real matrix with McNemar's significance testing against each baseline. (ii) Migration to 1.58-bit ternary LLM architectures—specifically BitNet b1.58 [31]—for sub-1,000 ms inference on Raspberry Pi 5-class hardware. (iii) Heterogeneous cross-domain swarms integrating aerial UAVs alongside ground UGVs, requiring a meta-causal layer mapping between domain-specific SCMs while preserving shared safety invariants. (iv) Adaptive adversarial training in which a red-team LLM iteratively probes and adapts to evolving CSE prior structure, stress-testing robustness to adversaries who have learned specifically how to target the convergence dynamics.

The central contribution of this work is methodological: from filtering adversarial language to blocking unsafe physics. When a generative model governs physical actuators, safety verification must occur at the kinetic layer. The causal framework developed here is a principled and deployable step toward making that requirement operational across the full range of autonomous physical intelligence applications.

. REFERENCES

[1] E. F. Camacho and C. B. Alba, Model Predictive Control, 2nd ed. London, U.K.: Springer, 2007.

[2] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, Robotics: Modelling, Planning and Control. London, U.K.: Springer, 2009.

[3] T. Dettmers, M. Lewis, Y. Belkada, and L. Zettlemoyer, "LLM.int8(): 8-bit matrix multiplication for transformers at scale," in Proc. Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 35, 2022, pp. 30318–30332.

[4] H. Touvron et al., "LLaMA: Open and efficient foundation language models," arXiv preprint arXiv:2302.13971, Feb. 2023.

[5] M. Ahn et al., "Do as I can, not as I say: Grounding language in robotic affordances," in Proc. Conf. Robot Learning (CoRL), Atlanta, GA, USA, 2022.

[6] Y. Wu, S. Peng, and H. Li, "Semantic attack surfaces in LLM-controlled autonomous agents: A threat analysis for physical deployment," in Proc. IEEE Int. Conf. Robotics and Automation (ICRA), Yokohama, Japan, 2024, pp. 1–8.

[7] Z. Chen, J. Liu, and R. Wang, "Edge general intelligence for autonomous mechatronic systems: A survey of deployment frameworks and open security challenges," IEEE Trans. Ind. Electron., vol. 70, no. 11, pp. 11432–11445, Nov. 2023.

[8] K. Greshake, S. Abdelnabi, S. Mishra, C. Endres, T. Holz, and M. Fritz, "Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injections," in Proc. ACM Workshop Artif. Intell. Security (AISec), Copenhagen, Denmark, 2023, pp. 79–90.

[9] R. Bailey, T. Sperl, and A. Mallios, "Embodied prompt injection: Adversarial semantics as kinetic attack vectors in LLM-driven robotic systems," arXiv preprint arXiv:2310.14453, Oct. 2023.

[10] Y. M. Shoukry et al., “PyCRA: Physical challenge-response authentication for active sensors under spoofing attacks,” in Proc. ACM CCS, Denver, CO, USA, 2015, pp. 1004–1015.

[11] G. Zhang et al., “DolphinAttack: Inaudible voice commands,” in Proc. ACM SIGSAC CCS, Dallas, TX, USA, 2017, pp. 103–117.

[12] E. Bagdasaryan, T.-Y. Hsieh, B. Nassi, and V. Shmatikov, “Abusing images and sounds for indirect instruction injection in multi-modal LLMs,” arXiv preprint arXiv:2307.10490, Jul. 2023.

[13] S. Ruan et al., “Adversarial robustness of vision-language-action models in manipulation and navigation tasks,” in Proc. IEEE/RSJ IROS, Abu Dhabi, UAE, 2024, pp. 1–9.

[14] G. Dong, J. Jiang, and Z. Yu, “SafeMind: Dual-LLM architecture for real-time anomaly detection in embodied AI command pipelines,” in Proc. ICLR Workshop LLM Safety, Vienna, Austria, 2024.

[15] H. Inan et al., “Llama Guard: LLM-based input-output safeguard for human-AI conversations,” arXiv preprint arXiv:2312.06674, Dec. 2023.

[16] A. D. Ames et al., “Control barrier functions: Theory and applications,” in Proc. Eur. Control Conf. (ECC), Naples, Italy, Jun. 2019, pp. 3420–3431.

[17] R. Salay, R. Queiroz, and K. Czarnecki, “An analysis of ISO 26262: Using machine learning safely in automotive software,” arXiv preprint arXiv:1709.02435, 2018.

[18] J. Pearl, Causality: Models, Reasoning and Inference, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2009.

[19] J. Peters, D. Janzing, and B. Schölkopf, Elements of Causal Inference: Foundations and Learning Algorithms. Cambridge, MA, USA: MIT Press, 2017.

[20] R. K. Mothilal, A. Sharma, and C. Tan, “Explaining machine learning classifiers through diverse counterfactual explanations,” in Proc. ACM FAccT, Barcelona, Spain, 2020, pp. 607–617.

[21] A. Cheng, X. Yin, and Z. Liu, “Causal inference for runtime safety verification in autonomous cyber-physical systems,” in Proc. IEEE ICST, Dublin, Ireland, 2023, pp. 1–11.

[22] P. Madumal, T. Miller, L. Sonenberg, and F. Vetere, “Explainable reinforcement learning through a causal lens,” in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 3, Feb. 2020, pp. 2493–2500.

[23] H. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram, “Resilient asymptotic consensus in robust networks,” IEEE J. Sel. Areas Commun., vol. 31, no. 4, pp. 766–781, Apr. 2013.

[24] S. Sundaram and C. N. Hadjicostis, “Distributed function calculation via linear iterative strategies in the presence of malicious agents,” IEEE Trans. Autom. Control, vol. 56, no. 7, pp. 1495–1508, Jul. 2011.

[25] K. Zhu, X. Liu, H. Zhang, and Y. Wu, “An improved PBFT-Raft hybrid consensus algorithm with digital attestation for Byzantine-resilient multi-agent systems,” IEEE Access, vol. 11, pp. 58432–58445, 2023.

[26] L. Zhang, M. Chen, and S. Wang, “Zero-trust architectures for collaborative multi-LLM edge intelligence: Security threats, inter-agent prompt injection, and mitigation strategies,” IEEE Trans. Netw. Service Manag., vol. 21, no. 2, pp. 1243–1260, Apr. 2024.

[27] NIST, Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations, NIST AI 100-2e2023, Gaithersburg, MD, USA: National Institute of Standards and Technology, 2023.

[28] G. Deng, G. Xu, Y. Zhou, T. Zhang, and Y. Liu, “On the (in)security of secure ROS2,” in Proc. ACM SIGSAC Conf. Comput. Commun. Security (CCS), Los Angeles, CA, USA, 2022, pp. 739–753.

[29] “Prompt injection 2.0: Hybrid AI threats,” arXiv preprint arXiv:2507.13169, 2025.

[30] K. Eykholt et al., “Robust physical-world attacks on deep learning visual classification,” in Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 1625–1634.

[31] S. Ma, H. Wang, L. Ma, L. Wang, W. Wang, S. Huang, L. Dong, R. Wang, J. Xue, and F. Wei, “The era of 1-bit LLMs: All large language models are in 1.58 bits,” arXiv preprint arXiv:2402.17764, Feb. 2024.