Event-Driven LLM-Augmented Multi-Agent Coordination for Real-Time Emergency Decision Making

Savitri Vemireddi[1], Dr. S Jhansi Rani[2]
[1,2]*Department of Computer Science and Systems Engineering, Andhra University College of Engineering, Andhra University, Visakhapatnam, Andhra Pradesh, India*

Abstract*—Emergency response requires rapid decisions under partial observability, heterogeneous data and uncertain evidence. Centralized decision-support systems become bottlenecks when many events arrive at once, while purely large-language-model (LLM) solutions can hallucinate, reason inconsistently and offer weak numerical guarantees. This paper proposes HEMA-CMS (Hierarchical Event-driven Multi-Agent Crisis Management System), an event-driven LLM-augmented architecture that combines four specialized agent tiers, a priority publish-subscribe Event Bus, a Raft-elected Global Coordinator (GC), deterministic decision-support services and Dempster-Shafer evidence fusion. Tier 1 agents ingest IoT/sensor, social-media, satellite/UAV and historical data; Tier 2 Situation Analysis Agents (SAA) fuse machine-learning (ML), LLM and geospatial belief masses with the proposed Fusion-SAA algorithm, which flags near-total evidence conflict (conflict mass $K \geq 1 - \varepsilon$) instead of hiding it; Tier 3 agents negotiate tasks; and Tier 4 agents issue validated recommendations. A seeded synthetic evaluation (30 runs × 2000 events) shows that conflict-triggered escalation lowers the rate of unsafe high-confidence errors of plain Dempster fusion from 5.4% to 3.9% under normal evidence and from 14.1% to 9.1% under conflicting evidence, at the price of escalating 18% and 35% of events to operators, and that a priority Event Bus cuts the mean waits of urgent events about four-fold at 80% broker utilization in an analytical queueing model.

***Index Terms*—Apache Kafka, crisis management, Dempster-Shafer theory, emergency management, event-driven architecture, information fusion, large language models, multi-agent systems, real-time decision support.**

I. INTRODUCTION

Emergency management is a distributed decision problem in which several organizations and computational components must act with incomplete and sometimes contradictory information. Disaster response may involve emergency services, hospitals, police, transportation authorities, field teams, unmanned aerial vehicles (UAVs), sensor networks and public communication channels. The decision makers do not necessarily observe the same state and must coordinate under time pressure. Lee and Lee modeled disaster response as a decentralized partially observable multi-agent problem, showing why distributed decision making is appropriate when agents possess incomplete system information [1].

The growth of large language models (LLMs) changes the design space for emergency decision support. LLMs can interpret natural-language incident reports, summarize heterogeneous observations, decompose tasks, formulate explanations and communicate with operators. ReAct demonstrated the benefit of combining reasoning with external actions and information retrieval rather than requiring a language model to solve every subproblem internally [2]. AutoGen showed that LLM applications can be composed from multiple conversable agents with different roles, tools and human participation modes [3].

LLMs alone, however, are not sufficient for safety-critical emergency management, which must distinguish semantic interpretation from verifiable computation. Routing, resource-capacity checks, optimization constraints and evidence aggregation should be performed by deterministic or explicitly auditable components. Moreover, emergency data can be mutually inconsistent: an ML classifier may indicate high risk, an LLM may interpret a textual report as medium risk, while geospatial evidence may indicate a low-risk location. Dempster-Shafer theory (DST) provides a formal framework for representing and combining uncertain evidence [4], [5], and a

recent systematic review found DST widely used for information fusion in natural-disaster emergency management [6].

Reviews of LLM research in disaster management point toward multimodal information fusion, multi-source validation and multi-agent collaboration. Xu et al. reviewed 70 LLM-focused disaster-management studies and identified persistent gaps in cross-stakeholder integration and in converting situation-awareness information into actionable decisions [7]. A 2026 review of multi-agent systems in emergency management likewise reports a transition from simulation-based systems toward optimization- and learning-based systems for adaptive real-time decision making [8]. These findings motivate an architecture that does not treat the LLM as a monolithic controller but embeds it inside an event-driven, hierarchical agent system.

This paper therefore proposes HEMA-CMS, a four-tier architecture with a priority Event Bus and a Global Coordinator. The central research question is whether event-triggered specialization, LLM-assisted situation analysis and explicit evidence fusion can provide a more responsive and auditable decision pipeline than centralized or single-agent alternatives. The contributions are: (1) a hierarchical event-driven multi-agent architecture; (2) an LLM-augmented situation-analysis layer; (3) a Dempster-Shafer Fusion-SAA algorithm for ML, LLM and geospatial evidence; (4) a coordination mechanism with feedback and escalation; and (5) a reproducible experimental and statistical evaluation protocol together with a seeded numerical evaluation of the fusion layer and an analytical model of the Event Bus.

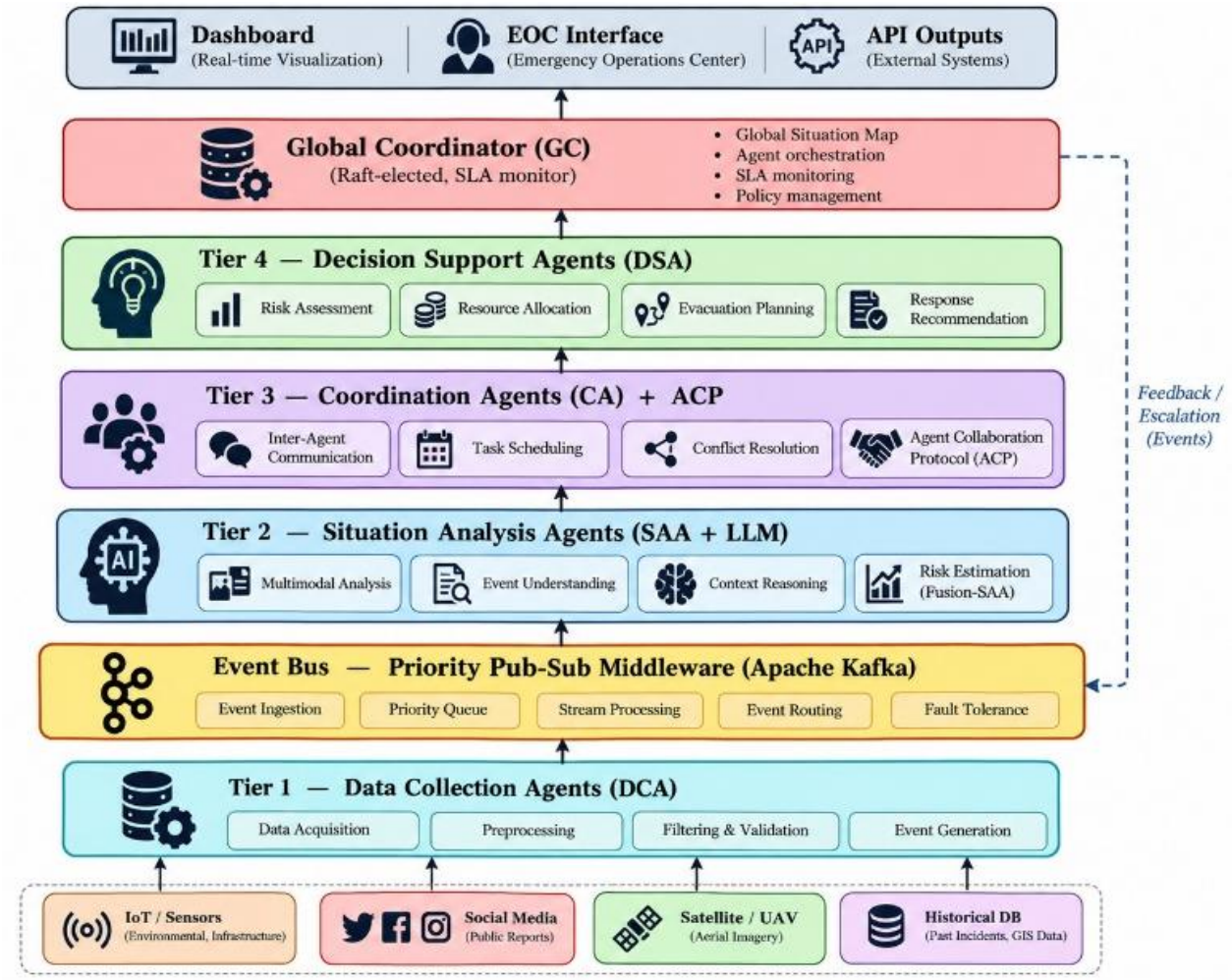


Fig. 1. HEMA-CMS high-level architecture. Data flows upward through four agent tiers; feedback and escalation events return to the Event Bus

II. RELATED WORK

A. Multi-Agent Emergency Decision Making

Multi-agent approaches suit emergency environments because tasks can be distributed among agents with specialized objectives. Lee and Lee used multi-agent reinforcement learning for selective patient admission under mass-casualty conditions and reported that decentralized decision making can exploit partial observations while reducing the difficulty of directly solving a decentralized partially observable Markov decision process [1].

More recent work continues this direction through learning-based multi-agent systems, resource-allocation optimization and adaptive emergency logistics [8], [9]. The limitation of many conventional designs is that input information is assumed to be structured. Real emergency operations contain free-form reports, social-media text, images and mixed-quality observations, which motivates LLM augmentation.

B. LLM Agents and Tool Use

ReAct established a reasoning-and-action pattern in which a language model can reason about a task and invoke external information sources or tools [2]. AutoGen generalized multi-agent LLM applications by allowing agents to converse and combine LLMs, tools, code execution and human input [3]. These ideas are useful for emergency systems when LLM calls are constrained by explicit tools and validation mechanisms. In HEMA-CMS the LLM is assigned to semantic interpretation, task decomposition, explanation and evidence extraction, while routing, resource feasibility and belief fusion remain explicitly defined computational services.

C. Evidence Fusion

Dempster introduced a formal treatment of combining evidence represented by multivalued mappings [4], while Shafer developed the mathematical theory of evidence and Dempster's rule of combination [5]. Smets and Kennes described a transferable belief model in which belief functions can be transformed into probabilities for decision making [10]. These foundations are particularly relevant when emergency evidence is incomplete rather than simply noisy. However, Dempster's normalized rule can become unstable under severe conflict. HEMA-CMS therefore makes conflict itself an observable event: when the conflict mass approaches one, the system raises a high-conflict flag instead of silently presenting an apparently precise risk probability.

D. Event-Driven Streaming

An event-driven architecture decouples event producers from downstream consumers. Apache Kafka provides publish/subscribe semantics, durable event storage, partitioning and stream processing, making it suitable as the Event Bus in a real-time prototype [11]. The proposed architecture uses event priorities and topic-level specialization so that urgent events are processed before low-priority historical or background updates. This reduces direct coupling between data collectors and decision agents and supports replay for audit and experimental reproducibility.

III. HEMA-CMS ARCHITECTURE

A. Design Principles

HEMA-CMS is designed around five principles: (i) event-driven activation, (ii) specialized agent responsibilities, (iii) hybrid LLM and deterministic reasoning, (iv) explicit uncertainty representation and (v) human escalation for high-impact actions. The architecture (Fig. 1) contains Tier 1 Data Collection Agents (DCA), Tier 2 Situation Analysis Agents (SAA), Tier 3 Coordination Agents (CA) with an Agent Coordination Protocol (ACP) and Tier 4 Decision Support Agents (DSA). A Global Coordinator (GC) sits above these tiers for global state maintenance, service-level agreement (SLA) monitoring, feedback and escalation.

B. Tier 1—Data Collection Agents

DCA instances are responsible for ingestion, normalization, provenance tagging, timestamp alignment and basic quality checks. Four principal sources are represented in Fig. 1: IoT/sensors, social media, satellite/UAV observations and historical databases.

Each observation is transformed into a canonical event $e_i = \langle \text{id, timestamp, source, location, type, payload, confidence, priority} \rangle$. Source confidence should be estimated independently of the LLM, using calibration statistics or known sensor characteristics where available.

C. Event Bus

The Event Bus is priority-aware publish/subscribe middleware. Kafka supports durable topics, producer-consumer decoupling and stream processing [11]. HEMA-CMS can define topics such as incident.raw, incident.high, fusion.request, coordination.task, decision.review and feedback.escalation. Event priority $P(e)$ can be computed from severity, uncertainty, affected population and time sensitivity. The broker does not decide the emergency action; it only ensures that the right event reaches the right consumers with appropriate ordering and persistence.

D. Tier 2—Situation Analysis Agents

SAA instances convert heterogeneous observations into structured situation hypotheses. A machine-learning component provides a calibrated class distribution; the LLM extracts entities, hazards, temporal cues and contextual relationships from text; and a geospatial module derives location-specific evidence. These three sources are represented as belief mass functions over $\Theta = \{\text{Low, Medium, High}\}$. The Fusion-SAA algorithm combines them and outputs a pignistic risk vector. The LLM is not permitted to overwrite sensor or ML evidence; it contributes an independently tracked evidence source.

E. Tier 3—Coordination Agents and ACP

Coordination Agents translate assessed situations into executable sub-tasks, for example resource-allocation, route, medical, evacuation, infrastructure and communications tasks. The ACP specifies task identity, required capabilities, deadline, dependencies, evidence requirements and completion status as ⟨task_id, incident_id, owner, dependencies, deadline, priority, status⟩. The coordinator can activate multiple agents concurrently when dependencies permit and can issue a new event when an upstream result invalidates a downstream assumption.

F. Tier 4—Decision Support Agents

DSA instances synthesize validated coordination outputs into recommended actions such as evacuation advice, resource deployment, shelter assignment, hospital routing and public-warning messages. Each recommendation contains an action, rationale, evidence identifiers, confidence, expected impact and required approval level. High-impact actions are sent to an operator rather than executed automatically.

G. Global Coordinator

The GC maintains a Global Situation Map and monitors service-level objectives. A Raft-style election provides coordinator failover so that the system does not depend on a single fixed coordinator. The GC does not execute all analytical work. Its role is supervisory: maintaining state references, detecting SLA violations, issuing feedback and escalation events, and selecting or recovering coordinator instances.

This avoids turning the global layer into the processing bottleneck that the architecture is intended to eliminate.

IV. MATHEMATICAL FORMULATION

A. Event Priority

For an event e, a normalized priority score is defined as

$$P(e) = \alpha S + \beta U + \gamma I + \delta T,\ \alpha + \beta + \gamma + \delta\ 1 \quad (1)$$

where S denotes severity, U uncertainty, I expected impact and T temporal urgency.

The broker maps $P(e)$ to discrete priority classes. The weights should be learned or calibrated on domain-specific historical incidents rather than chosen arbitrarily.

B. Belief-Mass Representation

Let $\Theta = \{L, M, H\}$ denote Low, Medium and High risk. A basic probability assignment (BPA) m satisfies $\Sigma_{A \subseteq \Theta}$ $m(A) = 1$ and $m(\emptyset) = 0$. Each evidence source produces m_{ML}, m_{LLM} and m_{Geo}. Focal sets such as $\{M, H\}$ allow the system to represent ambiguity rather than forcing every observation into a singleton class [5], [10].

C. Dempster Combination

For two BPAs m_a and m_b, the unnormalized conjunctive mass for A is

$$m \wedge (A) = \Sigma * B *\cap* C *=* A * ma(B)\, mb(C) \quad (2)$$

The conflict mass is

$$K = \Sigma * B *\cap* C *= \emptyset\ ma(B)\, mb(C) \quad (3)$$

For $A \neq \emptyset$, the normalized Dempster combination is

$$m(A) = m \wedge (A) / (1 - K) \quad (4)$$

When K approaches one, the denominator becomes unstable. HEMA-CMS treats this condition as a high-conflict event and escalates it rather than hiding the disagreement.

D. Pignistic Transformation

The pignistic probability used for decision ranking distributes the mass of composite focal sets among their singleton members.

For a singleton θ,

$$BetP(\theta) = \Sigma * A *\subseteq \theta,* \theta *\in* A * m(A) / |A| \quad (5)$$

The resulting vector p_{fused} = [BetP(L), BetP(M), BetP(H)] is published to the situation-assessment topic, following the decision-oriented role of pignistic probabilities in the transferable belief model [10].

V. FUSION-SAA ALGORITHM

Algorithm 1 is the core evidence-fusion procedure. The normalization factor is $1 - K$; the near-total-conflict test is therefore $K \geq 1 - \varepsilon$, where ε is a small operational threshold.

If this condition is reached, the system does not silently normalize the evidence. Instead, it publishes a high conflict event containing the contributing source identifiers and evidence vectors.

The second conflict check (line 8) is needed because conflict may be introduced by the geospatial evidence even when ML and LLM evidence agree.

ALGORITHM 1. Fusion-SAA: Dempster-Shafer evidence combination

Require: belief mass functions m_{ML}, m_{LLM}, m_{Geo} over Θ = {Low, Medium, High}; conflict threshold ε

Ensure: fused risk probability vector p_{fused}

1: $m_{12} \leftarrow \text{DempsterCombine}(m_{\text{ML}}, m_{\text{LLM}})$
2: $K_{12} \leftarrow \text{Conflict}(m_{\text{ML}}, m_{\text{LLM}})$
3: if $K_{12} \geq 1 - \varepsilon$ then
4: publish high_conflict event with source IDs and evidence summary
5: request anomaly / LLM / ML verification
6: end if
7: $m_{123} \leftarrow \text{DempsterCombine}(m_{12}, m_{\text{Geo}})$
8: $K_{123} \leftarrow \text{Conflict}(m_{12}, m_{\text{Geo}})$
9: if $K_{123} \geq 1 - \varepsilon$ then
10: publish high_conflict event and escalate to coordinator
11: end if
12: $p_{\text{fused}} \leftarrow \text{PignisticTransform}(m_{123})$
13: publish p_{fused} to situation/assessment topic
14: return p_{fused}

VI. EXPERIMENTAL DESIGN

A. Research Questions and Hypotheses

RQ1) Does event-driven activation reduce end-to-end decision latency relative to centralized processing?
RQ2) Does LLM augmentation improve interpretation and decision accuracy for unstructured reports?
RQ3) Does Fusion-SAA improve robustness when evidence sources disagree?
RQ4) Does hierarchical specialization reduce coordination overhead under increasing event rates?

H1 predicts lower median latency for HEMA-CMS than the centralized baseline. H2 predicts higher decision accuracy than a single LLM agent when evidence is heterogeneous. H3 predicts fewer unsafe high-confidence decisions under conflicting evidence. H4 predicts sublinear growth in coordination overhead as irrelevant agents remain inactive.

B. Scenario Set

Four scenario families are defined: flood, urban fire, earthquake and road/infrastructure disruption. Each includes normal, moderate, severe and conflicting-evidence variants. Events arrive in bursts to stress the Event Bus and in sparse mode to measure baseline overhead. A minimum of 30 independent runs per configuration is used, with fixed random seeds recorded for reproducibility.

C. Baselines

B1: centralized rule-based decision support; B2: conventional multi-agent system (MAS) without LLM reasoning; B3: single LLM agent with tool access; B4: HEMA-CMS without Fusion-SAA (ablation); B5: full HEMA-CMS.

The ablations isolate the contribution of event orchestration, LLM augmentation and evidence fusion rather than attributing all improvements to the complete system.

VII. RESULTS AND DISCUSSION

This section reports three numerical studies of the components that can be evaluated independently of a deployed prototype: a worked Fusion-SAA example, an analytical queueing model of the priority Event Bus (H1, H4) and a seeded Monte-Carlo evaluation of the fusion layer (H3). End-to-end latency, task completion

and coordination overhead of the full systems B1–B5 require the deployed prototype and are left to future work.

A. Worked Fusion-SAA Example

Consider the situation of Section I: the ML classifier indicates high risk ($m_{ML}(\{H\}) = 0.6$, $m_{ML}(\{M,H\}) = 0.2$, $m_{ML}(\Theta) = 0.2$), the LLM indicates medium risk ($m_{LLM}(\{M\}) = 0.5$, $m_{LLM}(\{M,H\}) = 0.3$, $m_{LLM}(\Theta) = 0.2$) and the geospatial module indicates low risk ($m_{Geo}(\{L\}) = 0.4$, $m_{Geo}(\{L,M\}) = 0.3$, $m_{Geo}(\Theta) = 0.3$). Fusion-SAA obtains $K_{12} = 0.30$ and $K_{123} = 0.51$, both below the threshold used later ($1 - \varepsilon = 0.8$), and the fused pignistic vector [BetP(L), BetP(M), BetP(H)] = [0.075, 0.584, 0.341] (Fig. 2).

The fused vector ranks Medium first because the medium and high hypotheses overlap in the composite focal set $\{M, H\}$, but it retains the substantial high-risk mass (0.341) so that the disagreement remains visible to the operator.

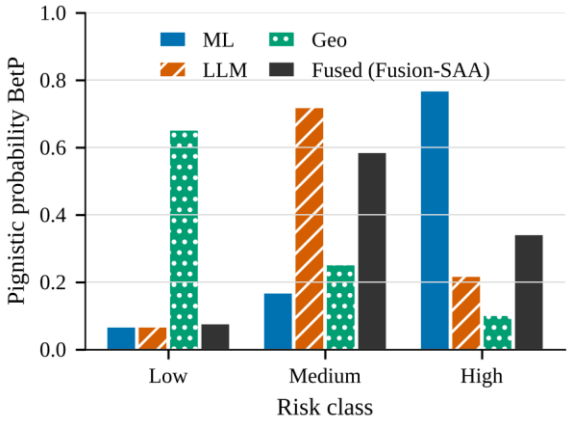


Fig. 2. Pignistic probability BetP of the three evidence sources and of the Fusion-SAA output for the worked example.

B. Priority Event Bus: Analytical Model

To quantify the benefit of priority-aware routing (H1, H4), the broker is modeled as a single-server queue with Poisson arrivals, exponential service of unit means and a non-preemptive priority discipline in which 20% of events are urgent. Cobham's formula gives the mean wait of class k as $W_k = W_0 / [(1 - \sigma_{k-1})(1 - \sigma_k)]$, where $W_0 = \rho$ for unit-mean exponential service and σ_k is the cumulative utilization of classes $1 \ldots k$. A first-in-first-out (FIFO) broker has $W = \rho/(1 - \rho)$. At $\rho = 0.8$ the mean waits of an urgent event fall from 4.0 service times under FIFO to 0.95 under priority routing, while low-priority events wait 4.76 service times (Fig. 3). A discrete-event simulation of 4×10^5 events confirmed these values (0.94 and 4.66). Priority routing therefore protects urgent events at only a modest cost to background traffic, and the benefit grows with load.

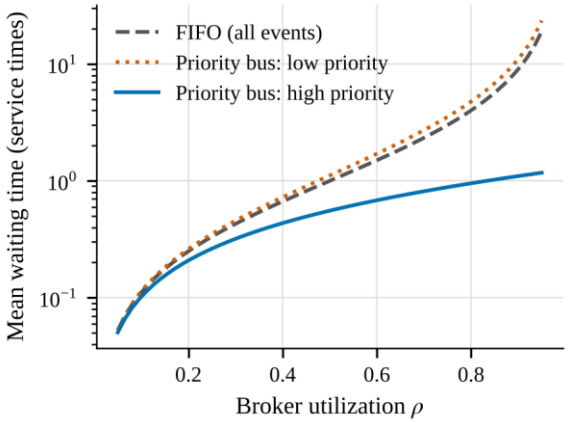


Fig. 3. Mean waiting time versus broker utilization for a FIFO broker and a two-class priority broker (20% urgent events).

C. Synthetic Evaluation of Fusion-SAA

Setup. Each event has a true risk class drawn uniformly from {Low, Medium, High}. The ML, LLM and geospatial sources report the true class with reliabilities 0.85, 0.80 and 0.75, and otherwise a uniformly chosen wrong class. A source that points to class c assigns $m(\{c\}) = q$ with $q \sim U(0.6, 0.95)$, and splits the remaining mass equally between an adjacent two-class focal set and Θ.

In the conflicting-evidence variant, with probability 0.5 one randomly chosen source is replaced by a corrupted source that confidently ($q \sim U(0.8, 0.95)$) points to a wrong class. Five decision rules are compared: ML only, LLM only, linear average of the three pignistic vectors, plain Dempster fusion (ablation B4 analogue) and Fusion-SAA with $\varepsilon = 0.2$, where flagged events are escalated to an operator and not decided automatically.

A decision is *unsafe* if it is executed automatically, is wrong and has max BetP ≥ 0.7. Each configuration uses 30 independent runs (seeds 1–30) of 2000 events; results are means ± 95% confidence intervals (t-distribution) over runs.

TABLE I Accuracy, unsafe-error rate and escalation rate (%) of the five decision rules (mean ± 95% CI over 30 runs)

Evidence	Method	Accuracy[a]	Unsafe error	Escalated
Normal	ML only	84.7 ± 0.3	15.3 ± 0.3	0
	LLM only	80.1 ± 0.3	19.9 ± 0.3	0
	Linear average	91.4 ± 0.2	0.2 ± 0.0	0
	Dempster fusion	90.0 ± 0.2	5.4 ± 0.2	0
	Fusion-SAA	92.0 ± 0.2	3.9 ± 0.1	18.0 ± 0.3
Conflicting	ML only	71.0 ± 0.3	29.0 ± 0.3	0
	LLM only	66.3 ± 0.5	33.7 ± 0.5	0
	Linear average	79.1 ± 0.3	0.6 ± 0.1	0
	Dempster fusion	76.1 ± 0.3	14.1 ± 0.3	0
	Fusion-SAA	79.0 ± 0.4	9.1 ± 0.2	35.4 ± 0.3

[a]Accuracy of Fusion-SAA is computed over the events it decides automatically; all other rules decide every event.

Results. Under normal evidence Fusion-SAA decides 82.0% of events automatically with 92.0 ± 0.2% accuracy and an unsafe-error rate of 3.9 ± 0.1%, against 5.4 ± 0.2% for plain Dempster fusion (Table I). Under conflicting evidence, the corresponding unsafe-error rates are 9.1 ± 0.2% and 14.1 ± 0.3%, while 35.4 ± 0.3% of events are escalated.

Fig. 4(a) shows that the accuracy of fusion rises with the reliability of the LLM source, overtakes the ML-only source at an LLM reliability of about 0.6–0.7 and falls below it when the LLM is unreliable, which is the regime in which conflict flags matter, while Fig. 4(b) shows that the conflict mass separates agreeing events ($K \approx 0$) from disputed events, with conflicting scenarios shifting mass above the threshold.

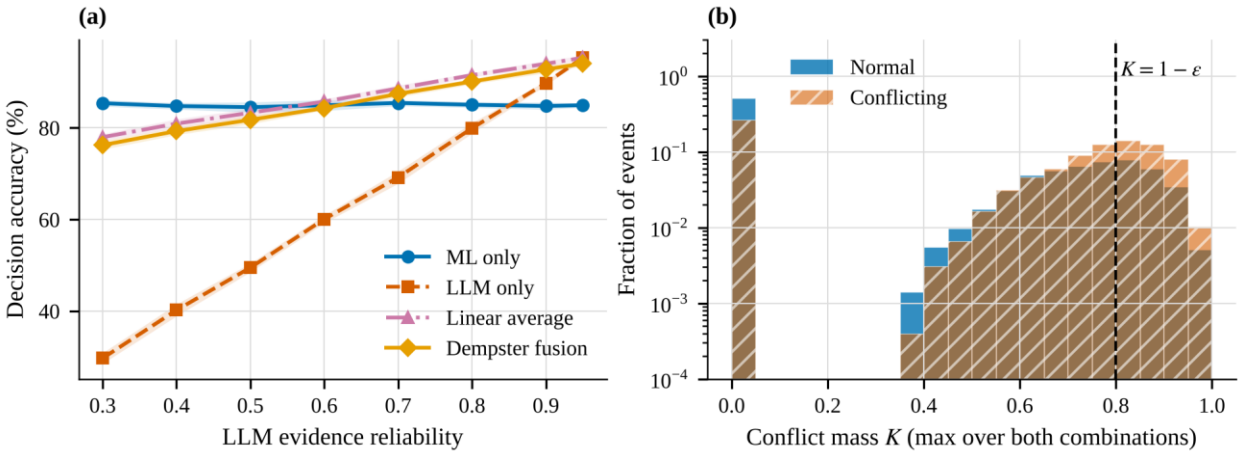


Fig. 4. (a) Decision accuracy versus LLM evidence reliability (shaded: 95% CI over 10 runs of 1000 events). (b) Distribution of the maximum conflict mass over both combination steps for normal and conflicting evidence; the dashed line marks the escalation threshold $1 - \varepsilon$.

TABLE II Paired comparison of Fusion-SAA with plain Dempster fusion over 30 runs (Δ in percentage points; Holm-adjusted p)

Evidence	Metric	Δ (pp)	95% CI	Test	p_{adj}	Effect size d_z
Normal	Unsafe-error rate	-1.56	[-1.65, -1.48]	Wilcoxon signed-rank	$< 10^{-5}$	-6.8
Normal	Accuracy (automated)	+2.05	[+1.92, +2.19]	Wilcoxon signed-rank	$< 10^{-5}$	+5.6
Conflicting	Unsafe-error rate	-4.92	[-5.11, -4.73]	Wilcoxon signed-rank	$< 10^{-5}$	-9.7
Conflicting	Accuracy (automated)	+2.83	[+2.54, +3.12]	Wilcoxon signed-rank	$< 10^{-5}$	+3.7

D. Statistical Reporting

Differences are reported as the paired estimate Δ, its 95% confidence interval, the test statistic, the adjusted *p*-value and an effect size, with sample size $n = 30$ runs (Table II). Fusion-SAA reduced the unsafe-error rate of plain Dempster fusion by 1.56 percentage points

(95% CI [-1.65, -1.48], $p_{adj} < 10^{-5}$, d_z = -6.8) under normal evidence and by 4.92 points (95% CI [-5.11, -4.73], $p_{adj} < 10^{-5}$, d_z = -9.7) under conflicting evidence. Accuracy on automatically decided events rose by 2.05 and 2.83 points, respectively. The very large effect sizes reflect the low run-to-run variance of a synthetic study and should not be extrapolated to field data.

E. Interpretation

Three observations qualify the benefit of Fusion-SAA. First, plain linear averaging of the pignistic vectors is at least as accurate as Dempster fusion on all events (Table I) and almost never exceeds the 0.7 confidence limit, so its unsafe-error rate is the lowest simply because it is under-confident; the value of Dempster fusion lies in its sharper, conflict-aware output rather than in raw accuracy. Second, the conflict flag is a safety signal and not an optimal selector: Fig. 5 shows that, at the same coverage in the conflicting scenario, rejecting events by low max BetP yields 83.9% (linear average) and 82.9% (Dempster fusion) accuracy on the decided events, against 78.9% when rejecting by high *K*. Operators should therefore use *K* to make disagreement visible and could combine it with a confidence criterion.

Third, escalation has a cost ('Escalated' column of Table I); this trade-off between responsiveness and reliability must be reported with any accuracy claim, because a fusion system can appear more accurate merely by being overconfident.

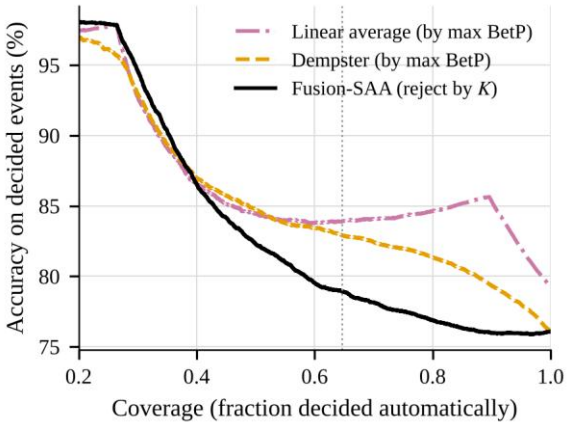


Fig. 5. Accuracy on automatically decided events versus coverage in the conflicting scenario. The dotted line marks the coverage obtained by the Fusion-SAA rule (ε = 0.2).

A lower latency measured for the full prototype should be attributed to event filtering and concurrent specialization, because irrelevant agents are not activated and independent tasks run in parallel, and not to LLM reasoning alone. Likewise, any accuracy gain on text-rich incidents should be traced to the SAA layer, where the LLM extracts semantic cues while ML and geospatial components provide independent evidence.

VIII. ROBUSTNESS, SAFETY, AND FAILURE HANDLING

A. LLM Failure. If the LLM becomes unavailable, Tier 2 falls back to ML and geospatial evidence where possible. The system marks the missing modality and reduces confidence rather than silently treating the missing evidence as neutral.

B. Sensor Anomaly. Sensor outliers are detected before fusion. A sudden sensor value that conflicts with neighboring measurements generates an anomaly event and reduces the source reliability used by the fusion layer.

C. Evidence Conflict. Near-total conflict triggers high_conflict events, which is important because the normalized Dempster rule becomes unstable when *K* approaches one. Alternative conflict-management rules such as proportional conflict redistribution can be evaluated as a future ablation [12].

D. Coordinator Failure. A Raft-style election mechanism maintains supervisory availability. The elected coordinator reconstructs state from durable event logs rather than depending on local memory alone.

E. Human Escalation. Actions involving evacuation orders, medical prioritization, infrastructure shutdown or public warnings include configurable approval thresholds. The architecture preserves the evidence trail used to generate each recommendation.

IX. LIMITATIONS

The proposed architecture has several limitations. First, the numerical results are synthetic: the evidence sources, reliabilities and conflict model are

assumptions, and a simulation cannot reproduce every property of a live emergency, including human behavior, communication outages, organizational policies and adversarial information. Second, LLM performance depends on model choice, prompt design, context length, retrieval quality and inference latency. Third, Dempster-Shafer fusion requires carefully calibrated BPAs; poor calibration can create misleading confidence even when the mathematical combination is correct. Fourth, a Raft-elected coordinator improves availability but does not automatically solve network partitioning, stale state or cross-agency trust. Finally, external validation is necessary before any automated decision is used in a safety-critical operational environment.

X. CONCLUSION

This paper presented HEMA-CMS, an event-driven LLM-augmented multi-agent architecture for real-time emergency decision making. The design separates data acquisition, situation analysis, coordination and decision support while connecting the tiers through a priority Event Bus. The Fusion-SAA algorithm combines ML, LLM and geospatial evidence using Dempster-Shafer theory and converts the resulting belief state into a pignistic probability vector, treating high evidence conflict as an explicit event. In a seeded synthetic evaluation, conflict-triggered escalation reduced unsafe high-confidence errors of plain Dempster fusion in both normal and conflicting scenarios, and an analytical queueing model showed a four-fold reduction in the mean wait of urgent events at 80% broker utilization. The same evaluation showed that conflict mass is a visibility mechanism rather than an optimal selector. The experimental protocol defined here can be applied to a deployed prototype to establish the latency, accuracy, resource-use and robustness benefits of the complete system.

XI. FUTURE SCOPE

1) Multimodal SAA agents that jointly reason over text, images, video, sensor streams and satellite observations, including knowledge-graph grounding of LLM reasoning [13].
2) Adaptive conflict-management mechanisms that compare Dempster's rule with proportional conflict redistribution and other combination approaches, and combine the conflict mass with confidence-based rejection.
3) Learning source-reliability weights from historical incidents while maintaining calibration and auditability.
4) Digital-twin simulation in which DSA recommendations are evaluated through what-if scenarios before execution, and LLM agents for infrastructure resilience [14].
5) Federated multi-agency learning so that organizations can improve models without sharing sensitive raw data.
6) Formal verification of high-risk workflows and provenance-aware explanations linking each decision to its evidence.
7) Real-world pilot deployment with emergency-management agencies, including measurement of the full systems B1–B5, prospective validation and human-factors evaluation.

REFERENCES

[1] H.-R. Lee and T. Lee, “Multi-agent reinforcement learning algorithm to solve a partially-observable multi-agent problem in disaster response,” European Journal of Operational Research, vol. 291, no. 1, pp. 296–308, 2021, doi: 10.1016/j.ejor.2020.09.018.

[2] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, “ReAct: Synergizing reasoning and acting in language models,” in Proc. ICLR, 2023.

[3] Q. Wu et al., “AutoGen: Enabling next-gen LLM applications via multi-agent conversation,” arXiv:2308.08155, 2023; in Proc. COLM, 2024.

[4] A. P. Dempster, “Upper and lower probabilities induced by a multivalued mapping,” Annals of Mathematical Statistics, vol. 38, no. 2, pp. 325–339, 1967, doi: 10.1214/aoms/1177698950.

[5] G. Shafer, A Mathematical Theory of Evidence. Princeton, NJ, USA: Princeton University Press, 1976, doi: 10.1515/9780691214696.

[6] L. Fei, T. Li, and W. Ding, “Dempster–Shafer theory-based information fusion for natural disaster emergency management: A systematic literature review,” Information Fusion, vol. 112, Art. no. 102585, 2024, doi: 10.1016/j.inffus.2024.102585.

[7] F. Xu, J. Ma, N. Li, and J. C. P. Cheng, "Large language model applications in disaster management: An interdisciplinary review," International Journal of Disaster Risk Reduction, vol. 127, Art. no. 105642, 2025, doi: 10.1016/j.ijdrr.2025.105642.

[8] "From simulation to learning: A comprehensive review of multi-agent systems in emergency management," Neurocomputing, vol. 692, Art. no. 133841, 2026, doi: 10.1016/j.neucom.2026.133841.

[9] "A machine learning-enhanced multi-agent framework for multi-timescale emergency resource allocation in natural disasters," in Proc. 2025 2nd Int. Symp. AI and Cybersecurity (ISAICS), 2025, doi: 10.1109/ISAICS66888.2025.11349790.

[10] P. Smets and R. Kennes, "The transferable belief model," Artificial Intelligence, vol. 66, no. 2, pp. 191–234, 1994, doi: 10.1016/0004-3702(94)90026-4.

[11] Apache Software Foundation, "Introduction to Apache Kafka," Apache Kafka Documentation, 2026. OnlineOnline. Available: https://kafka.apache.org/documentation/

[12] F. Smarandache and J. Dezert, "Information fusion based on new proportional conflict redistribution rules," in Proc. 7th Int. Conf. Information Fusion, 2005, doi: 10.1109/ICIF.2005.1591955.

[13] "Enhancing emergency decision-making with knowledge graphs and large language models," International Journal of Disaster Risk Reduction, 2024.

[14] J. Cui, C. Zhai, Y. Wang, and Y. Li, "Large language models and AI agents in disaster-resilient infrastructure: Concepts, applications, pathways, and challenges," Reliability Engineering & System Safety, vol. 272, Art. no. 112497, 2026, doi: 10.1016/j.ress.2026.112497.