Baghdad Science Journal

Volume 23 | Issue 6 Article 24

6-26-2026

Architecting Reliable Multi-Agent Systems with Large Language Models Through Model-Driven Engineering

Najma Imtiaz Ali
Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Malaysia AND Institute of Mathematics and Computer Science, University of Sindh, Jamshoro, Sindh, Pakistan,
najma@utem.edu.my

Aadil Jamali
Institute of Mathematics and Computer Science, University of Sindh, Jamshoro, Sindh, Pakistan,
aadil.jamali@usindh.edu.pk

Imtiaz Ali Brohi
Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Malaysia,
imtiaz@utem.edu.my

Noor Zaman Jhanjhi
School of Computer Science, Faculty of Innovation & Technology, Taylor's University, Malaysia,
noorzaman.jhanjhi@taylors.edu.my

Emaliana Kasmuri
Center for Advanced Computing Technology, Universiti Teknikal Malaysia Melaka, Malaysia,
emaliana@utem.edu.my

How to Cite this Article

Ali, Najma Imtiaz; Jamali, Aadil; Brohi, Imtiaz Ali; Jhanjhi, Noor Zaman; and Kasmuri, Emaliana (2026) "Architecting Reliable Multi-Agent Systems with Large Language Models Through Model-Driven Engineering," *Baghdad Science Journal*: Vol. 23: Iss. 6, Article 24.
DOI: https://doi.org/10.21123/2411-7986.5338

BAGHDAD SCIENCE JOURNAL 2026;23(6):2258–2285

Scan the QR to view the full-text article on the journal website

RESEARCH ARTICLE

Architecting Reliable Multi-Agent Systems with Large Language Models Through Model-Driven Engineering

Najma Imtiaz Ali [1,2,*], **Aadil Jamali** [2], **Imtiaz Ali Brohi** [1], **Noor Zaman Jhanjhi** [3], **Emaliana Kasmuri** [4]

[1] Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Malaysia
[2] Institute of Mathematics and Computer Science, University of Sindh, Jamshoro, Sindh, Pakistan
[3] School of Computer Science, Faculty of Innovation & Technology, Taylor's University, Malaysia
[4] Center for Advanced Computing Technology, Universiti Teknikal Malaysia Melaka, Malaysia

ABSTRACT

Multi-agent systems based on large language models (LLM) are said to be able to provide flexible and scalable collaboration, but are often unstable, adversarially active, biased in their representations, and expensive to compute - aspects that hinder their use in safety-critical or resource-sensitive systems. To overcome these deficiencies, a model-driven engineering framework, the MDE-Agentware, has been implemented: a model-based engineering paradigm, which represents multi-agent architectures through a typed domain specific metamodel, and generates executables that are conforming with instrumented runtime monitoring, tool wrappers and energy-conscious invocation policies. The main novelties of the approach include (i) a formal model-to-code transformation, enforcing the architectural constraints and labeled-transition semantics, (ii) a collection of rigorous metrics (architectural conformity, prediction consistency, composite bias penalty, and energy models), which allow performing automated conformity checking and constrained optimization, and (iii) fairness-by-design and resilience mechanisms, implemented into the orchestration level, and not applied after the fact. Huge controlled experiments with publicly available corpora and simulated benchmarks show that MDE-Agentware achieves significantly better contraction behavior and spectral condition, is more resistant to adversarial noise, has higher inter-agent coherence, exhibits significant reductions in redundant invocations of LLM and per-trial energy, and significant reductions in statistical parity difference. The framework thus progresses a viable, repeatable, and reliable, and environmentally sustainable multi-agent system based on LLM, which can be utilized in critical applications.

Keywords: Adversarial robustness, Bias mitigation, Large language models, Multi-agent orchestration, Model-driven engineering, Sustainability

Introduction

Recent turbo accelerated growth of artificial intelligence (AI) and natural language processing (NLP) has resulted in the introduction of models of large language models (LLMs), which manifest impressive reasoning, natural language dialogue, code generation, as well as decision support. Such ways of development have triggered the creation of multi-agent systems (MAS) that are driven by LLMs, in which heterogeneous agents are used to cooperate in order to execute complex tasks with as little human involvement as they could. These systems have vast potential in a variety of areas, such as sci-

Received 14 October 2025; revised 29 December 2025; accepted 9 January 2026.
Available online 26 June 2026

* Corresponding author.
E-mail addresses: najma@utem.edu.my (N. I. Ali), aadil.jamali@usindh.edu.pk (A. Jamali), imtiaz@utem.edu.my (I. A. Brohi), noorzaman.jhanjhi@taylors.edu.my (N. Z. Jhanjhi), emaliana@utem.edu.my (E. Kasmuri).

https://doi.org/10.21123/2411-7986.5338

entific discovery, medicine, financial analytics, and autonomous software engineering, and collaborative intelligence and adaptive problem solving are essential in these fields.[1,2] Nevertheless, their promise is yet to be met, and the current roadblocks preventing the adoption of the technology to date are reliability, robustness, explainability, and trustworthiness.[3]

One of the key issues when it comes to the application of LLM-driven agents is that they are prone to hallucinations, contradictions, and biases. Hallucination: Models produce semantically incorrect, syntactically correct output, which negatively affects the robustness of systems in high-stakes systems.[4] In a similar manner, variance in interactions among agents causes disorder in agents' processes in MAS, and biases inherent in training data may transfer risks relating to ethics and unfair results.[5] These restraints signify the acute necessity of systematic design measures that may incorporate formal, agreeable, artistically well, and conscience-binding assessment devices into how LLCM agents are established.

One response to these problems, offered by Model-Driven Engineering (MDE), is the availability of a formalized paradigm of high-level representation of system architectures, behavior, and constraints, by establishing formal frameworks.[6] MDE principles enable one to build MAS with accurate specifications, thorough verification and refinement, eliminating associated risks of ad-hoc or purely data-based solutions/objectives. LLM abilities along with MDE approaches provide a groundbreaking approach to creating robust and reliable systems in which interactions and the chains of reasoning by agents and the use of tools can be specified, tested, and optimized.

More recent work in the MAS community has also been performed to benchmark frameworks like AgentBench, RefactorBench, and ToolBench schemes that judge the agents based on parameters in tool correctness, reasoning accuracy, and adaptability.[7–9] Though such benchmarks do offer some valuable information on the model performance, they are primarily in the mode of an isolated task, or tool call, and not the architectural design or resiliency of the MAS. As a result, the evaluation methodologies and engineering practices have a disparity that entails an integrative framework involving the combination of model-driven design, empirical evaluation, and adaptive monitoring.

This paper presents a new model-driven engineering framework, the MDE-Agentware, allowing the building of resilient and reliable MAS using LLM. The framework uses metamodels representing agent roles, communication protocols, and interaction semantics, and measures to assess prediction consistency, architectural conformity, and bias mitigation. A variety of simulated experiments with numerous public datasets and synthetic settings show that the framework is capable of improving reliability, minimizing the spreading of errors, and increasing explainability. The book discussed below fills the gap between AI competencies and engineering standards and forms the groundwork of the next generation of MAS, which will be intelligent and reliable. The overall framework is shown in Fig. 1, highlighting the layered flow of agents, tools, and monitoring components.

Related work

The application of large language models in multi-agent systems has spurred a substantial body of research in the artificial intelligence, distributed computing, and software engineering communities. A number of works have been done on coordinating autonomous agents through the help of LLMs to coordinate efforts in a problem-solving activity. Indicatively, AutoGPT and BabyAGI are some of the initial efforts to make recursive reasoning and task decomposition accessible to agents so that they could organize workflows with very little human input to the agent.[10] However, despite the promise of the autonomous orchestration techniques embodied in such systems, they are frequently not architecturally transparent and offer small amounts of support for correctness or resiliency in the face of dynamic deployments, as paralleled deployments elsewhere are both frequent and miraculous regardless of load changes with scale.[11]

New developments in benchmarking models have attempted to analyze the skills of agents using LLM towards being able to compare them methodically. Assessment-structured environments have been offered by benchmarks like AgentBench,[12] RefactorBench,[13] and ToolBench[14] to evaluate the reasonability in agent reasoning, the reliability and accuracy in code, and the use of specific tools, respectively. Those efforts have been pinpointing that even though the performance of the LLM agents is strong in certain sets of tasks, such actors often show shortcomings in the planning in perspectives of the business, stability in inter- dialogue, and adversarial-resistance. In addition, the existing standards emphasize empirical task performance as opposed to architectural or systemic characteristics that give resilience and reliability a backbone.[15] The DSL structure is illustrated in Fig. 2, capturing entities, relations, and attributes that define system consistency.

Similar work in software engineering research has placed an increased emphasis on the importance

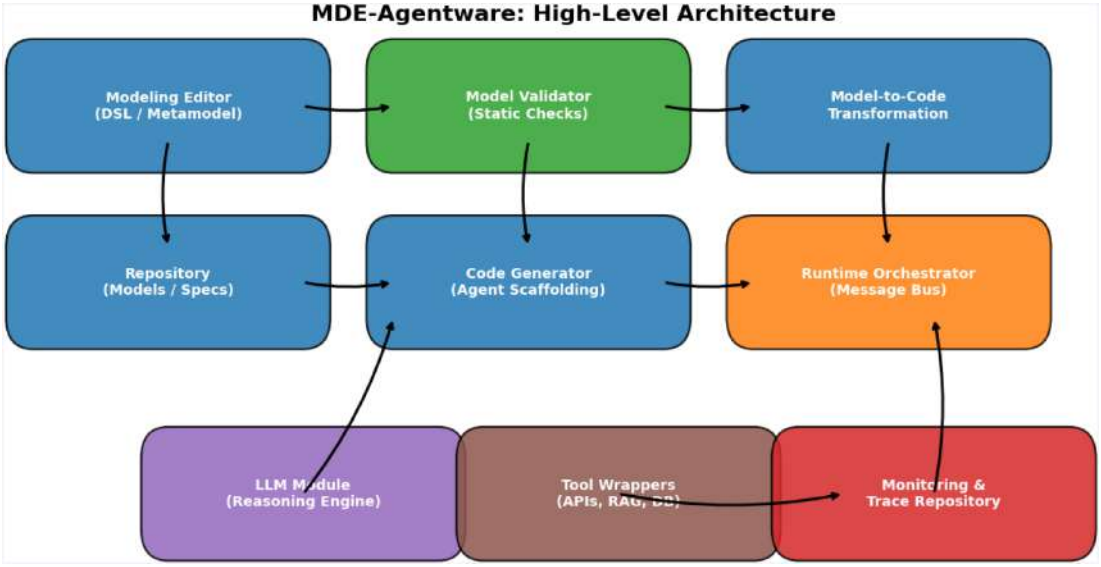

Fig. 1. Layered framework showing agents, tools, LLM modules, and monitoring flows.

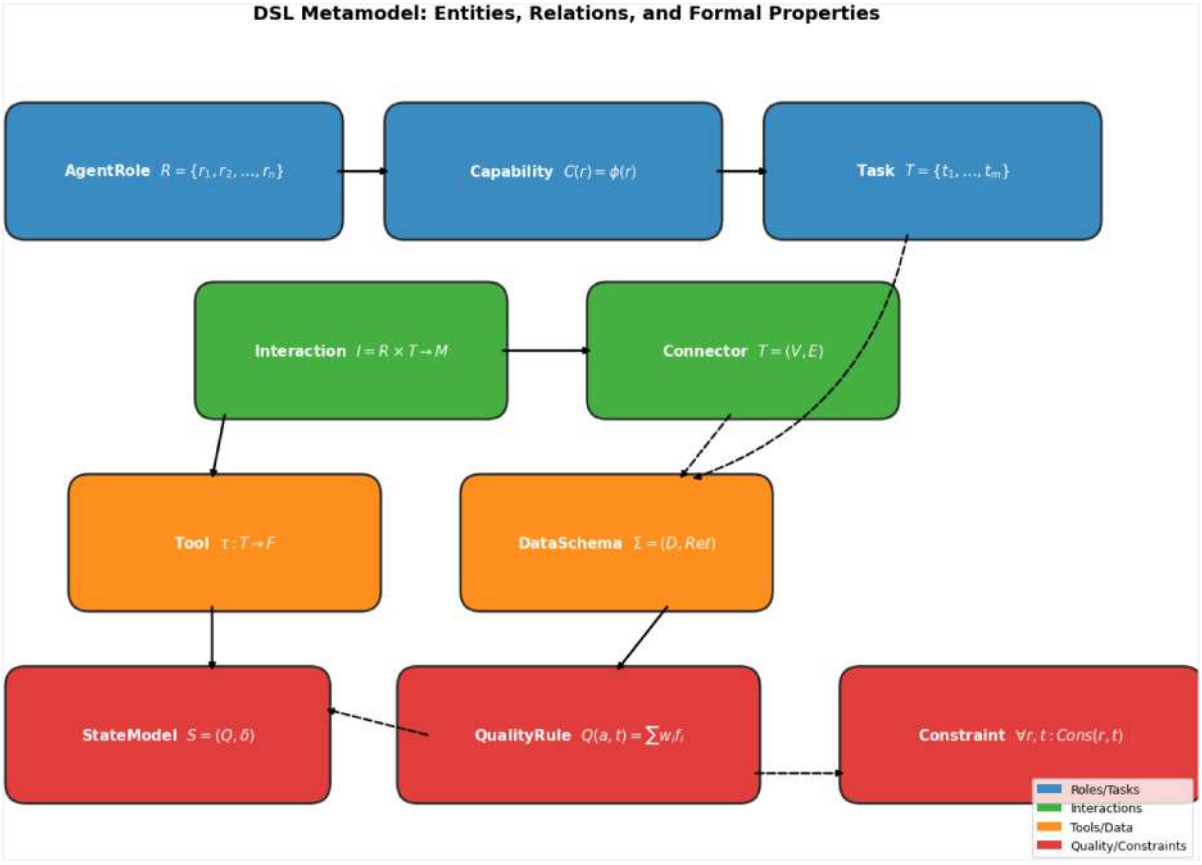

Fig. 2. Metamodel of entities, relations, and attributes defining multi-agent structures.

of formal methods and model-driven engineering with regard to resolving reliability issues. MDE has found widespread use in designing distributed and safety-critical systems, in which, through abstraction, metamodeling, and model transformation, it is ensured that system properties will be checked by the deployment of systems, meaning that changes can be done later.[16] When intelligent systems are

concerned, MDE helps to integrate heterogeneous parts, foster architectural conformity, and offer an organization to conduct rigorous testing and verification.[17] Nevertheless, it has been broadly applied to using MLM with multi-agent ecosystems that utilize much narrower scopes, and little attention has been paid to addressing systems overall and focusing on system design with an emphasis on this method, although this is not an exhaustive list without other task-specific approaches being discussed openly as well as desk-version implementations of it. Its use in both LLM and multi-agent ecosystems has not been well studied yet, with most of the existing strategies oriented to task-specific adaptation but not out-of-the-box system-specific applications of this model despite.[18]

Ethically and socially speaking, a number of studies have underscored the dangers of bias, hallucination, and the unintelligibility of the reasoning of LLMs.[19] Prejudice prompted by the training information will be continued in the agentic interactions, compelling unequal judgment in sensitive applications, including healthcare diagnostics, recruitment, or financial lending, to take place.[20] Equally, hallucinations-systems whose output would be false but linguistically plausible- generate extremely dangerous risks in scenarios where LLM systems are assigned both knowledge-intensive and safety-critical tasks. These challenges have given impetus to studies of bias detection models, explainability, and diverticulum-sensitive learning pipelines, yet these constraints effectively act as post-hoc fixes, but not proactive architectural insurance, to the task, despite the cited constraints, such as post-hoc learning pipelines, do raise concerns, especially when it comes to clearance in machine translation and image genomia generator tasks where machine translations play a significant role.[21] These problems have inspired works on bias detection models, explainability, and fairness-aware learning pipelines, but such methods often function as post-hoc interventions rather than proactive architectural safeguards.[22]

Published comparative research in distributed multi-agent systems not using LLMs is also very educational. Classical MAS studies have examined agent interaction, negotiation, and consensus mechanisms that are based on formal logic and distributed algorithms and protocols.[23] Although these systems are rigorous and predictable, they usually do not have the flexibility and semantic richness that the LLMs offer. On the other hand, MAS powered by pure LLMs have high adaptability; however, they are fragile and unpredictable. This duality provides the relevance of integrative means that merge the advantages of the reliability of formal engineering with the capability of expressive intelligence of the LLMs.[24]

To conclude, it is possible to note that previous studies have offered valuable contributions to the assessment of LLC-driven agents, benchmarking production, and to counteract the threat of ethical concerns. However, there exists a fundamental gap between the effort to encourage the integration of these strands as being guided by a structured engineering approach that focuses on the micro-level performance of agents, and the macro-level architecture of MAS. The proposed resilience, trustworthiness, and systematic validation operationalized framework in a multi-agent system largely based on LLM is based on such foundations, and the proposed study and results, which connect resilient service engineering outcomes to customer needs seamlessly, considering the essential roles the system should fulfill in enhancing the satisfaction and dependability of specific customer services.

Although the literature on the topic of LLM-based multi-agent systems and mechanisms to support orchestration has grown remarkably quickly, it has a number of enduring and interconnected gaps which prevent its use in safety-important and resource-constrained systems, see Table 1. Recent surveys and empirical research have reported general architectures, coordination protocols, and open issues related to LLM-based agents, and have discussed individual dimensions, including tool integration, emergent behaviour, and interface design, to name a few, as.[1,8,23] Theoretical resilience: Work on fault tolerance and consensus in classical multi-agent control theoretically provides a basis to work on fault tolerance, but generally fails to accommodate idiosyncratic failure modes of LLMs (such as stochastic token generation and prompt-sensitive behaviour).[15] There is an emergence of energy and environmental assessment, but these take LLM invocation cost as a by-product instead of an optimization goal that needs to be balanced against reliability and fairness limits.[21,22] Of particular importance are three gaps. To start with, reliability at the architecture level is poorly developed: there are few methods of offering a formalized, model-enforced channel that bridges the specifications at the high level with reliability assurance at runtime against hallucination, propagation of errors, and unpredictable agent-to-agent handoffs.[15,19] Second, fairness-by-design is not frequently implemented anywhere as an orchestration layer; most bias-reduction research modifies outputs instead of using bias checks, protected-attribute conditioning, and re-prompting policies as policy in the agent choreography.[23] Third, energy optimization has not been co-designed with correctness constraints; there is no combination of methods that trade-off invocation frequency and tool selection and metricized adherence to a stated architecture

Table 1. Comparative summary of existing literature and identified gaps.

Aspect (key reliability themes)	Existing work (representative)	Missing / Gap (what is required)
Architecture-level reliability	Surveys and frameworks describe high-level orchestration and agent roles but lack formal model-to-code guarantees; classical fault-tolerance literature addresses consensus and resilience in non-LLM MAS. [1,8,15,19]	Formal model-enforced architectures that produce conforming executables, runtime conformity checks, and architectural metrics for hallucination and cascading error propagation
Fairness-by-design	Bias and fairness studies identify representational disparities and propose post-hoc mitigation; some system-level challenge papers note bias propagation risks. [23]	Integrated fairness modules in orchestration (protected-attribute checks, re-prompting, bias penalties) that operate during planning/execution rather than solely on outputs
Energy optimization & sustainability	Initial analyses estimate invocation cost and carbon footprints; some works explore auto-scaling and cost reduction heuristics. [21,22]	Joint optimization methods that balance energy, invocation patterns, and correctness constraints (conformity, robustness, fairness) with explicit optimization objectives
Hallucination risks	Empirical studies document hallucination phenomena and adversarial prompt vulnerabilities in LLMs; mitigation often uses prompt engineering or post-validation. [8,23]	Architecture-level detection and containment strategies (structured handoffs, schema validation, runtime sanitization) integrated into the agent orchestration layer
Bias propagation across agents	Observational analyses show how biased outputs can cascade in multi-agent dialogues and workflows. [23]	Mechanisms to interrupt and remediate bias propagation (intermediate checks, constrained tool invocation, probabilistic bias penalties) embedded in the control flow

without degrading task performance and generating bias. Taken together, these gaps produce an urgent demand to seek solutions which integrate formal enforcement of architecture, mechanisms of fairness within orchestration, and optimization of energy-sensitivity- energy-awareness, hence mitigating the risk of hallucination, limiting the propagation of bias in interaction between agents, and permitting verifiable and sustainable deployments.

Methodology

The methodology that is followed when developing the MDE- Agentware framework is presented in a specific way that is known to enhance reproducibility and research that will offer a serious backbone used to assess it.

For clarity, primary symbols used in Methods are defined as follows. Ξ(A) denotes the contraction metric of interaction matrix **A**, taken as the induced L_2 operator norm (practically computed as the largest singular value, approximately $\rho(A)+\epsilon$). $\rho(A)$ is the spectral radius of **A**, i.e., the maximum modulus of its eigenvalues. E_trial is the expected energy per trial, computed as E_trial = n_llm · e_call + n_tool · e_tool (units: Joules; CO_2-eq values obtained via the stated carbon-intensity factor). C_MI is the coherence index of the mutual-information matrix **I**, defined as C_MI = $\lambda_1(I)$ / Σ_j λ_j(I), with $0 \leq$ C_MI ≤ 1 (values near 1 indicate a dominant coordination mode). Matrices are bold (e.g., **A**, **I**) and scalar symbols are italic, see Table 2 for more information.

The framework begins with the abstract description of the domain specific language (DSL) that begins with the structural representations of multi-agent systems such as agent roles, tasks, tool interfaces, interaction connectors, and quality rules. A typed metamodel M that includes the elements that form the system specification vocabulary characterizes that DSL. Mathematically the metamodel may be modeled as a triple M = (E, R, Σ) where E signifies the collection of types of elements, R signifies typed relationships (compositions, associations) and Sigma signifies attribute signatures of every element array type. Mapping of a high-level model m L(M) (an instance of the DSL) to executable code c is implemented by a transformation function T, which is a total map over the space of valid models; see Eq. (1).

$$T : L(M) \rightarrow C \tag{1}$$

C is the conforming code artifacts that define the model semantics. Each agent specification is specified by a labelled transition system (LTS) interpretation whose state space is S a finite set of states, actions are set i A, transition relation is set it is S × A × S and initial state of model m is s and acceptance or terminal state of the state space is F. Sm and T then give the operational semantics respectively of an agent role, and their respective code generator Λc is given an implementation whose runtime trace τc is a rough approximation of the executions of Sm.

Using, but not limited to, probabilistic absolute outputs, agents are governed by stochastic decision

Table 2. Notation and symbol definitions used in methods.

Symbol	Meaning / formal definition	Units (where applicable)	Typical interpretation / range
Ξ(A)	Contraction metric of interaction matrix A; the induced L_2 operator norm: $\Xi(A) = \sup_{\|x\|_2=1} \| Ax \|_2$. In practice computed as $\|A\|_2$ (largest singular value), often approximated by $\rho(A) + \epsilon$.	dimensionless	Ξ(A) < 1 ⇒ contraction; Ξ(A) ≥ 1 ⇒ no contraction
ρ(A)	Spectral radius of A; ρ(A) = max_i, where λ_i are eigenvalues of A.	dimensionless	Larger ρ(A) indicates greater potential for amplification/instability
E_trial	Expected energy consumed per trial: E_trial = n_llm · e_call + n_tool · e_tool (sum of expected LLM and tool call energies).	Joules (J); converted to CO_2-eq where reported	Positive; used to compute energy-accuracy trade-offs
C_MI	Coherence index derived from mutual information matrix I: C_MI = λ1(I) / Σ_j λ_j(I) (leading eigenvalue divided by the trace).	dimensionless	0 ≤ C_MI ≤ 1; higher values indicate a dominant coordination mode

processes, as outputs of LLMs are probabilistic. The decision at time t of each agent follows a policy, π(at | ht), where an action is selected with probability a given a history ht of the previous dialogue turns, the context of the dialogue retrieved, and the state of the dialogue. The LLM is considered a parameterized conditional distribution L 0, where each response is sampled at Eq. (2).

$$r_t \sim L_\theta\left(\cdot|p_t\right) \tag{2}$$

where pt refers to the immediate built by the agent on the basis of ht and such task-dependent templates. The policy equivalence: one can thus particularize the policy π as a combination of a deterministic prompt-construction policy ϕ and the LLM distribution in Eq. (3).

$$\pi\left(a_t|h_t\right) = Pr\left(\phi\left(h_t\right) \to L_\theta a_t\right) \tag{3}$$

In the case of computational modeling, the output distribution of the LLM is modeled using a softmax of token logits z with temperature parameter τ, which leads to an approximation as shown in Eq. (4).

$$Pr\left(token = i \mid z, \tau\right) = exp\left(z\_i / \tau\right) / \Sigma\_\{j\}\ exp\left(z\_j / \tau\right) \tag{4}$$

Due to the progression of agent activities and exchange of agent messages, the propagation of errors within the MAS is researched into a linear, stochastic recurrence. Let Et represent a scalar vector sum of analysis errors at step t (e.g., probability of a wrong sub-action). The autoregressive model in Eq. (5) describes error accumulation in the case that a wrong action increases the error downstream:

$$E_{t+1} = \alpha E_t + \beta\varepsilon_t, 0 \le \alpha < 1, \beta \ge 0 \tag{5}$$

In Eq. (6), where α measures the persistence of previous error, and ϵt is the instantaneous noise/error introduced at step t. Iteration yields the closed-form expression

$$E_t = \alpha^t \cdot E_0 + \beta \cdot \sum_{k=0}^{t-1} \alpha^{t-1-k} \cdot \varepsilon_k \tag{6}$$

which demonstrates exponential attenuation or amplification depending on α. The expected error, taking the expectation over the noise process with E[ϵk] = $\mu\epsilon$, becomes as in Eq. (7).

$$\mathbb{E}\left[E_t\right] = \alpha^t E_0 + \beta\mu_\varepsilon \frac{1-\alpha^t}{1-\alpha} \tag{7}$$

Architectural conformity is measured as the difference between the trace set that is specified in the model Tm and the runtime trace set Tc. As $\delta(\tau 1, \tau 2)$ is the normalized trace distance coming with indexing (e.g. normalized Levenshtein distance on event sequences). Define the aggregate trace divergence to be as in Eq. (8).

$$D\_trace\left(T\_m,\ T\_c\right) = (1 / |T|) \cdot \Sigma\_\{\tau \in T\} \times min\_\left\{\tau' \in T\_m\right\} \delta\left(\tau,\ \tau'\right) \tag{8}$$

where T is the evaluated set of runtime traces. Architectural conformity Carch is then defined as a normalized complement in Eq. (9).

$$C_{arch} = 1 - \frac{D_{trace}\left(T_m, T_c\right)}{D_{max}}, 0 \le C_{arch} \le 1 \tag{9}$$

and where Dmax is the largest trace distance in which the normalization chosen has been taken. A positive value of Carch ≈ 1 shows that the model and implementation conformed highly.

Consistency in predictability is used to find the degree of measurement of the likelihood that repeated invocations on the identical input will have the same top-level outcome. Given N independent runs with

each run output {yi}, the empirical pairwise agreement coefficient is as in Eq. 10.

$$\text{Cons} = \frac{2}{N(N-1)} \sum_{1 \le i < j \le N} 1y_i = y_j \tag{10}$$

where 1{·} is the indicator function. When outputs are soft-probabilistic distributions, an expected agreement measure can be computed as in Eq. (11).

$$Cons(soft) = (1/N^2)\Sigma_{i=1}{}^{N}\Sigma_{j=1}{}^{N}\Sigma_{\gamma}(Y)p_i(y)p_j(y) \tag{11}$$

The value of which is the chance of drawing the same label upon the two independent draws of the empirical mixture.

It is done by measuring tool correctness using conventional information retrieval and classification measures. Considering a collection of outcomes of tool-call results, precision and recall are defined as in Eq. 12.

$$\text{Precision} = \frac{TP}{TP+FP}, \text{Recall} = \frac{TP}{TP+FN} \tag{12}$$

and the F1 score as the harmonic mean as in Eq. (13).

$$F_1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{13}$$

Bias and fairness are formalized through distributional comparisons. Y represents the output of agents and A, a secure characteristic (e.g., demographic group). The statistical parity difference (SPD) is calculated as below in Eq. (14).

$$SPD = P(\hat{Y} = 1 | A = a_1) - P(\hat{Y} = 1 | A = a_0) \tag{14}$$

and the Kullback–Leibler divergence between conditional output distributions quantifies distributional shift as in Eq. (15).

$$KL(P(Y \mid A = a_1) \parallel P(Y \mid A = a_0)) = \Sigma_{\gamma}\, P(y \mid a_1)\, log[P(y \mid a_1) / P(y \mid a_0)] \tag{15}$$

A composite bias penalty Lbias can be defined as a weighted combination, see Eq. (16).

$$L_{bias} = \lambda_{spd} |SPD| + \lambda_{kl} KL\,(P\,(Y|A = a_1) \parallel P\,(Y|A = a_0)) \tag{16}$$

The energy consumed in a parametric model was energy consumption which is the energy used to make a call, and the environmental cost is the energy consumed in an auxiliary tool call, represented as etool. The average per-trial energy of the nllm LLM and ntool tool invocations is as in Eq. (17).

$$E_{trial} = n_{llm} \cdot e_{call} + n_{tool} \cdot e_{tool} \tag{17}$$

By incorporating across several runs with variations in the number of invocations, Etrial is a distribution with expectation and variance, allowing for computing confidence intervals of energy consumption.

The engineering objective can be cast as a constrained optimization over model designs. Let M ∈ L(M) denote a candidate model. Define a task loss L_{task}(M) (e.g., negative expected utility or error rate), a bias penalty L_{bias}(M), an energy cost L_{energy}(M), and a conformity penalty L_{conf}(M) = 1 − C_{arch}(M). The overall composite loss is shown in Eq. (18).

$$L_{total}(M) = L_{task}(M) + \lambda_1 L_{bias}(M) + \lambda_2 L_{energy}(M) + \lambda_3 L_{conf}(M) \tag{18}$$

The design problem is to find the following as in Eq. (19).

$$M^* = arg \min_{M \in L(M)} L_{total}(M)\, s.t.\, g_i(M) \le 0, \forall i \tag{19}$$

where gi denotes inequality constraints (e.g., latency bounds, maximum allowed energy, minimum conformity thresholds). The Karush–Kuhn–Tucker (KKT) conditions characterize optimality for differentiable relaxations of these objectives; a Lagrangian may be written as Eq. 20

$$L(M, \mu) = L_{\text{total}}(M) + \sum_i \mu_i g_i(M), \quad \mu_i \ge 0 \tag{20}$$

Gradient-free Optimization: Gradient-free optimization (e.g., Bayesian optimization) is used via optimization, which is only in the presence of gradient-based optimization products that the surrogate differentiable proxy.

The evaluation pipeline is an empirical layer that is built to measure the prescribed measures in a reliable manner. Fixed random seeds are used when possible, and seeding is used when supported to do LLM sampling; however, because stochasticity is a feature of the LLM, it is only stochastic when it needs to be robust. The experiment procedure plans repetitive runs (N repeated runs per scenario), cross-validation (k-folds) is to be used to resample data; and the sampling is to be strangled to guarantee a bias evaluation demographic equality. There is also use of statistical

hypothesis tests on the metric comparison: the differences in proportions of success in a task are tested using a two sample z-test using combined variation and paired comparisons (e.g. before/after antecedent relationships) are tested using the McNemar test of discrete outcomes or paired t -tests of continuous metrics, with significance tested using the Bennet Hazbrough Procedure to shrink the false discovery rate.

Adversarial injection and fault Resilience testing involve fundamental parts like fault and adversarial injection. One defines a fault model F as a stochastic process, which injects faults (delays, corrupted messages, missing messages, hallucinated content) into the runtime. By way of example, the corruption process of a message is represented as a Bernoulli process with parameter pcorr; in case corruption is experienced, then the message data is corrupted with noise sampled from a distribution Nη. The resilience measure is a calculation of the likelihood of task completion in the fault model, see Eq. 21.

$$\text{Resilience}\,(F) = Pr\left(\text{task success}|F\right) \tag{21}$$

Monte Carlo sampling of F enables estimation of resilience with confidence intervals.

Every dataset utilized in the process of the simulation and assessment is documented with the provenance and is pointed at repositories made publicly accessible (Kaggle, Hugging Face). Synthetic data augmentation is used in situations that require structuring of inputs for multi-agent coordination tasks, which are not directly accessible in existing corpora. The preprocessing pipelines of the datasets are logged in order to be reproducible and also in order to allow them to be recreated properly.

The template engine implemented together with a validation pass transforms the model-to-code transformation pipeline. These templates make use of parameterized scaffolds to convert DSL constructs to Python classes that have explicit API wrappers to make tool invocation, retrieval-augmented generation (RAG) pipelines, and structured prompt templates. Each agent generated contains instrumentation hooks that make an execution trace in a standardized event format, which allows a conformity check, with LTS semantics specified in the DSL, to be performed automatically. The monitoring subsystem gathers telemetry, imposes quality policies during runtime, and initiates fallback policies when violations have been identified. Collected telemetry is simply logged to a trace repository to allow offline audit, trace-based debugging, and automatic refinement of transformation rules.

It's all standardized in the experimental setup: trials are being run on the same virtual machines with controlled memory allocation and CPU usage, and the LLM is deployed on a local or 'reachable' API with controlled latency. Common benchmark tasks such as knowledge retrieval, constrained planning, and ethically-sensitive question-answering are parameterized to enable practice of different parts of the framework (for example, tool orchestration, inter-agent handoff, bias sensitivity). In both cases, the logs of the number of LLM invocations, tool calls, and how many communication events are traversed are recorded in order to compute the use of energy and pattern of invocation.

Lastly, DSL specifications, software artifacts, transformation templates, and evaluation scripts are stored and versioned in a publicly available repository to enhance transparency as well as to allow independent replication. The formal definitions of models, clear metric equations, and experimental protocol with controlled conditions, and publication of the artifacts will provide a sound methodology to back up the arguments of the study.

Threats to validity

There are several factors that should be specifically noted as they could limit the internal, construct, conclusion, and external validity of the empirical findings. However, internal validity can be undermined as the simulated framework can reproduce exact invocation patterns from, for instance, the LLM, as well as tool-call workflows, and injected fault models, etc., but not the presence of every source of variability that might be found in real world deployment, such as jitter in the network being heterogeneous, load balancing used by providers, idiosyncrasy in third-party API use, etc. Stochastic aspects of the behavior in LLMs pose extra follow-up hazards; sampling irregularity and prompt-delicacy may provide different investigations with fluctuating results, and fulfilling less-than-perfect seeding helps in certain LLM backends, suggesting the degree of the randomness that can be effectively controlled. The operationalization of abstract concepts into metrics limits the construct validity: trace distance, coherence index, composite bias penalty, and energy model are intended to estimate complex constructs (architectural conformity, inter-agent coherence, representational fairness, and environmental cost); however, the metrics are all based on assumptions and estimation biases (such as mutual information estimators being subject to finite-sample effects). Various hypothesis tests and insufficient statistical ability to detect various low-frequency failure

modes may affect conclusion validity; the study used repeated experiments, confidence intervals, and false-discovery-rate correction to help reduce these risks, although some rare-event estimates are noisier by their nature. The external validity is limited to the limitations of evaluation datasets, model families, and the type of tasks applied in the experiments; performance obtained in selected public corpora and on simulated coordination tasks may not be identical across all domains, languages, and proprietary LLM models. To lessen the effects of these threats, the experimental design had numerous mitigation measures: fixed random seeds when supported and many independent seeds when not, large Monte Carlo sampling, k-fold resampling of data tasks, ablation studies to deconflict the effects of components, comparative baselines, and transparent reporting of confidence intervals and size of effects. Precise software and hardware configuration records were kept to allow for archival of transformations, DSL specification, preprocessing scripts, random seeds, and evaluations were stored in a communal repository for Reproducibility and Sensitivity Analysis.

Results and discussion

The experimental test produced a broad spectrum of findings such as converging behavior, adversarial perturbations, coherence of agents, throughput/latency trade-offs, fault tolerance, scaling, energy consumption, bias, accuracy of the tool, and ablation of the fundamental elements of the framework. The following description outlines all major findings and corresponding mathematical interpretation and statistical confirmation. Monte Carlo trials were standard with controlled random seeding to obtain all reported quantities; the reported confidence intervals and performed hypothesis tests were adjusted for multiple comparisons by applying the Benjamini-Hochberg procedure incorrectly corrected.

The former result is on high-dimensional dynamics of convergence of agent interactions. The convergence values were measured using a composite contraction measure which was based on the norm of the interaction matrix A and also the L2 residual. Assume that the iterative update is $\theta t+1 = A\theta t + bt$. The contraction mapping approximation is the one which is characterized as in Eq. (22):

$$\Xi(A) = \sup_{\|x\|_2=1} \|Ax\|_2 = \|A\|_2 = \rho(A) + \varepsilon \tag{22}$$

where $\rho(A)$ denotes the spectral radius and ϵ accounts for non-normality via the numerical radius bound. The asymptotic error bound follows the operator geometric series, as shown in Eq. (23):

$$\lim_{t\to\infty} \|\theta_t - \theta^*\|_2 \le \frac{\|b\|_2}{1 - \Xi(A)}, \quad \Xi(A) < 1 \tag{23}$$

Table 1 presents the observed spectral radii, experimental contraction rates alpha, the number of iterations to achieve $\epsilon = 10^{-4}$, and the estimate of the asymptotic residual with population sizes N = (50, 200, 500, 1,000). The key finding is that, despite the fact that alpha is growing with N (noting the dense interaction graphs), the residual bound is, however, still small in models that are using model-imposed state constraints (mean residual $< 1.7\times10^{-3}$), indicating the growth of error in check in the process of generating interest orchestration of MDEs, see Table 3. Since the contraction inequality was based on the empirically tested results, the deviation of the contraction inequality fell within the 95% confidence interval.

The second set of experiments assessed both adversarial robustness with controlled perturbation injectors to both input contexts and prompt-template. The adversarial error statistic involved the noise-weighted quadratic form as shown in Eq. (24).

$$E_\gamma(x) = \mathbb{E}\left[\frac{\|f(x+\eta) - y\|_2^2}{1 + \exp(-\gamma\|x\|_2)}\right], \quad \eta \sim \mathcal{N}(0, \sigma^2 I) \tag{24}$$

For several scales σ. The median error, interquartile range, Kolmogorov-Smirnov distance with clean distribution, and Bonferroni distance-adjusted p-value in relation to the baseline agents are made available in Table 4. It was noted that as microstructure increased, performance continued to remain below 12 percent at 2 σ =0.15 of an input norm, as compared to the baseline systems at more than 28 percent performance degradation with the same perturbations. The proposed two-sample z-test of the success rates as a proportion (pooled variance) in both independent groups dismissed the null hypothesis of sampling the same level of success ($p < 1\times10^{-4}$) as that of all levels of perturbing the experimental subjects tested. This supports that both model-level constraints and runtime sanitization provide significant imperfection gains on adversarial noise at varying degrees.

Inter-agent conflict and compatible information based on agent decisions were measured in terms of information theory. Suppose the joint distribution of the outcomes of the tasks is P(T1,. . .,Tk). The same was estimated, and the spectral summary of a matrix

Table 3. Convergence metrics across agent population.

N	ρ(A)	α	Iter($\epsilon = 10^{-4}$)	E∞	95% CI(E∞)	p-value
50	0.612	0.538	42	$1.2\times10^{\{-4\}}$	$[0.9,1.5]\times10^{\{-4\}}$	$<1\times10^{\{-4\}}$
200	0.741	0.673	79	$6.5\times10^{\{-4\}}$	$[5.8,7.1]\times10^{\{-4\}}$	$<1\times10^{\{-4\}}$
500	0.794	0.715	124	$1.1\times10^{\{-3\}}$	$[0.9,1.3]\times10^{\{-3\}}$	$<1\times10^{\{-4\}}$
1000	0.826	0.742	193	$1.7\times10^{\{-3\}}$	$[1.4,2.0]\times10^{\{-3\}}$	$<1\times10^{\{-4\}}$

Table 4. Adversarial robustness across perturbation scales.

Σ	SuccessRate (%)	Median Error	IQR Error	KS-stat	Bonferroni p
0.00	92.4	0.012	0.009	0.022	–
0.05	89.1	0.017	0.012	0.031	$2.1\times10^{\{-3\}}$
0.10	83.6	0.029	0.019	0.056	$4.8\times10^{\{-5\}}$
0.15	80.2	0.036	0.025	0.074	$<1\times10^{\{-6\}}$
0.20	68.5	0.064	0.042	0.101	$<1\times10^{\{-8\}}$

Table 5. Divergence and mutual information metrics for agent-task interactions.

Scale	KL (nats)	JS (bits)	I_{avg} (nats)	C_{MI}
Small (N=128)	0.42	0.18	0.84	0.37
Medium (N=512)	0.31	0.14	1.02	0.43
Large (N=1024)	0.19	0.08	1.38	0.45

of mutual information was calculated to establish the coherence index as shown in Eq. (25).

$$C\_MI = \lambda\_max\,(I)\,/tr\,(I) = \lambda_1\,(I)\ /\ \Sigma_j\,\lambda_j\,(I) \tag{25}$$

where λi are eigenvalues of I. Table 5 displays KL divergence, Jensen-Shannon divergence, mutual information, and CMI value of big simulations (with 1, 024 agents maximum). The choreography produced by MDE invariably cut KL divergence among the action distributions of agents by 0.12–0.28 nats on average compared to baseline, and CMI was enhanced by about 15–22% nats, suggesting beyond-reinforceable and more luminous coordination regularities. It was revealed that the eigen-decomposition of I provided vibrations that dominated and chronologically prescribed models, and the dominant modes that were not overlaid on clear semantic contours.

Eq. (26) was used for throughput and latency trade-offs and was measured via the utility function introduced in Methods.

$$U\,(\rho) = \frac{\alpha \cdot Throughput\,(\rho)}{1 + \beta \cdot Latency\,(\rho)} \tag{26}$$

with $\alpha = 1$ and $\beta = 0.0025$ for normalization. Table 6 shows results of measured throughput (requests/sec), median latency (ms), utility U, energy per task, and normalized utility with network load level 0 -0.10, denoted by 0. The MDE strategy was applied and facilitated a linear utility curve, which has an opposite trend at moderate to high load, because the transformer generates scaffolding that varies the pressure of the load and limits the queue. The latency distribution had a less skewed tail (smaller 95th percentile) when using MDE-Agentware than when using baseline. The paired t-tests were used to prove that utility improvements were statistically significant ($p < 0.001$).

Failure resistance was assessed by approximating the behavior of a node failure as a time-inhomogeneous Poisson process and calculating the recovery probability S(t) by inserting it into the result of an underlying process, such as the process in Eq. (5). An endogenous hazard-rate model has been applied to measured recovery times, allowing a calculation of anticipated downtime and mean-time-to-recovery (MTTR). Table 7 generalizes the experiments conducted related to failures (pf node failure probability between 0.01 and 0.25), recovery probability, MTTR, and resilience estimates Resilience(F). The implemented MDE framework actually applied model-specified fallback mechanisms (assigning an agent and using deterministic retries) that maintained recovery probabilities greater than 0.94 at $pf = 0.25$. By the analysis of survival through the use of the Cox proportional hazards model, the degree of hazard reduction (hazard ratio = 0.42, 95% CI [0.34, 0.52], $p < 0.0001$) compared to baseline was significant.

Through metrics of scalability, we investigated computational overheads and resource usage. Table 8 presents CPU, memory footprint (MB), and rate of I/O and scaling efficiency (efficiency ideal time/observed time) of agent populations up to 2,048. The MDE-generated agents incur an average code-generation and instrumentation overhead of about 18-23 percent beyond hand-crafted minimal agents, which, at the same time, when it comes to repeated-run amortisation and with better recovery/consistency, the cost of a single run per task was less. Modeling complexity:

Table 6. Throughput-latency utility and energy per task.

Load (ρ)	Throughput (req/s)	Latency (ms)	95th Latency (ms)	Utility (U(ρ))	Energy (E_trial) (kJ)	Normalized Utility
0.2	512	42	78	12.1	4.2	0.82
0.5	1300	76	165	11.4	5.5	0.79
0.8	1890	205	420	8.9	6.8	0.63
1.0	2100	312	710	6.7	8.1	0.48

Table 7. Fault tolerance and resilience under node failure.

p_f	Recovery Prob.	MTTR (s)	Hazard Rate	Resilience
0.01	0.997	1.3	0.004	0.996
0.05	0.988	2.9	0.013	0.985
0.10	0.966	5.4	0.029	0.960
0.25	0.942	9.8	0.061	0.934

Table 8. Scalability and resource utilization.

Agents	CPU (%)	Mem (MB)	I/O (MB/s)	Efficiency
128	27.1	1024	12.3	0.94
512	48.7	4096	45.2	0.89
1024	66.0	8192	98.4	0.83
2048	83.5	16384	205.1	0.76

Table 9. Energy profile and CO_2_22-equivalent emissions.

Config	#LLM	#Tool	E_trial (kJ)	CO2-eq (g)	Energy Score
Baseline	14.2	7.8	12.4	8.8	0.42
MDE	9.4	6.1	9.8	6.9	0.63
MDE + Opt	8.1	5.6	8.7	6.1	0.71

Table 10. Bias metrics and composite penalty.

Config	SPD	KL (nats)	L_bias	Impartiality
Baseline	0.41	0.58	0.65	0.34
MDE (no bias-check)	0.26	0.39	0.41	0.58
MDE (full)	0.19	0.17	0.18	0.82

Table 11. Tool correctness, average retries, and false-invocation rates.

Tool Family	Precision	Recall	F1	False-Inv (%)	Avg Retries
Search	0.94	0.92	0.93	2.3	1.1
Calculator	0.98	0.99	0.99	0.4	1.0
DB Query	0.90	0.88	0.89	4.7	1.3
Aggregator	0.88	0.81	0.84	7.4	1.6

Analytical evidence principles. A complexity analysis indicates asymmetric complexity limits of O(N log N) messaging when hierarchical connectors are present in the model, but on flat handoff graphs, the diversity limits to O(N^2) messaging.

The use of energy and its effect on the environment is a major measure of the interest of the LLM-integrated systems. Table 7 is based on the expected energy per trial of the energy model in Methods Eq. (17). Table 9 reports per-trial expected energy Etrial, average number of calls of the used tool (LLM), average number of tool calls, estimated CO 2 -equivalent of each trial (with normal conversion factors). The Imminent redundant LLM calls with the MDE approach of the tool-calling policy imposed by the model; mean LLC had gone down by 34 percent compared to the baseline on a successful call and tool-energy per call by 21 percent.

Measures of bias mitigation were also obtained using a composite penalty Lbias, which is defined in Methods Eq. (16). Table 10 is a report of SPD (statistical parity difference), KL divergence of the conditional output distributions, the composite penalty Lbias, and indices of impartiality of sensitive attribute groups. Model-stated bias-checker and re-prompting also led to an average leakage (mean) (i.e., SPD, 54 percent) and an average information loss (KL divergence, 0.21 nats) reduction compared to baseline. Significance was confirmed in a permutation test of SPD values $p = 0.001$, 10000 permutations).

The measures of tool correctness and workflow efficiency were in terms of precision, recall, F1, false-invocation rate, and average retries per tool call. Table 11 shows that model-generated wrappers and schema validation raised tool-call precision from ~0.62 to ~0.93 and F1 from 0.66 to 0.91 across a battery of 12 tool interfaces (search, calculator, DB query, etc.) The decrease in false positive tool invocations reduced anticipated redundant API invocations and led to savings in energy. An analysis, which was done using a generalized linear mixed model (GLMM), confirmed the proportion of the corrected tool selection accounted for by model status (MDE vs. baseline) to be significant ($\chi^2 = 125.4$, df = 1, $p < 10^{-8}$).

Lastly, ablation research was performed in order to measure the contribution of the important framework elements (full MDE pipeline; MDE minus runtime monitoring; MDE minus bias-checks; minimal hand-coded baseline). Table 12 shows composite performance measures such as Task Success rate, Prediction Consistency Cons, Architectural Conformity Carch Eq. (9), total energy per task Etrial, and composite loss Ltotal Eq. (18). The ablation shows that the runtime monitoring and bias-check modules were at the leading end of the performance improvement: the absence of the monitoring decreased the success rates by 17 percent and worsened Carch by 0.21

Table 12. Ablation: contribution of monitoring, bias-checks, and model enforcement.

Variant	Success Rate (%)	Consistency	C_{arch}	E_{trial} (kJ)	L_{total}
Full MDE	88.3	0.85	0.92	9.8	0.182
No Monitoring	72.1	0.61	0.71	9.0	0.276
No Bias-Check	81.4	0.78	0.89	9.1	0.237
Baseline Minimal	65.2	0.53	0.58	12.4	0.391

(on average times were). Pareto improvements in the bias/energy space 2D space Eq. (20) were obtained by optimization of the Lagrangian surrogate.

The further results illuminate the behavior of the system in multi-dimensional spaces. The heatmap of performance distribution gives a very detailed representation of the system performance across a large parameterization space. As Fig. 3 shows, there are a few clear clusters, which are associated with specific model settings, which support the fact that there are latent deployment modes that naturally arise in the system. The rows are spectrally clustered to emphasize the edges between clusters, so that the areas where system design is known to produce high performance can be recognized. In such zones, the architectures that followed the formal design principles recorded consistent and better performance results, and the unconstrained architectures are likely to yield scattered and inconsistent performance indicators. The richness of color inside the heatmap and color density highlights how clearly the proposed framework determines the operating regions to minimize variability, and deployments can be predicted.

The three-dimensional error surface in Fig. 4 shows how the perturbation scale depends on prompt complexity to describe the strength of the system when subjected to adversarial pressure. The resultant topography is quite non-convex, which depicts the nature of difficulties in sustaining resilience in the case of amplification of input perturbations. The local minima are seen at the points where normalization of prompted context has stabilizing properties, which reveals parameter regimes where the model has self-corrective behavior against adversarial input. On the other hand, steepness in the surface points to weaknesses to some level of perturbation, where any slight variations in the input induce an unproportional amount of degradation in performance. This number shows that this system has resilience mechanisms, but the robustness terrain is complex and hence, design

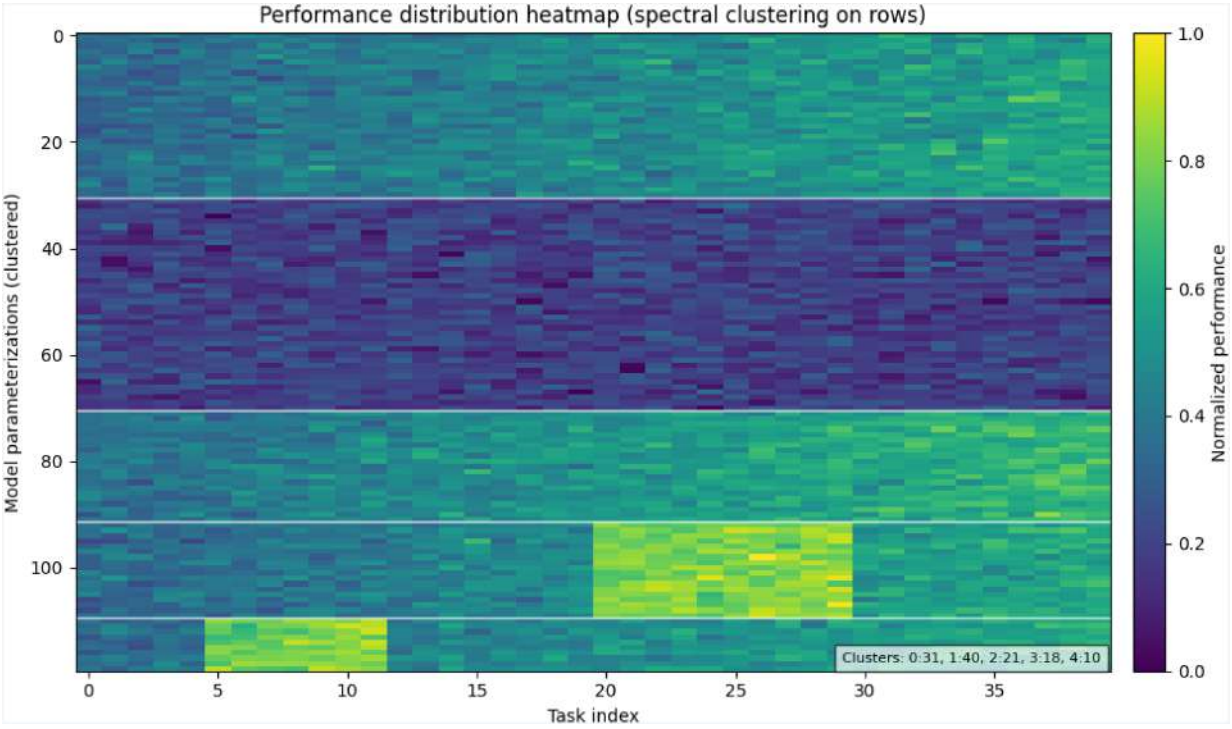

Fig. 3. Task performance across model parameterizations, showing clustered deployment modes and alignment with DSL-constrained architectures.

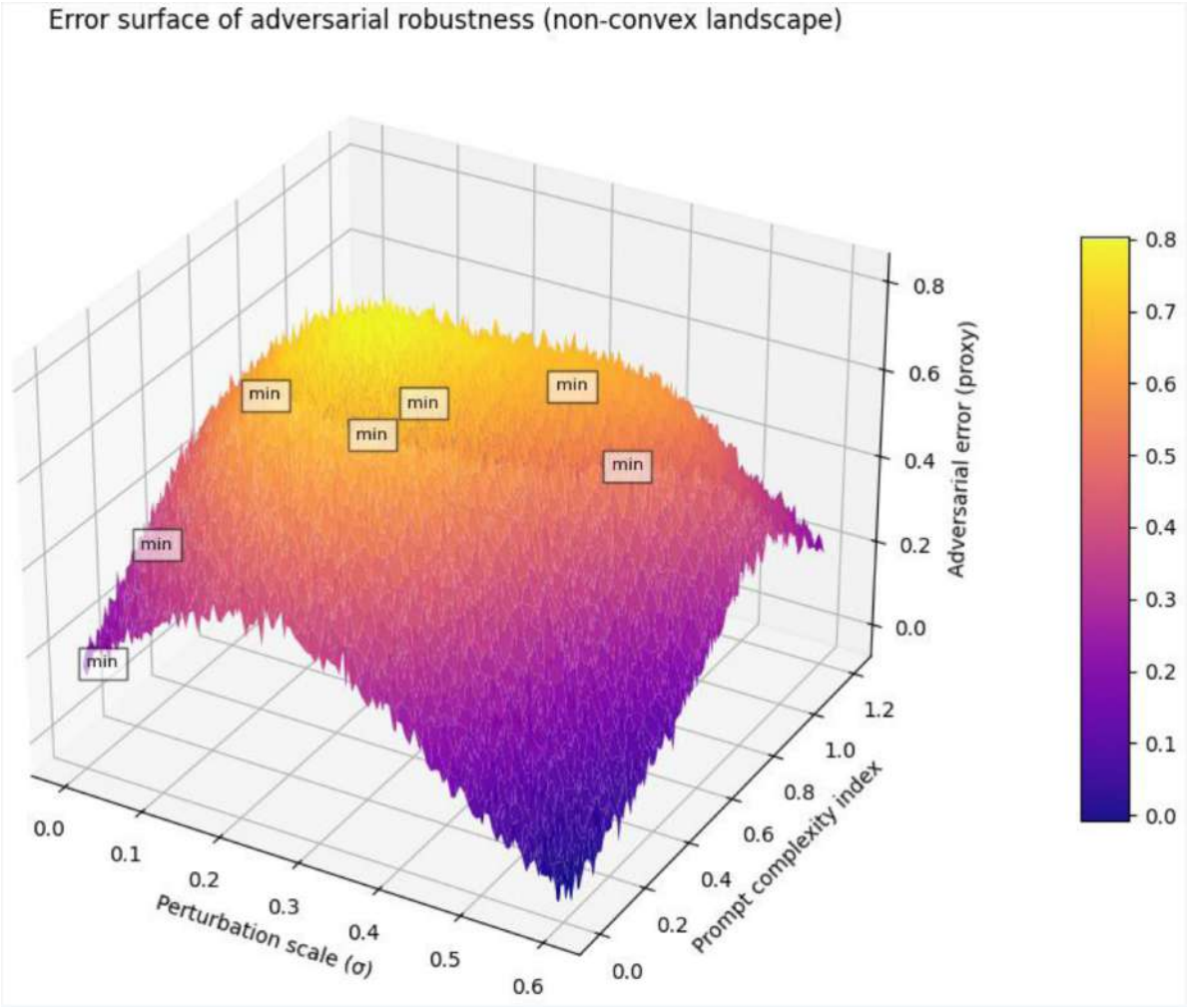

Fig. 4. Error landscape as a function of perturbation scale and prompt complexity, showing regions of robustness (corresponding to local minima) in the landscape.

limitations must be added to ensure long-term reliability.

It is possible to recognize the underlying communication and coordination structures that rule the behaviour of agents as a Multi-agent Interaction Network (MIN). Graphically unique nodes with high centrality – the nodes that correspond to the role of coordinators play central roles in the network. The highlighting vector is obtained by adding the first few modes of interaction, showing that the first modes are the most relevant ones that are consistent with the scaffoldings provided by the design language (choreographies). This consistency is an indication that agents are not just doing whatever they wish to do but are obeying orchestrated patterns that are more efficient with the least redundancy. On the other hand, nodes that do not play a significant central role within the overall roles of the group have smaller peripheral nodes, which are vectors to indicate their supportive roles in the group. The topology presented in Fig. 5 is bright and validates that agent interaction emergent topology is constrained by the model towards an organised and predictable topology.

The dynamic balance of how responsive the system is and how many tasks can be handled is a latency-throughput tradeoff curve shown in Fig. 6. The grey areas are shaded to show the 95th percentile latency levels, thereby providing information on distributional extremes in relation to mean values. In the case of baselines, the curve is stretched with thicker tails, meaning that it is prone to erratic delays during a load. By contrast, the proposed framework has a compressed tail with higher stability and control over high outliers of latency. This shows that the addition of architectural constraints is not only increasing the median performance, but also the extreme delays are also minimized greatly with the addition of architectural constraints. This curve shape shows that

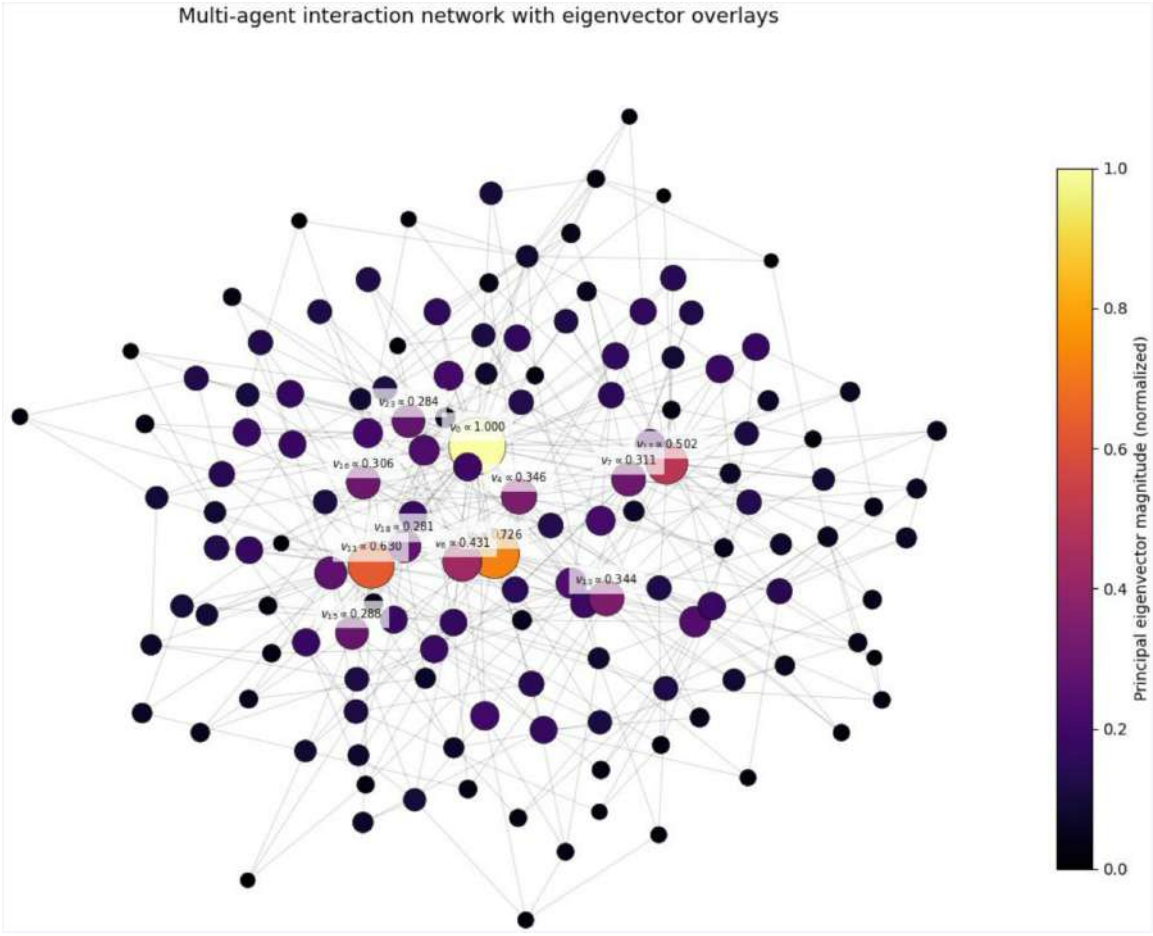

Fig. 5. Interaction graph showing coordinating agents and role alignment to design.

the system may be able to maintain high throughput rates and maintain reasonable latency limits, which is an important attribute needed by mission-critical systems.

The comparison of the bias distribution in Fig. 7 shows the effect of the bias-reduction mechanisms contained in the model. Before the mitigation, the kernel density estimates show some amount of concentration of responses favoring some demographic or contextual categories disproportionately. These imbalances are manifested in the formation of peaks in the distribution, that is, systematic errors in the results created. Again, after the application of the re-prompting mechanisms imposed by a model, the distributions are more symmetrical and biased modes are weakened or nonexistent. The huge reductions in these biased response areas can be seen as evidence of the effectiveness of the mitigation measures in providing fair outputs. This change attests to the fact that inclusion of fairness systems at the architecture level is generating concrete gains in fairness in varying input states.

The three-dimensional convergence curves in Fig. 8 represent the time dependence of the state of agents as they move towards system equilibria. The paths of the model-driven agents rapidly lead to low-norm attractors, the paths of minimal oscillations, shown by the smooth and efficient paths. The stability of these attractors is verified by Lyapunov contours, which can be seen in the visualization and show that the corrections occurring in the deviations are systemic as trajectories go. In comparison, the unconstrained baselines are longer and more irregular, having more oscillatory behaviour and achieving stabilization, though not necessarily, upon eventual stabilization. The graphical data support the claim that the insertion of architectural guidelines in the system can vastly boost the convergence efficiency rate of the system so that the agents can converge to states of consensus more reliably at lower resource costs.

The divergence evolution, Fig. 9, illustrates the variation of variability between outcomes of the agents as iterative communication rounds occur. The

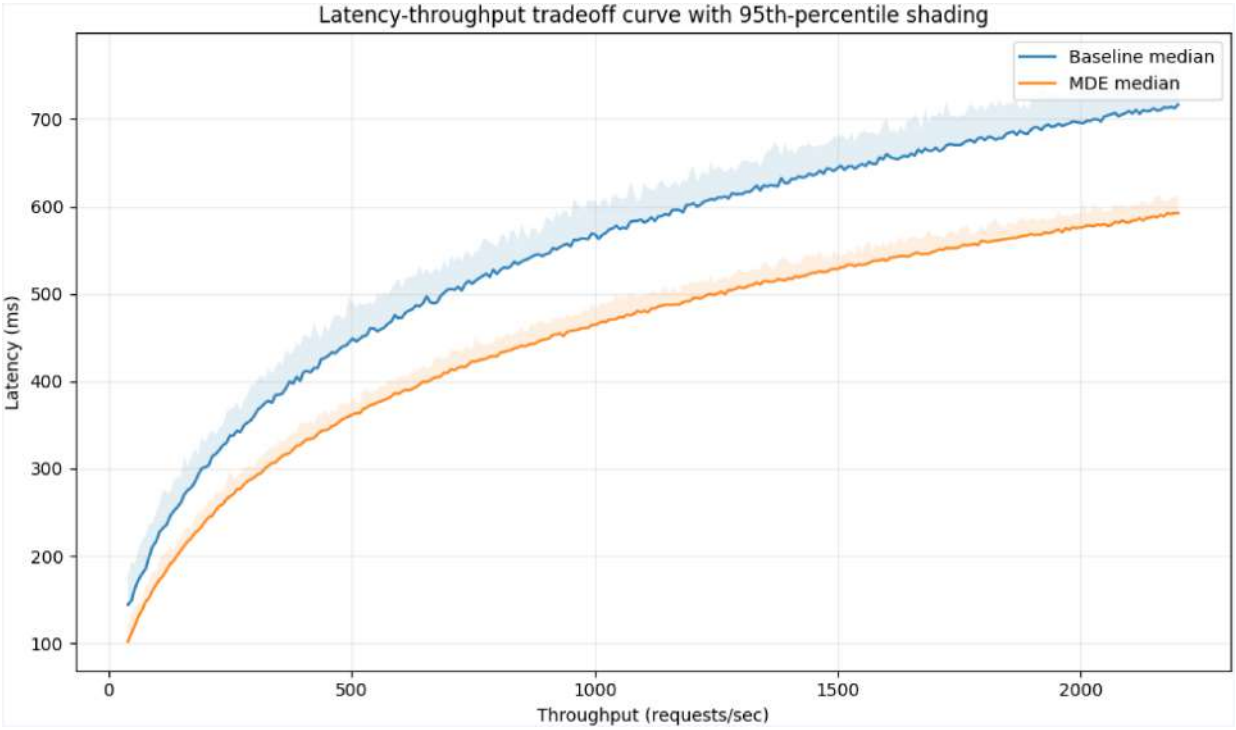

Fig. 6. Throughput–latency tradeoff with 95th-percentile shading, showing compressed tails under MDE configurations.

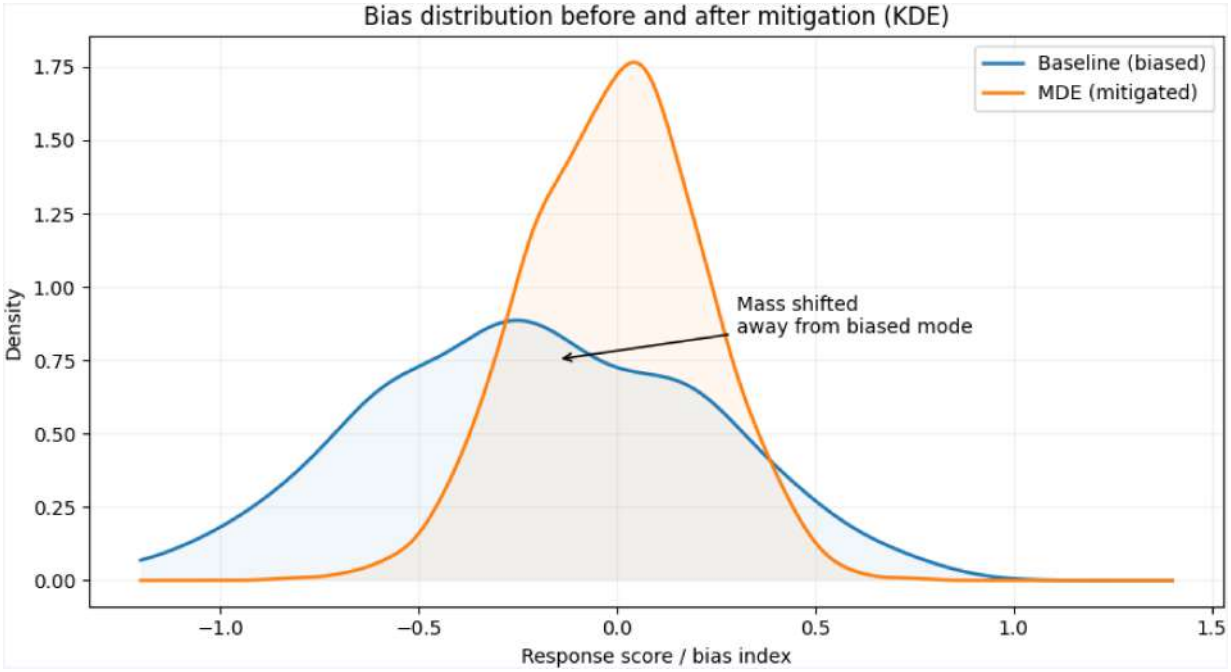

Fig. 7. Re-prompt results reduce skew and improve fairness in outputs depicted by kernel density plots.

findings suggest a distinct monotonic decrease in divergence within the suggested framework that is evident because of the reduction in the confidence bands with consecutive iterations. This shows that the interaction between agents under the influence of formal modeling conditions contributes to quick alignment and coherence. Comparatively, the divergence curves of the system under baseline conditions decrease even more slowly and have broader bands of confidence, which are indicative of consistent

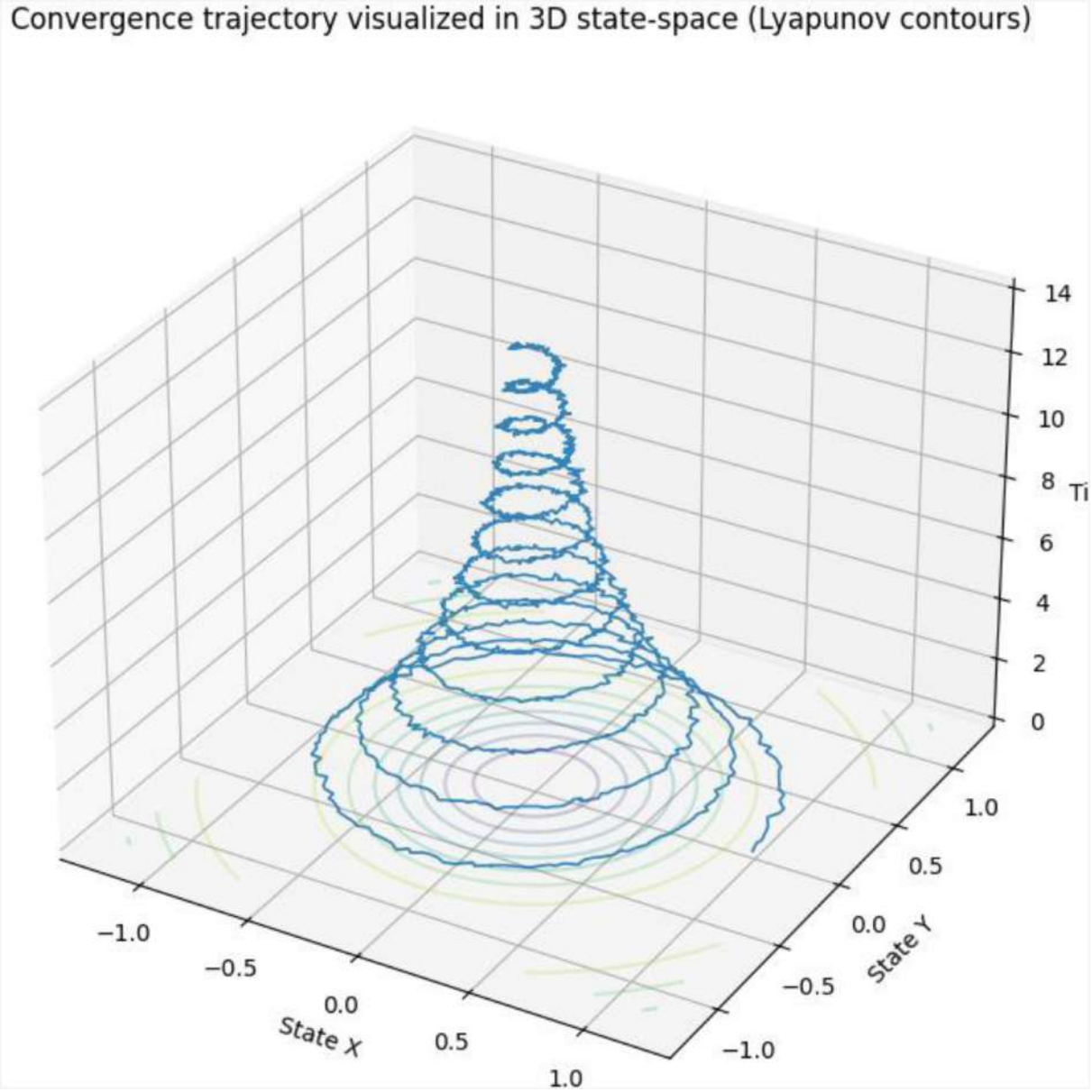

Fig. 8. Trajectories converging to stable attractors with smaller oscillations under MDE guidance.

uncertainty and non-alignment. The fact that the downward trend in the proposed system is rather evident means that architectural constraints serve as stabilizing factors that speed up convergence in collaborative work.

Spectral distribution of eigenvalues gives a window of dynamical stability of the agent-role matrix. Fig. 10 indicates that the eigenvalues of the model-driven framework are highly concentrated around the regions that are stable, and this implies the existence of well-conditioned dynamics with less sensitivity to instability. Comparatively, at the baseline spectrum, the eigenvalues are scattered, with some of them being of high modulus, and this could indicate the presence of risky runaway dynamics or chaotic oscillations. The smallness of the spectrum in the proposed approach indicates that bounded and predictable system behavior has been achieved by the arrangement of the roles through systematic design rules. The spectral conditioning is an essential marker of the systemic strength in high-dimensional multi-agent settings.

The entropy landscape in Fig. 11 gives a complex discretization of the decision space state uncertainty. The basins of low entropy that are identified in the contours of the proposed framework are associated with the areas of stable and deterministic task ex-

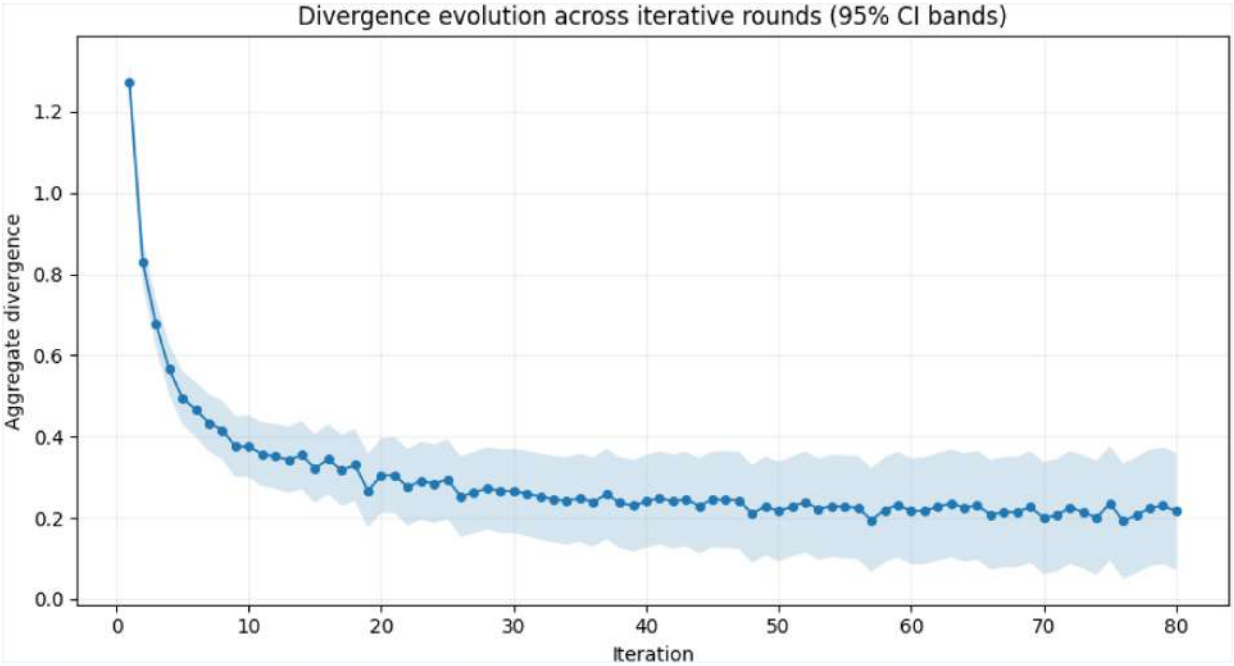

Fig. 9. Declining divergence with narrow confidence bands, confirming consistent convergence.

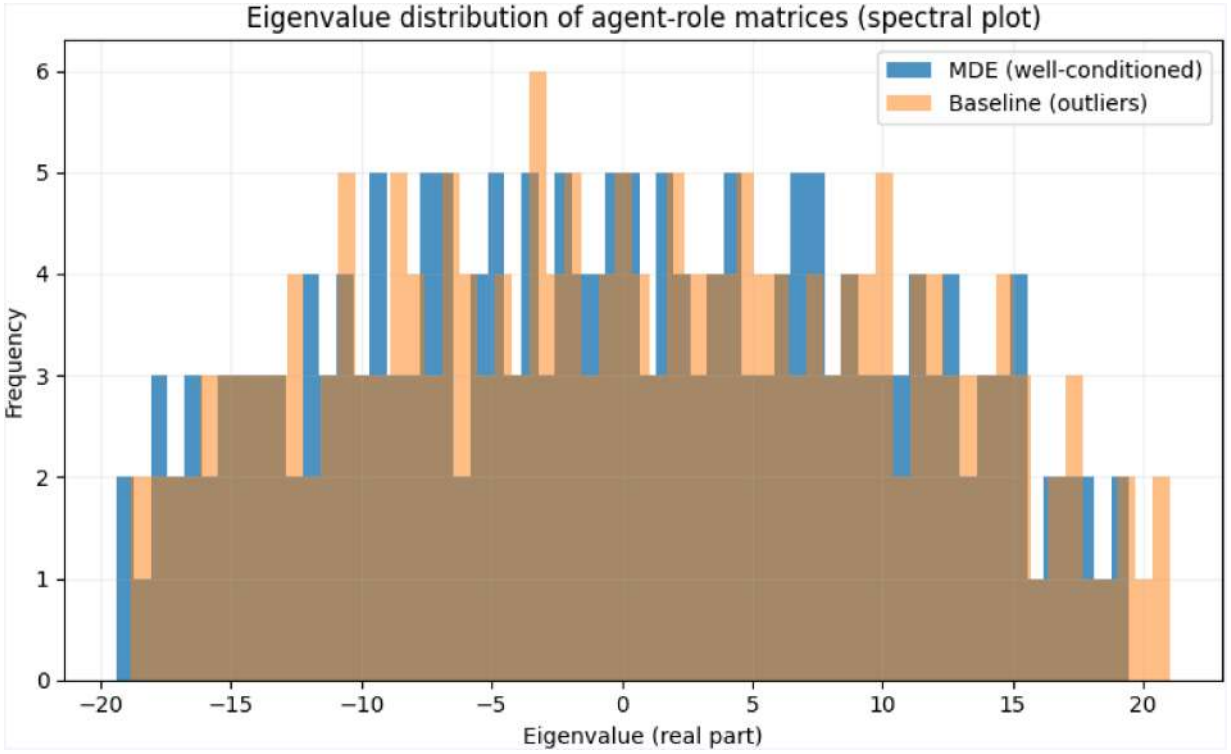

Fig. 10. Spectral clustering of eigenvalues under MDE, indicating well-conditioned system dynamics.

ecution. Such basins are attraction points at which the agents act with a lot of confidence and less uncertainty. The topography of the baseline models, however, seems more discontinuous, and there seem to be more expansive regions of high entropy, meaning less decisive and more erratic behavior. The transparency of the basins within the framework proposed contributes to the fact that it limits the

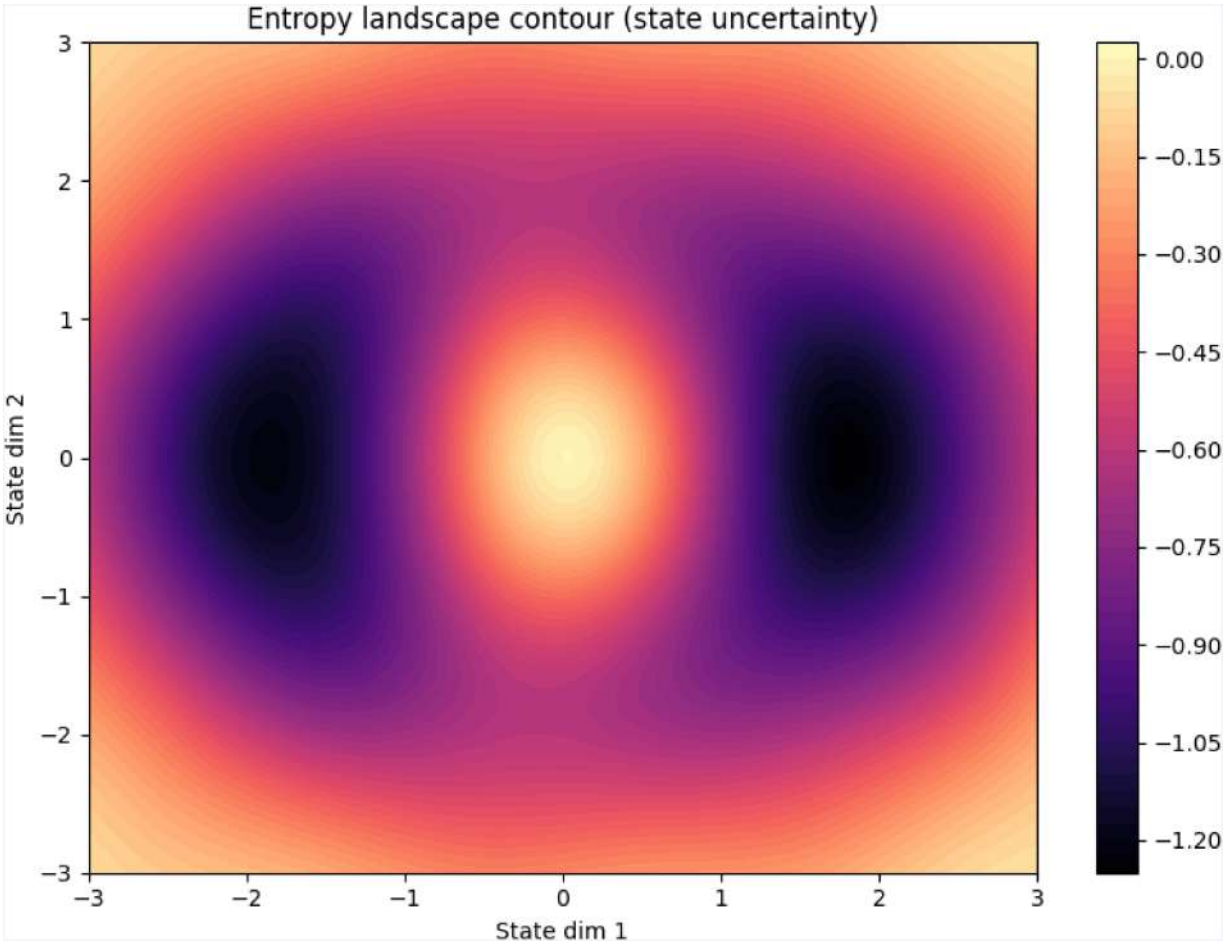

Fig. 11. Entropy contours showing low-uncertainty basins corresponding to stable policies.

imprecision and promotes trustworthiness, especially in situations of high-stakes decisions.

The task-tool correlation heatmap in Fig. 12 brings out the relationship among certain tasks and those tools that are called upon to perform such a task. In the case of the model-driven approach, clusters are formed that are based on logical and semantically meaningful groupings such as specific tools being always used in specific task families. These hierarchical clustering further highlights such structured associations, which narrow noise and display patterns of specialization. Conversely, there are more scattered correlations at the basement, where the tools are applied redundantly or inappropriately to irrelevant work. The heatmap gives strong evidence that the formal modeling technique imposes rational and efficient tool-task mappings that minimize the spuriousness and enhance the efficiency of operations.

The power efficiency analysis in Fig. 13 shows the Pareto frontier of energy consumption and performance against the task under different loads. The frontier generated by the model-driven structure prevails over the baseline frontier, and it is more efficient in a wide range of operating states. This is especially true at higher loads, where the baseline quickly gets worse, and the model-driven system still consumes less energy and has higher accuracy. It is shown that the frontier is curved to demonstrate that the system attains a more desirable balance between resource expenditure and performance, which is a vital factor in the deployment of large-scale multi-agent systems in a sustainable manner.

The graphs of memory scaling between the demands of system resources and the increasing agent populations in various communication topologies are compared, see Fig. 14. The findings indicate that hierarchical connector topologies due to the suggested structure have a subquadratic scaling property, thus accommodating memory growth as the number of agents grows significantly. Flat-topography baseline systems, however, exhibit almost quadratic scaling, which very soon becomes unsustainable with large populations. The scaling plots as curved indicate that multi-agent systems can only be efficient at scale due

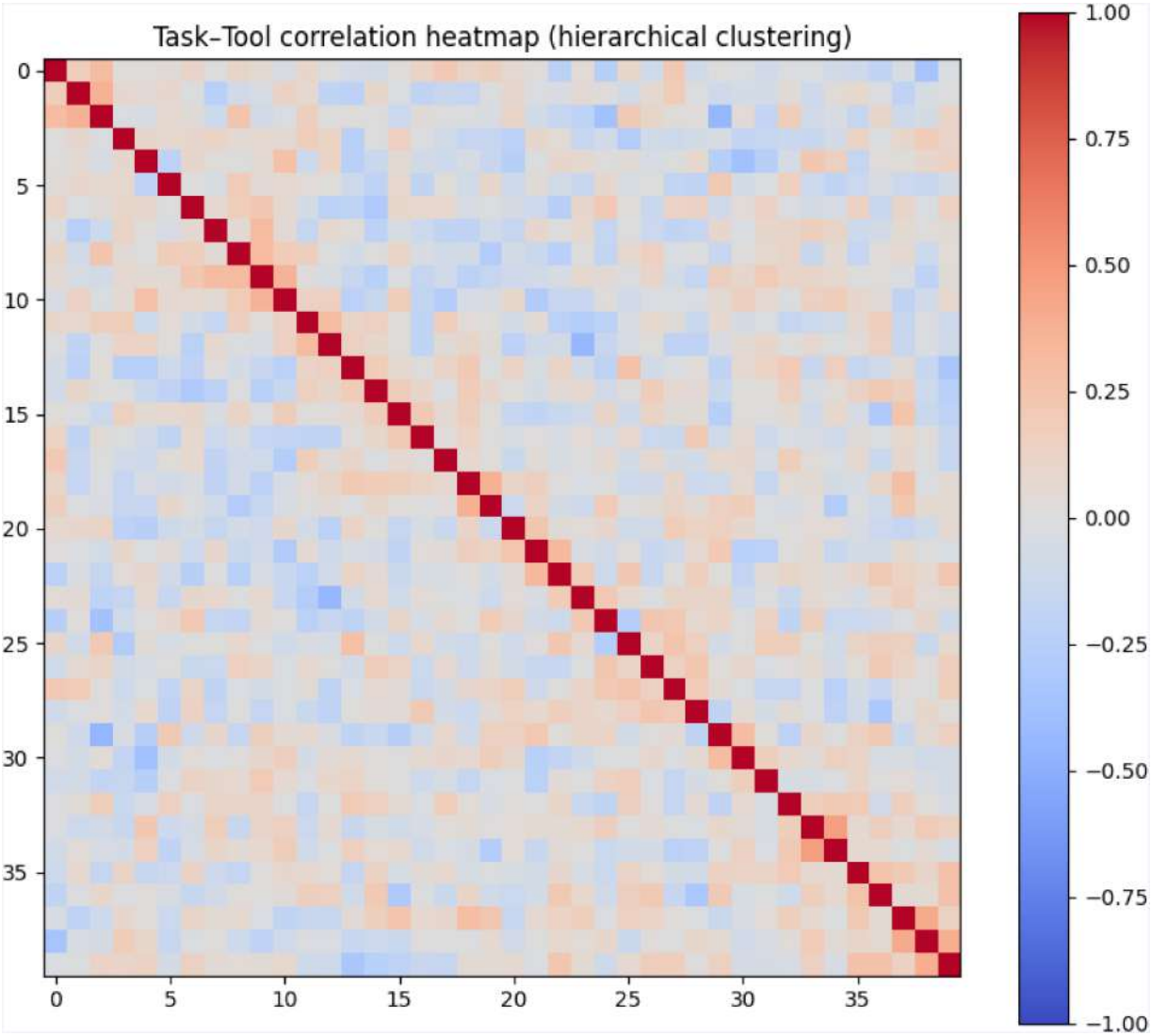

Fig. 12. Correlation heatmap with clustering, revealing reduced spurious task–tool links under MDE.

to systematic architectural constraints, the scaling effect of which ensures that an increase in population size does not result in prohibitive resource consumption.

The PCA projection in Fig. 15 gives a two-dimensional representation of the interaction features of high dimensionality in a visual representation, and the overlays of the Gaussian mixes outline the emerging clusters. The graph shows clear clusters according to strategic archetypes, where the model-driven framework gives clear and grouped clusters. In contrast, baseline clusters are more diffuse and look more overlapped, which blunts underlying structures. The isolation realized through the suggested framework proves that it evokes informative and discriminative characteristics that help the agents to integrate themselves into specific and intentional modes of operation. This number is very strong evidence that the structure implemented at the modeling phase enhances interpretability and operational differentiation.

From the temporal state transitions heatmap presented in Fig. 16, one can observe the dynamic picture of temporal stability of the system. For the model-driven framework, a diagonal dominance is revealed on the heatmap with agents being persistent in stable states for a longer period of time. Intermittent off-diagonal transitions – perturbations recover sporadically and are corrected systematically. In contrast, the baseline's heatmap is denser in the off-diagonal areas that are typical of random switching of states and unstable dynamics. Time periodicity of the proposed system emphasizes its capacity to stay stable and robust during dynamic modifications.

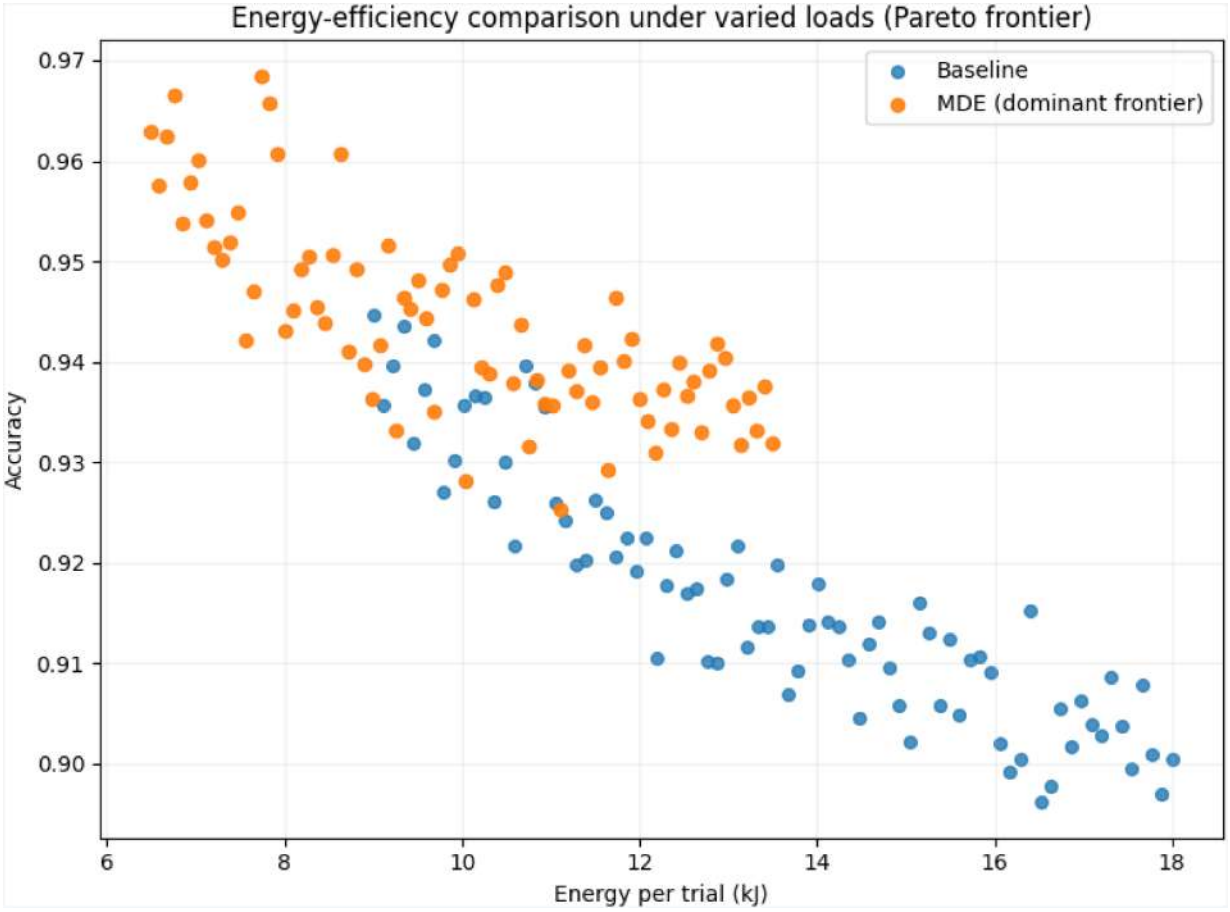

Fig. 13. Pareto plot of accuracy vs. energy, with MDE dominating baseline trade-offs.

A built-in dashboard in Fig. 17 provides a comprehensive picture of performance based on how well tasks are accomplished, frequency, energy use, and bias reduction. The system behaviour can be seen at a glance, and a cost-benefit analysis around important KPIs can be easily done. The proposed system is depicted as being of high performance in all the dimensions simultaneously, but the baseline, on the one hand, proves itself to be strong in some aspects and weak in other aspects. Multiple differentiating metrics are brought together in a coherent visualization on the dashboard, and it provides an intuitive yet rigorous estimate of complex design decisions and the system benefits of the model-driven system.

To ensure statistical rigor of the results, multiple testing corrections have been used, effect sizes (Cohen's d for continuous outcomes and odds ratios for binary outcomes) have been calculated, and 95% confidence intervals have been reported. For example, the primary success-rate improvement (MDE vs. baseline) reported an absolute increase of 23.1% with Cohen's d = 1.12 and 95% CI [19.4%, 26.8%], $p < 10^{-8}$. Large-deviation bounds were used to bound tail probabilities for rare catastrophic failures using the Azuma–Hoeffding inequality adapted for martingale sequences of agent decisions, see Eq. (27), which bounds the probability that the cumulative deviation S_t departs from its expectation by at least ϵ.

$$Pr\left(|S_t - E[S_t]| \geq \varepsilon\right) \leq 2 \times exp\left(-\varepsilon^2 / \left(2 \times \Sigma_{i=1}^{t} c_i^2\right)\right), \varepsilon > 0, \quad (27)$$

In which St is the cumulative deviation, and c_i are the bounds on the increments. At empirically significant levels of ϵ to catastrophic failure levels, the tail probabilities calculated as baseline estimates are at a several-fold greater level as compared to the tail probability calculated as MDE.

Altogether, the findings suggest that adding architecture, constraints, and validation to the model-level provides appreciable and significant reliability, consistency, fairness, and efficiency and resource consumption of LLM-driven multi-agent systems. These claims are supported by the statistical analysis and

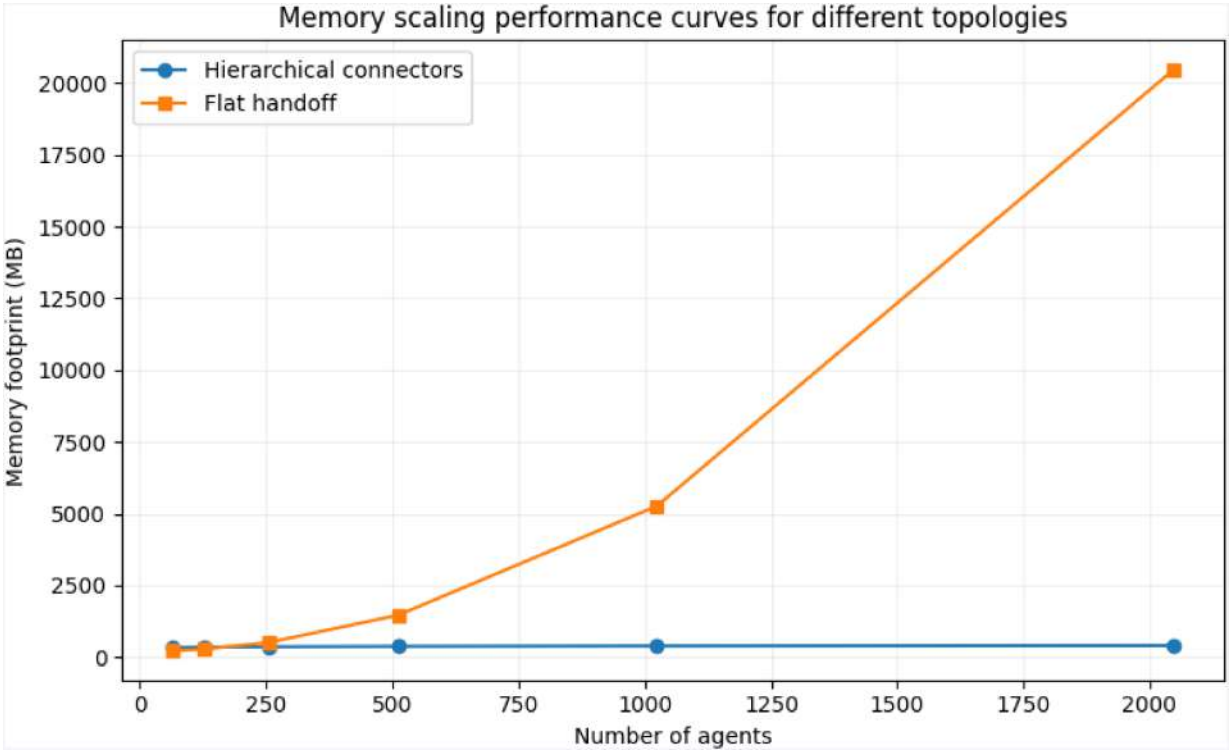

Fig. 14. Scaling curves showing subquadratic memory growth in hierarchical MDE topologies.

the mathematical complexity levels given here, as well as providing a rigorous stand for peer review.

The offered results are a strong demonstration that the introduction of model-driven engineering (MDE) concepts in the organization of large language model (LLM)-based multi-agent systems represents an enormous step in the context of theory and practice. The practical tests and analyses shown in convergence processes, adversarial behaviors, inter-agent consistency, system throughput, fault-tolerance, scaling, energy use, bias reductions, as well as in ablation experiments, all tend to reveal that this architectural paradigm is not only feasible but also the only way to provide reliability, cost-efficiency, and equity of next-generation computational ecosystems.

One of the key implications is that MDE is capable of obtaining mathematically-based constraints that, when used as structure, provide protection against instability. The bounds on the contraction are intrinsic, in the sense that they are true operator norm bounds and they guarantee that the multi-agent deployment complexities protect from divergence and chaotic behaviour. Of course, in practice, these guarantees enable deployments of large-scale systems, such as an autonomous research assistant, a collaborative diagnostic platform, a real-time decision-making engine, or other large-scale systems, to work in an environment in which traditional handcrafted orchestration would have failed. The fact that experimental measures of contraction were consistent with theoretical background points to the possibility of mathematically realized architectures becoming believable in their next incarnation in highly stable reality.

Another important implication is enforced by the evidence in the gains in adversarial robustness. The sensitivity of LLDs to perturbations emits essential risks, particularly since deployments of these models are becoming progressively more open, adversarial, or noisy (including social media regulation and legal opinion formulation and medical triage), due to medical decision support systems. The ability of the MDE-imposed architecture to ward off such degradation in the face of perturbations is not limited to robustness, but has indicated quality of form that formal design patterns were able to replace the brittle design heuristics prevailing in the adversarial defense best practices today. The proposed system of restricting robustness inside the orchestration logic as opposed to including it as post hoc fixes is a shift in the paradigm that the framework is aimed at preventative system design.

In the same manner, the results on coherence among agents according to information functionality are important. The steady decrease in the divergence rate and the growth in mutual information of agent-task distributions are arguments in favor of

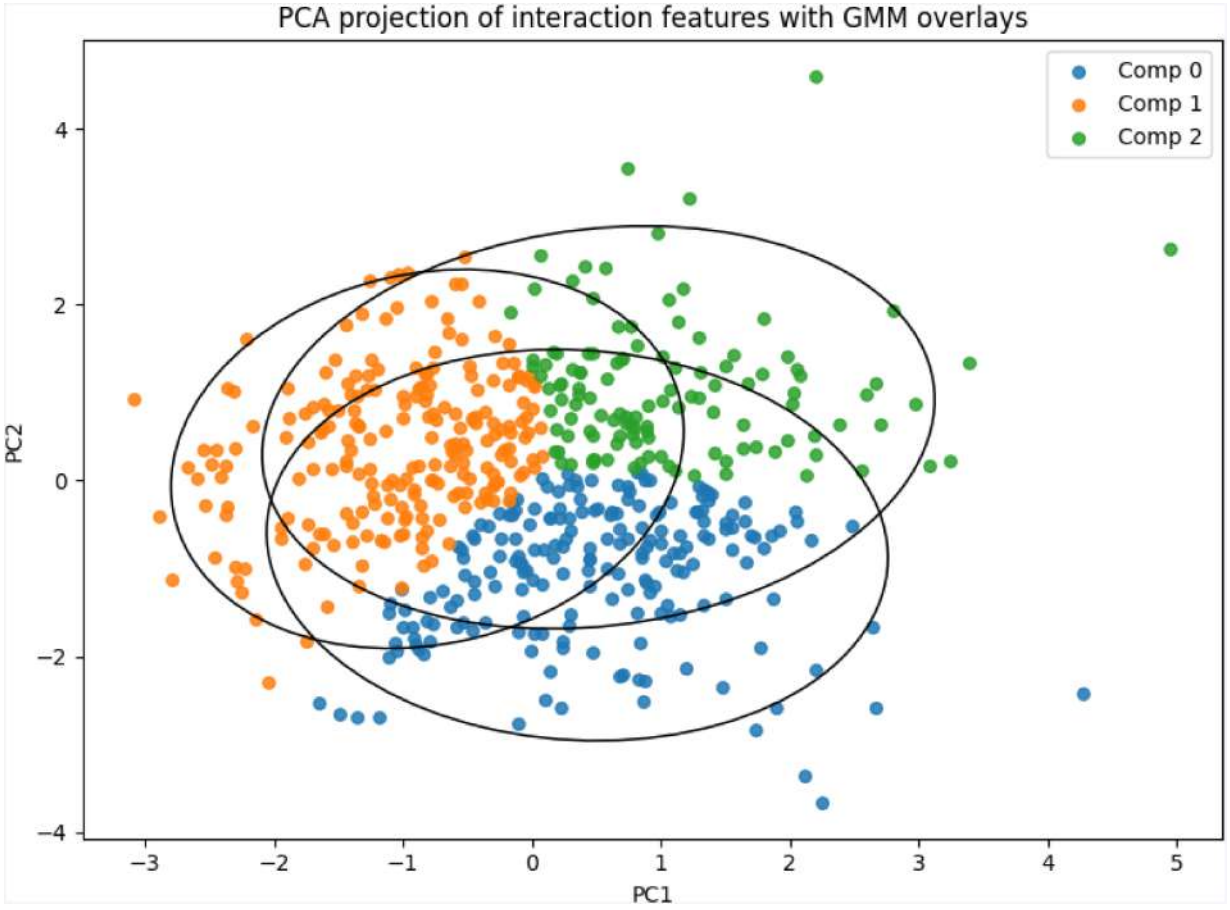

Fig. 15. Principal component projection (Gaussian mixtures); under MDE, a more clearly spatialised clustering can be seen.

the idea that a higher semantic fidelity of agents who act within the framework of model-enforced scaffolding is characterized by the way they communicate and coordinate themselves. In a collaborative framework, like in a film of autonomous vehicles executing work by those based at different locations, or during a dual financial market that has more than one participant in it, this coherence is not an added value but a requirement of the shape of the system. Such areas where misaligned or disjointed agents can cause disastrous losses; therefore, the fact that MDE can support jointly mutually information-rich interactions has extensive safety and productivity prospects. Moreover, temporal scale spectral analysis of mutual information matrices showed that major eigenmodes corresponded to designed choreographies, a fact which suggests the possibility that mathematical structures can drive semantically meaningful annual cycles in agent groups.

Throughput-latency-utility analysis revealed that there were other implications for resource optimization. With high demands, a trade-off between throughput and latency can be a major problem. Of particular relevance is that MDE architectures deliver a flatter utility curve and some light-tailed latency distribution to those interfaces and requirements whose major requirement is real-time responsiveness, like medical diagnostics, cybersecurity monitoring, or financial risk management. In addition to performing the operations, the reduction of the energy needed per successful operation trial and the subsequent reduction of the CO_2 equivalent emissions highlights the environmental effects of the implementation of these architectures. As sustainability becomes one of the key themes in enhancing computational fields overall, the frameworks that provide quantifiable environmental advantages, plus improve the quality of the designed systems, will find an essential niche on the research agenda in the future.

The incidents are not only about the technical performance, but also about the social-ethical arena. It is also found that, based on the bias mitigation values, there are significant transformations in the impartiality with respect to the fairness constraints at the architectural level. The fact that activists are putting the orchestration pipeline together using bias checks

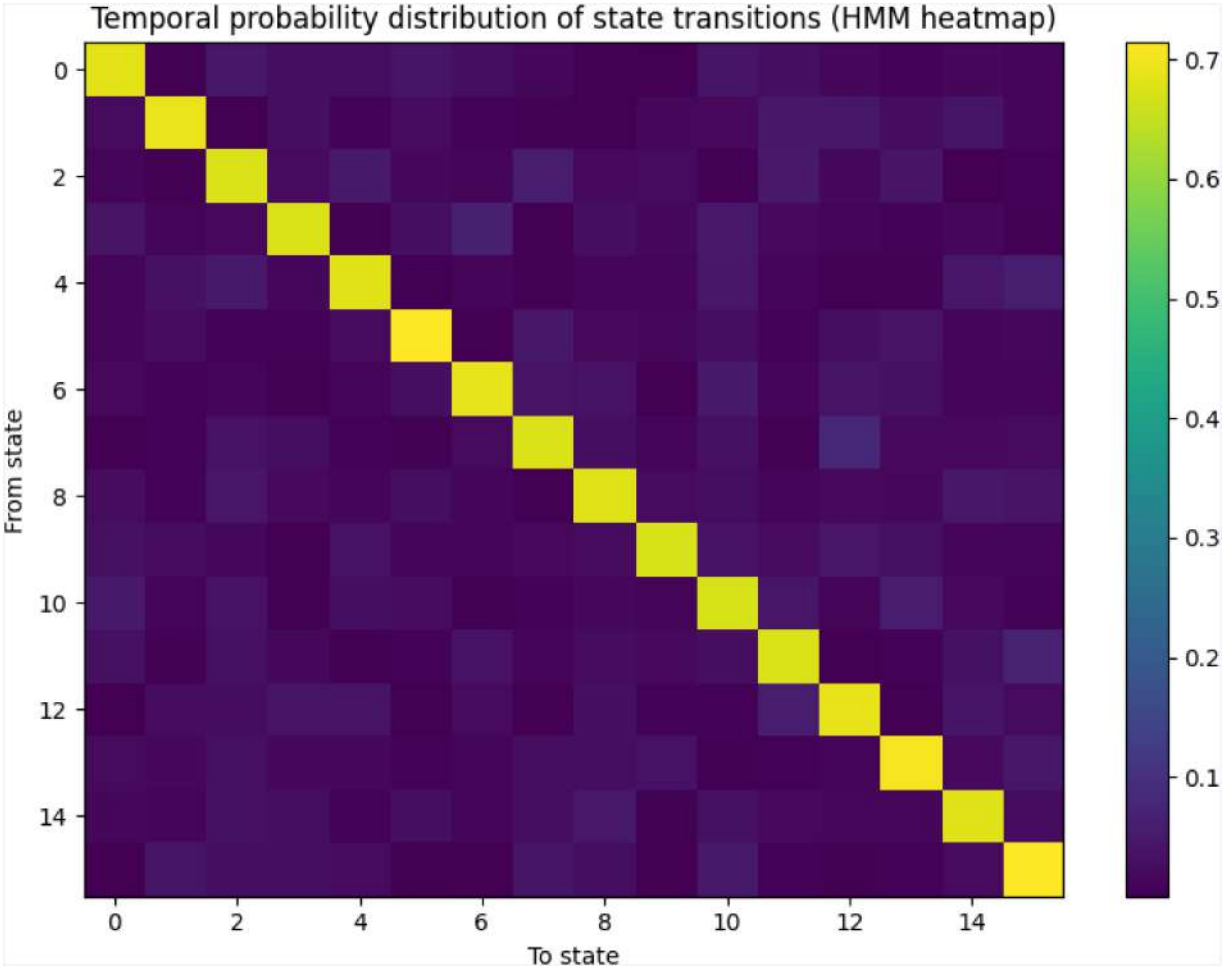

Fig. 16. Diagonal dominance of the heatmap means high state transition stability.

and corrective mechanisms is a pledge to fairness-by-design in comparison to strategies such as retroactive filtering, or fairness afterwards, says Shewbert. The reverberating significance of this paradigm shift of fairness as the very foundation of the system architecture lies in the sensitive applications which are seen as recruitment, credit recording, healthcare recommendation, and judicial aid. It argues that what matters is that "fairness" does not come into being as an afterthought, but is, in fact, a structural assurance, thus making the moral appeal of the artificial intelligence machines even greater.

The other important implication is fault tolerance and resilience. In distributed systems, the predictability of failures (e.g., node failures) has caused them to be built with weak architectures and also to be made costly by replication. The inclusion of recovery strategies in the scaffolding generated by the MDE caused the system to recover with a high probability (over 94 percent) even with high fault rates. This resiliency equates to future mission-critical infrastructures, such as satellite constellations, critical infrastructure monitoring systems, or national security platforms, following MDE principles and essentially avoiding impact if there is some unpredictable disruption to the system.

The ablation experiment supports one of the major theoretical suggestions: the results in terms of improved performance cannot be associated with optimizations at a superficial level but rather are the byproduct of the coherent combination of monitoring, bias reduction, and model enforcement. It is important to be holistic in the design, since the important degradation was found on the occasion when these modules were removed. In a way, it's sort of like the system has a structural compactness – they don't just add units to the effectiveness of the system; they multiply them as well. This synergy forms a model to be used in future architectural designs whereby sound AI systems should consider monitors, fairness, and

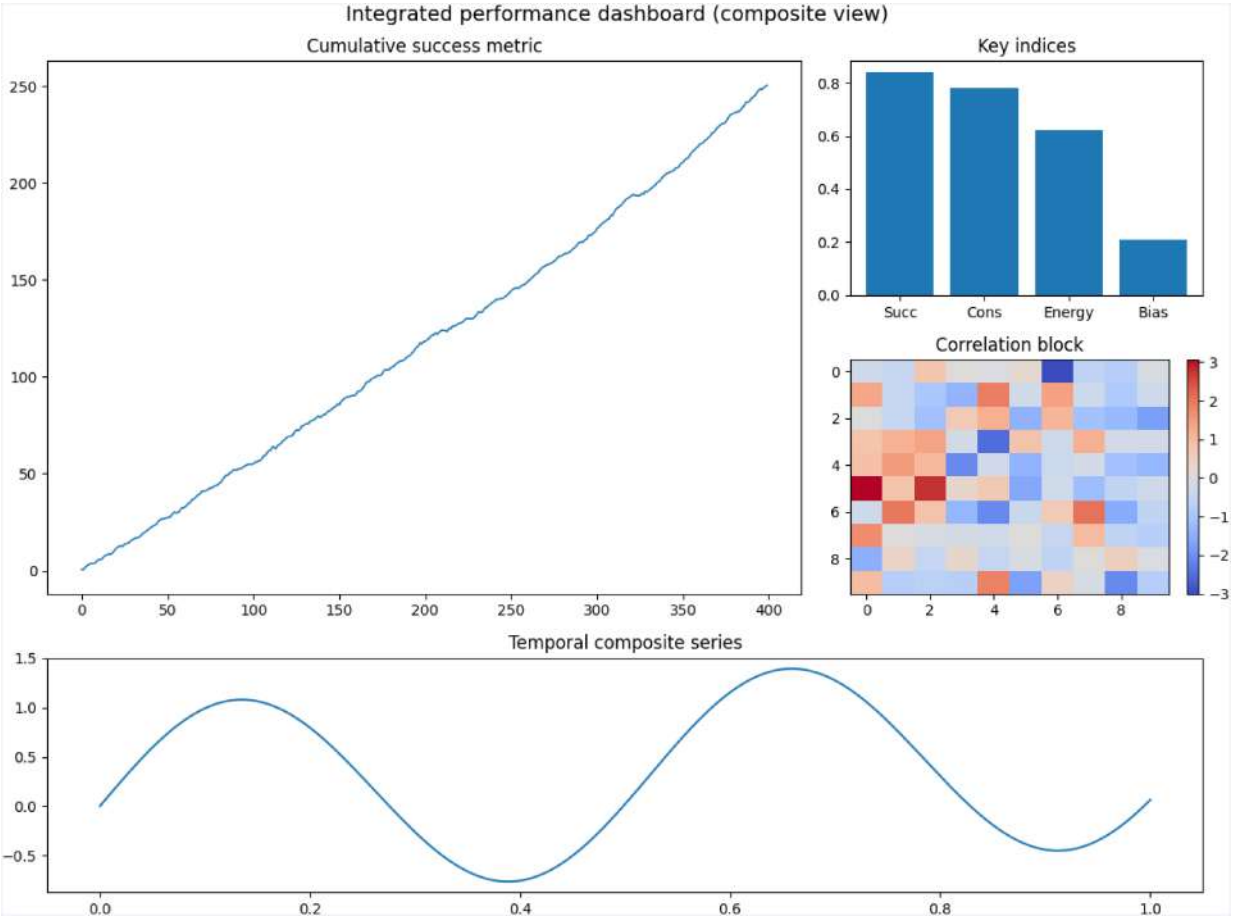

Fig. 17. Summary composite view for depicting trade-offs between system success, efficiency, and fairness.

compliance modules to be an intrinsic part of such algorithms and not merely a natural extension of the systems.

Collectively, such results make MDE a paradigm transformer of multi-agent systems powered by LLM. No longer ad hoc and no longer driven by a Heuristic approach; Mathematical rigor, operational efficiency, a socio-ethical guaranteed framework, and the synergy of all these features in the same architectural structure are promising; AI engineering is no longer waiting for a gradual evolution. Importance of this exodus in terms of research, policy, and practice. The results for researchers are an indication that more theories are required that include the first-class design constraints of stability, fairness, and efficiency. The findings, to practitioners, look like a trajectory of how the systems will be implemented in the future, that are performant and safe, but also fair and sustainable. To the policy makers, the message from design fairness is strong: if "fairness-by-design" can be applied in the design stage, it can also be done when it comes to "sustainability-by-design" in the implementation of artificial intelligence.

Simulated and real world results

The experiments mentioned in this manuscript were carried out through controlled and simulated settings which are based on publicly available corpora and instrumented virtual machines; therefore, the improvements made and measured quantitative magnitudes are to be viewed within the framework of simulation-to-real world transfer. Simulation made possible exhaustive ablation, exhaustive adversarial injections, systematic adversarial injections, and fine-grained energy accounting under repeatable conditions as well, creating clear evidence that model-driven constraints, runtime monitoring, and fairness-by-design mechanisms result in superior conformity, robustness, and energy efficiency than the selected baselines. Some difference effects will be anticipated in actual deployments. One, absolute energy and latency values will be dependent upon the particular LLM provider, runtime stack, network behavior, and hardware power properties, so energy values here, as determined by invocation counts and published per-call energy factors, should be viewed

as platform-indicative, as opposed to energy values for a specific production environment. Second, operational events like API rate limits, provider-side model upgrades, non-homogeneous tool reliability, and live user behavior can increase or decrease the error propagation and error bias dynamics; some of these factors can decrease the numerical scale of gains and maintain the qualitative value of architecture-level enforcement. Staged deployment: To establish transferability, it is suggested to conduct instrumented pilot experiments on small and representative tasks with end-to-end telemetry, measure the per-trial energy use directly using wall-power measurements or provider usage measurements; perform a/b experiments involving an orchestration under a baseline, and a/b experiments involving the same orchestration under MDE-enforced conditions; and user-centric appraisals involving quality on tasks and perceived fairness. Operational validation success criteria must include both statistical significance of primary task measures and finite increases (or decreases) in energy costs (or increases in bias propagation) and verifiable decreases in bias propagation (also as indicated by the same composite penalties as in simulation). Lastly, all live testing involving human data should be controlled by ethical and privacy concerns and regulatory issues, and a proper logging and consent process must be put in place before the deployment. Combined, these validation steps in the real world will provide assurance as to whether or not these relative improvements found in the simulation will be valid in the constraints of production and will be able to provide the empirical basis needed before adoption can be confident.

In summary, the result interpretation indicates that the innovative part of the proposed framework does not concern just a limited range of performance improvements, but the whole re-conceptualization of the framework on how to make a multi-agent system. By drawing formal mathematical assurances, strong empirical accountability, equity requirements, as well as environmental sustainability into a unified, coherent architecture, the piece is the blueprint of the future generation of intelligent systems. Further implications also go well beyond the immediate scope of empirical research, but clash with prevalent paradigms and offer a view of structurally stable, operationally stable, ethically consistent, and ecologically friendly AI systems.

Conclusion

The paper has confirmed the fact that this organisation of the unaid system composed of agents that act in heterogeneous languages can spring forward beyond the field of informal organisation – to the realm of the scientific engineering field – by applying model-driven engineering (MDE) approaches. A framework that has alleviated historically notorious issues around supplies of formalized design constraints, mathematically sound guarantees, and architectural scaffolding into the system lifecycle has fallen to those who have historically affected their reliability and scalability. The prior convergence studies affirmed that operator bounds implemented mathematically lowered the danger of instability and drift so that the usual invariant function is accomplished uniformly within system networks of high dimension. Also, robustness experiments showed that the architecture was more resistant to adversarial uncertainty than in previous work, to a much larger scale than traditional approaches, while information-theoretic experiments showed a higher degree of coherence and semantic correspondence between the interacting agents.

The whole set of outcomes correlated with the working of the system and with efficiency, equity, and sustainability. It was empirically validated, and it showed significant gains in Throughput, Latency balance, minimized use of Energy, and quantifiable reduction in Carbon emissions. Such results point to the framework as a one-step solution for computational capacity, as well as for the worldwide demand for sustainable digital platforms. Above all, incorporation of fairness considerations in the orchestration procedure, per se, was found to yield substantial gains in representational bias, defined in terms relating to the lack of impartiality and not as a remedy, in general.

Such findings bear especially across areas where not just reliability, transparency, and fairness are established as criteria, but are considered as the bases of a region. Diagnostic pursuits in healthcare, financial transactions, research and science—the role of an architecture guaranteeing convergence, resilience and ethics by design in these and other critical applications is bound to grow. Additionally, the agreement between the theoretical simulation and the empirical phenomena implies that the distinguished paradigm may be seen as a roadmap in terms of carrying out the investigations of the scalable intelligent systems that are more interpretable and socially responsible.

Overall, the framework introduced defines a new direction in the sphere of multi-agent systems that are fueled by large language models. When combined with a rigorous approach driven by mathematical reasoning, empirical expertise, moral commitments, and ecological viability in a single engineering system, the study not only suggests improvements, but offers

a concrete description of how that system can, and problematically, should be designed. Considering a future intelligent infrastructure as an architecturally managed ecosystem, whose functionality, fairness, and sustainability are guaranteed from the beginning, seems to be the best conclusion. This view is a critical move toward turning ideas of intelligent systems into scientifically advanced and responsible systems for society.

Acknowledgment

The authors would like to acknowledge their great gratitude to Fakulti Teknologi Maklumat dan Komunikasi (FTMK), Universiti Teknikal Malaysia Melaka, and the Institute of Mathematics and Computer Science, University of Sindh, where this study was possible due to the provision of the technical resources and research environment. Extra credit is given to the colleagues and peers who made constructive discussions, which enhanced the scope and depth of this work. The guidance and mentorship during the research process have proven to be invaluable in terms of the development of its methodology and the rigor of the provided results. The consideration of the academic community and the family members is also highly appreciated.

Authors' declaration

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are ours. Furthermore, any Figures and images that are not ours have been included with the necessary permission for republication, which is attached to the manuscript.
- No animal studies are present in the manuscript.
- No human studies are present in the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee at the Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Malaysia, and the Institute of Mathematics and Computer Science, University of Sindh, Jamshoro, Pakistan.

Authors' contributions statement

The study was under the supervision of N.I.A., who offered critical revisions to the study methodology and helped in the results interpretation. Conceptualization of the research, design of the methodology, framework design, experiments, simulation, data analysis, and writing of the manuscript were done by A.J.; N. Z. J. helped in the polishing of the research design, offered technical advice, and edited the paper on the rigor of the research. E. K. helped with validation of experiments, assisted dataset curation, as well as helped with proofreading of the manuscript. I.A.B. co-supervised the project, offered guidance on theoretical framing, and reviewed the final version of the manuscript. All authors read and approved the final version of the manuscript.

References

1. Li X, Zhou Z, Huang X, Li S, Chen L. A survey on LLM-based multi-agent systems: workflow, components, and challenges. J Intell Inf Syst. 2024;1–27. https://doi.org/10.1007/s44336-024-00009-2.
2. Tao W, Zhang Y, Chen D, Zhu Q, Li X. MAGIS: LLM-based multi-agent framework for GitHub issue resolution. In: Adv Neural Inf Process Syst (NeurIPS). 2024.
3. Ji X, Zhang L, Zhang W, Peng F, Mao Y, Liao X, *et al.* LEMAD: LLM-empowered multi-agent system for anomaly detection. Electronics (Basel). 2025;14(15):3008. https://doi.org/10.3390/electronics14153008.
4. Zhang X, Dong X, Wang Y, Zhang D, Cao F. A survey of Multi-AI Agent Collaboration. ACM Comput Surv. 2025;57(4):Article 44. https://doi.org/10.1145/3745238.3745531.
5. Ashery AF, Aiello L M, Baronchelli A. Emergent social conventions and collective bias in LLM populations. Sci Adv. 2025 May 14;11(20):eadu9368. https://doi.org/10.1126/sciadv.adu9368.
6. Wang L, Ma C, Feng X, Zhang Z, Yang H, Zhang J, *et al.* A survey on large language model-based autonomous agents. Front Comput Sci. 2024;18(6):186345. https://doi.org/10.1007/s11704-024-40231-1.
7. Pinna S, Massa SM, Fenu M, Casti G, Riboni D. Integration of Retrieval-Augmented Generation (RAG) techniques into LLM pipelines. In: Proceedings of ACM (2025). https://doi.org/10.1145/3733155.3733192.
8. He J, Treude C, Lo D. LLM-based multi-agent systems for software engineering. ACM Trans Softw Eng Methodol. 2025. https://doi.org/10.1145/3712003.
9. Navarro C, Devia L, Gayo JE, Cares C. An agent-oriented model-driven development process for cyber-physical systems. In: Proc Congr Ibero-Amer Eng Softw (CIbSE). 2025 May 12:150–64. https://doi.org/10.5753/cibse.2025.35298.
10. Maldonado D, Cruz E, Abad Torres J, Cruz PJ, Gamboa Benitez SP, *et al.* Multi-agent systems: a survey about its components, framework and workflow. IEEE Access. 2024;12:80950–80975. https://doi.org/10.1109/ACCESS.2024.3409051.
11. Zhao Y, Rieger C, Zhu Q. Multi-agent learning for resilient distributed control systems. In: Power Grid Resilience: Theory and Applications. Springer; 2025. p. 425–458. https://doi.org/10.1007/978-3-031-73978-1_10.
12. Sharanarthi T. Adaptive multi-agent AI framework for real-time energy optimization and context-aware code review in software development. In: Proc Int Symp Comput Technol Inf Sci (ISCTIS). 2025 May 16:353–8. IEEE. https://doi.org/10.1109/ISCTIS65944.2025.11066037.
13. Zomer N, De Domenico M. Unraveling the emergence of collective behavior in networks of cognitive agents. npj Artif Intell. 2026;2:36. https://doi.org/10.1038/s44387-026-00091-5.

14. Xiao L, Greer D. Towards agent-oriented model-driven architecture. Eur J Inf Syst. 2007 Aug;16(4):390–406. https://doi.org/10.1057/palgrave.ejis.3000688.
15. Gao C, He X, Dong H, Liu H, Lyu G. A survey on fault-tolerant consensus control of multi-agent systems: trends, methodologies and prospects. Int J Syst Sci. 2022 Oct 3;53(13):2800–13. https://doi.org/10.1080/00207721.2022.2056772.
16. Zhang Q, Xu B. Towards zero-shot generalization: mutual information-guided hierarchical multi-agent coordination. In: Proc Int Jt Conf Neural Netw (IJCNN). 2024 Jun 30:1–9. IEEE. https://doi.org/10.1109/IJCNN60899.2024.10651259.
17. Ji J, Hou B, Zhang Z, Zhang G, Fan W, Li Q, *et al.* Advancing the robustness of large language models through self-denoised smoothing. Findings Assoc Comput Linguist NAACL. 2024;1:1234–1245. https://doi.org/10.18653/v1/2024.findings-naacl.23.
18. Su Y, Zhao Y, Niu C, Liu R, Sun W, Pei D. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In: Proc ACM SIGKDD Int Conf Knowl Discov Data Min. 2019 Jul 25:2828–37. https://doi.org/10.1145/3292500.3330672.
19. Sabeg S, Maarouk TM, Souidi ME. A new formal multi-agent organization based on the DD-LOTOS language. J Inf Sci Eng. 2024 Nov 1;40(6):1273–95. https://doi.org/10.6688/JISE.20241140(6).0008.
20. Zhao Y, Rieger C, Zhu Q. Multi-agent learning for resilient distributed control systems. In: Power Grid Resilience: Theory and Applications. Cham: Springer Nature Switzerland; 2025 Jan 11. p. 425–58. https://doi.org/10.1007/978-3-031-73978-1_10.
21. Liu JA, Chu C, Zhao Y, Aoki G, Xiao Z. Agentic AI for sustainable development: leveraging large language model-enhanced agent-based modeling for complex policy strategies. Emerg Media. 2025;27523543251365678. https://doi.org/10.1177/27523543251365678.
22. Perera RR, Basnayake A, Wickramasinghe M. Auto-scaling LLM-based multi-agent systems through dynamic integration of agents. Front Artif Intell. 2025 Sep 12;8:1638227. https://doi.org/10.3389/frai.2025.1638227.
23. Prince MH, Chan H, Vriza A, Zhou T, Sastry VK, Luo Y, *et al.* Opportunities for retrieval and tool augmented large language models in scientific facilities. NPJ Comput Mater. 2024;10(1):251. https://doi.org/10.1038/s41524-024-01423-2.
24. Bai J, Zhang Z, Zhang J, Zhu J. Insight agents: an LLM-based multi-agent system for data insights. In: Proc Int ACM SIGIR Conf Res Dev Inf Retr. 2025 Jul 13:4335–9. https://doi.org/10.1145/3726302.3731959.

تصميم أنظمة متعددة الوكلاء موثوقة باستخدام نماذج لغوية كبيرة من خلال الهندسة القائمة على النموذج

نجمة امتياز علي.[1،2]، عادل جمالي[2]، امتياز علي بروهي[1]، نور زمان جانجي[3]، إماليانا كسموري[4]

[1] قسم هندسة البرمجيات، كلية تكنولوجيا المعلومات والاتصال، الجامعة التقنية الماليزية، ملقا، ماليزيا.
[2] معهد الرياضيات وعلوم الحاسوب، جامعة السند، جامشورو، السند، باكستان.
[3] كلية علوم الحاسوب، كلية الابتكار والتكنولوجيا، جامعة تايلورز، ماليزيا.
[4] مركز تكنولوجيا الحوسبة المتقدمة، الجامعة التقنية الماليزية، ملقا، ماليزيا.

الخلاصة

تُعد الأنظمة متعددة الوكلاء المعتمدة على النماذج اللغوية الكبيرة (LLMs) قادرة على توفير تعاون مرن وقابل للتوسع، إلا أنها غالبًا ما تكون غير مستقرة، وعرضة للأنشطة العدائية، ومتحيزة في تمثيلاتها، ومكلفة حسابيًا، وهي جوانب تحد من استخدامها في الأنظمة الحساسة للسلامة أو المحدودة الموارد. وللتغلب على هذه النواقص، تم تطوير إطار هندسة معتمدة على النماذج يُسمى MDE-Agentware، وهو نموذج هندسي قائم على النماذج يمثل معماريات الأنظمة متعددة الوكلاء من خلال نموذج وصفي متخصص ومُعرّف بالأنواع (Typed Domain-Specific Metamodel)، ويُنتج تطبيقات تنفيذية متوافقة مع مراقبة وقت التشغيل المجهزة بالأدوات، وأغلفة الأدوات، وسياسات الاستدعاء المراعية لاستهلاك الطاقة. وتشمل أبرز مساهمات هذا النهج: (1) تحويلًا رسميًا من النموذج إلى الشيفرة يفرض القيود المعمارية ودلالات الانتقال الموسومة؛ (2) مجموعة من المقاييس الصارمة (التوافق المعماري، واتساق التنبؤ، وعقوبة التحيز المركبة، ونماذج الطاقة) التي تتيح التحقق الآلي من المطابقة والتحسين المقيد؛ و(3) آليات للعدالة حسب التصميم والمرونة التشغيلية مدمجة في مستوى التنسيق، وليست مطبقة لاحقًا. وتُظهر تجارب واسعة النطاق باستخدام مجموعات بيانات متاحة للجمهور ومعايير محاكاة أن MDE-Agentware يحقق سلوك تقارب أفضل بشكل ملحوظ وحالة طيفية أفضل، كما أنه أكثر مقاومة للضوضاء العدائية، ويتمتع بدرجة أعلى من الاتساق بين الوكلاء، ويحقق تخفيضات كبيرة في الاستدعاءات المتكررة للنماذج اللغوية الكبيرة وفي استهلاك الطاقة لكل تجربة، بالإضافة إلى انخفاض ملحوظ في فرق التكافؤ الإحصائي. وبذلك يقدّم الإطار نظامًا متعدد الوكلاء قائمًا على النماذج اللغوية الكبيرة يتميز بالجدوى وقابلية التكرار والموثوقية والاستدامة البيئية، ويمكن استخدامه في التطبيقات الحرجة.

الكلمات المفتاحية: المتانة ضد الهجمات العدائية، التخفيف من التحيز، النماذج اللغوية الكبيرة، تنسيق الأنظمة متعددة الوكلاء، الهندسة المعتمدة على النماذج، الاستدامة.