OPEN CALL ARTICLE

A Trustworthy Agentic Multi-LLM Network: Challenges, Solutions, and a Use Case

Haoxiang Luo, Gang Sun, *Senior Member, IEEE*, Yinqiu Liu, Dusit Niyato, *Fellow, IEEE*, Hongfang Yu, *Senior Member, IEEE*, Mohammed Atiquzzaman, *Senior Member, IEEE*, and Schahram Dustdar, *Fellow, IEEE*

Abstract

Large Language Models (LLMs) demonstrate strong potential across a variety of tasks in communications and networking due to their advanced reasoning capabilities. However, since different LLMs have different model structures and are trained using distinct corpora and methods, they may offer varying optimization strategies for the same network issues. Moreover, the potential maliciousness of its hosting device can result in LLM responses with low confidence or even bias. To address these challenges, we propose a collaborative framework that organizes distributed LLMs into an Agentic Multi-LLM Network (Agentic MultiLLMN). This novel architecture transforms individual network nodes into collaborative autonomous agents equipped with LLMs. Through agentic functionalities such as autonomous planning, reasoning, and collaborative decision-making, the framework can provide high-quality responses for complex network optimization problems. Specifically, we first review related work and highlight the limitations of existing LLMs in collaboration and trust. We then introduce a workflow of the proposed Trustworthy Agentic MultiLLMN framework, which leverages blockchain to provide a verifiable audit trail for autonomous agent interactions. Given the severity of False Base Station (FBS) attacks in wireless communications, we present FBS defense as a case study. In this scenario, each Legitimate Base Station (LBS) acts as an agentic LLM node, collaborating to optimize power allocation and mitigate attacks. The simulation shows that this framework can more effectively resist this attack compared to other solutions. Finally, we outline promising future research directions in this emerging area.

I. Introduction

Large language models (LLMs) have become the cornerstone of artificial intelligence (AI), showing great potential in natural language understanding and generation tasks [1]. LLM provides services to users in the form of AI-Generated Content (AIGC), which has been widely used in various aspects of society, such as education, healthcare, and information. In particular, it is promising to use LLM to solve various optimization problems in networks, such as optimal resource or power allocation strategies [2]. In contrast to traditional Deep Learning (DL)-based approaches, LLM-based approaches can directly understand the user's natural language intent and efficiently match their requirements. Moreover, LLM shows strong zero-shot generation ability, which can be directly applied to some optimization problems without retraining. Additionally, the emerging paradigm of Agentic AI transforms LLMs from passive generators into autonomous agents capable of perception, planning, reasoning, and tool use to achieve complex, goal-oriented tasks [3]. These agentic capabilities can further enhance the LLM's ability to perceive an environment and process complex downstream tasks. Thus, it can provide more reasonable outputs than DL methods. As a result, an LLM establishes a new paradigm for optimization problems across a variety of scenarios, essentially for 6G.

However, different LLMs have been developed using diverse training data and methodologies, leading to variations in their outputs for the same input. Additionally, outdated or limited training corpora can cause LLMs to generate biased or low-quality responses, a phenomenon commonly referred to as hallucination. Also, the generalization capability of a single LLM remains limited, making it difficult to adapt effectively across diverse network scenarios [4]. For instance, in wireless networks, these hallucinations and biased results may cause the power allocated to the base station to exceed the hardware capacity limit. This, in turn, leads to the failure of the strategy and subsequently affects the normal operation of the communication system. Furthermore, a single LLM may generate unrealistic spectrum allocation strategies due to a lack of domain knowledge, thereby causing severe interference [5]. In the edge computing scenario, when edge nodes are added or removed, a single LLM lacks the real-time adaptive inference capability. They cannot dynamically integrate the parameters of new nodes and must be re-taught or fine-tuned [2]. To address these

Haoxiang Luo, Gang Sun (corresponding author), and Hongfang Yu are with the University of Electronic Science and Technology of China, Chengdu 611731, China; Yinqiu Liu and Dusit Niyato are with Nanyang Technological University, Singapore 639798; Mohammed Atiquzzaman is with the University of Oklahoma, Norman, OK 73019 USA; Schahram Dustdar is with the TU Wien, Vienna 1040, Austria, and also with ICREA, Barcelona 08002, Spain.

This work was supported in part by the National Science and Technology Major Project under Grant 2025ZD1302700 and Grant 2026ZD1307200, and in part by the National Natural Science Foundation of China under Grant 62394324.

Digital Object Identifier: 10.1109/MWC.2026.3666842

limitations, leveraging multiple LLMs to respond to the same query collaboratively is gaining increasing attention. For example, Wang et al. [6] employed GPT-3, GPT-4o, Llama3-8B, and others collectively to provide care services for the elderly. Similarly, Marro et al. [7] proposed *Agora*, a scalable communication protocol designed to automate and coordinate collaboration among multiple LLMs.

Although a single LLM can perform multiple inferences to provide different answers for users by adjusting hyperparameters, the multi-LLM collaboration still has the following advantages. Single LLM repeated inference is constrained by its fixed training corpus and model structure, which cannot be resolved by parameter tuning. In contrast, multi-LLM enables cross-model knowledge complementarity. For instance, LLMs that are good at reasoning are combined with vertical LLMs in specific professional fields. Then, their collaboration covers both efficiency and accuracy. Meanwhile, multi-LLM in a distributed network performs parallel local perception and reasoning, reducing latency for sequential repeated inference of a single LLM.

Building on this paradigm and the emerging research on LLM-based Multi-Agent Systems (MAS), we introduce the concept of an **Agentic Multi-LLM Network (Agentic MultiLLMN)**. This network organizes multiple specialized LLMs as autonomous agents jointly providing intelligent services to achieve collective wisdom that no single agent can offer. Although promising, the deployment of Agentic MultiLLMN for network optimization in 6G and other communication systems requires addressing two critical challenges:

- **Response Efficiency:** A key challenge lies in determining how to efficiently select the best response from multiple LLMs and ensure consensus among them. This is essential for enabling Agentic MultiLLMN to meet the ultra-low-latency requirements of modern communications.
- **Response Trustworthiness:** LLMs deployed on compromised or untrusted devices may exhibit malicious behavior due to Trojans, viruses, or the intent of their operators. Moreover, intentionally harmful LLMs, such as WormGPT [8], can actively mislead users by generating deceptive responses. Such threats pose significant risks to the reliability of Agentic MultiLLMN and can severely degrade network performance.

Therefore, we propose to use the blockchain as a solution to Agentic MultiLLMN to provide a trustworthy optimization method for the network. On the one hand, blockchain consensus enables Agentic MultiLLMN to decide the best quality responses without relying on trusted third parties. On the other hand, the immutable and traceable nature of the blockchain guarantees the credibility of the response generated by Agentic MultiLLMN. Specifically, our contributions are as follows.

- **Agentic MultiLLM Architecture Design:** We propose a distributed architecture that transforms LLMs into network-aware autonomous agents with a perceive-reason-plan-act-learn loop. Unlike traditional multi-LLM frameworks, this architecture enables active collaboration rather than passive output fusion.
- **Blockchain-Enhanced Trust Mechanism:** We propose a trust system for Agentic MultiLLM. It establishes an unalterable audit record for agent interactions through blockchain transactions and uses the Byzantine Fault Tolerance consensus to reject malicious proposals. This eliminates the reliance on a centralized controller and ensures the credibility of the response.
- **Domain Knowledge Embedding for Network Optimization:** We integrate 3GPP standards and network simulation tools into LLM reasoning prompts, enabling the framework to generate network-compliant strategies without retraining. It demonstrates great potential in resisting False Base Station (FBS) attacks.

II. The Trustworthy Agentic Multi-LLM Network

A. Related Work of Multi-LLM and Trustworthy LLM

Due to the differences in the learning corpus, training path, and scenario-based orientation of different LLMs, their answers to the same questions will inherently differ. Moreover, some LLMs could have limitations and obsolescence in the training data, resulting in biased content generation and even hallucinations. Thus, researchers propose the collaboration of multiple LLMs to provide a reliable response to users. For instance, Feng et al. [5] proposed a collaborative framework that integrates three LLMs. The framework systematically addresses knowledge gaps in the single LLM, such as those arising from incomplete or outdated training data, by leveraging cross-model validation and complementary expertise among the LLM ensemble. The framework achieved a 19.3% performance improvement over a single LLM on four tasks with different knowledge areas, including common sense selection, word puzzles, natural language reasoning, and news summary. In addition, Owens et al. [9] considered the bias of individual LLM outputs, which is also caused by limited training data. Despite some progress in natural language processing (NLP) techniques, such as data enhancement and model fine-tuning, biased results persist. Therefore, they built a communication model of multi-LLM to reduce bias in the generated results.

Additionally, since the existence of malicious LLMs such as WormGPT and the potential malicious behavior of the devices carrying LLMs, research on trustworthy LLMs has been conducted. Liu et al. [10] used blockchain to provide trusted endorsement and protection for AIGC products. They also provided traceable verification services for altering ownership of AIGC products based on blockchain and incentive mechanisms. In addition, Luo et al. [11] considered the trustworthiness of LLMs from three aspects: learning corpus, training process, and generated content. They also highlighted that blockchain will play an important role in these areas.

However, the aforementioned studies still have not solved the problem of trusted collaboration among multiple LLMs. As a result, it is difficult to fundamentally solve the problems of bias, hallucinations, and credibility in the content generated by LLMs [4]. We categorize existing related works into four groups and compare their capabilities, limitations, and our improvements, as shown in Table I. It shows that existing works either lack intelligent reasoning or trusted decentralized coordination. Our framework integrates their strengths, filling the gap of trusted, intelligent, and domain-specific multi-agent collaboration for 6G security.

Category	Representative Works	Core Capabilities	Limitations	Our Improvements
MAS	[6], [7]	Distributed collaboration for network control	Lack LLM-driven intelligent reasoning; centralized coordination; no trust mechanism	Integrate agentic LLMs for autonomous planning/reasoning; blockchain for decentralized trust
Multi-LLM Collaboration	[7], [5], [9]	Cross-model knowledge complementarity; output fusion	Centralized coordination; no network domain optimization; no Byzantine fault tolerance	Decentralized P2P network; embed 3GPP standards/simulation tools; consensus for fault tolerance
Blockchain for Trust	[10], [11], [12]	Data immutability; SI authenticity verification	No intelligent optimization; single-function trust; no dynamic decision-making	Combine with LLM agents for dynamic power allocation; consensus-driven decision validation

TABLE I. Related work categorization and comparison.

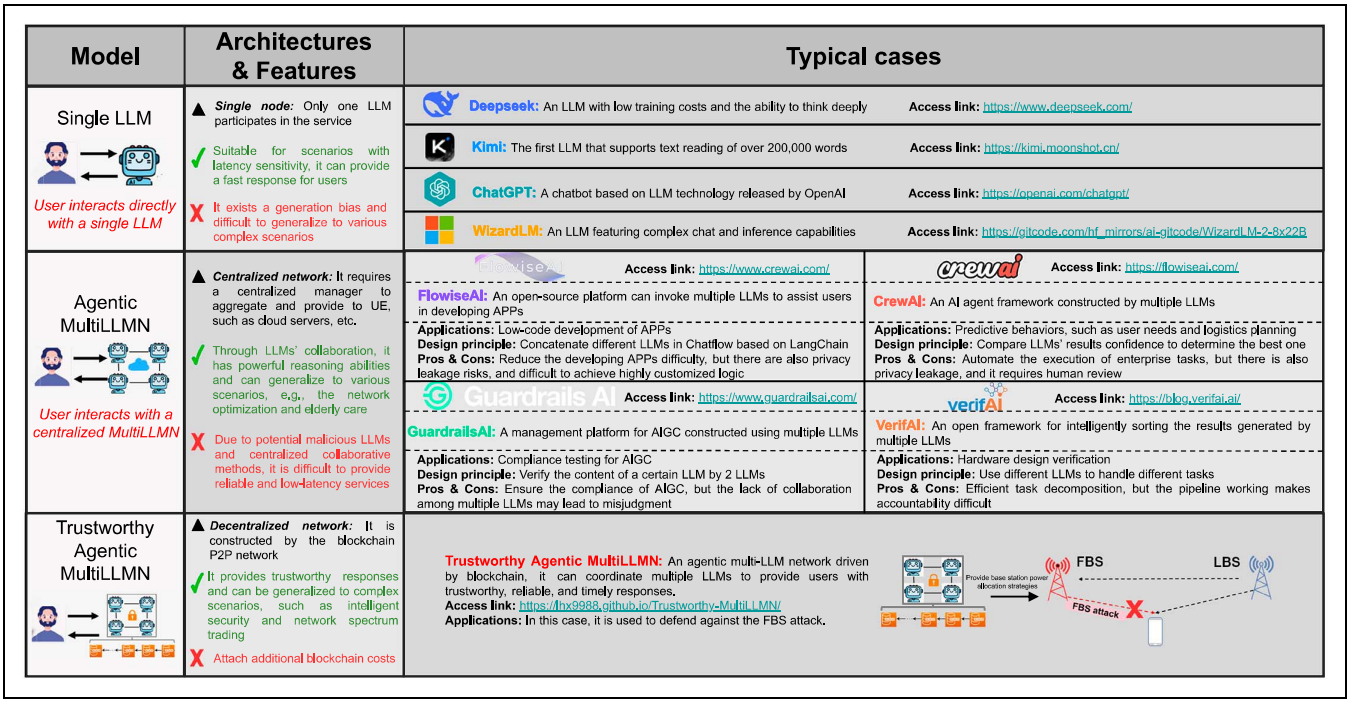


FIG. 1. Compared with a single LLM and Agentic MultiLLM, Trustworthy Agentic MultiLLMN has distributed characteristics and is less likely to be affected by malicious LLMs. It has higher network stability and robustness, and can provide trusted services for users.

B. Concepts for Agentic MultiLLMN and Trustworthy Agentic MultiLLMN

Fig. 1 illustrates the comparison among a single LLM, MultiLLMN, and Trustworthy MultiLLMN, including their architectures, application scenarios, and typical cases.

- **Single LLM** uses one LLM to answer users' questions or address optimization goals. In this regard, the LLM can directly provide a timely response to the question. However, a user cannot refer to the answers from other LLMs, even if the content generated by this LLM is incomplete, biased, or hallucinatory. Although users can obtain different answers from multiple LLMs separately in an offline state, it will cause more confusion and decision-making difficulties for them. Due to the answers given by different LLMs may not be consistent, and their perspectives of focus are completely different.
- **Agentic MultiLLMN** is an intelligent network composed of multiple LLMs operating as autonomous agents that can collaboratively respond to requests. Through specialized inter-agent communication and planning, the Agentic MultiLLMN not only leverages common capabilities of each LLM but also actively utilizes agentic mechanisms, such as goal decomposition, self-reflection, multi-step planning, and tool utilization, to autonomously construct and refine solutions to complex problems. This structured, collaborative reasoning process can avoid the biased or hallucinatory outcomes common to a single model. The typical applications of this method are 360 AI Assistant[1] and Corex[2]. The former can invoke three LLMs to work collaboratively. The latter is designed by the Shanghai AI Lab and enables multiple LLMs to reason together. Amazon[3] has also explored message routing strategies among multiple LLMs. Other examples are shown in Fig. 1. However, there are also problems with response efficiency and credibility in existing approaches. Traditional centralized networks require a central node to coordinate these responses from different LLMs, which is time-consuming. The centralized network architecture also has the risk of a single point of failure. Furthermore, there exists a

[1] https://bot.360.com
[2] https://link.zhihu.com/?target=https%3A//github.com/QiushiSun/Corex
[3] https://aws.amazon.com/cn/blogs/machine-learning/multi-llm-routing-strategies-for-generative-ai-applications-on-aws/

potential for malicious activities on the device that hosts the LLM, which could subsequently influence other benign LLMs within the Agentic MultiLLMN. This compromised interaction can lead to the dissemination of unreliable or untrusted responses.

- **Trustworthy Agentic MultiLLMN** integrates blockchain into the Agentic MultiLLMN. This integration enhances the governance and operational integrity of the collective agents. With security capabilities, blockchain can to ensure traceability and immutability of not just the final generated content, but also the intermediate agentic decisions, tool use, and interactions [10], [11]. It provides a cryptographically verifiable audit trail for all autonomous agent activities and behaviors, which is essential for governing such systems in untrusted environments. Furthermore, its consensus mechanism empowers the Agentic MultiLLMN to execute dependable decision-making processes efficiently, thereby obviating the necessity for reliance on a centralized controller [4]. This decentralized trust architecture is particularly apt for scenarios like resisting FBS attacks. In this scenario, the network must distinguish between legitimate and malicious signaling without a single point of failure. In an FBS attack, malicious nodes attempt to deceive users. A centralized LLM system could be a single point of failure or a target for hijacking. In contrast, the Trustworthy Agentic MultiLLMN ensures that power allocation strategies are verified by a consensus of honest nodes, making it robust against such deceptive adversarial behaviors. Although the integration of blockchain effectively mitigates the challenges of response efficiency and trust, it incurs costs associated with blockchain implementation related to storage and communication costs.

Existing multiple-LLM frameworks, such as Corex and 360 AI Assistant, focus on output fusion or centralized coordination, lacking three core capabilities. *(1) Agentic autonomy:* Most frameworks treat LLMs as passive generators rather than autonomous agents with perceive-reason-plan-act-learn loops; *(2) Trusted distributed collaboration:* Centralized coordination leads to single-point failure and low resistance to malicious nodes; *(3) Domain-specific optimization for networks:* General-purpose multi-LLM collaboration ignores network-specific constraints, such as power limits. In particular, in communication networks such as 6G, the open nature of wireless channels makes them inherently vulnerable to various security threats. Within such environments, it becomes difficult to determine whether devices hosting LLMs exhibit malicious activity and behavior. In this context, our Trustworthy Agentic MultiLLMN addresses these gaps by transforming LLMs into network-aware autonomous agents, integrating blockchain for decentralized trust. It also embeds 3GPP standards and network simulation tools into its reasoning processes. Thus, it can enable end-to-end intelligent and secure network optimization that existing frameworks cannot achieve.

C. Trustworthy Agentic MultiLLMN Workflow

We present how blockchain facilitates the Trustworthy Agentic MultiLLMN. Fig. 2 shows it built on a blockchain Peer-to-Peer (P2P) network. In our architecture, the nodes of this network are the Agentic LLMs hosted on the network entities, e.g., Base

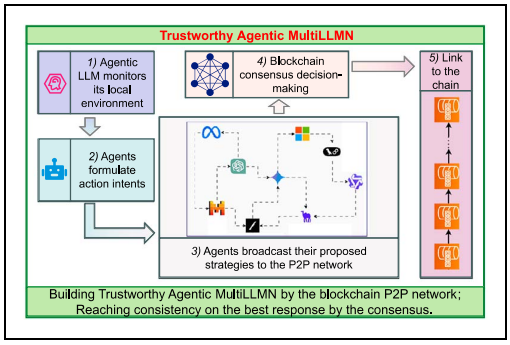


FIG. 2. The blockchain-driven Trustworthy Agentic MultiLLMN. The response provided by a certain LLM to a user will be verified and compared by all LLMs in Trustworthy Agentic MultiLLMN. Thus, it ensures that the blockchain-driven network can provide the user with the highest quality and trusted answer.

Stations. Regardless of their specific type, e.g., Llama 3, GPT-4, and Gemini, the Agentic LLMs can not only generate content but also assess and verify its accuracy. As a result, each LLM is regarded as an autonomous agent and a full blockchain node. It will contribute to the decentralized decision-making and consensus process for selecting the most suitable response to network optimization problems. The workflow is outlined as follows:

1) Distributed Perception (Perceive): Unlike centralized systems, each Agentic LLM node hosted on Legitimate Base Station (LBS) serves as a dual-role entity, namely, network agent and blockchain full node. It is authenticated via Decentralized Identifiers (DIDs) linked to LBS hardware credentials. During perception, the node's collected data, including power, interference, and FBS signatures, is hashed into a unique digest. The data is also signed with the node's private key and uploaded to the blockchain P2P network. It ensures data integrity. Any tampering with perception data, e.g., malicious LBS forging FBS-free signals, will be detected via hash mismatch during cross-node verification.

2) Intent Formulation & Reasoning (Reason): Based on perceived threats, e.g., FBS presence, the agents autonomously formulate defense intents. Each LLM agent reasons on the data, leveraging internal knowledge bases and external tools to devise local power allocation strategies. Typical knowledge bases and tools include 3GPP TS 38.213: NR[4], which defines the physical layer power control process; and MATLAB 5G Toolbox[5], which can serve as an external computing engine to simulate the impact of different wireless parameters. After generating quantitative power allocation strategies, the LLM agent encapsulates the strategy, reasoning steps, and tool utilization logs into a blockchain transaction. The transaction is signed with the node's DID and broadcast to the network. This links strategy outputs to their reasoning origins, addressing intent formulation by adding traceability.

3) Collaborative Planning (Plan): The conflict resolution between adjacent LBSs is executed via on-chain message passing. When LBS agents exchange strategy proposals, e.g., resolving total power constraint violation, each message is recorded as an intermediate transaction on the blockchain. The P2P network leverages blockchain's gossip protocol to ensure

[4] https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3213
[5] https://www.mathworks.com/products/5g.html

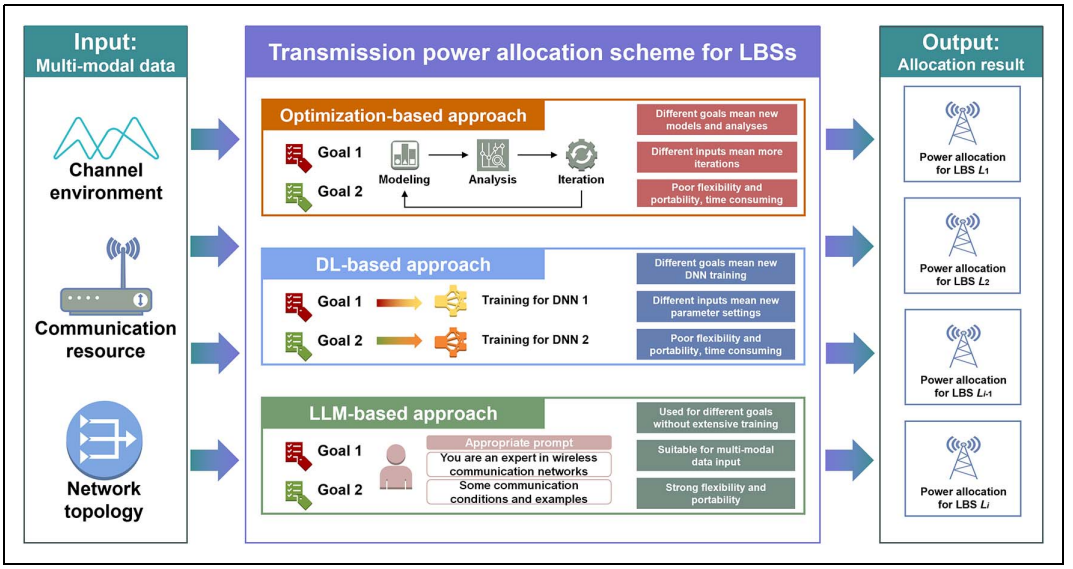


FIG. 3. Different wireless transmission power distribution methods for LBSs. Both optimization and DL-based methods suffer from poor flexibility and portability, while the LLM-based method only needs appropriate prompts to generate power allocation strategies.

message delivery. Also, transactions are time-stamped to avoid replay attacks, enhancing the reliability of collaborative planning compared to off-chain communication.

4) Blockchain Consensus (Act): To prevent malicious FBSs or compromised agents from disrupting the network, a consensus mechanism is activated. The agents vote on the validity and efficacy of the proposed power allocation map. The optimal, consensus-validated output is selected. Among them, each vote includes a hash of the proposed strategy and its reasoning digest, enabling nodes to verify consistency with on-chain intermediate transactions. Only strategies receiving $\geq 2/3$ of the votes are validated. It can reject malicious proposals, e.g., agents underpowering to facilitate FBS attacks.

5) Block Creation & Execution: Validated strategies are packaged into blocks. They contain: Final power allocation map; All intermediate transactions, including perception data digests, reasoning logs, and votes; Hash of the previous block. And blocks are replicated across all nodes, forming an immutable ledger. For adaptive learning, agents retrieve historical blocks to refine strategies, e.g., adjusting power for recurring FBS attack patterns. While blockchain ensures the authenticity of training data, unlike database-stored logs prone to tampering.

Lesson Learned: By integrating blockchain, Trustworthy Agentic MultiLLMN effectively addresses the issues of output bias, illusion, and low credibility caused by data limitations and malicious autonomous behavior. The decentralized consensus replaces the centralized coordination node. It automatically orchestrates multiple LLM agents to generate an optimal network optimization plan, such as the power allocation strategy for 6G anti-counterfeiting FBS attacks. It also avoids a single point of failure and resists the interference of malicious Byzantine agents. Specifically, in the context of FBS defense, this architecture ensures that power allocation decisions are validated by the majority, preventing a compromised LBS from lowering its power to facilitate an attack. Later, we will show how it can be applied in a real network.

III. Case Study: Trustworthy Agentic MultiLLMN-Enabled Defense Mechanism for FBS Attacks

To maintain network security, we present how to use Trustworthy Agentic MultiLLMN to enable wireless communication systems against FBS attacks as a case study. It is a typical cyber attack and has caused serious economic losses to 5G communications. In January 2023, China used 923 radio monitoring vehicles, 2,457 positioning devices, and dispatched 2,459 monitoring personnel for 37,575 hours to combat FBS attacks[6]. It is foreseeable that FBS attacks will also cause significant economic losses to 6G communication systems.

A. Motivations and LLM Advantages

We first introduce traditional optimization-based and DL-based methods. Fig. 3 shows the workflow and differences among three methods.

The first method is the optimization-based approach, which can analytically deal with clearly formulated optimization problems. Optimization problems are often complex and difficult to solve, involving integer value parameters and non-convex functions. Therefore, we often transform and simplify the problem to solve it efficiently. Then, some iterative method is used to find an optimal solution. Such methods often have problems, such as low solving efficiency. The other is a DL-based approach, which utilizes a specially designed deep neural network (DNN) to approximate the optimal scheme. However, to do so, targeted DNNs should be trained from scratch. This often takes a considerable amount of time/resources and a lot of reliable training data. Moreover, their common shortcomings are the lack of flexibility and portability. Specifically, the optimization model and the DNN are designed for specific network scenarios. When the channel environment, network topology, communication resources, etc., change, we may need to reformulate the optimization model or re-train the DNN. This means additional and significant time costs and computing overhead.

[6] https://cbgc.scol.com.cn/news/4035087

In contrast, an LLM provides a new way to implement network optimization. Since it has been pretrained on a wide range of datasets, it is not necessary to build models for specific tasks. Only a few rounds of appropriate prompts can accurately solve various optimization problems [13]. Additionally, in a complex wireless communication system, factors including multi-modal data of channel environment, communication resource, network topology, etc., often defy processing or comprehension through conventional methodologies. In this regard, LLMs offer a distinct advantage. Meanwhile, an LLM has excellent reasoning ability and can often find the underlying patterns behind complex data to generate more reasonable responses than traditional methods [14]. However, the LLM generation scheme still presents challenges in the face of high reliability and trustworthiness requirements. Especially in the FBS attack scenario, the open network environment will further introduce malicious interference to LBSs and UEs, which will affect the effectiveness of the defense scheme provided by the LLM. Thus, it is necessary to use Trustworthy MultiLLMN to prevent this attack.

B. Security Threats

In this attack, where the base station has an agent role, the 6G communication system may be subject to the following attack threats:

1) FBS Attacks: Malicious nodes forge LBS signals to hijack UEs, leading to privacy leaks and economic losses.

2) Interference Attacks: Unauthorized devices transmit high-power signals to disrupt LBS-UE communication, reducing network throughput.

3) SI Forging: Attackers: Tamper with SI, e.g., cell identity, frequency band, to mislead UEs into connecting to malicious nodes, enabling man-in-the-middle attacks.

4) Single-Point Failure Attacks: Compromising centralized controllers, e.g., in traditional multi-agent systems.

Trustworthy Agentic MultiLLMN addresses threats requiring distributed decision-making, dynamic optimization, and trusted collaboration. Specifically, the blockchain functions integrated by this framework can record real SI and policy logs, enabling user devices to verify the validity of the SI [10]; and the consensus mechanism can prevent single-point failures caused by the central controller [4]. Furthermore, for the former two attacks, in the following, we will analyze how Trustworthy Agentic MultiLLMN can defend against them through reasonable power allocation for LBSs.

C. Power Allocation Model for FBS Attacks

As described in [12], the success of an FBS attack depends on the Signal-to-Interference-plus-Noise Ratio (SINR) between the attacking FBS and the target LBS, a metric fundamentally determined by the transmit power of both devices. Therefore, for LBSs to protect a User Equipment (UE) by preventing the UE from connecting to FBSs, the wireless communication system is crucial for the transmission power allocation of each LBS under the limited total power. The transmission power of an LBS is the decisive factor in the SINR received by the UE. A higher or strategically allocated power ensures that the LBS signal strength exceeds that of the FBS, effectively drowning out the malicious signal and preventing the UE from camping on the false cell.

We can define the power allocation of i LBSs as p_1, p_3, p_3,..., p_{i-1}, p_i with $\sum_{i=1}^{n} p_i \leq p_{total}$, where p_{total} denotes the total transmission power, and n is the total number of LBSs. In general, to optimize communication performance, the total power is allocated to each LBS. Potential FBSs around LBSs, to replace LBS to bind with UE and obtain its private system information. We assume that an FBS attacks LBS L_i and successfully associates to UE, the probability is P_{FBS_i}. Then, our optimization goal is to maximize the average probability against FBS attacks through reasonable transmit power allocation for LBSs, namely $\max \frac{\sum_{i=1}^{n}(1-P_{FBS_i})}{n}$.

D. Power Allocation Based on Trustworthy Agentic MultiLLMN

To optimize the above goal under the power constraint, we propose a decentralized power allocation strategy based on Trustworthy Agentic MultiLLMN. In this framework, each LBS is equipped with an Agentic LLM, creating a distributed network of autonomous agents. The collective Agentic MultiLLMN takes the total power and the system constraints as input, and outputs the optimal power allocation for each LBS.

Specifically, the collaborative defense mechanism driven by Trustworthy Agentic MultiLLMN is shown in Fig. 4. This process transforms the LBSs from passive transmitters into active agents following the perceive-reason-act-learn loop.

1) Distributed Perception (LBS Agent): Each LBS Agent continuously monitors its local wireless environment, collecting data such as local channel state information (CSI), interference patterns, and potential FBS signatures. This decentralized data collection ensures that the network has a granular view of the threat landscape.

2) Intent Formulation (Agentic Reasoning): Based on perceived threats, each LBS Agent independently analyzes the situation and formulates a defense intent. The agent uses its internal LLM to process the multi-modal data and generate a preliminary power adjustment plan. For example, FBS presence with measured SINR = 12 dB, path loss exponent = 2.5, each LLM agent's reasoning process follows a structured prompt-embedded logic:

- *Input Data:* Local CSI, interference power 4×10^{-14} W, FBS attack probability 0.7, total power constraint 2 kW, and individual LBS power limit 80 W. *Embedded Domain Knowledge:* 3GPP TS 38.213 requires UE SINR $\geq$ 15 dB to resist FBS hijacking; MATLAB 5G Toolbox is called to simulate power-SINR relationships.
- *Quantitative Reasoning Output:* To ensure UE SINR $\geq$ 15 dB, the current power needs to be increased to 45 W.

3) Collaborative Planning (Consensus-driven): The LBS Agents exchange their preliminary plans via the blockchain network. They engage in a collaborative planning process to resolve conflicts, e.g., if multiple LBSs increasing power simultaneously violates the total power constraint P_{total}. The prompt for this collaborative reasoning can be:

- *"Agent LBS-1 reports FBS threat level High. Proposed Power: 45W; Current Total Power usage: 1.8kW; Constraint: 2kW. Neighbors, please validate and adjust your plans to maximize global defense probability."*

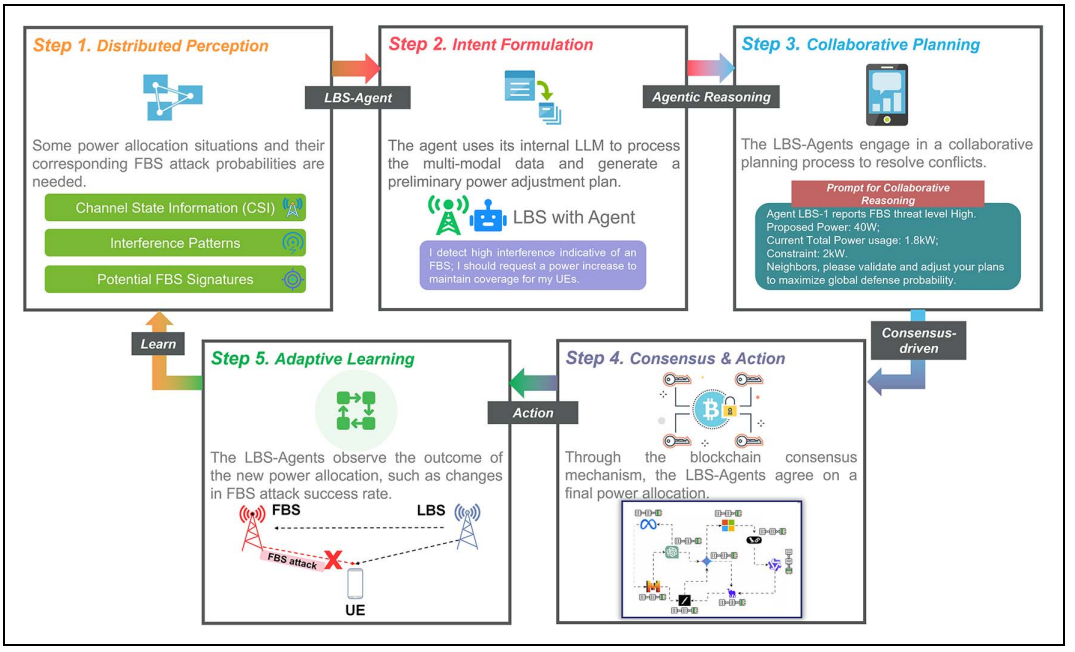


FIG. 4. Trustworthy Agentic MultiLLMN-enabled defense mechanism for the FBS attack. It can provide an LBSs power allocation method for wireless communication system to resist this attack with just a few prompts.

4) Consensus & Action (Act): Through the blockchain consensus mechanism, the LBS Agents agree on a final power allocation $\{p_1, p_2, ..., p_n\}$. This ensures that no single compromised LBS can disrupt the network stability. Upon consensus, the *Act* phase is executed via software-defined networking interfaces, such as the O-RAN E2 interface[7]. The Agentic LLM triggers the specific API call to the Radio Unit (RU) to adjust the transmission gain to the agreed level.

5) Adaptive Learning (Learn): The LBS Agents observe the outcome of the new power allocation, such as changes in FBS attack success rate. This feedback is recorded on the blockchain, allowing the Agentic MultiLLMN to refine its strategies for future attacks, creating a continuously improving defense system.

IV. Performance Evaluation

A. Simulation Setup

We first introduce the parameter settings. The parameters include both LLM-related and wireless communication system-related parameters.

For the former, we simulate a Trustworthy Agentic MultiLLMN where each of the 10 LBSs in the cluster operates as an independent agentic node. To represent the diversity of agents, we assign different LLM backends to these nodes, including Llama 3.3 (8B), WizardLM 2 (13B), GPT-4o-mini (8B), Gemini 2 Flash (34B), ERNIE Bot 4.0 (47B), SparkDesk V4.0 (180B), Qwen 2.5 (72B), Doubao pro 4k (100B), Hunyuan-Large (52B), and Kimi (32B). These agents are networked via a blockchain layer. To test the response efficiency of the blockchain-driven Agentic MultiLLMN, we set the communication environment between these LLM-Agents to bandwidth, channel capacity, and transmission rates of 80 kHz, 15 kps, and 10 kps, respectively.

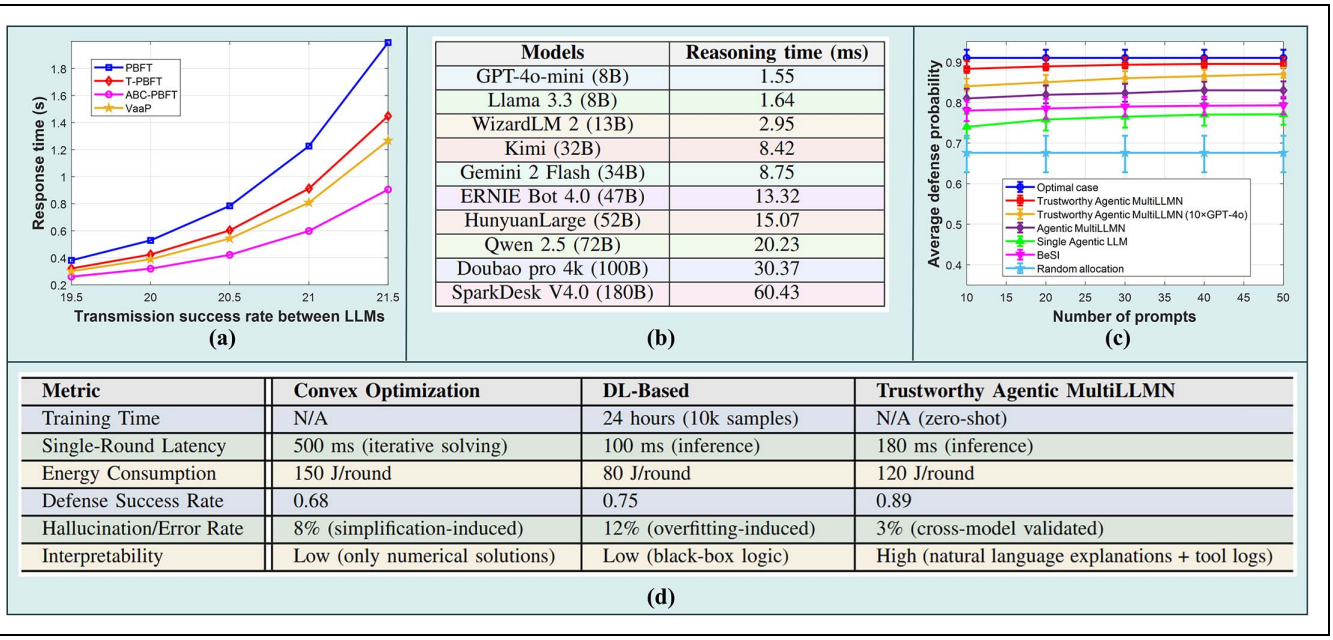
For the latter, we consider a wireless network in a circular area with a radius of 5 km, where the UE is at the center. In this case, we suppose there are 30 LBSs and 10 FBSs in the region. The wireless communication system has a total power of 2 kW and transmission power of 80 W per FBS. In addition, the path loss exponent is 2.5, the noise power of both UE and FBSs is 4×10^{-14} W, the redundancy rate is 1 bps/Hz, and the bandwidth is 20 MHz. The simulation runs on a server equipped with three 96-core Intel(R) Xeon(R) Gold 5220R CPUs, 1 TB of memory, and 8 NVIDIA GeForce RTX 3090 GPUs.

B. Response Efficiency

To verify the response efficiency of Trustworthy Agentic MultiLLMN, we compare several typical blockchain consensus mechanisms through simulations, including Practical Byzantine Fault Tolerance (PBFT), Trust PBFT (T-PBFT), Artificial Bee Colony PBFT (ABC-PBFT), and Votes-as-a-Proof (VaaP), which are described in detail in [15]. We use the delay of these consensus-driven Trustworthy Agentic MultiLLMN operations to represent their response efficiency.

As shown in Fig. 5(a), it illustrates the response time, namely the latency, of the aforementioned four consensus mechanisms in driving a Trustworthy Agentic MultiLLMN. As the transmission success rate among LLMs increases, there is a corresponding rise in the response latency. This means that it is difficult to achieve both reliability and efficiency in the wireless network environment. Notably, PBFT exhibits the highest response time, while its variant ABC-PBFT shows the lowest. T-PBFT and VaaP display intermediate levels of efficiency. The reason why ABC-PBFT has the best efficiency lies in that it screens some nodes to participate in the consensus and narrows the consensus scope. These findings offer a temporal benchmark for the operation of Trustworthy Agentic MultiLLMN in wireless environments and serve as a crucial foundation for selecting an appropriate consensus mechanism based on practical situational requirements.

[7] https://www.telecomtrainer.com/inside-the-o-ran-e2-interface-understanding-e2ap-and-e2sm-for-intelligent-ran-control/



Models	Reasoning time (ms)
GPT-4o-mini (8B)	1.55
Llama 3.3 (8B)	1.64
WizardLM 2 (13B)	2.95
Kimi (32B)	8.42
Gemini 2 Flash (34B)	8.75
ERNIE Bot 4.0 (47B)	13.32
HunyuanLarge (52B)	15.07
Qwen 2.5 (72B)	20.23
Doubao pro 4k (100B)	30.37
SparkDesk V4.0 (180B)	60.43

(b)

Metric	Convex Optimization	DL-Based	Trustworthy Agentic MultiLLMN
Training Time	N/A	24 hours (10k samples)	N/A (zero-shot)
Single-Round Latency	500 ms (iterative solving)	100 ms (inference)	180 ms (inference)
Energy Consumption	150 J/round	80 J/round	120 J/round
Defense Success Rate	0.68	0.75	0.89
Hallucination/Error Rate	8% (simplification-induced)	12% (overfitting-induced)	3% (cross-model validated)
Interpretability	Low (only numerical solutions)	Low (black-box logic)	High (natural language explanations + tool logs)

(d)

FIG. 5. (a) Response time. It compares the time required for different consensus to drive the work of Trustworthy Agentic MultiLLMN. (b) The inference time for each LLM. (c) Average defense probability. It proves the superiority of Trustworthy Agentic MultiLLMN in dealing with FBS attacks. (d) Comparison of LLM-based, DL-based, and optimization-based methods for FBS defense.

Furthermore, Fig. 5(b) presents the reasoning time required by each LLM. This time basically shows a direct proportional relationship with the size of the LLM. The total reasoning time of these 10 LLMs is 180 ms. This result indicates that although the inference delay times of different LLMs vary, the consensus stage is the bottleneck of the response time. This also suggests the importance of an appropriate consensus mechanism in driving MultiLLMN.

C. Attack Defend Performance

To evaluate the gain that Trustworthy Agentic MultiLLMN provides for wireless communication systems against FBS attacks, we compare it with homogeneous Trustworthy Agentic MultiLLMN (10 GPT-4o-mini), Agentic MultiLLMN (without blockchain consensus), a single Agentic LLM (GPT-4o-mini), a blockchain-only approach (BeSI, which leverages blockchain's immutability and traceability to protect System Information (SI) from FBS attacks [12]), optimal, and random power allocation strategies. The PBFT consensus is used to drive Trustworthy Agentic MultiLLMN. Specifically, we calculate the average probability of all LBSs in the cellular network's coverage area to protect the UE against FBS attacks, that is, the average resistance probability.

Fig. 5(c) shows the simulation results, which clearly demonstrate the advantages of adopting consensus to establish a Trustworthy Agentic MultiLLMN framework for power allocation within wireless communication networks. Here, we set two LBS Agents to have malicious behavior (simulating compromised base stations) for two configurations of Trustworthy Agentic MultiLLMN. Both approach the theoretical optimum with only partial prompts, improving operational costs and practicality. They achieve final defense probabilities of 0.895 (heterogeneous) and 0.87 (homogeneous). They outperform Agentic MultiLLMN of 0.83 as blockchain filters malicious nodes' deceptive tactics. The 2.5% gap between them is from cross-model complementarity. This heterogeneous framework fully leverages various advantages. For instance, Llama 3.3's low-latency inference capability, GPT-4o-mini's reasoning accuracy in mathematics, and ERNIE Bot 4.0's adaptability to Chinese operator data can be combined to handle complex scenarios, thereby providing reliable responses.

Furthermore, compared to a single LLM, the Trustworthy Agentic MultiLLMN can achieve an optimization effect of 16%. The BeSI approach achieves an average resistance probability between that of the single Agentic LLM and Agentic MultiLLMN. It outperforms the single LLM because its decentralized SI verification mechanism can effectively prevent UEs from being induced by fake SI broadcast by FBSs. The main reasons for the poor decision-making performance when using a single LLM are mainly due to the risk of single-point failure, the difficulty in distributed perception of each LBS's channel situation, and the vulnerability to malicious data injection. The integration of blockchain and multi-LLM can precisely avoid these drawbacks.

All LLM-based and blockchain-only methods outperform random allocation. They confirm the architecture's core gain lies in distributed collaboration and blockchain trust. The LLM-centric approach balances FBS attack resistance, while blockchain-only secures cellular initial access SI, underscoring their synergistic value for 6G security.

D. Algorithms Comparison

Finally, to make the Trustworthy Agentic MultiLLMN engineering feasible, we compared its applicability with convex optimization and DL-based methods in resisting FBS. This included the total latency, energy consumption, hallucination/error rate, and interpretability. Each method underwent 100 rounds of tests, with each round corresponding to one FBS random attack. Among them, convex optimization utilizes the CVX toolkit[8] to solve the power allocation model; the DL-based method is a 6-layer fully connected DNN, trained under the same large-scale 6G network scenario as in Section IV-A. To adapt to the

[8] https://cvxr.com/cvx/download/

high-dimensional state space (128-dimensional), the DL training adopts 10,000 samples covering diverse interference scenarios and channel realizations, with a total of 8,000 training episodes. The extended training time with 24 hours is attributed to the large-scale network topology and high computational complexity per episode with 8.8 seconds. The test results are shown in Fig. 5(d).

Compared to convex optimization and DL methods, our framework has demonstrated superior performance in all aspects. In reliability, Trustworthy Agentic MultiLLMN leverages pre-trained network knowledge and cross-model validation to reduce the hallucination/error rate to 3%. In interpretability, LLM-based reasoning outputs natural language explanations, such as "Increase power to 45 W to ensure UE SINR $\geq$ 15 dB", while convex optimization outputs only numerical solutions, and DL lacks transparent decision logic. In inference latency, our framework achieves 180 ms single-round latency, which is higher than DL's 100 ms but lower than convex optimization's 500 ms. Notably, LLMs require no task-specific training, enabling zero-shot adaptation to dynamic network changes. It addresses the key limitation of DL with 24 hours of training and convex optimization, needing re-formulation for new scenarios.

V. Future Directions

A. Dedicated Blockchain Consensus

One aspect for improvement lies in the variability of response quality from individual LLMs and the trustworthiness of the devices that host them. Thus, assigning uniform voting weights to all LLMs is unwarranted. Furthermore, alongside voting-based consensus, we propose exploring consensus models grounded in verifiable reasoning. For example, a Proof-of-Thought (PoT) mechanism could reward agents based on the verifiable quality of their contributed reasoning steps. While a deliberation-based consensus framework could enable agents to engage in a structured, rational discussion to reach a provable, unanimous agreement.

B. New Model and System for Agentic MultiLLMN

Due to the blockchain consensus and distributed storage, additional overheads are imposed on this framework. Some new technologies, such as the Mixed of Expert (MoE) model and the Secure Multi-party Computation (SMPC) system, should be further explored. They have the potential to treat each LLM as an independent expert and provide a secure collaboration mechanism for the agent-based learning network. This also includes establishing governance frameworks and standardized protocols for secure agent-to-agent interoperability. Future systems may require Decentralized Identifiers (DIDs) to provide verifiable identity for each autonomous agent.

C. Multi-Modal LLMs in Agentic MultiLLMN

In addition to defending against FBS attacks, there may also be multimodal signals in other scenarios, such as visual and audio. This is particularly relevant for autonomous network management, which often requires multi-modal data fusion to understand the complete network state. For example, in a smart city composed of autonomous aircraft, intelligent devices equipped with the Large Vision Model (LVM) can analyze and judge a collected image to provide reliable references for urban managers. To ensure the normal operation of Agentic MultiLLMN with multi-modal LLMs, we need to further design cross-modal collaboration strategies.

D. LLM-Communication Security Integration

By leveraging LLM agents' ability to perceive and analyze threat information, the framework can dynamically adjust core parameters of communication security protocols based on real-time threats. It includes encryption algorithms, authentication frequency, and key update cycles. For example, when LLM agents detect signs of an FBS attack through analyzing abnormal communication traffic and threat intelligence, they can initiate reasoning to evaluate the attack's intensity and potential impact. Then, they propose switching the current encryption algorithm to a more robust one and increasing authentication frequency from periodic to real-time verification.

VI. Conclusion

This paper presents the challenges, solutions, and a use case of Trustworthy Agentic MultiLLMN. This framework transforms LLMs into collaborative autonomous agents, enabling them to avoid hallucinations and biased results, as well as security risks such as single-point failures. Then, we have conducted a case study to demonstrate the effectiveness and excellence of this design. In this case, Trustworthy Agentic MultiLLMN can provide a power allocation strategy for 6G against FBS attacks. Numerical results have demonstrated the advantages of our proposal over others, converging notably towards the optimal solution. Finally, prospective avenues for further inquiry into governable and trustworthy autonomous agent systems have been deliberated.

References

[1] Y. Liu et al., "Generative AI in data center networking: Fundamentals, perspectives, and case study," *IEEE Netw.*, early access, Apr. 22, 2025, doi: 10.1109/MNET.2025.3563262.

[2] H. Luo et al., "Toward edge general intelligence with multiple-large language model (multi-LLM): architecture, trust, and orchestration," *IEEE Trans. Cogn. Commun. Netw.*, vol. 11, no. 6, pp. 3563-3585, Dec. 2025.

[3] J. Wen et al., "Generative AI for low-carbon artificial intelligence of things with large language models," *IEEE Internet Things Mag.*, vol. 8, no. 1, pp. 82–91, Jan. 2024.

[4] H. Luo et al., "A weighted byzantine fault tolerance consensus driven trusted multiple large language models network," *IEEE Trans. Cogn. Commun. Netw.*, vol. 12, pp. 3815–3830, 2026.

[5] S. Feng et al., "Don't hallucinate, abstain: Identifying LLM knowledge gaps via multi-LLM collaboration," 2024, *arXiv:2402.00367*.

[6] S. Wang, J. Deng, Q. Li, J. Wu, and Z. Zhao, "Performance analysis on the applications of large language models: A case for elderly care," in *Proc. IEEE Int. Conf. High Perform. Comput. Commun. (HPCC)*. Piscataway, NJ, USA: IEEE Press, 2024, pp. 145–151.

[7] S. Marro et al., "A scalable communication protocol for networks of large language models," 2024, *arXiv:2410.11905*.

[8] M. F. M. Firdhous, W. Elbreiki, I. Abdullahi, B. Sudantha, and R. Budiarto, "WormGPT: A large language model chatbot for criminals," in *Proc. 24th Int. Arab Conf. Inf. Technol. (ACIT)*. Piscataway, NJ, USA: IEEE Press, 2023, pp. 1–6.

[9] D. M. Owens et al., "A multi-LLM debiasing framework," 2024, *arXiv:2409.13884*.

[10] Y. Liu et al., "Blockchain-empowered lifecycle management for ai-generated content products in edge networks," *IEEE Wireless Commun.*, vol. 31, no. 3, pp. 286–294, Jun. 2024.

[11] H. Luo, J. Luo, and A. V. Vasilakos, "BC4LLM: A perspective of trusted artificial intelligence when blockchain meets large language models," *Neurocomputing*, vol. 599, 2024, Art. no. 128089.

[12] Z. Wang, B. Cao, Y. Sun, C. Liu, Z. Wan, and M. Peng, "Protecting system information from false base station attacks: A blockchain-based approach," *IEEE Trans. Wireless Commun.*, vol. 23, no. 10, pp. 13920–13934, Oct. 2024.

[13] S. Liu, C. Chen, X. Qu, K. Tang, and Y.-S. Ong, "Large language models as evolutionary optimizers," in *Proc. IEEE Congr. Evol. Comput. (CEC).* Piscataway, NJ, USA: IEEE Press, 2024, pp. 1–8.
[14] R. Zhang et al., "Generative AI agents with large language model for satellite networks via a mixture of experts transmission," *IEEE J. Sel. Areas Commun.*, vol. 42, no. 12, pp. 3581–3596, Dec. 2024.
[15] H. Luo, Q. Zhang, G. Sun, H. Yu, and D. Niyato, "Symbiotic blockchain consensus: Cognitive backscatter communications-enabled wireless blockchain consensus," *IEEE/ACM Trans. Netw.*, vol. 32, no. 6, pp. 5372–5387, Dec. 2024.

Biographies

Haoxiang Luo is currently working toward the Ph.D. degree with the University of Electronic Science and Technology of China (UESTC), China, and a Visiting Student with Nanyang Technological University (NTU), Singapore.

Gang Sun (Senior Member, IEEE) (gangsun@uestc.edu.cn) is a Full Professor with the University of Electronic Science and Technology of China (UESTC), China.

Yinqiu Liu is currently working toward the Ph.D. degree with Nanyang Technological University (NTU), Singapore.

Dusit Niyato (Fellow, IEEE) is a Full Professor with Nanyang Technological University (NTU), Singapore. He is serving as an Editor-in-Chief of IEEE Transactions on Network Science and Engineering.

Hongfang Yu (Senior Member, IEEE) is a Full Professor and Vice Dean with the University of Electronic Science and Technology of China (UESTC), China.

Mohammed Atiquzzaman (Senior Member, IEEE) is an Edith Kinney Gaylord Presidential Professor with the University of Oklahoma, USA. He is serving as Editor-in-Chief of the *Journal of Network and Computer Applications* and the Editor-in-Chief of *Vehicular Communications*.

Schahram Dustdar (Fellow, IEEE) is a Full Professor with the Technische Universität Wien (TU Wien), Austria, and an Elected Member of the Academia Europaea, where he is the Chairman of the Informatics Section. He is also the President of AIIA (International AI Industry Alliance).