Task-Oriented 6G Native-AI Network Architecture

Yang Yang[0,1,2,↑], Jianjun Wu[3], Tianjiao Chen[4], Chenghui Peng[3], Jun Wang[3], Juan Deng[4], Xiaofeng Tao[1,5], Guangyi Liu[4], Wenjing Li[5], Li Yang[6], Yufeng He[7], Tingting Yang[1], A. Hamid Aghvami[8], Frank Eliassen[9], Schahram Dustdar[10], Dusit Niyato[11], Wanfei Sun[12], Yang Xu[13], Yannan Yuan[14], Jiang (Linda) Xie[15], Rongpeng Li[16], Cuiqin Dai[17] .

0. The Hong Kong University of Science and Technology (Guangzhou), China
1. Peng Cheng Laboratory, China
2. Terminus Group, China
3. Huawei Technologies Co., Ltd., China
4. China Mobile Research Institute, China
5. Beijing University of Posts and Telecommunications, China
6. ZTE Corporation, China
7. Research Institute of China Telecom
8. King's College London, UK
9. University of Oslo, Norway
10. TU Wien, Austria
11. Nanyang Technological University, Singapore
12. CICT Mobile Communication Technology Co., Ltd., China
13. Beijing OPPO Telecommunications Corp., Ltd., China
14. vivo Mobile Communication Co., Ltd., China
15. University of North Carolina at Charlotte, USA
16. Zhejiang University, China
17. Chongqing University of Posts and Telecommunications, China

Abstract

The vision for 6G networks is to offer pervasive intelligence and internet of intelligence, in which the networks natively support artificial intelligence (AI), empower smart applications and scenarios in various fields, and create a "ubiquitous-intelligence" world. In this vision, the traditional session-oriented architecture cannot achieve flexible per-user customization, ultimate performance, security and reliability required by future AI services.

In addition, users' requirements for personalized AI services may become a key feature in the near future. However, the traditional AI deployment based on cloud/mobile edge computing (MEC) has limitations such as low throughput, long delay, poor privacy, and high carbon emissions, resulting in the inability to provide personalized quality of experience (QoE) assurance. By integrating AI in the network, the network AI has more advantages than cloud/MEC AI, such as better QoS assurance, lower latency, less transmission and computing overhead, and stronger security and privacy. Therefore, this article proposes the task-oriented native-AI network architecture (TONA), to natively support the network AI. By introducing task control and quality of AI services (QoAIS) assurance mechanisms at the control layer of 6G, the TONA can achieve the finest service granularity at the task level for guaranteeing every user's personalized QoE.

1 Introduction

The traditional communications system is session-oriented and typically provides connections between specific terminals or between terminals and application servers. Its network architecture offers a complete lifecycle management mechanism (such as creation, modification, deletion, and anchor transfer of end-to-end (E2E) communication tunnels) and quality of service (QoS) assurance for sessions, aiming to provide connections for data transmission, support user mobility, and ensure user experience. To achieve the 6G vision of pervasive intelligence and internet of intelligence, support for native AI at the 6G network architecture level is necessary [1] [2] [3]. Unlike the traditional communication services, AI is a data- and computing-intensive process, which requires ubiquitous distribution, high real-time performance, and high security and privacy — aspects that 6G needs to support.

To achieve the 6G vision of pervasive intelligence and internet of intelligence, the challenges and solutions have been provided in lots of articles [4] [5] [6] [7], they think the network architecture in 6G requires significant transformation compared to traditional communications systems, as normally the system architecture can only be re-designed in the

Yang Yang and Jianjun Wu are the corresponding authors.

initial stage of one generation of radio network, and will keep unchanged in later releases. Such transformation will involve the introduction of new resources — computing, data, and algorithms — required by AI, and the design of real-time management and control mechanisms to support multi-node collaboration and heterogeneous resources collaboration, as well as security and privacy mechanisms for distributed AI workflows across multiple nodes (including terminals and network nodes).

Based on the proceeding transformation, this article further proposes a task-oriented native-AI network architecture (TONA) to meet personalized AI service demand and requirements. This article mainly:

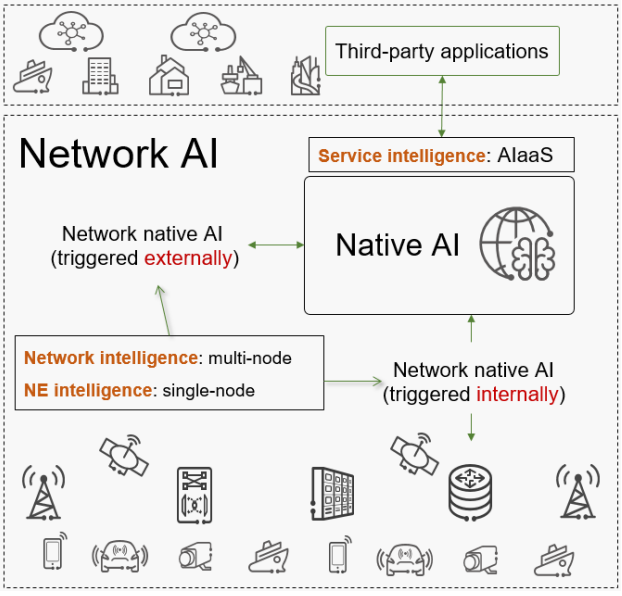
(1) Introduces three-layer logical architecture of task management and control system, and designs the task lifecycle management procedures, which include the collaboration of multi-dimension heterogeneous resources (communication, computing, data, and algorithm) and multi-node at the control layer.
(2) Defines task-specific QoAIS indicators for the mapping from Service Level Agreement (SLA) indicators — e.g., service requirement zone (SRZ) and user satisfaction ratio (USR) — to QoAIS indicators, and discusses task-level QoS assurance to meet individual requirements of different users.
(3) Compares the network AI and cloud/mobile edge computing (MEC) in terms of QoAIS indicators. Thanks to providing the AI executing environments closer to UE, TONA is anticipated to have some advantages, such as better data privacy protection, lower latency, and lower energy consumption.
(4) Lists some open issues, including distributed AI learning, mobility management, and security assurance.

2 Network AI

The cloud AI architecture has been widely used in the 5G era to provide centralized computing, big data analysis, and AI training and inference services, where terminals provide data, mobile networks provide communication channels, and clouds provide AI capabilities. Coordinating these independent functions and resources among multiple facilities provides effective, flexible, smooth, and stable services and ensures QoE is extremely difficult. For latency-sensitive ultra-reliable low-latency communication (URLLC) services, MEC deploys application servers close to base stations and therefore has lower latency than cloud AI. However, the AI platform is still deployed at the application layer. Joint optimization of connection and AI resources (i.e., computing, data, model/algorithm) still requires cross-layer collaboration in MEC. Consequently, the preceding problems involved in cloud AI remain unresolved.

Deploying AI functions (such as cloud and MEC) at the application layer leads to low throughput, high latency, poor privacy, and high carbon emissions. To address these problems, the network AI is launched to extend computing from the cloud to physically closer edges to end users. It also provides data storage and processing functions as well as AI capabilities inside the network, achieving higher security. Although this "device-edge-cloud" architecture with edge cloud is expensive to deploy, it can support compute-intensive, latency-sensitive, security-assured, and privacy-sensitive applications such as interactive virtual reality (VR) and augmented reality (AR) games, autonomous driving, and smart manufacturing [7]. Therefore, it is becoming promising in various high-value-added application scenarios.

By introducing AI in the network, 6G network AI applies to three scenarios (shown in Figure 1): Network element (NE) intelligence, network intelligence, and service intelligence. NE intelligence is the native intelligence of single nodes, e.g., core network (CN) or radio access network (RAN) nodes. Network intelligence refers to the collaboration of multiple intelligent NEs to achieve swarm intelligence. Both NE intelligence and network intelligence can be triggered internally or externally via open interface. Moreover service intelligence refers to the 6G network AI being provided as a service, which is generally triggered by external services and implemented in the network, without understanding the application service logic. Put simply, NE intelligence and network intelligence provide AI services for internal network modules, and service intelligence provides AI services for external third-party applications. Here, we assume that some network units like base stations and UEs will have some type of AI processor which can be used for themselves and the third parties.



Yang Yang and Jianjun Wu are the corresponding authors.

Fig.1 Scenarios and requirements of 6G network AI

To support the three scenarios, the 6G native AI network architecture should have a unified framework for different types of AI training and inference. For example, a distributed AI environment must be built on the 6G network. Specifically, the 6G native AI network architecture must be able to: (1) use various native AI capabilities (e.g., connection, computing, data, and AI training and inference capabilities) of NEs and terminals; (2) provide on-demand AI, computing, and data services for networks and third-party applications; and (3) guarantee the QoAIS in heterogeneous, dynamic, fully distributed, and other complex wireless environments. This is the reason that our proposed solution shift from a session-oriented to a task-oriented architecture to address the preceding challenges.

3 Network Paradigm Change

The TONA, as shown in Figure 2, introduces the orchestration and control functions as well as the resource layer in network AI. The control function uses control layer signaling to control multi-nodes (UEs, base stations, and CN NEs) and heterogeneous resources in real-time.

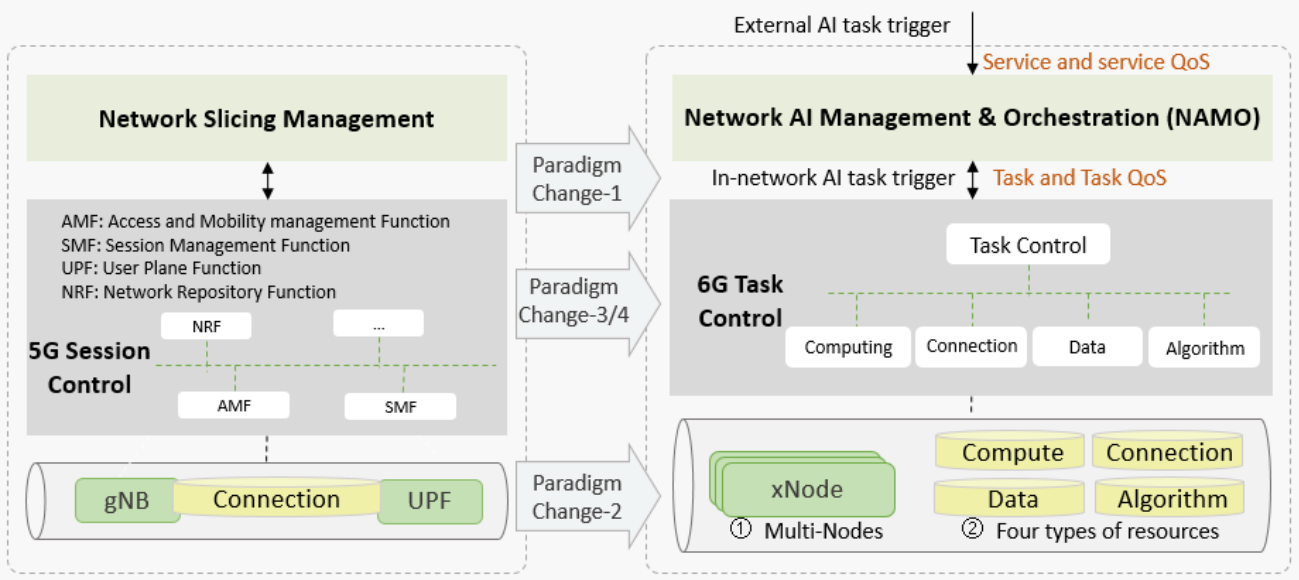


Fig.2 Network paradigm changes

We believe that the 6G network architecture requires the following changes in the design paradigm:

(1) Change 1: The object to be managed and controlled in network are changed from sessions to tasks.
(2) Change 2: The resources of the object are changed from one dimension to multi-dimensions, from homogeneous to heterogeneous.
(3) Change 3: The object control mechanism are changed from session-control to task-control.
(4) Change 4: The performance indicators of the object are changed from session-QoS to task-QoS.

3.1 Change 1: From Session to Task

AI tasks differ from traditional sessions in terms of technical objectives and methods.

In terms of technical purposes, a traditional communications system provides session services, typically between terminals or between terminals and application servers, to transmit user data (including voice). Conversely, network AI (i.e., NE intelligence and network intelligence) aims to provide intelligent services for networks and improve communication network efficiency. Service intelligence seeks to provide app-specific intelligent services for third parties. Thus, sessions and AI tasks have different purposes.

In terms of technical methods, to transmit user data, a traditional communications service needs to maintain a QoS assurance mechanism for user-oriented connection channels as well as their lifecycle management, such as E2E tunnels from UEs to base stations and then to the CN. This is necessary to provide QoS guarantee for the data transmission. Conversely, AI is a data- and computing-intensive service. Compared with sessions, AI introduces new resources, including computing (e.g., CPU, GPU, and network processing unit (NPU)), data (generated or used by AI), and algorithms (e.g., neural network models and reinforcement learning). Thus 6G networks need to introduce new resource management mechanisms. However, it is difficult to efficiently implement AI services on a single node due to the bottlenecks in single-point computing, data privacy protection, and ultra-large model storage. Consequently, a new collaboration mechanism in 6G networks is required to implement computing,

Yang Yang and Jianjun Wu are the corresponding authors.

algorithm, and data collaboration among multiple nodes. Hence, sessions and AI tasks have different technical methods.

These differences show that the session-oriented system cannot support native AI and that a new task-oriented system needs to be designed for the new resource management mechanism and multi-node collaboration mechanism. This article defines a task that coordinates multi-node and multi-dimensional resources at the 6G network layer to achieve a given objective. For example, a federated learning in network needs the coordination of multiple nodes of the base station and multiple UEs, and the coordination of communication, AI model, and computing resources.

3.2 Change 2: From single-dimension to multi-dimension heterogeneous resources

The traditional wireless system establishes tunnels and allocates radio resources for data transmission. Conversely, TONA implements collaboration among heterogeneous resources of connection, computing, data and model/algorithm to execute AI tasks. Take an AI inference task as an example. In this case, executors need to obtain certain resources to execute the tasks. Specifically, the executors need to obtain computing resources like computing timeslots for the tasks, data resources like the data collected in real-time or external data input, and algorithm resources including a possible AI model such as a graph neural network (GNN), a convolutional neural network (CNN), or reinforcement learning.

3.3 Change 3: From Session-control to Task-control

Unlike session control, task management and control in network AI includes the following functions: (1) Decomposing and mapping from external services to internal tasks, (2) Decomposing and mapping from service QoS to task QoS, and (3) Providing heterogeneous and multi-node collaboration mechanisms to orchestrate and control heterogeneous resources of multiple nodes at the infrastructure layer in real-time (to implement distributed serial or parallel processing of tasks and real-time QoS assurance). For a simple service request, one service may correspond to or be mapped as one task. For a complex service request (e.g., integration of multiple service flows, or a service flow with numerous calculations), one service may be mapped to multiple nodes for systematic execution.

For function (3), the execution of an AI task requires collaboration in two dimensions:

Heterogeneous resources collaboration: The execution of a task may require some or all of the heterogeneous resources. For example, task deployment requires configuring the heterogeneous resources, and task execution requires scheduling the heterogeneous resources in real-time.

Multi-node collaboration: First, in a traditional communications network, connection-specific computing is mainly implemented on a single NE, and computing sharing and collaboration are not required. The emergence of AI is accompanied by large-scale AI training, large-model AI inference, and massive perceptual image processing, requiring significantly more computing than traditional networks do. Simply expanding the computing capability of each NE across the entire network will result in high deployment costs. Hence, distributed computing is needed, which completes a task collaboratively among multiple nodes through shared computing. Second, as data ownership awareness grows, data privacy protection requirements become more stringent. For example, the raw data of User Equipment (UE) cannot be uploaded to networks for training. Federated learning solves this problem through collaborative learning and gradient transfer at the data layer among multiple nodes. Third, model training consumes substantial computing and storage resources to support native AI, and thus a good model needs to be shared within the network, and collaboration for models among multiple nodes is required.

3.4 Change 4: From Session-QoS to Task-QoS

Unlike previous generations of mobile networks, 6G networks are not just channels that serve traditional communications services. Different AI scenarios have different requirements for AI service quality. They demand an indicator mechanism to quantitatively or hierarchically convey user requirements while also orchestrating and controlling the comprehensive effect of AI resources. Therefore, this article proposes the quality of AI service (QoAIS).

The QoS of traditional communication networks mainly considers connection-specific performance indicators such as latency and throughput of communication services [8]. In addition to these traditional communication resources, 6G networks will introduce new resources such as computing, algorithm, and data, requiring an extension of evaluation indicators. At the same time, with the implementation of "Carbon Neutrality" and "Peak Carbon Dioxide Emissions" policies, the global AI industry's attention on data security and privacy, and users' increasing requirements for network autonomy, users will focus on more than just performance indicators in the future. The requirements on aspects such as overhead, security, privacy, and autonomy will increase, and these aspects will become new dimensions for evaluating QoS. Consequently, the QoAIS indicator system needs to be extended from the existing indicators during the initial design [9].

For example, the QoAIS indicators for AI training services are as follows:

Yang Yang and Jianjun Wu are the corresponding authors.

(1) **Efficiency**: efficiency indicator boundary, training duration, generalization, reusability, robustness, explainability, consistency between the loss function and optimization objective, and fairness
(2) **Overhead**: storage overhead, computing overhead, transmission overhead, and power consumption
(3) **Security**: storage security, computing security, and transmission security
(4) **Privacy**: data privacy, and algorithm privacy
(5) **Autonomy**: fully autonomous, partially autonomous, and manually controllable

QoAIS is an essential input for the network AI orchestration and management system and control functions. The orchestration and management system decomposes and maps QoAIS to generate QoS requirements of AI tasks, and then maps the task QoS to QoS requirements of multi-dimensional heterogeneous resources. The management, control, and user plane mechanisms are designed to ensure continuous QoAIS assurance.

4 Architecture and Key Technologies

This section describes the logical architecture and deployment options of TONA, and QoAIS details.

4.1 Logical Architecture of TONA

First, we introduce fundamental basic concepts in wireless network. A communications system consists of a management domain and a control domain. The Operations Administration and Maintenance (OAM) deployed in management domain is used to operate and manage NEs through non-real-time (usually within minutes) management plane signaling. The control domain is deployed on core network (CN) NEs, base stations, and terminals, and features with real-time controlling signaling (usually within milliseconds). For example, an E2E tunnel for a voice call can be established within dozens of milliseconds by control signaling.

Unlike the centralized, homogeneous, and stable AI environment provided by the cloud, the network AI faces the following technical challenges when embedded in the wireless networks: (1) AI needs to be distributed on numerous CN NEs, base stations, and UEs. Therefore, it is necessary to consider how to manage the massive number of nodes efficiently in the architecture design. (2) The computing, memory, data, and algorithm capabilities of different nodes vary significantly, requiring the architecture design to also consider how to efficiently manage these heterogeneous nodes efficiently. (3) The dynamic variation of the channel status and the computing load need to be factored into the architecture design.

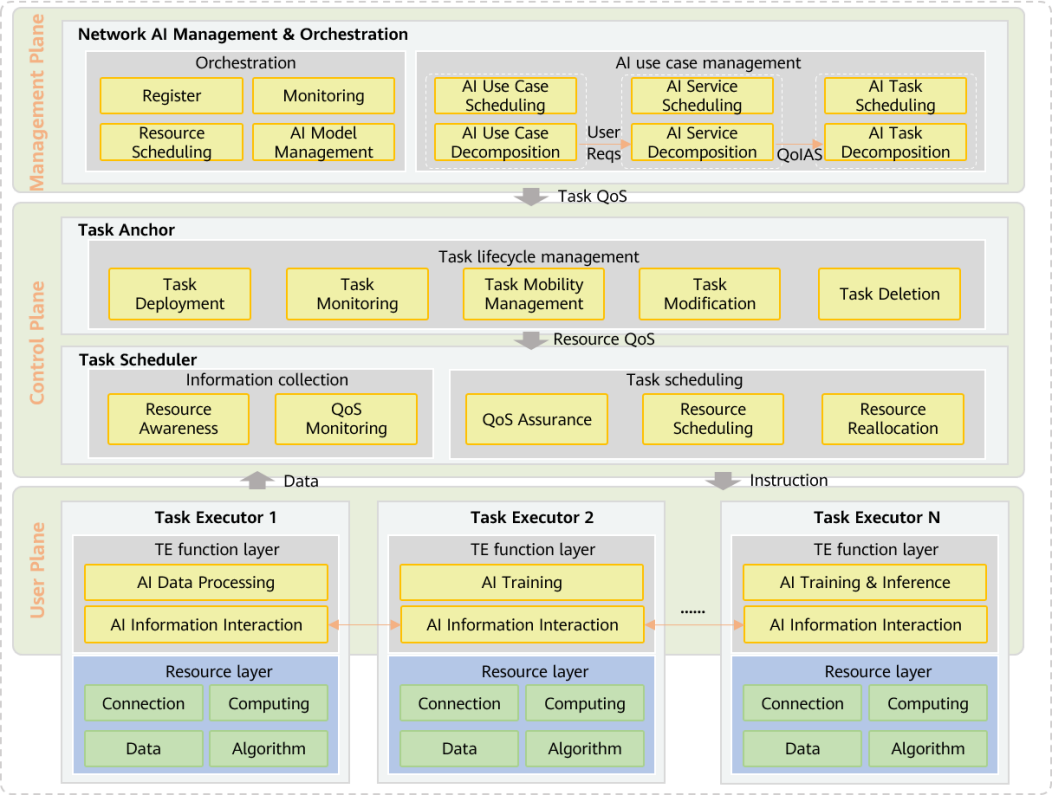


Fig.3 Logical architecture of TONA

Yang Yang and Jianjun Wu are the corresponding authors.

To address the aforementioned challenges, TONA includes two logical functions, as shown in Figure 3: (1) AI orchestration and management, called Network AI Management & Orchestration (NAMO); and (2) task control. NAMO decomposes and maps AI services to tasks and orchestrates the AI service flows. It is not performed in real-time and is generally deployed in the management domain. Task control introduces the Task Anchor (TA), Task Scheduler (TS), and Task Executor (TE) functions in the control domain in three layers. This layered design strikes a balance between the task scope and real-time task scheduling, and effectively manages the numerous, heterogeneous nodes and aware of dynamic change of heterogeneous resources (e.g. channel status and computing load).

The following describes the detailed functionalities of TA, TS, and TE.

TA manages the lifecycle of tasks (including deploying, starting, deleting, modifying, and monitoring tasks) based on task QoS requirements. It also implements collaboration among heterogeneous resources to guarantee coarse-grained QoS in the initial deployment phase.

TS controls and schedules tasks in the task execution phase. It consists of the information collection and resource management modules. Information collection requires that TS senses the computing load, data processing capabilities, algorithm models being used, and channel conditions on a plurality of nodes in real-time. Based on this information, TS has a more real-time resource management capability than TA. For example, when the network environment changes, TS adjusts AI models and data processing functions or schedules connection and computing resources in real-time to achieve timely QoS assurance.

TE is responsible for task execution and possible service data interaction. For example, federated learning needs to transfer intermediate gradient information among multiple nodes.

4.2 Deployment Architectures

The statuses of TEs (e.g., the CPU load, memory, electricity, and UE channel status) change in real-time. As such, deploying TA and TS close to each other can reduce the management delay. According to the design logic of wireless networks, the CN and RAN need to be decoupled as much as possible. For example, the CN should be independent of RAN Radio Resource Management (RRM) and Radio Transmission Technology (RTT) algorithms. Therefore, this article recommends that TA/TS be deployed on RAN and CN, named RAN TA/TS and CN TA/TS, respectively. This way will allow TA to manage TEs in real-time flexibly. Four deployment scenarios of TONA are shown in Figure 4 to describe the necessity and rationality of CN TA and RAN TA. These scenarios are only examples — there may be other deployment scenarios and architectures.

Assume that TA, TS, and TEs are deployed in RAN to perform federated learning between the base station and UEs. Considering the 6G architecture is undetermined, this article reuses the 5G RAN architecture for reference. A gNodeB is a 5G base station, which can be deployed in standalone mode or by separating the centralized unit (CU) from the distributed units (DUs). In the latter mode, the CU may be deployed on the cloud for non-real-time signaling control and data transmission. The DUs may be deployed closer to UEs for real-time resource allocation, data transmission and retransmission.

Yang Yang and Jianjun Wu are the corresponding authors.

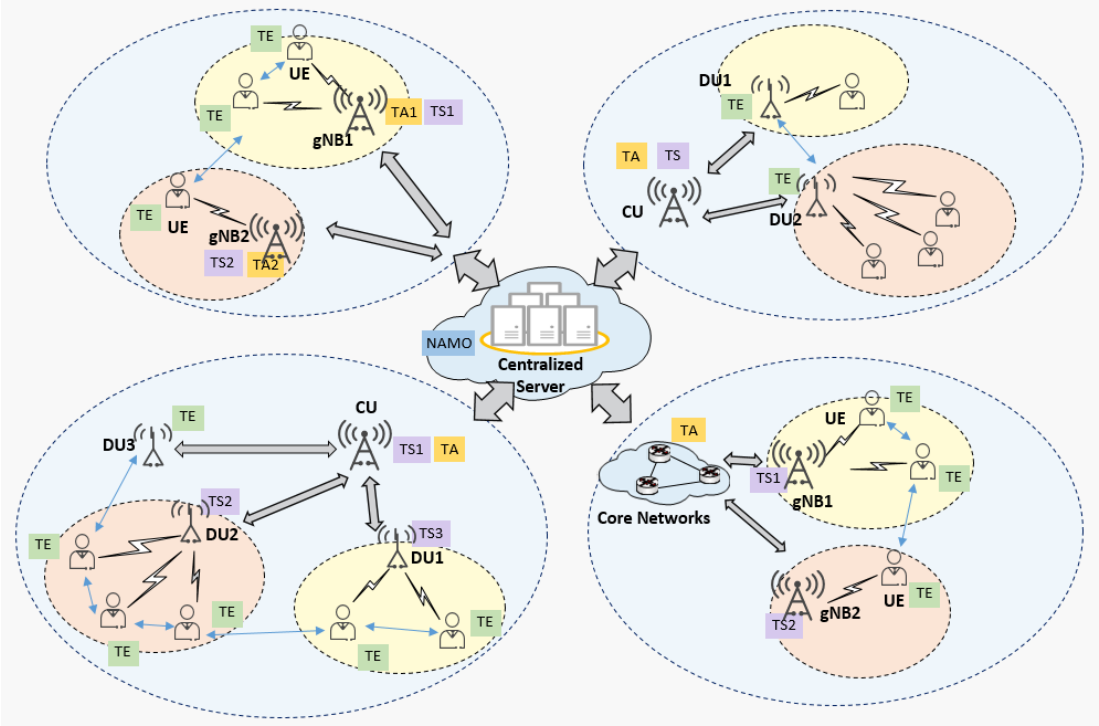


Fig.4 Four deployment scenarios of TONA

Scenario 1: gNodeB + UEs. In this scenario, the gNodeB serves as both TA and TS, and the UEs serve as TEs. Here, a UE is a computing provider and task executor, which accepts task assignment and scheduling from the gNodeB. The Uu interface and Radio Resource Control (RRC) layer between the gNodeB and the UE can be enhanced to support task controlling and scheduling purposes.

Scenario 2: CU + DUs. In this scenario, the CU serves as both TA and TS, and the DUs serve as TEs. Here, a DU is the computing provider and task executor. The F1 interface and F1-AP layer between the CU and the DU can be enhanced to support task controlling and scheduling purposes.

Scenario 3: CU + DUs + UEs. In this scenario, the CU serves as TA, the DUs as TSs, and the UEs as TEs. Here, a UE is a computing provider and task executor, and the CU is the task manager. A DU observes a task allocated by the CU to UEs, and performs heterogeneous resources scheduling and real-time QoS guarantee. This scenario separates TA from TSs. TSs are deployed lower than TA is; TSs can therefore acquire the status of TE heterogeneous resources more quickly to achieve real-time task QoS monitoring and rapid adjustment of heterogeneous resources. The Uu interface and RRC/Medium Access Control (MAC) layer between the CU/DU and the UE can be enhanced to support task controlling and scheduling purposes.

Scenario 4: CN + gNodeB + UEs. In this scenario, the CN serves as TA, the gNodeB serves as TS, and the UEs serve as TEs. Here, a UE is the computing provider and task executor. The Non Access Stratum (NAS) interface and NAS layer between the CN and the UE can be enhanced to support task controlling purposes, and the Uu interface and RRC/MAC layer between the gNodeB and the UE can be enhanced to support task scheduling purposes.

In this example, TA, TS, and TE are only logical functions, which may be deployed on the same or different nodes depending on the scenarios. Logically, a single node may have multiple functions (any combination of TA, TS, and TE).

4.3 Task QoS Assurance

To guarantee QoAIS, the aforementioned hierarchical management and control architecture is implemented through three-layer closed-loop management. The TS layer monitors and optimizes the heterogeneous resources in real-time to ensure task QoS within TA resource configurations. When the task QoS guarantee is beyond the range of the TS layer (e.g. the computing resource controlled by the TS is not sufficient for a task, the TS should report to TA to allocate more computing resource to guarantee the QoS of this task), the TA layer modifies the overall resource configuration. For example, the TA layer adjusts the network nodes involved in the task and replaces the model or data warehouse. When the task QoS guarantee is beyond the range of the TA layer, NAMO performs optimization by changing the anchor position of an AI task or decomposing the mapping between AI services and AI tasks.

Yang Yang and Jianjun Wu are the corresponding authors.

Table 1 lists the mapping between the QoAIS and resource QoS indicators. The QoAIS indicators are decomposed into task QoS indicators and then mapped to resource QoS indicators, which jointly guarantee the QoAIS at the management, control, and user planes. The QoS indicators in each resource dimension are classified into indicators suitable for quantitative evaluation (such as resource overheads) and qualitative evaluation (such as security level, privacy level, and autonomy level). For the indicators suitable for quantitative evaluation, the quantization solutions are mature or easy to formulate, such as training duration, algorithm performance boundary, computing precision, and resource overheads. However, some other indicators (e.g., model robustness, reusability, generalization, and explainability) cannot be evaluated quantitatively. Therefore, we need to consider indicators that reflect user requirements and introduce them by phase.

Table 1. Mapping between QoAIS indicators and resource QoS indicators in the AI training service

QoAIS Indicator	Resource-specific	Quantitative Indicator	Non-quantitative Indicator
Performance indicator boundary, training duration, generalization, reusability, robustness, explainability, consistency with optimization objective, and fairness	Data	Feature redundancy, integrity, data accuracy, and data preparation duration	Sample space balance, integrity, and sample distribution dynamics
	Algorithm	Performance indicator boundary, training duration, convergence, and optimization objective matching degree	Robustness, reusability, generalization, explainability, and fairness
	Computing	Computing precision, duration, and efficiency	None
	Connection	Bandwidth and jitter, delay and jitter, bit error rate and jitter, and reliability	None

5 Task Procedures

From the E2E procedure perspective, NAMO submits the AI service request to TA for execution after receiving an external service request. The E2E procedure is as follows:

(1) Generate or import an AI use case, which is an AI service request submitted by a user to the network. This use case may call one or more types of network AI services, such as AI training, verification, and inference.
(2) Decompose the use case into one or more AI services.
(3) Decompose an AI service into one or more AI tasks (AITs), and decompose the AI service QoAIS into the AI task QoS.
(4) Determine the anchor position of an AIT.
(5) Decompose the task QoS into resource QoS requirements, and specify the heterogeneous resources required by the AIT.
(6) Determine and configure the heterogeneous resources required by the AIT. This involves selecting nodes (those that participate in computing and provide data and algorithms/models), establishing connections between nodes, and updating the configurations.
(7) Among the selected nodes, determine and adjust the computing allocation in real-time, optimize the communication connection quality, collect the required data, and replace or optimize the algorithms/models. This is necessary to ensure the task QoS and further guarantee the QoAIS.

The management layer has poor real-time performance. Although it can obtain a wide range of network information, such information is coarse-grained. Furthermore, the management layer cannot obtain real-time information about radio links and terminal resources. Conversely, the control layer has good real-time performance. However, while it can obtain accurate information, the range of this information is limited. Hence, some functions are suitable for the management or the control layer, and others may be achieved through the collaboration of both layers.

6 Advantage Analysis

Compared with cloud/MEC AI, the TONA and QoAIS have the following advantages (summarized in Table 2) in meeting users' customized AI service requirements:

(1) QoAIS assurance

Dynamic wireless environments require joint optimization of the heterogeneous resources (connection and three AI resources) to achieve precise QoAIS assurance. In TONA, all heterogeneous resources are inside the network and can perceive each other. Furthermore, a real-time (within milliseconds) collaboration mechanism is

Yang Yang and Jianjun Wu are the corresponding authors.

designed at the control layer. Conversely, the cloud/MEC AI lacks a collaboration mechanism between the communication resources and the three AI resources, meaning that these resources cannot observe each other in real-time. Generally, they observe each other through the management layer (with non-real-time capability openness) or the application layer (within seconds or minutes), which cannot adapt to dynamic wireless environment changes in real-time, and cannot guarantee QoAIS.

Take device-cloud joint AI training as an example [10] [11], the Cloud AI cannot be aware of the connection's real-time status to adjust the heterogeneous resources and thus cannot provide customized training solutions for users with different connection performances. Meanwhile, for TONA, the network detects environment changes (such as terminal movement, disconnection, and burst interference) in real-time and quickly adjusts the joint training solution. For example, in TONA the network can change the split learning [12] [13] point to reduce the intermediate data size when the UE is far away from the base station. Thus, QoAIS can be achieved for customized AI training services.

Take device-cloud computing offloading as an example. If a terminal's local computing resource does not meet the requirements of computing-intensive services, cloud/MEC AI offloads some computation to the cloud. During the execution of computing tasks, the computing resource utilization of terminals changes in real-time (within milliseconds). The non-real-time collaboration of cloud AI (within seconds or minutes) cannot trace users' computing requirements in real-time, nor can it promptly offload computing to the cloud. As such, this approach fails to meet users' customized QoAIS requirements. On the other hand, for TONA, during task execution, TONA can detect the dynamic changes of computing loads on terminals in real-time and promptly adjust the computing resource allocation, calculation precision, and serial or parallel computing mode on the network. As such, this approach can ensure QoAIS for customized computing offloading services.

(2) Latency

TONA computing is distributed on NEs closer to UEs or even directly on UEs to process data locally. This not only successfully achieves real-time and low-latency AI services, but also significantly reduces data transmission. In the cloud/MEC AI mode, a large amount of data needs to be transmitted to the cloud/MEC for training, meaning that E2E data transmission takes longer to complete.

Take joint device-cloud AI inference as an example [14] [15]. Cloud/MEC AI transmits data from devices to the cloud, performs real-time training/interference, and transmits the results back to devices. The long transmission distance causes high latency, making it difficult to meet the requirements of ultra-low latency scenarios such as Industrial Internet of Things (IIoT), even if the application server is deployed on the MEC. By contrast, for TONA, data processing is terminated within a network, the E2E transmission latency is as low as 1 millisecond, enabling ultra-low latency.

(3) Overhead

TONA can optimally allocate resources through the real-time collaboration mechanism of the heterogeneous resources, maximizing the overall resource utilization and reducing the transmission and computing overheads. Conversely, because the cloud/MEC AI cannot adapt to dynamic environments, it allocates resources based on only the maximum resource consumption to ensure QoAIS. As a result, the overall resource utilization is low, and the resource overhead is high.

Take joint device-cloud AI training as an example. For Cloud/MEC AI, long device-cloud distance causes large transmission overheads. On the other hand, for TONA, data is processed nearby, effectively reducing data transmission overheads.

Furthermore, Cloud/MEC AI cannot measure quality of wireless connections in real-time. Different connections status of TEs lead to low AI efficiency and increase computing overhead. In federated learning, for example, straggler terminals may be abruptly disconnected from the network or cause a long delay. If a large amount of straggler data is discarded, the number of training samples is reduced, affecting the convergence efficiency of the current round. If a long delay occurs, the iteration time of the current round is prolonged. However, for TONA, the network can detect each UE's channel status and set a longer local training period for the straggler to reduce the total reporting numbers when the UE's data rate is low. This improves the overall AI training efficiency and reduces the computing overhead.

(4) Security

TONA has native data security and privacy protection capabilities because it processes data inside the network. Unlike TONA, the cloud/MEC AI protects data privacy only at the application layer.

Table 2. Performance comparison of cloud/MEC AI and TONA

	Cloud/MEC AI	**TONA**
QoAIS	Difficulty to guarantee the personalized QoAIS.	Easy to guarantee the personalized QoAIS.
Latency	Higher latency due to out-network processing (e.g. second/minute level).	Lower latency due to in-network processing (e.g. millisecond level).

Yang Yang and Jianjun Wu are the corresponding authors.

Resource overhead	Larger transmission overheads.	Less transmission overheads.
	Larger computation overheads.	Less computation overheads.
Security	Data privacy is ensured by the application layer.	Native security and privacy via in-network processing.

7 Open Research Issues

Although the industry has reached a preliminary consensus on 6G native AI networks, some efficient support and standardization methods need further research and development.

(1) **Distributed AI learning:** Distributed AI learning involves collaboration and interaction among multiple TEs. The definition of the collaboration mechanisms and interaction information vary according to algorithms. Therefore, we need to study how to natively and efficiently support distributed AI from the architectural perspective.

(2) **Mobility:** Assuming a UE participates in the task process (e.g., the UE is a participant in task execution), when the UE moves from one base station to another, how to guarantee the service continuity for AI tasks is a critical problem (e.g., how to achieve zero-millisecond interruption).

(3) **Security assurance:** Considering the distributed deployment of TEs in TONA, the access of multiple heterogeneous devices and distributed AI learning pose significant challenges to network AI security. Therefore, for the distributed communication and learning of TEs, the implementation of port monitoring, privacy protection and security isolation is an important research direction to ensure the security of TONA.

8 Conclusion

To meet the 6G vision of pervasive intelligence and internet of intelligence, TONA is proposed to support efficient collaboration of heterogeneous resources and multi-node in wireless networks, and to provide new services in the form of tasks at the network layer. By bringing new dimensions of resources to 6G networks (i.e., computing, data, and model/algorithm), this architecture enables the SLA assurance of computing related services such as AI services, further explores the application scenarios of 6G networks, and enriches the value of wireless networks. Furthermore, the task concept and TONA proposed in this article support not only AI tasks, but also sensing-, computing- and data processing-specific tasks.

Acknowledgment

Yang's work was partially supported by the National Key R&D Program of China (2020YFB2104300). Special thanks should be given to 6GANA, which has fully discussed most of the technical viewpoints and solutions mentioned in this article.

[1] Wen Tong, and Peiying Zhu. "6G: The next horizon from connected people and things to connected intelligence," Cambridge University Press, 2021. https://www.vitalsource.com/en-ca/products/6g-the-next-horizon-v9781108997287

[2] 6GANA Whitearticle (2021-05-31). https://www.6g-ana.com/upload/file/20211011/6376956658802608005614761.pdf

[3] Guangyi Liu, Na Li, Juan Deng, et al., "The SOLIDS 6G Mobile Network Architecture: Driving Forces, Features, and Functional Topology," Engineering, 2021, 8(1):42-59.

[4] Q. Tang, F. R. Yu, R. Xie, A. Boukerche, T. Huang, and Y. Liu, "Internet of Intelligence: A Survey on the Enabling Technologies, Applications, and Challenges," in IEEE Communications Surveys & Tutorials, vol. 24, no. 3, pp. 1394-1434, third-quarter 2022.

[5] Yang Yang, et al., "Intelligent IoT for the Digital World," New Jersey: John Wiley and Sons, 2021.

[6] Yang Yang. "Multi-tier Computing Networks for Intelligent IoT," Nature Electronics, 2019, 2: 4-5.

[7] Yang Yang, et al., "6G Network AI Architecture for Everyone-Centric Customized Services," in submission, available at https://arxiv.org/abs/2205.09944

[8] SPECIFICATION, "Quality of Service (QoS) concept and architecture: TS23.107 v16.0.0" 3GPP, 2010.

[9] U. Mikko, et al., "6G Vision, Value, Use Cases and Technologies from European 6G Flagship Project Hexa-X," IEEE Access, 2021, 9: 160004-160020.

[10] Zhi Zhou, et al., "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," Proceedings of the IEEE 107.8 (2019): 1738-1762.

[11] Yong Xiao, et al. "Toward self-learning edge intelligence in 6G," IEEE Communications Magazine 58.12 (2020): 34-40.

[12] C. Thapa, et al., "Splitfed: When federated learning meets split learning," Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 36. No. 8. 2022.

[13] P. Vepakomma, et al., "Split learning for health: Distributed deep learning without sharing raw patient data," arXiv preprint

Yang Yang and Jianjun Wu are the corresponding authors.

arXiv:1812.00564 (2018).

[14] K. B. Letaief, Yuanming Shi, et al., "Edge artificial intelligence for 6G: Vision, enabling technologies, and applications," IEEE Journal on Selected Areas in Communications 40.1 (2021): 5-36.

[15] Jiawei Shao, and Jun Zhang. "Communication-computation trade-off in resource-constrained edge inference," IEEE Communications Magazine 58.12 (2020): 20-26.

Yang Yang and Jianjun Wu are the corresponding authors.