XXXX

面向实时通信媒体面的智能体与AI插件关键技术研究

安玮[1]，魏彬[1]，宋月[1]，王辰[1]，章璐[2]
（1. 中国移动通信有限公司研究院，北京 100053；
2. 中兴通讯股份有限公司，广东 深圳 518063）

摘　要：随着生成式人工智能快速发展，智能体作为新一代人机交互与自动化决策的核心，正逐步融入实时通信核心流程。然而智能体灵活多变的控制流与实时通信固定的媒体处理范式之间存在显著矛盾，制约了AI技术在实时通信场景中的深度应用。本文提出一种面向实时通信网络的融合媒体面与智能体协同调度架构，构建统一媒体框架（General Media Framework, GMF），支持智能体以插件形式高效嵌入媒体处理流水线。针对媒体流高并发处理场景，提出内存零拷贝机制以压缩流处理时延；提出基于通用智能体的二级调度优化模型以消除多智能体并行带来的冗余计算。实验表明各类AI插件可稳定嵌入GMF流水线，所提方案满足实时通信时延要求，为实时通信网络的原生AI演进提供了理论基础和工程参考。

关键词：实时通信；智能体；插件化；流水线
中图分类号：TN915.81
文献标志码：A
doi: 10.11959/j.issn.1000−0801.

Research on Key Technologies of Intelligent Agents and AI Plugins for Real-Time Communication Media Plane

An Wei[1], Wei Bin[1], Song Yue[1], Wang Chen[1], Zhang Lu[2]
1. China Mobile Communications Corporation Research Institute, Beijing 100053, China
2. Zhongxing Telecommunication Equipment Corporation, Shenzhen 518063, China

Abstract: With the rapid advancement of generative artificial intelligence, AI agents, as the core of next-generation human-computer interaction and automated decision-making, are gradually being integrated into the core processes of real-time communication. However, a fundamental conflict exists between the flexible control flow of agents and the fixed data processing patterns of the media plane, which severely restricts the in-depth application of AI technologies in real-time communication scenarios. To address this challenge, this paper proposes a plug-in and agent collaborative scheduling architecture for the converged media plane oriented towards real-time communication networks. A General Media Framework is constructed to support the efficient embedding of agents into the media pipelines in the form of plug-ins. For high-concurrency media stream processing scenarios, a zero-copy memory mechanism is proposed to significantly reduce latency during stream processing. Furthermore, a two-level scheduling optimization model based

收稿日期：2026−02−28；修回日期：2026−05−07
通信作者：魏彬，weibin@chinamobile.com

on a universal agent is presented to address the redundant computation problem caused by multiple agents running in parallel. Experimental results demonstrate that various AI plug-ins can be stably integrated into the GMF pipeline, and the proposed approach satisfies the end-to-end latency requirements of real-time communication, providing both a theoretical foundation and engineering reference for the AI-native evolution of real-time communication networks.

Key words: Real-Time Communication, AI Agents, Plug-in Architecture, Pipelines

0 引言

近年来，智能体技术快速发展，凭借自主任务规划、多工具协同与复杂决策等能力，成为新一代人工智能应用的核心形态[1-2]。在实时通信领域，智能体技术同样展现出广阔的应用潜力，本文围绕实时通信业务智能化升级、媒体面处理架构演进及AI与媒体流处理深度融合三个方向开展研究。

在智能体与实时通信业务的融合应用方面，基于IMS（IP Multimedia Subsystem）的实时通信网络正逐步引入智能体，支持实时翻译、语音转写、沉浸式交互等新型场景，使通话媒体突破传统“传输”的单一定位，具备可理解、可交互、可决策的智能化特征[3]。然而相关成果大多停留在应用层面，尚未探索媒体面底层架构与智能体的深度协同机制，难以充分释放AI效能。

在媒体面处理架构演进方面，传统媒体处理流水线以高效传输、低延迟处理、高稳定性运行为核心目标，处理路径固定、时序约束严格。业界逐步提出媒体处理插件化思路，通过可插拔组件化架构实现媒体功能的灵活叠加与快速部署，一定程度上提升了媒体面的可扩展性。但现有插件化架构多面向静态功能组件设计，缺乏与动态智能体协同的机制，难以支撑智能体自主决策、动态任务跳转等复杂行为，制约了智能体与媒体面架构的深度融合。

在AI与媒体流处理的融合创新方面，自动语音识别（Automatic Speech Recognition，ASR）、文本转语音（Text To Speech，TTS）、自然语言处理（Natural Language Processing，NLP）等AI能力被逐步嵌入媒体处理全链路，实现对媒体流的内容解析与智能处理。国内运营商与设备商纷纷提出通算智融、智简网络等新型架构理念[4]，推动网络、算力、智能能力的一体化部署[5]。但现有研究多以AI能力作为外挂式补充为主，缺少网络原生AI的统一架构设计。

总体而言，智能体与AI插件在实时通信领域的研究仍处于“功能叠加”的初级阶段，在动态智能范式与确定性媒体处理范式的深度协同、插件化架构的标准开放、多智能体资源调度优化等关键问题上，尚未形成成熟解决方案。本文针对上述痛点展开研究，为推动智能体与实时通信领域的深度融合提供技术参考。

1 融合媒体面与智能体架构

1.1 媒体能力分层

为适配AI应用的演进需求，IMS实时通信网络需构建具备高度智能能力与强灵活性的融合媒体面。该融合媒体面支持各类AI能力的标准化引入与本地部署，满足多样化媒体智能处理需求。图1所示融合媒体面由核心层、插件引擎及外部插件层三部分组成：

（1）核心层：承担媒体流的底层处理（编解码、服务质量保障等）及基础AI能力支撑（ASR、话音降噪等），通过服务化接口实现核心能力的开放[6]，为上层模块提供统一、稳定、高效的基础支撑，保障媒体流处理的低延迟与高可靠性。

（2）插件引擎：作为融合媒体面的核心调度

中枢，负责外部插件的注册、鉴权、生命周期管理及资源动态调度；通过流水线动态编排机制灵活组合处理单元；通过智能体与流水线的深度融合，为不同业务场景提供差异化AI服务。

（3）外部插件层：涵盖第三方AI服务（如多模态大模型服务[7]），通过自定义协议、模型上下文协议（model context protocol, MCP）、智能体间通信协议（agent to agent protocol, A2A）等与融合媒体面进行交互，接收脱敏后媒体数据（文本、特征向量等），完成智能处理后返回结果。

媒体面以插件形态提供标准化媒体处理能力，通过插件编排组合形成媒体处理流水线，高效完成业务功能。其插件化架构如图2所示：

本地轻量网管：实现插件的导入、激活、卸载、升级等全生命周期管理。

插件引擎框架：作为插件管理与流水线调度核心，接收本地轻量网管指令完成插件操作，驱动媒体处理流水线按插件连接关系有序处理媒体流。

流水线：由功能插件组合而成，实现具体业务（如背景替换流水线由解码、背景替换、编码等插件串联构成）；流水线管理模块负责编排、流水线状态管理及与业务的关联。

插件：媒体处理基本单元，分为原生插件和第三方服务插件，实现编解码、AI服务等功能，是功能扩展的核心载体。原生插件以动态链接库形式提供，适用于基础媒体处理；第三方服务插件以独立服务形式存在，集成GPU驱动、AI模型等，适用于高复杂度任务。

第三方插件管理：统一纳管第三方服务插件，实现状态监测、配置更新、告警及日志分析等功能，保障其稳定运行。

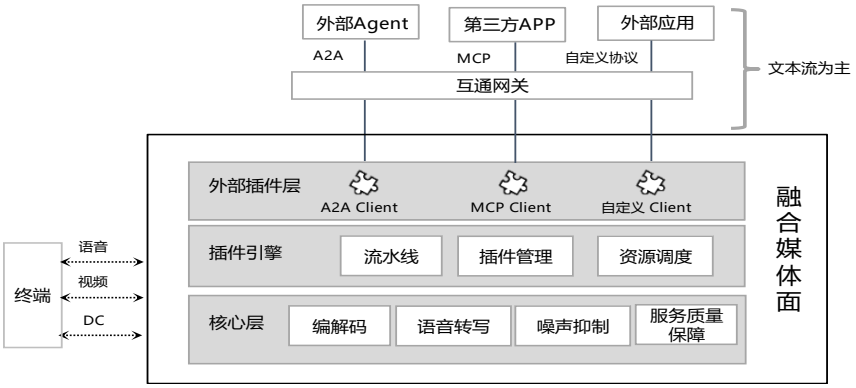

图1　融合媒体面-媒体能力分层图

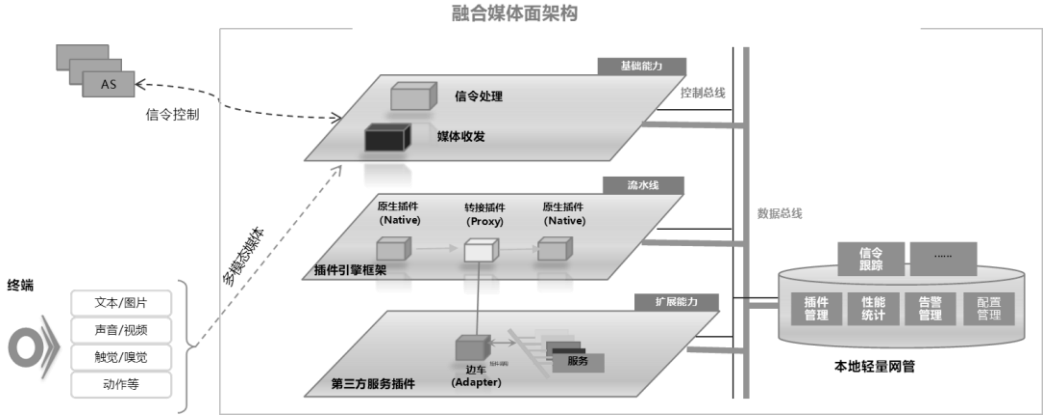

图2　融合媒体面-插件化架构图

转接插件与边车：协同实现第三方服务与插件引擎的适配对接。其中，转接插件作为代理参与流水线编排；边车适配差异化第三方接口。针对ASR、TTS等流式服务，通过WebSocket协议建立交互通道；非流式服务则通过HTTP交互，确保第三方能力无缝集成。

1.2 流水线与智能体协同机制

1.2.1 统一媒体框架基础元素结构

插件化媒体面引入统一媒体框架GMF，为公共加工环节定义了标准化的数据描述和访问接口，实现智能体与流水线的深度协同，关键元素包括：

（1）图（Graph）：本质为有向无环图（Directed Acyclic Graph，DAG），是处理流程的宏观表示，通过网管配置连接处理节点（Node）构成流水线。单个图可包含多条流水线实现复杂业务处理。

（2）节点（Node）：Graph中的原子执行实体，封装具体算法或操作（如RTP处理、AI模型等），分为内置Native节点、与第三方连接的proxy节点。按位置与数据处理方式可分为输入型、输出型、处理型、混合型（如图3所示）。

（3）流（Stream）：连接各Node的数据通道，定义了数据流向，是构建Graph拓扑结构的基础。

（4）数据包（Packet）：数据流动的基本载体，各类数据（音视频帧、文本等）均封装成Packet传递，使媒体面可处理多种形式的数据流。

（5）任务（Task）：即流水线（Pipeline），是插件引擎调度的基本单元，由多个Node组成，实现完整业务。节点间通过共享缓存（PacketBuf）传递媒体数据，通过Event存放控制信息，确保数据与控制流高效协同（如图4所示）。

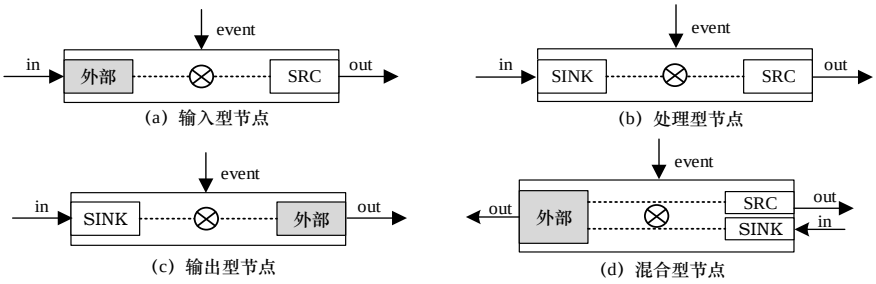

图3 GMF中的节点类型分类

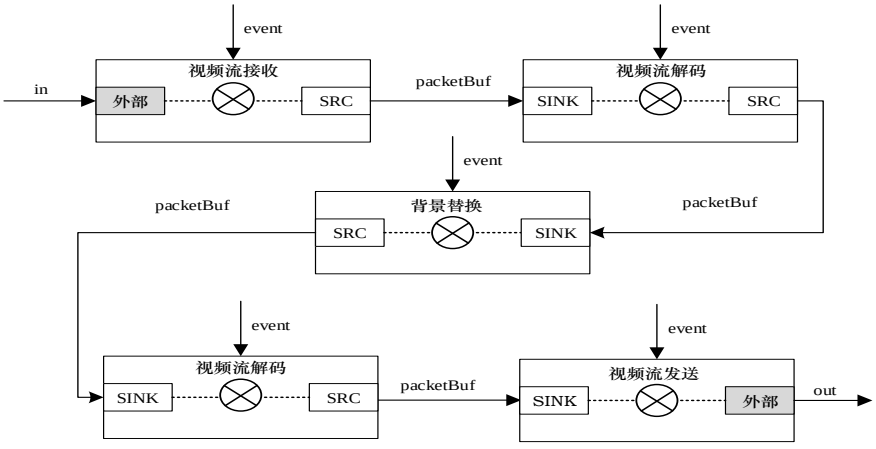

图4 GMF中的任务功能举例

1.2.2　智能体数据订阅与控制注入

实现智能体与流水线无缝嵌入的核心，是对媒体处理流水线进行语义化抽象，提取出智能体可交互的标准化数据节点，建立标准化交互接口。典型音视频媒体流水线可解构为以下公共加工环节：

（1）音频处理链：音频解码→ PCM（脉冲编码调制音频）→ ASR→文本→TTS→ PCM→混音器→音频编码→RTP 发送

（2）视频处理链：视频解码→H.264/H.265 裸帧→视频抽帧→图像→图像识别

智能体以插件形式注册至 GMF，根据功能需求订阅目标数据节点（如“实时字幕”智能体订阅 ASR，“内容安全”智能体订阅视频抽帧）。流水线处理至对应节点时，GMF 自动将数据精准分发给订阅智能体，智能体处理数据后，通过 GMF 向流水线注入控制指令实现动态调控（如向 TTS 节点注入文本驱动语音合成），无需关注底层编解码细节，聚焦高层语义处理，实现控制流与数据流解耦[8]。图 5 展示了智能体与流水线的协同架构。

1.3　智能体部署方案

智能体在实时通信网络中可提供多样化 AI 服务（如智能教学、通话实时辅助等），拓展了实时通信的服务边界。将智能体部署于融合媒体面内部，相比外部更能适配电信级要求：内部部署可直接访问原始数据，规避跨进程传输开销及序列化开销，保障实时响应（端到端时延 200ms 以内）；媒体面通过数据闭环流转，分权分域，数据脱敏等功能，提供框架级安全防护，贴合行业监管要求；媒体面与 NPU、DSP 等底层硬件深度集成，智能体可直接调用硬件加速资源，提升算力利用率[9]。

当前智能体调用外部工具扩展能力边界已成为主流[10]，但如何与电信级媒体流水线深度融合仍是难题。为平衡开放生态与系统可控性[11]，本文设计两种智能体部署方案：

（1）**原生插件方式部署：**如图 6 所示，智能体以原生插件形式部署于媒体面内部，设备提供统一的工作流引擎，第三方开发者仅需提供 JSON/YAML 格式工作流描述文件即可实现业务。该方案架构简洁、开发门槛低，但功能与性能受限于工作流引擎，灵活性不足。

（2）第三方插件方式部署：如图 7 所示，智能体以第三方插件形式，通过边车接入插件引擎，第三方提供包含 NLP 微服务、推理引擎等组件的完整应用包，部署于媒体面容器平台。应用包内部组件通过容器内网直接通信，减少转发延

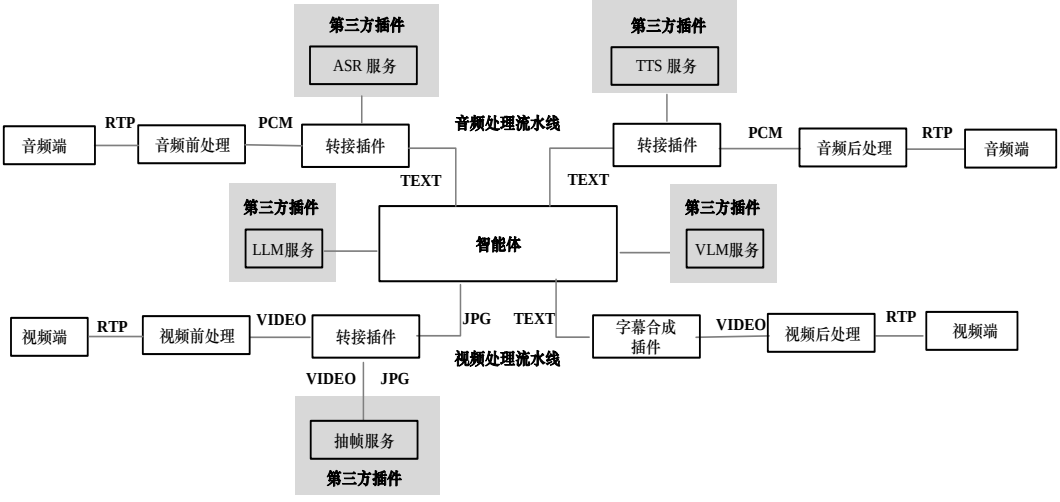

图 5　智能体与媒体面流水线的协同架构

图6　原生插件方式部署

迟与总线负载，开发灵活性高，但对容器平台安全与资源调度能力要求较高。

2　智能体流式处理性能分析与优化

在媒体处理流水线中，当多个垂直领域智能体（如字幕生成智能体、内容审核智能体、内容摘要智能体等）并行处理同一路媒体流时，系统面临着对底层AI组件的重复调用问题。以ASR为例，若每个垂直智能体均独立触发一次ASR计算，将导致对同一音频执行N次完全相同的计算，不仅消耗大量计算资源，还会引发延迟累加，为解决这一问题，本文提出二级智能体调度优化模型（如图8所示），引入通用智能体（Universal Agent）作为智能协调中心，通过“需求聚合”与“数据复用”机制实现对底层计算组件调用的统一管理。

图8展示了优化后的数据流形成清晰的树状结构：源数据先汇聚至通用智能体，经其统一处理后分发给各垂直智能体，数据传输路径简洁高效，有效规避冗余计算与延迟累积问题。

2.1　智能体资源调度

二级调度模型由通用智能体和若干垂直领域智能体构成，实现机制分为三个阶段：

（1）**需求收集与聚合分解阶段：**会话初始建立阶段，各垂直领域智能体向通用智能体注册其数据处理需求，通用智能体构建统一“需求清单”并进行聚合分析，分解出最细粒度的公共需求单元。例如，字幕生成、内容审核、内容摘要三个智能体的公共需求为“ASR文本”。

（2）**数据流复用与统一分发阶段：**通用智能体切换为数据代理模式投入工作，直接向GMF订阅源头的未经处理公共媒体数据（如原始音频流），实现统一计算一次调用和一对多复制分发。例如面向字幕生成、内容审核、内容摘要三个智

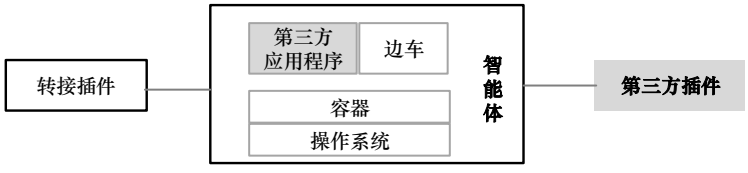

图7　第三方插件方式部署

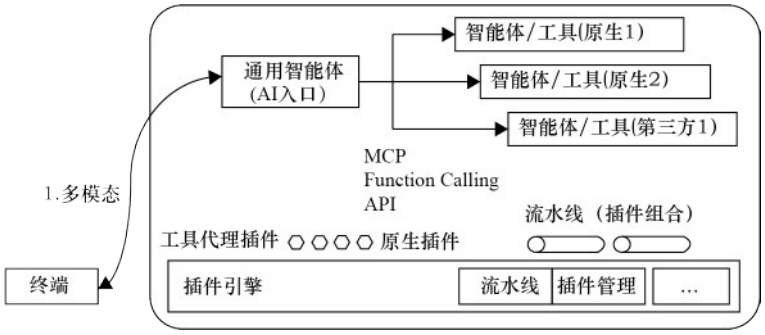

图8　智能体的调度优化方案

能体的需求，通用智能体仅调用一次ASR服务，从根源上消除冗余计算，在获取数据后将其复制多份，通过内置数据通道，分发给所有注册该需求的垂直智能体。

（3）垂直领域智能体的角色演进：各个垂直领域智能体无需直接与GMF底层节点交互，仅需订阅通用智能体，专注于自身特定逻辑，如字幕生成智能体专注于时序对齐与字幕渲染，内容审核智能体专注于敏感词检测与图像违规识别，内容摘要智能体专注于关键信息提取与摘要文本生成。

该资源调度机制为融合媒体面智能体带来显著性能提升，在计算复杂度方面，假设系统中有N个垂直智能体需要ASR服务，传统模式下ASR的调用次数复杂度为$O(N)$。本文二级调度模型中，无论N如何增长，ASR服务始终只被调用一次，复杂度降至$O(1)$。这一优化极大地减轻了ASR服务端的负载，使系统能够以恒定的资源开销支持多类垂直业务的开展。

2.2　插件化流水线端到端时延模型

在实时音视频通信场景中，端到端时延是核心指标，通常要求200ms以内。本文基于统一媒体框架（GMF）开展设计，插件化流水线的端到端时延T_{E2E}可设定为排队等待时延T_{wait}、算法执行时延T_{exec}、节点间传输时延T_{trans}的叠加，如式（1）所示：

$$T_{\mathrm{E2E}}=\sum_{i=1}^{K}\left(T_{\mathrm{wait}}^{(i)}+T_{\mathrm{exec}}^{(i)}(L)+T_{\mathrm{trans}}^{(i,i+1)}\right) \quad (1)$$

（1）排队等待时延：受用户并发量与调度机制影响。本文引入通用智能体，通过统一计算与一对多复制分发机制，将并发带来的排队与计算复杂度降到最低，尽可能减小整体等待时延T_{wait}。

（2）算法执行时延：受输入长度与硬件利用率影响。原生插件部署可最大程度利用GPU和CPU资源，硬件利用率可达85%~95%；第三方插件受容器化隔离影响，硬件利用率仅为60%~75%，直接影响T_{exec}的下限。

（3）节点间传输时延：GMF框架下节点之间通过共享缓存进行数据流转，有效规避了跨进程或跨节点的网络传输及序列化开销。原生插件部署通信时延仅为1ms至5ms；第三方插件部署内部组件通过gRPC等网络协议通信，导致时延额外增加约5ms。

基于统一媒体框架的插件流水线处理流程可抽象为DAG，表示为$G=(V,E)$。其中集合$V=\{v_1,v_2,\ldots,v_K\}$表示流水线中从1至K的处理节点（基础插件、智能体插件），集合E表示节点间的数据依赖关系。

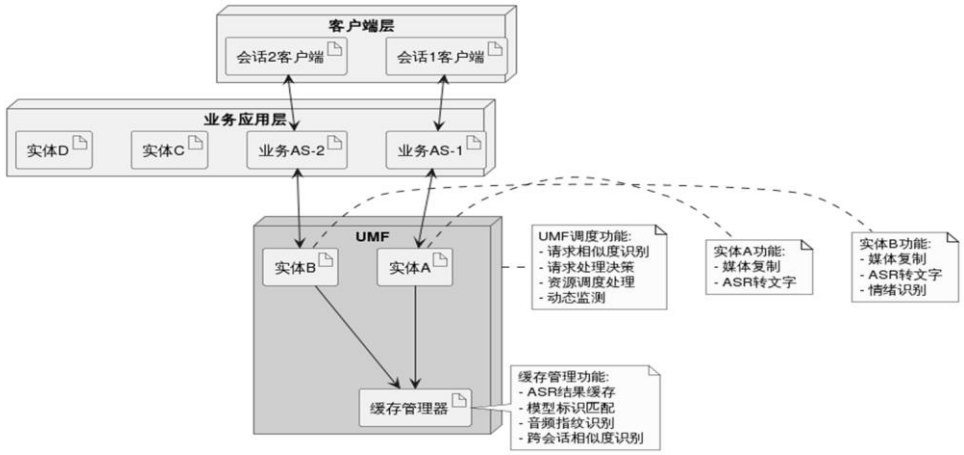

图9　智能体的调度优化内部实现逻辑

实时通信场景中，音频或视频帧数据包从输入节点v_1流经特定业务路径P最终到达输出节点v_K，整个流处理过程中的端到端总时延T_{E2E}形式化表示为式（2）：

$$T_{\mathrm{E2E}}=\sum_{v_i\in P}\left(T_{\mathrm{wait}}^{(i)}+T_{\mathrm{exec}}^{(i)}(L)\right)+\sum_{(v_i,v_j)\in P}T_{\mathrm{trans}}^{(i,j)} \quad (2)$$

当采用原生插件方式部署时，$T_{\mathrm{trans}}^{(i,j)}$为进程内函数调用的资源耗时，是较小的常数。当采用第三方插件方式部署时，受容器间的隔离和gRPC协议的影响，总的传输时延变为式（3）：

$$T_{\mathrm{trans}}^{(i,j)}=T_{\mathrm{s}}+T_{\mathrm{n}}+T_{\mathrm{d}} \quad (3)$$

其中T_{s}表示发送端时延，T_{n}表示介质传输时延，T_{d}表示接收端时延。

2.3 基于全局共享缓存池的内存零拷贝模型

在实时通信的高并发场景下，高频内存读写会严重占用总线带宽，导致节点间传输时延T_{trans}增大。为解决多智能体插件串联带来的时延问题，本文提出的GMF引入共享缓存与零拷贝机制，进一步优化该场景下的时延。

传统内存深拷贝模型下，设一帧媒体载荷大小为S（以字节计），单字节复制的时钟周期开销（或等效时间延迟）为C_{byte}，对于包含K个处理节点的流水线$P=\{v_1,v_2,\ldots,v_K\}$，数据需要经历$K-1$次跨节点流转，系统总内存拷贝时延E_{c}表现为数据大小与链路长度的线性叠加，具体见式（4）：

$$E_{\mathrm{c}}=\sum_{i=1}^{K-1}\left(S\cdot C_{\mathrm{byte}}\right)=(K-1)\cdot S\cdot C_{\mathrm{byte}} \quad (4)$$

基于式（4）可以得出，当引入多个AI插件时K值增大，处理高清媒体流时S值激增，均会导致E_{c}迅速增大，不仅带来极大的传输延迟，还会造成CPU缓存命中率（Cache Hit Rate）断崖式下降，进一步影响系统性能。

GMF设计了全局共享缓存池，数据载荷在输入节点v_1处仅被分配和写入一次，后续流转过程中，数据实体在内存中的物理位置保持静止，节点之间仅通过事件总线传递包含内存指针以及引用计数的控制信息。设传递一次内存指针及元数据结构体的时间开销为C_{ptr}。此时，GMF框架下的节点间传输总开销被极大地压缩，表示为式（5）：

$$E_{\mathrm{c\_opt}}=\sum_{i=1}^{K-1}C_{\mathrm{ptr}}=(K-1)\cdot C_{\mathrm{ptr}} \quad (5)$$

通过对比传统内存深拷贝与全局共享缓存池零拷贝两种模型，可推导出该零拷贝模型的理论加速比η为式（6）：

$$\eta=\frac{E_{\mathrm{c}}}{E_{\mathrm{c\_opt}}}=\frac{S\cdot C_{\mathrm{byte}}}{C_{\mathrm{ptr}}} \quad (6)$$

由于C_{ptr}仅为传递一个8字节内存地址的开销，而$S\cdot C_{\mathrm{byte}}$代表媒体流数据量，两者量级差异显著（$C_{\mathrm{ptr}}\ll S\cdot C_{\mathrm{byte}}$），传输时延从$O(K\cdot S)$降低至$O(K)$。理论加速比$\eta$与媒体流的帧大小$S$成正比。媒体数据越庞大，智能体的逻辑链路越长，统一媒体框架下的零拷贝机制为系统降低的时延就越明显。

2.4 多并发二级智能体调度优化模型

在融合媒体面实际应用中，多个垂直领域智能体并发处理同一路媒体流会引发底层基础AI组件被重复调用，本文建立的二级智能体调度优化模型可有效解决该问题。

传统无聚合调度模式下，设系统存在N个垂直领域智能体，集合记为$A=\{a_1,a_2,\ldots,a_N\}$。每个智能体的计算量拆分为基础计算C_{b}（如ASR调用）与特异性业务C_n（如字幕生成智能体的时序对齐渲染），系统处理单帧媒体数据的总算力负载L_{trad}可以表示为式（7）：

$$L_{\mathrm{trad}}=\sum_{n=1}^{N}\left(C_{\mathrm{b}}+C_n\right)=N\cdot C_{\mathrm{b}}+\sum_{n=1}^{N}C_n \quad (7)$$

其中C_{b}是计算密集型模型推理操作，$C_{\mathrm{b}}\gg C_n$。传统模型的底层重计算资源消耗随智能体数量N线性增长，严重影响实时性。

通过在 GMF 框架内引入“通用智能体”作为统一调度协调中心：聚合阶段，所有垂直领域智能体将自身需求 r_n 注册至通用智能体，形成系统总需求集合 $R=\{r_1,r_2,\ldots,r_N\}$。通用智能体对该集合进行依赖树分析，提取出最细粒度的公共基底需求集 R_c，表示为式（8）：

$$R_c=\bigcap_{n=1}^{N} r_n \tag{8}$$

分发阶段，通用智能体仅对 R_c 发起一次算力调用（即仅执行一次 C_b），利用全局共享缓存池机制，将结果的内存指针分发给下游垂直智能体。设单次指针分发开销为 C_d（远小于 C_b），优化后系统总算力负载 L_{opt} 可表示为式（9）：

$$L_{opt}=C_b+\sum_{n=1}^{N}\left(C_d+C_n\right)=C_b+N\cdot C_d+\sum_{n=1}^{N}C_n \tag{9}$$

渐进资源消耗比可表示为式（10）：

$$\lim_{N\to\infty}\frac{L_{opt}}{L_{trad}}=\lim_{N\to\infty}\frac{C_b+N\cdot C_d+\sum C_n}{N\cdot C_b+\sum C_n} \tag{10}$$

由于特异业务消耗 C_n 是必须执行的有效负荷，可将其从分子分母中剥离，仅考虑整体系统框架的公共调度成本，如式（11）所示：

$$O_{trad}=N\cdot C_b, O_{opt}=C_b+N\cdot C_d \tag{11}$$

此时原极限可表示为式（12）：

$$\lim_{N\to\infty}\frac{O_{opt}}{O_{trad}}=\lim_{N\to\infty}\frac{C_b+N\cdot C_d}{N\cdot C_b}\approx\frac{C_d}{C_b} \tag{12}$$

基于 GMF 的全局共享缓存池设计，跨节点分发指针的开销可忽略不计，上述极限比率接近于 0。计算复杂度从 $O(N)$ 降低至 $O(1)$，无论垂直领域智能体数量如何增加，系统基础媒体特征提取的资源消耗都将保持在恒定范围。

3　系统实现与验证

为验证本文提出的融合媒体面智能体系统架构的通用性、调度性能及工程实用性，本章从 AI 插件接入验证、部署方案性能对比、端到端时延分析三个维度设计系列实验。

3.1　AI 插件接入验证

选取实时通信场景中三类典型开源 AI 模型：声音降噪、TTS、ASR，作为测试插件，分别接入 GMF 流水线开展实验。

3.1.1　声音降噪插件接入流水线验证

选取 FRCRN_SE_16K[12] 与 MossFormer-GAN_SE_16K[13-14] 两款推理模型，分别代表频域递归网络与生成对抗网络两种降噪技术路线。GMF 采用标准化节点抽象设计，仅需将其封装为统一的媒体处理节点即可实现插件与框架的无缝对接。

实验硬件选用英伟达 A40 图形计算卡，选取不同时长的实时语音片段作为测试输入，将两款降噪插件分别接入 GMF 流水线，验证模型推理性能。每个测试用例重复执行 100 次，统计平均推理时长。实验结果（图 10）表明，FRCRN_SE_16K 模型推理时延更低，完全适配实时通信低时延要求；MossFormerGAN_SE_16K 模型降噪性能更优，二者均可通过 GMF 框架稳定集成，验证了框架对降噪插件的适配能力。

3.1.2　TTS 插件接入流水线验证

选取 MeloTTS 模型[15] 作为基准测试负载，该模型在语音自然度、推理时延与算力开销间实现了较优平衡。在 GMF 架构中，该模型被封装为标准化“媒体流生成节点”，通过零拷贝技术将生成的音频波形直接注入下游混音与编码节点。

实验选取不同长度文本（1 字，5 字，10 字，15 字，20 字）作为测试输入，验证模型推理时延与输入长度的关联性，并测试不同并发数下的 GPU 利用率和时延变化。实验结果（图 11、图 12）表明，合理控制输入文本长度，可将 MeloTTS 插件的推理时延稳定控制在 500ms 以内，满足实时通信场景语音合成时延要求，验证了 GMF 框架对生成类 AI 插件的调度管控能力。

3.1.3　ASR 插件接入流水线验证

选取开源模型 SenseVoice-Small[16] 作为测试

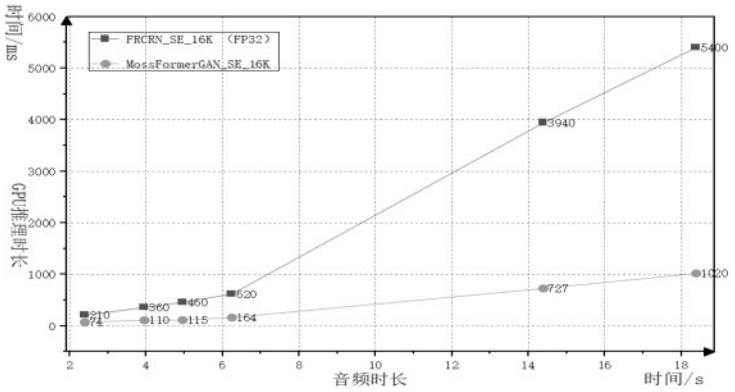

图10　两种降噪模型接入流水线推理时长对比

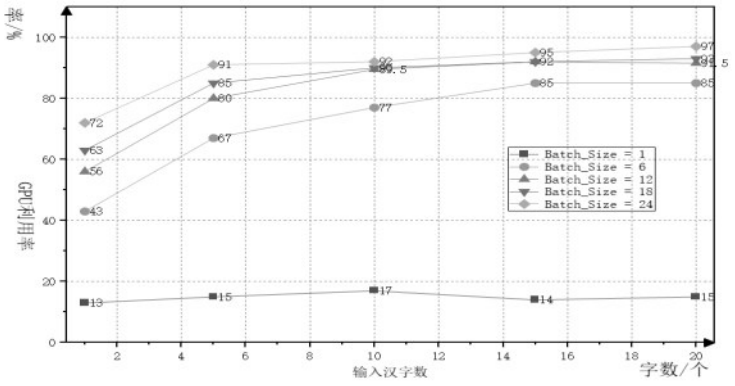

图11　MeloTTS模型接入流水线资源利用测试结果

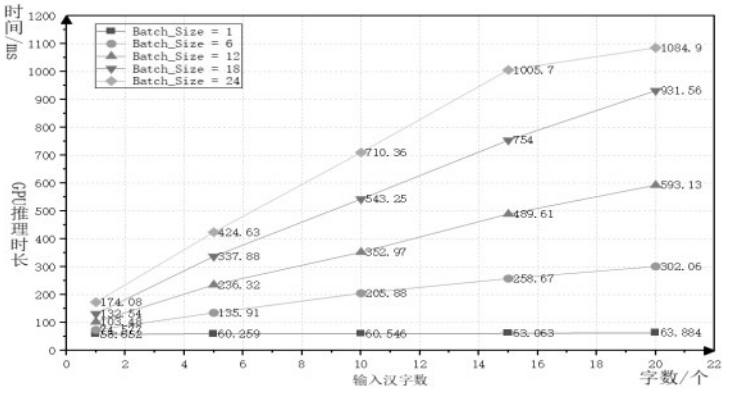

图12　MeloTTS模型接入流水线推理时长测试结果

负载嵌入流水线。该模型采用面向多语言、多任务的非自回归纯编码器架构，支持语音识别、情感分析等并发任务，高度契合实时通信场景需求。

实验硬件环境与前文一致，采用英伟达A40 GPU，选取不同时长实时语音片段作为输入，每个测试用例重复测试100次，统计模型平均推理时长。实验结果（图13）表明，合理控制输入语

音切片长度，可将 SenseVoice-Small 模型推理时延稳定控制在 200ms 以内，完全满足实时通信场景语音识别时延要求，验证了系统对计算密集型 AI 插件的高效调度能力与兼容性。

3.2　部署方案性能对比与实验验证

为明确智能体在实时通信融合媒体面中的最优部署模式，对“原生插件方式部署”与“第三方插件方式部署”两种方案开展对比实验，从媒体处理时延、CPU 占用率、内存占用量三个维度量化分析。实验采用 100 路呼叫并发场景，重复测试 15 次，统计关键指标均值，具体实验数据如表 1 所示。

结果表明：原生插件部署方案较第三方插件部署方案平均处理时延降低约 30%，CPU 平均占用率降低约 20%，内存占用降低约 18%。原生部署方案依托进程内集成与零拷贝机制，硬件资源利用率更高、时延更低，更适配核心网高性能场景；第三方插件部署方案依托容器化隔离优势，更适配第三方能力快速集成场景。本文提出的梯度部署体系覆盖从高性能原生集成到灵活容器化部署的全场景需求，为开放可控的智能通信系统构建提供了可行路径。

3.3　端到端时延验证

为验证本文提出的融合媒体面智能体系统架

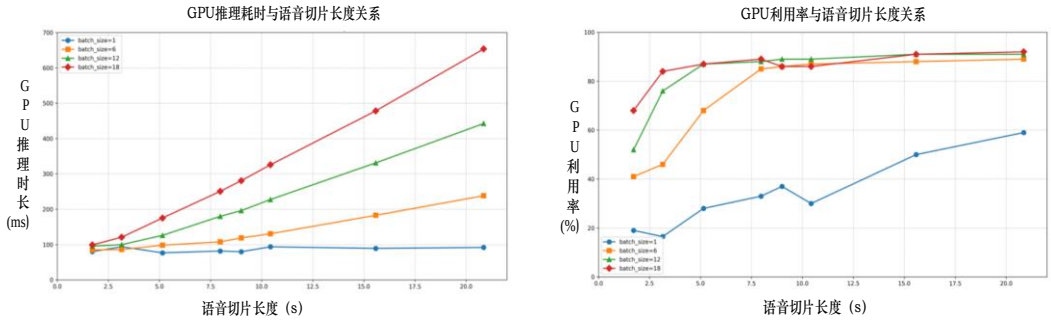

图 13　SenseVoice-Small 模型推理性能对比

表 1　两种部署方案性能测试数据

序号	原生插件平均处理时延(ms)	第三方插件平均处理时延(ms)	原生插件CPU平均占用率（%）	第三方插件CPU平均占用率（%）	原生插件内存占用（MB）	第三方插件内存占用（MB）
1	32	46	12	15	585	710
2	30	44	11	14	580	705
3	29	43	10	13	575	700
4	31	45	12	15	582	708
5	28	42	10	13	572	698
6	33	48	13	16	590	715
7	35	50	14	17	595	722
8	27	41	9	12	570	695
9	34	49	13	16	592	718
10	36	51	15	18	600	728
11	31	45	11	14	581	706
12	29	43	10	13	576	701
13	30	44	11	14	578	703
14	32	47	12	15	586	712
15	28	42	9	12	571	696

构在实时通信业务场景中的可行性与适用性，搭建端到端（E2E）实验环境（如图14所示），IMS核心网负责提供基础注册、路由、会话控制等能力；插件化媒体面通过AI插件对音视频流进行实时加工处理，由智能体完成业务逻辑的自主决策与功能协同；运营管理平台和业务AS负责业务的签约与生命周期管理。

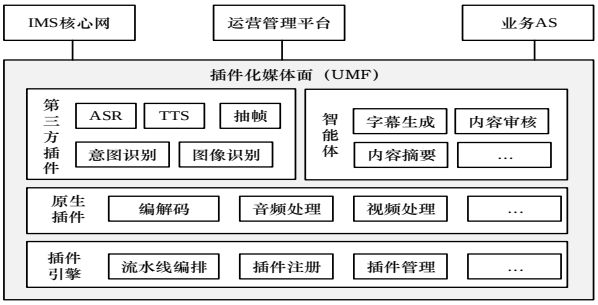

图14　端到端时延实验环境组网

实验采用真实终端进行拨测，模拟用户实时通信场景，在IMS侧部署抓包工具，分别记录“经过插件化媒体面”与“不经过插件化媒体面”两种场景下的时延数据，通过差值分析引入GMF系统后带来的时延增量。实验数据表明：引入GMF系统后，音频端到端处理时延增量小于100ms，视频端到端处理时延增量小于200ms，均满足实时通信网络中音视频传递的时延要求（通常要求音频时延≤150ms、视频时延≤250ms）。实验结果充分证明了本文提出的GMF系统架构在实时通信场景中的可行性与实用性。

4　结束语

本文系统地研究了智能体以插件形式嵌入媒体处理流水线所面临的协同、部署和性能挑战，并提出了相应的解决方案。主要贡献包括：（1）提出了面向实时通信网络的融合媒体面与智能体协同调度架构，通过提取公共加工内容并定义GMF标准化接口，实现了智能体控制流与媒体数据流的有效解耦与高效协同；（2）提出了基于全局共享缓存池的内存零拷贝机制与基于通用智能体的二级调度优化模型，从理论上证明了该方法可将计算复杂度从O(N)降至O(1)、传输时延从O(K·S)降至O(K)；（3）通过两种部署方案的对比，为构建从封闭到开放的AI生态提供了清晰的技术演进路径。随着6G网络向“通感算智”深度融合演进，这种插件化、可编排的媒体智能处理模式有望为沉浸式通信、全息通话等未来业务提供可演进的架构底座。

未来将进一步深入研究跨智能体的安全隔离与性能保障，为构建智能、自适应、安全可信的实时通信系统提供更加完善的技术支撑。

参考文献：

[1] XI Z, CHEN W, GUO X, et al. The Rise and Potential of Large Language Model Based Agents: A Survey[J]. arXiv preprint arXiv:2309.07864, 2023.

[2] 郭先会, 张梦姣, 马军. 基于大语言模型的智能体构建综述[J]. 通信技术, 2024, 57(9): 873-879.
GUO X H, ZHANG M J, MA J. A survey on the construction of agents based on large language models[J]. Communications Technology, 2024, 57(9): 873-879.

[3] 3GPP TR 23.700-91: Study on enablers for network automation for the 5G System (5GS); Phase 2[S]. 2020

[4] 王辰,白雪茜,魏彬等.面向实时通信的边缘智能关键技术研究[J].电信科学,2025,41(11):14-30. DOI: 10.11959/j.issn.1000-0801.2025194.
WANG Chen,BAI Xueqian,WEI Bin,et al.Research on the key technologies of edge intelligent for real-time communication[J]. Telecommunications Science, 2025, 41(11): 14-30. DOI: 10.11959/j.issn.1000-0801.2025194.

[5] 3GPP TS 23.288 V18.11.0 (2025-09). Architecture enhancements for 5G System (5GS) to support network data analytics services (Release 18)[S]. 3GPP, 2025.

[6] ETSI GS ENI 005 V2.1.1. Experiential Networked Intelligence (ENI); System Architecture[S]. Sophia-Antipolis: ETSI, 2021.

[7] FUNAUDIOLLM TEAM (AN K Y, CHEN Q, DENG C, et al.). FunAudioLLM: Voice Understanding and Generation Foundation Models for Natural Interaction Between Humans and LLMs[EB/OL]. arXiv preprint arXiv:2407.04051, 2024.

[8] YAO S, ZHAO J, YU D, et al. ReAct: Synergizing Reasoning and Acting in Language Models[J]. arXiv preprint arXiv:

2210.03629, 2022.

[9] 张万才, 张楠, 杨文清, 王涛, 张文强. 基于虚拟化的GPU异构资源池平台架构设计、关键技术及应用研究[J]. 电信科学, 2024, 40(9): 162-175.
ZHANG W C, ZHANG N, YANG W Q, et al. Architecture design, key technologies and application research of GPU heterogeneous resource pool platform based on virtualization[J]. Telecommunications Science, 2024, 40(9): 162-175.

[10] 周涛，李鑫，周俊临，李奕. 大模型智能体：概念、前沿和产业实践 [J]. 电子科技大学学报 (社科版), 2024, 26 (4): 57-62.
ZHOU Tao, LI Xin, ZHOU Jun-lin, LI Yi. Large-Model-Based Agents: Concepts, Frontiers and Industrial Practices[J]. Journal of University of Electronic Science and Technology of China (Social Sciences Edition), 2024, 26(4): 57-62.

[11] CASTRO L F S, RIGO S. Relating Edge Computing and Microservices by means of Architecture Approaches and Features, Orchestration, Choreography, and Offloading: A Systematic Literature Review[J]. arXiv preprint arXiv:2301.07803, 2023.

[12] ZHAO S K, MA B, WATCHARASUPAT K N, et al. FRCRN: Boosting Feature Representation Using Frequency Recurrence for Monaural Speech Enhancement[C]//ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Singapore: IEEE Press, 2022: 9281-9285.

[13] ZHAO S, MA B. MossFormer: Pushing the Performance Limit of Monaural Speech Separation Using Gated Single-Head Transformer with Convolution-Augmented Joint Self-Attentions[C]//ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Rhodes Island, Greece: IEEE Press, 2023: 1-5. doi: 10.1109/ICASSP49357.2023.10096646.

[14] ZHAO S, MA Y, NI C, et al. MossFormer2: Combining Transformer and RNN-Free Recurrent Network for Enhanced Time-Domain Monaural Speech Separation[J]. arXiv preprint arXiv: 2312.11825, 2023. (Accepted by ICASSP 2024)

[15] ZHAO W L, YU X M, QIN Z Y. MeloTTS: High-quality Multi-lingual Multi-accent Text-to-Speech[CP]. MyShell.ai and MIT, 2023[2024-12-25].

[16] FunAudioLLM Team. SenseVoice: Multi-lingual Voice Understanding Model with Emotion and Intent Recognition[EB/OL]. arXiv preprint arXiv:2407.04051, 2024.

[作者简介]

魏彬，出生于1983年，性别男，硕士毕业于北京邮电大学，现任中国移动研究院网络与IT技术研究所副所长，主要从事5/6G核心网标准化、5G专网、实时通信、人工智能等领域研究。

宋月，出生于1984年，性别男，硕士毕业于北京邮电大学，现任中国移动研究院主任研究员、3GPP CT4主席，主要研究方向为5/6G核心网、实时通信网络。

王辰，出生于1993年，性别男，硕士毕业于北京大学，现任中国移动研究院研究员，主要研究方向为5G-A/6G、实时通信网络AI系统架构及前沿技术研究。

章璐，出生于1978年，性别男，硕士毕业于南京航空航天大学，高级工程师。现任中兴通讯股份有限公司电信云与核心网产品线产品规划总工，专业方向为核心网、实时通信、人工智能。

安玮，出生于1996年，性别男，硕士毕业于北京科技大学，现任中国移动研究院研究员，主要从事5/6G核心网、实时通信、人工智能等领域研究。