评估 Wi-Fi 的 VR 流媒体传输性能：关于 RealisticHEVCVideo 流量的研究

Ferran Maura、Francesc Wilhelmi、Boris Bellalta 庞培法布拉大学，巴塞罗那，西班牙 庞培法布拉大学无线网络研究组罗克·博罗纳特街138号，08018 巴塞罗那，西班牙 {ferranjosep.maura, francisco.wilhelmi, boris.bellalta}@upf.edu

arXiv:2601.16950v1 [cs.NI] 23 Jan 2026

摘要—**云端虚拟现实（VR）流媒体传输由于对高吞吐量和低时延的要求，给 802.11网络带来了重大挑战。当多个虚拟现实用户共享一个 Wi-Fi 网络时，由此产生的上行链路和下行链路流量可能会迅速使信道饱和。本文研究了 802.11网络在不同帧率、比特率、编解码器设置和用户数量下支持真实 VR 流媒体传输工作负载的能力。我们开发了一个仿真框架，用于复现 Air Light VR（ALVR）的运行方式，将真实的高效视频编码视频流量输入 802.11仿真模型。我们的研究结果揭示了 Wi-Fi 的性能异常，并表明帧内刷新（IR）编码能够有效降低时延变化、改善服务质量，在信道饱和前支持最多 4 个并发虚拟现实用户以恒定比特率（CBR）100 兆比特每秒运行。**

索引词—**虚拟现实、Wi-Fi、流媒体、IEEE 802.11、服务质量、体验质量、视频压缩、ALVR**

I. 引言

扩展现实（XR）涵盖虚拟现实（VR）和增强现实（AR），预计将在未来几年成为一项变革性技术。尽管XR已在游戏领域广泛应用，但它有望颠覆众多新兴应用场景，包括沉浸式远程协作、虚拟培训与教育、远程医疗操作以及工业机械的远程操作 [1]。

云端VR流媒体传输是一种带宽密集型扩展现实应用，其中交互式VR内容由高性能远程服务器（例如云端服务器）渲染，再传输到性能较弱的终端设备。这些服务器必须提供充足的CPU和GPU资源，以便以较高帧率渲染场景，并实时运行VR流媒体应用。具体而言，服务器会按照各自的帧率和采样率对视频流和音频流进行编码，并传输到用户的头戴式显示器（HMD）。与此同时，HMD会跟踪用户的姿态和动作，并将跟踪数据发送回服务器，以更新玩家状态。服务器根据这些反馈渲染新帧，再将其传回HMD进行显示 [2]。

云渲染的优势在于降低对客户端硬件的要求。只要端到端时延足够低，VR 流媒体传输就能呈现高保真图形，否则这些图形将超出典型 HMD 设备的渲染能力。尽管有线连接能够支持此类应用，但无线技术——尤其是 IEEE 802.11（商业上称为 Wi-Fi），这一应用最广泛的室内连接标准——因支持用户移动性并提升舒适度而日益受到青睐。然而，无线网络要满足实时 VR 流媒体传输对时延和可靠性的严格要求并非易事。Wi-Fi 尤其面临若干局限。首先，802.11 通过分布式协调功能（DCF）实现随机信道接入，这意味着设备必须竞争介质的访问权。其次，服务器、接入点（AP）和 HMD 站点（STA）不会协调其下行链路（DL）与上行链路（UL）传输，这可能进一步降低 VR 体验质量 [3]。最后，邻近网络在同一信道上运行所产生的干扰会造成不可预测的中断，进一步降低 VR 应用的性能 [4]。

本文研究了通过 Wi-Fi 运行 VR流媒体应用的适用性。本研究的主要创新之处在于，在 802.11 仿真中使用真实的编码视频传输，这对于准确再现不同编解码器配置所产生的帧大小至关重要。我们的主要贡献

本研究部分由Wi-XR PID2021-123995NB-I00和TRUE-Wi-Fi PID2024-155470NB-I00（MCIU/AEI/FEDER,UE）（MCIU/AEI/FEDER, UE）资助，部分由MCIN/AEI通过Maria de Maeztu卓越单位计划（CEX2021-001195-M）资助，并由西班牙科学、创新与大学部通过FPI奖学金（PRE2022-101847）资助。

如下：

- **基于 IEEE 802.11 的 VR 流媒体仿真：**我们开发了一个模块化仿真框架，用于仿真服务器与头戴式显示器客户端在共享 Wi-Fi 网络上进行的一个或多个 VR 流媒体会话。该框架包括分别封装的服务器和客户端应用程序、能够以可配置的帧率和比特率处理 4K 视频的视频编解码器流水线，以及用于表示 IEEE 802.11 物理层/媒体访问控制层行为的排队与传输模型。
- **网络与编解码器层动态的联合分析：**我们使用集成仿真框架评估关键服务质量（QoS）指标（例如时延、丢包）如何受到编解码器参数（例如帧率、比特率）、网络竞争（例如恒定比特率用户数量）和物理层条件（例如站点到接入点距离）的影响。我们的结果揭示了密集 VR 场景中视频编码策略与无线性能之间相互作用的规律。

II. 基于云的 VR 流媒体

本节介绍云端VR流媒体传输的架构，概述当前采用的视频编码策略，并讨论确保良好用户体验所需的性能要求。

A. 服务器-客户端架构

在 VR 流媒体传输中，渲染通常在服务器上进行，服务器会对视频、音频和触觉数据进行编码，并将其传输至头戴式显示器。头戴式显示器会持续跟踪用户的姿态和动作，并向服务器发送跟踪数据包。此反馈使服务器能够更新用户的位置、据此渲染场景，并将后续帧传输至头戴式显示器进行显示。

此类架构的典型示例包括 Air Light VR (ALVR)[1] 和 WiVRn[2]。ALVR 是一款采用麻省理工学院许可的虚拟现实流媒体传输应用，可在 SteamVR 服务器与头戴式显示器客户端之间充当无线桥梁。服务器传输视频、音频和触觉数据，而客户端则将跟踪信息和统计信息发回服务器。ALVR 还支持虚拟现实专用功能，例如注视点渲染——降低图像边缘的分辨率，以提高编码效率——以及重投影，这有助于补偿服务器尚未完成渲染的快速头部运动 [2]。

[1]https://github.com/alvr-org/ALVR[2] https://github.com/WiVRn/WiVRn

B. 实时流媒体视频编码

服务器与客户端之间的视频流量通常经过压缩，与原始 RGB 视频相比，可大幅降低带宽需求。然而，压缩会为每个视频帧 (VF) 引入编码和解码时延，从而增加端到端延迟。H.264 和高效视频编码等现代编解码器通常会将帧编码为图像组 (GoP)，其中包括帧内编码帧 (I 帧) 和预测帧 (P 帧) [5]。对于实时流媒体传输，不考虑双向预测帧 (B 帧)，因为编码器端需要访问未来帧才能处理它们。

高效视频编码的帧内刷新模式通过逐步将帧内编码单元插入后续帧中，增强了抗扰能力。这不仅提高了在帧丢失情况下的鲁棒性 [6, 7] ，还使帧大小分布的变化更小 [8]。

C. 高体验质量的要求

由于头戴式显示器的屏幕距离用户眼睛很近，VR 流媒体传输需要较高的分辨率和比特率，以确保视觉保真度并提供良好的体验质量（QoE）。比特率规定了编码过程中采用的压缩程度，并与帧率（FPS）共同决定平均帧大小和视觉质量。本研究考察的比特率范围为 10 至 100 兆比特每秒。

关于时延，我们区分以下两类：a）运动到光子（MTP）时延，即用户实际运动与头戴式显示器中相应视觉更新之间的延迟；b）网络延迟。对于（a），国际电信联盟电信标准化部门建议，360度虚拟现实体验的 MTP 时延应保持在 50 毫秒以内 [9]。这一限制涵盖渲染、编码、网络传输和解码所需的时间。对于（b），涉及强交互的应用场景（例如云端虚拟现实游戏）要求网络往返时间（RTT）为 ≤ 20 毫秒。

对于 ALVR，下行视频丢包的影响是：如果承载视频帧片段的任何一个数据包丢失，整帧就会被视为丢失。在典型比特率下，每个视频帧可能由数十个数据包组成。因此，即使恒定丢包概率看起来很低，只有 0.01%，也可能导致约 1% 的帧丢失率（FLR）。

根据 [10], ，为了确保 ALVR 用户体验流畅，视频帧往返时间（VF-RTT）不应超过 33 ms，且至少 99% 的传输帧应成功接收。在我们的评估中，我们采用 FLR $\leq 1\%$ 和网络延迟 ≤ 33 ms 作为可接受体验质量的阈值。

III. 仿真框架

图1展示了所提出框架中各个主要组件（即服务器、客户端、STA 和接入点）的功能。该框架使用 Rust 和 Nexosim 库 [11] 实现，用于离散事件仿真。

仿真输入（例如用户数量、设备位置、比特率、帧每秒）通过 Bash 脚本配置，该脚本还会组合不同的输入参数，自动迭代场景。对于每个仿真的流媒体会话，都会初始化一对服务器和客户端 `Model` 实例 [11]，并配置各自的地址以及与 STA 和接入点的连接。这些服务器-客户端模块与网络模型相互解耦；网络模型则单独刻画 802.11设备在共享信道上的行为。嵌入接入点和服务器/客户端模块中的日志模块会为每个场景将各用户的指标存储到 CSV文件中。服务器端的比特率管理器模块为未来工作预留；此处假设采用恒定比特率。

通过使用 ALVR 原生数据结构以及真实的应用层下行/上行传输速率来仿真 ALVR 的行为。除控制流量外，所有应用流都通过 ALVR流套接字实例在服务器与客户端之间交换。在通过传输协议发送数据之前，每个有效载荷都会封装在符合 ALVR 规范的结构中，其中包括数据包分片，以及添加应用层标头；标头携带流类型和序列号等元数据。在接收端，流套接字会根据收到的分片重建原始帧。保留这种原生结构，以便在头戴式显示器应用层计算逼真的指标。

每个仿真场景都以配对就绪的服务器和客户端实例开始，可随时启动流传输。流传输会话由仿真模型之间的共享事件触发。服务器和客户端实现的详细信息如下。

A. VR 服务器

SteamVR 和 ALVR 分离式渲染所承担的功能由可配置的编解码流水线取代，该流水线能够以可变的帧率、比特率和转码参数处理 4K 视频。会话中的每条下行链路/上行链路流都使用专用的流套接字，使服务器能够独立调度视频和音频。

音频数据包每 10 毫秒通过音频流套接字成对传输，每个数据包的内容为空，合计大小为 2000 字节。视频由调用 FFmpeg [12] 将示例内容转码为时长为 T_{CHUNK} 秒的编码数据块，并采用用户定义的设置（例如编解码器预设、比特率、GoP 大小、帧内刷新）。编码后的输出存储在视频缓冲区中。

从该缓冲区中以恒定帧每秒提取单个高效视频编码网络抽象层（NAL）单元，将其分割为数据包，并通过流套接字以突发方式发送。缓冲区耗尽后，会对新的分块进行转码，然后重复这一循环。与逐帧编码相比，基于分块的编码更易于集成自适应码率（ABR）算法，并能减少计算开销。视频转码过程的更多详情见第 IV 节。

B. VR 客户端

客户端以 $3\times$ 帧每秒的速率定期生成并发送单个上行跟踪数据包。收到视频帧的所有分片后，ALVR 接收套接字会重建原始编码视频帧。如果在收到首个分片后的 0.1 秒内未能完整重建一帧，则会将其丢弃，并立即通过上行数据包发送帧丢失报告。

成功重建的帧会计入下一个上行统计数据包，编码后的视频帧则被放入容量为 B_{DEC} 的抖动缓冲区 中。该缓冲区会暂时存储待显示的帧，以缓解网络延迟波动并维持稳定的播放速率。模拟的垂直同步机制以恒定帧每秒的速率从缓冲区中取出帧，对其进行解码，并呈现最终的 RGB 输出。

C. Wi-Fi 网络

通过离散事件模拟器模拟 Wi-Fi，该模拟器旨在表示 IEEE 802.11be（Wi-Fi 7）网络。[3] 该模型涵盖接入点和站点的操作，并实现带 RTS/CTS 机制的分布式协调功能。它支持多种传输速率，速率根据接收功率并参照 802.11be 调制与编码方案（MCS）表确定；同时还支持重传、单用户多输入多输出（MIMO），以及通过聚合 MAC 协议数据单元（A-MPDU）进行数据包聚合。

IV. 系统模型

A. VR 仿真框架设置

我们考虑一个由一个接入点和N 个站点组成的 Wi-Fi 网络，每个站点与

[3]虽然尚未纳入多链路操作（MLO）等高级功能，但本研究采用了所有 IEEE 802.11be 基本参数。

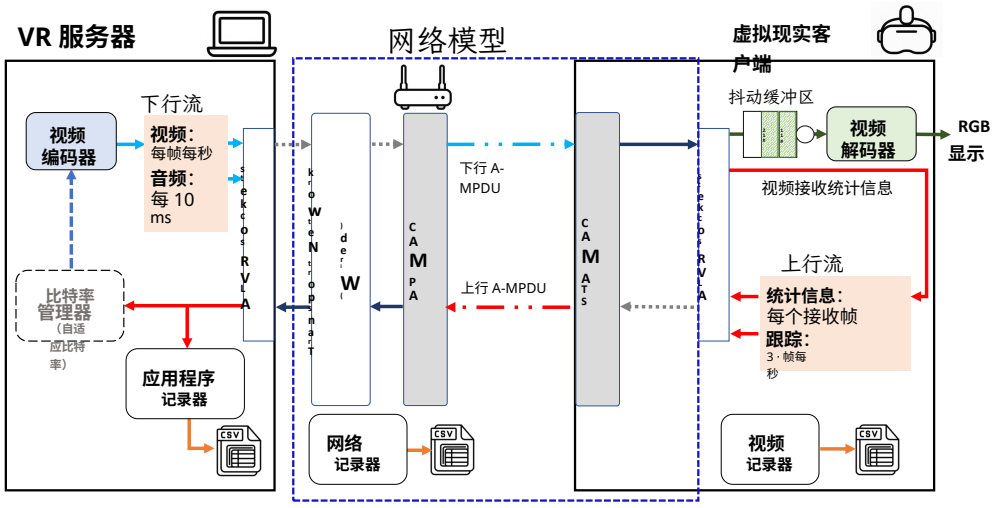


图1： 我们提出的用于模拟多个 VR 流媒体会话的框架示意图。

接入点之间的距离均可配置。与典型的 VR 流媒体传输设置一样，每个站点都运行一个与专用服务器配对的客户端应用程序，服务器通过有线链路连接到接入点。所有流量均通过用户数据报协议 (UDP) 传输。

为了计算每次传输的接收功率，我们采用 [13]，中的 TMB 室内路径损耗模型，并据此为每个用户选择 MCS。当接收信号强度信息 (RSSI) 超过 -46 dBm 时，将选择最高可实现速率的 MCS_{13}，。我们假设采用单用户传输，不包含正交频分多址 (OFDMA) 等功能。传输延迟考虑了采用请求发送 (RTS)/允许发送 (CTS) 方案进行信道接入时，载波侦听多路访问／冲突避免 (CSMA/CA) 引入的时间开销。A-MPDU 中的每个数据包都有 10% 的错误概率，因此，加上冲突的影响，会导致 MAC 层重传。

为生成流媒体内容，我们使用两个 3840×2160 分辨率、帧每秒为 60 的视频样本：*BigBuckBunny*，一部标准的立体视频 3D 动画短片，[4] 以及 *snow*，一段展现专业单板滑雪场景的高动态第一人称视角视频。使用 `ffprobe`测得，它们的平均比特率分别为 12.006 兆比特每秒和86.627 兆比特每秒。

为了在不需要高帧率视频样本的情况下模拟更高的帧率，我们根据模拟帧每秒与原始样本帧每秒之比，按比例调整 FFmpeg 编码命令中使用的比特率。

[4]http://distribution.bbb3d.renderfarming.net/video/mp4/bbb_sunflower_2160p_60帧每秒_stereo_abl.mp4

尽管 FFmpeg 支持其他现代编解码器（例如 AV1、H.264），但为了保持一致性并简洁表达，所有实验均使用带有 快速 预设的高效视频编码。

B. 考虑的性能指标

为了评估 VR 应用在各种条件下的性能，我们考虑以下指标：

- **视频帧往返时间（VF-RTT）：** 服务器传输完整视频帧所需的时间，加上相应的上行链路统计信息数据包到达服务器所需的时间 [10]。该指标反映了应用所体验到的网络延迟。

- **信道利用率（CU）：** 无线传输和冲突所占用的时间与整个仿真时长的比值。

- **帧丢失率（FLR）：** 客户端未能完整接收的视频帧数占服务器传输的总帧数的比例。

- **VMAF 分数：** 基于 Netflix 视频多方法评估融合模型（VMAF） [14]，，该分数通过将头戴式显示器中的每个解码 RGB 帧与同一视频的高比特率参考版本（100 兆比特每秒）进行比较来计算。两个版本采用相同的帧内刷新或图像组配置。所得感知质量分数范围为 0 到 100。

V. 结果

本节利用所提出的框架评估 Wi-Fi 支持及时虚拟现实流量的能力。通过改变网络和编解码器参数，并监测应用和网络指标，我们评估不同虚拟现实会话配置、用户数量以及异构调制编码方案速率的影响。关键仿真参数列于表 I 中。

表 I：仿真参数。

参数	值
仿真时间	20 s
频段	5 GHz
信道带宽	80 MHz
发射功率	20 dBm
最大 A-MPDU 大小	64 个数据包
最大 A-MPDU 传输时间	4.85 毫秒
竞争窗口	8 个时隙
FFmpeg 版本	7.1
视频块时长（T_{CHUNK}）	1.5 秒
抖动缓冲区容量（B_{DEC}）	2 帧
VMAF 模型	`libvmaf 4K v0.6.1`

A. GoP 大小、帧内刷新和比特率对视频质量及时延的影响

视频帧大小分布会随编码场景和编解码器设置而变化。我们考虑一个场景，其中单个头戴式显示器放置在距离接入点 1.5 m 处，以隔离 GoP 大小、比特率和帧内刷新等编解码器参数对感知质量及时延的影响。图 2 展示了不同编解码器设置下、以 90 帧每秒传输的两个视频样本的帧大小、VMAF 和 VF-RTT 分布（使用小提琴图）。图中每个元素代表观测值的总体范围，黑线表示中位数。

结果表明，与 GoP 相比，帧内刷新可显著降低帧大小和 VF-RTT 的变化，但由于冗余增加，VMAF 会略有降低 [6, 7]。较大的 GoP 大小通过降低 I 帧频率来减少变化，从而提高压缩效率，并使给定比特率下的 VMAF 略有提升。在较高比特率下，VMAF 的提升逐渐减小；60 兆比特每秒的比特率在所有编解码器设置下都能达到良好的感知质量（VMAF ≈ 90）。虽然较高比特率会降低 VMAF 分数的变化，但帧大小和时延变化会随比特率上升而增加。具体而言，80 兆比特每秒的帧内刷新流，其 VF-RTT 方差范围与 40 兆比特每秒的 GoP 流相当，VMAF 范围也相似。因此，我们建议优先选择较高比特率的帧内刷新，而非较低比特率的 GoP，以提升时延稳定性。

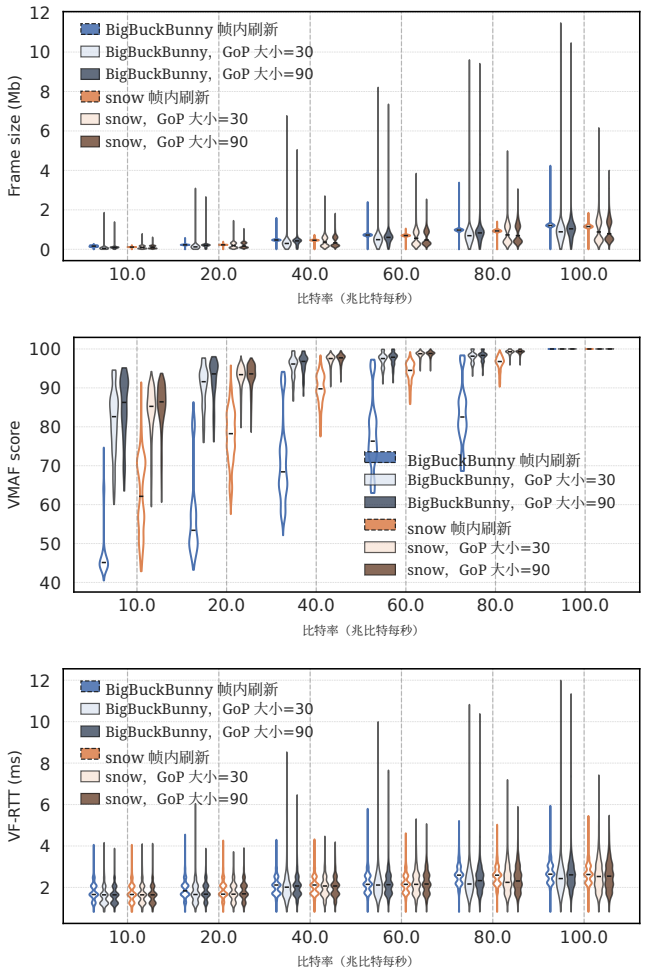
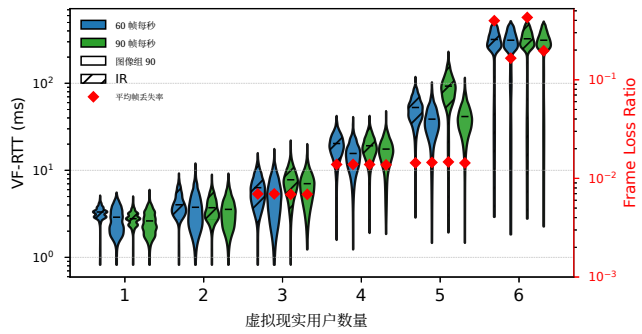


图2：以 90 FPS 流传输的两个视频样本，在 6 种比特率下采用不同图像组和帧内刷新设置时，帧大小（上）、VMAF 分数（中）和视频帧往返时间（下）的分布。

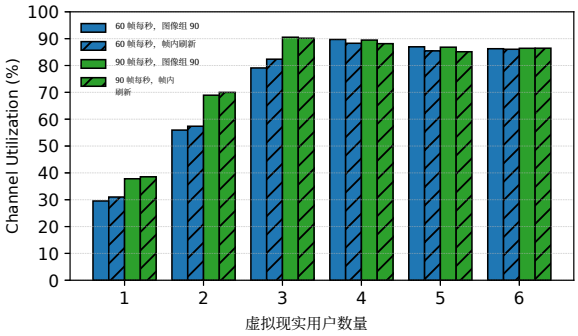
B. 虚拟现实用户数量增加的影响

接下来，我们分析同时使用虚拟现实用户数量增加时的网络服务质量，并比较帧内刷新与大小为 90 的图像组。每位用户都以恒定比特率 100 兆比特每秒、60 或 90 FPS 流传输*snow* 视频样本，且所有用户均位于距接入点 1.5 米处。图3展示了用户数量从 1 增至 6 时，视频帧往返时间（针对正确接收的帧）和帧丢失率的分布，以及平均信道利用率。用户数不超过 3 时，帧丢失率保持在 1% 以下；用户数为 4 和 5 时，视频帧往返时间的波动急剧增大（中位数约为 20 和 50 毫秒），且帧丢失率超过 1%。因此，在我们的模型中，4 个用户是使用 100 兆比特每秒恒定比特率流时维持边缘服务质量的临界值。

尽管帧内刷新始终能降低视频帧往返时间的方差，但无论使用何种编解码器或

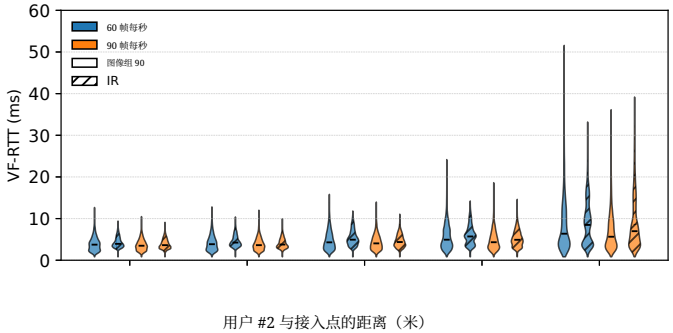


(a)

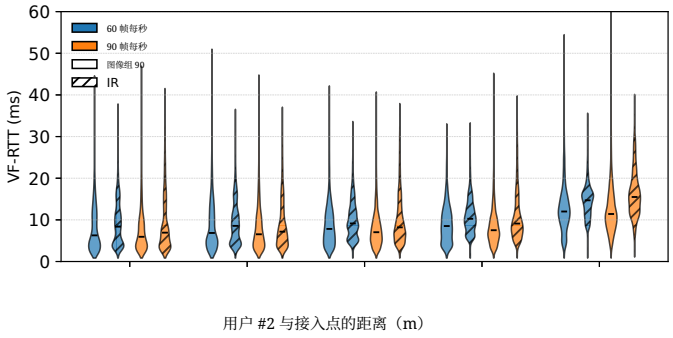


(b)

图3：不同用户数下的网络性能，通过（a）VF-RTT分布和FLR平均值、（b）CU展示。



(a)



(b)

图4：两名用户的VF-RTT分布，其中一名位于固定位置，另一名逐渐远离接入点。（a）用户#1位于距接入点1.5 m处；（b）用户#1位于距接入点11 m处。

在 4 个用户时，信道都会饱和。较高的帧每秒会因传输频率更高而增加信道利用率（图3b）。在 5 至 6 个用户时，信道利用率开始下降，这是因为现有的网络拥塞妨碍了及时获得传输机会，从而导致聚合 MAC 协议数据单元传输规模更大、空中传输时间效率更高。

C. 接入点与头戴式显示器的距离及异构调制编码方案的影响

传输速率取决于 MCS，而 MCS 会随用户距离变化。MCS 较低的用户会占用更多信道空闲时间，从而显著增加其他用户的排队时延——这一现象称为 Wi-Fi 性能异常 [15]。我们评估两名虚拟现实用户以 100 Mbps 和 60/90 FPS 流式传输*snow*视频的情况：将其中一名用户（用户 # 1）固定在距离 AP 1.5 米（图 4a）或 11 米（图 4b）处，同时改变另一名用户（用户 #2）的距离，以引入 MCS 差异。我们同时考虑帧内刷新和大小为 90 的图像组。图 4 证实，当一台头戴式显示器（用户 #2）逐渐远离 AP 时，其时延服务质量会下降，而处于有利位置的另一名用户（用户 #1）的时延服务质量也会下降。在图 4a 中，用户 #1 的 VF-RTT 增加源于用户 #2 采用较低 MCS 进行传输所引起的排队时延。这一时延源于向距离较远的站点传输全部下行流量，或接收来自该站点的上行传输所需的时间。无论选择何种编解码器或 FPS，VF-RTT 的中位数和方差都会随着用户 #2 与接入点之间距离的增加而增大。当用户 #1 距离 AP 11 米时（图 4b），时延下降更加明显，在所有配置下都超过 33 毫秒的阈值。

VI. 结论

本文提出了一个虚拟现实流媒体仿真框架，用于联合评估 Wi-Fi 场景和视频编解码配置。仿真结果展示了网络性能如何随不同编解码器和网络参数选择而变化，并基于这些观察结果提出了相关建议。具体而言，结果突出了通过视频多方法评估融合（VMAF）衡量的视频质量与图像组（GoP）编码策略和帧内刷新（IR）编码策略下帧大小之间的权衡。当并发用户数超过四个时，无论选择 GoP 还是 IR，信道都会达到饱和。对于较少的用户数（达到饱和之前），IR 可降低视频帧往返时间（VF-RTT）的方差。在未来的工作中，我们计划将所提出的框架扩展到开放测试平台，以评估自适应码率算法和

先进的 Wi-Fi 解决方案。其中包括支持下行链路/上行链路协调式 XR 流量调度、多链路操作和多接入点协调

参考文献

[1]F. Alriksson、C. Phillips、J. L. Pradas、A. Zaidi等，“XR 与 5G：通过时间关键型通信实现大规模扩展现实”，*Ericsson technologyreview*，第 2021 卷，第 8 期，第 2–13 页，2021年。[2] alvr-org，“ALVR 的工作原理”，https://github.com/alvr-org/ALVR/wiki/How-ALVR-works，访问于2025年6月27日。[3] M. Carrascosa-Zamacois 等，“面向 XR 流媒体的 MLO 性能评估：Wi-Fi 7 能否达到预期？”，载于*2024 IEEE* 第*29*届计算机辅助建模、设计、通信与网络会议（*CAMAD*）。IEEE，2024年，第 1–6 页。[4]M. Casasnovas、M. Carrascosa-Zamacois 和 B. Bellalta，“基于云的 VR 流媒体能否应对 Wi-Fi 重叠基本服务集争用？”*arXiv* 预印本 *arXiv:2507.07677*，2025年。[5] I. E. Richardson，视频编码：高效视频编码及后续技术实用指南。John Wiley & Sons，2024年。[6]H. Chen、C. Zhao、M.-T. Sun 和 A. Drake，“面向低时延抗错误视频编码的自适应帧内刷新”，*Journal of Visual CommunicationandImage Representation*，第 31 卷，第 294–304 页，2015年。[7] A. Belbel、A. Bekhouch 和 N. Doghmane，“帧内刷新对多视点高效视频编码抗错误能力与压缩效率的影响”，第 1–8 页，2024年。[8]E. Korneev、M. Liubogoshchev、D. Bankov 和 E. Khorov，“如何对云 VR 建模：影响因素的实证研究”，*IEEE Open Journalof the Communications Society*，第 5 卷，第 4155–4170 页，2024年。[9]国际电信联盟电信标准化部门，“通过集成宽带有线网络增强云 VR 服务交付的端到端网络平台功能要求”，国际电信联盟电信标准化部门，技术报告 J.1631，2021年11月，2021-11-24 批准，状态：生效。[在线]。可查阅：https://www.itu.int/rec/T-REC-J.1631-202111-I/en [10]M. Casasnovas、F. Maura、I. Vandebroeck、H. Sukmawanto、E. Joris 和 B. Bellalta，“NeSt-VR：面向 Wi-Fi 虚拟现实流媒体的自适应码率算法”，2025年。 [在线]。可查阅：https://arxiv.org/abs/2502.14947[11] Asynchronics，“Nexosim”，GitHub代码仓库，5月 2025, [在线]。版本 0.3.3；于5月16日发布，2025年。访问日期：2025年7月2日。 [在线]。可查阅：https://github.com/asynchronics/nexosim [12]FFmpeg开发者，“FFmpeg 工具”，https://ffmpeg.org， 2025, [软件]。[13] T. Adame、M. Carrascosa 和 B. Bellalta，“5 GHz 室内WiFi场景的 TMB路径损耗模型：RSSI、调制与编码方案和空间流之间的经验关系”，载于*2019*年无线日（*WD*），2019年，第 1–8 页。[14]Netflix，“VMAF：视频多方法评估融合”，GitHub代码仓库，2016年，可查阅：https: //github.com/Netflix/vmaf。[15]B. Bellalta、A. Zocca、C. Cano、A. Checco、J. Barcelo 和 A. Vinel，“采用连续时间马尔可夫网络分析 CSMA/CA 网络吞吐量：教程”，无线网络移动对象：协议、架构、工具、服务与应用，第 115–133 页，2014年。