

一、关于传统Edge caching

现有研究内容：

- 1、调度
- 2、卸载
- 3、可靠性
- 4、这里找到个很全的论文合集 [QAQ](#)

目前其他的可研究方向：

二、VR Edge caching

- 1、Challenges in caching tile-based 360° videos are:
- 2、VR transmission solution
- 3、VR Caching Algorithms

针对Fov:

内容替换策略:

结合神经网络:

4、一些案例

VR单玩家案例Furion:

多用户移动VR:

VR 直播特殊的地方:

三、区别

四、参考文献

一、关于传统Edge caching

现有研究内容：

- 调度
- 卸载（卸载决策、资源分配和卸载系统实现）
- 可靠性

1、调度

1. 提出了一个应用卸载优化模型，该模型是在一个名为移动边缘云计算的集成计算环境中调度应用任务：**a mixed integer linear programming model**（一种混合整数线性规划模型）权衡cost和energy。

From: Data-intensive application scheduling on Mobile Edge Cloud Computing

2. 研究了具有社会群体的复杂网络在边缘计算环境下的最大完工时间最小化工作流调度问题--整数规划，利用贪婪搜索（IGS）算法提出了一种改进的复合启发式算法（ICH）。与传统的循环式调度算法相比可行性有显著提升。

From: Online Deadline-Aware Task Dispatching and Scheduling in Edge Computing

3. 一个基本问题是如何调度和调度作业，以便最小化作业响应时间（定义为作业释放与计算结果到达设备之间的间隔）：提出了在移动设备上以任意顺序和任意时间生成作业，然后将其卸载到具有上载和下载延迟的服务器上来。与启发式算法相比，最大程度地减少所有作业的总加权响应时间。

From: OnDisc: Online Latency-Sensitive Job Dispatching and Scheduling in Heterogeneous Edge-Clouds

4. FPS在线调度算法

From: Energy-Efficient Scheduling of Interactive Services on Heterogeneous Multicore Processors

2、卸载

卸载决策主要解决的是移动终端决定如何卸载、卸载多少以及卸载什么问题；

表 2 MEC 中卸载决策方案总结归类			
优化目标	卸载类型	相关文献	关键研究点
以降低时延为目标	全部卸载	文献[24-26]	<ul style="list-style-type: none">以降低时延为目标的卸载决策方案分析时延模型多用户的计算卸载问题，建模为 NP-hard 问题基于 Lyapunov 优化的动态卸载算法联合优化通信资源和计算资源分配采用 Stackelberg 博弈论的方法优化多用户卸载方案
	部分卸载	文献[27-28]	<ul style="list-style-type: none">研究卸载内容之间的依赖关系与计算任务全部卸载的时延相对比采用启发式算法
以降低能量消耗为目标	全部卸载	文献[29-31]	<ul style="list-style-type: none">在保证时延的要求下，以降低能耗为目标的卸载决策方案能量模型的优化方案相关参数的对比分析，如信道链路状况、CPU、系统容量等在线学习方案和预先计算的离线策略采用人工鱼群算法、约束性马尔可夫链方法
	部分卸载	文献[32-33]	<ul style="list-style-type: none">研究卸载内容之间的依赖关系与计算任务全部卸载的能耗相对比基于阈值结构的最优卸载方案采用凸优化的方法
权衡能耗和时延为目标	全部卸载	文献[35-37]	<ul style="list-style-type: none">以权衡能量和时延为目标的卸载决策方案能量和时延的权衡模型建立采用基于 Lyapunov 的优化算法、在线学习策略
	部分卸载	文献[34]	<ul style="list-style-type: none">研究卸载内容之间的依赖关系与计算任务全部卸载的时延和能耗均衡方案对比

资源分配则重点解决终端在实现卸载后如何分配资源的问题；

表 3 MEC 中资源分配方案总结归类				
节点数量	优化目标	参考文献	计算卸载方案	仿真优化结果
单节点	时延	文献[38]	通过动态考虑整个迁移过程来最优化 VM 分配方案	减少 46%的时延以及减小 80%的迁移成本
		文献[39]	提出了 MEC 在满足应用程序时延要求的同时,提供服务的应用程序数量最大化为目标的卸载方案，通过优先级来分配计算节点	减小 25%的时延
		文献[26]	考虑了 MEC 的计算资源有限的情况，如何进行资源分配的问题。提出了分层的 MEC 部署架构	—
	时延和能耗	文献[40]	通过分配索引策略来让 UE 选择合适的 MEC 服务器,解决了高复杂度和高通信开销问题	索引策略和普通策略相比减小了 7%的能耗
	能耗	文献[43]	提出了合作式缓存和卸载方案，MEC 服务器联合起来为 UE 执行计算和缓存任务	—
多节点	时延	文献[44]	采用深度学习的方法分配资源，提出了动态的卸载方案	减小了 50%的能耗
		文献[45]	多用户卸载，将问题建模为 NP-hard 问题，并采用背包模型去优化整个资源分配和负载均衡的问题	相比于全部在 UE 执行任务减少了 70%的延时，比在 CC 执行减小了 58%的时延
	能耗和时延	文献[46]	提出了干扰管理的方案，在最小化干扰的条件下进行通信资源分配、计算资源分配的方案	减小了 40%的时延
		文献[41]	提出了通过优化时延和基站能耗的集群选择策略，并对比分析不同的回传技术和网络拓扑的影响	—
		文献[42]	提出 3 种不同的云集群选择策略，分别以优化时延、优化集群总能耗和优化集群中每个 SCeNB 的能耗为目标	减少了 22%的时延，降低了 61%的能耗

以上两张图From: 《移动边缘计算卸载技术综述》

卸载系统的实现，则侧重于移动用户迁移过程中的实现方案。

常用的清空策略：

- FIFO(first in first out)

先进先出策略，最先进入缓存的数据在缓存空间不够的情况下（超出最大元素限制）会被优先被清除掉，以腾出新的空间接受新的数据。策略算法主要比较缓存元素的创建时间。在数据实效性要求场景下可选择该类策略，优先保障最新数据可用。

- **LFU(less frequently used)**

最少使用策略，无论是否过期，根据元素的被使用次数判断，清除使用次数较少的元素释放空间。策略算法主要比较元素的hitCount（命中次数）。在保证高频数据有效性场景下，可选择这类策略。

- **LRU(least recently used)**

最近最少使用策略，无论是否过期，根据元素最后一次被使用的时间戳，清除最远使用时间戳的元素释放空间。策略算法主要比较元素最近一次被get使用时间。在热点数据场景下较适用，优先保证热点数据的有效性。

除此之外，还有一些简单策略比如：

- 根据过期时间判断，清理过期时间最长的元素；
- 根据过期时间判断，清理最近要过期的元素；
- 随机清理；
- 根据关键字（或元素内容）长短清理等。

[文章地址](#)

3、可靠性

...好像不是重点）

4、这里找到个很全的[论文合集](#) QAQ

目前其他的可研究方向：

1. 将深度强化学习技术和联合学习框架与移动边缘系统相集成，以优化移动边缘计算，缓存和通讯

<https://zhuanlan.zhihu.com/p/91815676>

2. 设计一个优秀的内容缓存策略：要解决两个问题：哪些内容和多少内容应该被缓存

解决1：作为一个群体推荐系统，预测移动边缘上要缓存的热门内容的 Top-N 列表。如果请求的内容已经被缓存在移动边缘，用户可以直接快速访问，而无需连接到云端。确定哪些被缓存：潜在语义索引（latent semantic indexing, LSI）和协同过滤（collaborative filtering, CF）

3. 如何高效按需刷新缓存？

Shan Zhang, Junjie Li, Hongbin Luo, Jie Gao, Lian Zhao, Xuemin Shen, “Low-latency and fresh content provision in information-centric vehicular networks,” IEEE Transactions on Mobile Computing, 2020, online published.

4. 时效性和延时能同时满足吗？

Shan Zhang, Hongbin Luo, Junling Li, Weisen Shi, and Xuemin Shen, “Hierarchical soft slicing to meet multi-dimensional QoS demand in cache-enabled vehicular networks” IEEE Transactions on Wireless Communications, vol. 19, issue 3, pp. 2150 – 2162, Mar. 2020.

5. 缓存价值：流行度与动态性如何取舍？

Shan Zhang, Liudi Wang, Nu Zhang, Hongbin Luo, Sheng Zhou, “Placement of dynamic content items in mobile edge caching,” in ITC’32, Osaka, Japan, Sep. 2020.

D2D 通信中的缓存是有限的。

移动边缘的资源利用不足。许多重要的工作都集中在利用移动边缘的计算和控制资源的高效计算卸载方案上。但是，移动边缘的缓存能力还没有得到充分的讨论。例如，为了满足主流用户的需求，哪些内容、多少内容应该被缓存到移动边缘上。

情境信息对于终端的智能决策和控制至关重要，其内容的动态变化提出了信息时效性的新需求。传统缓存方法关注静态内容，会导致情境信息过期失效等问题。那么如何在边缘缓存里高效满足这类新应用的需求呢？

二、VR Edge caching

1、Challenges in caching tile-based 360° videos are:

1. How to populate the finite cache sized with tiles of the appropriate resolution（缓存什么和合适的分辨率是什么）
2. which tiles to evict from the cache to make space for new tiles whenever the cache capacity is exceeded.（清除策略）

2、VR transmission solution

为提高360 VR视频传输效率，利用adopting tiling and multicast technologies（平铺和组播技术）最优的组播传输方案：一是通过优化子载波、传输功率和传输速率分配，最大限度地提高正交频分多址(OFDMA)系统的接收视频质量，另一种是通过优化传输时间和功率分配，使平均传输能量最小。

在[26]中，在混合质量片的传输时间和功率分配约束和质量平滑约束下，对VR视频质量等级选择、传输时间分配和传输功率分配进行优化，使总效用最大化。

在[27]中，作者提出了一种基于多播虚线的平铺流解决方案，包括一种基于用户视图的平铺加权方法和速率自适应算法，为VR用户提供沉浸式体验。

在[28]中，作者利用概率的方法来预取贴图来对抗视口预测错误，并设计了一个基于qos驱动的视口自适应系统，该系统可以实现较高的视口PSNR

3、VR Caching Algorithms

针对Fov:

在[34]中，作者提出了一种基于公共视场的学习概率用户请求模型的视场感知缓存策略，该策略比传统的缓存策略至少提高了缓存命中率40%。

在[33]中，作者提出了一种新的基于mec的移动通信系统VR交付框架，能够在移动VR设备上提前缓存部分视场(fov)，并运行某些后处理程序。

在[1]为了解决离线优化方法的不实际的开销，提出了一种在线复杂度较低的MaxMinDistance在线缓存算法来提高命中率。同时通过上下左右的视图来合成中心图块。

内容替换策略：

在[11]中,作者提出一种新的方法对缓存的内容替换,允许传输延迟并设计一个优化框架,允许基站选择合作的缓存/呈现/流策略，当为每个基站的用户提供给定的缓存/计算资源时，最大限度地提高他们的总奖励

MM18-Fov Aware caching 当缓存容量超过时，缓存模块应用这些知识来确定清除块的优先级。

结合神经网络：

[2]利用AI的方法决策寻找最优解来解决资源的分配问题 通过考虑CPU cycle frequency, access jurisdiction, radio-frequency, bandwidth等等.

[12]的作者研究内容缓存和传输在无人机(UAV)无线网络和提出一个分布式虚拟现实(VR)深层神经网络学习算法，汇集了新的想法从液体状态机(LSM)和回声状态网络(esn)来解决联合内容缓存和传输问题。

[35]的作者设计了一种基于网络功能虚拟化(NFV)的虚拟缓存(vCache)来动态管理视频块，在存储和计算成本之间进行了权衡，从而降低了ABR流媒体的运营成本。

4、一些案例

微软提出的FlashBack：预渲染+缓存预测

VR单玩家案例Furion：

Furion：是一个VR视频的框架，分为前台交互(FE)和后台交互(BE)。利用了有关VR前景交互和后台环境具有可预测性和渲染工作负载的对比，并采用了在手机和服务器上运行的拆分渲染器体系结构。辅以视频压缩，全景帧的使用，电话上多个内核的并行解码以及基于视图的比特率适配。[4]

多用户移动VR：

Coterie：通过以下4个任务来渲染下一格的帧：FI和近BE渲染，解码从服务器接收到的远BE帧，在所有玩家之间同步FI，合并远BE，FI和近BE帧。每个Coterie客户端的帧缓存都缓存了克隆所预取的BE帧。高速缓存查找方案可以将一些帧重用于附近的位置。[3]

VR 直播特殊的地方：

- 一个视频块最多应该在缓存中存储几十秒。
- 如果视频中有一个吸引点，用户很可能会要求围绕这个点的贴图，但不会要求远离这个点的贴图。(通过用户的观看行为可以推断出贴图的人气)

From: Flock-based

三、区别

VR的：

- 更短的时延
- 不确定的Fov
- 媒体的需要储存量大

建立VR框架，替换策略，结合Fov，多用户VR。

传统的：

- 调度
- 卸载

四、参考文献

[1]A View Synthesis-Based 360° VR Caching System Over MEC-Enabled C-RAN

[2]Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence.

[3]Coterie: Exploiting Frame Similarity to Enable High-Quality Multiplayer VR on Commodity Mobile Devices

[4]Furion: Engineering High-Quality Immersive Virtual Reality on Today's Mobile Devices

[11] J. Chakareski, “VR/AR immersive communication: Caching, edge computing, and transmission trade-offs,” in *Proc. Workshop Virtual Reality Augmented Reality Netw.*, Aug. 2017, pp. 36–41.

[12] M. Chen, W. Saad, and C. Yin, “Echo-liquid state deep learning for 360° content transmission and caching in wireless VR networks with cellular-connected UAVs,” 2018, arXiv:1804.03284. [Online]. Available: <https://arxiv.org/abs/1804.03284>

[26] K. Long, C. Ye, Y. Cui, and Z. Liu, “Optimal multi-quality multicast for 360 virtual reality video,” in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–6.

[27] H. Ahmadi, O. Eltobgy, and M. Hefeeda, Adaptive multicast streaming of virtual reality content to mobile users, in Proc. Thematic Workshops ACM Multimedia (Thematic Workshops), New York, NY, USA, 2017, pp. 170 178. doi: 10.1145/3126686.3126743.

[28] L. Xie, Z. Xu, Y. Ban, X. Zhang, and Z. Guo, 360ProbDash: Improving QoE of 360 video streaming using tile-based HTTP adaptive streaming, in Proc. ACM Multimedia Conf. (MM), Mountain View, CA, USA, Oct. 2017, pp. 315 323. doi: 10.1145/3123266.3123291.

[33] Y. Sun, Z. Chen, M. Tao, and H. Liu, Communications, caching and computing for mobile virtual reality: Modeling and tradeoff, 2018, arXiv:1806.08928. [Online]. Available: <https://arxiv.org/abs/1806.08928>

[34] A. Mahzari, A. T. Nasrabadi, A. Samiei, and R. Prakash, FoVaware edge caching for adaptive 360 video streaming, in Proc. ACM Multimedia Conf. Multimedia Conf., 2018, pp. 173 181.

[35] G. Gao, Y. Wen, and J. Cai, “vCache: Supporting cost-efficient adaptive bitrate streaming,” IEEE MultimediaMag., vol. 24, no. 3, pp. 19–27, Aug. 2017.