基于视口预测的全景视频传输关键技术研究

摘要

由于计算机科学和多媒体技术的不断创新，虚拟现实因其沉浸式构想式特性得到越来越多的关注，全景视频作为虚拟现实技术重要的组成部分也在休闲、教育和医学等领域得到了应用。但是，全景视频数据量大、时延敏感的特点对现有传输网络造成了巨大的压力。因此，如何在有限的网络带宽中最大程度地满足多个用户的观看体验至关重要。本文主要从全景视频视口预测和物理资源调度两方面来优化全景视频的传输。

近年来，基于视口自适应的全景视频传输方案得到越来越多的研究，而该方案的关键前提就是视口位置的精准预测。我们基于现有研究改进了基于用户历史观看轨迹的视口预测算法，首先根据当前用户的历史视口数据使用长短期记忆模型初步预测出视口中心位置的经纬度坐标，然后我们结合其他用户的视口数据对初步预测结果进行调整以调高准确度。实验结果表明，我们提出的预测算法在准确度方面比起基准算法有了一定的提升。

另一方面，如何科学地调度带宽资源满足不同类型业务的用户需求也是研究重点。我们首先研究了LTE系统中几种常见的下行资源调度算法，考虑到这些算法在调度视频等实时业务时有一些不足之处，我们提出了一种使用Q学习来进行调度决策的改进算法，可以根据实时变化的网络状态在每个调度间隔自适应地选择最合适的调度算法。仿真实验表明我们提出的算法在吞吐量、丢包率和时延等指标上优于经典算法。

关键词 全景视频 视口预测 LTE 资源调度

RESEARCH ON PANORAMIC VIDEO TRANSMISSION

BASED ON VIEWPORT PREDICTION

ABSRTCT

Due to the continuous innovation of computer science and multimedia technology, virtual reality has received more and more attention due to its immersive conception. As an important part of virtual reality technology, panoramic video has also been applied in the fields of leisure, education and medicine. However, the large amount of panoramic video data and delay-sensitive features have put tremendous pressure on existing transmission networks. Therefore, how to maximize the viewing experience of multiple users in a limited network bandwidth is crucial. This article mainly optimizes the transmission of panoramic video from the perspective of panoramic video viewport prediction and physical resource scheduling.

In recent years, more and more researches have been made on the viewport adaptive panoramic video transmission scheme, and the key premise of this scheme is the accurate prediction of the viewport position. Based on the existing research, we improved the viewport prediction algorithm based on the historical viewing trajectory of the user. First, based on the historical viewport data of the current user, we used the long-term and short-term memory model to initially predict the latitude and longitude coordinates of the center of the viewport. Then we combined the views of other users Adjust the preliminary forecast results to improve the accuracy. The experimental results show that the accuracy of the proposed prediction algorithm is improved compared with the benchmark algorithm.

On the other hand, how to scientifically schedule bandwidth resources to meet the needs of users of different types of services is also a research focus. We first studied several common downlink resource scheduling algorithms in LTE systems. Considering that these algorithms have some shortcomings when scheduling real-time services such as video, we propose an improved algorithm that uses Q learning to make scheduling decisions. The most appropriate scheduling algorithm is adaptively selected at each scheduling interval according to the network state that changes in real time. Simulation experiments show that our proposed algorithm is superior to classical algorithms in terms of throughput, packet loss rate, and delay.

目录

[第一章 绪论 1](#_Toc33123538)

[1.1 研究背景与意义(6537) 1](#_Toc33123539)

[1.2 国内外研究现状 3](#_Toc33123540)

[1.2.1 全景视频视口预测研究现状 3](#_Toc33123541)

[1.2.2 LTE资源调度算法研究现状 5](#_Toc33123542)

[1.3 主要研究内容 7](#_Toc33123543)

[1.4 本文组织结构 7](#_Toc33123544)

[第二章 基于历史观看轨迹的视口预测算法 9](#_Toc33123545)

[2.1 视口预测相关技术 9](#_Toc33123546)

[2.1.1 相关定义 9](#_Toc33123547)

[2.1.2 常用预测模型 12](#_Toc33123548)

[2.2 基于历史观看轨迹的视口预测算法 16](#_Toc33123549)

[2.2.1 数据集介绍 16](#_Toc33123550)

[2.2.2 LSTM介绍 16](#_Toc33123551)

[2.2.3 视口预测模型 18](#_Toc33123552)

[2.3 实验与结果分析 21](#_Toc33123553)

[2.3.1 实验过程 21](#_Toc33123554)

[2.3.1 评价指标 23](#_Toc33123555)

[2.3.2 实验与结果分析 23](#_Toc33123556)

[2.4 本章小结 26](#_Toc33123557)

[第三章 基于Q-Learning的LTE资源调度算法 28](#_Toc33123558)

[3.1 LTE架构与资源调度 28](#_Toc33123559)

[3.1.1 LTE网络架构与协议架构 28](#_Toc33123560)

[3.1.2 LTE架构关键技术 30](#_Toc33123561)

[3.1.3 无线资源调度 32](#_Toc33123562)

[3.2 基于Q-Learning的资源调度算法 36](#_Toc33123563)

[3.2.2 DSQL算法介绍 37](#_Toc33123564)

[3.3 实验与结果分析 41](#_Toc33123565)

[3.3.1 仿真工具介绍 41](#_Toc33123566)

[3.3.2 仿真环境与参数 43](#_Toc33123567)

[3.3.3 仿真结果与分析 44](#_Toc33123568)

[3.4 本章小结 48](#_Toc33123569)

[第四章 总结与展望 49](#_Toc33123570)

[参考文献 51](#_Toc33123571)

[致谢 56](#_Toc33123572)

[攻读学位期间发表的学术论文 58](#_Toc33123573)

# 第一章 绪论

1.1 研究背景与意义

由于计算机科学和多媒体技术的不断创新，VR（Virtual Reality, 虚拟现实）技术开始逐步进入大众的关注范围，VR技术构建的虚拟环境与真实环境真假难辨，用户从传统的以视觉为主的局限体验扩展到包含视觉、听觉、触觉甚至嗅觉的综合感官体验，在虚拟环境中体验到彷佛身处在真实世界的感觉。

VR有沉浸性、交互性和构想性的特点[1]。沉浸性是指当用户进入到虚拟环境时，听到的看到的体验到的感受与真实世界完全一致，这是衡量虚拟现实技术优劣的关键指标；交互性指用户能够与虚构世界互动。区别于传统的只能观看的用户体验，虚拟现实技术通过力反馈装置、陀螺仪、跟踪器等多种传感设备使得用户可以跟虚拟环境进行交互，例如，用户可以在射击游戏中拿起枪支对目标进行射击，可以蹲起躲避敌人，且整个交互过程遵循各种物理学定律，与真实世界一致；构想性也被称作想象性，用户在虚拟环境中不仅能够体验到实际环境中的感觉，还可以自由丰富地创建出真实世界不可能发生的事情，激发用户的创新性。

VR的概念可以追溯到20世纪80年代初，发展至今已经在影视休闲、教育学习和远程医疗等领域得到了广泛的应用。工业和信息化部表示，2018年全世界内VR的产业规模已将近千亿元，同时中国的VR产业规模也在一直扩大，预计到2021年VR的市场规模将达到554亿元，工信部还表示将给予虚拟现实制造业多方面的支持，助力VR技术在教育、商业业、文娱等领域得到更多运用[2]。VR技术巨大的发展市场使得人们不断对其进行探索与研究，其中最流行的应用方向之一的全景视频也逐渐成为人们争相研究的重点。

全景视频，又被称作360度视频，是一种用户可以随意转变观看方向的三维视频。传统的二维视频的缺点主要是缺乏互动性，用户只能被动观看视频内容，无法像在真实世界中那样可以随意转动头部观看不同的场景画面，全景视频的出现则打破了这种僵局。目前在各大网络平台看到的全景视频主要由特殊的全景摄像机拍摄而成，全景摄像机通常由若干个普通相机组成，可以拍摄多个角度的画面，然后通过特征点检测、特征匹配和融合等一系列步骤将各个角度的视频画面拼接成360度得到的。用户在观看全景视频时需要佩戴特殊的硬件设备VR眼镜或者HTM（Head Mounted Display, 头戴式显示仪），然后用户就处在全景摄像机的位置观看360度的视频画面，用户可以自由地转动头部，视频画面也会随着变化，如同在真实环境中体验的一样。

由于相对于一般的平面视频而言，全景视频更加贴近用户的眼睛而对视频分辨率有着更为苛刻的要求，所以即使在相同的单眼分辨率情况下，由于全景视频需要保留360度的画面信息，全景视频的数据量也要比普通视频的数据量高许多，一般为前者的5~6倍。同时，全景视频对网络系统的传输时延有着较为严格的要求，行业认为传输系统的MTP（Motion-to-Photos Latency, 头动到画面延迟），即转动头部和视频画面对应切换之间的响应时间必须控制在20ms以内[3]，否则由于生理原因，用户会感到头晕恶心从而造成极差的观看体验。因此，全景视频数据量大、时延敏感的传输特性对现有的无线传输网络造成了极大的压力，如何在现有传输网络中为多个用户提供高水平的全景视频业务成为了棘手问题。而在实际观看场景中，由于人眼角度的特性，在某一时刻，用户实际只能看到全景视频中的部分内容即FOV（Field Of Viewport, 视口区域），如果以同等质量传输所有的画面信息势必会造成网络资源的巨大浪费，如何根据视口区域自适应传输全景视频内容也收到越来越多的关注。

除此之外，不断增长的用户数量使得接入无线网络的终端设备逐年增多，无线网络传输的业务类型也从传统的二维视频、音频及图文扩展到 新兴的全景视频等等，不同业务的QoS（Quality of Service, 服务质量）参数不同，如全景视频这种实时业务对系统的传输时延有着较高的要求，而文本这种非实时业务则要求传输系统有着较低的丢包率来保证信息的完整度，如何合理利用有限的带宽资源来满足不同业务的需求也是极为关键的问题之一。LTE（Long Term Evolution, 长期演进）系统基站侧的无线资源管理有着十分重要的功能，，而其中涉及的资源调度算法则是关键。所谓资源调度，就是通过一定方式计算小区内等待调度的业务的优先等级，然后根据等级顺序将带宽资源依次分配给业务，直到所有待调度业务都满足了传输需求或者带宽资源全被占用。通常而言，资源调度算法需要考虑网络时延、吞吐量、时延以及用户间的公平性等多个方面，一个高效的调度算法则可以在各个方面取得均衡，使得效益最大化。因此，如何通过合理的调度与分配来有效利用短缺的带宽资源，，为多用户提供高质量低时延的全景视频业务并且保证其他传统业务的质量，也是目前全景视频传输优化工作的重点与难点。

1.2 国内外研究现状

全景视频作为一种新兴的多媒体业务，涉及到多方面的关键技术，包括图像拼接、三维投影、自适应传输以及客户端的渲染重建等等。本文主要聚焦于全景视频的视口预测算法和底层物理资源调度两方面。

1.2.1 全景视频视口预测研究现状

目前全景视频的传输方案主要分为全部传输和自适应传输两种[4]。全部传输是类似于传统视频传输方案，即将全景视频完整的传输给用户，目前主流的流媒体平台（例如YouTube）都是采用这种传输方式。这种传输方案可以保证用户看到全部的视频内容但是对网络负担极大，且会造成较多的带宽浪费。自适应传输方案则是针对视口区域进行差别传输，其中基于区块（tile）的传输方式被认为是传输全景视频的理想方案[5,6]。不同于传输整个画面，基于tile的方式以较高质量传输视口区域所在的tile，而以低质量传输非视口区域的tile甚至选择不传输。这种传输方式可以很好的降低带宽需求，但很依赖准确的视口预测算法，一旦视口区域估计错误，将会导致用户观看区域内出现明显的低质量画面甚至是黑块区域从而严重降低用户体验，因此，如何准确有效地预测出用户的视口区域对全景视频的自适应传输有着极为重要的作用。

对视口进行预测，首先要明确的问题是视口是否可预测。文献[7]对视口中心的位置，即视点（viewpoint）数据进行了实验分析。文献主要研究表征视点位置的三个参数：偏航角（yaw）、俯仰角（pitch）和翻滚角（roll）的相关特性。首先分析了三个角度的累积分布图，实验结果表明偏航角的变化幅度最大，最难预测，其次是俯仰角；然后又分析了三个角度和各自差值的自相关函数，结果显示在500ms内有较强的自相关性，这表明视点数据是可以预测的。

目前全景视频视口预测的方法主要有基于运动轨迹和基于内容显著性检测两种[37]。。

基于轨迹预测主要是根据该用户历史一段时间内的视口位置来预测接下来的视口位置，视口位置主要通过头部移动轨迹或者眼球移动轨迹来确定，而这些轨迹数据可以通过HMD的定位功能采集到。

由于具有时间相关性的运动轨迹从本质上可以看作是时间序列，，因此很多时间预测的方法被应用进来，如滑动平均、线性回归和机器学习等。文献[8]对比了滑动平均、线性回归和加权线性回归三种预测算法分别预测yaw、pitch和roll三个角度的准确度。滑动平均复杂度最低，即使用历史窗口中所有视点数据的采样点的平均值作为下一时刻的预测值，当数据变化幅度较小时这种预测方法也可以获得不错的预测准确度。线性回归则是应用统计学中的回归模型来对历史一定时间内的用户视点数据进行拟合，然后通过拟合出来的函数模型获取下一时刻的预测值，加权线性回归则是给予历史数据不同的权重值，如越近的数据拥有越高的权重值，以此来获得不同的影响程度，一定程度上提升预测准确度。[8]的结果表明，线性回归与加权线性回归预测结果要好于滑动平均，当使用历史1s的视点数据预测接下来1s的视点数据时，加权线性回归的预测准确度高达90%以上，但是当预测未来长时间断的数据时准确度会显著降低，当预测接下来2s的视点时，预测准确度则不足70%。文献[7]则对比了朴素预测，即使用当前的角度值当作预测值，线性回归和神经网络预测未来0.1s~1s范围内的偏航角，实验结果证明神经网络的预测效果要好于前两个算法。

除了使用当前用户的视点数据外，一些研究还探索了跨用户的行为。文献[9]使用了结合KNN（K-Nearest Neighbor, K最邻近） 聚合的线性回归模型，基于当前用户的历史视点，使用线性回归模型预测出当前用户下一时刻的视点后，再结合观看同一视频的其他用户的历史数据，使用KNN来对预测结果进行纠正，进一步提升预测准确度。文献[10]对三个角度的轨迹进行建模，具有相似观看行为的轨迹被聚合在一起，并且为每个聚类计算出不同的轨迹函数。文献[11]则首次使用一种基于密度的聚类算法DBSCAN，在服务器端先对用户进行分类，然后在客户端，通过SVM（Support Vector Machine, 支持向量机）分类器来预测用户的类别，最后预测该类别所对应的观看轨迹。

另外一类是基于内容显著性检测的预测算法，对视频而言，视频显著性表征了用户对视频不同内容的关注程度，一般来说，视频内容的显著性越强则说明用户对该内容越感兴趣，内容被观看的可能性也越大。近年来，基于深度CNN（Convolutional Neural Networks, CNN）的显著性检测方法相比于传统方法可以取得较优的结果[12,13]。文献[14]训练了一个浅层网络进行显着性检测，并应用了转移学习生成一个深层网络；同样的作者在文献[15]中介绍了SalGAN，这是一种通过对抗进行训练的深度网络；但是，这些网络模型都是应用在传统二维图像或者视频中，直接应用到全景视频时可能会造成中心偏差或者多目标混淆的问题，造成错误的检测。

近些年来一些学者开始关注面向全景视频的显著性检测问题。由于用户观看全景视频是更倾向于观看赤道附近的内容，一些算法就将该特性应用到显著性检测中。文献[16]将全景视频的每帧图像分割成若干个区块后给每一部分区块设置不同的显著性权重。由于全景视频投影为二维视频的时候会存在边界失真的情况，文献[17]分别计算出几个投影平面的显著性图后计算平均值作为全景视频的显著性结果。文献[18]则是将全景图像映射为6个视口子图像输入到CNN网络中，输出子图像的显著性图，最后将6个显著性图映射到矩形平面后得到全景图像的显著性。

为了更准确地捕获未来视口与过去视口之间的非线性和长期依赖关系，文献[19]提出了针对固定视点的预测模型，该模型同时利用过去的视点位置和视频内容特征来预测接下来n帧中的视口轨迹或基于tile的观看概率。文献[20]提出了两种深度增强学习模型：首先使用一个仅基于视觉特征来估计视频中每帧热图的离线模型，然后使用一个在线模型，根据过去观察到的头部移动位置以及离线模型得到的热图来预测头部运动。文献[21]使用CNN开发了一个视口预测模型，模型删除了池化层，并添加了更多的卷积层以增强非线性拟合能力。

在处理长时间内数据关系时常被使用的LSTM（Long-Short Term Memory, 长短期记忆）模型也被逐渐应用在视口预测中。文献[22]使用LSTM编码视口区域扫描路径的历史记录，并将隐藏状态特征与视觉特征结合起来进行未来1秒的预测，结果表明基于LSTM模型的预测优于几种基准方案。

1.2.2 LTE资源调度算法研究现状

如果在现有的无线网络中通过有效的资源调度来满足多用户不同业务的服务质量需求也一直是学术人员的重点关注领域，本文主要聚焦于LTE系统中下行链路的资源调度问题。

资源调度领域的相关研究能够追溯到20世纪80年代左右，最初学者们提出的算法较为简单，忽略了无线信道的传输状况，其中最为典型的是RR算法（Round Robin, 轮询）[23,24]，该算法将资源轮流公平地分配给网络中的用户，而不考虑实时变化的信道条件，RR算法可以给与用户平等的调度机会，但带宽利用率低，导致系统吞吐量较小；与RR算法形成对比的Max C/I（Maximum Carrier to Interference, 最大载干比算法）[25,26]算法则是以系统吞吐量最大化为目标，只考虑用户信道条件的好坏，造成信道条件差的边缘用户一直得不到系统资源；结合RR和Max C/I算法的优点学者们提出了PF算法（Proportional Fairness, 比例公平）。但是上述这些算法都忽略了业务的QoS需求导致在实时业务的调度场景中表现较差。鉴于视频和语音通话这类实时业务需求的增加，研究人员们又提出了一些面向实时业务的调度算法，比如EDF算法（Earliest Deadline First, 最早截至优先）[27]，该算法根据每个业务的QoS需求中的最大时延对所有任务队列进行排序，每次优先调度截止时间最小的业务，保证了时延敏感业务的调度；MLWDF算法（Modified Largest Weighted Delay, 改进的最大权重时延优先）[28,29]在PF算法的基础上同时衡量了分组数据的时延阈值和信道条件两方面的信息，可以在系统吞吐量和实时业务丢包率之间取得较好的平衡。

由于经典算法性能的局限性大量改进的优良算法也被陆续提出。基于PF算法，文献[30]对非实时业务和实时业务进行了区分，对于实时业务，改进的算法考虑了其时延特性，保证了时延较为敏感的业务可以优先被分配资源得以调度；文献[31]分析了两种业务的调度算法后改进了实时业务的优先级计算方式。

除了从网络系统方面对算法进行优化外，研究人员也从用户侧出发如QoE（Quality of Experience, 用户体验）为优化目标改进调度算法。文献[32]根据平均意见得分指标评估了LTE网络中三种流行的调度算法（PF、EXP-PF和MLWDF）。实验结果表明，最适合的下行调度算法是MLWDF，其端到端时延小于50ms，并且在可以接受的平均意见得分（超过3.5分）下运行最大数量的用户接入（超过50个用户）。当用户超过30时，最为广泛使用的PF算法的端到端时延会增加到200ms以上，这表明PF算法并不适合调度实时业务；文献[33]则是面向针对VoIP业务的QoE指标优化了调度算法。

为了更好地结合多方面的因素来达到高效调度的目的，综合考虑多个协议栈参数的跨层调度也得到了进一步的研究。文献[34]，协同无线协议栈的应用层、MAC层和物理层参数提出了一种跨出优化方案，最大化利用网络资源并提高用户的体验水平。仿真结果显示提出的跨层框架可以最大程度的提高用户的感知质量并且在用户的公平性方面得到了显著的改进。文献[35]则在跨层优化的基础上，采取多点协作的方式进一步减小了实时业务的传输时延。

目前完全针对全景视频的资源调度研究十分有限。文献[36]以多路径协作路由的方案来促进5G小型小区网络中的全景视频的传输优化；文献[37]则设计了面向多用户的全景视频自适应传输框架，并通过部署移动边缘计算来增强基站的功能。

在最近的研究中，深度学习的相关方法也被应用在无线网络传输中。针对资源调度问题，文献[38,39]基于Q学习（Q-Learning）根据不同类别用户的CQI（Channel Quality Index, 信道质量指标）在每个传输时间间隔采用不同的调度规则。

1.3 主要研究内容

综上所述，全景视频的准确预测是传输优化问题的重要前提之一，然而，基于运动轨迹的预测没有考虑视频内容对用户观看行为的影响，在长期预测场景中存在较大问题；基于视频显著性检测的算法复杂度不利用对时延比较严格要求的全景视频，同时显著性检测也未能考虑观看者在年龄、性别、兴趣爱好等方面的区别，在内容提取方面也存在一些误差。因此，本文在研究了视口预测的常见算法后对基于运动轨迹的预测算法进行了改进优化。

随着全景视频等视频业务在无线网络中的增加，如何科学调度现有的带宽资源来满足多用户多业务的需求也是亟待解决的问题之一。考虑到现有的调度算法在整个调度期间只使用一个调度策略，不能很好地适应动态变化的网络情况，本文根据吞吐量、时延和公平性指标定义了若干种状态，然后基于Q-Learning算法来决定在每种状态下应该选择何种调度策略来达到系统的性能最优化。

1.4 本文组织结构

本文共分为四个章节，各章节安排如下：

第一章：绪论。该章节主要介绍了全景视频传输的研究背景和研究意义，针对传输框架中的视口预测和资源调度两方面，然后介绍了国内外研究现状和研究内容，最后给出本文的组织结构；

第二章：基于历史观看轨迹的视口预测算法。本章介绍了视口预测的常见方案并分析了各自的优劣之处，然后详细介绍了本章提出的基于历史观看轨迹的视口预测算法，最后阐述了视口预测算法的实验过程与详细的实验结果说明。

第三章：基于Q-Learning的LTE资源调度算法。本章首先分析了LTE网络的基础架构和关键技术，然后介绍了常见的资源调度算法与资源调度流程，在此基础上，本章提出一种基于Q-Learning算法的下线资源调度算法。该算法根据动态变化的网络状态，使用Q-Learning算法在每个传输间隔选择不同的调度算法，最后使得系统性能最优化。最后介绍了仿真平台与仿真环境，从系统吞吐量、丢包率和用户业务时延等方面对实验结果进行了分析与对比。

第四章：总结与展望。本章节对论文做了回顾总结与未来展望。

# 第二章 基于历史观看轨迹的视口预测算法

根据第一章的介绍，现阶段国内外视口预测主要分为两种方式：基于用户历史观看轨迹和基于视频内容。由于基于视频内容的视口预测算法时间复杂度较高，同时，不同年龄、性别的用户在观看视频时的感兴趣区域差异较大造成预测偏差，因此，本章只聚焦于基于历史观看轨迹的预测算法。首先介绍视口预测相关技术的理论基础，然后详细介绍基于当前用户视口数据的预测算法，并结合其他用户的视口数据辅助预测结果，最后，对比分析了其他算法与我们算法的预测准确度。

2.1 视口预测相关技术

视口预测作为全景视频传输优化问题的前提之一，已经出现了一些针对该问题的研究。为了方便后续算法的描述，本节首先介绍了视口的相关定义，然后概述了现阶段的视口预测算法。

2.1.1 相关定义

（1）视口区域

在给出视口区域的定义前，首先介绍人眼和VR眼镜（或HMD）的特性。

人类眼睛单眼在水平方向最大可以观看到150度左右的画面，双眼在水平方向则可高达188度左右。但是双眼重合视区为120度左右，即在这个角度范围内，观看到的物体才会呈现出立体感。超过水平视角30度之外的视角称为诱导视角，即我们常说的余光，在这些区域的事物人眼是不敏感的。而在垂直方向上，单眼感光区域约为120度，视觉敏感区为60度，如图2-1所示。

(a) 水平视角 (b) 垂直视角

图2-1 人眼视角区域

由于全景视频的特性，必须佩戴相关的观看设备才能观看，这些硬件设备在生产时都会有各自固定的视角区域。目前市面上入门级的VR眼镜的视角区域水平为90度，垂直区域为90度；像Facebook的Oculus和HTC Vive之类昂贵的头显设备的视角区域则为水平110度，垂直90度。

全景视频是一个横向呈360度，纵向成180度的三维视频，但由于人眼特性和全景视频观看设备的参数，在某个特定时刻，用户只能观看到全景视频的一部分内容，这部分就称为FOV（Field Of Viewport, 视口），如图2-2所示：



图2-2 视口区域

为了方便后续的实验说明，本文中所指的视口区域特定为水平110度，垂直90度的矩形区域。

（2）视点

视口区域是一个矩形区域，我们定义该矩形的中心点所在位置为视点。当视口大小固定时，只要知道视点位置即可获取到视口区域。我们给出可以描述视点的几种常用参数如下：

a）x、y、z坐标

我们将全景视频构建成笛卡儿坐标系（Cartesian Coordinates）中的一个单位圆，视点位置即为圆上一点，可以使用空间三维坐标x、y、z来表示。

b）经纬度

全景视频可以通过多种投影方式展开为二维平面视频，常见的投影格式有EPR（Equirectangular Projection, 等角投影）和CMP（Cube Map Projection, 立方体投影）等。我们使用国内外主流内容平台都使用的EPR方式将全景视频投影为二维平面视频，投影方式如图2-3所示。以地球仪展开为地图为例，ERP就是假设地球仪嵌入一个圆柱体内，然后地球仪的球心处放置一个灯泡，灯打开后地球仪上的画面照应到圆柱上，再把圆柱水平展开为二维平面，即为地图。这种投影方式在赤道部分的误差较小，越往两级误差越大。



图2-3 EPR投影示意图

经投影展开后，三维球面上的视点就落在了平面上，我们使用经纬度来表示。经纬度与三维坐标可以通过下面的公式互相转化：

其中，R代表三维球体的半径，为纬度，为经度。

c）偏航角、俯仰角和翻滚角

用户在佩戴VR设备观看全景视频时，头部从某个位置转到另外一个位置的旋转量可以使用三个欧拉角（Euler Angles）来表示，即偏航角（yaw）、俯仰角（pitch）和翻滚角（roll），如图2-4所示。俯仰角为头部水平左右摆动的角度，向右转为正方向，范围为-180度到180度；俯仰角为头部垂直上下摆动的角度，向上抬头为正方向，范围为-90度到90度；翻滚角则是视线看向正前方，头顶在视线的正交平面左右摆动，以向右摆为正方向，范围为-90度到90度。

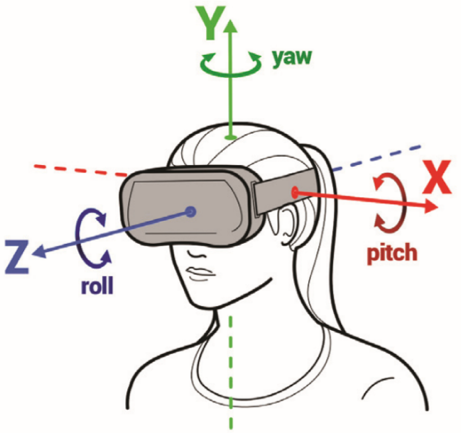


图2-4 方向角示意图

（3）视频区块

现阶段，全景视频的传输主要有两种方式，一种是传输全部内容，一种是传输部分内容。第二种传输方案中，需要从空间上将全景视频分割成不同的区域，称为区块（tile），如图2-5所示，我们将全景视频分成了16\*8个区块。假设视口区域为绿色区域，则基于tile的传输方式只需要传输视口区域所覆盖的6\*4块tile，如图中红色区域所示：



图2-5 tile示意图

2.1.2 常用预测模型

全景视频视口预测主要有两种方法，基于历史运动轨迹和基于视频显著性检测。考虑到后一种方法的时间复杂度不太适合实时预测，本文只探讨第一种预测算法。

基于历史运动轨迹的预测，通常是使用历史的视点数据来预测接下来一段时间的视点数据，本质上可以划分为时间序列预测问题。时间序列的预测是数据挖掘领域热门的研究方向，研究人员们已经提出了许多优秀的预测模型，主要可以分为基于统计分析、基于机器学习和基于神经网络，本节分别介绍这三种类别中的常见的预测模型。

（1）移动平均

MA（Moving Average, 移动平均）是一种简单的时间预测模型，主要思想是根据历史一段时间（即窗口）内序列的平均值来预测接下来某一时刻的序列值，然后移动窗口继续预测，计算公式如下：

其中，表示时刻的预测值，为历史个序列值。

当数据变化范围较小，整体呈现平稳状态时，该方法可以取得较好的预测准确度。在移动平均预测的基础上，又提出了加权移动平均法，即给历史窗口中的每个序列不同的权重值，计算公式如下：

其中，为第个序列的权重值。使用该方法进行预测时，权重值的选择尤为重要，最为简单的是经验法和试验法，通常情况下，最近的序列值最能影响未来预测值，所以权重值应该大一些。

（2）ARIMA

ARIMA（Auto-Regressive Integrated Moving Average , 整合移动平均自回归模型）是基于统计学的十分流行的一种预测模型。该模型有着完善的理论依据，简单高效。

ARIMA模型是在ARMA（Auto Regression and Moving Average, 移动平均自回归模型）的基础上进行d阶差分得到的。差分是将不平稳序列转为平稳序列的常用方法之一，所谓差分，就是使用后一个序列值减去前一个序列值后的结果，一阶差分即进行一次减法运算，二阶差分则在一阶差分的结果上再进行一次减法运算，d阶差分依此类推，计算公式如下：

ARMA模型则是由AR（Auto Regression, 自回归模型）和MA模型两部分组成，表达如下：

其中，是模型的输出，即t时刻的预测值，是AR模型的公式，是系数，代表时间序列值；AR模型主要是探索预测值与历史序列值之间的线性关系。而MA模型主要是探索预测值与预测误差的线性关系。由公式可以看出，ARIMA模型的关键是确定三个超算数的值，差分中的可以通过平稳性检验来确定，和一般是观察时间序列的字相关图和偏相关图[40]的截尾和拖尾来确定。

ARIMA模型简单高效，但仅适用于线性规律，同时关键的和的值需要人工判断，且参数一旦固定后就无法应对数据的变动，在实际的视口预测模型中，由于用户观看不同视频时的观看行为不同，需要对每一个视频都进行参数估计建立新的模型，工程量大效率低下，因此该模型并不适用于本文探讨的预测场景。

（3）线性回归

随着机器学习技术的发展，研究人员也开始使用一些机器学习的模型来进行预测。LR（Linear Regression, 线性回归）[41]是机器学习中较为基础的一种算法。线性回归主要是找到数据之间的规律函数，对于单变量而言，线性回归拟合出一条直线，方程可表示为：，对于多个变量的话，拟合出来的表达式可表示为：

其中，是常数项，为对应的偏回归系数。

基于机器学习的预测算法主要是模型的训练，一般是根据历史窗口的大小将数据集划分为多个子序列段，每个子序列段后的个值就是待预测的真实值，模型输出的即为对应的个预测值。在这种情况下，我们通常使用真实值与预测值的MSE（Mean Squared Error, 均方误差）作为训练时的损失函数，计算公式如下：

其中，为预测值个数，为系数，为真实值，为对应的预测值。

可以看出，回归问题最后就转变成了优化问题，即找到使损失函数最小的解，划分好训练集与测试集之后训练模型即可。

除了LR模型外，常见的机器学习模型还有SVM模型（Support Vector Machine, 支持向量机）[42]和RF模型（Random Forest, 随机森林）[43]。相比于基于统计学的预测方法，基于机器学习的方法可以处理非线性关系，同时训练出来的预测模型具有普适性。

（4）MLP

由于深度学习的崛起，一些神经网络的模型也被应用到时间序列的预测中来。与机器学习算法类似，我们也可以通过训练神经网络模型来获取到历史序列值与预测值之间的规律关系。

MLP（Multilayer Perception, 多层感知机）[44]是最基本的人工神经网络模型，除了输入层和输出层外MLP还包括若干个隐藏层，而最简单的MLP模型只有一层隐藏层，不同层之间使用全连接，如图2-6（a）所示。

(a) 简单MLP模型示意图 (b) 隐藏层示意图

图2-6 MLP示意图

如图2-6（b）所示，假设输入层有个输入，记为，则隐藏层的输出可表示为，其中为权重，代表着两个神经元之间的连接程度，权重越大，可能性越大；是偏执，模型中重要的参数，偏置的设置是为了正确分类样本；是激活函数，如果不使用激活函数，每一层的输出都和输入呈线性关系，激活函数在神经网络中引入非线性关系，可以将神经元的输出映射到有限范围内，如-1~1或者0~1。常见的激活函数有Sigmoid函数、Tanh函数和ReLU函数等，根据实际环境来选择适合的激活函数。

MLP被应用到了许多预测场景中。文献[45,46]对比了ARIMA模型和MLP模型在时间序列预测方面的性能差异，实验结果表明，短时间内MLP可以获得更高的预测准确度。文献[47]使用MLP预测了河流流量，文献[48]则使用MLP预测了金融领域的数据。

（5）RNN

RNN（Recurrent Neural Network, 循环神经网络）[49]是以序列数据为输入，建模学习序列关系的深度神经网络。RNN在一般神经网络的基础上，在不同层的神经元之间也建立了互联关系，结构示意图如下：



图2-7 RNN结构图

从图中可以看出，在计算当前的最优解时，前一时刻的输出信息通过互联传递到了当前时刻，与当时时刻的序列值共同影响输出结果，A为神经元，假设激活函数为tanh，则当前的输出，其中为权重值，为转换参数，因此，历史序列的部分信息可以被保存，即“记忆”，这是RNN可以处理序列之间依赖关系的关键所在。

比起统计学模型和机器学习模型，神经网络在处理非线性关系的序列数据问题中有着更为突出的处理能力。文献[50]证明一般的神经网络可以拟合任意一个非线性函数。虽然RNN有着强大的学习能力，但是在处理长期依赖（即时间序列中距离较长的数据）时存在梯度消失的问题。梯度消失是指在训练神经网络模型的过程中，梯度传递到某个隐藏层时已经减小到极小值甚至消失，造成权重无法更新模型无法训练的问题。

2.2 基于历史观看轨迹的视口预测算法

我们将基于历史观看轨迹的视口预测问题归类为时间序列预测问题，根据上述介绍可以看出，针对视口数据这种呈现非线性关系的时间序列进行预测，使用神经网络更能取得较好的预测准确度。目前的研究文献中主要提出了基于线性回归和一般神经网络模型的预测算法预测准确度随着预测时间的增长下降明显，无法做到长期预测。

为解决上述问题，本文提出了改进的视口预测算法，首先使用LSTM模型分别预测当前用户视口中心位置的经度和纬度坐标，然后结合其他用户观看同一个视频时的观看轨迹，计算出最终视口区域所落在的tile位置。

2.2.1 数据集介绍

视点数据集的合理选择是预测算法成功的重要前提。由于视频内容、用户特性对用户的观看轨迹影响很大，因此合理的数据集应该包含各种类型的全景视频，观看视频的用户也应该分散到各个群体。

我们分析了现有公开的几个视点数据集[51,52,53]，最终使用wu等人采集的全景视频用户头部移动数据集[53]。该数据集为48个用户观看18个全景视频的头部移动数据，48名用户包括24名女性和24名男性，覆盖了各个年龄范围，18个全景视频则包含了演出、体育、电影、脱口秀和纪录片5种最流行的类别。

18个全景视频的时长从2分44秒到10分55秒不等，帧率为30fps，而头部移动采集设备每秒可以采集60~80个数据，且每两个采样点之间的时间间隔有轻微的随机性，为了后续数据分析的便捷性，本文使用线性插值法将原始数据下采样为每秒30个数据，与视频的帧率保持一致。最后我们将原始数据转换为经度和纬度值，并分别将经度的取值范围映射到0~360度，纬度的取值范围映射到0~90度。

2.2.2 LSTM介绍

由于后续的视口预测算法是基于LSTM模型的，本节先介绍一下LSTM的相关理论知识。

通过前面的介绍可知，RNN在对长距离的序列数据建模时会出现梯度小时的问题，为了应对这个问题，文献[54]中提出了LSTM（Long Short -Term Memory, 长短期记忆）模型。

LSTM是RNN的一种变体，比起传统RNN的神经元只有一层tanh层，如图2-8所示，其中，代表激活函数，代表激活函数，为点乘操作，为加法操作，隐藏状态为细胞状态（Cell State）。LSTM的神经元引入了门函数来控制信息流，以规避因为过多的记忆信息而造成梯度消失。



图2-8 LSTM结构示意图

LSTM的门函数有三种，即输入门，输出门和忘记门，其中忘记门是LSTM设计思想的精髓所在，三个门的主要功能如下：

（a）忘记门

忘记门在当前时刻使用函数对输入数据和上一时刻的输出信息进行取舍，函数的输出范围为0~1，当输出接近0时代表丢弃（忘记）该信息，接近1是代表使用（记忆）该信息，表达式如下：

其中和分别代表权重和偏置。

（b）输入门

输入门决定输入多少新数据，通过更新数据，将新数据加入当前状态，表达式如下：

中，表示上一时刻的细胞状态，与忘记门的输出点乘后加上新数据即为当前的细胞状态：

（c）输出门

输出门给出最后的输出值，表达式如下：

其中，为当前的输出状态，为最终的输出值。

基于LSTM对时间序列进行预测的过程与一般神经网络的过程一样，即划分训练集与测试集，对模型进行训练，训练结束后即可进行预测。

2.2.3 视口预测模型

由前面的分析可知，当视点位置确定时，视口区域也可以确定，因此，本节聚焦于视点的预测，根据历史的经纬度数据来预测未来一段时间内的经纬度数据。

假设和分别为t时刻的经度和维度值，和为用户初始的经纬度，和分别为到时刻之间的经纬度值，即和。在t时刻的时候，用户的头显设备记录下来从初始值到t时刻的全部的经纬度值（），视点预测即使用历史窗口中的若干经纬度值来预测接下来时刻的经纬度值，记为（）,其中为预测窗口，然后移动历史窗口继续预测，直至视频观看结束。文献[7]认为，视点的几个特征维度之间的自相关性大于互相关性，因此本文也采用独立预测经纬度的方式。考虑到经纬度序列的时间特性和非线性特性，我们引入LSTM模型进行长期时间的预测，历史窗口的数据即为模型的输入，而模型的输出则为我们想要的经纬度的预测值，模型结构图如下所示：



图2-9 基于LSTM的预测模型结构图

从图中可以看出，在t时刻可以根据历史的一段数据预测出下一个时刻的数据，表示为，其中为训练好的LSTM模型的输入输出函数；为了继续预测接下来的数据值，我们将预测值做为额外的输入值，组成新的历史数据，则，依次类推，直至预测出需要长度的数据值。

可以看出，随着预测时间的增加，预测误差将会逐渐增大，为了进一步提高预测准确率，我们参考基于内容显著性检测的视点预测方法，这种方法根据视频内容的显著性来辅助预测，由于视频各个区域内容的显著性不同，人们在某个时刻观看某个区域的概率也不相同，当该区域视频内容较为显著性时，用户观看的可能性也较大，所以，多个用户的观看轨迹可以从侧面反映视频内容的显著性。为了验证这一假设，我们从数据集中随机抽取了2个视频文件，绘制了48名用户观看这两个视频的轨迹散点图，如图2-10和图2-11所示，其中蓝色散点为用户的观看位置，红色散点为平均值。



(a) 多名用户的经度轨迹



(b) 多用户的纬度轨迹

图2-10 多用户观看视频1的轨迹图



(a) 多用户的经度轨迹



（b）多用户的纬度轨迹

图2-11 多用户观看视频2的轨迹图

从图2-10中可以看出，视频开始时，由于大部分用户第一次观看全景视频，对全景视频体验感到新奇，会随意转动头部以探索更多的画面，而当完全沉浸在视频内容中后（采样点大概在1000时），用户会密切关注视频的显著性内容，所以几乎所有的用户都保持着相同的观看轨迹，其他用户的观看行为对当前用户有着很强的参考价值；图2-11显示的用户观看轨迹相对图2-10的轨迹来说杂乱些，说明视频2可能存在多个感兴趣区域，但超过一半的用户也会倾向观看同一位置，这种情况下其他用户的观看轨迹对当前用户有着一定的参考价值。因此，我们认为不管哪种全景视频其他用户的观看轨迹都与当前用户的观看轨迹存在一定的相似性。

基于上述分析，使用LSTM模型初步预测出当前用户的视口数据后，我们在假设视频服务器存储了观看同一视频的其他用户的视口数据的前提下，结合其他用户的观看行为，对初步预测出来的数据进行调整，以提高预测准确度。

为了衡量其他用户历史数据的参考重要性，我们引入基于tile的传输机制，将视频划分为16\*8块tile，记第i行j列的tile的观看次数为，初始值设为0。首先，我们根据当前用户的历史数据分别预测出接下来的经纬度值，确定视口区域以及视口所覆盖的tile块，这些区域每块tile的观看次数加，然后根据其他用户在同一时刻的视口区域位置，将覆盖的tile的观看次数依次加，最后，每个tile的观看次数即为两者之和，计算公式如下：

其中，取值0或1，表示根据LSTM模型预测出来的区域是否覆盖该tile，n为视口区域覆盖该tile的其他用户的个数。和取值的比例关系代表了其他用户数据的参考性，例如，即我们认为最终预测的视口区域由LSTM模型预测出来的视口区域和其他用户的视口区域以相同的重要性共同确定。考虑到LSTM预测出来的视口数据的准确度会随着预测时间的增加而降低，的取值也应该随着的变化而变化，为了减小后续计算的复杂度，我们定义和的计算公式如下：

其中，N为其他用户的总个数，为系数，可以根据实验结果对的取值进行调整以提高预测准确度，为向下取整函数。



图2-12 tile观看次数示意图

图2-12为一帧视频的观看次数示意图，假设视频帧划分为16\*8块tile，则视口区域占6\*4块tile，。我们假设共有3名用户观看该全景视频，绿色区域为使用LSTM初步预测出来的当前用户的视口位置，蓝色和橙色区域为其他2名用户同一时刻的的视口位置，按照公式2-14计算出所有tile的观看次数后，我们可以获得一个二维数组，然后计算出6\*4块tile的和最大区域即为最终预测的视口区域所占tile，如图2-12中红色边框所框出来的区域所示。

2.3 实验与结果分析

本节介绍视口预测算法的实验过程和结果分析，并且介绍衡量预测算法准确度的两个评价指标，最后对比基于LSTM模型的预测算法与其他基准算法在不同评价指标下的预测准确度。

2.3.1 实验过程

根据之前的介绍，我们对经纬度坐标值分别预测，两者的预测过程一样，以纬度预测为例，主要包含以下几步：

（1）数据处理

我们首先对原始的纬度序列进行归一化处理，归一化可以将数据分布映射到一个确定的区间上，提高模型的收敛速度。我们使用线性函数转换方法将纬度特征映射到0-1的区间，线性函数转化方法可以在python的sklearn包中调用preproccessing库中的MinMaxScaler类实现。

归一化处理后，我们需要将时间序列转化为监督学习数据。为了验证我们的算法在长时间预测方面的准确度，使用历史1s的30个纬度数据来预测接下来1~6s的纬度，即我们将历史窗口pw的数据做为监督学习中的输入x，将预测窗口hw的数据作为监督学习中的输出y，并通过移动窗口的方式构造出n组样本数据，然后将数据格式转化成LSTM模型期望的3维格式[n组，时间步长30，特征纬度1]，为了有足够的数据进行学习，我们将观看同一视频的所有用户的数据拼接起来作为我们最终的数据集，采用交叉验证的方式，取钱80%的数据作为训练集，后20%的数据作为测试集。

（2）生成LSTM模型

我们调用Keras库定义了由2层LSTM共128个神经元构成的网络结构，然后使用全连接层Dense用于输出。采用均方误差MSE作为损失函数，ADAM[55]作为优化函数，比起传统的优化算法SGD,ADAM有着更快的收敛速度和更低的错误率，relu作为激活函数。设定好损失函数和优化函数后，调用compile()编译模型，编译好模型后调用fit()来训练模型，该网络模型采用参数分享的反向传播算法进行训练。

在模型的训练过程中，因为训练数据量巨大，没办法将全部数据一次性输入模型进行计算，因此需要设定不同的周期（epoch）和批次（batch）。一个epoch是指全部数据训练并输出一次。当一个epoch的数据过多给计算机造成计算压力时，就需要分割数据，分割后的每块数据称为batch，然后将数据一块一块地传递给计算机进行计算。而在神经网络的训练过程中，指传递一次完整的数据集是完全不够的，我们需要将有限的数据集多次传递给神经网络，多次更新权重值，使得拟合曲线从欠拟合到过拟合。本节最终设置epoch为50，batch为30。

（3）预测数据

当模型训练完毕后，便可调用predict()输入测试数据进行预测了，预测出来的数据区间也在0~1之间，使用MinMaxScaler类再将数据转换到原始区间，即可得到预测的纬度值。我们将预测的纬度值跟同一时刻真实的纬度值对应存储为csv文件，用于后续准确度的计算。

通过上述几步分别预测出经度和纬度值后，可以计算出视口所在位置以及对应的tile块。然后我们将同一视口其他用户的经纬度数据也按照历史窗口和预测窗口的大小划分成若干组，与预测数据的时间对应，并计算出每个用户视口所在tile，按照公式 更新tile数组。最终我们获得一个16\*8的数组，以6\*4的窗口去遍历该数组，计算出窗口中所有tile值的总和，总和最大的窗口即最终预测出来的视口区域。如果存在多个最大值，我们认为这种情况是无法区分哪个最大值更准确，因此就以最后一个最大值作为输出。

2.3.1 评价指标

为了对比我们提出的算法与其他基准算法的预测准确度，本节使用两种评价指标，经纬度绝对差值和tile重叠比。

（1）经纬度绝对差值

为了衡量LSTM模型的预测准确度，我们使用同一时刻真实的经纬度与预测的经纬度的绝对差值来代表预测误差，由于全景视频的特性，到在经度上，-150度和150度直接取差值为300度，但实际两者只差了60度，因此，我们使用文献[]中定义的角度差值的计算方式，公式如下：

其中，代表角度和的差值，为取模运算，基于此，我们定义预测时刻经纬度的绝对差值如下：

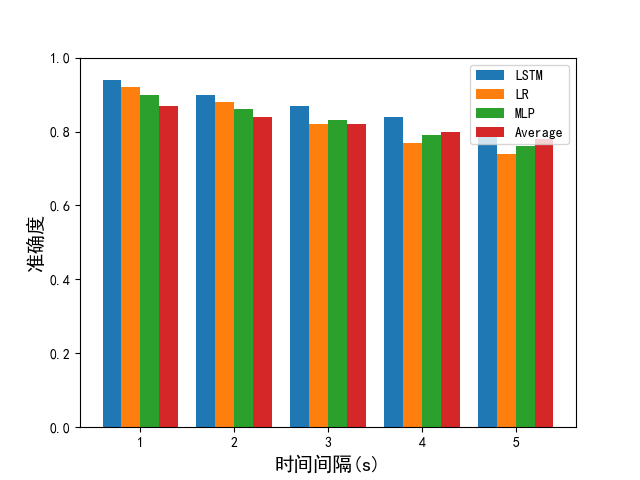
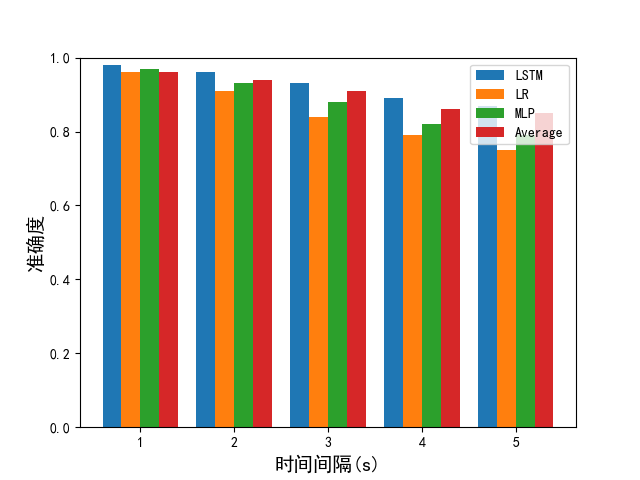
（2）tile重叠比

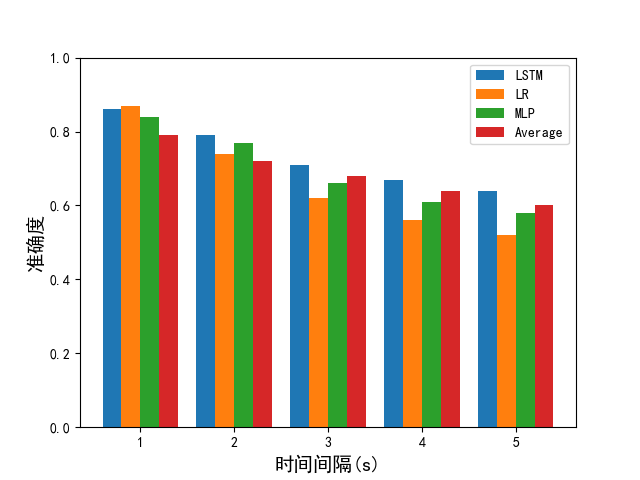
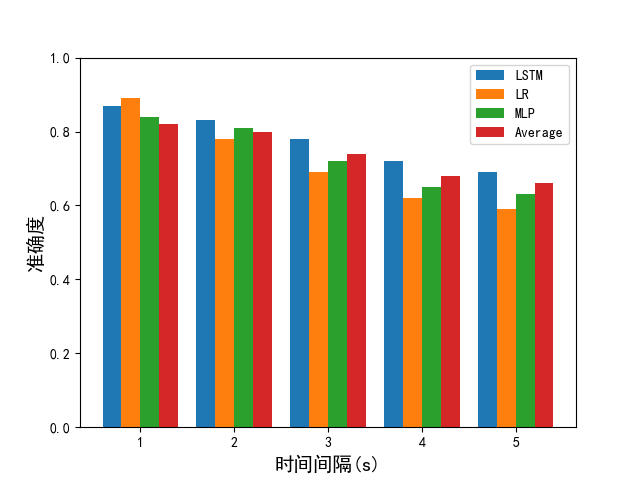
由于我们提出的预测算法最终预测的是视口区域所占tile的情况，所以我们使用同一时刻预测的视口所占tile在真实视口所在tile中的重叠比做为最终的预测准确度指标。

2.3.2 实验与结果分析

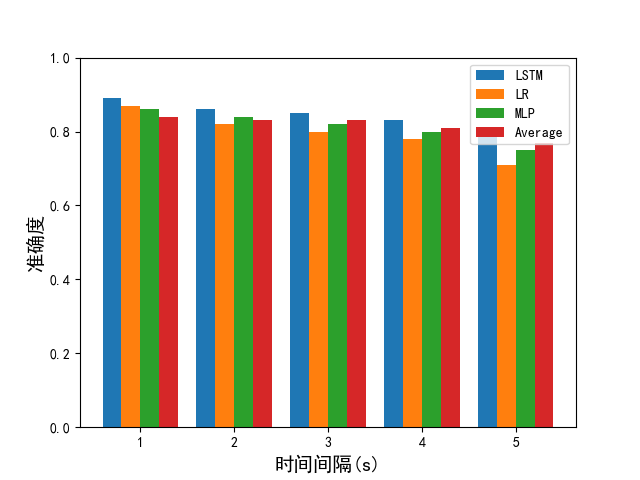
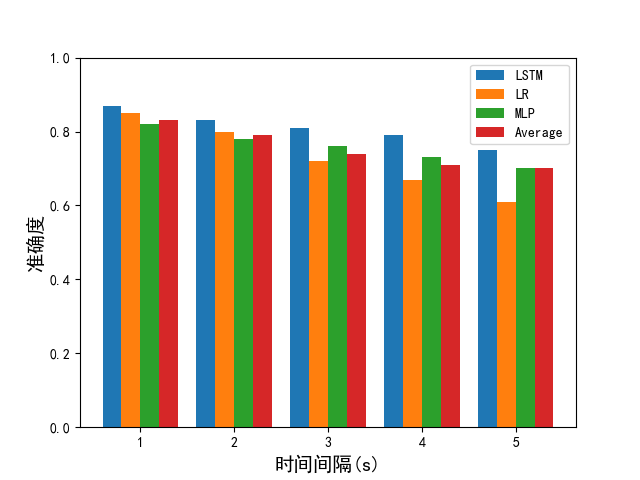
为了对比预测准确度，我们研究了两种回归方法，线性回归-LR和非线性回归-MLP，将时间和经纬度值视为独立变量，目标是根据历史窗口的数据拟合出回归函数，然后根据预测窗口得出预测数据，我们对经纬度分别建模，对于MLP，我们采用了含有64个神经元的模型结构，同样使用relu作为激活函数，ADME作为优化函数。最后，我们使用移动平均算法作为基准算法，即使用历史窗口中的数据的平均值作为整个预测窗口中的预测值。

首先，我们选取了数据集中三种代表性视频，对比了几种算法在预测经纬度方面的准确度，定义经度的绝对差值小于20度为预测准确，纬度也是，结果如图2-13~2-15所示。

(a) 纬度准确度 (b) 经度准确度

图2-13 女子篮球比赛预测经纬度准确度

(a) 纬度准确度 (b) 经度准确度

图2-14 自由滑雪预测经纬度准确度

(a) 纬度准确度 (b) 经度准确度

图2-15台表演预测经纬度准确度

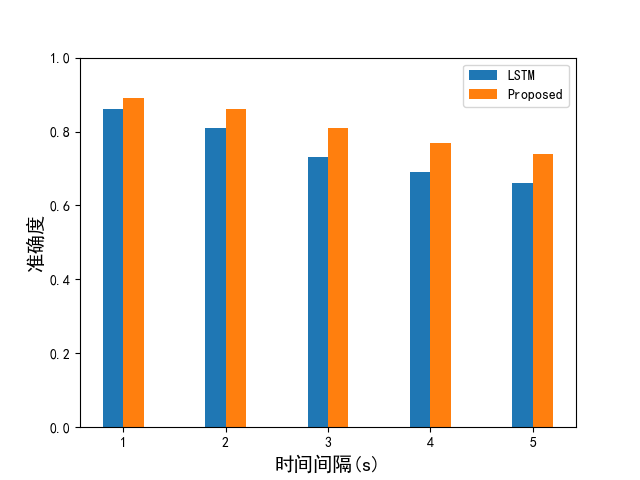
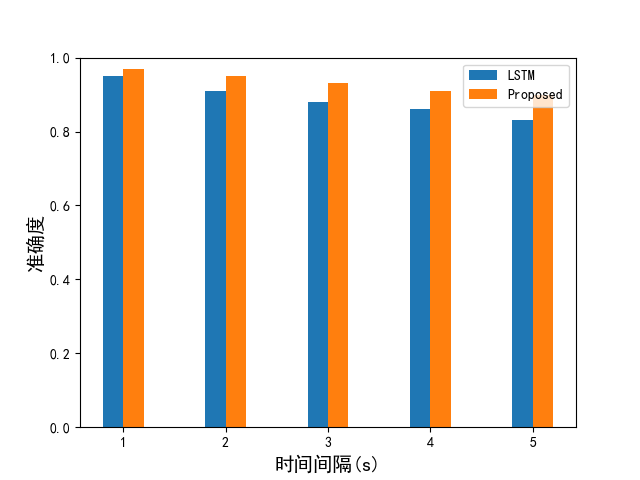
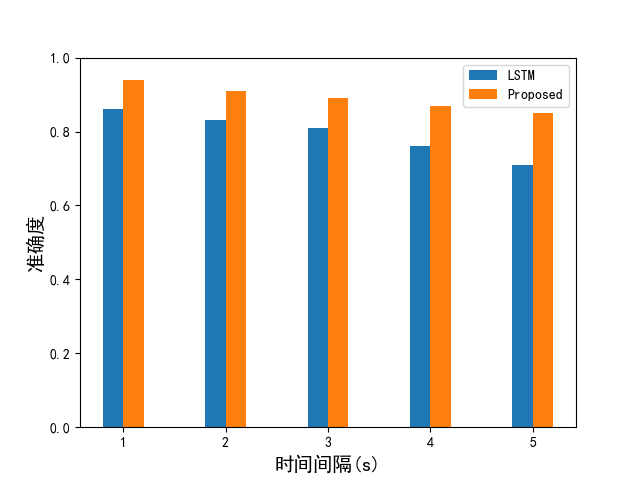
从上面的多张图中可以看出，随着预测时间的增加，所有算法的预测准确率都在降低，尤其是线性回归算法的预测准确度显著下降，这说明视口的长期预测是比较困难的，这主要是因为用户在观看全景视频内容的过程中，随着时间长度的增加，用户会随着内容的变换而自由地切换感兴趣区域，造成移动的随机性增强，从而降低预测的准确度。

对比了三个视频在经度和纬度方面的预测准确度后，区别于文献[56]认为的经度的预测难度高于纬度，我们认为经纬度的预测难度没有绝对的高低之分。我们分析，虽然通常情况下，人们更倾向于左右移动头部而不是上下移动，但当用户完全沉浸在全景视频中后，头部移动轨迹跟视频内容密切相关，同时由于眩晕感，在短时间内用户并不会迅速地转动头部，所以纬度的变化范围在很多情况下也会远远大于纬度的变化范围。

对比四种预测算法可以看出，无论是预测经度还是纬度，基于LSTM模型的预测算法的准确度在各个预测窗口下普遍优于回归算法，尤其在长时间预测时。LR在短时间（1s）内有较高的预测准确度，在某些情况下甚至高于LSTM的准确度，但是随着预测窗口的增加，LR的预测准确度显著下降，低于MLP甚至移动平均，这说明视口数据在短时间内呈现较强的线性关系，而在长时间内则呈现非线性关系。由于视口数据更多的是以赤道为中心上下分布[56]，所以长时间预测时移动平均算法会优于LR。

另外，虽然都是运动场景，图2-14中的预测准确度却要明显低于图2-13，这主要是由于，女子篮球比赛是直播场景，视频内容只有一个感兴趣区域位于视口位置，用户以较小幅度左右移动头部，在纬度范围上，用户头部随着篮球上下移动而呈现规律性移动，因此预测准确度较高。而自由滑雪的视频中有多个目标，目标移动幅度较大且分散，造成了预测的困难。舞台表演视频由于感兴趣区域集中，但目标移动的规律性小，所以预测准确度在前两者之间。

由实验结果图可以看出，LSTM算法的预测效果更加优越，但在某些场景下的预测仍差强人意，为了提高预测度，我们在LSTM预测的基础上引入其他用户的视口数据对结果进行调整，结果如图2-16所示。

(a) 女子篮球比赛预测tile准确度 (b) 自由滑雪预测tile准确度

(c) 舞台表演预测tile准确度

图2-16 结合其他用户预测tile准确度

从图2-16中可以看出，我们提出的结合其他用户的视口数据的预测算法在预测视口所在tile时要优于只基于当前用户的LSTM算法，从数值上看，当预测窗口为5s时，提出的算法的预测准确率分别为83%、64%和68%，比单用户LSTM算法的预测准确度分别提高了8%、10%和16%，平均提高了11.3%。

2.4 本章小结

全景视频的视口预测是全景视频传输优化问题的重要前提，优秀的预测算法可以显著降低全景视频的传输数据量。本节针对现阶段视口预测算法长时间预测准确度明显下降的问题，提出了一种基于历史观看轨迹的预测算法。首先，该算法根据当前用户的历史观看轨迹，使用LSTM模型对轨迹的经纬度分别预测，得到初步预测结构。为了进一步提高预测准确度，在视频服务器存储了观看同一视频的其他用户的轨迹数据的情况下，我们计算其他用户在全景帧每块tile的观看次数，再结合LSTM模型的输出结果，综合计算出预测窗口所覆盖的tile块。实验结果表明，基于当前用户和其他用户历史观看轨迹的预测算法比起其他常用的预测算法有着较高的准确度。

# 第三章 基于Q-Learning的LTE资源调度算法

本章聚焦于LTE网络中下线链路的资源调度问题，由于经典的调度算法在整个调度过程中只使用一种调度策略导致算法不能很好地适应动态变化的网络情况，本章基于Q-Learning来决定每个调度间隔选择何种调度策略使得网络性能最大化，并通过仿真实验对比了提出的算法与经典算法的在系统吞吐量、丢包率和用户公平性方面的性能优劣。

3.1 LTE架构与资源调度

本节主要介绍了LTE网络架构、协议架构和关键技术，然后介绍了无线资源的帧结构与资源块概念，最后介绍了资源调度过程。

3.1.1 LTE网络架构与协议架构

LTE是3G技术的演进版本，相较于3G网络LTE主要为了实现提高链路的峰值速率、减小系统时延和尽量满足小区中所有用户的QoS需求等几个目标[57]。

为了实现上述目标，LTE网络采用了一种扁平化的架构即只由EPC（Evolved Packet Core, 演进核心网）和E-UTRAN（Evolved Universal Terrestrial Radio Access Network , 演进通用地面无线接入网）两部分组成，如图3-1所示。这种架构一方面可以降低基础设备的数量，另一方面也可以降低系统的传输时延。



图3-1 LTE网络架构图

其中，EPC中三个网络节点的功能如下：

（1）PGW（Packet Data Network Gateway, 分组数据网关）主要是给LTE系统和其他非3GPP技术提供交互连接支持；

（2）SGW（Serving Gateway, 服务网关）主要给用户数据包提供路由和传输支持并保证QoS需求；

（3）MME（Mobility Management Entity, 移动管理实体）主要负责用户移动、用户跨小区时切换小区和对信令进行加密和控制等控制面功能。

与3G网络相比，LTE系统中的E-UTRAN仅由演进型基站（eNodeB）组成，eNodeB具有基站的全部功能和大部分无线网络控制器在接入层的功能。

接口定义了网络中不同单元之间进行数据沟通的方式，接口协议的架构被称为协议栈。LTE系统中的协议栈由不同功能的两大块组成：一块是控制平面协议栈，主要负责承载用户和系统之间的控制信令；一块是用户平面协议，主要处理用户的业务数据如对数据进行压缩、加密和调度等操作。

根据功能可以分为两大块：一是控制平面协议栈，

和用户平面两部分，其中控制面主要负责对无线接口的管理和控制，用户平面下图为LTE系统协议栈示意图：



(a) 用户平面 (b) 控制平面

图3-2 协议栈

用户平面各层主要功能如下：

（1）PHY（Physical Layer, 物理层）主要负责编解码、调制解调、多天线等处理以及为MAC层的数据传输提供信道支持；

（2）MAC（Media Access Control, 媒体接入控制层）主要负责处理逻辑信道和传输信道的对应关系，处理用户信息的测量和报告以及基站侧上下行链路的资源调度问题；

（3）RLC（Radio Link Control, 无线链路控制层）主要负责分段重组用户的业务数据、实现错误重传机制和进行重复检查等等；

（4）PDCP（Packet Data Convergence Protocol, 分组数据汇聚协议层）在发送端对流经的IP数据包进行头压缩以降低传输的数据量;对控制平面的信令进行加密和保护；同时在接收端可以进行解压缩和解密等逆操作。

而控制平面则在用户平面上增加了负责创建和销毁用户承载的RRC（Radio Resource Control, 无线资源控制层）和用于控制非接入层的NAS（Non-Access Stratum, 非接入层）。

3.1.2 LTE架构关键技术

LTE架构中主要的关键技术如下：

（1）OFDM

LTE有两项关键的物理层技术：OFDM（Orthogonal Frequency Division multiplexing, 正交频分复用）技术和MIMO（Multiple Input Multiple Output, 多输入多输出）。OFDM是一种应用广泛的多载波接入方案，核心思想是在频域内将无线信道划分为若干个正交的互不干扰的子信道，接收端接收到信息后，可以通过一定的技术手段将正交信号分开，这就类似于把高速传输的串行数据信号转换成低速传输的多个并行信号，OFDM的频谱示意图如下图所示：



(a) 传统的多载波技术



(b) OFDM技术

图3-3 OFDM频谱图

在以往频分复用的系统中，为了更好地消除子信道之间的干扰通常会在相邻的两个子信道之间增加足够的保护频带，而正交的技术允许没有保护频带，甚至相互覆盖仍不会造成干扰，比起传统的系统即降低了子信道之间的干扰性又极大提高了带宽利用率[58]；由于在每个子载波上传输信号的时间要比单载波信号传输的时间长很多，使得OFDM在对抗信道衰落方面有着较好的表现。

正交的子载波信号传输信息，比单载波信号的传输时间消耗更长，这使得在信道衰落和脉冲噪声的环境下 OFDM 可以更好地对抗衰落。另一方面OFDM使用了计算简单易于实现的快速傅立叶反变化和快速傅立叶变化来进行调制解调。

虽然OFDM有着很多优点，也仍存在一些缺点。一方面，由于OFDM的核心是依赖子载波之间严格的正交特性，而频偏或者相位噪声会造成不准确的正交，使得OFDM的性能大幅降低。另一方面，多个子载波信号合成的OFDM信号也会造成较高的峰均功率比。

　　（2）MIMO

区别于传统系统中发射端和接收端只有一根天线的形式MIMO则在两端同时采用了多个天线。理论上来说信道容量与天线数量呈线性正比关系。，这使得在带宽和发射功率都保持不变的条件下显著提高了LTE系统的吞吐量和信号的传送质量。图3-4展示了MIMO系统的框架结构：



图3-4 MIMO系统框架图

MIMO的关键技术主要包括：a)空间分集，由于多天线技术在实现方面存在着一定的复杂度，空间分集可以把复杂度放在发射端，然后在接收端只使用单根天线即可获得增益；b)空间复用，可以在多根天线上并行发射多个数据信号，提高了数据的峰值速率；c)波束赋形，指利用信号的相干特性将能量汇聚在某个或某几个指定方向上产生波束来降低信号之间的干扰性。。

（3）高阶调制技术

调制技术是把要传输的原始信号处理为适合在信道中传输的信号。理论上而言，调制方式越高阶，带宽的利用率也越高，高阶是指每个调制符号可以承载较多的信息，在信噪比比较好的情况下（比如离基站比较近的用户）可以通过提高调制阶数来提升信道的使用率。LTE中主要有QPSK、16QAM和64QAM三种调制方式，其中QPSK的理论速率为2bit/符号，16QAM的理论速率是QPSK的2倍，64QAM的理论速率则是QPSK的3倍。

3.1.3 无线资源调度

3.1.3.1 无线资源概述

无线资源分为时域和频域两个维度，在时域上表现为无线帧。LTE系统中定义了两种帧结构：TDD（Time Division Duplexing, 时分双工）帧结构和FDD（Frequency Division Duplexing, 频分双工）帧结构。

TDD帧结构如图3-5所示，每个TDD帧的长度固定为10ms并且可以等分为10个长度为1ms的子帧（每个子帧又可以等分为2个长度为0.5ms的时隙），其中有8个常规子帧和2个特殊帧。



图3-5 TDD帧结构

FDD帧结构如图3-6所示，每个FDD帧的长度也为10ms，不同于TDD，FDD的无线帧由10个常规子帧组成，上、下行传输时每个子帧可以在不同的频率上同时进行传输。



图3-6 FDD帧结构示意图

频域上无线资源则是由若干个资源块（RB）构成。我们将占一个时隙（0.5ms）、12个子载波（180kHz）的物理资源称作一个资源块， 如图3-7所示，RB是资源调度中可分配的最小单元。



图3-7 资源块结构图

系统带宽不同，物理资源块的数量也不同。在LTE系统中可以通过动态配置子载波数量以满足不同场景的需求。系统带宽、RB数量和子载波三者的关系如表3-1所示：

表3-1 LTE系统带宽配置

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 系统带宽（MHz） | 1.4 | 3 | 5 | 10 | 15 |
| 子载波数量 | 72 | 180 | 300 | 600 | 900 |
| RB数量 | 6 | 15 | 25 | 50 | 75 |

3.1.3.2资源调度流程

LTE系统中上下行链路的调度过程是分开的，本文重点探讨下行链路的调度过程。当小区内的多个用户请求不同业务时，基站处的缓冲池就保存着等待传输的不同应用的任务队列。总结来说，资源调度就是决策当前时刻给哪些待调度的任务队列分配哪些RB，主要分为两个流程，即计算用户优先级和分配资源块。

（1）计算用户优先级

优先级计算定义的是第个数据流在第个资源块上的调度优先顺序。不同的调度算法计算优先级的公式不同，以此来达到各自的分配目标。例如轮询算法，每个数据流的优先级相同，资源就被依次分配。

（2）分配资源块

计算出优先级顺序后则要进行资源块的分配。假设系统带宽有4个资源块，记为RB1-RB4，有3个待调度的用户，记为user1-user3，根据调度算法计算出优先级结果如表 所示：

表3-2 优先级矩阵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | RB1 | RB2 | RB3 | RB4 |
| user1 | 15 | 8 | 10 | 8 |
| user2 | 11 | 12 | 14 | 13 |
| user3 | 9 | 4 | 7 | 11 |

我们假设每个用户分配到两个资源块后即可达到传输需要。首先调度器开始遍历资源块RB1上的各个用户的优先级大小，发现user1的优先级最高，就将RB1分配给user1，然后在遍历RB2，发现user2的优先级最高，将RB2分配给user3，然后再遍历RB3，发现user2的优先级最高，将RB3分配给user2，最后遍历RB4，发现user2的优先级依旧最高，但是此时user2已经分配到两个资源块了，无需再分配，调度器就会比较剩下的用户user1和user3的优先级，发现user3的优先级较高，因此将RB4分配给user3。最后各个资源块的分配结果如下表所示：

表3-3 资源块分配结果

|  |  |
| --- | --- |
| user1 | RB1 |
| user2 | RB2 RB3 |
| user3 | RB4 |

调度器就按照上述步骤依次遍历用户和资源块，当所有的RB都被使用掉或者系统中所有等待的用户无需再分配资源时停止调度。

3.2 基于Q-Learning的资源调度算法

本节首先介绍了常见的四种资源调度算法的原理以及优缺点，在这些研究的基础上提出一种改进算法并详细阐述了改进算法的设计思想和系统框架。

3.2.1 经典的资源调度算法

在有限的带宽资源条件下如何有效地采取措施进行分配调度来尽可能满足小区内不同用户的需求至关重要。通常需要考虑用户公平性、系统吞吐量、系统丢包率和分组时延等用来衡量调度算法好坏的常用指标，接下来从这几个角度来介绍四个经典的调度算法，这些算法对后续改进算法的提出奠定了基础。

（1）PF算法

最初研究人员提出的轮询算法RR和最大载干比Max C/I算法都只考虑单个方面的性能最优，应用到实际网络中存在着一定问题，因此，研究人员提出了同时考虑吞吐量和公平性两方面参数的比例公平算法PF。该算法在定义用户优先级上同时考虑了用户的历史吞吐量和信道质量两个因素，优先级公式定义如下：

其中，表示第个用户在第个资源块上的优先级；表示将第个资源块分配给用户时的用户所获取到的即时传输速率（可以通过用户的CQI反馈计算出来），的值越大说明用户的信道质量条件越好；则表示用户的历史平均传输速率，在仿真中，该值的计算一般遵循以下公式：

从公式中可以清楚的看出，如果刚开始某个用户的信道条件比较良好时瞬时速率较高，因此，用户的优先级较大从而可以优先调度分配到资源，但随着资源的逐渐积累，该用户的历史平均速率也会升高，用户优先级随之下降，系统则会有可能将资源分配给小区边缘信道条件不好的用户。

PF算法在吞吐量和公平性方面较好的性能表现使得其成为了LTE下行链路中最常使用的调度算法。但是该算法没有考虑到各个业务QoS参数的不同，尤其是实时业务对时延的敏感性，所以需要更加适合实时业务的调度策略。

（2）EDF算法

为了解决实时业务的调度问题，最早截止先EDF算法被提出。该算法给予即将到达时延阈值被丢弃的业务最高优先级，给出最简单的计算公式如下：

为第个用户业务流的时延阈值，超过该阈值的数据包将会被丢弃，为用户的缓存队列首个数据包的等待时间。可以看出，越临近时延阈值的用户优先级越高。

（3）MLWDF算法

应对上述问题，改进的最大权重延迟优先算法MLWDF被提出。对于非实时业务，该算法计算优先级的公式等同于PF算法，而对于实时业务，该算法在PF算法计算优先级的公式的基础之上，引进了服务质量参数。优先级公式定义如下：

为队头时延超过时延阈值的最大概率，在仿真过程中一般设置为常数。

（4）EXP/PF算法

EXP/PF算法是对M-LWDF算法的一种改进，也同样适用于非实时业务和实时业务。该算法引入了缓冲时间来保证对实时业务对优先调度。对于实时业务，优先级公式定义如下：

其中的参数含义等同于公式 ，X计算公式如下：

其中，表示系统中等待调度的实时用户数。对于非实时业务，该算法的优先级公式与PF算法相同。

3.2.2 DSQL算法介绍

上述这些经典算法虽然被广泛使用，但在整个调度过程中始终使用单一的调度策略，这导致这些算法不能很好地适应不断变化的网络情况。我们希望可以根据不同的网络状态来选择最佳的调度策略以达到性能最优化。

近年来开始出现一些基于增强学习进行优化的研究。对调度而言，增强学习是一个很好的切入点，因其可以较好地学习动态变化的网络状态然后做出更高效的策略，Q-Learning算法就是这方面的翘楚，因此我们提出一种DSQL算法（Downlink Scheduler based on Q-Learning, 基于Q学习的下行调度算法）。

首先介绍Q学习的理论知识。

Q学习（Q-Learning）是增强学习领域最广为人知的算法之一。Q学习与模型状态无关，可以在环境未知的情况下，根据“状态-动作”获得环境的奖励，然后根据奖励不断调整动作使得奖励最大化，最后学习到最佳动作。可以看出，Q学习主要包含三个要素：（1）状态空间（2）动作空间（3）环境的奖励。

我们将实施上述过程的客体称为智能体（Agent），在每一次迭代中，Agent根据当前的状态从动作空间中选择一个动作，然后获得环境的即刻奖励，然后环境的状态转移到，继续迭代，记为多次迭代后的累计回报，使用一个的二维表来记录该值，称为Q表（Q-table），在每次迭代中根据公式 来不断更新Q表的值。

其中，为学习率，范围在0~1之间，代表之前训练结果的影响权重，学习率越大，之前训练结果的影响越小；则为折扣因子，范围在0~1之间，代表了未来奖励的重要性，如果=0，只有当前的奖励会被考虑。代表了下一个状态时Agent可以获得的最大的累计奖励。经过多次更新后，Q值收敛到最优值。

从上述分析可以看出，将Q学习应用到资源调度场景中，负责调度过程的基站就充当Agent的角色，动态变化的无线网络为环境的状态，动作则是不同的调度算法，我们希望在每一个TTI（Transmission Time Interval，传输时间间隔，LTE系统中设置为1ms）都可以选择一个最适合当前环境的调度算法，最后使得系统的累计性能最大，而常用的衡量系统性能的网络参数如系统吞吐量等即为奖励，基于Q学习的调度过程如下图所示：



图3-8 基于Q学习的调度过程

首先，基站从环境中获取到信息来判断当前所处状态，然后从定义好的若干个调度算法中选择一个，计算用户优先级完成资源分配后，环境状态从变化到，并且可以此次调度过程的即刻奖励，按照公式 更新Q表的值并开始下一次的迭代，直至调度完成。

根据上述分析，基于Q学习的调度算法重要的是需要定义状态空间、决策空间和奖励函数，接下来将详细介绍这几个关键点。

（1）状态空间

状态空间的划分是Agent可以学习到动作策略的基础，合理的状态应该满足以下两点要求：一是状态可以被Agent感知和处理，二是后一个状态只能依赖于前一个状态和选择的决策。一次资源调度后，系统中用户需要的数据量和等待时间都会发生变化，同时考虑到我们算法的目的主要是为了满足实时业务的QoS需求，例如系统吞吐量和数据包时延，定义状态空间为<*dataIndex, delayIndex*>，其中：

为等待调度的所有用户的数据量总和，为系统可传输的最大的数据量，表征了无线资源的短缺程度；为等待调度的用户数，表征了待调度应用流的时间紧迫度。当和都比较小时，说明更多的用户快要到了时间阈值且系统的资源较为充足，此时应该给时间紧迫的用户更高的优先级；而当和都比较大时，应该给信道质量较好的用户更高的优先级。

如果直接使用上述公式来定义状态的话，由于实际的网络情况比较随机，状态空间的数量将是巨大的。为了降低计算的复杂度提高收敛速度，我们首先统计了测试环境下和的数值范围和分布概率，然后将状态划分为16个区间，落在同一个区间的状态被赋予相同的值。

（2）动作空间

通过前面的介绍，显而易见的，决策空间应该被定义为不同的资源调度算法。为了保证QoS需求，计算方式需要综合考虑信道质量、用户公平性和实时业务的时延需求。因此，基于MLWDF算法，定义优先级的一般计算公式如下：

其中参数的相关含义与公式 相同。

考虑到系统吞吐量、用户公平性和业务时延的权重关系，我们选择其中特定4种情况构成动作空间，如下表所示。

表3-4 动作空间

|  |  |  |  |
| --- | --- | --- | --- |
| 动作 | θ | β | ω |
| A1 | 1 | 1 | 1 |
| A2 | 1 | 1 | 2 |
| A3 | 1 | 0 | 1 |
| A4 | 2 | 1 | 1 |

每个调度动作可以得到不同的RB矩阵，根据RB矩阵带宽资源依次分配给用户完成一次调度过程。

（3）即刻奖励

即刻奖励是用来评估前一个动作是否合适的重要指标，奖励函数的设计需要基于系统性能参数。我们希望在每个TTI选择某个调度策略后，系统的QoS指标（吞吐量、公平性和时延）可以得到提升，因此，奖励与这三个指标紧密相连，由于这三个指标的数量级不同，需要将取值映射到同一区间，下面依次给出三个指标的计算公式，首先是系统吞吐量指标STI：

其中，N为当前调度流程时系统中待调度的用户数，为时刻用户实际获得的吞吐量，则为当前调度流程可以调度的资源总和，可以看出UTI的取值范围在0~1之间。

我们使用AJFI（Average Jain’s Fairness Index, 平均Jain公平指标）[59]作为公平性指标，定义如下：

其中参数的含义与公式 相同。

我们定义系统时延指标SDI如下：

SDI的值越小，用户的等待时间越短。

由于学习目标是使得奖励最大化，所以即刻奖励应该跟吞吐量和公平性成正比，跟时延成反比，因此，我们定义即刻奖励的计算公式如下：

其中，η、λ和μ为权重值，可以通过设置不同的权重来达到所期望的权衡。因此，这个奖励函数期望系统吞吐量和公平性最大化，同时降低系统时延。

（4）探索与开发

除了上述三个要素外，Q学习算法还需要确定搜索策略。如果每次都是选择Q值最大的动作，这种搜索方式称为开发（exploit），开发模式会导致Q值小的动作一直不被执行，每次只选择当前阶段的最优动作而忽略了全局，探索（explore）则使得系统去尝试更多没有被选择过的策略。考虑到我们算法使用的仿真环境，我们使用一个简单的ε-贪婪算法，按照概率ε随机去选择动作，然后以1-ε的概率根据Q值来选择最优的策略。在仿真实验时，ε初始为1，即最开始完全随机的探索，然后逐渐降低到0.1后保持不变。

最后，我们给出DSQL算法的调度流程，如表 所示：

表3-5 DSQL算法的调度流程

|  |
| --- |
| 1：初始化Q表：，设置权重值γ和α |
| 2：for t=0 to T-1 do |
| 3： 根据公式 从状态空间中选择当前的状态 |
| 4： 根据ε贪婪算法选择一种调度策略 |
| 5： 进行包调度和资源分配 |
| 6： 根据公式 计算奖励 |
| 7： 根据公式 更新Q表 |
| 8：end for |

3.3 实验与结果分析

本节介绍DSQL算法的实验过程与结果分析。

3.3.1 仿真工具介绍

由于LTE系统涉及的技术繁杂，在真实的网络环境中进行实验论证存在着诸多困难，而使用仿真工具可以显著提高实验的可行性，节约实验之间。仿真方法主要分为两种：一种是系统级仿真，一种是链路级仿真。系统级仿真实现了整个网络架构，而链路级仿真主要针对物理层。因此，为了验证DSQL算法的性能，我们需要进行系统级的仿真实验。

目前开源的LTE系统级仿真工具主要由三个：维也纳大学开发的Vienna LTE-A Simmulators，基于Matlab；意大利都灵理工大学开发的LTE-Sim，基于C++，运行在Linux环境下；NS-3，考虑到仿真工具的复杂度和对C++开发语言的熟悉度，我们使用LTE-Sim[60]作为仿真工具完成实验。

LTE-Sim是一个开源的LTE系统级仿真平台，使用C++作为开发语言，因其具有封装与继承的特点，易于扩展开发，近些年来被研究资源调度的学者广泛使用。该平台以事件驱动，定义了90个类函数，共有220个文件大约2万多行代码。

LTE-Sim平台再现了完整的LTE网络结构，包括E-UUTRAN和EPS，支持

支持单/多小区场景、单/多用户环境、QoS管理、多小区切换、频分复用技术、AMC调制等多个场景和技术。LTE-Sim主要包含四个模块：Simulator、NetworkManager、FlowsManager和FrameManager，主要负责开始、处理和结束事件；定义并处理LTE帧和子帧；管理应用流；创建小区节点、用户设备并设置用户位置以及实现小区切换和频分复用。LTE-Sim还实现了三个网络节点：UE、eNodeB和MME/GW，这三个节点是由NetworkManager创建和处理的。

LTE-Sim定义了三种用户移动状态：静止、步行和乘坐交通工具，设置对应的移动速度为0、3、30和120km/h，定义了两种移动模式：一种是用户随机选择一个方向一直走到小区边缘，然后再随机选择一个方向走动；另一种是用户选择一个方向走指定距离后再切换方向走指定距离。我们选择了前一种移动模式进行仿真。

LTE-Sim仿真了四种最常见的应用类：（1）Trace-based，即视频流，主要是根据真实的视频文件获得到的数据包；（2）VoIP，语音流，根据G.729标准构建的on/off马尔科夫链模型；（3）CBR，有固定比特率的数据包，可指定数据包的到达时间；（4）Infinite-buffer，模拟的持续发送的数据包，一般用户模拟背景业务。

针对资源调度算法的仿真问题，LTE-Sim基本实现了资源调用与分配的整个过程，DownlinkPacketScheduler类包含的函数实现了从设定用户流承载到计算优先级再到分配资源的全部功能，其中，与资源调度算法有着最直接关系的是ComputeSchedulingMetric()函数，在该函数里可以编写不同的计算用户优先级的公式，默认实现了PF、MLWDF和EXP/PF算法。由于CQI反馈是用户与基站之间重要的交互过程，LTE-Sim也实现了该功能。

在LTE-Sim中对资源调度过程进行仿真的主要步骤如下：

（1）初始化仿真器、网络管理器、帧管理器和流管理器

（2）创建小区、基站、网关和信道，创建用户并设置位置和移动模式

（3）为用户指定承载的应用流，定义QoS参数和持续时间

（4）指定资源调度算法，分配无线资源

（5）指定仿真运行时间，开始仿真

仿真开始后，用户开始请求应用，基站侧的调度器根据数据包的QoS参数和用户发过来的CQI反馈等信息选择调制编码方式，然后根据资源调度算法计算用户优先级，最后分配无线资源块后完成一次调度过程。在整个仿真运行时间内，基站在每一个TTI执行一次上述过程。

3.3.2 仿真环境与参数

我们模拟了单个小区多用户的环境。小区半径为1km，小区中央有一个基站，用户数量从10递增到40，随机分布在不同位置，同时用户在小区内以3km/h的移动速度选择随机的方向运动。为了防止仿真随机性对结果造成的影响，我们进行多次实验取平均值作为最终的实验结果，仿真环境具体参数见表3-6。

表3-6 仿真环境参数设置

|  |  |
| --- | --- |
| 参数 | 参数值 |
| 带宽 | 10MHz |
| 小区半径 | 1Km |
| 频率 | 2GHz |
| 衰落模型 | Urban |
| 复用模式 | FDD |
| 帧长 | 1ms |
| 仿真时长 | 66s |
| 业务流时长 | 55s |
| 用户数 | 10-40 |
| 用户速度 | 3km/h |
| 应用 | Video, VoIP |
| 视频时延 | 20-100ms |
| VoIP时延 | 100ms |
| 学习速率 | 0.05 |
| 折扣因子 | 0.01 |

3.3.3 仿真结果与分析

首先介绍衡量调度算法性能的几种指标。

（1）系统吞吐量

系统吞吐量表征了单位时间内系统传输的数据量总和，在其他条件相同的情况下，系统吞吐量越大表明无线资源的利用率越高。本文实用如下公式来计算系统吞吐量大小：

其中，表示仿真时间内传输间隔的数量，为等待调度的用户总数，为用户在时刻传输的数据量。

（2）用户公平性

追求系统吞吐量最大化可以充分利用带宽资源，但是会导致一些边缘用户一直分配不到资源，为了保证同一小区内的用户得到自由的机会平等，还需要定义用户公平性。公平性的评定方式有多种，本文采用使用AJFI作为公平性指标,计算公式如 所示。可以看出，系统吞吐量跟用户公平性是互相矛盾的。

（3）丢包率

丢包率是丢失数据在总传输数据中的占比，丢包率小可以很好保证传输数据的完整度。丢包率主要取决于数据包的长度和排队时间。丢包率表示如下：

其中，为用户i在时刻t丢弃的数据包大小，则为用户i在t时刻成功到达基站缓存队列的数据包大小。

（4）系统时延

系统时延主要是衡量实时业务的参数，即数据端到端传输所消耗的时间，实时业务的等待时间超过时延阈值时会被丢弃。定义系统的平均时延如下：

为了强调我们对视频业务时延性能的考虑，首先模拟了小区内有30个用户且每个用户都请求视频业务的场景，并给每个视频业务设置不同的时延阈值，我们选择经典的实时业务资源调度算法EXP、MLWDF和EXP/PF作为对比算法，仿真结果如图3-9~3-11所示。

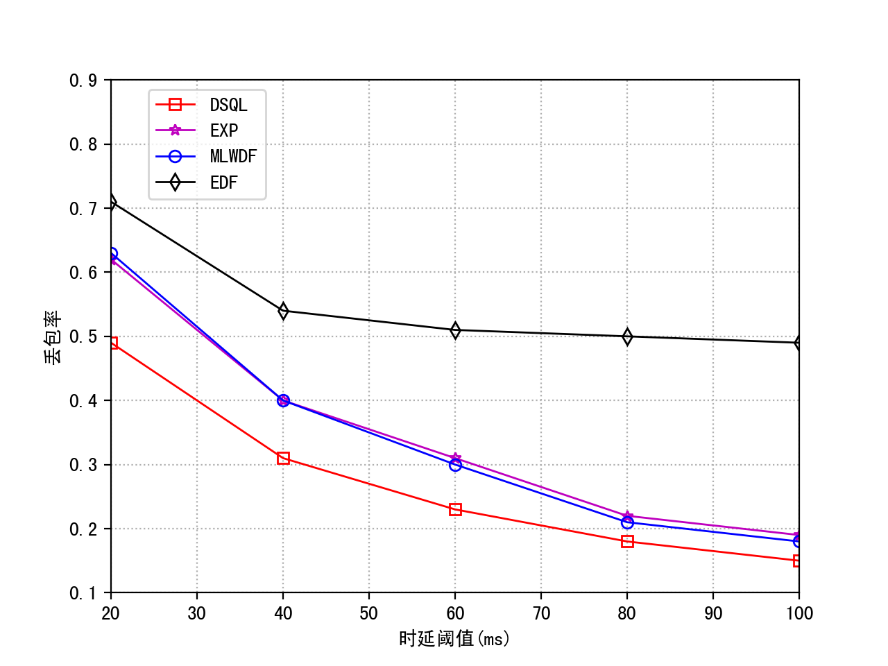


图3-9 不同时延阈值时系统丢包率

通常来说，丢包率是评估实时业务调度算法的重要指标之一。如图3-9所示， 显而易见的，所有算法的丢包率都与时延阈值呈反比。EDF算法只根据业务的时间紧迫度计算优先级，理论上会降低因超过时延阈值而被丢弃的数据包，但在实际调度过程中，可能由优先级最高的用户的信道质量很差，导致一次调度过程中用户只能传输少部分数据，只能继续等待下次调度，资源利用率较大导致后面资源急缺，造成大量用户得不到调度被丢弃。MLWDF和EXP均考虑了时延和信道质量参数，且计算优先级公式类似，因此两者的丢包率相似。DSQL算法可以自适应的选择最优的调度策略，因此丢包率最小，尤其在时延阈值效于40ms时显著优于其他三种算法。

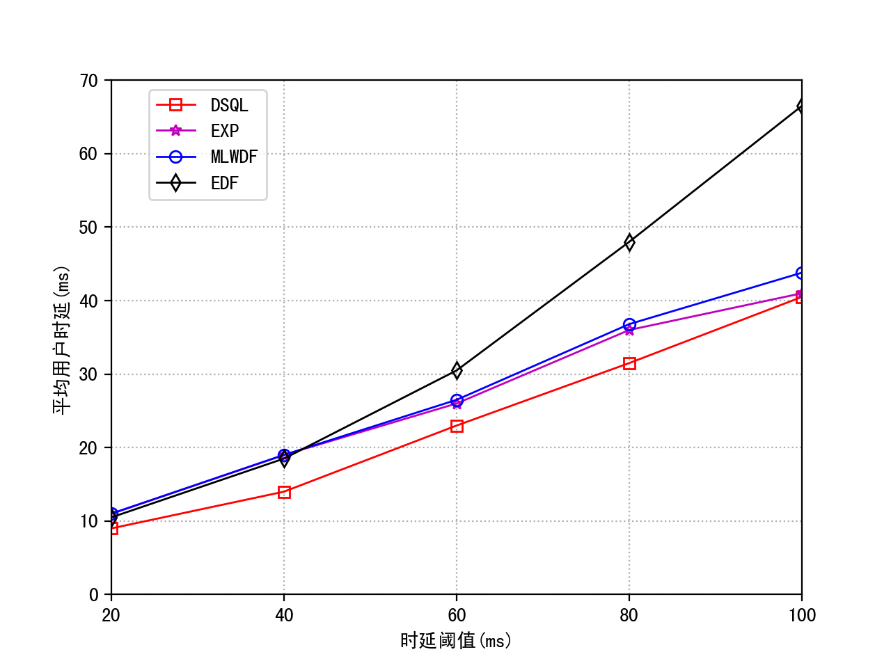


图3-10 不同时延阈值下平均用户时延

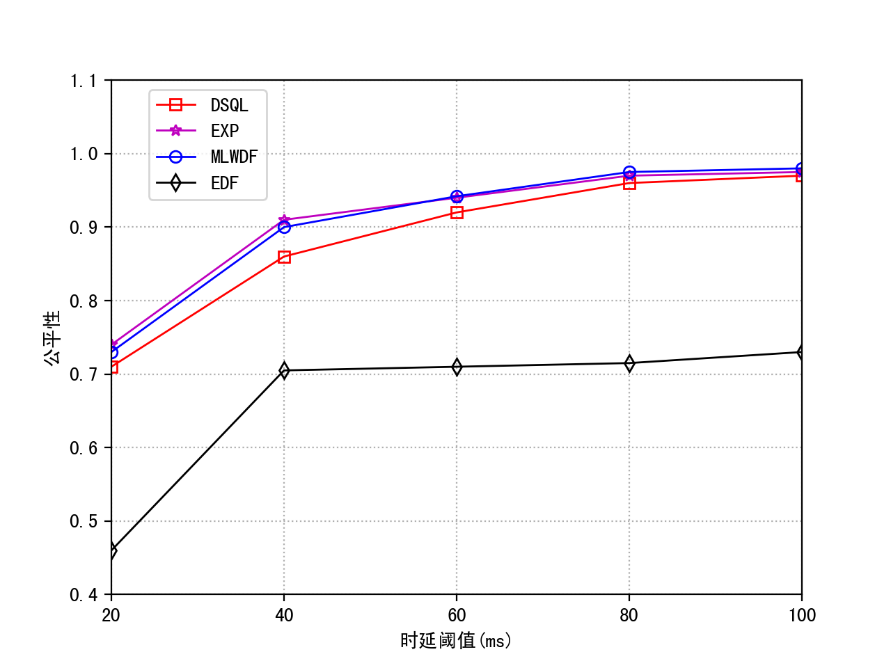


图3-11 不同时延阈值下用户公平性

图3-10展示了不同时延阈值下的平均用户时延。在时延阈值小于40ms的时候，可以看出EDF、EXP和MLWDF的平均用户时延基本一致，EDF略低于后两者，但这并不是说明EDF算法在时延方面优于后两者，结合之前的丢包率可以分析出，由于EDF的丢包率较大，许多时延较大的数据被丢弃了，导致参与计算的都是时延较小的用户。由于EXP算法中优先级与时延呈指数关系，对时延的反应更加灵敏，所以它在时延方面的性能要略优于MLWDF算法。与丢包率类似，在用户平均时延上DSQL算法仍保持着最优的性能，这说明变化的网络状态下灵活选择调度算法要优于只使用一种特定的算法。

图3-11展示了各算法在用户公平性方面的性能。理所当然，EDF算法有着较差的公平性，这也说明，通常情况下只考虑一个方面的调度算法的性能比较差。

由于我们在设计决策空间和奖励函数的时候，重点考虑了系统吞吐量和时延参数，所以在公平性方面DSQL算法略差于EXP和MLWDF算法，而要做到可以在这几个指标上都得到优化是十分困难的。所以考虑到DSQL算法在系统丢包率和用户时延方面取得的优异表现，用户公平性方面的轻微牺牲是可以接受的。

为了进一步评估我们算法的性能，我们改变了仿真环境，用户数范围从10个以5个的间隔递增到40个，并且每个用户都请求视频业务和语音业务，两种业务的时延阈值分别设置为40ms和100ms。图3-12~3-13显示了混合实时业务下的算法性能。

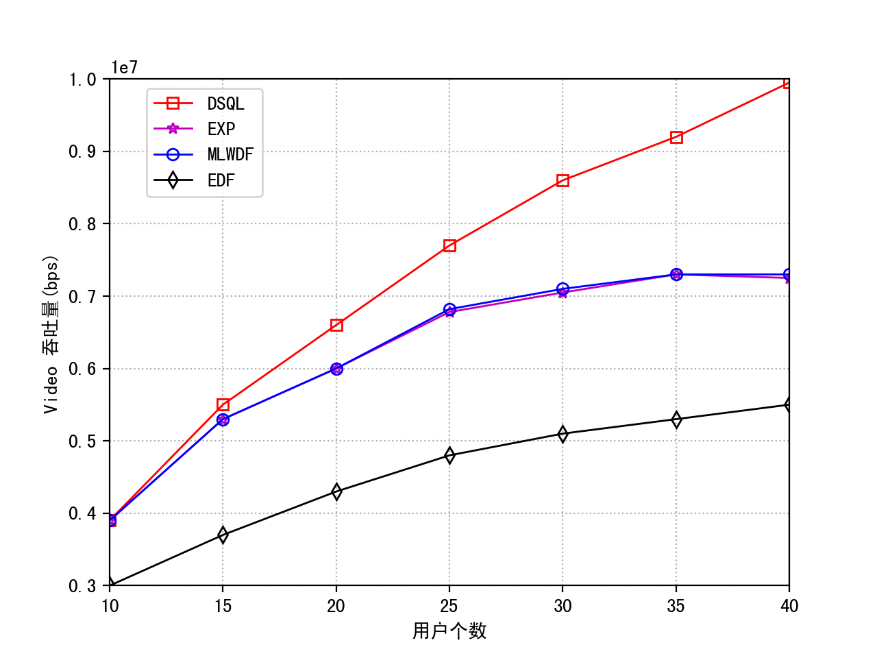


图3-12 不同用户时video吞吐量

从图3-12可以看出，当系统中有10个用户时，DSQL、EXP和MLWDF的吞吐量一样，这是因为此时用户数少，系统资源充足，所有用户都分配到了资源，系统吞吐量即为所有用户所需的数据量总和。随着用户数的增加，系统吞吐量逐渐上升，但随之丢包率也开始上升，所以当用户数达到35时，EXP和MLWDF算法开始出现轻微下降的趋势，而DSQL算法仍在持续上升。EDF依旧表现最差。

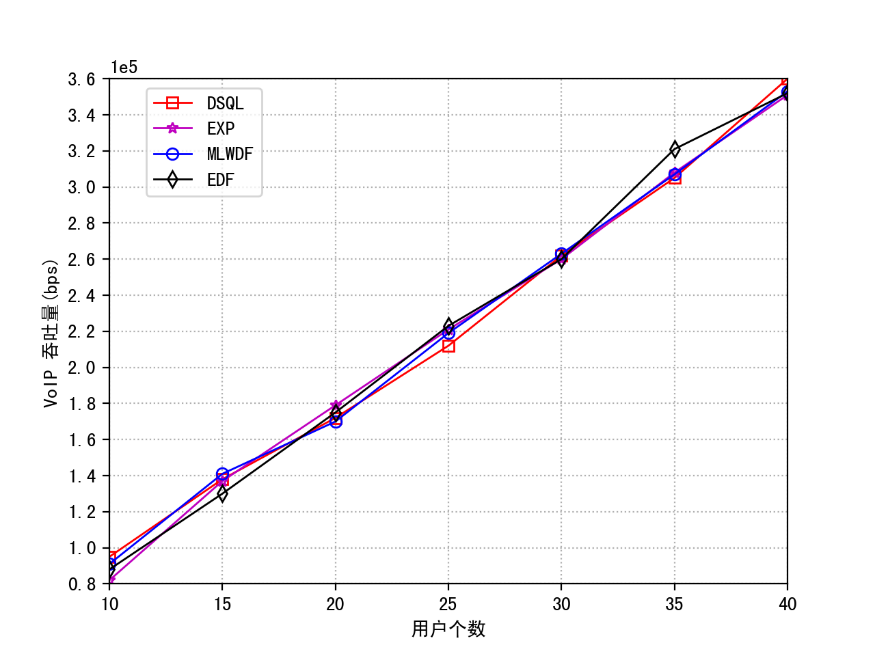


图3-13 不同用户时VoIP吞吐量

图3-13则为混合业务中VoIP的吞吐量，可以看出几种算法有着相似的性能。这表明，相比于其他算法，DSQL可以在显著提高视频业务性能的同时维持着VoIP业务的性能。

3.4 本章小结

本章首先介绍了LTE架构和几种关键技术；然后介绍了TDD和FDD两种无线帧结构，引出LTE系统中资源块的定义，描述了资源调度和资源分配的过程，并着重介绍了经典的资源调度算法。

本章针对视频这类实时业务，结合增强学习中的Q学习算法，提出一种基于M-LWDF改进的下行资源调度算法DSQL，该算法可以根据变化的网络状态和用户状态，动态地选择不同的调度策略，在系统吞吐量、系统时延和用户公平性之间取得平衡使得综合性能最大化。然后，本章介绍了使用的仿真平台LTE-Sim，描述了常用的仿真功能函数和仿真过程。最后，通过仿真实验从系统吞吐量、丢包率、时延和用户公平性几个方面对比了提出的算法与经典算法的性能优劣，最终验证了DSQL算法的性能优越性。

# 第四章 总结与展望

4.1 工作总结

随着计算机科学和多媒体技术的发展，全景视频作为一种全新的信息媒介以其沉浸式构想性的特点给人们带来了身临其境的观感，并逐渐应用到各个领域中。然而，全景视频数据量大、对时延要求严格的特性也给现有的传输网络带来了一些挑战，如何优化全景视频的传输一直是受人关注的研究领域。因此，本文从降低传输数据量和高效调度无线资源两个方面进行改进，主要工作研究与成果如下：

（1）为了降低全景视频的传输量，本文首先研究了视口预测算法。考虑到基于内容分析的预测算法的复杂性，本文只聚焦于基于历史轨迹的预测算法，将视口预测问题划分为时间序列预测，首先介绍了几种常见的预测算法，基于视口的二维特性，即经纬度，初步提出了基于序列到序列模型的预测算法，在数据集上初步验证了该算法的预测准确度高于其他一些算法，然后，为了进一步提高预测准确度，本文分析了观看同一个视频的其他用户的视口轨迹，发现轨迹之间的相似度极高，基于此前提，本文结合其他用户的视口数据对初步预测的结果进行矫正，最终，实验结果证明联合其他用户的预测算法在长时间预测时可以有效提高预测准确度。

（2）减少全景视频的传输量之后，本文又考虑了LTE网络中基站侧的无线资源调度。首先介绍了LTE的相关理论，包括系统架构和关键技术。然后介绍了无线资源块的概念，给出了资源调度和分配流程。针对全景视频这种对时延极为敏感的业务，本文研究了经典的调度算法的优缺点后，基于强化学习中的Q-Learning算法，提出了一种创新的下线资源调度算法DSQL，该算法根据动态变化的网络状态和用户状态，使用Q-Learning算法在每个传输间隔选择不同的调度策略，最终达到系统性能最大化目的。为了验证DSQL算法的性能优越性，本文介绍了仿真平台LTE-Sim的相关功能和仿真流程，通过仿真实验，从系统吞吐量、系统丢包率、时延和用户公平性几个方面综合对比了DSQL和经典算法的性能，最终验证了DSQL算法的优越性，尤其在时延敏感的实时业务调度方面。

4.2 未来展望

全景视频传输优化是一个庞大的研究领域，设计到计算机视觉、数据挖掘、无线网络传输优化等多个研究学科，本文只在现有研究基础上做了部分改进，仍有一些问题需要未来做进一步深入研究：

（1）本文只聚焦于基于历史轨迹的视口预测算法，没有充分考虑到影响视口的多个因素，如视频内容的多样性和用户的观看兴趣与习惯，没有建立充分完善的视口模型；在对视口预测时，分别预测了视口中心位置经纬度的规律，没有进一步探索其余预测模式，如基于区块的预测，这些因素都将会影响预测准确度，后续需要综合考虑深入研究。

（2）在资源调度方面，本文只是使用数据量较大这一个特征来代表全景视频业务进行调度，没有考虑全景视频的其他特性，如视口区域数据的重要性明显高于其他区域的数据，应当提高这些数据的优先级。同时在仿真过程中只考虑了单小区多用户的场景，而实际中更多的是多小区场景，存在小区间干扰等现象，之后应当更加紧密结合全景视频的特性，针对多小区的场景，进一步优化资源调度算法的性能。

（3）本文在研究资源调度算法时聚焦于LTE网络，而在研究后期，5G网络的发展逐渐加快，相关技术研究也不断完善，未来5G网络的覆盖将改善全景视频的传输问题，因此后续可以考虑结合新型网络架构与技术，进一步优化全景视频的传输。

# 参考文献

[1] 孟祥翔. 大视场虚拟现实头盔显示器光学系统研究[J]. 2015.

[2] <http://www.gov.cn/xinwen/2019-06/20/content_5401891.htm>

[3] Jerald J J. Scene-motion-and latency-perception thresholds for head-mounted

displays [D]. North Carolina: The University of North Carolina at Chapel Hill, 2009.

[4] 赖立冬. 全景视频基于视点的自适应传输技术研究[D]. 北京邮电大学, 2018.

[5] 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, 2017, pp. 315–323.

[6] M.Hosseini and V.Swaminathan, “Adaptive 360 vr video streaming based on mpeg-dash srd,” in Proc. Multimedia (ISM),2016IEEEInternationalSymposiumon,2016,pp. 407–408.

[7] Yanan Bao, Huasen Wu, Tianxiao Zhang,el. Shooting a Moving Target: Motion-Prediction-Based Transmission for 360-Degree Videos[C]// 2016 IEEE International Conference on Big Data. IEEE, 2016.

[8] Feng Qian, Lusheng Ji, Bo Han, et al. Optimizing 360 video delivery over cellular networks[C]// the 5th Workshop. ACM, 2016.

[9] Ban Y , Xie L , Xu Z , et al. CUB360: Exploiting Cross-Users Behaviors for Viewport Prediction in 360 Video Adaptive Streaming[C]// 2018 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2018.

[10] Petrangeli, Stefano & Simon, Gwendal & Swaminathan, Viswanathan. (2018). Trajectory-Based Viewport Prediction for 360-Degree Virtual Reality Videos. 157-160. 10.1109/AIVR.2018.00033.

[11] Xie, Lan & Zhang, Xinggong & Guo, Zongming. (2018). CLS: A Cross-user Learning based System for Improving QoE in 360-degree Video Adaptive Streaming. 564-572.

[12] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara. A deep multi-level network for saliency prediction. In Pattern Recognition (ICPR), 2016 23rd International Conference on, pages 3488–3493. IEEE, 2016.

[13] S. Jetley, N. Murray, and E. Vig. End-to-end saliency mapping via probability distribution prediction. Proceedings of Computer Vision and Pattern Recognition 2016, pages 5753–5761, 2016.

[14] J. Pan, E. Sayrol, X. Giro-i Nieto, K. McGuinness, and N. E. O’Connor. Shallow and deep convolutional networks for saliency prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 598–606, 2016.

[15] J. Pan, C. Canton, K. McGuinness, N. E. O’Connor, J. Torres, E. Sayrol, and X. Giro-i Nieto. Salgan: Visual saliency prediction with generative adversarial networks. arXiv preprint arXiv:1701.01081, 2017.

[16] Battisti, Federica, Baldoni, Sara, Brizzi, Michele,el. A feature-based approach for saliency estimation of omni-directional images[J]. Signal Processing Image Communication,2018

[17]Lebreton P, Raake A. GBVS360, BMS360, ProSal: Extending existing saliency prediction models from 2D to omnidirectional images[J]. Signal Processing Image Communication,2018,69:69-78

[18] Monroy, Rafael, Lutz, Sebastian, Chalasani, Tejo,等. SalNet360: Saliency maps for omni-directional images with CNN[J]. Signal Processing Image Communication, 2018,69:26-34

[19] C.-L. Fan, J. Lee, W.-C. Lo, C.-Y. Huang, K.-T. Chen, and C.-H. Hsu. Fixation prediction for 360 video streaming in head-mounted virtual reality. In Proceedings of the 27th Workshop on Network and Operating Systems Support for Digital Audio and Video, pages 67–72. ACM, 2017.

[20] Xu Mai, Song Yuhang, Wang Jianyi,el. Predicting Head Movement in Panoramic Video: A Deep Reinforcement Learning Approach[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence:1-1.

[21] Yang, Qin & Zou, Junni & Tang, Kexin & Li, Chenglin & Xiong, Hongkai. (2019). Single and Sequential Viewports Prediction for 360-Degree Video Streaming. 1-5. 10.1109/ISCAS.2019.8702654.

[22] Y. Xu, Y. Dong, J. Wu, Z. Sun, Z. Shi, J. Yu, and S. Gao. Gaze prediction in dynamic 360 immersive videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5333–5342, 2018.

[23] Zhu X J , Zeng H , Huang K , et al. Round-robin based scheduling algorithms for FIFO IQ switch[C]// Proceedings of the IEEE International Conference on Networking, Sensing and Control, ICNSC 2008, Hainan, China, 6-8 April 2008. IEEE, 2008.

[24] Aoude M , Rahal K . Scheduling Algorithms Performance of HSDPA over Wireless Channels[C]// International Conference on Information & Communication Technologies: from Theory to Applications. IEEE, 2008.

[25] Cui Chunfeng, Du Lei, Zhang Ping. A CDMA based scheduling algorithm with IP QoS guarantee[C]// Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual. IEEE, 2003.

[26] 付军峰.HSDPA中的分组调度算法[J].世界电信，2006，19(4):47-50.

[27] Kargahi M, Movaghar A. A method for performance analysis of earliest-deadline-first scheduling policy[C]// 2004.

[28] P. Ameigeiras, J. Wigard, P. Mogensen. Performance of the M-LWDF scheduling algorithm for streaming services in HSDPA[C]// Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th. IEEE, 2004.

[29] Andrews M. Providing Quality of Service over a shared wireless link[J]. 2001, 39(2):150-154.

[30] Mahfoudi M , Bekkali M E , Najd A , et al. A New Downlink Scheduling Algorithm Proposed for Real Time Traffic in LTE System[J]. International Journal of Electronics and Telecommunications, 2015, 61(4).

[31] 李苑平. LTE下行资源调度算法研究[D]. 2016.

[32] Alfayly A , Mkwawa I H . QoE-based performance evaluation of scheduling algorithms over LTE[C]// Globecom Workshops (GC Wkshps). IEEE, 2012.

[33] Mushtaq M S , Augustin B , Mellouk A . QoE-based LTE downlink scheduler for VoIP[C]// IEEE Wireless Communications and Networking Conference (WCNC). IEEE, 2014.

[34] Oyman O, Singh S. Quality of experience for HTTP adaptive streaming services[J]. IEEE Communications Magazine, 2012,50(4): 20-27.

[35] Xie, Lan & Zhang, Xinggong & Guo, Zongming. (2018). CLS: A Cross-user Learning based System for Improving QoE in 360-degree Video Adaptive Streaming. 564-572.

[36] Ge X , Pan L , Li Q , et al. Multipath Cooperative Communications Networks for Augmented and Virtual Reality Transmission[J]. IEEE transactions on multimedia, 2017, 19(10):2345-2358.

[37] 邓瑞. 虚拟现实视频无线传输研究现状及发展动态分析[J]. 移动通信, 2019, 43(03):58-67.

[38] Ioan-Sorin Comşa, Zhang S , Aydin M E , et al. A novel dynamic Q-learning-based scheduler technique for LTE-advanced technologies using neural networks[C]// Local Computer Networks (LCN), 2012 IEEE 37th Conference on. IEEE, 2012.

[39] Santos, Einar Cesar. A Simple Reinforcement Learning Mechanism for Resource Allocation in LTE-A Networks with Markov Decision Process and Q-Learning[J].

[40] Dickey, David. Time Series Theory and Methods[J]. Technometrics, 31(1):121-121.

[41] Dennis Arjo. Statistical Models: Theory and Practice[J]. Technometrics, 2009, 48(2):315-315.

[42] C. Cortes, VN Vapnik. Support Vector Networks[J]. Machine Learning, 1995, 20(3):273-297.

[43] Tin Kam Ho. Random decision forests[C]// Document Analysis and Recognition, 1995. Proceedings of the Third International Conference on. IEEE Computer Society, 1995.

[44] B. W. White, Frank Rosenblatt. Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms[J]. The American Journal of Psychology, 1963, 76(4):705.

[45] Thomas Kolarik G R . Time Series Forecasting Using Neural Networks[J]. Acm Sigapl Apl Quote Quad, 1994, 25(1):86-94.

[46] G.Peter Zhang. Time series forecasting using a hybrid ARIMA and neural network model[J]. Neurocomputing, 50(none):159-175.

[47] G.B. Sahoo, C. Ray. Flow forecasting for a Hawaii stream using rating curves and neural networks[J]. 317(1-2):0-80.

[48] Arzum Erken Celik, Yalcin Karatepe. Evaluating and forecasting banking crises through neural network models: An application for Turkish banking sector[J]. Expert Systems with Applications, 33(4):809-815.

[49] Schmidhuber, Jürgen. Deep learning in neural networks: An overview[J]. Neural Netw, 61:85-117.

[50] Hornic K. Multilayer feedforward networks are universal approximators[J]. 1989, 2(5):359-366.

[51] David, E. J., J. Guti´errez, A. Coutrot, M. P. Da Silva, and P. L. Callet (2018). A dataset of head and eye movements for 360 videos. In Proceedings of the 9th ACM Multimedia Systems Conference, pp. 432–437. ACM.

[52] Corbillon, X., F. De Simone, and G. Simon (2017). 360-degree video head movement dataset. In Proceedings of the 8th ACM on Multimedia Systems Conference, pp. 199–204. ACM.

[53] Wu, C., Z. Tan, Z. Wang, and S. Yang (2017). A dataset for exploring user behaviors in vr spherical video streaming. In Proceedings of the 8th ACM on Multimedia Systems Conference, pp. 193–198. ACM.

[54] Hochreiter, Sepp, Schmidhuber, Jürgen. Long Short-Term Memory[J]. Neural Computation, 9(8):1735-1780.

[55] Kingma D , Ba J . Adam: A Method for Stochastic Optimization[J]. Computer Science, 2014.

[56] Xu, Tan & Qian, Feng & Han, Bo. (2019). Content Assisted Viewport Prediction for Panoramic Video Streaming.

[57] 官微, 段红光. LTE关键技术及其发展趋势分析[J]. 电子测试(5):25-28+60. 2009

[58] Sesia S, Toufik I, Mathew Baker. LTE-UMTS长期演进理论与实践[M]. 2009.

[59] Cavalcanti, Francisco Rodrigo Porto. Resource Allocation and MIMO for 4G and Beyond ||[J]. 10.1007/978-1-4614-8057-0.

[60] Piro, G, Grieco, L.A, Boggia, G,等. Simulating LTE Cellular Systems: An Open-Source Framework[J]. IEEE Transactions on Vehicular Technology, 60(2):498-513.