

大作业事项

1. 分数占比

大作业分数占期末总评的 30%。大作业内分数分为基础分+附加分两部分，基础分占比 75%，附加分占比 25%。基础分为必做内容的完成度，附加分为扩展内容的完成度，扩展内容鼓励多做，至少选做 2 项。

2. 开展形式

同学们自由组队，推荐为 4-5 人每组。本次大作业共 10 个题目，每个题目供 2 个组选择。16 周课上 poster 展示，现场打分。

3. 提交内容

主要分为两部分。

- 1) 项目海报。参照给出的学术会议 poster 形式，总结自己组的工作内容，制作一份海报。
- 2) 一份提交文件，包括项目报告文档，代码，以及可能有的演示材料，如截图、视频等，具体在每道题后面都有要求。

题目三——流式视频传输系统设计与码率自适应优化

实验目标

1. 掌握基于 TCP 的流式视频传输原理与分片封装技术；
2. 实现客户端与服务端的动态交互及视频实时播放功能；
3. 设计码率自适应算法（ABR）以应对网络波动；
4. 探索流式视频的高效路径优化与数据分片重组机制。

实验要求

1. 基础功能（必做）

服务端功能

- 支持视频分片处理：将长视频分割为固定大小的片段（如每段 5 秒），封装为 TS 格式文件；
- 基于 TCP 协议实现分片传输，支持并发处理多个客户端请求；
- 记录每个分片的发送时间与码率，并生成传输日志。

客户端功能

- 动态请求视频分片：循环向服务端请求分片文件（如 ocean-1080p-8000k-0.ts），支持分片序号拼接；
- 接收分片后调用播放器库实现实时播放
- 本地缓存管理：将接收的分片临时存储至 download 文件夹，播放完成后自动清理。

2. 扩展功能（选做，任选 2 项）

- 码率自适应传输：根据网络带宽波动动态切换视频码率（如 1080p→480p），需实现 ABR 算法并统计 QoE（用户体验质量）；
- 动态路径优化：集成 Q 学习算法优化传输路径，降低传输延迟并提升吞吐量；
- 滑动窗口流式处理：实现分片数据的滑动窗口重组机制，支持长视频流无缝播放；
- 端到端加密传输：使用 AES 加密视频分片，确保传输安全性。

技术栈要求

- 编程语言：C/Python（推荐 C 语言，与实验框架兼容）；
- 通信协议：TCP（保证分片顺序与完整性），UDP 可选用于状态监控；

- 工具库：FFmpeg（视频分片处理）、Wireshark（网络抓包分析）；
- 数据存储：本地文件系统（分片存储）、SQLite（传输日志记录）。

分工建议

服务端开发（分片生成、并发传输与日志管理） 1-2 人

客户端开发（分片请求、播放器集成与缓存） 1-2 人

算法设计 （ABR 算法实现与路径优化） 1 人

测试与调优（网络波动模拟与性能分析） 1 人

实验提交内容

1. 代码

- 服务端与客户端完整源码（含分片处理与 ABR 算法实现）；

2. 文档

- 设计文档：系统架构图、分片协议格式、ABR 算法流程图；
- 测试报告：不同网络带宽下的卡顿率与 PSNR 对比分析；
- 用户手册：环境配置步骤与播放器调试说明。

3. 演示

- 实时展示分片传输与播放过程（至少 3 个客户端同时运行）；
- 模拟网络带宽骤降场景，展示码率自适应切换效果。