

**Assignment 1 Solution**

**Statistical Methods for Data Science (CDS 6212)**

**Abdulrahman Alosaimi**

# **Introduction**

In this assignment solution I would like to do all examples in same filed as its one of current projects that I work on.

Video conferencing become more popular solution after covid19 and  the main video conference platforms such as Microsoft Teams, Zoom, Google Meet are used as educational system, business meeting …etc.



WebRTC (Web Real-Time Communication) allows real-time communication in web browsers and mobile applications. The WebRTC project is open-source and supported by Apple,

Google, Microsoft and Mozilla, amongst others.

When trying to develop or host a solution for multi-party video conferencing and real-time streaming applications, there are some challenges that need to be considered depending on the organization, the type of users, type of the meetings, in addition to the technical challenges.

So I will suggest four examples in video conference system as following:

1. Effect of Gender on Communication Patterns in Virtual Meetings
2. Effect of Cache Memory on Website Loading Time
3. Impact of Different Codec Configurations on WebRTC Call Quality
4. Implements MCU vs SFU for WebRTC Scalability (Bandwidth)

**☰ Note:** While all examples are in the same domain, some variables are expected to be similar. However, the most prominent variables will be selected for each example.

# **Observational Studies for WebRTC Meeting:**

**Effect of Gender on Communication Patterns in Virtual Meetings**

This observational study looks at how participant gender (male vs. female) affects communication styles and engagement levels in online meetings.

The frequency of speaking, user feedback and modes of interaction between participants of various genders during the sessions will be monitored for this study.

### Variables

* **Independent variable:**

Gender of the participant (Male, Female)

* **Dependent variable:** 
  + Frequency of speaking
  + Interruption rates
  + Use of video/audio
  + User feedback
* **Extraneous variables:**

|  |  |
| --- | --- |
| **Variable** | **Impact** |
| **Meeting time** | The time difference may correspond to later times for some participants |
| **Meeting type** | Formal meetings seem to be organized rather than a brain-storming meetings |
| **Meeting size** | Larger groups may see more people speaking less due to the higher number of participants |
| **Meeting duration** | Long meeting may effect on users interaction |

### Study Design:

* **Advantages:**
  + **Controlled:** we can manipulate codec configurations and observe the output.
  + **Meeting distribution:** it help moderator to determine meeting duration, distribute meeting in two or three meeting.
* **Disadvantages:**
  + **Bias selection**.
  + **Technical issues**: some users may facing technical issues that can effect on study results.

**Effect of Cache Memory on Website Loading Time**

In modern web applications, caching is a technique used to reduce website load times by temporarily storing copies data or information mostly from remote server (database) in a faster storage location (RAM). Cache memory solutions, such as Redis, can significantly impact a website’s performance by decreasing server load and speeding up data retrieval.

This observational study investigates how cache memory affects website loading time by observing performance metrics under different cache configurations.

We will split the users into two groups, assign members randomly:

* Group A: Website with **no caching**.
* Group B: Website with **caching** enabled (e.g., Redis cache).

Each group will be monitored over a specific period, and the website loading time will be measured, result shown in figure 1.

Figure 1 Loading Time for Website

### Variables

* **Independent variable:**
  + **Memory configuration:** 
    - * Cache disabled
      * TTL (time to live) for cache data
      * Caching mechanisms
* **Dependent variable:** 
  + **Loading time:** it represents how long time a website takes to fully loads for a user/browser.
* **Extraneous variables:**

|  |  |
| --- | --- |
| **Variable** | **Impact** |
| **Browser Differences** | Variations in how different browsers (mobile vs. desktop) load the website |
| **Server Performance** | Differences in the server’s CPU, memory, and disk usage |
| **Network Latency** | With high network latency, it may incorrectly appear that cache memory has little or no effect on load time |

### Study Design:

* **Advantages:**
  + **Easy to setup:** We can do this study on local or remote server.
* **Disadvantages:**
  + **Lack or realism:** Caching on local server doesn’t take a other factors in result.
* **Improvements:**
  + **Replicate**  testing and randomly sample website load times across different browsers, devices, and locations to ensure a more representative sample

# **Experimental Studies for WebRTC Meeting:**

**Impact of Different Codec Configurations on WebRTC Call Quality**

This study focus in WebRTC codecs and analyze and compare how codecs impact on meeting quality, The IETF [[RFC7742](https://datatracker.ietf.org/doc/html/rfc7742)] specifies the video codecs and their parameters for WebRTC applications.

**What is H.264?**

The H.264 format, also known as AVC or Advanced Video Coding, is a part of the MPEG4-Part 10 specification. endpoints MUST support the payload formats defined in [RFC6184]

**What is VP8/VP9?**

VP9 is the successor of the earlier VP8 codec defined in [RFC6386]. It is an open-source codec developed by Google and officially released in 2013. The aim of building VP9 was to provide a high-quality video streaming experience without using too much bandwidth.

**Testing environment**

Tests were done with a simple web page establishing a connection between 2 peer connections

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | % of CPU usage with different codecs | | | | | **QVGA** | **200Kbps** | **800Kbps** | **2Mbps** | | **VP8** | 18 | 22 | 28 | | **VP9** | 20 | 28 | 33 | | **H264** | 10 | 14 | 15 | | **AV1** | 36 | 46 | 50 | |  |

Figure 2 CPU Usage

In our study we will randomly assign participants to different codec configurations two or three groups and manipulate WebRTC video codec (VP8, VP9 or H.264) to measure the quality of video, and latency to evaluate the impact of each codec under controlled conditions.

### Variables

* **Independent variable:**

The codecs being used, such as VP8, VP9, or H.264

* **Dependent variable:** 
  + Video clarity (resolution, frame rate)
* **Extraneous variables:**

|  |  |
| --- | --- |
| **Variable** | **Impact** |
| **Network conditions** | Low quality might be attributed to network conditions rather than the codec itself |
| **Device Performance** | Call quality may be lower for participants with less capable devices |
| **Participant Location** | Geographic distance could affect call quality |

### Study Design:

* **Advantages:**
  + **Controlled:** we can manipulate codec configurations and observe the output.
  + **Cost reduction:** Its help users to choice codec.
* **Disadvantages:**
  + **Unrealism:** Network conditions simulated in a lab may not reflect real-world scenarios accurately.
  + **Proprietary codecs.**
  + **Complexity**.

**Implements MCU vs SFU for WebRTC Scalability (Bandwidth)**

In this study the main focus will be on the bandwidth consumption of MCU vs SFU WebRTC architecture, given that bandwidth is a critical resource that can affect call quality, latency, and overall performance. The study will simulate multiple scenarios with varying numbers of participants to measure the differences in bandwidth usage.

**Multipoint Control Unit (MCU)**

Multipoint Control Unit (MCU) is a server in the middle of all participants functioning as a central gateway that is combining everyone's video and audio feeds.

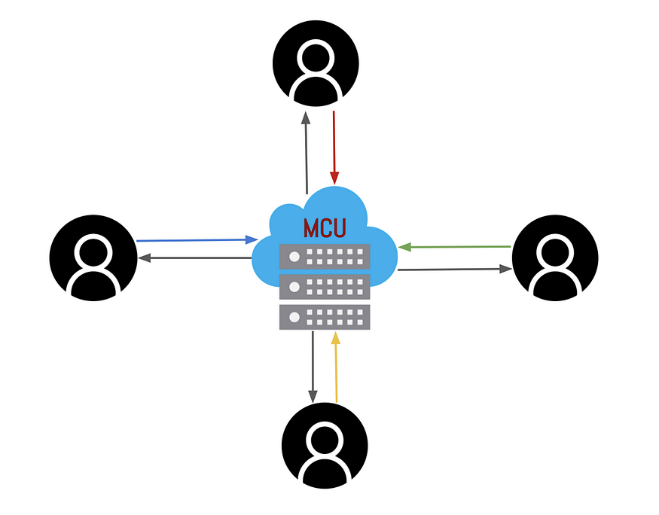


Figure 3 MCU

**Selective Forwarding Unit (SFU)**

Selective Forwarding Unit (SFU) is a server that receives audio and video streams from endpoints and relays them to everyone else (endpoints send one and receive many). More information about architecture of an SFU can be found at [RFC766] “RTP Topologies” [section 3.7](https://tools.ietf.org/html/rfc7667#section-3.7).

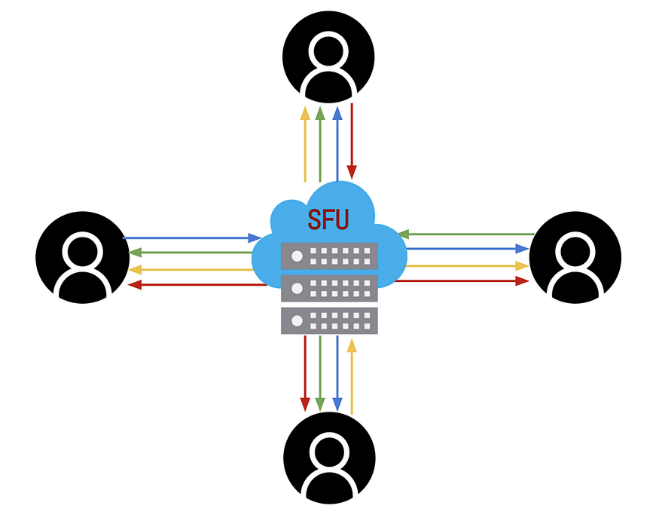


Figure 4 SFU

### Variables

* **Independent variable:**
  + **WebRTC Architecture**:
    - **MCU:** Centralized processing of all streams and forwarding of a composite stream to participants.
    - **SFU:** Selectively forwards streams without centralized processing or mixing.
  + **Number of Participants**: Increasing group sizes, such as small (2–4 participants), medium (5–10), large (10–20), and very large (20+).
  + **Video resolution:** 480p, 720p, 1080p.
* **Dependent variable:** 
  + **Bandwidth Consumption** : The amount of data transferred per second per participant.
* **Extraneous variables:**

|  |  |
| --- | --- |
| **Variable** | **Impact** |
| **Hardware Appliance** | The physical hardware used in the server (CPU, RAM, etc.) and the server’s overall load may affect performance and bandwidth usage |
| **infrastructure** | Other services on infrastructure may increase bandwidth traffic |
| **Firewall** | Apply strict rules may effects on latency |
| **Resolution and Frame Rate** | High video resolutions (e.g., 720p, 1080p) and frame rates can affect the bandwidth |

### Study Design:

The design will involve multiple test scenarios that vary the number of participants and the resolution of video streams. For each test, the same scenarios will be run using both MCU and SFU to ensure a valid comparison.

* **Advantages:**
  + **High level of control**.
  + **Measurable.**
  + **Optimal use of infrastructure resources.**
* **Disadvantages:**
  + **Costs**: To cover all types of cases we need a multiple camera with different resolutions, reliable infrastructure and high speed internet.
  + **Bandwidth only**: This study focuses on bandwidth only and leave other important factor such as CPU usage, memory consumption …etc.
* **Improvements:**
* Implement full study on more factors and do this study on real world
* **Testing Replication**

# **Ethical Considerations**

While all studies in the same domain I set all ethical considerations in same section

* **Data Privacy**:

Users need to be sure that their data (e.g., video and audio streams) is treated confidentially and not stored or shared without consent.

* **Informed Participants:**

Real users should be informed about the nature of the study, including the fact that their traffic usage and performance metrics will be monitored. They should provide informed consent before participating.

* **Server Load Considerations**:

The experiment should be designed so that users do not experience significant degradation in their call quality due to overloaded servers. If the server becomes overloaded, causing call drop services or poor call quality in production environment.

* **Transparency**:

It is essential to accurately report the findings of the study, including both the successes and the limitations.

# **References**

* **IEEE Communications Standards Magazine**: This magazine covers various standards and protocols related to communication technologies, including WebRTC. It can be an authoritative source for understanding WebRTC's role in real-time communication systems.

Link: [IEEE Communications Standards Magazine](https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=7910733)

* **IEEE Standard 802.1**: Though WebRTC does not have a dedicated IEEE standard, it often interacts with protocols governed by networking standards like IEEE 802.1 (Ethernet), which can be relevant for understanding the network layer's impact on real-time communication.

Link: [IEEE 802.1 Standards](https://standards.ieee.org/standard/802_1-2022.html)

* **RFC 8825 and RFC 8827 by IETF**: You can refer to IEEE papers that discuss IETF's RFC 8825 and RFC 8827 on WebRTC architecture and use cases.

IETF links:

* RFC 8825 - WebRTC Overview
* RFC 8827 - WebRTC Media Transport
* **RFC 7742, RFC 6184 and RFC 6386 by IETF:** Data Format and Decoding Guide
* **WebRTC official website:** This page is maintained by the Google WebRTC team.

Links: https://webrtc.org