

# 视频用户体验评估——函数关系的确立

2017 年 5 月 28 日

## 1 摘要

## 2 问题重述

### 2.1 问题的提出

随着科技的进步——无线宽带网络的升级、智能用户端的便携化、各类视频直播软件的普及，愈来愈多的用户选择在移动智能终端上使用应用客户端（APP）观看网络视频，这就是我们常说的流媒体业务。

流媒体的典型传输方式为：连续采样的多媒体信息经编码压缩后，从源端连续、实时地发送到网络，接收端从网络收到部分数据后就可以开始解码播放，而后续接收的数据则持续不断地存入本地缓存，形成可连续播放的媒体流。可以看出，相对传统E-mail等传统网络业务，实时流媒体对于实时性、网络带宽、容错性、媒体同步、播放平滑性等方面均有严格要求。

相比UDP和IP等传输协议，TCP具有拥塞控制，丢包重传等机制，因而更加可靠。因此，目前的流媒体传输技术主要基于TCP传输控制协议。

流媒体业务模型中，影响用户体验的两个关键指标——初始缓冲等待时间和视频播放过程中的卡顿缓冲时间，主要使用初始缓冲时延(sLoading)和卡顿占比(sStalling)（卡顿时长的占比=卡顿时长/视频播放时长）来定量地评价。影响两个关键指标的主要因素包括初始缓冲峰值速率、播放阶段平

均下载速率、端到端环回时间（E2E RTT）以及视频参数。但目前以上这些因素与两个关键指标之间的关系未知。<sup>1</sup>

显然，建立用户体验变量（初始缓冲时延，卡顿时长占比）与网络侧变量（初始缓冲峰值速率，播放阶段平均下载速率，E2E RTT）之间的函数关系对于改善用户体验，提高流媒体业务服务水平具有重要意义。

## 2.2 问题的分析

由题意可知，本题的目的是为了建立函数模型来确定影响视频用户体验的两个关键指标与视频相关参数的关系。由于本题目数据较多，便于计算机进行统计和分析，故我们采用两种不同的策略进行建模。

### 2.2.1 白箱方法

白箱（class box）方法即传统意义上的数学建模方法——通过查找文献获取问题相关知识，分析其机理并建立数学模型，最终通过数据确定参数，并分析模型准确度和误差。在此，最大的挑战便在于查找文献确定端到端环回时间（RTT）在TCP丢包重传和拥塞控制机制中所起到的作用，并确定其对客户端播放产生的影响。

### 2.2.2 黑箱方法

黑箱（black box）方法，即结合计算机进行数据分析的统计学方法。黑箱方法将数据内部关系视为黑箱，仅仅从输入和输出数据的角度进行观测，并通过统计方法拟合出变量间函数关系。

近年来，以神经网络为代表的机器学习技术在工程和科研计算中得到了越来越广泛的应用，其对任意非线性多元函数进行拟合的能力相较于传统方法更加简单高效且容易推广。在拥有大量数据的基础上，神经网络可以达到相当的准确度，这是我们采取这一方法的主要原因。

---

<sup>1</sup>关于用户体验评价指标的详细信息，参见文献[1]华为用户体验白皮书。

### 3 模型假设及符号说明

#### 3.1 符号说明

1.  $v_{max}$ : 初始缓冲峰值速率。
2.  $RTT$ : E2E RTT, 往返时延。
3.  $\bar{v}$ : 播放阶段平均速率。
4.  $\tau$ : 初始缓冲时延。
5.  $D_0$ : 初始缓冲下载数据量。
6.  $\mu$ : 视频码率。
7.  $t_{total}$ : 播放阶段总时长。
8.  $t_p$ : 播放时长。
9.  $t_s$ : 卡顿时长。
10.  $\eta$ : 卡顿占比
10.  $MSS$ : 最大报文段长度。
11.  $t_r$ : 卡顿重播门限
12.  $m$ : 卡顿次数

#### 3.2 模型的假设

1. TCP传输协议带宽大于视频码率。这是TCP用于流媒体业务的基本要求。
2. TCP客户端和服务端的缓冲区大小能够容TCP的视频帧,而不会在TCP接收缓冲区溢出。考虑到目前普及的个人终端设备已经具有较高性能,因此该条件易满足。
3. 视频码率固定不变,由题目数据可以看出该条件默认成立。
4. 假设TCP协议将视频数据分包发送时,每个包的大小相同,且大小均为最大报文段长度。
5. 假设客户端的视频连续播放,即在视频播放中用户没有进行手动快进或者暂停操作。由题目数据可以看出该条件默认成立。

图 1: TCP机制示意图

6. 平均播放速率正比于网络吞吐量。

## 4 模型的建立

### 4.1 TCP协议和相关机制

在建模过程中，我们假设读者对已对TCP协议和流媒体传输的相关背景有所了解并熟悉相关术语。

在TCP协议中，影响网络能力的关键机制有如下三点。如图所示：

#### 4.1.1 1.慢启动

- 刚开始发送报文段时，发送窗口设置为一个最大报文段MSS的数值。
- 在收到对上一轮报文段的确认后，发送窗口的数值加倍。
- 其结果是在不出现丢包且发送窗口小于最大限制时发送窗口的大小指数增加。

#### 4.1.2 2.拥塞避免

- 为了防止拥塞窗口增长过大引起网络拥塞，还需要维护一个慢启动门限的状态变量，当拥塞窗口的值小于慢启动门限时，使用慢启动算法，一旦拥塞窗口的值大于慢启动门限的值，就改用拥塞避免算法。
- 拥塞避免算法的思路是让拥塞窗口缓慢地增大，收到每一轮的确认后，将拥塞窗口的值加1，而不是加倍，其结果是拥塞窗口的值按照线性规律缓慢增长。
- 无论是在慢启动阶段还是在拥塞避免阶段，只要发送方判断网络出现拥塞（没有按时收到确认），就把慢启动门限设置为出现拥塞时发送窗口值的一半，但最小不能小于2个MSS值，而后把拥塞窗口的值重新设置为1个MSS，执行慢启动算法。

图 2: 初始缓冲阶段示意图

### 4.1.3 3.丢包重传

每个数据包发送后均启动一个计时器。之后，TCP通过两种方式判断发送的数据包是否丢失。第一种是收到三个连续ACK，第二种是数据包计时器时间截止后仍未收到该包的ACK确认。当确认丢包之后，拥塞避免机制启动。发送窗口减小，同时重传丢失数据包。并将计时器时间加倍。这是TCP协议确保数据完整传输的关键机制。

在建模需要的前提下，我们对以上机制进行了简要的说明。详细信息可见参考文献[1]。

## 4.2 时间离散化

对于本模型而言，每个时间单位对应一个回合的长度，一个回合的长度通常被定义为RTT。<sup>2</sup> 一个回合开始的标志为拥塞窗口中的数据包进入传输层开始发送。当拥塞窗口中所有的数据包被发出后，在收到ACKs之前，TCP将不会再发出任何包。ACK标志着本轮传输的结束与下轮传输的开始。

## 4.3 白箱——分析方法

### 4.3.1 初始缓冲时延

依据文献[2]，初始加载阶段分为视频解析和数据下载缓冲两个子阶段。如图所示：视频解析阶段的持续时长和终端设备的设计原理有关，通常为端到端回环时间 $RTT$ 的某个倍数 $n$ 。而数据下载缓冲阶段的持续时长与所需最小初始缓冲数据量以及初始缓冲峰值速率 $v_{max}$ 有关。

当TCP连接初次建立并开始视频缓冲时需要经历慢启动过程。假设在TCP慢启动过程中不存在丢包现象，则每一个RTT时间回合内TCP发送端的发送窗口增加一倍。其过程如下图所示：

<sup>2</sup>时间离散化是TCP建模的核心思想，见参考文献[1]

图 3: TCP慢启动示意图

图 4: TCP周期波动示意图

设 $s$ 为TCP慢启动到达峰值速率所需的时间对应的 $RTT$ 数。可知,  $v_{max}$ 与 $s$ 关系为:

$$v_{max} = MSS * 2^s / RTT$$

其中 $MSS$ 为最大报文段长度, 在当前技术标准下其值通常为536Bytes或1460Bytes[]。

慢启动阶段下载的数据量与TCP峰值速度的关系为:

$$Ds = 2(v_{max} * RTT - MSS)$$

然而, 由于TCP的拥塞控制机制, TCP在到达峰值速度后, 无法持续维持这一速度, 其传输速率周期性波动。其到达最大之后其速度变化如下图所示。出现这一现象的原因是数据包在传输过程中有 $p$ 的概率丢失, 此时TCP拥塞控制机制会将发送窗口数量减半。而后拥塞控制机制控制发送窗口线性增长, 具体细节参见文献[]

故而在实际传输中, 缓冲平均阶段速度 $\bar{v}_0 = \frac{3}{4}v_{max}$

综合以上结论, 可得初始缓冲时延与 $RTT$ 和 $v_{max}$ 的关系

$$\tau = \frac{D_0 - Ds}{\bar{v}_0} + (n + s) * RTT \quad (1)$$

其中 $n$ 为常数, 表示视频解析时间对应的 $RTT$ 数。 $s$ 为TCP慢启动到达峰值速率所需的时间对应的 $RTT$ 数。

在后续模型检验中, 我们发现这一同时, 文献还推导了传输速率的Mathis公式:

#### 4.3.2 卡顿占比

依据文献[]流媒体视频产生卡顿的判断标准为 $\bar{v} \leq 1.5\mu$ 。然而, 实际数据中却有很大一部分满足这一标准却仍然产生卡顿。显然, 产生卡顿的原

因并非网络带宽不够，而是丢包率过高。假设数据包丢失和视频卡顿同时发生，根据TCP拥塞避免机制，此时TCP发送窗口被设置为1，随后发生的过程接近TCP缓冲阶段。在这一阶段，缓冲的平均速度应大于播放过程中的平均速率。且其应正比于此时的网络吞吐量。故处于简化模型的考虑，可直接

简单模型如下：

由于卡顿占比

$$\eta = \frac{t_s}{t_p}$$

故在视频播放时间 $t_p$ 已知的情况下，只需要找到卡顿时间 $t_s$ 与网络侧参数的关系即可完成对卡顿占比的建模。依据题目给出资料，每次视频播放出现卡顿的重播门限为2.7s。故每次卡顿后缓冲数据量

$$Dr = 2.7\mu$$

$$t_r = \frac{Dr}{2 * \bar{v}}$$

总卡顿时间

$$t_s = m * t_r$$

#### 4.4 黑箱——统计方法

由于网络侧其他变量如视频码率、初始缓冲时间等均为常量或仅在小范围内取值。故不考虑其对用户体验变量的影响。因此用户体验变量 $\tau\eta$ 和网络侧变量 $RTT$ 、 $v_{max}$ 、 $\bar{v}$ 的关系可以视为一个多元非线性函数。

$$\begin{bmatrix} \tau \\ \eta \end{bmatrix} = f\left(\begin{bmatrix} RTT \\ v_{max} \\ \bar{v} \end{bmatrix}\right) \quad (2)$$

在神经网络的具体设计中，该函数可以由输入层包含三个神经元，输出层包含两个神经元并且含有若干隐藏层的神经网络表示。在此，我们采用了含有六个神经元的单隐藏层，其结构如图所示：

由于拟合目标是连续的，故设计神经网络的关键指标如下：

- 激活函数：Sigmoid即S形函数

$$Sigmoid(\sigma) = \frac{1}{1 + e^{-\sigma}}$$

- 损失函数：Cross-Entropy交叉熵函数

$$C() = -\frac{1}{n} \sum_x [y \log a + (1 - y) \log(1 - a)]$$

其中 $n$ 为该组训练数据的包含数据数目，求和作用于所有输入数据 $x$ ， $a$ 为神经网络输出结果， $y$ 为预期结果。

神经网络的训练采用反向传播结合梯度下降算法。具体计算技术参见模型的求解方法部分。

## 5 模型的求解及结果分析

### 5.1 模型的求解方法

#### 5.1.1 回归和拟合

#### 5.1.2 神经网络

### 5.2 结果分析

由相上述分析可知，缓冲阶段时长与初始缓冲速度近似线性负相关。其相关系数  $R =$

## 6 模型的检验

## 7 模型的优缺点及改进方向

### 7.1 优点

### 7.2 缺点

### 7.3 改进方向

## 8 附录