

(19) 中华人民共和国国家知识产权局



(12) 发明专利申请

(10) 申请公布号 CN 103702237 A

(43) 申请公布日 2014. 04. 02

(21) 申请号 201210366974. 2

(22) 申请日 2012. 09. 28

(71) 申请人 北京大学

地址 100871 北京市海淀区颐和园路 5 号

申请人 北大方正集团有限公司

北京北大方正电子有限公司

(72) 发明人 周超 张行功 郭宗明

(74) 专利代理机构 北京英赛嘉华知识产权代理
有限责任公司 11204

代理人 王达佐

(51) Int. Cl.

H04N 21/643 (2011. 01)

H04N 21/238 (2011. 01)

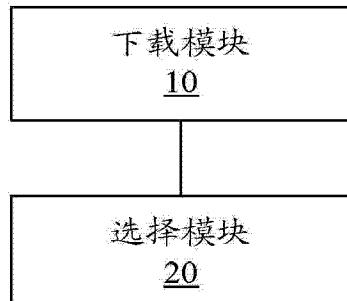
权利要求书3页 说明书5页 附图1页

(54) 发明名称

HTTP 流媒体的速率自适方法及装置

(57) 摘要

本发明公开了一种 HTTP 流媒体的速率自适方法和装置, 方法包括: 客户端在传输层采用 TCP 协议, 在应用层采用 HTTP 协议, 从服务端获取选择码率的流媒体, 其中, 选择的码率使得在当前的下载速率下用于缓存流媒体的缓冲区的缓存数据量保持在设定的范围内。本发明提高了用户体验。



1. 一种 HTTP 流媒体的速率自适方法,其特征在于,包括:

客户端在传输层采用 TCP 协议,在应用层采用 HTTP 协议,从服务端获取选择码率的所述流媒体,其中,选择的码率使得在当前的下载速率下用于缓存所述流媒体的缓冲区的缓存数据量保持在设定的范围内。

2. 根据权利要求 1 所述的方法,其特征在于,还包括:

预先将视频内容编码成具有不同码率的多个视频文件,将每个所述视频文件按时间先后切分成设定长度的多个视频片段,并作为所述流媒体存储在所述服务端。

3. 根据权利要求 2 所述的方法,其特征在于,客户端通过 HTTP/TCP 协议从服务端获取选择码率的所述流媒体包括:

所述客户端下载完第 k-1 个视频片段的所述流媒体之后,根据当前的下载速率和所述缓冲区的数据量确定相应的码率;

所述客户端从所述服务端获取具有所述相应码率的第 k 个视频片段的所述流媒体。

4. 根据权利要求 3 所述的方法,其特征在于,根据当前的下载速率和所述缓冲区的数据量确定相应的码率包括:设置

$$r(k) = \begin{cases} Q(r^{over}(k)) & \text{如果 } d(t) > p(t) \& r^{over}(k) \geq d(t) \\ Q(r^{under}(k)) & \text{如果 } d(t) < p(t) \& r^{under}(k) \geq \alpha r_1 \\ r(k-1) & \text{如果 } d(t) = p(t) \\ r_1 & \text{其他} \end{cases}$$

公式中各个变量的定义如下:

$r^{over}(k)$:使所述缓冲区不上溢所允许的最大码率;

$r^{under}(k)$:使所述缓冲区不下溢所允许的最大码率;

$d(t)$:在时刻 t 时的下载速率;

$p(t)$:在时刻 t 时的播放速率;

$V = \{r_1, r_2, \dots, r_L\}$:可选用视频码率的集合,满足 $r_1 < r_2 < \dots < r_L$;

$r(k)$:被选择的第 k 个片段的视频码率;

$Q(r^{over}(k))$:不大于 $r^{over}(k)$ 的最大可用码率;

$Q(r^{under}(k))$:不大于 $r^{under}(k)$ 的最大可用码率。

5. 根据权利要求 4 所述的方法,其特征在于,设置 $r^{over}(k) = \frac{b_{max} - b(t_k^s)}{\left(1 - \frac{p(t)}{d(t)}\right)L}$,公式中各个变

量的定义如下: t_k^s :第 k 个片段开始下载的时刻;

$b(t)$:在时刻 t 时的接收端缓存的数据量;

L :每个视频片段的持续时间;

b_{max} :所述缓冲区允许的上溢数据量。

6. 根据权利要求 4 所述的方法,其特征在于,设置 $r^{under}(k) = \frac{b_{min} - b(t_k^s)}{\left(1 - \frac{p(t)}{d(t)}\right)L}$,其中, b_{min} 是

所述缓冲区允许的下溢数据量。

7. 根据权利要求 3 所述的方法, 其特征在于, 根据当前的下载速率和所述缓冲区的数据量确定相应的码率包括: 设置

$$r(k) = \begin{cases} Q\left(K_p * \left(b_{\max} - b(t_k^s)\right) + K_d \frac{b(t_k^s) - b(t_{k-1}^s)}{t_k^s - t_{k-1}^s}\right) & \text{如果 } d(t) > p(t) \& r^{over}(k) \geq d(t) \\ Q\left(K_p * \left(b_{\min} - b(t_k^s)\right) + K_d \frac{b(t_k^s) - b(t_{k-1}^s)}{t_k^s - t_{k-1}^s}\right) & \text{如果 } d(t) < p(t) \& r^{under}(k) \geq \alpha r_1 \\ r(k-1) & \text{如果 } d(t) = p(t) \\ r_1 & \text{其他} \end{cases}$$

公式中各个变量的定义如下:

$r^{over}(k)$ 是使所述缓冲区不上溢所允许的最大码率,

$$r^{over}(k) = \frac{b_{\max} - b(t_k^s)}{\left(1 - \frac{p(t)}{d(t)}\right)L};$$

$r^{under}(k)$ 是使所述缓冲区不下溢所允许的最小码率,

$$r^{under}(k) = \frac{b_{\min} - b(t_k^s)}{\left(1 - \frac{p(t)}{d(t)}\right)L};$$

$d(t)$ 是在时刻 t 时的下载速率;

$p(t)$ 是在时刻 t 时的播放速率;

$V = \{r_1, r_2, \dots, r_L\}$ 是可选用视频码率的集合, 满足 $r_1 < r_2 < \dots < r_L$;

$r(k)$ 是被选择的第 k 个片段的视频码率;

$Q(r^{over}(k))$ 是不大于 $r^{over}(k)$ 的最大可用码率;

$Q(r^{under}(k))$ 是不低于 $r^{under}(k)$ 的最低可用码率;

t_k^s 是第 k 个片段开始下载的时刻;

$b(t)$ 是在时刻 t 时的接收端缓存的数据量;

b_{\max} 是所述缓冲区允许的上溢数据量;

b_{\min} 是所述缓冲区允许的下溢数据量;

K_p 和 K_d 是设置的参数。

8. 根据权利要求 7 所述的方法, 其特征在于, 设置 $\begin{cases} K_p > -\frac{1}{K_1} \text{ 或 } \\ K_d > 0 \end{cases}$ $\begin{cases} K_p < -\frac{1}{K_1} \\ K_d < 0 \end{cases}$

, 其中, $\begin{cases} K_1 = \left(1 - \frac{p(t)}{d(t)}\right)L \\ K_1 = \frac{1}{K_2} \end{cases}$ 。

9. 一种 HTTP 流媒体的速率自适装置，其特征在于，包括：

下载模块，用于使客户端在传输层采用 TCP 协议，在应用层采用 HTTP 协议，从服务端获取选择码率的所述流媒体；

选择模块，用于选择所述码率，使得在当前的下载速率下用于缓存所述流媒体的缓冲区的缓存数据量保持在设定的范围内。

10. 根据权利要求 9 所述的装置，其特征在于，还包括：

编码模块，用于将视频内容编码成具有不同码率的多个视频文件；

分段模块，用于将每个所述视频文件按时间先后切分成设定长度的多个视频片段，并作为所述流媒体存储在所述服务端；

所述选择模块用于在所述客户端下载完第 $k-1$ 个视频片段的所述流媒体之后，根据当前的下载速率和所述缓冲区的数据量确定相应的码率；

所述下载模块用于使所述客户端从所述服务端获取具有所述相应码率的第 k 个视频片段的所述流媒体。

HTTP 流媒体的速率自适方法及装置

技术领域

[0001] 本发明属于网络视频传输技术领域,具体涉及 HTTP 流媒体的速率自适方法及装置。

背景技术

[0002] 近几年来,动态 HTTP 流媒体传输技术广泛应用于网络视频传输。选择视频码率过大,将导致下载速度小于播放速度,引起播放停顿;选择视频码率过小,将导致带宽浪费,视频质量较差。

[0003] 在现有系统中,由于网络带宽的时变特性,如何选择合适的视频码率从而最大化用户接受视频质量,仍然面临挑战。因此,选择合适的视频码率比较重要。

发明内容

[0004] 本发明旨在提供一种 HTTP 流媒体的速率自适方法和装置,以解决上述的问题。

[0005] 在本发明的实施例中,提供了一种 HTTP 流媒体的速率自适方法,包括:客户端在传输层采用 TCP 协议,在应用层采用 HTTP 协议,从服务端获取选择码率的流媒体,其中,选择的码率使得在当前的下载速率下用于缓存流媒体的缓冲区的缓存数据量保持在设定的范围内。

[0006] 在本发明的实施例中,提供了一种 HTTP 流媒体的速率自适装置,包括:下载模块,用于使客户端在传输层采用 TCP 协议,在应用层采用 HTTP 协议,从服务端获取选择码率的流媒体;选择模块,用于选择码率,使得在当前的下载速率下用于缓存流媒体的缓冲区的缓存数据量保持在设定的范围内。

[0007] 本发明上述实施例的 HTTP 流媒体的速率自适方法和装置通过控制缓存区的饱和度来选择速率,从而在充分利用带宽的同时,保证视频播放的连续性以及平滑性,提高了用户体验。

附图说明

[0008] 此处所说明的附图用来提供对本发明的进一步理解,构成本申请的一部分,本发明的示意性实施例及其说明用于解释本发明,并不构成对本发明的不当限定。在附图中:

[0009] 图 1 示出了根据本发明实施例的动态 HTTP 流媒体系统结构示意图;

[0010] 图 2 示出了根据本发明实施例的速率调整的比例控制示意图;

[0011] 图 3 示出了根据本发明实施例的速率调整的比例微分控制示意图;

[0012] 图 4 示出了根据本发明实施例的 HTTP 流媒体的速率自适装置的示意图。

具体实施方式

[0013] 下面将参考附图并结合实施例,来详细说明本发明。

[0014] 图 1 示出了根据本发明实施例的动态 HTTP 流媒体系统结构示意图,包括:客户端

在传输层采用 TCP 协议，在应用层采用 HTTP 协议，从服务端获取选择码率的流媒体，其中，选择的码率使得在当前的下载速率下用于缓存流媒体的缓冲区的缓存数据量保持在设定的范围内。

[0015] 在该实施例中，将接收端缓冲区中的缓存数据量作为反馈信号，调整视频码率，这相当于依据网络带宽状况调整下载速率，从而在充分利用带宽的同时，保证视频播放的连续性以及平滑性，提高了用户体验。

[0016] 优选地，在服务端利用转码技术，将视频内容编码成具有不同码率的多个视频文件；将每个转码后的视频文件切分成一定长度的多个视频片段，并以文件的形式存储在服务端；客户端依次下载视频片段，解码并播放。具体的，包括以下步骤：

[0017] 1、在服务器端，依据转码技术，将每个视频文件转码成具有不同码率的多个视频文件（图中的 Seg 1、Seg 2…… Seg N），然后将每个转码后的视频文件，进一步切分成一定时间长度的视频片段，并以文件的形式存储于服务器上；

[0018] 2、客户端依据时间顺序，依次请求下载视频片段；

[0019] 3、接收端下载完一个视频片段后，对其进行解码并播放，并开始下一个片段的下载。

[0020] 步骤 2 中视频传输采用标准的 HTTP/TCP 协议。步骤 3 中在进行视频片段的请求下载时，需要动态地决定视频的码率，从而选择相应的视频片段。

[0021] 在动态地决定视频的码率时，可以考虑码率的调整时间和调整频率的问题。

[0022] 调整时间：网络带宽波动性比较大，在进行速率调整时，如何依据带宽的变换，及时地进行速率切换，也是速率调整中面临的一大问题，调整时间过长将导致速率切换置换滞后带宽变化，进而引起接收端缓冲区上溢或下溢。

[0023] 调整频率：视频码率的频繁调整也会导致用户观看视频的体验降低，也即在带宽发生短暂的抖动时，应该避免速率的调整，从而保证视频码率的平滑性。因此，在动态 HTTP 流媒体系统中，如何选择合适的视频码率从而提高用户观看视频的体验，显得尤为必要。

[0024] 本发明的优选实施例以控制论为理论基础，下面进一步详细说明。

[0025] 首先给出一些符号定义：

[0026] t_k^s : 第 k 个片段开始下载的时刻；

[0027] t_k^e : 第 k 个片段开始下载完成的时刻；

[0028] $d(t)$: 在时刻 t 时的下载速率；

[0029] $p(t)$: 在时刻 t 时的播放速率；

[0030] $b(t)$: 在时刻 t 时的接收端缓存的数据量；

[0031] L : 每个视频片段的持续时间；

[0032] $V = \{r_1, r_2, \dots, r_L\}$: 可选用视频码率的集合，满足 $r_1 < r_2 < \dots < r_L$ ；

[0033] $r(k)$: 被选择的第 k 个片段的视频码率；

[0034] $Q(x)$: 量化函数，其定义为 $Q(x) = \arg \max_{r \in V, r \leq x} r$ 。

[0035] 第一优选实施例：

[0036] 本优选实施例的速率调整策略是一种比例控制，如图 2 所示，其中 K_1 和 K_2 为系统参数， K_1 是速率调整的正向比例控制系数， K_2 是速率调整的反馈比例控制系数，并且

$$\begin{cases} K_1 = \left(1 - \frac{p(t)}{d(t)}\right) L \\ K_2 = \frac{1}{K_1} \end{cases}$$

[0037] 当 $d(t) > p(t)$ 时, 触头 c 与触头 c_1 连接, 当 $d(t) < p(t)$ 时, 触头 c 与触头 c_2 连接。

[0038] 当下载完某一个片段时, 缓冲区中缓存数据的变化如下:

$$[0039] b(t_k^e) = b(t_k^s) + \int_{t_k^s}^{t_k^e} (d(t) - p(t)) dt$$

[0040] 假设在下载一个片段的过程中, 播放速率与下载速率均保持不变, 则:

$$[0041] t_k^e - t_k^s = \frac{r(k)}{d(t)} L$$

$$[0042] b(t_k^e) = b(t_k^s) + (d(t) - p(t))(t_k^e - t_k^s) = b(t_k^s) + r(k)L \left(1 - \frac{p(t)}{d(t)}\right)$$

[0043] 为了实现视频的流畅播放, 需要选择一种合适的视频码率, 使得缓冲区既不上溢, 也不下溢。

[0044] 控制上溢: 当 $d(t) > p(t)$ 时, 缓冲区有可能出现上溢, 导致播放异常, 为了避免上溢发生, 必须满足 $b(t_k^e) \leq b_{\max}$, 即第 k 个视频片段的速率需要满足:

$$[0045] r(k) \leq \frac{b_{\max} - b(t_k^s)}{\left(1 - \frac{p(t)}{d(t)}\right) L}$$

[0046] 定义上述不等式的右边为 $r^{\text{over}}(k)$, 代表了速率 $r(k)$ 的一个上界。当 $r^{\text{over}}(k) < d(t)$ 时, 意味着所选择的视频码率小于下载带宽, 在实际系统中, 这是不合理的。这主要是因为 $r^{\text{over}}(k)$ 是缓存数据的减函数, 意味着缓存数据很多, 此时选择一个比带宽更小的视频速率, 是没必要的。因此, 针对这种情况, 本发明优选实施例提出了一种睡眠机制, 用于消耗部分缓存数据, 从而使得 $r^{\text{over}}(k)$ 不小于下载带宽, 因此, 可以得到睡眠时间为:

$$[0047] t_s = \frac{(d(t) - p(t))L + b(t_k^s) - b_{\max}}{p(t)}$$

[0048] 控制下溢: 当 $d(t) < p(t)$ 时, 缓冲区有可能出现下溢, 导致播放停顿, 为了避免下溢发生, 必须满足 $b(t_k^e) \geq b_{\min}$, 也即第 k 个视频片段的速率需要满足:

$$[0049] r(k) \leq \frac{b_{\min} - b(t_k^s)}{\left(1 - \frac{p(t)}{d(t)}\right) L}$$

[0050] 定义上述不等式的右边为 $r^{\text{under}}(k)$, 代表了在可能出现缓冲区下溢时, 速率 $r(k)$ 的另一个上界。当 $r^{\text{under}}(k) < r_1$, 则意味着不存在可用的视频码率, 能避免缓冲区下溢。因此, 针对这种情况, 本发明优选实施例提出了一种速率重置策略, 当 $r^{\text{under}}(k)$

$< \alpha r_1$ ($\alpha \geq 1$) 时, 直接将视频码率设置为最低速率 : $r(k) = r_1$

[0051] 综上所述, 速率调整过程可以概括为 :

[0052]

$$r(k) = \begin{cases} Q(r^{over}(k)) & \text{if } d(t) > p(t) \& r^{over}(k) \geq d(t) \\ Q(r^{under}(k)) & \text{if } d(t) < p(t) \& r^{under}(k) \geq \alpha r_1 \\ r(k-1) & \text{if } d(t) = p(t) \\ r_1 & \text{else} \end{cases}$$

[0053] 其中 $Q(\cdot)$ 代表了量化函数, 上式表明, 当 $d(t) > p(t)$ 或 $d(t) < p(t)$ 时, 分别选择合适的视频码率避免缓冲区的上溢和下溢; 当 $d(t) > p(t)$ 时, 选择与上一个视频片段相同的视频码率作为目标码率; 当 $r^{under}(k) < \alpha r_1$ 太小时, 直接重置视频片段的码率为最低可用的视频码率。

[0054] 第二优选实施例 :

[0055] 上述比例控制系统一般具有稳态误差, 而且容易导致输出码率的震荡。本优选实施例的速率调整策略是一种比例微分系统, 即将图 2 中的比例环节换成比例微分环节, 从而提高速率调整的性能, 如图 3 所示, 其中的比例微分控制环节为 $G_c(s)$, 其形式为 : $G_c(s) = K_p + K_d s$

[0056] 此时, 系统的传输函数为 : $H(s) = \frac{K_1 G_c(s)}{1 + K_1 G_c(s)} = \frac{K_1 K_p + K_1 K_d s}{1 + K_1 K_p + K_1 K_d s}$

[0057] 其中, K_p 和 K_d 是系统参数, 为了保证系统的稳定性, 系统传输函数的极点必须位于 s 平面的左边, 也即 $s_p = -\frac{1 + K_1 K_p}{K_1 K_d s} < 0$, 从而 K_p 和 K_d 需满足如下条件 :

[0058] $\begin{cases} K_p > -\frac{1}{K_1} \text{ 或 } \\ K_d > 0 \end{cases} \quad \begin{cases} K_p < -\frac{1}{K_1} \\ K_d < 0 \end{cases}$

[0059] 图 4 示出了根据本发明实施例的 HTTP 流媒体的速率自适装置的示意图, 包括 :

[0060] 下载模块 10, 用于使客户端在传输层采用 TCP 协议, 在应用层采用 HTTP 协议, 从服务端获取选择码率的流媒体;

[0061] 选择模块 20, 用于选择码率, 使得在当前的下载速率下用于缓存流媒体的缓冲区的缓存数据量保持在设定的范围内。

[0062] 本装置在充分利用带宽的同时, 保证视频播放的连续性以及平滑性, 提高了用户体验。

[0063] 优选地, 本装置还包括 :

[0064] 编码模块, 用于将视频内容编码成具有不同码率的多个视频文件;

[0065] 分段模块, 用于将每个视频文件按时间先后切分成设定长度的多个视频片段, 并作为流媒体存储在服务端;

[0066] 选择模块用于在客户端下载完第 $k-1$ 个视频片段的流媒体之后, 根据当前的下载速率和缓冲区的数据量确定相应的码率;

[0067] 下载模块用于使客户端从服务端获取具有相应码率的第 k 个视频片段的流媒体。

[0068] 显然,本领域的技术人员应该明白,上述的本发明的各模块或各步骤可以用通用的计算装置来实现,它们可以集中在单个的计算装置上,或者分布在多个计算装置所组成的网络上,可选地,它们可以用计算装置可执行的程序代码来实现,从而,可以将它们存储在存储装置中由计算装置来执行,或者将它们分别制作成各个集成电路模块,或者将它们中的多个模块或步骤制作成单个集成电路模块来实现。这样,本发明不限制于任何特定的硬件和软件结合。

[0069] 以上所述仅为本发明的优选实施例而已,并不用于限制本发明,对于本领域的技术人员来说,本发明可以有各种更改和变化。凡在本发明的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

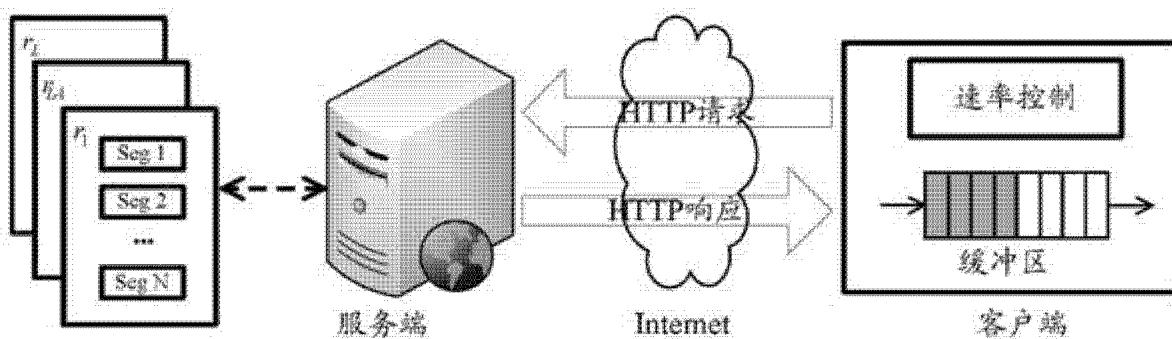


图 1

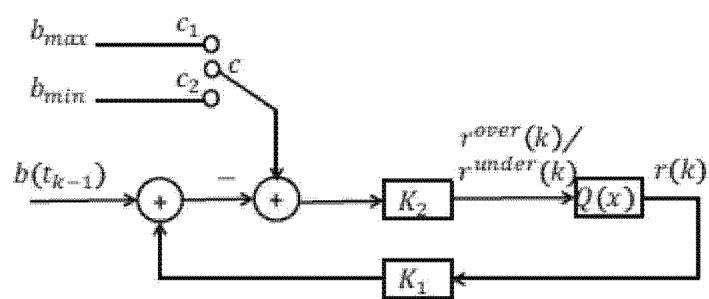


图 2

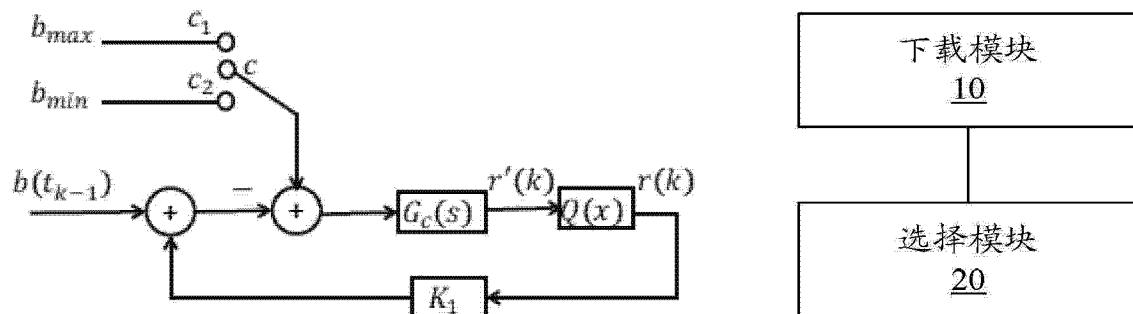


图 3

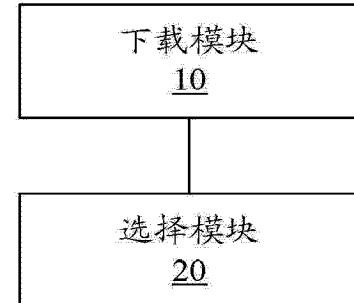


图 4